resolwe-bio

Release 55.1.1.dev2+gc704b3e

Genialis, Inc.

Jan 24, 2024

CONTENTS

Bioinformatics pipelines for the Resolwe dataflow package for Django framework.

CHAPTER

ONE

CONTENTS

1.1 Writing processes

A tutorial about writing bioinformatics pipelines (process is a step in the pipeline) is in Resolwe SDK for Python documentation.

1.1.1 Tools

Frequently, it is very useful to write a custom script in Python or R to perform a certain task in process' algorithm. For an example, see the tutorial in Resolwe SDK for Python documentation.

Custom scripts needed by processes included with Resolwe Bioinformatics are located in the resolwe_bio/tools directory.

Note: A Resolwe's `Flow Executor` searches for tools in a Django application's `tools` directory or directories specified in the `RESOLWE_CUSTOM_TOOLS_PATHS` Django setting.

1.2 Process catalog

Resolwe Bioinformatics includes over 100 processes. They are organized in categories. The type tree will help process developers with pipeline design. For process details browse process definitions.

1.2.1 Processes by category

Atac-seq

- ATAC-Seq

Align

- BWA MEM2
- Bowtie (Dicty)
- Bowtie2
- HISAT2
- BWA MEM
- BWA SW
- BWA ALN
- STAR

Bam processing

- Bam split
- Bedtools bamtobed
- Calculate coverage (bamCoverage)
- Bamplot
- Bamliquidator
- Annotate novel splice junctions (regtools)
- Bamclipper
- MarkDuplicates
- alignmentSieve

Chip-seq

- MACS 1.4
- Pre-peakcall QC
- ChIP-Seq (Peak Score)
- ChIP-Seq (Gene Score)
- ChIP-seq (MACS2)
- ChIP-seq (MACS2-ROSE2)
- ROSE2
- MACS 2.0
- ChipQC

Differential expression

- edgeR
- Cuffdiff 2.2
- DESeq2
- Differential expression of shRNA

Enrichment and clustering

- PCA
- Find similar genes
- Hierarchical clustering of time courses

Fastq processing

- UMI-tools dedup
- Subsample FASTQ (single-end)
- Subsample FASTQ (paired-end)
- Merge FASTQ (single-end)
- Merge FASTQ (paired-end)
- Reverse complement FASTQ (single-end)
- Reverse complement FASTQ (paired-end)
- Cutadapt (single-end)
- Cutadapt (paired-end)
- Trimmomatic (single-end)
- Trimmomatic (paired-end)
- Trim Galore (paired-end)
- Cutadapt (3' mRNA-seq, single-end)
- Cutadapt (Corall RNA-Seq, single-end)
- Cutadapt (Corall RNA-Seq, paired-end)
- BBDuk (single-end)
- BBDuk (paired-end)

Gatk

- GATK SelectVariants (multi-sample)
- GATK SelectVariants (single-sample)
- GATK VariantFiltration (multi-sample)
- GATK VariantFiltration (single-sample)
- GATK GenomicsDBImport
- GATK4 (HaplotypeCaller)
- GATK filter variants (VQSR)
- GATK HaplotypeCaller (GVCF)
- GATK MergeVcfs
- GATK GenotypeGVCFs
- GATK VariantsToTable
- GATK refine variants
- BaseQualityScoreRecalibrator
- RNA-seq variant calling preprocess
- GATK SplitNCigarReads

Genome index

- Bowtie2 genome index
- BWA genome index
- Bowtie genome index
- BWA-MEM2 genome index
- HISAT2 genome index
- STAR genome index

Import

- GEO import
- Upload microarray expression (unmapped)
- VCF file
- GAF file
- GFF3 file
- GTF file
- SAM header
- OBO file
- Mappability info

- Cuffquant results
- Reads (QSEQ multiplexed, single)
- Reads (QSEQ multiplexed, paired)
- Differential Expression (table)
- BED file
- BAM file
- BAM file and index
- Secondary hybrid BAM file
- Expression time course
- IDAT file
- ML-ready expression
- Ensembl-VEP cache directory
- BaseSpace file
- Metadata table (one-to-one)
- Metadata table
- FASTQ file (single-end)
- FASTQ file (paired-end)
- Convert files to reads (single-end)
- Convert files to reads (paired-end)
- Gene set
- Gene set (create)
- Gene set (create from Venn diagram)
- BEDPE file
- SRA data
- SRA data (single-end)
- SRA data (paired-end)
- Upload proteomics sample
- Upload proteomics sample set
- Reads (scRNA 10x)
- BWA-MEM2 index files
- Single cell BAM file and index
- FASTA file

Other

- Prepare GEO - ChIP-Seq
- Prepare GEO - RNA-Seq
- Convert GFF3 to GTF
- Archive samples

Picard

- Picard AlignmentSummary
- Picard CollectRrbsMetrics
- Picard InsertSizeMetrics
- Picard WGS Metrics

Pipeline

- Cut & Run
- WGBS (single-end)
- WGBS (paired-end)
- miRNA pipeline
- RNA-Seq (Cuffquant)
- Cutadapt - STAR - StringTie (Corall, single-end)
- Cutadapt - STAR - StringTie (Corall, paired-end)
- WGS (paired-end) analysis
- Whole exome sequencing (WES) analysis
- shRNA quantification
- Chemical Mutagenesis
- MACS2 - ROSE2
- MACS2
- Subsample FASTQ and BWA Aln (single-end)
- Subsample FASTQ and BWA Aln (paired-end)
- STAR-based gene quantification workflow
- BBDuk - Salmon - QC
- BBDuk - STAR - featureCounts - QC
- Beta Cut & Run workflow
- RNA-seq Variant Calling Workflow
- QuantSeq workflow
- WGS analysis (GVCF)

Qc

- Spike-ins quality control
- QoRTs QC
- MultiQC
- RNA-SeQC

Quantify

- Detect library strandedness
- Merge Expressions (ETC)
- Quantify shRNA species using bowtie2
- Cuffnorm
- Expression aggregator
- Expression matrix
- Cufflinks 2.2
- Dictyostelium expressions
- Salmon Index
- Cuffquant 2.2
- Mappability
- Expression Time Course
- Cuffmerge

Samtools

- Samtools idxstats
- Samtools view
- Samtools coverage (multi-sample)
- Samtools coverage (single-sample)
- Samtools bedcov
- Samtools fastq (paired-end)

Test

- Abstract alignment process
- Abstract expression process
- Abstract differential expression process
- Test basic fields
- Test sleep progress
- Test hidden inputs
- Test disabled inputs
- Test select controler
- Abstract annotation process
- Abstract bed process

Vep

- Ensembl Variant Effect Predictor

Wgbs

- HMR
- methcounts
- WALT
- WALT genome index
- Bisulfite conversion rate

Wgs

- Variant calling (CheMut)
- Variant filtering (CheMut)
- snpEff (General variant annotation) (multi-sample)
- snpEff (General variant annotation) (single-sample)
- WGS preprocess data with bwa-mem2

Xenograft processing

- Xengsort index
- Xengsort classify

Scrna-seq

- Cell Ranger Mkref
- Cell Ranger Count

Uncategorized

- Map microarray probes

1.2.2 Type tree

Process types are listed alphabetically. Next to each type is a list of processes of that type. Types are hierarchical, with levels of hierarchy separated by colon ":". The hierarchy defines what is accepted on inputs. For instance, Expression (Cuffnorm) process' input is `data:alignment:bam`. This means it also accepts all subtypes (*e.g.,* `data:alignment:bam:bwasw`, `data:alignment:bam:bowtie1` and `data:alignment:bam:tophat`). We encourage the use of existing types in custom processes.

- `data:aggregator:expression` - Expression aggregator
- `data:alignment` - Abstract alignment process
- `data:alignment:bam:bamclipped:` - Bamclipper
- `data:alignment:bam:bowtie1` - Bowtie (Dicty)
- `data:alignment:bam:bowtie2` - Bowtie2
- `data:alignment:bam:bqsr:` - BaseQualityScoreRecalibrator
- `data:alignment:bam:bwaaln` - BWA ALN
- `data:alignment:bam:bwamem` - BWA MEM
- `data:alignment:bam:bwamem2` - BWA MEM2
- `data:alignment:bam:bwasw` - BWA SW
- `data:alignment:bam:hisat2` - HISAT2
- `data:alignment:bam:markduplicate:` - MarkDuplicates
- `data:alignment:bam:primary` - Bam split
- `data:alignment:bam:rnaseqvc:` - RNA-seq variant calling preprocess
- `data:alignment:bam:samtools:` - Samtools view
- `data:alignment:bam:scseq:` - Single cell BAM file and index
- `data:alignment:bam:secondary` - Secondary hybrid BAM file
- `data:alignment:bam:sieve:` - alignmentSieve
- `data:alignment:bam:splitncigar:` - GATK SplitNCigarReads
- `data:alignment:bam:star:` - STAR

- `data:alignment:bam:umitools:dedup:` - UMI-tools dedup
- `data:alignment:bam:upload` - BAM file, BAM file and index
- `data:alignment:bam:walt` - WALT
- `data:alignment:bam:wgsbwa2:` - WGS preprocess data with bwa-mem2
- `data:annotation` - Abstract annotation process
- `data:annotation:cuffmerge` - Cuffmerge
- `data:annotation:gff3` - GFF3 file
- `data:annotation:gtf` - Convert GFF3 to GTF, GTF file
- `data:archive:samples` - Archive samples
- `data:bam:plot:bamliquidator` - Bamliquidator
- `data:bam:plot:bamplot` - Bamplot
- `data:bed` - Abstract bed process, BED file
- `data:bedcov:` - Samtools bedcov
- `data:bedpe:` - BEDPE file, Bedtools bamtobed
- `data:chipqc:` - ChipQC
- `data:chipseq:batch:macs2` - ChIP-seq (MACS2), ChIP-seq (MACS2-ROSE2)
- `data:chipseq:callpeak:macs14` - MACS 1.4
- `data:chipseq:callpeak:macs2:` - MACS 2.0
- `data:chipseq:genescore` - ChIP-Seq (Gene Score)
- `data:chipseq:peakscore` - ChIP-Seq (Peak Score)
- `data:chipseq:rose2:` - ROSE2
- `data:clustering:hierarchical:etc:` - Hierarchical clustering of time courses
- `data:coverage:bigwig:` - Calculate coverage (bamCoverage)
- `data:cufflinks:cufflinks` - Cufflinks 2.2
- `data:cufflinks:cuffquant` - Cuffquant 2.2, Cuffquant results
- `data:cuffnorm` - Cuffnorm
- `data:differentialexpression` - Abstract differential expression process
- `data:differentialexpression:cuffdiff:` - Cuffdiff 2.2
- `data:differentialexpression:deseq2:` - DESeq2
- `data:differentialexpression:edger:` - edgeR
- `data:differentialexpression:upload` - Differential Expression (table)
- `data:etc` - Expression Time Course, Expression time course
- `data:expression` - Abstract expression process
- `data:expression:polya` - Dictyostelium expressions
- `data:expression:shrna2quant` - Quantify shRNA species using bowtie2
- `data:expressionset` - Expression matrix

- `data:expressionset:etc` - Merge Expressions (ETC)
- `data:file:` - BaseSpace file
- `data:gaf:2:0` - GAF file
- `data:geneset:` - Gene set, Gene set (create)
- `data:geneset:venn:` - Gene set (create from Venn diagram)
- `data:genomeindex:10x:` - Cell Ranger Mkref
- `data:genomicsdb:` - GATK GenomicsDBImport
- `data:geo:` - GEO import
- `data:index:bowtie2:` - Bowtie2 genome index
- `data:index:bowtie:` - Bowtie genome index
- `data:index:bwa:` - BWA genome index
- `data:index:bwamem2:` - BWA-MEM2 genome index, BWA-MEM2 index files
- `data:index:hisat2:` - HISAT2 genome index
- `data:index:salmon` - Salmon Index
- `data:index:star:` - STAR genome index
- `data:index:walt:` - WALT genome index
- `data:junctions:regtools` - Annotate novel splice junctions (regtools)
- `data:mappability:bcm` - Mappability, Mappability info
- `data:mergereads:paired:` - Merge FASTQ (paired-end)
- `data:mergereads:single:` - Merge FASTQ (single-end)
- `data:metadata:` - Metadata table
- `data:metadata:unique:` - Metadata table (one-to-one)
- `data:methylationarray:idat:` - IDAT file
- `data:microarray:mapping:` - Map microarray probes
- `data:microarray:normalized:` - Upload microarray expression (unmapped)
- `data:ml:table:expressions:` - ML-ready expression
- `data:multiplexed:qseq:paired` - Reads (QSEQ multiplexed, paired)
- `data:multiplexed:qseq:single` - Reads (QSEQ multiplexed, single)
- `data:multiqc:` - MultiQC
- `data:ontology:obo` - OBO file
- `data:other:geo:chipseq` - Prepare GEO - ChIP-Seq
- `data:other:geo:rnaseq` - Prepare GEO - RNA-Seq
- `data:pca` - PCA
- `data:picard:insert:` - Picard InsertSizeMetrics
- `data:picard:rrbs:` - Picard CollectRrbsMetrics
- `data:picard:summary:` - Picard AlignmentSummary

- `data:picard:wgsmetrics:` - Picard WGS Metrics
- `data:prepeakqc` - Pre-peakcall QC
- `data:proteomics:massspectrometry:` - Upload proteomics sample
- `data:proteomics:sampleset:` - Upload proteomics sample set
- `data:qorts:qc:` - QoRTs QC
- `data:reads:fastq:paired:` - Convert files to reads (paired-end), FASTQ file (paired-end), SRA data (paired-end)
- `data:reads:fastq:paired:bamtofastq:` - Samtools fastq (paired-end)
- `data:reads:fastq:paired:bbduk:` - BBDuk (paired-end)
- `data:reads:fastq:paired:cutadapt` - Cutadapt (paired-end)
- `data:reads:fastq:paired:cutadapt:` - Cutadapt (Corall RNA-Seq, paired-end)
- `data:reads:fastq:paired:seqtk:` - Reverse complement FASTQ (paired-end), Subsample FASTQ (paired-end)
- `data:reads:fastq:paired:trimgalore:` - Trim Galore (paired-end)
- `data:reads:fastq:paired:trimmomatic` - Trimmomatic (paired-end)
- `data:reads:fastq:single:` - Convert files to reads (single-end), FASTQ file (single-end), SRA data (single-end)
- `data:reads:fastq:single:bbduk:` - BBDuk (single-end)
- `data:reads:fastq:single:cutadapt` - Cutadapt (single-end)
- `data:reads:fastq:single:cutadapt:` - Cutadapt (3' mRNA-seq, single-end), Cutadapt (Corall RNA-Seq, single-end)
- `data:reads:fastq:single:seqtk:` - Reverse complement FASTQ (single-end), Subsample FASTQ (single-end)
- `data:reads:fastq:single:trimmomatic` - Trimmomatic (single-end)
- `data:rnaseqc:qc:` - RNA-SeQC
- `data:sam:header` - SAM header
- `data:samtools:idxstats:` - Samtools idxstats
- `data:samtoolscoverage:multi:` - Samtools coverage (multi-sample)
- `data:samtoolscoverage:single:` - Samtools coverage (single-sample)
- `data:scexpression:10x:` - Cell Ranger Count
- `data:screads:10x:` - Reads (scRNA 10x)
- `data:seq:nucleotide:` - FASTA file
- `data:shrna:differentialexpression:` - Differential expression of shRNA
- `data:similarexpression:` - Find similar genes
- `data:spikeins` - Spike-ins quality control
- `data:sra:` - SRA data
- `data:strandedness` - Detect library strandedness
- `data:test:disabled` - Test disabled inputs

- `data:test:fields` - Test basic fields
- `data:test:hidden` - Test hidden inputs
- `data:test:result` - Test select controler, Test sleep progress
- `data:variants:gvcf:` - GATK HaplotypeCaller (GVCF)
- `data:variants:vcf` - VCF file
- `data:variants:vcf:chemut:` - Variant calling (CheMut)
- `data:variants:vcf:filtering:` - Variant filtering (CheMut)
- `data:variants:vcf:gatk:hc:` - GATK4 (HaplotypeCaller)
- `data:variants:vcf:genotypegvcfs:` - GATK GenotypeGVCFs
- `data:variants:vcf:mergevcfs:` - GATK MergeVcfs
- `data:variants:vcf:refinevariants:` - GATK refine variants
- `data:variants:vcf:selectvariants:` - GATK SelectVariants (multi-sample)
- `data:variants:vcf:selectvariants:single:` - GATK SelectVariants (single-sample)
- `data:variants:vcf:snpeff:` - snpEff (General variant annotation) (multi-sample)
- `data:variants:vcf:snpeff:single:` - snpEff (General variant annotation) (single-sample)
- `data:variants:vcf:variantfiltration:` - GATK VariantFiltration (multi-sample)
- `data:variants:vcf:variantfiltration:single:` - GATK VariantFiltration (single-sample)
- `data:variants:vcf:vep:` - Ensembl Variant Effect Predictor
- `data:variants:vcf:vqsr:` - GATK filter variants (VQSR)
- `data:variantstable:` - GATK VariantsToTable
- `data:vep:cache:` - Ensembl-VEP cache directory
- `data:wgbs:bsrate:` - Bisulfite conversion rate
- `data:wgbs:hmr` - HMR
- `data:wgbs:methcounts` - methcounts
- `data:workflow:atacseq` - ATAC-Seq
- `data:workflow:chemut` - Chemical Mutagenesis
- `data:workflow:chipseq:macs2rose2` - MACS2, MACS2 - ROSE2
- `data:workflow:chipseq:seqtkbwaaln` - Subsample FASTQ and BWA Aln (paired-end), Subsample FASTQ and BWA Aln (single-end)
- `data:workflow:cutnrun` - Cut & Run
- `data:workflow:cutnrun:` - Beta Cut & Run workflow
- `data:workflow:mirna` - miRNA pipeline
- `data:workflow:quant:featurecounts:` - QuantSeq workflow
- `data:workflow:rnaseq:corall` - Cutadapt - STAR - StringTie (Corall, paired-end), Cutadapt - STAR - StringTie (Corall, single-end)
- `data:workflow:rnaseq:cuffquant` - RNA-Seq (Cuffquant)
- `data:workflow:rnaseq:featurecounts:qc:` - BBDuk - STAR - featureCounts - QC

- `data:workflow:rnaseq:salmon:` - BBDuk - Salmon - QC
- `data:workflow:rnaseq:star:qc:` - STAR-based gene quantification workflow
- `data:workflow:rnaseq:variants:` - RNA-seq Variant Calling Workflow
- `data:workflow:trimalquant` - shRNA quantification
- `data:workflow:wes` - Whole exome sequencing (WES) analysis
- `data:workflow:wgbs` - WGBS (paired-end), WGBS (single-end)
- `data:workflow:wgs` - WGS (paired-end) analysis
- `data:workflow:wgs:gvcf:` - WGS analysis (GVCF)
- `data:xengsort:classification:` - Xengsort classify
- `data:xengsort:index:` - Xengsort index

1.2.3 Process definitions

ATAC-Seq

data:workflow:atacseqworkflow-atac-seq (*data:reads:fastq* **reads**, *data:index:bowtie2* **genome**, *data:bed* **promoter**, *basic:string* **mode**, *basic:string* **speed**, *basic:boolean* **use_se**, *basic:boolean* **discordantly**, *basic:boolean* **rep_se**, *basic:integer* **minins**, *basic:integer* **maxins**, *basic:integer* **trim_5**, *basic:integer* **trim_3**, *basic:integer* **trim_iter**, *basic:integer* **trim_nucl**, *basic:string* **rep_mode**, *basic:integer* **k_reports**, *basic:integer* **q_threshold**, *basic:integer* **n_sub**, *basic:boolean* **tn5**, *basic:integer* **shift**, *basic:boolean* **tagalign**, *basic:string* **duplicates**, *basic:string* **duplicates_prepeak**, *basic:decimal* **qvalue**, *basic:decimal* **pvalue**, *basic:decimal* **pvalue_prepeak**, *basic:integer* **cap_num**, *basic:integer* **mfold_lower**, *basic:integer* **mfold_upper**, *basic:integer* **slocal**, *basic:integer* **llocal**, *basic:integer* **extsize**, *basic:integer* **shift**, *basic:integer* **band_width**, *basic:boolean* **nolambda**, *basic:boolean* **fix_bimodal**, *basic:boolean* **nomodel**, *basic:boolean* **nomodel_prepeak**, *basic:boolean* **down_sample**, *basic:boolean* **bedgraph**, *basic:boolean* **spmr**, *basic:boolean* **call_summits**, *basic:boolean* **broad**, *basic:decimal* **broad_cutoff**) [Source: v3.1.1]

This ATAC-seq pipeline closely follows the official ENCODE DCC pipeline. It is comprised of three steps; alignment, pre-peakcall QC, and calling peaks (with post-peakcall QC).

First, reads are aligned to a genome using [Bowtie2](http://bowtie-bio.sourceforge.net/index.shtml) aligner. Next, pre-peakcall QC metrics are calculated. QC report contains ENCODE 3 proposed QC metrics – [NRF](https://www.encodeproject.org/data-standards/terms/), [PBC bottlenecking coefficients, NSC, and RSC](https://genome.ucsc.edu/ENCODE/qualityMetrics.html#chipSeq). Finally, the peaks are called using [MACS2](https://github.com/taoliu/MACS/). The post-peakcall QC report includes additional QC metrics – number of peaks, fraction of reads in peaks (FRiP), number of reads in peaks, and if promoter regions BED file is provided, number of reads in promoter regions, fraction of reads in promoter regions, number of peaks in promoter regions, and fraction of reads in promoter regions.

Input arguments reads

label
Select sample(s)

type
`data:reads:fastq`

genome

label
Genome

type
`data:index:bowtie2`

promoter

label
Promoter regions BED file

type
`data:bed`

description
BED file containing promoter regions (TSS+-1000 bp for example). Needed to get the number of peaks and reads mapped to promoter regions.

required
False

alignment.mode

label
Alignment mode

type
`basic:string`

description
End to end: Bowtie 2 requires that the entire read align from one end to the other, without any trimming (or "soft clipping") of characters from either end. Local: Bowtie 2 does not require that the entire read align from one end to the other. Rather, some characters may be omitted ("soft clipped") from the ends in order to achieve the greatest possible alignment score.

default
`--local`

choices

- end to end mode: `--end-to-end`
- local: `--local`

alignment.speed

label
Speed vs. Sensitivity

type
`basic:string`

default
`--sensitive`

choices

- Very fast: `--very-fast`
- Fast: `--fast`
- Sensitive: `--sensitive`
- Very sensitive: `--very-sensitive`

alignment.PE_options.use_se

label
Map as single-ended (for paired-end reads only)

type
`basic:boolean`

description
If this option is selected paired-end reads will be mapped as single-ended and other paired-end options are ignored.

default
`False`

alignment.PE_options.discordantly

label
Report discordantly matched read

type
`basic:boolean`

description
If both mates have unique alignments, but the alignments do not match paired-end expectations (orientation and relative distance) then alignment will be reported. Useful for detecting structural variations.

default
`True`

alignment.PE_options.rep_se

label
Report single ended

type
`basic:boolean`

description
If paired alignment can not be found Bowtie2 tries to find alignments for the individual mates.

default
`True`

alignment.PE_options.minins

label
Minimal distance

type
`basic:integer`

description
The minimum fragment length for valid paired-end alignments. 0 imposes no minimum.

default
`0`

alignment.PE_options.maxins

label
Maximal distance

type
`basic:integer`

description
The maximum fragment length for valid paired-end alignments.

default
`2000`

alignment.start_trimming.trim_5

label
Bases to trim from 5'

type
`basic:integer`

description
Number of bases to trim from from 5' (left) end of each read before alignment.

default
`0`

alignment.start_trimming.trim_3

label
Bases to trim from 3'

type
`basic:integer`

description
Number of bases to trim from from 3' (right) end of each read before alignment

default
`0`

alignment.trimming.trim_iter

label
Iterations

type
`basic:integer`

description
Number of iterations.

default
`0`

alignment.trimming.trim_nucl

label
Bases to trim

type
`basic:integer`

description
Number of bases to trim from 3' end in each iteration.

default
`2`

alignment.reporting.rep_mode

label
Report mode

type
`basic:string`

description
Default mode: search for multiple alignments, report the best one; -k mode: search for one or more alignments, report each; -a mode: search for and report all alignments

default
`def`

choices

- Default mode: `def`
- -k mode: `k`
- -a mode (very slow): `a`

alignment.reporting.k_reports

label
Number of reports (for -k mode only)

type
`basic:integer`

description
Searches for at most X distinct, valid alignments for each read. The search terminates when it can't find more distinct valid alignments, or when it finds X, whichever happens first.

default
`5`

prepeakqc_settings.q_threshold

label
Quality filtering threshold

type
`basic:integer`

default
`30`

prepeakqc_settings.n_sub

label
Number of reads to subsample

type
`basic:integer`

default
`25000000`

prepeakqc_settings.tn5

label
Tn5 shifting

type
`basic:boolean`

description
Tn5 transposon shifting. Shift reads on "+" strand by 4 bp and reads on "-" strand by 5 bp.

default
`True`

prepeakqc_settings.shift

label
User-defined cross-correlation peak strandshift

type
`basic:integer`

description
If defined, SPP tool will not try to estimate fragment length but will use the given value as fragment length.

default
`0`

settings.tagalign

label
Use tagAlign files

type
`basic:boolean`

description
Use filtered tagAlign files as case (treatment) and control (background) samples. If extsize parameter is not set, run MACS using input's estimated fragment length.

default
`True`

settings.duplicates

label
Number of duplicates

type
`basic:string`

description
It controls the MACS behavior towards duplicate tags at the exact same location – the same coordination and the same strand. The 'auto' option makes MACS calculate the maximum tags at the exact same location based on binomal distribution using 1e-5 as pvalue cutoff and the 'all' option keeps all the tags. If an integer is given, at most this number of tags will be kept at the same location. The default is to keep one tag at the same location.

required
False

hidden
settings.tagalign

choices

- 1: `1`
- auto: `auto`
- all: `all`

settings.duplicates_prepeak

label
Number of duplicates

type
`basic:string`

description
It controls the MACS behavior towards duplicate tags at the exact same location – the same coordination and the same strand. The 'auto' option makes MACS calculate the maximum tags at the exact same location based on binomal distribution using 1e-5 as pvalue cutoff and the 'all' option keeps all the tags. If an integer is given, at most this number of tags will be kept at the same location. The default is to keep one tag at the same location.

required
False

hidden
!settings.tagalign

default
`all`

choices

- 1: `1`
- auto: `auto`
- all: `all`

settings.qvalue

label
Q-value cutoff

type
`basic:decimal`

description
The q-value (minimum FDR) cutoff to call significant regions. Q-values are calculated from p-values using Benjamini-Hochberg procedure.

required
False

disabled
settings.pvalue && settings.pvalue_prepeak

settings.pvalue

label
P-value cutoff

type
`basic:decimal`

description
The p-value cutoff. If specified, MACS2 will use p-value instead of q-value cutoff.

required
False

disabled
settings.qvalue

hidden
settings.tagalign

settings.pvalue_prepeak

label
P-value cutoff

type
`basic:decimal`

description
The p-value cutoff. If specified, MACS2 will use p-value instead of q-value cutoff.

disabled
settings.qvalue

hidden
!settings.tagalign || settings.qvalue

default
`0.01`

settings.cap_num

label
Cap number of peaks by taking top N peaks

type
`basic:integer`

description
To keep all peaks set value to 0.

disabled
settings.broad

default
`300000`

settings.mfold_lower

label
MFOLD range (lower limit)

type
`basic:integer`

description
This parameter is used to select the regions within MFOLD range of high-confidence enrichment ratio against background to build model. The regions must be lower than upper limit, and higher than the lower limit of fold enrichment. DEFAULT:10,30 means using all regions not too low (>10)

and not too high (<30) to build paired-peaks model. If MACS can not find more than 100 regions to build model, it will use the –extsize parameter to continue the peak detection ONLY if –fix-bimodal is set.

required
False

settings.mfold_upper

label
MFOLD range (upper limit)

type
`basic:integer`

description
This parameter is used to select the regions within MFOLD range of high-confidence enrichment ratio against background to build model. The regions must be lower than upper limit, and higher than the lower limit of fold enrichment. DEFAULT:10,30 means using all regions not too low (>10) and not too high (<30) to build paired-peaks model. If MACS can not find more than 100 regions to build model, it will use the –extsize parameter to continue the peak detection ONLY if –fix-bimodal is set.

required
False

settings.slocal

label
Small local region

type
`basic:integer`

description
Slocal and llocal parameters control which two levels of regions will be checked around the peak regions to calculate the maximum lambda as local lambda. By default, MACS considers 1000 bp for small local region (–slocal), and 10000 bp for large local region (–llocal) which captures the bias from a long range effect like an open chromatin domain. You can tweak these according to your project. Remember that if the region is set too small, a sharp spike in the input data may kill the significant peak.

required
False

settings.llocal

label
Large local region

type
`basic:integer`

description
Slocal and llocal parameters control which two levels of regions will be checked around the peak regions to calculate the maximum lambda as local lambda. By default, MACS considers 1000 bp for small local region (–slocal), and 10000 bp for large local region (–llocal) which captures the bias from a long range effect like an open chromatin domain. You can tweak these according to your project. Remember that if the region is set too small, a sharp spike in the input data may kill the significant peak.

required
False

settings.extsize

label
extsize

type
`basic:integer`

description
While '–nomodel' is set, MACS uses this parameter to extend reads in 5'->3' direction to fix-sized fragments. For example, if the size of binding region for your transcription factor is 200 bp, and you want to bypass the model building by MACS, this parameter can be set as 200. This option is only valid when –nomodel is set or when MACS fails to build model and –fix-bimodal is on.

default
`150`

settings.shift

label
Shift

type
`basic:integer`

description
Note, this is NOT the legacy –shiftsize option which is replaced by –extsize! You can set an arbitrary shift in bp here. Please Use discretion while setting it other than default value (0). When –nomodel is set, MACS will use this value to move cutting ends (5') then apply –extsize from 5' to 3' direction to extend them to fragments. When this value is negative, ends will be moved toward 3'->5' direction, otherwise 5'->3' direction. Recommended to keep it as default 0 for ChIP-Seq datasets, or -1 * half of EXTSIZE together with –extsize option for detecting enriched cutting loci such as certain DNAseI-Seq datasets. Note, you can't set values other than 0 if format is BAMPE for paired-end data. Default is 0.

default
`-75`

settings.band_width

label
Band width

type
`basic:integer`

description
The band width which is used to scan the genome ONLY for model building. You can set this parameter as the sonication fragment size expected from wet experiment. The previous side effect on the peak detection process has been removed. So this parameter only affects the model building.

required
False

settings.nolambda

label
Use backgroud lambda as local lambda

type
`basic:boolean`

description
With this flag on, MACS will use the background lambda as local lambda. This means MACS will not consider the local bias at peak candidate regions.

default
`False`

settings.fix_bimodal

label
Turn on the auto paired-peak model process

type
`basic:boolean`

description
Turn on the auto paired-peak model process. If it's set, when MACS failed to build paired model, it will use the nomodel settings, the '–extsize' parameter to extend each tag. If set, MACS will be terminated if paired-peak model has failed.

default
`False`

settings.nomodel

label
Bypass building the shifting model

type
`basic:boolean`

description
While on, MACS will bypass building the shifting model.

hidden
settings.tagalign

default
`False`

settings.nomodel_prepeak

label
Bypass building the shifting model

type
`basic:boolean`

description
While on, MACS will bypass building the shifting model.

hidden
!settings.tagalign

default
`True`

settings.down_sample

label
Down-sample

type
`basic:boolean`

description
When set to true, random sampling method will scale down the bigger sample. By default, MACS uses linear scaling. This option will make the results unstable and irreproducible since each time, random reads would be selected, especially the numbers (pileup, pvalue, qvalue) would change.

default
`False`

settings.bedgraph

label
Save fragment pileup and control lambda

type
`basic:boolean`

description
If this flag is on, MACS will store the fragment pileup, control lambda, -log10pvalue and -log10qvalue scores in bedGraph files. The bedGraph files will be stored in current directory named NAME+'_treat_pileup.bdg' for treatment data, NAME+'_control_lambda.bdg' for local lambda values from control, NAME+'_treat_pvalue.bdg' for Poisson pvalue scores (in -log10(pvalue) form), and NAME+'_treat_qvalue.bdg' for q-value scores from Benjamini-Hochberg-Yekutieli procedure.

default
`True`

settings.spmr

label
Save signal per million reads for fragment pileup profiles

type
`basic:boolean`

disabled
settings.bedgraph === false

default
`True`

settings.call_summits

label
Call summits

type
`basic:boolean`

description
MACS will now reanalyze the shape of signal profile (p or q-score depending on cutoff setting) to deconvolve subpeaks within each peak called from general procedure. It's highly recommended to detect adjacent binding events. While used, the output subpeaks of a big peak region will have the same peak boundaries, and different scores and peak summit positions.

default
`True`

settings.broad

label
Composite broad regions

type
`basic:boolean`

description
When this flag is on, MACS will try to composite broad regions in BED12 (a gene-model-like format) by putting nearby highly enriched regions into a broad region with loose cutoff. The broad region is controlled by another cutoff through –broad-cutoff. The maximum length of broad region length is 4 times of d from MACS.

disabled
settings.call_summits === true

default
`False`

settings.broad_cutoff

label
Broad cutoff

type
`basic:decimal`

description
Cutoff for broad region. This option is not available unless –broad is set. If -p is set, this is a p-value cutoff, otherwise, it's a q-value cutoff. DEFAULT = 0.1

required
False

disabled
settings.call_summits === true || settings.broad !== true

Output results

Abstract alignment process

`data:alignmentabstract-alignment ()`[Source: v1.0.1]

Input arguments

Output results bam

label
Alignment file

type
`basic:file`

bai

label
Alignment index BAI

type
`basic:file`

species

label
Species

type
`basic:string`

build

label
Build

type
`basic:string`

Abstract annotation process

`data:annotationabstract-annotation ()`[Source: v1.0.1]

Input arguments

Output results annot

label
Uploaded file

type
`basic:file`

source

label
Gene ID source

type
`basic:string`

species

label
Species

type
`basic:string`

build

label
Build

type
`basic:string`

Abstract bed process

`data:bedabstract-bed ()`[Source: v1.0.2]

Input arguments

Output results bed

label
BED

type
`basic:file`

species

label
Species

type
`basic:string`

build

label
Build

type
`basic:string`

Abstract differential expression process

`data:differentialexpressionabstract-differentialexpression ()`[Source: v1.0.1]

Input arguments

Output results raw

label
Differential expression (gene level)

type
`basic:file`

de_json

label
Results table (JSON)

type
`basic:json`

de_file

label
Results table (file)

type
`basic:file`

source

label
Gene ID source

type
`basic:string`

species

label
Species

type
`basic:string`

build

label
Build

type
`basic:string`

feature_type

label
Feature type

type
`basic:string`

Abstract expression process

`data:expressionabstract-expression ()`[Source: v1.0.1]

Input arguments

Output results exp

label
Normalized expression

type
`basic:file`

rc

label
Read counts

type
`basic:file`

required
False

exp_json

label
Expression (json)

type
`basic:json`

exp_type

label
Expression type

type
`basic:string`

source

label
Gene ID source

type
`basic:string`

species

label
Species

type
`basic:string`

build

label
Build

type
`basic:string`

feature_type

label
Feature type

type
`basic:string`

Annotate novel splice junctions (regtools)

`data:junctions:regtoolsregtools-junctions-annotate` (*data:seq:nucleotide* **genome**, *data:annotation:gtf* **annotation**, *data:alignment:bam:star* **alignment_star**, *data:alignment:bam* **alignment**, *data:bed* **input_bed_junctions**) [Source: v1.3.1]

Identify novel splice junctions by using regtools to annotate against a reference. The process accepts reference genome, reference genome annotation (GTF), and input with reads information (STAR aligment or reads aligned by any other aligner or junctions in BED12 format). If STAR aligner data is given as input, the process calculates BED12 file from STAR 'SJ.out.tab' file, and annotates all junctions with 'regtools junctions annotate' command. When reads are aligned by other aligner, junctions are extracted with 'regtools junctions extract' tool and then annotated with 'junction annotate' command. Third option allows user to provide directly BED12 file with junctions, which are then annotated. Finnally, annotated novel junctions are filtered in a separate output file. More information can be found in the [regtools manual](https://regtools.readthedocs.io/en/latest/).

Input arguments genome

label
Reference genome

type
`data:seq:nucleotide`

annotation

label
Reference genome annotation (GTF)

type
`data:annotation:gtf`

alignment_star

label
STAR alignment

type
`data:alignment:bam:star`

description
Splice junctions detected by STAR aligner (SJ.out.tab STAR output file). Please provide one input 'STAR alignment' or 'Alignment' by any aligner or directly 'Junctions in BED12 format'.

required
False

alignment

label
Alignment

type
`data:alignment:bam`

description
Aligned reads from which splice junctions are going to be extracted. Please provide one input 'STAR alignment' or 'Alignment' by any aligner or directly 'Junctions in BED12 format'.

required
False

input_bed_junctions

label
Junctions in BED12 format

type
`data:bed`

description
Splice junctions in BED12 format. Please provide one input 'STAR alignment' or 'Alignment' by any aligner or directly 'Junctions in BED12 format'.

required
False

Output results novel_splice_junctions

label
Table of annotated novel splice junctions

type
`basic:file`

splice_junctions

label
Table of annotated splice junctions

type
`basic:file`

novel_sj_bed

label
Novel splice junctions in BED format

type
`basic:file`

bed

label
Splice junctions in BED format

type
`basic:file`

novel_sj_bigbed_igv_ucsc

label
Novel splice junctions in BigBed format

type
`basic:file`

required
False

bigbed_igv_ucsc

label
Splice junctions in BigBed format

type
`basic:file`

required
False

novel_sj_tbi_jbrowse

label
Novel splice junctions bed tbi index for JBrowse

type
`basic:file`

tbi_jbrowse

label
Bed tbi index for JBrowse

type
`basic:file`

species

label
Species

type
`basic:string`

build

label
Build

type
`basic:string`

Archive samples

`data:archive:samplesarchive-samples` (*list:data* **data**, *list:basic:string* **fields**, *basic:boolean* **j**) [Source: v0.5.2]

Create an archive of output files. The ouput folder structure is organized by sample slug and data object's output-field names.

Input arguments data

label
Data list

type
`list:data`

fields

label
Output file fields

type
`list:basic:string`

j

label
Junk paths

type
`basic:boolean`

description
Store just names of saved files (junk the path)

default
`False`

Output results archive

label
Archive

type
`basic:file`

BAM file

`data:alignment:bam:uploadupload-bam` (*basic:file* **src**, *basic:string* **species**, *basic:string* **build**)[Source: v1.8.0]

Import a BAM file (.bam), which is the binary format for storing sequence alignment data. This format is described on the [SAM Tools web site](http://samtools.github.io/hts-specs/).

Input arguments src

label
Mapping (BAM)

type
`basic:file`

description
A mapping file in BAM format. The file will be indexed on upload, so additional BAI files are not required.

validate_regex
`\.(bam)$`

species

label
Species

type
`basic:string`

description
Species latin name.

choices

- Homo sapiens: `Homo sapiens`
- Mus musculus: `Mus musculus`
- Rattus norvegicus: `Rattus norvegicus`
- Dictyostelium discoideum: `Dictyostelium discoideum`
- Odocoileus virginianus texanus: `Odocoileus virginianus texanus`
- Solanum tuberosum: `Solanum tuberosum`

build

label
Build

type
`basic:string`

Output results bam

label
Uploaded file

type
`basic:file`

bai

label
Index BAI

type
`basic:file`

stats

label
Alignment statistics

type
`basic:file`

species

label
Species

type
`basic:string`

build

label
Build

type
`basic:string`

BAM file and index

`data:alignment:bam:uploadupload-bam-indexed` (*basic:file* **src**, *basic:file* **src2**, *basic:string* **species**, *basic:string* **build**) [Source: v1.8.0]

Import a BAM file (.bam) and BAM index (.bam.bai). BAM file is the binary format for storing sequence alignment data. This format is described on the [SAM Tools web site](http://samtools.github.io/hts-specs/).

Input arguments src

label
Mapping (BAM)

type
`basic:file`

description
A mapping file in BAM format.

validate_regex
`\.(bam)$`

src2

label
bam index (*.bam.bai file)

type
`basic:file`

description
An index file of a BAM mapping file (ending with bam.bai).

validate_regex
`\.(bam.bai)$`

species

label
Species

type
`basic:string`

description
Species latin name.

choices

- Homo sapiens: `Homo sapiens`
- Mus musculus: `Mus musculus`
- Rattus norvegicus: `Rattus norvegicus`
- Dictyostelium discoideum: `Dictyostelium discoideum`
- Odocoileus virginianus texanus: `Odocoileus virginianus texanus`
- Solanum tuberosum: `Solanum tuberosum`

build

label
Build

type
`basic:string`

Output results bam

label
Uploaded file

type
`basic:file`

bai

label
Index BAI

type
`basic:file`

stats

label
Alignment statistics

type
`basic:file`

species

label
Species

type
`basic:string`

build

label
Build

type
`basic:string`

BBDuk (paired-end)

`data:reads:fastq:paired:bbduk:bbduk-paired` (*data:reads:fastq:paired* **reads**, *basic:integer* **min_length**, *list:data:seq:nucleotide* **sequences**, *list:basic:string* **literal_sequences**, *basic:integer* **kmer_length**, *basic:boolean* **check_reverse_complements**, *basic:boolean* **mask_middle_base**, *basic:integer* **min_kmer_hits**, *basic:decimal* **min_kmer_fraction**, *basic:decimal* **min_coverage_fraction**, *basic:integer* **hamming_distance**, *basic:integer* **query_hamming_distance**, *basic:integer* **edit_distance**, *basic:integer* **hamming_distance2**, *basic:integer* **query_hamming_distance2**, *basic:integer* **edit_distance2**, *basic:boolean* **forbid_N**, *basic:boolean* **find_best_match**, *basic:boolean* **remove_if_either_bad**, *basic:boolean* **perform_error_correction**, *basic:string* **k_trim**, *basic:string* **k_mask**, *basic:boolean* **mask_fully_covered**, *basic:integer* **min_k**, *basic:string* **quality_trim**, *basic:integer* **trim_quality**, *basic:string* **quality_encoding_offset**, *basic:boolean* **ignore_bad_quality**, *basic:integer* **trim_poly_A**, *basic:decimal* **min_length_fraction**, *basic:integer* **max_length**, *basic:integer* **min_average_quality**, *basic:integer* **min_average_quality_bases**, *basic:integer* **min_base_quality**, *basic:integer* **min_consecutive_bases**, *basic:integer* **trim_pad**, *basic:boolean* **trim_by_overlap**, *basic:boolean* **strict_overlap**, *basic:integer* **min_overlap**, *basic:integer* **min_insert**, *basic:boolean* **trim_pairs_evenly**, *basic:integer* **force_trim_left**, *basic:integer* **force_trim_right**, *basic:integer* **force_trim_right2**, *basic:integer* **force_trim_mod**, *basic:integer* **restrict_left**, *basic:integer* **restrict_right**, *basic:decimal* **min_GC**, *basic:decimal* **max_GC**, *basic:integer* **maxns**, *basic:boolean* **toss_junk**, *basic:boolean* **chastity_filter**, *basic:boolean* **barcode_filter**, *list:data:seq:nucleotide* **barcode_files**, *list:basic:string* **barcode_sequences**, *basic:integer* **x_min**, *basic:integer* **y_min**, *basic:integer* **x_max**, *basic:integer* **y_max**, *basic:decimal* **entropy**, *basic:integer* **entropy_window**, *basic:integer* **entropy_k**, *basic:boolean* **entropy_mask**, *basic:integer* **min_base_frequency**, *basic:boolean* **nogroup**)[Source: v3.1.2]

Run BBDuk on paired-end reads.

BBDuk combines the most common data-quality-related trimming, filtering, and masking operations into a single high-

performance tool. It is capable of quality-trimming and filtering, adapter-trimming, contaminant-filtering via kmer matching, sequence masking, GC-filtering, length filtering, entropy-filtering, format conversion, histogram generation, subsampling, quality-score recalibration, kmer cardinality estimation, and various other operations in a single pass. See [here](https://jgi.doe.gov/data-and-tools/bbtools/bb-tools-user-guide/bbduk-guide/) for more information.

Input arguments reads

label
Reads

type
`data:reads:fastq:paired`

required
True

disabled
False

hidden
False

min_length

label
Minimum length

type
`basic:integer`

description
Reads shorter than the minimum length will be discarded after trimming.

required
True

disabled
False

hidden
False

default
`10`

reference.sequences

label
Sequences

type
`list:data:seq:nucleotide`

description
Reference sequences include adapters, contaminants, and degenerate sequences. They can be provided in a multi-sequence FASTA file or as a set of literal sequences below.

required
False

disabled
False

hidden
False

reference.literal_sequences

label
Literal sequences

type
`list:basic:string`

description
Literal sequences can be specified by inputting them one by one and pressing Enter after each sequence.

required
False

disabled
False

hidden
False

default
`[]`

processing.kmer_length

label
Kmer length

type
`basic:integer`

description
Kmer length used for finding contaminants. Contaminants shorter than kmer length will not be found. Kmer length must be at least 1.

required
True

disabled
False

hidden
False

default
`27`

processing.check_reverse_complements

label
Check reverse complements

type
`basic:boolean`

description
Look for reverse complements of kmers in addition to forward kmers.

required
True

disabled
False

hidden
False

default
`True`

processing.mask_middle_base

label
Mask the middle base of a kmer

type
`basic:boolean`

description
Treat the middle base of a kmer as a wildcard to increase sensitivity in the presence of errors.

required
True

disabled
False

hidden
False

default
`True`

processing.min_kmer_hits

label
Minimum number of kmer hits

type
`basic:integer`

description
Reads need at least this many matching kmers to be considered matching the reference.

required
True

disabled
False

hidden
False

default
1

processing.min_kmer_fraction

label
Minimum kmer fraction

type
`basic:decimal`

description
A read needs at least this fraction of its total kmers to hit a reference in order to be considered a match. If this and 'Minimum number of kmer hits' are set, the greater is used.

required
True

disabled
False

hidden
False

default
`0.0`

processing.min_coverage_fraction

label
Minimum kmer fraction

type
`basic:decimal`

description
A read needs at least this fraction of its total bases to be covered by reference kmers to be considered a match. If specified, 'Minimum coverage fraction' overrides 'Minimum number of kmer hits' and 'Minimum kmer fraction'.

required
True

disabled
False

hidden
False

default
`0.0`

processing.hamming_distance

label
Maximum Hamming distance for kmers (substitutions only)

type
`basic:integer`

description
Hamming distance i.e. the number of mismatches allowed in the kmer.

required
True

disabled
False

hidden
False

default
`0`

processing.query_hamming_distance

label
Hamming distance for query kmers

type
`basic:integer`

description
Set a hamming distance for query kmers instead of the read kmers. This makes the read processing much slower, but does not use additional memory.

required
True

disabled
False

hidden
False

default
`0`

processing.edit_distance

label
Maximum edit distance from reference kmers (substitutions and indels)

type
`basic:integer`

required
True

disabled
False

hidden
False

default
`0`

processing.hamming_distance2

label
Hamming distance for short kmers when looking for shorter kmers

type
`basic:integer`

required
True

disabled
False

hidden
False

default
`0`

processing.query_hamming_distance2

label
Hamming distance for short query kmers when looking for shorter kmers

type
`basic:integer`

required
True

disabled
False

hidden
False

default
`0`

processing.edit_distance2

label
Maximum edit distance from short reference kmers (substitutions and indels) when looking for shorter kmers

type
`basic:integer`

required
True

disabled
False

hidden
False

default
`0`

processing.forbid_N

label
Forbid matching of read kmers containing N

type
`basic:boolean`

description
By default, these will match a reference 'A' if 'Maximum Hamming distance for kmers' > 0 or 'Maximum edit distance from reference kmers' > 0, to increase sensitivity.

required
True

disabled
False

hidden
False

default
`False`

processing.find_best_match

label
Find best match

type
`basic:boolean`

description
If multiple matches, associate read with sequence sharing most kmers.

required
True

disabled
False

hidden
False

default
`True`

processing.remove_if_either_bad

label
Remove both sequences of a paired-end read, if either of them is to be removed

type
`basic:boolean`

required
True

disabled
False

hidden
False

default
`True`

processing.perform_error_correction

label
Perform error correction with BBMerge prior to kmer operations

type
`basic:boolean`

required
True

disabled
False

hidden
False

default
`False`

operations.k_trim

label
Trimming protocol to remove bases matching reference kmers from reads

type
`basic:string`

required
True

disabled
False

hidden
False

default
`f`

choices

- Don't trim: `f`
- Trim to the right: `r`
- Trim to the left: `l`

operations.k_mask

label
Symbol to replace bases matching reference kmers

type
`basic:string`

description
Allows any non-whitespace character other than t or f. Processes short kmers on both ends.

required
True

disabled
False

hidden
False

default
`f`

operations.mask_fully_covered

label
Only mask bases that are fully covered by kmers

type
`basic:boolean`

required
True

disabled
False

hidden
False

default
`False`

operations.min_k

label
Look for shorter kmers at read tips down to this length when k-trimming or masking

type
`basic:integer`

description
-1 means disabled. Enabling this will disable treating the middle base of a kmer as a wildcard to increase sensitivity in the presence of errors.

required
True

disabled
False

hidden
False

default
`-1`

operations.quality_trim

label
Trimming protocol to remove bases with quality below the minimum average region quality from read ends

type
`basic:string`

description
Performed after looking for kmers. If enabled, set also 'Average quality below which to trim region'.

required
True

disabled
False

hidden
False

default
`f`

choices

- Trim neither end: `f`
- Trim both ends: `rl`
- Trim only right end: `r`
- Trim only left end: `l`
- Use sliding window: `w`

operations.trim_quality

label
Average quality below which to trim region

type
`basic:integer`

description
Set trimming protocol to enable this parameter.

required
True

disabled
operations.quality_trim === 'f'

hidden
False

default
`6`

operations.quality_encoding_offset

label
Quality encoding offset

type
`basic:string`

description
Quality encoding offset for input FASTQ files.

required
True

disabled
False

hidden
False

default
`auto`

choices

- Sanger / Illumina 1.8+ (33): `33`
- Illumina up to 1.3+, 1.5+ (64): `64`
- Auto: `auto`

operations.ignore_bad_quality

label
Don't crash if quality values appear to be incorrect

type
`basic:boolean`

required
True

disabled
False

hidden
False

default
`False`

operations.trim_poly_A

label
Minimum length of poly-A or poly-T tails to trim on either end of reads

type
`basic:integer`

required
True

disabled
False

hidden
False

default
`0`

operations.min_length_fraction

label
Minimum length fraction

type
`basic:decimal`

description
Reads shorter than this fraction of original length after trimming will be discarded.

required
True

disabled
False

hidden
False

default
`0.0`

operations.max_length

label
Maximum length

type
`basic:integer`

description
Reads longer than this after trimming will be discarded.

required
False

disabled
False

hidden
False

operations.min_average_quality

label
Minimum average quality

type
`basic:integer`

description
Reads with average quality (after trimming) below this will be discarded.

required
True

disabled
False

hidden
False

default
`0`

operations.min_average_quality_bases

label
Number of initial bases to calculate minimum average quality from

type
`basic:integer`

description
If positive, calculate minimum average quality from this many initial bases

required
True

disabled
False

hidden
False

default
`0`

operations.min_base_quality

label
Minimum base quality below which reads are discarded after trimming

type
`basic:integer`

required
True

disabled
False

hidden
False

default
`0`

operations.min_consecutive_bases

label
Minimum number of consecutive called bases

type
`basic:integer`

required
True

disabled
False

hidden
False

default
`0`

operations.trim_pad

label
Number of bases to trim around matching kmers

type
`basic:integer`

required
True

disabled
False

hidden
False

default
`0`

operations.trim_by_overlap

label
Trim adapters based on where paired-end reads overlap

type
`basic:boolean`

required
True

disabled
False

hidden
False

default
`False`

operations.strict_overlap

label
Adjust sensitivity in 'Trim adapters based on where paired-end reads overlap' mode

type
`basic:boolean`

required
True

disabled
False

hidden
False

default
`True`

operations.min_overlap

label
Minum number of overlapping bases

type
`basic:integer`

description
Require this many bases of overlap for detection.

required
True

disabled
False

hidden
False

default
`14`

operations.min_insert

label
Minimum insert size

type
`basic:integer`

description
Require insert size of at least this for overlap. Should be reduced to 16 for small RNA sequencing.

required
True

disabled
False

hidden
False

default
`40`

operations.trim_pairs_evenly

label
Trim both sequences of paired-end reads to the minimum length of either sequence

type
`basic:boolean`

required
True

disabled
False

hidden
False

default
`False`

operations.force_trim_left

label
Position from which to trim bases to the left

type
`basic:integer`

required
True

disabled
False

hidden
False

default
`0`

operations.force_trim_right

label
Position from which to trim bases to the right

type
`basic:integer`

required
True

disabled
False

hidden
False

default
`0`

operations.force_trim_right2

label
Number of bases to trim from the right end

type
`basic:integer`

required
True

disabled
False

hidden
False

default
`0`

operations.force_trim_mod

label
Modulo to right-trim reads

type
`basic:integer`

description
Trim reads to the largest multiple of modulo.

required
True

disabled
False

hidden
False

default
`0`

operations.restrict_left

label
Number of leftmost bases to look in for kmer matches

type
`basic:integer`

required
True

disabled
False

hidden
False

default
`0`

operations.restrict_right

label
Number of rightmost bases to look in for kmer matches

type
`basic:integer`

required
True

disabled
False

hidden
False

default
`0`

operations.min_GC

label
Minimum GC content

type
`basic:decimal`

description
Discard reads with lower GC content.

required
True

disabled
False

hidden
False

default
`0.0`

operations.max_GC

label
Maximum GC content

type
`basic:decimal`

description
Discard reads with higher GC content.

required
True

disabled
False

hidden
False

default
`1.0`

operations.maxns

label
Max Ns after trimming

type
`basic:integer`

description
If non-negative, reads with more Ns than this (after trimming) will be discarded.

required
True

disabled
False

hidden
False

default
`-1`

operations.toss_junk

label
Discard reads with invalid characters as bases

type
`basic:boolean`

required
True

disabled
False

hidden
False

default
`False`

header_parsing.chastity_filter

label
Discard reads that fail Illumina chastity filtering

type
`basic:boolean`

description
Discard reads with id containing ‘ 1:Y:’ or ‘ 2:Y:’.

required
True

disabled
False

hidden
False

default
`False`

header_parsing.barcode_filter

label
Remove reads with unexpected barcodes

type
`basic:boolean`

description
Remove reads with unexpected barcodes if barcodes are set, or barcodes containing ‘N’ otherwise. A barcode must be the last part of the read header.

required
True

disabled
False

hidden
False

default
`False`

header_parsing.barcode_files

label
Barcode sequences

type
`list:data:seq:nucleotide`

description
FASTA file(s) with barcode sequences.

required
False

disabled
False

hidden
False

header_parsing.barcode_sequences

label
Literal barcode sequences

type
`list:basic:string`

description
Literal barcode sequences can be specified by inputting them one by one and pressing Enter after each sequence.

required
False

disabled
False

hidden
False

default
`[]`

header_parsing.x_min

label
Minimum X coordinate

type
`basic:integer`

description
If positive, discard reads with a smaller X coordinate.

required
True

disabled
False

hidden
False

default
`-1`

header_parsing.y_min

label
Minimum Y coordinate

type
`basic:integer`

description
If positive, discard reads with a smaller Y coordinate.

required
True

disabled
False

hidden
False

default
`-1`

header_parsing.x_max

label
Maximum X coordinate

type
`basic:integer`

description
If positive, discard reads with a larger X coordinate.

required
True

disabled
False

hidden
False

default
`-1`

header_parsing.y_max

label
Maximum Y coordinate

type
`basic:integer`

description
If positive, discard reads with a larger Y coordinate.

required
True

disabled
False

hidden
False

default
`-1`

complexity.entropy

label
Minimum entropy

type
`basic:decimal`

description
Set between 0 and 1 to filter reads with entropy below that value. Higher is more stringent.

required
True

disabled
False

hidden
False

default
`-1.0`

complexity.entropy_window

label
Length of sliding window used to calculate entropy

type
`basic:integer`

description
To use the sliding window set minimum entropy in range between 0.0 and 1.0.

required
True

disabled
False

hidden
False

default
`50`

complexity.entropy_k

label
Length of kmers used to calcuate entropy

type
`basic:integer`

required
True

disabled
False

hidden
False

default
`5`

complexity.entropy_mask

label
Mask low-entropy parts of sequences with N instead of discarding

type
`basic:boolean`

required
True

disabled
False

hidden
False

default
`False`

complexity.min_base_frequency

label
Minimum base frequency

type
`basic:integer`

required
True

disabled
False

hidden
False

default
`0`

fastqc.nogroup

label
Disable grouping of bases for reads >50bp

type
`basic:boolean`

description
All reports will show data for every base in the read. Using this option will cause fastqc to crash and burn if you use it on really long reads.

required
True

disabled
False

hidden
False

default
`False`

Output results fastq

label
Remaining upstream reads

type
`list:basic:file`

required
True

disabled
False

hidden
False

fastq2

label
Remaining downstream reads

type
`list:basic:file`

required
True

disabled
False

hidden
False

statistics

label
Statistics

type
`list:basic:file`

required
True

disabled
False

hidden
False

fastqc_url

label
Upstream quality control with FastQC

type
`list:basic:file:html`

required
True

disabled
False

hidden
False

fastqc_url2

label
Downstream quality control with FastQC

type
`list:basic:file:html`

required
True

disabled
False

hidden
False

fastqc_archive

label
Download upstream FastQC archive

type
`list:basic:file`

required
True

disabled
False

hidden
False

fastqc_archive2

label
Download downstream FastQC archive

type
`list:basic:file`

required
True

disabled
False

hidden
False

BBDuk (single-end)

`data:reads:fastq:single:bbduk:bbduk-single` (*data:reads:fastq:single* **reads**, *basic:integer* **min_length**, *list:data:seq:nucleotide* **sequences**, *list:basic:string* **literal_sequences**, *basic:integer* **kmer_length**, *basic:boolean* **check_reverse_complements**, *basic:boolean* **mask_middle_base**, *basic:integer* **min_kmer_hits**, *basic:decimal* **min_kmer_fraction**, *basic:decimal* **min_coverage_fraction**, *basic:integer* **hamming_distance**, *basic:integer* **query_hamming_distance**, *basic:integer* **edit_distance**, *basic:integer* **hamming_distance2**, *basic:integer* **query_hamming_distance2**, *basic:integer* **edit_distance2**, *basic:boolean* **forbid_N**, *basic:boolean* **find_best_match**, *basic:string* **k_trim**, *basic:string* **k_mask**, *basic:boolean* **mask_fully_covered**, *basic:integer* **min_k**, *basic:string* **quality_trim**, *basic:integer* **trim_quality**, *basic:string* **quality_encoding_offset**, *basic:boolean* **ignore_bad_quality**, *basic:integer* **trim_poly_A**, *basic:decimal* **min_length_fraction**, *basic:integer* **max_length**, *basic:integer* **min_average_quality**, *basic:integer* **min_average_quality_bases**, *basic:integer* **min_base_quality**, *basic:integer* **min_consecutive_bases**, *basic:integer* **trim_pad**, *basic:integer* **min_overlap**, *basic:integer* **min_insert**, *basic:integer* **force_trim_left**, *basic:integer* **force_trim_right**, *basic:integer* **force_trim_right2**, *basic:integer* **force_trim_mod**, *basic:integer* **restrict_left**, *basic:integer* **restrict_right**, *basic:decimal* **min_GC**, *basic:decimal* **max_GC**, *basic:integer* **maxns**, *basic:boolean* **toss_junk**, *basic:boolean* **chastity_filter**, *basic:boolean* **barcode_filter**, *list:data:seq:nucleotide* **barcode_files**, *list:basic:string* **barcode_sequences**, *basic:integer* **x_min**, *basic:integer* **y_min**, *basic:integer* **x_max**, *basic:integer* **y_max**, *basic:decimal* **entropy**, *basic:integer* **entropy_window**, *basic:integer* **entropy_k**, *basic:boolean* **entropy_mask**, *basic:integer* **min_base_frequency**, *basic:boolean* **nogroup**)[Source: v3.1.2]

Run BBDuk on single-end reads.

BBDuk combines the most common data-quality-related trimming, filtering, and masking operations into a single high-performance tool. It is capable of quality-trimming and filtering, adapter-trimming, contaminant-filtering via kmer matching, sequence masking, GC-filtering, length filtering, entropy-filtering, format conversion, histogram generation, subsampling, quality-score recalibration, kmer cardinality estimation, and various other operations in a single pass. See [here](https://jgi.doe.gov/data-and-tools/bbtools/bb-tools-user-guide/bbduk-guide/) for more information.

Input arguments reads

label
Reads

type
`data:reads:fastq:single`

required
True

disabled
False

hidden
False

min_length

label
Minimum length

type
`basic:integer`

description
Reads shorter than the minimum length will be discarded after trimming.

required
True

disabled
False

hidden
False

default
`10`

reference.sequences

label
Sequences

type
`list:data:seq:nucleotide`

description
Reference sequences include adapters, contaminants, and degenerate sequences. They can be provided in a multi-sequence FASTA file or as a set of literal sequences below.

required
False

disabled
False

hidden
False

reference.literal_sequences

label
Literal sequences

type
`list:basic:string`

description
Literal sequences can be specified by inputting them one by one and pressing Enter after each sequence.

required
False

disabled
False

hidden
False

default
`[]`

processing.kmer_length

label
Kmer length

type
`basic:integer`

description
Kmer length used for finding contaminants. Contaminants shorter than kmer length will not be found. Kmer length must be at least 1.

required
True

disabled
False

hidden
False

default
`27`

processing.check_reverse_complements

label
Check reverse complements

type
`basic:boolean`

description
Look for reverse complements of kmers in addition to forward kmers

required
True

disabled
False

hidden
False

default
`True`

processing.mask_middle_base

label
Mask the middle base of a kmer

type
`basic:boolean`

description
Treat the middle base of a kmer as a wildcard to increase sensitivity in the presence of errors.

required
True

disabled
False

hidden
False

default
`True`

processing.min_kmer_hits

label
Minimum number of kmer hits

type
`basic:integer`

description
Reads need at least this many matching kmers to be considered matching the reference.

required
True

disabled
False

hidden
False

default
1

processing.min_kmer_fraction

label
Minimum kmer fraction

type
`basic:decimal`

description
A read needs at least this fraction of its total kmers to hit a reference in order to be considered a match. If this and 'Minimum number of kmer hits' are set, the greater is used.

required
True

disabled
False

hidden
False

default
`0.0`

processing.min_coverage_fraction

label
Minimum coverage fraction

type
`basic:decimal`

description
A read needs at least this fraction of its total bases to be covered by reference kmers to be considered a match. If specified, 'Minimum coverage fraction' overrides 'Minimum number of kmer hits' and 'Minimum kmer fraction'.

required
True

disabled
False

hidden
False

default
`0.0`

processing.hamming_distance

label
Maximum Hamming distance for kmers (substitutions only)

type
`basic:integer`

description
Hamming distance i.e. the number of mismatches allowed in the kmer.

required
True

disabled
False

hidden
False

default
`0`

processing.query_hamming_distance

label
Hamming distance for query kmers

type
`basic:integer`

description
Set a hamming distance for query kmers instead of the read kmers. This makes the read processing much slower, but does not use additional memory.

required
True

disabled
False

hidden
False

default
`0`

processing.edit_distance

label
Maximum edit distance from reference kmers (substitutions and indels)

type
`basic:integer`

required
True

disabled
False

hidden
False

default
`0`

processing.hamming_distance2

label
Hamming distance for short kmers when looking for shorter kmers

type
`basic:integer`

required
True

disabled
False

hidden
False

default
`0`

processing.query_hamming_distance2

label
Hamming distance for short query kmers when looking for shorter kmers

type
`basic:integer`

required
True

disabled
False

hidden
False

default
`0`

processing.edit_distance2

label
Maximum edit distance from short reference kmers (substitutions and indels) when looking for shorter kmers

type
`basic:integer`

required
True

disabled
False

hidden
False

default
`0`

processing.forbid_N

label
Forbid matching of read kmers containing N

type
`basic:boolean`

description
By default, these will match a reference 'A' if 'Maximum Hamming distance for kmers' > 0 or 'Maximum edit distance from reference kmers' > 0, to increase sensitivity.

required
True

disabled
False

hidden
False

default
`False`

processing.find_best_match

label
Find best match

type
`basic:boolean`

description
If multiple matches, associate read with sequence sharing most kmers.

required
True

disabled
False

hidden
False

default
`True`

operations.k_trim

label
Trimming protocol to remove bases matching reference kmers from reads

type
`basic:string`

required
True

disabled
False

hidden
False

default
`f`

choices

- Don't trim: `f`
- Trim to the right: `r`
- Trim to the left: `l`

operations.k_mask

label
Symbol to replace bases matching reference kmers

type
`basic:string`

description
Allows any non-whitespace character other than t or f. Processes short kmers on both ends.

required
True

disabled
False

hidden
False

default
`f`

operations.mask_fully_covered

label
Only mask bases that are fully covered by kmers

type
`basic:boolean`

required
True

disabled
False

hidden
False

default
`False`

operations.min_k

label
Look for shorter kmers at read tips down to this length when k-trimming or masking

type
`basic:integer`

description
-1 means disabled. Enabling this will disable treating the middle base of a kmer as a wildcard to increase sensitivity in the presence of errors.

required
True

disabled
False

hidden
False

default
`-1`

operations.quality_trim

label
Trimming protocol to remove bases with quality below the minimum average region quality from read ends

type
`basic:string`

description
Performed after looking for kmers. If enabled, set also 'Average quality below which to trim region'.

required
True

disabled
False

hidden
False

default
`f`

choices

- Trim neither end: `f`
- Trim both ends: `rl`
- Trim only right end: `r`
- Trim only left end: `l`
- Use sliding window: `w`

operations.trim_quality

label
Average quality below which to trim region

type
`basic:integer`

description
Set trimming protocol to enable this parameter.

required
True

disabled
operations.quality_trim === 'f'

hidden
False

default
`6`

operations.quality_encoding_offset

label
Quality encoding offset

type
`basic:string`

description
Quality encoding offset for input FASTQ files.

required
True

disabled
False

hidden
False

default
`auto`

choices

- Sanger / Illumina 1.8+ (33): `33`
- Illumina up to 1.3+, 1.5+ (64): `64`
- Auto: `auto`

operations.ignore_bad_quality

label
Don't crash if quality values appear to be incorrect

type
`basic:boolean`

required
True

disabled
False

hidden
False

default
`False`

operations.trim_poly_A

label
Minimum length of poly-A or poly-T tails to trim on either end of reads

type
`basic:integer`

required
True

disabled
False

hidden
False

default
`0`

operations.min_length_fraction

label
Minimum length fraction

type
`basic:decimal`

description
Reads shorter than this fraction of original length after trimming will be discarded.

required
True

disabled
False

hidden
False

default
`0.0`

operations.max_length

label
Maximum length

type
`basic:integer`

description
Reads longer than this after trimming will be discarded.

required
False

disabled
False

hidden
False

operations.min_average_quality

label
Minimum average quality

type
`basic:integer`

description
Reads with average quality (after trimming) below this will be discarded.

required
True

disabled
False

hidden
False

default
`0`

operations.min_average_quality_bases

label
Number of initial bases to calculate minimum average quality from

type
`basic:integer`

description
If positive, calculate minimum average quality from this many initial bases

required
True

disabled
False

hidden
False

default
`0`

operations.min_base_quality

label
Minimum base quality below which reads are discarded after trimming

type
`basic:integer`

required
True

disabled
False

hidden
False

default
`0`

operations.min_consecutive_bases

label
Minimum number of consecutive called bases

type
`basic:integer`

required
True

disabled
False

hidden
False

default
`0`

operations.trim_pad

label
Number of bases to trim around matching kmers

type
`basic:integer`

required
True

disabled
False

hidden
False

default
`0`

operations.min_overlap

label
Minum number of overlapping bases

type
`basic:integer`

description
Require this many bases of overlap for detection.

required
True

disabled
False

hidden
False

default
`14`

operations.min_insert

label
Minimum insert size

type
`basic:integer`

description
Require insert size of at least this for overlap. Should be reduced to 16 for small RNA sequencing.

required
True

disabled
False

hidden
False

default
`40`

operations.force_trim_left

label
Position from which to trim bases to the left

type
`basic:integer`

required
True

disabled
False

hidden
False

default
`0`

operations.force_trim_right

label
Position from which to trim bases to the right

type
`basic:integer`

required
True

disabled
False

hidden
False

default
`0`

operations.force_trim_right2

label
Number of bases to trim from the right end

type
`basic:integer`

required
True

disabled
False

hidden
False

default
`0`

operations.force_trim_mod

label
Modulo to right-trim reads

type
`basic:integer`

description
Trim reads to the largest multiple of modulo.

required
True

disabled
False

hidden
False

default
`0`

operations.restrict_left

label
Number of leftmost bases to look in for kmer matches

type
`basic:integer`

required
True

disabled
False

hidden
False

default
`0`

operations.restrict_right

label
Number of rightmost bases to look in for kmer matches

type
`basic:integer`

required
True

disabled
False

hidden
False

default
`0`

operations.min_GC

label
Minimum GC content

type
`basic:decimal`

description
Discard reads with lower GC content.

required
True

disabled
False

hidden
False

default
`0.0`

operations.max_GC

label
Maximum GC content

type
`basic:decimal`

description
Discard reads with higher GC content.

required
True

disabled
False

hidden
False

default
`1.0`

operations.maxns

label
Max Ns after trimming

type
`basic:integer`

description
If non-negative, reads with more Ns than this (after trimming) will be discarded.

required
True

disabled
False

hidden
False

default
`-1`

operations.toss_junk

label
Discard reads with invalid characters as bases

type
`basic:boolean`

required
True

disabled
False

hidden
False

default
`False`

header_parsing.chastity_filter

label
Discard reads that fail Illumina chastity filtering

type
basic:boolean

description
Discard reads with id containing ‘ 1:Y:’ or ‘ 2:Y:’.

required
True

disabled
False

hidden
False

default
False

header_parsing.barcode_filter

label
Remove reads with unexpected barcodes

type
basic:boolean

description
Remove reads with unexpected barcodes if barcodes are set, or barcodes containing ‘N’ otherwise. A barcode must be the last part of the read header.

required
True

disabled
False

hidden
False

default
False

header_parsing.barcode_files

label
Barcode sequences

type
list:data:seq:nucleotide

description
FASTA file(s) with barcode sequences.

required
False

disabled
False

hidden
False

header_parsing.barcode_sequences

label
Literal barcode sequences

type
`list:basic:string`

description
Literal barcode sequences can be specified by inputting them one by one and pressing Enter after each sequence.

required
False

disabled
False

hidden
False

default
`[]`

header_parsing.x_min

label
Minimum X coordinate

type
`basic:integer`

description
If positive, discard reads with a smaller X coordinate.

required
True

disabled
False

hidden
False

default
`-1`

header_parsing.y_min

label
Minimum Y coordinate

type
`basic:integer`

description
If positive, discard reads with a smaller Y coordinate.

required
True

disabled
False

hidden
False

default
-1

header_parsing.x_max

label
Maximum X coordinate

type
basic:integer

description
If positive, discard reads with a larger X coordinate.

required
True

disabled
False

hidden
False

default
-1

header_parsing.y_max

label
Maximum Y coordinate

type
basic:integer

description
If positive, discard reads with a larger Y coordinate.

required
True

disabled
False

hidden
False

default
-1

complexity.entropy

label
Minimum entropy

type
basic:decimal

description
Set between 0 and 1 to filter reads with entropy below that value. Higher is more stringent.

required
True

disabled
False

hidden
False

default
`-1.0`

complexity.entropy_window

label
Length of sliding window used to calculate entropy

type
`basic:integer`

description
To use the sliding window set minimum entropy in range between 0.0 and 1.0.

required
True

disabled
False

hidden
False

default
`50`

complexity.entropy_k

label
Length of kmers used to calcuate entropy

type
`basic:integer`

required
True

disabled
False

hidden
False

default
`5`

complexity.entropy_mask

label
Mask low-entropy parts of sequences with N instead of discarding

type
`basic:boolean`

required
True

disabled
False

hidden
False

default
`False`

complexity.min_base_frequency

label
Minimum base frequency

type
`basic:integer`

required
True

disabled
False

hidden
False

default
`0`

fastqc.nogroup

label
Disable grouping of bases for reads >50bp

type
`basic:boolean`

description
All reports will show data for every base in the read. Using this option will cause fastqc to crash and burn if you use it on really long reads.

required
True

disabled
False

hidden
False

default
`False`

Output results fastq

label
Remaining reads

type
`list:basic:file`

required
True

disabled
False

hidden
False

statistics

label
Statistics

type
`list:basic:file`

required
True

disabled
False

hidden
False

fastqc_url

label
Quality control with FastQC

type
`list:basic:file:html`

required
True

disabled
False

hidden
False

fastqc_archive

label
Download FastQC archive

type
`list:basic:file`

required
True

disabled
False

hidden
False

BBDuk - STAR - featureCounts - QC

`data:workflow:rnaseq:featurecounts:qc:workflow-bbduk-star-featurecounts-qc` (*data:reads:fastq* **reads**, *data:index:star* **genome**, *data:annotation* **annotation**, *basic:string* **assay_type**, *data:index:salmon* **cdna_index**, *data:index:star* **rrna_reference**, *data:index:star* **globin_reference**, *list:data:seq:nucleotide* **adapters**, *list:basic:string* **custom_adapter_sequences**, *basic:integer* **kmer_length**, *basic:integer* **min_k**, *basic:integer* **hamming_distance**, *basic:integer* **maxns**, *basic:integer* **trim_quality**, *basic:integer* **min_length**, *basic:string* **quality_encoding_offset**, *basic:boolean* **ignore_bad_quality**, *basic:boolean* **unstranded**, *basic:boolean* **noncannonical**, *basic:boolean* **chimeric**, *basic:integer* **chim_segment_min**, *basic:boolean* **quant_mode**, *basic:boolean* **single_end**, *basic:string* **out_filter_type**, *basic:integer* **out_multimap_max**, *basic:integer* **out_mismatch_max**, *basic:decimal* **out_mismatch_nl_ma**
basic:integer **out_score_min**,
basic:decimal **out_mismatch_nrl_ma**
basic:integer **align_overhang_min**,

RNA-seq pipeline comprised of preprocessing, alignment and quantification.

First, reads are preprocessed by __BBDuk__ which removes adapters, trims reads for quality from the 3'-end, and discards reads that are too short after trimming. Compared to similar tools, BBDuk is regarded for its computational efficiency. Next, preprocessed reads are aligned by __STAR__ aligner. At the time of implementation, STAR is considered a state-of-the-art tool that consistently produces accurate results from diverse sets of reads, and performs well even with default settings. For more information see [this comparison of RNA-seq aligners](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5792058/). Finally, aligned reads are summarized to genes by __featureCounts__. Gaining wide adoption among the bioinformatics community, featureCounts yields expressions in a computationally efficient manner. All three tools in this workflow support parallelization to accelerate the analysis.

rRNA contamination rate in the sample is determined using the STAR aligner. Quality-trimmed reads are downsampled (using __Seqtk__ tool) and aligned to the rRNA reference sequences. The alignment rate indicates the percentage of the reads in the sample that are derived from the rRNA sequences.

Input arguments reads

label
Reads (FASTQ)

type
`data:reads:fastq`

description
Reads in FASTQ file, single or paired end.

required
True

disabled
False

hidden
False

genome

label
Indexed reference genome

type
`data:index:star`

description
Genome index prepared by STAR aligner indexing tool.

required
True

disabled
False

hidden
False

annotation

label
Annotation

type
`data:annotation`

description
GTF and GFF3 annotation formats are supported.

required
True

disabled
False

hidden
False

assay_type

label
Assay type

type
`basic:string`

description
In strand non-specific assay a read is considered overlapping with a feature regardless of whether it is mapped to the same or the opposite strand as the feature. In strand-specific forward assay and single reads, the read has to be mapped to the same strand as the feature. For paired-end reads, the first read has to be on the same strand and the second read on the opposite strand. In strand-specific reverse assay these rules are reversed.

required
True

disabled
False

hidden
False

default
`non_specific`

choices

- Strand non-specific: `non_specific`
- Strand-specific forward: `forward`
- Strand-specific reverse: `reverse`
- Detect automatically: `auto`

cdna_index

label
cDNA index file

type
`data:index:salmon`

description
Transcriptome index file created using the Salmon indexing tool. cDNA (transcriptome) sequences used for index file creation must be derived from the same species as the input sequencing reads to obtain the reliable analysis results.

required
False

disabled
False

hidden
assay_type != 'auto'

rrna_reference

label
Indexed rRNA reference sequence

type
`data:index:star`

description
Reference sequence index prepared by STAR aligner indexing tool.

required
True

disabled
False

hidden
False

globin_reference

label
Indexed Globin reference sequence

type
`data:index:star`

description
Reference sequence index prepared by STAR aligner indexing tool.

required
True

disabled
False

hidden
False

preprocessing.adapters

label
Adapters

type
`list:data:seq:nucleotide`

description
FASTA file(s) with adapters.

required
False

disabled
False

hidden
False

preprocessing.custom_adapter_sequences

label
Custom adapter sequences

type
`list:basic:string`

description
Custom adapter sequences can be specified by inputting them one by one and pressing Enter after each sequence.

required
False

disabled
False

hidden
False

default
`[]`

preprocessing.kmer_length

label
K-mer length [k=]

type
`basic:integer`

description
Kmer length used for finding contaminants. Contaminants shorter than kmer length will not be found. Kmer length must be at least 1.

required
True

disabled
False

hidden
False

default
`23`

preprocessing.min_k

label
Minimum k-mer length at right end of reads used for trimming [mink=]

type
`basic:integer`

required
True

disabled
preprocessing.adapters.length === 0 && preprocessing.custom_adapter_sequences.length === 0

hidden
False

default
11

preprocessing.hamming_distance

label
Maximum Hamming distance for k-mers [hammingdistance=]

type
basic:integer

description
Hamming distance i.e. the number of mismatches allowed in the kmer.

required
True

disabled
False

hidden
False

default
1

preprocessing.maxns

label
Max Ns after trimming [maxns=]

type
basic:integer

description
If non-negative, reads with more Ns than this (after trimming) will be discarded.

required
True

disabled
False

hidden
False

default
-1

preprocessing.trim_quality

label
Average quality below which to trim region [trimq=]

type
basic:integer

description
Phred algorithm is used, which is more accurate than naive trimming.

required
True

disabled
False

hidden
False

default
`10`

preprocessing.min_length

label
Minimum read length [minlength=]

type
`basic:integer`

description
Reads shorter than minimum read length after trimming are discarded.

required
True

disabled
False

hidden
False

default
`20`

preprocessing.quality_encoding_offset

label
Quality encoding offset [qin=]

type
`basic:string`

description
Quality encoding offset for input FASTQ files.

required
True

disabled
False

hidden
False

default
`auto`

choices

- Sanger / Illumina 1.8+: `33`
- Illumina up to 1.3+, 1.5+: `64`
- Auto: `auto`

preprocessing.ignore_bad_quality

label
Ignore bad quality [ignorebadquality]

type
`basic:boolean`

description
Don't crash if quality values appear to be incorrect.

required
True

disabled
False

hidden
False

default
`False`

alignment.unstranded

label
The data is unstranded [–outSAMstrandField intronMotif]

type
`basic:boolean`

description
For unstranded RNA-seq data, Cufflinks/Cuffdiff require spliced alignments with XS strand attribute, which STAR will generate with –outSAMstrandField intronMotif option. As required, the XS strand attribute will be generated for all alignments that contain splice junctions. The spliced alignments that have undefined strand (i.e. containing only non-canonical unannotated junctions) will be suppressed. If you have stranded RNA-seq data, you do not need to use any specific STAR options. Instead, you need to run Cufflinks with the library option –library-type options. For example, cufflinks –library-type fr-firststrand should be used for the standard dUTP protocol, including Illumina's stranded TruSeq. This option has to be used only for Cufflinks runs and not for STAR runs.

required
True

disabled
False

hidden
False

default
`False`

alignment.noncannonical

label
Remove non-cannonical junctions (Cufflinks compatibility)

type
`basic:boolean`

description
It is recommended to remove the non-canonical junctions for Cufflinks runs using –outFilterIntronMotifs RemoveNoncanonical.

required
True

disabled
False

hidden
False

default
`False`

alignment.chimeric_reads.chimeric

label
Detect chimeric and circular alignments [–chimOutType SeparateSAMold]

type
`basic:boolean`

description
To switch on detection of chimeric (fusion) alignments (in addition to normal mapping), –chimSegmentMin should be set to a positive value. Each chimeric alignment consists of two segments. Each segment is non-chimeric on its own, but the segments are chimeric to each other (i.e. the segments belong to different chromosomes, or different strands, or are far from each other). Both segments may contain splice junctions, and one of the segments may contain portions of both mates. –chimSegmentMin parameter controls the minimum mapped length of the two segments that is allowed. For example, if you have 2x75 reads and used –chimSegmentMin 20, a chimeric alignment with 130b on one chromosome and 20b on the other will be output, while 135 + 15 won't be.

required
True

disabled
False

hidden
False

default
`False`

alignment.chimeric_reads.chim_segment_min

label
Minimum length of chimeric segment [–chimSegmentMin]

type
`basic:integer`

required
True

disabled
!alignment.chimeric_reads.chimeric

hidden
False

default
`20`

alignment.transcript_output.quant_mode

label
Output in transcript coordinates [–quantMode]

type
`basic:boolean`

description
With –quantMode TranscriptomeSAM option STAR will output alignments translated into transcript coordinates in the Aligned.toTranscriptome.out.bam file (in addition to alignments in genomic coordinates in Aligned.*.sam/bam files). These transcriptomic alignments can be used with various transcript quantification software that require reads to be mapped to transcriptome, such as RSEM or eXpress.

required
True

disabled
False

hidden
False

default
`False`

alignment.transcript_output.single_end

label
Allow soft-clipping and indels [–quantTranscriptomeBan Singleend]

type
`basic:boolean`

description
By default, the output satisfies RSEM requirements: soft-clipping or indels are not allowed. Use –quantTranscriptomeBan Singleend to allow insertions, deletions and soft-clips in the transcriptomic alignments, which can be used by some expression quantification softwares (e.g. eXpress).

required
True

disabled
!t_coordinates.quant_mode

hidden
False

default
`False`

alignment.filtering_options.out_filter_type

label
Type of filtering [–outFilterType]

type
`basic:string`

description
Normal: standard filtering using only current alignment; BySJout: keep only those reads that contain junctions that passed filtering into SJ.out.tab.

required
True

disabled
False

hidden
False

default
`Normal`

choices

- Normal: `Normal`
- BySJout: `BySJout`

alignment.filtering_options.out_multimap_max

label
Maximum number of loci [–outFilterMultimapNmax]

type
`basic:integer`

description
Maximum number of loci the read is allowed to map to. Alignments (all of them) will be output only if the read maps to no more loci than this value. Otherwise no alignments will be output, and the read will be counted as 'mapped to too many loci' (default: 10).

required
False

disabled
False

hidden
False

alignment.filtering_options.out_mismatch_max

label
Maximum number of mismatches [–outFilterMismatchNmax]

type
`basic:integer`

description
Alignment will be output only if it has fewer mismatches than this value (default: 10). Large number (e.g. 999) switches off this filter.

required
False

disabled
False

hidden
False

alignment.filtering_options.out_mismatch_nl_max

label
Maximum no. of mismatches (map length) [–outFilterMismatchNoverLmax]

type
`basic:decimal`

description
Alignment will be output only if its ratio of mismatches to *mapped* length is less than or equal to this value (default: 0.3). The value should be between 0.0 and 1.0.

required
False

disabled
False

hidden
False

alignment.filtering_options.out_score_min

label
Minimum alignment score [–outFilterScoreMin]

type
`basic:integer`

description
Alignment will be output only if its score is higher than or equal to this value (default: 0).

required
False

disabled
False

hidden
False

alignment.filtering_options.out_mismatch_nrl_max

label
Maximum no. of mismatches (read length) [–outFilterMismatchNoverReadLmax]

type
`basic:decimal`

description
Alignment will be output only if its ratio of mismatches to *read* length is less than or equal to this value (default: 1.0). Using 0.04 for 2x100bp, the max number of mismatches is calculated as 0.04*200=8 for the paired read. The value should be between 0.0 and 1.0.

required
False

disabled
False

hidden
False

alignment.alignment_options.align_overhang_min

label
Minimum overhang [–alignSJoverhangMin]

type
`basic:integer`

description
Minimum overhang (i.e. block size) for spliced alignments (default: 5).

required
False

disabled
False

hidden
False

alignment.alignment_options.align_sjdb_overhang_min

label
Minimum overhang (sjdb) [–alignSJDBoverhangMin]

type
`basic:integer`

description
Minimum overhang (i.e. block size) for annotated (sjdb) spliced alignments (default: 3).

required
False

disabled
False

hidden
False

alignment.alignment_options.align_intron_size_min

label
Minimum intron size [–alignIntronMin]

type
`basic:integer`

description
Minimum intron size: the genomic gap is considered an intron if its length >= alignIntronMin, otherwise it is considered Deletion (default: 21).

required
False

disabled
False

hidden
False

alignment.alignment_options.align_intron_size_max

label
Maximum intron size [–alignIntronMax]

type
`basic:integer`

description
Maximum intron size, if 0, max intron size will be determined by (2pow(winBinNbits)*winAnchorDistNbins)(default: 0).

required
False

disabled
False

hidden
False

alignment.alignment_options.align_gap_max

label
Minimum gap between mates [–alignMatesGapMax]

type
`basic:integer`

description
Maximum gap between two mates, if 0, max intron gap will be determined by (2pow(winBinNbits)*winAnchorDistNbins) (default: 0).

required
False

disabled
False

hidden
False

alignment.alignment_options.align_end_alignment

label
Read ends alignment [–alignEndsType]

type
`basic:string`

description
Type of read ends alignment (default: Local). Local: standard local alignment with soft-clipping allowed. EndToEnd: force end-to-end read alignment, do not soft-clip. Extend5pOfRead1: fully extend only the 5p of the read1, all other ends: local alignment. Extend5pOfReads12: fully extend only the 5' of the both read1 and read2, all other ends use local alignment.

required
True

disabled
False

hidden
False

default
`Local`

choices

- Local: `Local`
- EndToEnd: `EndToEnd`
- Extend5pOfRead1: `Extend5pOfRead1`
- Extend5pOfReads12: `Extend5pOfReads12`

alignment.output_options.out_unmapped

label
Output unmapped reads (SAM) [–outSAMunmapped Within]

type
`basic:boolean`

description
Output of unmapped reads in the SAM format.

required
True

disabled
False

hidden
False

default
`False`

alignment.output_options.out_sam_attributes

label
Desired SAM attributes [–outSAMattributes]

type
`basic:string`

description
A string of desired SAM attributes, in the order desired for the output SAM.

required
True

disabled
False

hidden
False

default
`Standard`

choices

- Standard: `Standard`
- All: `All`
- NH HI NM MD: `NH HI NM MD`
- None: `None`

alignment.output_options.out_rg_line

label
SAM/BAM read group line [–outSAMattrRGline]

type
`basic:string`

description
The first word contains the read group identifier and must start with ID:, e.g. –outSAMattrRGline ID:xxx CN:yy ”DS:z z z” xxx will be added as RG tag to each output alignment. Any spaces in the tag values have to be double quoted. Comma separated RG lines correspons to different (comma separated) input files in -readFilesIn. Commas have to be surrounded by spaces, e.g. -outSAMattrRGline ID:xxx , ID:zzz ”DS:z z” , ID:yyy DS:yyyy.

required
False

disabled
False

hidden
False

quantification.n_reads

label
Number of reads in subsampled alignment file

type
`basic:integer`

description
Alignment (.bam) file subsample size. Increase the number of reads to make automatic detection more reliable. Decrease the number of reads to make automatic detection run faster.

required
True

disabled
False

hidden
assay_type != ‘auto’

default
`5000000`

quantification.feature_class

label
Feature class [-t]

type
`basic:string`

description
Feature class (3rd column in GTF/GFF3 file) to be used. All other features will be ignored.

required
True

disabled
False

hidden
False

default
`exon`

quantification.feature_type

label
Feature type

type
`basic:string`

description
The type of feature the quantification program summarizes over (e.g. gene or transcript-level analysis). The value of this parameter needs to be chosen in line with 'ID attribute' below.

required
True

disabled
False

hidden
False

default
`gene`

choices

- gene: `gene`
- transcript: `transcript`

quantification.id_attribute

label
ID attribute [-g]

type
`basic:string`

description
GTF/GFF3 attribute to be used as feature ID. Several GTF/GFF3 lines with the same feature ID will be considered as parts of the same feature. The feature ID is used to identify the counts in the output table. In GTF files this is usually 'gene_id', in GFF3 files this is often 'ID', and 'transcript_id' is frequently a valid choice for both annotation formats.

required
True

disabled
False

hidden
False

default
`gene_id`

choices

- gene_id: `gene_id`
- transcript_id: `transcript_id`
- ID: `ID`
- geneid: `geneid`

quantification.by_read_group

label
Assign reads by read group

type
`basic:boolean`

description
RG tag is required to be present in the input BAM files.

required
True

disabled
False

hidden
False

default
`True`

downsampling.n_reads

label
Number of reads

type
`basic:integer`

description
Number of reads to include in subsampling.

required
True

disabled
False

hidden
False

default
`1000000`

downsampling.advanced.seed

label
Seed [-s]

type
`basic:integer`

description
Using the same random seed makes reads subsampling more reproducible in different environments.

required
True

disabled
False

hidden
False

default
11

downsampling.advanced.fraction

label
Fraction of reads used

type
`basic:decimal`

description
Use the fraction of reads [0.0 - 1.0] from the orignal input file instead of the absolute number of reads. If set, this will override the 'Number of reads' input parameter.

required
False

disabled
False

hidden
False

downsampling.advanced.two_pass

label
2-pass mode [-2]

type
`basic:boolean`

description
Enable two-pass mode when down-sampling. Two-pass mode is twice as slow but with much reduced memory.

required
True

disabled
False

hidden
False

default
`False`

Output results

BBDuk - Salmon - QC

data:workflow:rnaseq:salmon:workflow-bbduk-salmon-qc (*data:reads:fastq* **reads**, *data:index:salmon* **salmon_index**, *data:index:star* **genome**, *data:annotation* **annotation**, *data:index:star* **rrna_reference**, *data:index:star* **globin_reference**, *list:data:seq:nucleotide* **adapters**, *list:basic:string* **custom_adapter_sequences**, *basic:integer* **kmer_length**, *basic:integer* **min_k**, *basic:integer* **hamming_distance**, *basic:integer* **maxns**, *basic:integer* **trim_quality**, *basic:integer* **min_length**, *basic:string* **quality_encoding_offset**, *basic:boolean* **ignore_bad_quality**, *basic:boolean* **seq_bias**, *basic:boolean* **gc_bias**, *basic:decimal* **consensus_slack**, *basic:decimal* **min_score_fraction**, *basic:integer* **range_factorization_bins**, *basic:integer* **min_assigned_frag**, *basic:integer* **num_bootstraps**, *basic:integer* **num_gibbs_samples**, *basic:integer* **n_reads**, *basic:integer* **seed**, *basic:decimal* **fraction**, *basic:boolean* **two_pass**)[Source: v4.3.1]

Alignment-free RNA-Seq pipeline.

Salmon tool and tximport package are used in quantification step to produce gene-level abundance estimates.

rRNA and globin-sequence contamination rate in the sample is determined using STAR aligner. Quality-trimmed reads are down-sampled (using Seqtk tool) and aligned to the genome, rRNA and globin reference sequences. The rRNA and globin-sequence alignment rates indicate the percentage of the reads in the sample that are of rRNA and globin origin, respectively. Alignment of down-sampled data to a whole genome reference sequence is used to produce an alignment file suitable for Samtools and QoRTs QC analysis.

Per-sample analysis results and QC data is summarized by the MultiQC tool.

Input arguments reads

label
Select sample(s) (FASTQ)

type
`data:reads:fastq`

description
Reads in FASTQ file, single or paired end.

required
True

disabled
False

hidden
False

salmon_index

label
Salmon index

type
`data:index:salmon`

description
Transcriptome index file created using the Salmon indexing tool.

required
True

disabled
False

hidden
False

genome

label
Indexed reference genome

type
`data:index:star`

description
Genome index prepared by STAR aligner indexing tool.

required
True

disabled
False

hidden
False

annotation

label
Annotation

type
`data:annotation`

description
GTF and GFF3 annotation formats are supported.

required
True

disabled
False

hidden
False

rrna_reference

label
Indexed rRNA reference sequence

type
`data:index:star`

description
Reference sequence index prepared by STAR aligner indexing tool.

required
True

disabled
False

hidden
False

globin_reference

label
Indexed Globin reference sequence

type
`data:index:star`

description
Reference sequence index prepared by STAR aligner indexing tool.

required
True

disabled
False

hidden
False

preprocessing.adapters

label
Adapters

type
`list:data:seq:nucleotide`

description
FASTA file(s) with adapters.

required
False

disabled
False

hidden
False

preprocessing.custom_adapter_sequences

label
Custom adapter sequences

type
`list:basic:string`

description
Custom adapter sequences can be specified by inputting them one by one and pressing Enter after each sequence.

required
False

disabled
False

hidden
False

default
`[]`

preprocessing.kmer_length

label
K-mer length

type
`basic:integer`

description
K-mer length must be smaller or equal to the length of adapters.

required
True

disabled
False

hidden
False

default
`23`

preprocessing.min_k

label
Minimum k-mer length at right end of reads used for trimming

type
`basic:integer`

required
True

disabled
preprocessing.adapters.length === 0 && preprocessing.custom_adapter_sequences.length === 0

hidden
False

default
`11`

preprocessing.hamming_distance

label
Maximum Hamming distance for k-mers

type
`basic:integer`

required
True

disabled
False

hidden
False

default
`1`

preprocessing.maxns

label
Max Ns after trimming

type
`basic:integer`

description
If non-negative, reads with more Ns than this (after trimming) will be discarded.

required
True

disabled
False

hidden
False

default
`-1`

preprocessing.trim_quality

label
Quality below which to trim reads from the right end

type
`basic:integer`

description
Phred algorithm is used, which is more accurate than naive trimming.

required
True

disabled
False

hidden
False

default
`10`

preprocessing.min_length

label
Minimum read length

type
`basic:integer`

description
Reads shorter than minimum read length after trimming are discarded.

required
True

disabled
False

hidden
False

default
`20`

preprocessing.quality_encoding_offset

label
Quality encoding offset

type
`basic:string`

description
Quality encoding offset for input FASTQ files.

required
True

disabled
False

hidden
False

default
`auto`

choices

- Sanger / Illumina 1.8+: `33`
- Illumina up to 1.3+, 1.5+: `64`
- Auto: `auto`

preprocessing.ignore_bad_quality

label
Ignore bad quality

type
`basic:boolean`

description
Don't crash if quality values appear to be incorrect.

required
True

disabled
False

hidden
False

default
`False`

quantification.seq_bias

label
Perform sequence-specific bias correction

type
`basic:boolean`

description
Perform sequence-specific bias correction.

required
True

disabled
False

hidden
False

default
`True`

quantification.gc_bias

label
Perform fragment GC bias correction

type
`basic:boolean`

description
Perform fragment GC bias correction. If single-end reads are selected as input in this workflow, it is recommended that you set this option to False. If you selected paired-end reads as input in this workflow, it is recommended that you set this option to True.

required
False

disabled
False

hidden
False

quantification.consensus_slack

label
Consensus slack

type
`basic:decimal`

description
The amount of slack allowed in the quasi-mapping consensus mechanism. Normally, a transcript must cover all hits to be considered for mapping. If this is set to a fraction, X, greater than 0 (and in [0,1)), then a transcript can fail to cover up to (100 * X)% of the hits before it is discounted as

a mapping candidate. The default value of this option is 0.2 in selective alignment mode and 0 otherwise.

required
False

disabled
False

hidden
False

quantification.min_score_fraction

label
Minimum alignment score fraction

type
`basic:decimal`

description
The fraction of the optimal possible alignment score that a mapping must achieve in order to be considered valid - should be in (0,1].

required
True

disabled
False

hidden
False

default
`0.65`

quantification.range_factorization_bins

label
Range factorization bins

type
`basic:integer`

description
Factorizes the likelihood used in quantification by adopting a new notion of equivalence classes based on the conditional probabilities with which fragments are generated from different transcripts. This is a more fine-grained factorization than the normal rich equivalence classes. The default value (4) corresponds to the default used in Zakeri et al. 2017 and larger values imply a more fine-grained factorization. If range factorization is enabled, a common value to select for this parameter is 4. A value of 0 signifies the use of basic rich equivalence classes.

required
True

disabled
False

hidden
False

default
4

quantification.min_assigned_frag

label
Minimum number of assigned fragments

type
`basic:integer`

description
The minimum number of fragments that must be assigned to the transcriptome for quantification to proceed.

required
True

disabled
False

hidden
False

default
`10`

quantification.num_bootstraps

label
–numBootstraps

type
`basic:integer`

description
Salmon has the ability to optionally compute bootstrapped abundance estimates. This is done by resampling (with replacement) from the counts assigned to the fragment equivalence classes, and then re-running the optimization procedure, either the EM or VBEM, for each such sample. The values of these different bootstraps allows us to assess technical variance in the main abundance estimates we produce. Such estimates can be useful for downstream (e.g. differential expression) tools that can make use of such uncertainty estimates. This option takes a positive integer that dictates the number of bootstrap samples to compute. The more samples computed, the better the estimates of varaiance, but the more computation (and time) required.

required
False

disabled
quantification.num_gibbs_samples

hidden
False

quantification.num_gibbs_samples

label
–numGibbsSamples

type
`basic:integer`

description
Just as with the bootstrap procedure above, this option produces samples that allow us to estimate the variance in abundance estimates. However, in this case the samples are generated using posterior Gibbs sampling over the fragment equivalence classes rather than bootstrapping. We are currently

analyzing these different approaches to assess the potential trade-offs in time / accuracy. The –numBootstraps and –numGibbsSamples options are mutually exclusive (i.e. in a given run, you must set at most one of these options to a positive integer.)

required
False

disabled
quantification.num_bootstraps

hidden
False

downsampling.n_reads

label
Number of reads

type
`basic:integer`

description
Number of reads to include in subsampling.

required
True

disabled
False

hidden
False

default
`10000000`

downsampling.seed

label
Number of reads

type
`basic:integer`

description
Using the same random seed makes reads subsampling reproducible in different environments.

required
True

disabled
False

hidden
False

default
`11`

downsampling.fraction

label
Fraction of reads

type
`basic:decimal`

description
Use the fraction of reads [0.0 - 1.0] from the orignal input file instead of the absolute number of reads. If set, this will override the 'Number of reads' input parameter.

required
False

disabled
False

hidden
False

downsampling.two_pass

label
2-pass mode

type
`basic:boolean`

description
Enable two-pass mode when down-sampling. Two-pass mode is twice as slow but with much reduced memory usage.

required
True

disabled
False

hidden
False

default
`False`

Output results

BED file

`data:bedupload-bed` (*basic:file* **src**, *basic:string* **species**, *basic:string* **build**)[Source: v1.5.0]

Import a BED file (.bed) which is a tab-delimited text file that defines a feature track. It can have any file extension, but .bed is recommended. The BED file format is described on the [UCSC Genome Bioinformatics web site](http://genome.ucsc.edu/FAQ/FAQformat#format1).

Input arguments src

label
BED file

type
`basic:file`

description
Upload BED file annotation track. The first three required BED fields are chrom, chromStart and chromEnd.

required
True

validate_regex
`\.(bed|narrowPeak)$`

species

label
Species

type
`basic:string`

description
Species latin name.

choices

- Homo sapiens: `Homo sapiens`
- Mus musculus: `Mus musculus`
- Rattus norvegicus: `Rattus norvegicus`
- Dictyostelium discoideum: `Dictyostelium discoideum`
- Odocoileus virginianus texanus: `Odocoileus virginianus texanus`
- Solanum tuberosum: `Solanum tuberosum`

build

label
Genome build

type
`basic:string`

Output results bed

label
BED file

type
`basic:file`

bed_jbrowse

label
Bgzip bed file for JBrowse

type
`basic:file`

tbi_jbrowse

label
Bed file index for Jbrowse

type
`basic:file`

species

label
Species

type
`basic:string`

build

label
Build

type
`basic:string`

BEDPE file

data:bedpe:upload-bedpe (*basic:file* **src**, *basic:string* **species**, *basic:string* **build**)[Source: v1.3.1]

Upload BEDPE files.

Input arguments src

label
Select BEDPE file to upload

type
`basic:file`

required
True

disabled
False

hidden
False

species

label
Species

type
`basic:string`

required
True

disabled
False

hidden
False

choices

- Homo sapiens: `Homo sapiens`
- Mus musculus: `Mus musculus`
- Rattus norvegicus: `Rattus norvegicus`
- Dictyostelium discoideum: `Dictyostelium discoideum`
- Odocoileus virginianus texanus: `Odocoileus virginianus texanus`
- Solanum tuberosum: `Solanum tuberosum`

build

label
Build

type
`basic:string`

required
True

disabled
False

hidden
False

Output results bedpe

label
BEDPE file

type
`basic:file`

required
True

disabled
False

hidden
False

species

label
Species

type
`basic:string`

required
True

disabled
False

hidden
False

build

label
Build

type
`basic:string`

required
True

disabled
False

hidden
False

BWA ALN

data:alignment:bam:bwaalnalignment-bwa-aln (*data:index:bwa* **genome**, *data:reads:fastq* **reads**, *basic:integer* **q**, *basic:boolean* **use_edit**, *basic:integer* **edit_value**, *basic:decimal* **fraction**, *basic:boolean* **seeds**, *basic:integer* **seed_length**, *basic:integer* **seed_dist**) [Source: v2.6.2]

Read aligner for mapping low-divergent sequences against a large reference genome. Designed for Illumina sequence reads up to 100bp.

Input arguments genome

label
Reference genome

type
`data:index:bwa`

reads

label
Reads

type
`data:reads:fastq`

q

label
Quality threshold

type
`basic:integer`

description
Parameter for dynamic read trimming.

default
`0`

use_edit

label
Use maximum edit distance (excludes fraction of missing alignments)

type
`basic:boolean`

default
`False`

edit_value

label
Maximum edit distance

type
`basic:integer`

hidden
!use_edit

default
`5`

fraction

label
Fraction of missing alignments

type
`basic:decimal`

description
The fraction of missing alignments given 2% uniform base error rate. The maximum edit distance is automatically chosen for different read lengths.

hidden
use_edit

default
`0.04`

seeds

label
Use seeds

type
`basic:boolean`

default
`False`

seed_length

label
Seed length

type
`basic:integer`

description
Take the first X subsequence as seed. If X is larger than the query sequence, seeding will be disabled. For long reads, this option is typically ranged from 25 to 35 for value 2 in seed maximum edit distance.

hidden
!seeds

default
`35`

seed_dist

label
Seed maximum edit distance

type
`basic:integer`

hidden
!seeds

default
2

Output results bam

label
Alignment file

type
`basic:file`

description
Position sorted alignment

bai

label
Index BAI

type
`basic:file`

unmapped

label
Unmapped reads

type
`basic:file`

required
False

stats

label
Statistics

type
`basic:file`

species

label
Species

type
`basic:string`

build

label
Build

type
`basic:string`

BWA MEM

data:alignment:bam:bwamemalignment-bwa-mem (*data:index:bwa* **genome**, *data:reads:fastq* **reads**, *basic:integer* **seed_l**, *basic:integer* **band_w**, *basic:decimal* **re_seeding**, *basic:boolean* **m**, *basic:integer* **match**, *basic:integer* **missmatch**, *basic:integer* **gap_o**, *basic:integer* **gap_e**, *basic:integer* **clipping**, *basic:integer* **unpaired_p**, *basic:boolean* **report_all**, *basic:integer* **report_tr**)[Source: v3.6.0]

BWA MEM is a read aligner for mapping low-divergent sequences against a large reference genome. Designed for longer sequences ranged from 70bp to 1Mbp. The algorithm works by seeding alignments with maximal exact matches (MEMs) and then extending seeds with the affine-gap Smith-Waterman algorithm (SW). See [here](http://bio-bwa.sourceforge.net/) for more information.

Input arguments genome

label
Reference genome

type
`data:index:bwa`

reads

label
Reads

type
`data:reads:fastq`

seed_l

label
Minimum seed length

type
`basic:integer`

description
Minimum seed length. Matches shorter than minimum seed length will be missed. The alignment speed is usually insensitive to this value unless it significantly deviates from 20.

default
`19`

band_w

label
Band width

type
`basic:integer`

description
Gaps longer than this will not be found.

default
`100`

re_seeding

label
Re-seeding factor

type
`basic:decimal`

description
Trigger re-seeding for a MEM longer than minSeedLen*FACTOR. This is a key heuristic parameter for tuning the performance. Larger value yields fewer seeds, which leads to faster alignment speed but lower accuracy.

default
`1.5`

m

label
Mark shorter split hits as secondary

type
`basic:boolean`

description
Mark shorter split hits as secondary (for Picard compatibility)

default
`False`

scoring.match

label
Score of a match

type
`basic:integer`

default
`1`

scoring.missmatch

label
Mismatch penalty

type
`basic:integer`

default
`4`

scoring.gap_o

label
Gap open penalty

type
`basic:integer`

default
`6`

scoring.gap_e

label
Gap extension penalty

type
basic:integer

default
1

scoring.clipping

label
Clipping penalty

type
basic:integer

description
Clipping is applied if final alignment score is smaller than (best score reaching the end of query) - (Clipping penalty)

default
5

scoring.unpaired_p

label
Penalty for an unpaired read pair

type
basic:integer

description
Affinity to force pair. Score: scoreRead1+scoreRead2-Penalty

default
9

reporting.report_all

label
Report all found alignments

type
basic:boolean

description
Output all found alignments for single-end or unpaired paired-end reads. These alignments will be flagged as secondary alignments.

default
False

reporting.report_tr

label
Report threshold score

type
basic:integer

description
Don't output alignment with score lower than defined number. This option only affects output.

default
30

Output results bam

label
Alignment file

type
`basic:file`

description
Position sorted alignment

bai

label
Index BAI

type
`basic:file`

unmapped

label
Unmapped reads

type
`basic:file`

required
False

stats

label
Statistics

type
`basic:file`

species

label
Species

type
`basic:string`

build

label
Build

type
`basic:string`

BWA MEM2

`data:alignment:bam:bwamem2alignment-bwa-mem2` (*data:index:bwamem2* **genome**, *data:reads:fastq* **reads**, *basic:integer* **seed_l**, *basic:integer* **band_w**, *basic:decimal* **re_seeding**, *basic:boolean* **m**, *basic:integer* **match**, *basic:integer* **missmatch**, *basic:integer* **gap_o**, *basic:integer* **gap_e**, *basic:integer* **clipping**, *basic:integer* **unpaired_p**, *basic:boolean* **report_all**, *basic:integer* **report_tr**) [Source: v1.3.0]

Bwa-mem2 is the next version of the bwa-mem algorithm in bwa. It produces alignment identical to bwa and is ~1.3-3.1x faster depending on the use-case, dataset and the running machine. See [here](https://github.com/bwa-mem2/bwa-mem2) for more information.

Input arguments genome

label
Reference genome

type
`data:index:bwamem2`

reads

label
Reads

type
`data:reads:fastq`

seed_l

label
Minimum seed length

type
`basic:integer`

description
Minimum seed length. Matches shorter than minimum seed length will be missed. The alignment speed is usually insensitive to this value unless it significantly deviates from 20.

default
`19`

band_w

label
Band width

type
`basic:integer`

description
Gaps longer than this will not be found.

default
`100`

re_seeding

label
Re-seeding factor

type
`basic:decimal`

description
Trigger re-seeding for a MEM longer than minSeedLen*FACTOR. This is a key heuristic parameter for tuning the performance. Larger value yields fewer seeds, which leads to faster alignment speed but lower accuracy.

default
`1.5`

m

label
Mark shorter split hits as secondary

type
`basic:boolean`

description
Mark shorter split hits as secondary (for Picard compatibility)

default
`False`

scoring.match

label
Score of a match

type
`basic:integer`

default
1

scoring.missmatch

label
Mismatch penalty

type
`basic:integer`

default
4

scoring.gap_o

label
Gap open penalty

type
`basic:integer`

default
6

scoring.gap_e

label
Gap extension penalty

type
`basic:integer`

default
1

scoring.clipping

label
Clipping penalty

type
`basic:integer`

description
Clipping is applied if final alignment score is smaller than (best score reaching the end of query) - (Clipping penalty)

default
`5`

scoring.unpaired_p

label
Penalty for an unpaired read pair

type
`basic:integer`

description
Affinity to force pair. Score: scoreRead1+scoreRead2-Penalty

default
`9`

reporting.report_all

label
Report all found alignments

type
`basic:boolean`

description
Output all found alignments for single-end or unpaired paired-end reads. These alignments will be flagged as secondary alignments.

default
`False`

reporting.report_tr

label
Report threshold score

type
`basic:integer`

description
Don't output alignment with score lower than defined number. This option only affects output.

default
`30`

Output results bam

label
Alignment file

type
`basic:file`

description
Position sorted alignment

bai

label
Index BAI

type
`basic:file`

unmapped

label
Unmapped reads

type
`basic:file`

required
False

stats

label
Statistics

type
`basic:file`

species

label
Species

type
`basic:string`

build

label
Build

type
`basic:string`

BWA SW

`data:alignment:bam:bwaswalignment-bwa-sw` (*data:index:bwa* **genome**, *data:reads:fastq* **reads**, *basic:integer* **match**, *basic:integer* **missmatch**, *basic:integer* **gap_o**, *basic:integer* **gap_e**) [Source: v2.5.2]

Read aligner for mapping low-divergent sequences against a large reference genome. Designed for longer sequences ranged from 70bp to 1Mbp. The paired-end mode only works for reads Illumina short-insert libraries.

Input arguments genome

label
Reference genome

type
`data:index:bwa`

reads

label
Reads

type
`data:reads:fastq`

match

label
Score of a match

type
`basic:integer`

default
1

missmatch

label
Mismatch penalty

type
`basic:integer`

default
3

gap_o

label
Gap open penalty

type
`basic:integer`

default
5

gap_e

label
Gap extension penalty

type
`basic:integer`

default
2

Output results bam

label
Alignment file

type
`basic:file`

description
Position sorted alignment

bai

label
Index BAI

type
`basic:file`

unmapped

label
Unmapped reads

type
`basic:file`

required
False

stats

label
Statistics

type
`basic:file`

species

label
Species

type
`basic:string`

build

label
Build

type
`basic:string`

BWA genome index

`data:index:bwa:bwa-index` (*data:seq:nucleotide* **ref_seq**)[Source: v1.2.0]

Create BWA genome index.

Input arguments ref_seq

label
Reference sequence (nucleotide FASTA)

type
`data:seq:nucleotide`

required
True

disabled
False

hidden
False

Output results index

label
BWA index

type
`basic:dir`

required
True

disabled
False

hidden
False

fastagz

label
FASTA file (compressed)

type
`basic:file`

required
True

disabled
False

hidden
False

fasta

label
FASTA file

type
`basic:file`

required
True

disabled
False

hidden
False

fai

label
FASTA file index

type
`basic:file`

required
True

disabled
False

hidden
False

species

label
Species

type
`basic:string`

required
True

disabled
False

hidden
False

build

label
Build

type
`basic:string`

required
True

disabled
False

hidden
False

BWA-MEM2 genome index

`data:index:bwamem2:bwamem2-index` (*data:seq:nucleotide* **ref_seq**)[Source: v1.1.0]

Create BWA-MEM2 genome index.

Input arguments ref_seq

label
Reference sequence (nucleotide FASTA)

type
`data:seq:nucleotide`

required
True

disabled
False

hidden
False

Output results index

label
BWA-MEM2 index

type
`basic:dir`

required
True

disabled
False

hidden
False

fastagz

label
FASTA file (compressed)

type
`basic:file`

required
True

disabled
False

hidden
False

fasta

label
FASTA file

type
`basic:file`

required
True

disabled
False

hidden
False

fai

label
FASTA file index

type
`basic:file`

required
True

disabled
False

hidden
False

species

label
Species

type
`basic:string`

required
True

disabled
False

hidden
False

build

label
Build

type
`basic:string`

required
True

disabled
False

hidden
False

BWA-MEM2 index files

`data:index:bwamem2:upload-bwamem2-index` (*basic:file* **ref_seq**, *basic:file* **index_name**, *basic:string* **species**, *basic:string* **build**) [Source: v1.0.0]

Import BWA-MEM2 index files.

Input arguments ref_seq

label
Reference sequence (nucleotide FASTA)

type
`basic:file`

required
True

disabled
False

hidden
False

index_name

label
BWA-MEM2 index files

type
`basic:file`

required
True

disabled
False

hidden
False

species

label
Species

type
`basic:string`

description
Select a species name from the dropdown menu or write a custom species name in the species field. For sequences that are not related to any particular species (e.g. adapters file), you can select the value Other.

required
True

disabled
False

hidden
False

choices

- Homo sapiens: `Homo sapiens`
- Mus musculus: `Mus musculus`
- Rattus norvegicus: `Rattus norvegicus`
- Macaca mulatta: `Macaca mulatta`
- Dictyostelium discoideum: `Dictyostelium discoideum`
- Other: `Other`

build

label
Genome build

type
`basic:string`

required
True

disabled
False

hidden
False

Output results index

label
BWA-MEM2 index

type
`basic:dir`

required
True

disabled
False

hidden
False

fastagz

label
FASTA file (compressed)

type
`basic:file`

required
True

disabled
False

hidden
False

fasta

label
FASTA file

type
`basic:file`

required
True

disabled
False

hidden
False

fai

label
FASTA file index

type
`basic:file`

required
True

disabled
False

hidden
False

species

label
Species

type
`basic:string`

required
True

disabled
False

hidden
False

build

label
Build

type
`basic:string`

required
True

disabled
False

hidden
False

Bam split

`data:alignment:bam:primarybam-split` (*data:alignment:bam* **bam**, *data:sam:header* **header**, *data:sam:header* **header2**) [Source: v0.9.1]

Split hybrid bam file into two bam files.

Input arguments bam

label
Hybrid alignment bam

type
`data:alignment:bam`

header

label
Primary header sam file (optional)

type
`data:sam:header`

description
If no header file is provided, the headers will be extracted from the hybrid alignment bam file.

required
False

header2

label
Secondary header sam file (optional)

type
`data:sam:header`

description
If no header file is provided, the headers will be extracted from the hybrid alignment bam file.

required
False

Output results bam

label
Uploaded file

type
`basic:file`

bai

label
Index BAI

type
`basic:file`

species

label
Species

type
`basic:string`

build

label
Build

type
`basic:string`

Bamclipper

`data:alignment:bam:bamclipped:bamclipper` (*data:alignment:bam* **alignment**, *data:bedpe* **bedpe**, *basic:boolean* **skip**) [Source: v1.5.1]

Remove primer sequence from BAM alignments by soft-clipping.

This process is a wrapper for bamclipper which can be found at https://github.com/tommyau/bamclipper.

Input arguments alignment

label
Alignment BAM file

type
`data:alignment:bam`

required
True

disabled
False

hidden
False

bedpe

label
BEDPE file

type
`data:bedpe`

required
False

disabled
False

hidden
False

skip

label
Skip Bamclipper step

type
`basic:boolean`

description
Use this option to skip Bamclipper step.

required
True

disabled
False

hidden
False

default
`False`

Output results bam

label
Clipped BAM file

type
`basic:file`

required
True

disabled
False

hidden
False

bai

label
Index of clipped BAM file

type
`basic:file`

required
True

disabled
False

hidden
False

stats

label
Alignment statistics

type
`basic:file`

required
True

disabled
False

hidden
False

species

label
Species

type
`basic:string`

required
True

disabled
False

hidden
False

build

label
Build

type
`basic:string`

required
True

disabled
False

hidden
False

Bamliquidator

`data:bam:plot:bamliquidatorbamliquidator` (*basic:string* **analysis_type**, *list:data:alignment:bam* **bam**, *basic:string* **cell_type**, *basic:integer* **bin_size**, *data:annotation:gtf* **regions_gtf**, *data:bed* **regions_bed**, *basic:integer* **extension**, *basic:string* **sense**, *basic:boolean* **skip_plot**, *list:basic:string* **black_list**, *basic:integer* **threads**)[Source: v0.3.3]

Set of tools for analyzing the density of short DNA sequence read alignments in the BAM file format.

Input arguments analysis_type

label
Analysis type

type
`basic:string`

default
`bin`

choices

- Bin mode: `bin`
- Region mode: `region`
- BED mode: `bed`

bam

label
BAM File

type
`list:data:alignment:bam`

cell_type

label
Cell type

type
`basic:string`

default
`cell_type`

bin_size

label
Bin size

type
`basic:integer`

description
Number of base pairs in each bin. The smaller the bin size the longer the runtime and the larger the data files. Default is 100000.

required
False

hidden
analysis_type != 'bin'

regions_gtf

label
Region gff file / Annotation file (.gff|.gtf)

type
`data:annotation:gtf`

required
False

hidden
analysis_type != 'region'

regions_bed

label
Region bed file / Annotation file (.bed)

type
`data:bed`

required
False

hidden
analysis_type != 'bed'

extension

label
Extension

type
`basic:integer`

description
Extends reads by number of bp

default
`200`

sense

label
Mapping strand to gff file

type
`basic:string`

default
`.`

choices

- Forward: `+`
- Reverse: `-`
- Both: `.`

skip_plot

label
Skip plot

type
`basic:boolean`

required
False

black_list

label
Black list

type
`list:basic:string`

description
One or more chromosome patterns to skip during bin liquidation. Default is to skip any chromosomes that contain any of the following substrings chrUn _random Zv9_ _hap.

required
False

threads

label
Threads

type
`basic:integer`

description
Number of threads to run concurrently during liquidation.

default
1

Output results analysis_type

label
Analysis type

type
`basic:string`

hidden
True

output_dir

label
Output directory

type
`basic:file`

counts

label
Counts HDF5 file

type
`basic:file`

matrix

label
Matrix file

type
`basic:file`

required
False

hidden
analysis_type != 'region'

summary

label
Summary file

type
`basic:file:html`

required
False

hidden
analysis_type != 'bin'

Bamplot

`data:bam:plot:bamplotbamplot` (*basic:string* **genome**, *data:annotation:gtf* **input_gff**, *basic:string* **input_region**, *list:data:alignment:bam* **bam**, *basic:integer* **stretch_input**, *basic:string* **color**, *basic:string* **sense**, *basic:integer* **extension**, *basic:boolean* **rpm**, *basic:string* **yscale**, *list:basic:string* **names**, *basic:string* **plot**, *basic:string* **title**, *basic:string* **scale**, *list:data:bed* **bed**, *basic:boolean* **multi_page**) [Source: v1.4.3]

Plot a single locus from a bam.

Input arguments genome

label
Genome

type
`basic:string`

choices

- HG19: `HG19`
- HG18: `HG18`
- MM8: `MM8`
- MM9: `MM9`
- MM10: `MM10`
- RN6: `RN6`
- RN4: `RN4`

input_gff

label
Region string

type
`data:annotation:gtf`

description
Enter .gff file.

required
False

input_region

label
Region string

type
`basic:string`

description
Enter genomic region e.g. chr1:+:1-1000.

required
False

bam

label
Bam

type
`list:data:alignment:bam`

description
bam to plot from

required
False

stretch_input

label
Stretch-input

type
`basic:integer`

description
Stretch the input regions to a minimum length in bp, e.g. 10000 (for 10kb).

required
False

color

label
Color

type
`basic:string`

description
Enter a colon separated list of colors e.g. 255,0,0:255,125,0, default samples the rainbow.

default
`255,0,0:255,125,0`

sense

label
Sense

type
`basic:string`

description
Map to forward, reverse or'both strands. Default maps to both.

default
`both`

choices

- Forward: `forward`
- Reverse: `reverse`
- Both: `both`

extension

label
Extension

type
`basic:integer`

description
Extends reads by n bp. Default value is 200bp.

default
`200`

rpm

label
rpm

type
`basic:boolean`

description
Normalizes density to reads per million (rpm) Default is False.

required
False

yscale

label
y scale

type
`basic:string`

description
Choose either relative or uniform y axis scaling. Default is relative scaling.

default
`relative`

choices

- relative: `relative`
- uniform: `uniform`

names

label
Names

type
`list:basic:string`

description
Enter a comma separated list of names for your bams.

required
False

plot

label
Single or multiple polt

type
`basic:string`

description
Choose either all lines on a single plot or multiple plots.

default
`merge`

choices

- single: `single`
- multiple: `multiple`
- merge: `merge`

title

label
Title

type
`basic:string`

description
Specify a title for the output plot(s), default will be the coordinate region.

default
`output`

scale

label
Scale

type
`basic:string`

description
Enter a comma separated list of multiplicative scaling factors for your bams. Default is none.

required
False

bed

label
Bed

type
`list:data:bed`

description
Add a space-delimited list of bed files to plot.

required
False

multi_page

label
Multi page

type
`basic:boolean`

description
If flagged will create a new pdf for each region.

default
`False`

Output results plot

label
region plot

type
`basic:file`

BaseQualityScoreRecalibrator

`data:alignment:bam:bqsr:bqsr` (*data:alignment:bam* **bam**, *data:seq:nucleotide* **reference**, *list:data:variants:vcf* **known_sites**, *data:bed* **intervals**, *basic:string* **read_group**, *basic:string* **validation_stringency**, *basic:boolean* **use_original_qualities**, *basic:integer* **java_gc_threads**, *basic:integer* **max_heap_size**) [Source: v2.5.1]

A two pass process of BaseRecalibrator and ApplyBQSR from GATK.

See GATK website for more information on BaseRecalibrator.

It is possible to modify read group using GATK's AddOrReplaceGroups through Replace read groups in BAM (``read_group``) input field.

Input arguments bam

label
BAM file containing reads

type
`data:alignment:bam`

required
True

disabled
False

hidden
False

reference

label
Reference genome file

type
`data:seq:nucleotide`

required
True

disabled
False

hidden
False

known_sites

label
List of known sites of variation

type
`list:data:variants:vcf`

required
True

disabled
False

hidden
False

intervals

label
One or more genomic intervals over which to operate.

type
`data:bed`

description
This field is optional, but it can speed up the process by restricting calculations to specific genome regions.

required
False

disabled
False

hidden
False

read_group

label
Replace read groups in BAM

type
`basic:string`

description
Replace read groups in a BAM file.This argument enables the user to replace all read groups in the INPUT file with a single new read group and assign all reads to this read group in the OUTPUT BAM file. Addition or replacement is performed using Picard's AddOrReplaceReadGroups tool. Input should take the form of -name=value delimited by a ";", e.g. "-ID=1;-LB=GENIALIS;-PL=ILLUMINA;-PU=BARCODE;-SM=SAMPLENAME1". See tool's documentation for more information on tag names. Note that PL, LB, PU and SM are require fields. See caveats of rewriting read groups in the documentation.

required
True

disabled
False

hidden
False

default

validation_stringency

label
Validation stringency

type
`basic:string`

description
Validation stringency for all SAM files read by this program. Setting stringency to SILENT can improve performance when processing a BAM file in which variable-length data (read, qualities, tags) do not otherwise need to be decoded. Default is STRICT. This setting is used in BaseRecalibrator and ApplyBQSR processes.

required
True

disabled
False

hidden
False

default
`STRICT`

choices

- STRICT: `STRICT`
- LENIENT: `LENIENT`
- SILENT: `SILENT`

advanced.use_original_qualities

label
Use the base quality scores from the OQ tag

type
`basic:boolean`

description
This flag tells GATK to use the original base qualities (that were in the data before BQSR/recalibration) which are stored in the OQ tag, if they are present, rather than use the post-recalibration quality scores. If no OQ tag is present for a read, the standard qual score will be used.

required
True

disabled
False

hidden
False

default
`False`

advanced.java_gc_threads

label
Java ParallelGCThreads

type
`basic:integer`

description
Sets the number of threads used during parallel phases of the garbage collectors.

required
True

disabled
False

hidden
False

default
`2`

advanced.max_heap_size

label
Java maximum heap size (Xmx)

type
`basic:integer`

description
Set the maximum Java heap size (in GB).

required
True

disabled
False

hidden
False

default
12

Output results bam

label
Base quality score recalibrated BAM file

type
basic:file

required
True

disabled
False

hidden
False

bai

label
Index of base quality score recalibrated BAM file

type
basic:file

required
True

disabled
False

hidden
False

stats

label
Alignment statistics

type
basic:file

required
True

disabled
False

hidden
False

species

label
Species

type
basic:string

required
True

disabled
False

hidden
False

build

label
Build

type
`basic:string`

required
True

disabled
False

hidden
False

recal_table

label
Recalibration tabled

type
`basic:file`

required
True

disabled
False

hidden
False

BaseSpace file

`data:file:basespace-file-import` (*basic:string* **file_id**, *basic:secret* **access_token_secret**, *basic:string* **output**, *basic:integer* **tries**, *basic:boolean* **verbose**) [Source: v1.5.1]

Import a file from Illumina BaseSpace.

Input arguments file_id

label
BaseSpace file ID

type
`basic:string`

required
True

disabled
False

hidden
False

access_token_secret

label
BaseSpace access token

type
`basic:secret`

description
BaseSpace access token secret handle needed to download the file.

required
True

disabled
False

hidden
False

advanced.output

label
Output

type
`basic:string`

description
Sets what is printed to standard output. Argument 'Full' outputs everything, argument 'Filename' outputs only file names of downloaded files.

required
True

disabled
False

hidden
False

default
`filename`

choices

- Full: `full`
- Filename: `filename`

advanced.tries

label
Tries

type
`basic:integer`

description
Number of tries to download a file before giving up.

required
True

disabled
False

hidden
False

default
`3`

advanced.verbose

label
Verbose

type
`basic:boolean`

description
Print detailed exception information to standard output when error occurs. Output argument had no effect on this argument.

required
True

disabled
False

hidden
False

default
`False`

Output results file

label
File with reads

type
`basic:file`

required
True

disabled
False

hidden
False

Bedtools bamtobed

data:bedpe:bedtools-bamtobed (*data:alignment:bam* **alignment**)[Source: v1.3.1]

Takes in a BAM file and calculates a normalization factor in BEDPE format.

Done by sorting with Samtools and transformed with Bedtools.

Input arguments alignment

label
Alignment BAM file

type
data:alignment:bam

required
True

disabled
False

hidden
False

Output results bedpe

label
BEDPE file

type
basic:file

required
True

disabled
False

hidden
False

species

label
Species

type
basic:string

required
True

disabled
False

hidden
False

build

label
Build

type
`basic:string`

required
True

disabled
False

hidden
False

Beta Cut & Run workflow

`data:workflow:cutnrun:workflow-cutnrun-beta` (*data:reads:fastq:paired* **reads**, *basic:integer* **quality**, *basic:integer* **nextseq**, *basic:integer* **min_length**, *list:basic:string* **adapter_1**, *list:basic:string* **adapter_2**, *data:seq:nucleotide* **adapter_file_1**, *data:seq:nucleotide* **adapter_file_2**, *basic:string* **universal_adapter**, *basic:integer* **stringency**, *basic:decimal* **error_rate**, *data:index:bowtie2* **genome**, *data:index:bowtie2* **spikein_genome**, *basic:string* **alignment_mode**, *basic:string* **speed**, *basic:boolean* **dovetail**, *basic:boolean* **rep_se**, *basic:integer* **minins**, *basic:integer* **maxins**, *basic:boolean* **discordantly**, *basic:boolean* **no_unal**, *basic:boolean* **skip_norm**, *basic:decimal* **scale**, *basic:boolean* **downsample_reads**, *basic:integer* **n_reads**, *basic:boolean* **remove_duplicates**) [Source: v2.0.0]

Beta Cut & Run workflow.

Analysis of samples processed for high resolution mapping of DNA binding sites using targeted nuclease strategy. The process is named CUT&RUN, which stands for Cleavage Under Target and Release Using Nuclease. Workflow includes steps of trimming the reads with trimgalore, aligning them using bowtie2 to target species genome as well as a spike-in genome (optional). Aligned reads are processed to produce bigwig files to be viewed in a genome browser.

Input arguments reads

label
Input Reads (FASTQ)

type
`data:reads:fastq:paired`

description
Paired-end reads in FASTQ file.

required
True

disabled
False

hidden
False

trimming_options.quality

label
Quality cutoff

type
`basic:integer`

description
Trim low-quality ends from reads based on Phred score. Default: 20.

required
True

disabled
False

hidden
False

default
`20`

trimming_options.nextseq

label
NextSeq/NovaSeq trim cutoff

type
`basic:integer`

description
NextSeq/NovaSeq-specific quality trimming. Trims also dark cycles appearing as high-quality G bases. This will set a specific quality cutoff, but qualities of G bases are ignored. This can not be used with Quality cutoff and will override it.

required
False

disabled
False

hidden
False

trimming_options.min_length

label
Minimum length after trimming

type
`basic:integer`

description
Discard reads that became shorter than selected length because of either quality or adapter trimming. Both reads of a read-pair need to be longer than the specified length to be printed out to validated paired-end files. A value of 0 disables filtering based on length. Default: 20.

required
True

disabled
False

hidden
False

default
`20`

trimming_options.adapter_options.adapter_1

label
Read 1 adapter sequence

type
`list:basic:string`

description
Adapter sequences to be trimmed. Also see universal adapters field for predefined adapters. This is mutually exclusive with Read 1 adapters file and Universal adapters.

required
False

disabled
False

hidden
False

default
`[]`

trimming_options.adapter_options.adapter_2

label
Read 2 adapter sequence

type
`list:basic:string`

description
Optional adapter sequence to be trimmed off read 2 of paired-end files. This is mutually exclusive with Read 2 adapters file and Universal adapters.

required
False

disabled
False

hidden
False

default
`[]`

trimming_options.adapter_options.adapter_file_1

label
Read 1 adapters file

type
`data:seq:nucleotide`

description
This is mutually exclusive with Read 1 adapters and Universal adapters.

required
False

disabled
False

hidden
False

trimming_options.adapter_options.adapter_file_2

label
Read 2 adapters file

type
`data:seq:nucleotide`

description
This is mutually exclusive with Read 2 adapters and Universal adapters.

required
False

disabled
False

hidden
False

trimming_options.adapter_options.universal_adapter

label
Universal adapters

type
`basic:string`

description
Instead of default detection use specific adapters. Use 13bp of the Illumina universal adapter, 12bp of the Nextera adapter or 12bp of the Illumina Small RNA 3' Adapter. Selecting to trim smallRNA adapters will also lower the min length value to 18bp. If smallRNA libraries are paired-end, then Read 2 adapter will be set to the Illumina small RNA 5' adapter automatically (GATCGTCGGACT) unless defined explicitly. This is mutually exclusive with manually defined adapters and adapter files.

required
False

disabled
False

hidden
False

choices

- Illumina: `--illumina`
- Nextera: `--nextera`
- Illumina small RNA: `--small_rna`

trimming_options.adapter_options.stringency

label
Overlap with adapter sequence required to trim

type
`basic:integer`

description
Defaults to a very stringent setting of 1, i.e. even a single base pair of overlapping sequence will be trimmed of the 3' end of any read.

required
True

disabled
False

hidden
False

default
`1`

trimming_options.adapter_options.error_rate

label
Maximum allowed error rate

type
`basic:decimal`

description
Number of errors divided by the length of the matching region. Default: 0.1.

required
True

disabled
False

hidden
False

default
`0.1`

alignment_options.genome

label
Species genome

type
`data:index:bowtie2`

required
True

disabled
False

hidden
False

alignment_options.spikein_genome

label
Spike-in genome

type
`data:index:bowtie2`

required
False

disabled
normalization_options.skip_norm == true

hidden
False

alignment_options.alignment_mode

label
Alignment mode

type
`basic:string`

description
Local: Some characters may be omitted ('soft clipped') from the ends in order to achieve the greatest possible alignment score. End-to-end: Option without any trimming (or 'soft clipping') of bases from either end. This option is enabled by default and is suitable if reads have been clipped beforehand.

required
True

disabled
False

hidden
False

default
`--end-to-end`

choices

- Local: `--local`
- End-to-end: `--end-to-end`

alignment_options.speed

label
Speed vs. Sensitivity

type
`basic:string`

description
Setting for aligning fast or accurately. Default: Very sensitive.

required
True

disabled
False

hidden
False

default
`--very-sensitive`

choices

- Very fast: `--very-fast`
- Fast: `--fast`
- Sensitive: `--sensitive`
- Very sensitive: `--very-sensitive`

alignment_options.pe_options.dovetail

label
Dovetail

type
`basic:boolean`

description
If the mates dovetail, it implies that if the alignment of one mate extends beyond the starting point of the other, it results in the incorrect mate initiating upstream. This condition is considered concordant. Default: True.

required
True

disabled
False

hidden
False

default
`True`

alignment_options.pe_options.rep_se

label
Report single ended

type
`basic:boolean`

description
If paired alignment cannot be found, Bowtie2 tries to find alignments for the individual mates. Default: False.

required
True

disabled
False

hidden
False

default
`False`

alignment_options.pe_options.minins

label
Minimal distance

type
`basic:integer`

description
The minimum fragment length (–minins) for valid paired-end alignments. Default: 10.

required
True

disabled
False

hidden
False

default
`10`

alignment_options.pe_options.maxins

label
Maximal distance

type
`basic:integer`

description
The maximum fragment length (–maxins) for valid paired-end alignments. Default: 700.

required
True

disabled
False

hidden
False

default
`700`

alignment_options.pe_options.discordantly

label
Report discordantly matched read

type
`basic:boolean`

description
If both mates have unique alignments, but the alignments do not match paired-end expectations (orientation and relative distance), alignment will still be reported. Useful for detecting structural variations. Default: False.

required
True

disabled
False

hidden
False

default
`False`

alignment_options.output_options.no_unal

label
Suppress SAM records for unaligned reads

type
`basic:boolean`

description
When enabled, suppress SAM records for unaligned reads. Default: True.

required
True

disabled
False

hidden
False

default
`True`

normalization_options.skip_norm

label
Skip normalization

type
`basic:boolean`

description
Skip the spike-in normalization step of BigWig output. Use this if you don't provide a spike-in. Default: False.

required
True

disabled
False

hidden
False

default
`False`

normalization_options.scale

label
Scale factor

type
`basic:decimal`

description
Magnitude of the scale factor. The scaling factor is calculated by dividing the scale with the number of features in BEDPE (scale/(number of features)). Default: 10000.

required
True

disabled
normalization_options.skip_norm == true

hidden
False

default
`10000`

downsampling_options.downsample_reads

label
Downsample reads

type
`basic:boolean`

description
Option to downsample reads before trimming. Default: True

required
True

disabled
False

hidden
False

default
`True`

downsampling_options.n_reads

label
Number of reads to downsample

type
`basic:integer`

description
Number of reads to downsample from the input FASTQ file. Default: 10M.

required
True

disabled
downsampling_options.downsample_reads == false

hidden
False

default
`10000000`

deduplication_options.remove_duplicates

label
Remove duplicates

type
`basic:boolean`

description
Option on how to handle duplicate reads. True: Mark and remove duplicate reads. False: Only

mark duplicate reads. Note that this option is only available for species genome. In case of spike-in genome, duplicate reads are always removed. Default: False.

required
True

disabled
False

hidden
False

default
`False`

Output results

Bisulfite conversion rate

`data:wgbs:bsrate:bs-conversion-rate` (*data:alignment:bam:walt* **mr**, *basic:boolean* **skip**, *data:seq:nucleotide* **sequence**, *basic:boolean* **count_all**, *basic:integer* **read_length**, *basic:decimal* **max_mismatch**, *basic:boolean* **a_rich**) [Source: v1.3.1]

Estimate bisulfite conversion rate in a control set.

The program bsrate included in [Methpipe] (https://github.com/smithlabcode/methpipe) will estimate the bisulfite conversion rate.

Input arguments mr

label
Aligned reads from bisulfite sequencing

type
`data:alignment:bam:walt`

description
Bisulfite specifc alignment such as WALT is required as .mr file type is used. Duplicatesshould be removed to reduce any bias introduced by incomplete conversion on PCR duplicatereads.

required
True

disabled
False

hidden
False

skip

label
Skip Bisulfite conversion rate step

type
`basic:boolean`

description
Bisulfite conversion rate step can be skipped.

required
True

disabled
False

hidden
False

default
`False`

sequence

label
Unmethylated control sequence

type
`data:seq:nucleotide`

description
Separate unmethylated control sequence FASTA file is required to estimate bisulfiteconversion rate.

required
False

disabled
False

hidden
False

count_all

label
Count all cytosines including CpGs

type
`basic:boolean`

required
True

disabled
False

hidden
False

default
`True`

read_length

label
Average read length

type
`basic:integer`

required
True

disabled
False

hidden
False

default
`150`

max_mismatch

label
Maximum fraction of mismatches

type
`basic:decimal`

required
False

disabled
False

hidden
False

a_rich

label
Reads are A-rich

type
`basic:boolean`

required
True

disabled
False

hidden
False

default
`False`

Output results report

label
Bisulfite conversion rate report

type
`basic:file`

required
True

disabled
False

hidden
False

Bowtie (Dicty)

`data:alignment:bam:bowtie1alignment-bowtie` (*data:index:bowtie* **genome**, *data:reads:fastq* **reads**, *basic:string* **mode**, *basic:integer* **m**, *basic:integer* **l**, *basic:boolean* **use_se**, *basic:integer* **trim_5**, *basic:integer* **trim_3**, *basic:integer* **trim_nucl**, *basic:integer* **trim_iter**, *basic:string* **r**) [Source: v2.5.2]

An ultrafast memory-efficient short read aligner.

Input arguments genome

label
Reference genome

type
`data:index:bowtie`

reads

label
Reads

type
`data:reads:fastq`

mode

label
Alignment mode

type
`basic:string`

description
When the -n option is specified (which is the default), bowtie determines which alignments are valid according to the following policy, which is similar to Maq's default policy. 1. Alignments may have no more than N mismatches (where N is a number 0-3, set with -n) in the first L bases (where L is a number 5 or greater, set with -l) on the high-quality (left) end of the read. The first L bases are called the "seed". 2. The sum of the Phred quality values at all mismatched positions (not just in the seed) may not exceed E (set with -e). Where qualities are unavailable (e.g. if the reads are from a FASTA file), the Phred quality defaults to 40. In -v mode, alignments may have no more than V mismatches, where V may be a number from 0 through 3 set using the -v option. Quality values are ignored. The -v option is mutually exclusive with the -n option.

default
`-n`

choices

- Use qualities (-n): `-n`
- Use mismatches (-v): `-v`

m

label
Allowed mismatches

type
`basic:integer`

description
When used with "Use qualities (-n)" it is the maximum number of mismatches permitted in the "seed", i.e. the first L base pairs of the read (where L is set with -l/–seedlen). This may be 0, 1, 2 or 3 and the default is 2 When used with "Use mismatches (-v)" report alignments with at most <int> mismatches.

default
`2`

l

label
Seed length (for -n only)

type
`basic:integer`

description
Only for "Use qualities (-n)". Seed length (-l) is the number of bases on the high-quality end of the read to which the -n ceiling applies. The lowest permitted setting is 5 and the default is 28. bowtie is faster for larger values of -l.

default
`28`

use_se

label
Map as single-ended (for paired end reads only)

type
`basic:boolean`

description
If this option is selected paired-end reads will be mapped as single-ended.

default
`False`

start_trimming.trim_5

label
Bases to trim from 5'

type
`basic:integer`

description
Number of bases to trim from from 5' (left) end of each read before alignment

default
`0`

start_trimming.trim_3

label
Bases to trim from 3'

type
`basic:integer`

description
Number of bases to trim from from 3' (right) end of each read before alignment

default
`0`

trimming.trim_nucl

label
Bases to trim

type
`basic:integer`

description
Number of bases to trim from 3' end in each iteration.

default
`2`

trimming.trim_iter

label
Iterations

type
`basic:integer`

description
Number of iterations.

default
`0`

reporting.r

label
Reporting mode

type
`basic:string`

description
Report up to <int> valid alignments per read or pair (-k) (default: 1). Validity of alignments is determined by the alignment policy (combined effects of -n, -v, -l, and -e). If more than one valid alignment exists and the –best and –strata options are specified, then only those alignments belonging to the best alignment "stratum" will be reported. Bowtie is designed to be very fast for small -k but bowtie can become significantly slower as -k increases. If you would like to use Bowtie for larger values of -k, consider building an index with a denser suffix-array sample, i.e. specify a smaller -o/–offrate when invoking bowtie-build for the relevant index (see the Performance tuning section for details).

default
`-a -m 1 --best --strata`

choices

- Report unique alignments: `-a -m 1 --best --strata`
- Report all alignments: `-a --best`
- Report all alignments in the best stratum: `-a --best --strata`

Output results bam

label
Alignment file

type
`basic:file`

description
Position sorted alignment

bai

label
Index BAI

type
`basic:file`

unmapped

label
Unmapped reads

type
`basic:file`

required
False

stats

label
Statistics

type
`basic:file`

species

label
Species

type
`basic:string`

build

label
Build

type
`basic:string`

Bowtie genome index

`data:index:bowtie:bowtie-index` (*data:seq:nucleotide* **ref_seq**)[Source: v1.2.1]

Create Bowtie genome index.

Input arguments ref_seq

label
Reference sequence (nucleotide FASTA)

type
`data:seq:nucleotide`

required
True

disabled
False

hidden
False

Output results index

label
Bowtie index

type
`basic:dir`

required
True

disabled
False

hidden
False

fastagz

label
FASTA file (compressed)

type
`basic:file`

required
True

disabled
False

hidden
False

fasta

label
FASTA file

type
`basic:file`

required
True

disabled
False

hidden
False

fai

label
FASTA file index

type
`basic:file`

required
True

disabled
False

hidden
False

species

label
Species

type
`basic:string`

required
True

disabled
False

hidden
False

build

label
Build

type
`basic:string`

required
True

disabled
False

hidden
False

Bowtie2

`data:alignment:bam:bowtie2alignment-bowtie2` (*data:index:bowtie2* **genome**, *data:reads:fastq* **reads**, *basic:string* **mode**, *basic:string* **speed**, *basic:boolean* **use_se**, *basic:boolean* **discordantly**, *basic:boolean* **rep_se**, *basic:integer* **minins**, *basic:integer* **maxins**, *basic:boolean* **no_overlap**, *basic:boolean* **dovetail**, *basic:integer* **N**, *basic:integer* **L**, *basic:integer* **gbar**, *basic:string* **mp**, *basic:string* **rdg**, *basic:string* **rfg**, *basic:string* **score_min**, *basic:integer* **trim_5**, *basic:integer* **trim_3**, *basic:integer* **trim_iter**, *basic:integer* **trim_nucl**, *basic:string* **rep_mode**, *basic:integer* **k_reports**, *basic:boolean* **no_unal**) [Source: v2.8.2]

Bowtie is an ultrafast, memory-efficient short read aligner. It aligns short DNA sequences (reads) to the human genome at a rate of over 25 million 35-bp reads per hour. Bowtie indexes the genome with a Burrows-Wheeler index to keep its memory footprint small–typically about 2.2 GB for the human genome (2.9 GB for paired-end). See [here](http://bowtie-bio.sourceforge.net/index.shtml) for more information.

Input arguments genome

label
Reference genome

type
`data:index:bowtie2`

reads

label
Reads

type
`data:reads:fastq`

mode

label
Alignment mode

type
`basic:string`

description
End to end: Bowtie 2 requires that the entire read align from one end to the other, without any trimming (or "soft clipping") of characters from either end. local: Bowtie 2 does not require that the entire read align from one end to the other. Rather, some characters may be omitted ("soft clipped") from the ends in order to achieve the greatest possible alignment score.

default
`--end-to-end`

choices

- end to end mode: `--end-to-end`
- local: `--local`

speed

label
Speed vs. Sensitivity

type
`basic:string`

description
A quick setting for aligning fast or accurately. This option is a shortcut for parameters as follows:

For –end-to-end: –very-fast -D 5 -R 1 -N 0 -L 22 -i S,0,2.50 –fast -D 10 -R 2 -N 0 -L 22 -i S,0,2.50 –sensitive -D 15 -R 2 -N 0 -L 22 -i S,1,1.15 (default) –very-sensitive -D 20 -R 3 -N 0 -L 20 -i S,1,0.50

For –local: –very-fast-local -D 5 -R 1 -N 0 -L 25 -i S,1,2.00 –fast-local -D 10 -R 2 -N 0 -L 22 -i S,1,1.75 –sensitive-local -D 15 -R 2 -N 0 -L 20 -i S,1,0.75 (default) –very-sensitive-local -D 20 -R 3 -N 0 -L 20 -i S,1,0.50

required
False

choices

- Very fast: `--very-fast`
- Fast: `--fast`
- Sensitive: `--sensitive`
- Very sensitive: `--very-sensitive`

PE_options.use_se

label
Map as single-ended (for paired-end reads only)

type
`basic:boolean`

description
If this option is selected paired-end reads will be mapped as single-ended and other paired-end options are ignored.

default
`False`

PE_options.discordantly

label
Report discordantly matched read

type
`basic:boolean`

description
If both mates have unique alignments, but the alignments do not match paired-end expectations (orientation and relative distance) then alignment will be reported. Useful for detecting structural variations.

default
`True`

PE_options.rep_se

label
Report single ended

type
`basic:boolean`

description
If paired alignment can not be found Bowtie2 tries to find alignments for the individual mates.

default
`True`

PE_options.minins

label
Minimal distance

type
`basic:integer`

description
The minimum fragment length for valid paired-end alignments. 0 imposes no minimum.

default
`0`

PE_options.maxins

label
Maximal distance

type
`basic:integer`

description
The maximum fragment length for valid paired-end alignments.

default
`500`

PE_options.no_overlap

label
Not concordant when mates overlap

type
`basic:boolean`

description
When true, it is considered not concordant when mates overlap at all. Defaul is false.

default
`False`

PE_options.dovetail

label
Dovetail

type
`basic:boolean`

description
If the mates "dovetail", that is if one mate alignment extends past the beginning of the other such that the wrong mate begins upstream, consider that to be concordant. Default: mates cannot dovetail in a concordant alignment.

default
`False`

alignment_options.N

label
Number of mismatches allowed in seed alignment (N)

type
`basic:integer`

description
Sets the number of mismatches to allowed in a seed alignment during multiseed alignment. Can be set to 0 or 1. Setting this higher makes alignment slower (often much slower) but increases sensitivity. Default: 0.

required
False

alignment_options.L

label
Length of seed substrings (L)

type
`basic:integer`

description
Sets the length of the seed substrings to align during multiseed alignment. Smaller values make alignment slower but more sensitive. Default: the –sensitive preset is used by default for end-to-end alignment and –sensitive-local for local alignment. See documentation for details.

required
False

alignment_options.gbar

label
Disallow gaps within positions (gbar)

type
`basic:integer`

description
Disallow gaps within <int> positions of the beginning or end of the read. Default: 4.

required
False

alignment_options.mp

label
Maximal and minimal mismatch penalty (mp)

type
`basic:string`

description
Sets the maximum (MX) and minimum (MN) mismatch penalties, both integers. A number less than or equal to MX and greater than or equal to MN is subtracted from the alignment score for each position where a read character aligns to a reference character, the characters do not match, and neither is an N. If –ignore-quals is specified, the number subtracted quals MX. Otherwise, the number subtracted is MN + floor((MX-MN)(MIN(Q, 40.0)/40.0)) where Q is the Phred quality value. Default for MX, MN: 6,2.

required
False

alignment_options.rdg

label
Set read gap open and extend penalties (rdg)

type
`basic:string`

description
Sets the read gap open (<int1>) and extend (<int2>) penalties. A read gap of length N gets a penalty of <int1> + N * <int2>. Default: 5,3.

required
False

alignment_options.rfg

label
Set reference gap open and close penalties (rfg)

type
`basic:string`

description
Sets the reference gap open (<int1>) and extend (<int2>) penalties. A reference gap of length N gets a penalty of <int1> + N * <int2>. Default: 5,3.

required
False

alignment_options.score_min

label
Minimum alignment score needed for "valid" alignment (score_min)

type
`basic:string`

description
Sets a function governing the minimum alignment score needed for an alignment to be considered "valid" (i.e. good enough to report). This is a function of read length. For instance, specifying L,0,-0.6 sets the minimum-score function to f(x) = 0 + -0.6 * x, where x is the read length. The default in –end-to-end mode is L,-0.6,-0.6 and the default in –local mode is G,20,8.

required
False

start_trimming.trim_5

label
Bases to trim from 5'

type
`basic:integer`

description
Number of bases to trim from from 5' (left) end of each read before alignment

default
`0`

start_trimming.trim_3

label
Bases to trim from 3'

type
`basic:integer`

description
Number of bases to trim from from 3' (right) end of each read before alignment

default
`0`

trimming.trim_iter

label
Iterations

type
`basic:integer`

description
Number of iterations.

default
`0`

trimming.trim_nucl

label
Bases to trim

type
`basic:integer`

description
Number of bases to trim from 3' end in each iteration.

default
`2`

reporting.rep_mode

label
Report mode

type
`basic:string`

description
Default mode: search for multiple alignments, report the best one; -k mode: search for one or more alignments, report each; -a mode: search for and report all alignments

default
`def`

choices

- Default mode: `def`
- -k mode: `k`
- -a mode (very slow): `a`

reporting.k_reports

label
Number of reports (for -k mode only)

type
`basic:integer`

description
Searches for at most X distinct, valid alignments for each read. The search terminates when it can't find more distinct valid alignments, or when it finds X, whichever happens first. default: 5

default
`5`

output_opts.no_unal

label
Suppress SAM records for unaligned reads

type
`basic:boolean`

description
When true, suppress SAM records for unaligned reads. Default is false.

default
`False`

Output results bam

label
Alignment file

type
`basic:file`

description
Position sorted alignment

bai

label
Index BAI

type
`basic:file`

unmapped

label
Unmapped reads

type
`basic:file`

required
False

stats

label
Statistics

type
`basic:file`

species

label
Species

type
`basic:string`

build

label
Build

type
`basic:string`

Bowtie2 genome index

data:index:bowtie2:bowtie2-index (*data:seq:nucleotide* **ref_seq**)[Source: v1.2.1]

Create Bowtie2 genome index.

Input arguments ref_seq

label
Reference sequence (nucleotide FASTA)

type
`data:seq:nucleotide`

required
True

disabled
False

hidden
False

Output results index

label
Bowtie2 index

type
`basic:dir`

required
True

disabled
False

hidden
False

fastagz

label
FASTA file (compressed)

type
`basic:file`

required
True

disabled
False

hidden
False

fasta

label
FASTA file

type
`basic:file`

required
True

disabled
False

hidden
False

fai

label
FASTA file index

type
`basic:file`

required
True

disabled
False

hidden
False

species

label
Species

type
`basic:string`

required
True

disabled
False

hidden
False

build

label
Build

type
`basic:string`

required
True

disabled
False

hidden
False

Calculate coverage (bamCoverage)

data:coverage:bigwig:calculate-bigwig (*data:alignment:bam* **alignment**, *data:bedpe* **bedpe**, *basic:decimal* **scale**, *basic:integer* **bin_size**) [Source: v2.0.1]

Calculate bigWig coverage track.

Deeptools bamCoverage takes an alignment of reads or fragments as input (BAM file) and generates a coverage track (bigWig) as output. The coverage is calculated as the number of reads per bin, where bins are short consecutive counting windows of a defined size. For more information is available in the [bamCoverage documentation](https://deeptools.readthedocs.io/en/latest/content/tools/bamCoverage.html).

Input arguments alignment

label
Alignment BAM file

type
`data:alignment:bam`

required
True

disabled
False

hidden
False

bedpe

label
BEDPE Normalization factor

type
`data:bedpe`

description
The BEDPE file describes disjoint genome features, such as structural variations or paired-end sequence alignments. It is used to estimate the scale factor [–scaleFactor].

required
False

disabled
False

hidden
False

scale

label
Scale for the normalization factor

type
`basic:decimal`

description
Magnitude of the scale factor. The scaling factor [–scaleFactor] is calculated by dividing the scale with the number of features in BEDPE (scale/(number of features)).

required
True

disabled
!bedpe

hidden
False

default
`10000`

bin_size

label
Bin size[–binSize]

type
`basic:integer`

description
Size of the bins (in bp) for the output bigWig file. A smaller bin size value will result in a higher resolution of the coverage track but also in a larger file size.

required
True

disabled
False

hidden
False

default
`50`

Output results bigwig

label
Coverage file (bigWig)

type
`basic:file`

required
True

disabled
False

hidden
False

species

label
Species

type
`basic:string`

required
True

disabled
False

hidden
False

build

label
Build

type
`basic:string`

required
True

disabled
False

hidden
False

Cell Ranger Count

`data:scexpression:10x:cellranger-count` (*data:screads:10x:* **reads**, *data:genomeindex:10x:* **genome_index**, *basic:string* **chemistry**, *basic:integer* **trim_r1**, *basic:integer* **trim_r2**, *basic:integer* **expected_cells**, *basic:integer* **force_cells**) [Source: v1.2.2]

Perform gene expression analysis.

Generate single cell feature counts for a single library. https://support.10xgenomics.com/single-cell-gene-expression/software/pipelines/latest/using/count

Input arguments reads

label
10x reads data object

type
`data:screads:10x:`

required
True

disabled
False

hidden
False

genome_index

label
10x genome index data object

type
`data:genomeindex:10x:`

required
True

disabled
False

hidden
False

chemistry

label
Chemistry

type
`basic:string`

description
Assay configuration. By default the assay configuration is detected automatically, which is the recommended mode. You should only specify chemistry if there is an error in automatic detection.

required
False

disabled
False

hidden
False

default
`auto`

choices

- auto: `auto`
- threeprime: `Single Cell 3'`
- fiveprime: `Single Cell 5'`
- SC3Pv1: `Single Cell 3' v1`
- SC3Pv2: `Single Cell 3' v2`
- SC3Pv3: `Single Cell 3' v3`
- C5P-PE: `Single Cell 5' paired-end`
- SC5P-R2: `Single Cell 5' R2-only`

trim_r1

label
Trim R1

type
`basic:integer`

description
Hard-trim the input R1 sequence to this length. Note that the length includes the Barcode and UMI sequences so do not set this below 26 for Single Cell 3' v2 or Single Cell 5'. This and "Trim R2" are useful for determining the optimal read length for sequencing.

required
False

disabled
False

hidden
False

trim_r2

label
Trim R2

type
`basic:integer`

description
Hard-trim the input R2 sequence to this length.

required
False

disabled
False

hidden
False

expected_cells

label
Expected number of recovered cells

type
`basic:integer`

required
True

disabled
False

hidden
False

default
`3000`

force_cells

label
Force cell number

type
`basic:integer`

description
Force pipeline to use this number of cells, bypassing the cell detection algorithm. Use this if the number of cells estimated by Cell Ranger is not consistent with the barcode rank plot.

required
False

disabled
False

hidden
False

Output results matrix_filtered

label
Matrix (filtered)

type
`basic:file`

required
True

disabled
False

hidden
False

genes_filtered

label
Genes (filtered)

type
`basic:file`

required
True

disabled
False

hidden
False

barcodes_filtered

label
Barcodes (filtered)

type
`basic:file`

required
True

disabled
False

hidden
False

matrix_raw

label
Matrix (raw)

type
`basic:file`

required
True

disabled
False

hidden
False

genes_raw

label
Genes (raw)

type
`basic:file`

required
True

disabled
False

hidden
False

barcodes_raw

label
Barcodes (raw)

type
`basic:file`

required
True

disabled
False

hidden
False

report

label
Report

type
`basic:file:html`

required
True

disabled
False

hidden
False

build

label
Build

type
`basic:string`

required
True

disabled
False

hidden
False

species

label
Species

type
`basic:string`

required
True

disabled
False

hidden
False

source

label
Gene ID source

type
`basic:string`

required
True

disabled
False

hidden
False

Cell Ranger Mkref

`data:genomeindex:10x:cellranger-mkref` (*data:seq:nucleotide:* **genome**, *data:annotation:gtf:* **annotation**) [Source: v2.1.3]

Reference preparation tool for 10x Genomics Cell Ranger.

Build a Cell Ranger-compatible reference from genome FASTA and gene GTF files. https://support.10xgenomics.com/single-cell-gene-expression/software/pipelines/latest/advanced/references

Input arguments genome

label
Reference genome

type
`data:seq:nucleotide:`

required
True

disabled
False

hidden
False

annotation

label
Annotation

type
`data:annotation:gtf:`

required
True

disabled
False

hidden
False

Output results genome_index

label
Indexed genome

type
`basic:dir`

required
True

disabled
False

hidden
False

build

label
Build

type
`basic:string`

required
True

disabled
False

hidden
False

species

label
Species

type
`basic:string`

required
True

disabled
False

hidden
False

source

label
Gene ID source

type
`basic:string`

required
True

disabled
False

hidden
False

ChIP-Seq (Gene Score)

`data:chipseq:genescorechipseq-genescore` (*data:chipseq:peakscore* **peakscore**, *basic:decimal* **fdr**, *basic:decimal* **pval**, *basic:decimal* **logratio**) [Source: v1.3.1]

Chip-Seq analysis - Gene Score (BCM)

Input arguments peakscore

label
PeakScore file

type
`data:chipseq:peakscore`

description
PeakScore file

fdr

label
FDR threshold

type
`basic:decimal`

description
FDR threshold value (default = 0.00005).

default
`5e-05`

pval

label
Pval threshold

type
`basic:decimal`

description
Pval threshold value (default = 0.00005).

default
`5e-05`

logratio

label
Log-ratio threshold

type
`basic:decimal`

description
Log-ratio threshold value (default = 2).

default
`2.0`

Output results genescore

label
Gene Score

type
`basic:file`

ChIP-Seq (Peak Score)

`data:chipseq:peakscorechipseq-peakscore` (*data:chipseq:callpeak:macs2* **peaks**, *data:bed* **bed**) [Source: v2.3.1]

Chip-Seq analysis - Peak Score (BCM)

Input arguments peaks

label
MACS2 results

type
`data:chipseq:callpeak:macs2`

description
MACS2 results file (NarrowPeak)

bed

label
BED file

type
`data:bed`

Output results peak_score

label
Peak Score

type
`basic:file`

ChIP-seq (MACS2)

`data:chipseq:batch:macs2macs2-batch` (*list:data:alignment:bam* **alignments**, *data:bed* **promoter**, *basic:boolean* **tagalign**, *basic:integer* **q_threshold**, *basic:integer* **n_sub**, *basic:boolean* **tn5**, *basic:integer* **shift**, *basic:string* **duplicates**, *basic:string* **duplicates_prepeak**, *basic:decimal* **qvalue**, *basic:decimal* **pvalue**, *basic:decimal* **pvalue_prepeak**, *basic:integer* **cap_num**, *basic:integer* **mfold_lower**, *basic:integer* **mfold_upper**, *basic:integer* **slocal**, *basic:integer* **llocal**, *basic:integer* **extsize**, *basic:integer* **shift**, *basic:integer* **band_width**, *basic:boolean* **nolambda**, *basic:boolean* **fix_bimodal**, *basic:boolean* **nomodel**, *basic:boolean* **nomodel_prepeak**, *basic:boolean* **down_sample**, *basic:boolean* **bedgraph**, *basic:boolean* **spmr**, *basic:boolean* **call_summits**, *basic:boolean* **broad**, *basic:decimal* **broad_cutoff**, *data:bed* **blacklist**, *basic:boolean* **calculate_enrichment**, *basic:integer* **profile_window**, *basic:string* **shift_size**) [Source: v1.5.1]

This process runs MACS2 in batch mode. MACS2 analysis is triggered for pairs of samples as defined using treatment-background sample relations. If there are no sample relations defined, each sample is treated individually for the MACS analysis.

Model-based Analysis of ChIP-Seq (MACS 2.0), is used to identify transcript factor binding sites. MACS 2.0 captures the influence of genome complexity to evaluate the significance of enriched ChIP regions, and MACS improves the spatial resolution of binding sites through combining the information of both sequencing tag position and orientation. It has also an option to link nearby peaks together in order to call broad peaks. See [here](https://github.com/taoliu/MACS/) for more information.

In addition to peak-calling, this process computes ChIP-Seq and ATAC-Seq QC metrics. Process returns a QC metrics report, fragment length estimation, and a deduplicated tagAlign file. QC report contains ENCODE 3 proposed QC metrics – [NRF](https://www.encodeproject.org/data-standards/terms/), [PBC bottlenecking coefficients, NSC, and RSC](https://genome.ucsc.edu/ENCODE/qualityMetrics.html#chipSeq).

Input arguments alignments

label
Aligned reads

type
`list:data:alignment:bam`

description
Select multiple treatment/background samples.

promoter

label
Promoter regions BED file

type
`data:bed`

description
BED file containing promoter regions (TSS+-1000 bp for example). Needed to get the number of peaks and reads mapped to promoter regions.

required
False

tagalign

label
Use tagAlign files

type
`basic:boolean`

description
Use filtered tagAlign files as case (treatment) and control (background) samples. If extsize parameter is not set, run MACS using input's estimated fragment length.

default
`True`

prepeakqc_settings.q_threshold

label
Quality filtering threshold

type
`basic:integer`

default
`30`

prepeakqc_settings.n_sub

label
Number of reads to subsample

type
`basic:integer`

default
`15000000`

prepeakqc_settings.tn5

label
Tn5 shifting

type
`basic:boolean`

description
Tn5 transposon shifting. Shift reads on "+" strand by 4 bp and reads on "-" strand by 5 bp.

default
`False`

prepeakqc_settings.shift

label
User-defined cross-correlation peak strandshift

type
`basic:integer`

description
If defined, SPP tool will not try to estimate fragment length but will use the given value as fragment length.

required
False

settings.duplicates

label
Number of duplicates

type
`basic:string`

description
It controls the MACS behavior towards duplicate tags at the exact same location – the same coordination and the same strand. The 'auto' option makes MACS calculate the maximum tags at the exact same location based on binomal distribution using 1e-5 as pvalue cutoff and the 'all' option keeps all the tags. If an integer is given, at most this number of tags will be kept at the same location. The default is to keep one tag at the same location.

required
False

hidden
tagalign

choices

- 1: `1`
- auto: `auto`
- all: `all`

settings.duplicates_prepeak

label
Number of duplicates

type
`basic:string`

description
It controls the MACS behavior towards duplicate tags at the exact same location – the same coordination and the same strand. The 'auto' option makes MACS calculate the maximum tags at the exact same location based on binomal distribution using 1e-5 as pvalue cutoff and the 'all' option keeps all the tags. If an integer is given, at most this number of tags will be kept at the same location. The default is to keep one tag at the same location.

required
False

hidden
!tagalign

default
`all`

choices

- 1: `1`
- auto: `auto`
- all: `all`

settings.qvalue

label
Q-value cutoff

type
`basic:decimal`

description
The q-value (minimum FDR) cutoff to call significant regions. Q-values are calculated from p-values using Benjamini-Hochberg procedure.

required
False

disabled
settings.pvalue && settings.pvalue_prepeak

settings.pvalue

label
P-value cutoff

type
`basic:decimal`

description
The p-value cutoff. If specified, MACS2 will use p-value instead of q-value cutoff.

required
False

disabled
settings.qvalue

hidden
tagalign

settings.pvalue_prepeak

label
P-value cutoff

type
`basic:decimal`

description
The p-value cutoff. If specified, MACS2 will use p-value instead of q-value cutoff.

disabled
settings.qvalue

hidden
!tagalign || settings.qvalue

default
`1e-05`

settings.cap_num

label
Cap number of peaks by taking top N peaks

type
`basic:integer`

description
To keep all peaks set value to 0.

disabled
settings.broad

default
`500000`

settings.mfold_lower

label
MFOLD range (lower limit)

type
`basic:integer`

description
This parameter is used to select the regions within MFOLD range of high-confidence enrichment ratio against background to build model. The regions must be lower than upper limit, and higher than the lower limit of fold enrichment. DEFAULT:10,30 means using all regions not too low (>10) and not too high (<30) to build paired-peaks model. If MACS can not find more than 100 regions to build model, it will use the –extsize parameter to continue the peak detection ONLY if –fix-bimodal is set.

required
False

settings.mfold_upper

label
MFOLD range (upper limit)

type
`basic:integer`

description
This parameter is used to select the regions within MFOLD range of high-confidence enrichment ratio against background to build model. The regions must be lower than upper limit, and higher than the lower limit of fold enrichment. DEFAULT:10,30 means using all regions not too low (>10) and not too high (<30) to build paired-peaks model. If MACS can not find more than 100 regions to build model, it will use the –extsize parameter to continue the peak detection ONLY if –fix-bimodal is set.

required
False

settings.slocal

label
Small local region

type
`basic:integer`

description
Slocal and llocal parameters control which two levels of regions will be checked around the peak regions to calculate the maximum lambda as local lambda. By default, MACS considers 1000 bp for small local region (–slocal), and 10000 bp for large local region (–llocal) which captures the bias from a long range effect like an open chromatin domain. You can tweak these according to your project. Remember that if the region is set too small, a sharp spike in the input data may kill the significant peak.

required
False

settings.llocal

label
Large local region

type
`basic:integer`

description
Slocal and llocal parameters control which two levels of regions will be checked around the peak regions to calculate the maximum lambda as local lambda. By default, MACS considers 1000 bp for small local region (–slocal), and 10000 bp for large local region (–llocal) which captures the bias from a long range effect like an open chromatin domain. You can tweak these according to your project. Remember that if the region is set too small, a sharp spike in the input data may kill the significant peak.

required
False

settings.extsize

label
extsize

type
`basic:integer`

description
While '–nomodel' is set, MACS uses this parameter to extend reads in 5'->3' direction to fix-sized fragments. For example, if the size of binding region for your transcription factor is 200 bp, and you want to bypass the model building by MACS, this parameter can be set as 200. This option is only valid when –nomodel is set or when MACS fails to build model and –fix-bimodal is on.

required
False

settings.shift

label
Shift

type
`basic:integer`

description
Note, this is NOT the legacy –shiftsize option which is replaced by –extsize! You can set an arbitrary shift in bp here. Please Use discretion while setting it other than default value (0). When –nomodel is set, MACS will use this value to move cutting ends (5') then apply –extsize from 5' to 3' direction to extend them to fragments. When this value is negative, ends will be moved toward 3'->5' direction, otherwise 5'->3' direction. Recommended to keep it as default 0 for ChIP-Seq datasets, or -1 * half of EXTSIZE together with –extsize option for detecting enriched cutting loci such as certain DNAseI-Seq datasets. Note, you can't set values other than 0 if format is BAMPE for paired-end data. Default is 0.

required
False

settings.band_width

label
Band width

type
`basic:integer`

description
The band width which is used to scan the genome ONLY for model building. You can set this parameter as the sonication fragment size expected from wet experiment. The previous side effect on the peak detection process has been removed. So this parameter only affects the model building.

required
False

settings.nolambda

label
Use backgroud lambda as local lambda

type
`basic:boolean`

description
With this flag on, MACS will use the background lambda as local lambda. This means MACS will not consider the local bias at peak candidate regions.

default
`False`

settings.fix_bimodal

label
Turn on the auto paired-peak model process

type
`basic:boolean`

description
Turn on the auto paired-peak model process. If it's set, when MACS failed to build paired model, it will use the nomodel settings, the '–extsize' parameter to extend each tag. If set, MACS will be terminated if paired-peak model has failed.

default
`False`

settings.nomodel

label
Bypass building the shifting model

type
`basic:boolean`

description
While on, MACS will bypass building the shifting model.

hidden
tagalign

default
`False`

settings.nomodel_prepeak

label
Bypass building the shifting model

type
`basic:boolean`

description
While on, MACS will bypass building the shifting model.

hidden
!tagalign

default
`True`

settings.down_sample

label
Down-sample

type
`basic:boolean`

description
When set to true, random sampling method will scale down the bigger sample. By default, MACS uses linear scaling. This option will make the results unstable and irreproducible since each time, random reads would be selected, especially the numbers (pileup, pvalue, qvalue) would change.

default
`False`

settings.bedgraph

label
Save fragment pileup and control lambda

type
`basic:boolean`

description
If this flag is on, MACS will store the fragment pileup, control lambda, -log10pvalue and -log10qvalue scores in bedGraph files. The bedGraph files will be stored in current directory named NAME+'_treat_pileup.bdg' for treatment data, NAME+'_control_lambda.bdg' for local lambda values from control, NAME+'_treat_pvalue.bdg' for Poisson pvalue scores (in -log10(pvalue) form), and NAME+'_treat_qvalue.bdg' for q-value scores from Benjamini-Hochberg-Yekutieli procedure.

default
`True`

settings.spmr

label
Save signal per million reads for fragment pileup profiles

type
`basic:boolean`

disabled
settings.bedgraph === false

default
`True`

settings.call_summits

label
Call summits

type
`basic:boolean`

description
MACS will now reanalyze the shape of signal profile (p or q-score depending on cutoff setting) to deconvolve subpeaks within each peak called from general procedure. It's highly recommended to detect adjacent binding events. While used, the output subpeaks of a big peak region will have the same peak boundaries, and different scores and peak summit positions.

default
`False`

settings.broad

label
Composite broad regions

type
`basic:boolean`

description
When this flag is on, MACS will try to composite broad regions in BED12 (a gene-model-like format) by putting nearby highly enriched regions into a broad region with loose cutoff. The broad region is controlled by another cutoff through –broad-cutoff. The maximum length of broad region length is 4 times of d from MACS.

disabled
settings.call_summits === true

default
`False`

settings.broad_cutoff

label
Broad cutoff

type
`basic:decimal`

description
Cutoff for broad region. This option is not available unless –broad is set. If -p is set, this is a p-value cutoff, otherwise, it's a q-value cutoff. DEFAULT = 0.1

required
False

disabled
settings.call_summits === true || settings.broad !== true

chipqc_settings.blacklist

label
Blacklist regions

type
`data:bed`

description
BED file containing genomic regions that should be excluded from the analysis.

required
False

chipqc_settings.calculate_enrichment

label
Calculate enrichment

type
`basic:boolean`

description
Calculate enrichment of signal in known genomic annotation. By default annotation is provided from the TranscriptDB package specified by genome bulid which should match one of the supported annotations (hg19, hg38, hg18, mm10, mm9, rn4, ce6, dm3). If annotation is not supported the analysis is skipped.

default
`False`

chipqc_settings.profile_window

label
Window size

type
`basic:integer`

description
An integer indicating the width of the window used for peak profiles. Peaks will be centered on their summits and include half of the window size upstream and half downstream of this point.

default
`400`

chipqc_settings.shift_size

label
Shift size

type
`basic:string`

description
Vector of values to try when computing optimal shift sizes. It should be specifeird as consecutive numbers vector with start:end

default
`1:300`

Output results

ChIP-seq (MACS2-ROSE2)

data:chipseq:batch:macs2macs2-rose2-batch (*list:data:alignment:bam* **alignments**, *data:bed* **promoter**, *basic:boolean* **tagalign**, *basic:integer* **q_threshold**, *basic:integer* **n_sub**, *basic:boolean* **tn5**, *basic:integer* **shift**, *basic:string* **duplicates**, *basic:string* **duplicates_prepeak**, *basic:decimal* **qvalue**, *basic:decimal* **pvalue**, *basic:decimal* **pvalue_prepeak**, *basic:integer* **cap_num**, *basic:integer* **mfold_lower**, *basic:integer* **mfold_upper**, *basic:integer* **slocal**, *basic:integer* **llocal**, *basic:integer* **extsize**, *basic:integer* **shift**, *basic:integer* **band_width**, *basic:boolean* **nolambda**, *basic:boolean* **fix_bimodal**, *basic:boolean* **nomodel**, *basic:boolean* **nomodel_prepeak**, *basic:boolean* **down_sample**, *basic:boolean* **bedgraph**, *basic:boolean* **spmr**, *basic:boolean* **call_summits**, *basic:boolean* **broad**, *basic:decimal* **broad_cutoff**, *basic:boolean* **use_filtered_bam**, *basic:integer* **tss**, *basic:integer* **stitch**, *data:bed* **mask**, *data:bed* **blacklist**, *basic:boolean* **calculate_enrichment**, *basic:integer* **profile_window**, *basic:string* **shift_size**) [Source: v1.5.1]

This process runs MACS2 in batch mode. MACS2 analysis is triggered for pairs of samples as defined using treatment-background sample relations. If there are no sample relations defined, each sample is treated individually for the MACS analysis.

Model-based Analysis of ChIP-Seq (MACS 2.0), is used to identify transcript factor binding sites. MACS 2.0 captures the influence of genome complexity to evaluate the significance of enriched ChIP regions, and MACS improves the spatial resolution of binding sites through combining the information of both sequencing tag position and orientation. It has also an option to link nearby peaks together in order to call broad peaks. See [here](https://github.com/taoliu/MACS/) for more information.

In addition to peak-calling, this process computes ChIP-Seq and ATAC-Seq QC metrics. Process returns a QC metrics report, fragment length estimation, and a deduplicated tagAlign file. QC report contains ENCODE 3 proposed QC metrics – [NRF](https://www.encodeproject.org/data-standards/terms/), [PBC bottlenecking coefficients, NSC, and RSC](https://genome.ucsc.edu/ENCODE/qualityMetrics.html#chipSeq).

For identification of super enhancers R2 uses the Rank Ordering of Super-Enhancers algorithm (ROSE2). This takes the peaks called by RSEG for acetylation and calculates the distances in-between to judge whether they can be considered super-enhancers. The ranked values can be plotted and by locating the inflection point in the resulting graph, super-enhancers can be assigned. It can also be used with the MACS calculated data. See [here](http://younglab.wi.mit.edu/super_enhancer_code.html) for more information.

Input arguments alignments

label
Aligned reads

type
`list:data:alignment:bam`

description
Select multiple treatment/background samples.

promoter

label
Promoter regions BED file

type
`data:bed`

description
BED file containing promoter regions (TSS+-1000 bp for example). Needed to get the number of peaks and reads mapped to promoter regions.

required
False

tagalign

label
Use tagAlign files

type
`basic:boolean`

description
Use filtered tagAlign files as case (treatment) and control (background) samples. If extsize parameter is not set, run MACS using input's estimated fragment length.

default
`True`

prepeakqc_settings.q_threshold

label
Quality filtering threshold

type
`basic:integer`

default
`30`

prepeakqc_settings.n_sub

label
Number of reads to subsample

type
`basic:integer`

default
`15000000`

prepeakqc_settings.tn5

label
Tn5 shifting

type
`basic:boolean`

description
Tn5 transposon shifting. Shift reads on "+" strand by 4 bp and reads on "-" strand by 5 bp.

default
`False`

prepeakqc_settings.shift

label
User-defined cross-correlation peak strandshift

type
`basic:integer`

description
If defined, SPP tool will not try to estimate fragment length but will use the given value as fragment length.

required
False

settings.duplicates

label
Number of duplicates

type
`basic:string`

description
It controls the MACS behavior towards duplicate tags at the exact same location – the same coordination and the same strand. The 'auto' option makes MACS calculate the maximum tags at the exact same location based on binomal distribution using 1e-5 as pvalue cutoff and the 'all' option keeps all the tags. If an integer is given, at most this number of tags will be kept at the same location. The default is to keep one tag at the same location.

required
False

hidden
tagalign

choices

- 1: `1`
- auto: `auto`
- all: `all`

settings.duplicates_prepeak

label
Number of duplicates

type
`basic:string`

description
It controls the MACS behavior towards duplicate tags at the exact same location – the same coordination and the same strand. The 'auto' option makes MACS calculate the maximum tags at the exact same location based on binomal distribution using 1e-5 as pvalue cutoff and the 'all' option keeps all the tags. If an integer is given, at most this number of tags will be kept at the same location. The default is to keep one tag at the same location.

required
False

hidden
!tagalign

default
`all`

choices

- 1: `1`
- auto: `auto`
- all: `all`

settings.qvalue

label
Q-value cutoff

type
`basic:decimal`

description
The q-value (minimum FDR) cutoff to call significant regions. Q-values are calculated from p-values using Benjamini-Hochberg procedure.

required
False

disabled
settings.pvalue && settings.pvalue_prepeak

settings.pvalue

label
P-value cutoff

type
`basic:decimal`

description
The p-value cutoff. If specified, MACS2 will use p-value instead of q-value cutoff.

required
False

disabled
settings.qvalue

hidden
tagalign

settings.pvalue_prepeak

label
P-value cutoff

type
`basic:decimal`

description
The p-value cutoff. If specified, MACS2 will use p-value instead of q-value cutoff.

disabled
settings.qvalue

hidden
!tagalign || settings.qvalue

default
`1e-05`

settings.cap_num

label
Cap number of peaks by taking top N peaks

type
`basic:integer`

description
To keep all peaks set value to 0.

disabled
settings.broad

default
`500000`

settings.mfold_lower

label
MFOLD range (lower limit)

type
`basic:integer`

description
This parameter is used to select the regions within MFOLD range of high-confidence enrichment ratio against background to build model. The regions must be lower than upper limit, and higher than the lower limit of fold enrichment. DEFAULT:10,30 means using all regions not too low (>10) and not too high (<30) to build paired-peaks model. If MACS can not find more than 100 regions to build model, it will use the –extsize parameter to continue the peak detection ONLY if –fix-bimodal is set.

required
False

settings.mfold_upper

label
MFOLD range (upper limit)

type
`basic:integer`

description
This parameter is used to select the regions within MFOLD range of high-confidence enrichment ratio against background to build model. The regions must be lower than upper limit, and higher than the lower limit of fold enrichment. DEFAULT:10,30 means using all regions not too low (>10) and not too high (<30) to build paired-peaks model. If MACS can not find more than 100 regions to build model, it will use the –extsize parameter to continue the peak detection ONLY if –fix-bimodal is set.

required
False

settings.slocal

label
Small local region

type
`basic:integer`

description
Slocal and llocal parameters control which two levels of regions will be checked around the peak

regions to calculate the maximum lambda as local lambda. By default, MACS considers 1000 bp for small local region (–slocal), and 10000 bp for large local region (–llocal) which captures the bias from a long range effect like an open chromatin domain. You can tweak these according to your project. Remember that if the region is set too small, a sharp spike in the input data may kill the significant peak.

required
False

settings.llocal

label
Large local region

type
`basic:integer`

description
Slocal and llocal parameters control which two levels of regions will be checked around the peak regions to calculate the maximum lambda as local lambda. By default, MACS considers 1000 bp for small local region (–slocal), and 10000 bp for large local region (–llocal) which captures the bias from a long range effect like an open chromatin domain. You can tweak these according to your project. Remember that if the region is set too small, a sharp spike in the input data may kill the significant peak.

required
False

settings.extsize

label
extsize

type
`basic:integer`

description
While '–nomodel' is set, MACS uses this parameter to extend reads in 5'->3' direction to fix-sized fragments. For example, if the size of binding region for your transcription factor is 200 bp, and you want to bypass the model building by MACS, this parameter can be set as 200. This option is only valid when –nomodel is set or when MACS fails to build model and –fix-bimodal is on.

required
False

settings.shift

label
Shift

type
`basic:integer`

description
Note, this is NOT the legacy –shiftsize option which is replaced by –extsize! You can set an arbitrary shift in bp here. Please Use discretion while setting it other than default value (0). When –nomodel is set, MACS will use this value to move cutting ends (5') then apply –extsize from 5' to 3' direction to extend them to fragments. When this value is negative, ends will be moved toward 3'->5' direction, otherwise 5'->3' direction. Recommended to keep it as default 0 for ChIP-Seq datasets, or -1 * half of EXTSIZE together with –extsize option for detecting enriched cutting loci such as certain DNAseI-

Seq datasets. Note, you can't set values other than 0 if format is BAMPE for paired-end data. Default is 0.

required
False

settings.band_width

label
Band width

type
`basic:integer`

description
The band width which is used to scan the genome ONLY for model building. You can set this parameter as the sonication fragment size expected from wet experiment. The previous side effect on the peak detection process has been removed. So this parameter only affects the model building.

required
False

settings.nolambda

label
Use backgroud lambda as local lambda

type
`basic:boolean`

description
With this flag on, MACS will use the background lambda as local lambda. This means MACS will not consider the local bias at peak candidate regions.

default
`False`

settings.fix_bimodal

label
Turn on the auto paired-peak model process

type
`basic:boolean`

description
Turn on the auto paired-peak model process. If it's set, when MACS failed to build paired model, it will use the nomodel settings, the '–extsize' parameter to extend each tag. If set, MACS will be terminated if paired-peak model has failed.

default
`False`

settings.nomodel

label
Bypass building the shifting model

type
`basic:boolean`

description
While on, MACS will bypass building the shifting model.

hidden
tagalign

default
`False`

settings.nomodel_prepeak

label
Bypass building the shifting model

type
`basic:boolean`

description
While on, MACS will bypass building the shifting model.

hidden
!tagalign

default
`True`

settings.down_sample

label
Down-sample

type
`basic:boolean`

description
When set to true, random sampling method will scale down the bigger sample. By default, MACS uses linear scaling. This option will make the results unstable and irreproducible since each time, random reads would be selected, especially the numbers (pileup, pvalue, qvalue) would change.

default
`False`

settings.bedgraph

label
Save fragment pileup and control lambda

type
`basic:boolean`

description
If this flag is on, MACS will store the fragment pileup, control lambda, -log10pvalue and -log10qvalue scores in bedGraph files. The bedGraph files will be stored in current directory named NAME+'_treat_pileup.bdg' for treatment data, NAME+'_control_lambda.bdg' for local lambda values from control, NAME+'_treat_pvalue.bdg' for Poisson pvalue scores (in -log10(pvalue) form), and NAME+'_treat_qvalue.bdg' for q-value scores from Benjamini-Hochberg-Yekutieli procedure.

default
`True`

settings.spmr

label
Save signal per million reads for fragment pileup profiles

type
`basic:boolean`

disabled
settings.bedgraph === false

default
`True`

settings.call_summits

label
Call summits

type
`basic:boolean`

description
MACS will now reanalyze the shape of signal profile (p or q-score depending on cutoff setting) to deconvolve subpeaks within each peak called from general procedure. It's highly recommended to detect adjacent binding events. While used, the output subpeaks of a big peak region will have the same peak boundaries, and different scores and peak summit positions.

default
`False`

settings.broad

label
Composite broad regions

type
`basic:boolean`

description
When this flag is on, MACS will try to composite broad regions in BED12 (a gene-model-like format) by putting nearby highly enriched regions into a broad region with loose cutoff. The broad region is controlled by another cutoff through –broad-cutoff. The maximum length of broad region length is 4 times of d from MACS.

disabled
settings.call_summits === true

default
`False`

settings.broad_cutoff

label
Broad cutoff

type
`basic:decimal`

description
Cutoff for broad region. This option is not available unless –broad is set. If -p is set, this is a p-value cutoff, otherwise, it's a q-value cutoff. DEFAULT = 0.1

required
False

disabled
settings.call_summits === true || settings.broad !== true

rose_settings.use_filtered_bam

label
Use Filtered BAM File

type
`basic:boolean`

description
Use filtered BAM file from a MACS2 object to rank enhancers by.

default
`True`

rose_settings.tss

label
TSS exclusion

type
`basic:integer`

description
Enter a distance from TSS to exclude. 0 = no TSS exclusion

default
`0`

rose_settings.stitch

label
Stitch

type
`basic:integer`

description
Enter a max linking distance for stitching. If not given, optimal stitching parameter will be determined automatically.

required
False

rose_settings.mask

label
Masking BED file

type
`data:bed`

description
Mask a set of regions from analysis. Provide a BED of masking regions.

required
False

chipqc_settings.blacklist

label
Blacklist regions

type
`data:bed`

description
BED file containing genomic regions that should be excluded from the analysis.

required
False

chipqc_settings.calculate_enrichment

label
Calculate enrichment

type
`basic:boolean`

description
Calculate enrichment of signal in known genomic annotation. By default annotation is provided from the TranscriptDB package specified by genome bulid which should match one of the supported annotations (hg19, hg38, hg18, mm10, mm9, rn4, ce6, dm3). If annotation is not supported the analysis is skipped.

default
`False`

chipqc_settings.profile_window

label
Window size

type
`basic:integer`

description
An integer indicating the width of the window used for peak profiles. Peaks will be centered on their summits and include half of the window size upstream and half downstream of this point.

default
`400`

chipqc_settings.shift_size

label
Shift size

type
`basic:string`

description
Vector of values to try when computing optimal shift sizes. It should be specifeird as consecutive numbers vector with start:end

default
`1:300`

Output results

Chemical Mutagenesis

`data:workflow:chemutworkflow-chemut` (*basic:string* **analysis_type**, *data:seq:nucleotide* **genome**, *list:data:alignment:bam* **parental_strains**, *list:data:alignment:bam* **mutant_strains**, *basic:boolean* **base_recalibration**, *data:variants:vcf* **known_sites**, *list:data:variants:vcf* **known_indels**, *basic:integer* **stand_call_conf**, *basic:integer* **mbq**, *basic:integer* **read_depth**) [Source: v2.1.0]

Input arguments analysis_type

label
Analysis type

type
`basic:string`

description
Choice of the analysis type. Use "SNV" or "INDEL" options to run the GATK analysis only on the haploid portion of the dicty genome. Choose options SNV_CHR2 or INDEL_CHR2 to run the analysis only on the diploid portion of CHR2 (-ploidy 2 -L chr2:2263132-3015703).

default
`snv`

choices

- SNV: `snv`
- INDEL: `indel`
- SNV_CHR2: `snv_chr2`
- INDEL_CHR2: `indel_chr2`

genome

label
Reference genome

type
`data:seq:nucleotide`

parental_strains

label
Parental strains

type
`list:data:alignment:bam`

mutant_strains

label
Mutant strains

type
`list:data:alignment:bam`

Vc.base_recalibration

label
Do variant base recalibration

type
`basic:boolean`

default
`False`

Vc.known_sites

label
Known sites (dbSNP)

type
`data:variants:vcf`

required
False

Vc.known_indels

label
Known indels

type
`list:data:variants:vcf`

required
False

hidden
Vc.base_recalibration === false

Vc.stand_call_conf

label
Calling confidence threshold

type
`basic:integer`

description
The minimum confidence threshold (phred-scaled) at which the program should emit variant sites as called. If a site's associated genotype has a confidence score lower than the calling threshold, the program will emit the site as filtered and will annotate it as LowQual. This threshold separates high confidence calls from low confidence calls.

default
`30`

Vc.mbq

label
Min base quality

type
`basic:integer`

description
Minimum base quality required to consider a base for calling.

default
`10`

Vf.read_depth

label
Read depth cutoff

type
`basic:integer`

description
The minimum number of replicate reads required for a variant site to be included.

default
`5`

Output results

ChipQC

`data:chipqc:chipqc` (*data:alignment:bam* **alignment**, *data:chipseq:callpeak* **peaks**, *data:bed* **blacklist**, *basic:boolean* **calculate_enrichment**, *basic:integer* **quality_threshold**, *basic:integer* **profile_window**, *basic:string* **shift_size**) [Source: v1.4.2]

Calculate quality control metrics for ChIP-seq samples.

The analysis is based on ChIPQC package which computs a variety of quality control metrics and statistics, and provides plots and a report for assessment of experimental data for further analysis.

Input arguments alignment

label
Aligned reads

type
`data:alignment:bam`

required
True

disabled
False

hidden
False

peaks

label
Called peaks

type
`data:chipseq:callpeak`

required
True

disabled
False

hidden
False

blacklist

label
Blacklist regions

type
`data:bed`

description
BED file containing genomic regions that should be excluded from the analysis.

required
False

disabled
False

hidden
False

calculate_enrichment

label
Calculate enrichment

type
`basic:boolean`

description
Calculate enrichment of signal in known genomic annotation. By default annotation is provided from the TranscriptDB package specified by genome bulid which should match one of the supported annotations (hg19, hg38, hg18, mm10, mm9, rn4, ce6, dm3). If annotation is not supported the analysis is skipped.

required
True

disabled
False

hidden
False

default
`False`

advanced.quality_threshold

label
Mapping quality threshold

type
`basic:integer`

description
Only reads with mapping quality scores above this threshold will be used for some statistics.

required
True

disabled
False

hidden
False

default
15

advanced.profile_window

label
Window size

type
basic:integer

description
An integer indicating the width of the window used for peak profiles. Peaks will be centered on their summits and include half of the window size upstream and half downstream of this point.

required
True

disabled
False

hidden
False

default
400

advanced.shift_size

label
Shift size

type
basic:string

description
Vector of values to try when computing optimal shift sizes. It should be specifeird as consecutive numbers vector with start:end

required
True

disabled
False

hidden
False

default
1:300

Output results report_folder

label
ChipQC report folder

type
basic:dir

required
True

disabled
False

hidden
False

ccplot

label
Cross coverage score plot

type
`basic:file`

required
True

disabled
False

hidden
False

coverage_histogram

label
SSD metric plot

type
`basic:file`

required
True

disabled
False

hidden
False

peak_profile

label
Peak profile plot

type
`basic:file`

required
True

disabled
False

hidden
False

peaks_barplot

label
Barplot of reads in peaks

type
`basic:file`

required
True

disabled
False

hidden
False

peaks_density_plot

label
Density plot of reads in peaks

type
`basic:file`

required
True

disabled
False

hidden
False

enrichment_heatmap

label
Heatmap of reads in genomic features

type
`basic:file`

required
False

disabled
False

hidden
False

species

label
Species

type
`basic:string`

required
True

disabled
False

hidden
False

build

label
Build

type
`basic:string`

required
True

disabled
False

hidden
False

Convert GFF3 to GTF

`data:annotation:gtfgff-to-gtf` (*data:annotation:gff3* **annotation**)[Source: v0.6.0]

Convert GFF3 file to GTF format.

Input arguments annotation

label
Annotation (GFF3)

type
`data:annotation:gff3`

description
Annotation in GFF3 format.

Output results annot

label
Converted GTF file

type
`basic:file`

annot_sorted

label
Sorted GTF file

type
`basic:file`

annot_sorted_idx_igv

label
Igv index for sorted GTF file

type
`basic:file`

annot_sorted_track_jbrowse

label
Jbrowse track for sorted GTF

type
`basic:file`

source

label
Gene ID database

type
`basic:string`

species

label
Species

type
`basic:string`

build

label
Build

type
`basic:string`

Convert files to reads (paired-end)

`data:reads:fastq:paired:files-to-fastq-paired` (*list:data:file* **src1**, *list:data:file* **src2**, *basic:boolean* **merge_lanes**) [Source: v1.6.0]

Convert FASTQ files to paired-end reads.

Input arguments src1

label
Mate1

type
`list:data:file`

required
True

disabled
False

hidden
False

src2

label
Mate2

type
`list:data:file`

required
True

disabled
False

hidden
False

merge_lanes

label
Merge lanes

type
`basic:boolean`

description
Merge sample data split into multiple sequencing lanes into a single FASTQ file.

required
True

disabled
False

hidden
False

default
`False`

Output results fastq

label
Reads file (mate 1)

type
`list:basic:file`

required
True

disabled
False

hidden
False

fastq2

label
Reads file (mate 2)

type
`list:basic:file`

required
True

disabled
False

hidden
False

fastqc_url

label
Quality control with FastQC (Upstream)

type
`list:basic:file:html`

required
True

disabled
False

hidden
False

fastqc_url2

label
Quality control with FastQC (Downstream)

type
`list:basic:file:html`

required
True

disabled
False

hidden
False

fastqc_archive

label
Download FasQC archive (Upstream)

type
`list:basic:file`

required
True

disabled
False

hidden
False

fastqc_archive2

label
Download FasQC archive (Downstream)

type
`list:basic:file`

required
True

disabled
False

hidden
False

Convert files to reads (single-end)

`data:reads:fastq:single:files-to-fastq-single` (*list:data:file* **src**, *basic:boolean* **merge_lanes**) [Source: v1.6.0]

Convert FASTQ files to single-end reads.

Input arguments src

label
Reads

type
`list:data:file`

description
Sequencing reads in FASTQ format

required
True

disabled
False

hidden
False

merge_lanes

label
Merge lanes

type
`basic:boolean`

description
Merge sample data split into multiple sequencing lanes into a single FASTQ file.

required
True

disabled
False

hidden
False

default
`False`

Output results fastq

label
Reads file

type
`list:basic:file`

required
True

disabled
False

hidden
False

fastqc_url

label
Quality control with FastQC

type
`list:basic:file:html`

required
True

disabled
False

hidden
False

fastqc_archive

label
Download FastQC archive

type
`list:basic:file`

required
True

disabled
False

hidden
False

Cuffdiff 2.2

`data:differentialexpression:cuffdiff:cuffdiff` (*list:data:cufflinks:cuffquant* **case**, *list:data:cufflinks:cuffquant* **control**, *list:basic:string* **labels**, *data:annotation* **annotation**, *data:seq:nucleotide* **genome**, *basic:boolean* **multi_read_correct**, *basic:boolean* **create_sets**, *basic:decimal* **gene_logfc**, *basic:decimal* **gene_fdr**, *basic:decimal* **fdr**, *basic:string* **library_type**, *basic:string* **library_normalization**, *basic:string* **dispersion_method**) [Source: v3.4.0]

Run Cuffdiff 2.2 analysis.

Cuffdiff finds significant changes in transcript expression, splicing, and promoter use. You can use it to find differentially expressed genes and transcripts, as well as genes that are being differentially regulated at the transcriptional and post-transcriptional level. See [here](http://cole-trapnell-lab.github.io/cufflinks/cuffdiff/) and [here](https://software.broadinstitute.org/cancer/software/genepattern/modules/docs/Cuffdiff/7) for more information.

Input arguments case

label
Case samples

type
`list:data:cufflinks:cuffquant`

required
True

disabled
False

hidden
False

control

label
Control samples

type
`list:data:cufflinks:cuffquant`

required
True

disabled
False

hidden
False

labels

label
Group labels

type
`list:basic:string`

description
Define labels for each sample group.

required
True

disabled
False

hidden
False

default
`['control', 'case']`

annotation

label
Annotation (GTF/GFF3)

type
`data:annotation`

description
A transcript annotation file produced by cufflinks, cuffcompare, or other tool.

required
True

disabled
False

hidden
False

genome

label
Run bias detection and correction algorithm

type
`data:seq:nucleotide`

description
Provide Cufflinks with a multifasta file (genome file) via this option to instruct it to run a bias detection and correction algorithm which can significantly improve accuracy of transcript abundance estimates.

required
False

disabled
False

hidden
False

multi_read_correct

label
Do initial estimation procedure to more accurately weight reads with multiple genome mappings

type
`basic:boolean`

required
True

disabled
False

hidden
False

default
`False`

create_sets

label
Create gene sets

type
`basic:boolean`

description
After calculating differential gene expressions create gene sets for up-regulated genes, down-regulated genes and all genes.

required
True

disabled
False

hidden
False

default
`False`

gene_logfc

label
Log2 fold change threshold for gene sets

type
`basic:decimal`

description
Genes above Log2FC are considered as up-regulated and genes below -Log2FC as down-regulated.

required
True

disabled
False

hidden
!create_sets

default
`1.0`

gene_fdr

label
FDR threshold for gene sets

type
`basic:decimal`

required
True

disabled
False

hidden
!create_sets

default
`0.05`

fdr

label
Allowed FDR

type
`basic:decimal`

description
The allowed false discovery rate. The default is 0.05.

required
True

disabled
False

hidden
False

default
`0.05`

library_type

label
Library type

type
`basic:string`

description
In cases where Cufflinks cannot determine the platform and protocol used to generate input reads, you can supply this information manually, which will allow Cufflinks to infer source strand information with certain protocols. The available options are listed below. For paired-end data, we currently only support protocols where reads point towards each other: fr-unstranded - Reads from the left-most end of the fragment (in transcript coordinates) map to the transcript strand and the right-most end maps to the opposite strand; fr-firststrand - Same as above except we enforce the rule that the right-most end of the fragment (in transcript coordinates) is the first sequenced (or only sequenced for single-end reads). Equivalently, it is assumed that only the strand generated during first strand synthesis is sequenced; fr-secondstrand - Same as above except we enforce the rule that the left-most end of the fragment (in transcript coordinates) is the first sequenced (or only sequenced for single-end reads). Equivalently, it is assumed that only the strand generated during second strand synthesis is sequenced.

required
True

disabled
False

hidden
False

default
`fr-unstranded`

choices

- fr-unstranded: `fr-unstranded`
- fr-firststrand: `fr-firststrand`
- fr-secondstrand: `fr-secondstrand`

library_normalization

label
Library normalization method

type
`basic:string`

description
You can control how library sizes (i.e. sequencing depths) are normalized in Cufflinks and Cuffdiff. Cuffdiff has several methods that require multiple libraries in order to work. Library normalization methods supported by Cufflinks work on one library at a time.

required
True

disabled
False

hidden
False

default
`geometric`

choices

- geometric: `geometric`
- classic-fpkm: `classic-fpkm`
- quartile: `quartile`

dispersion_method

label
Dispersion method

type
`basic:string`

description
Cuffdiff works by modeling the variance in fragment counts across replicates as a function of the mean fragment count across replicates. Strictly speaking, models a quantitity called dispersion - the variance present in a group of samples beyond what is expected from a simple Poisson model of RNA_Seq. You can control how Cuffdiff constructs its model of dispersion in locus fragment counts. Each condition that has replicates can receive its own model, or Cuffdiff can use a global model for all conditions. All of these policies are identical to those used by DESeq (Anders and Huber, Genome Biology, 2010).

required
True

disabled
False

hidden
False

default
`pooled`

choices

- pooled: `pooled`
- per-condition: `per-condition`
- blind: `blind`
- poisson: `poisson`

Output results raw

label
Differential expression

type
`basic:file`

required
True

disabled
False

hidden
False

de_json

label
Results table (JSON)

type
`basic:json`

required
True

disabled
False

hidden
False

de_file

label
Results table (file)

type
`basic:file`

required
True

disabled
False

hidden
False

transcript_diff_exp

label
Differential expression (transcript level)

type
`basic:file`

required
True

disabled
False

hidden
False

tss_group_diff_exp

label
Differential expression (primary transcript)

type
`basic:file`

required
True

disabled
False

hidden
False

cds_diff_exp

label
Differential expression (coding sequence)

type
`basic:file`

required
True

disabled
False

hidden
False

cuffdiff_output

label
Cuffdiff output

type
`basic:file`

required
True

disabled
False

hidden
False

source

label
Gene ID database

type
`basic:string`

required
True

disabled
False

hidden
False

species

label
Species

type
`basic:string`

required
True

disabled
False

hidden
False

build

label
Build

type
`basic:string`

required
True

disabled
False

hidden
False

feature_type

label
Feature type

type
`basic:string`

required
True

disabled
False

hidden
False

Cufflinks 2.2

`data:cufflinks:cufflinkscufflinks` (*data:alignment:bam* **alignment**, *data:annotation* **annotation**, *data:seq:nucleotide* **genome**, *data:annotation:gtf* **mask_file**, *basic:string* **library_type**, *basic:string* **annotation_usage**, *basic:boolean* **multi_read_correct**) [Source: v3.2.1]

Cufflinks assembles transcripts, estimates their abundances, and tests for differential expression and regulation in RNA-Seq samples. It accepts aligned RNA-Seq reads and assembles the alignments into a parsimonious set of transcripts. Cufflinks then estimates the relative abundances of these transcripts based on how many reads support each one, taking into account biases in library preparation protocols. See [here](http://cole-trapnell-lab.github.io/cufflinks/) for more information.

Input arguments alignment

label
Aligned reads

type
`data:alignment:bam`

annotation

label
Annotation (GTF/GFF3)

type
`data:annotation`

required
False

genome

label
Run bias detection and correction algorithm

type
`data:seq:nucleotide`

description
Provide Cufflinks with a multifasta file (genome file) via this option to instruct it to run a bias detection and correction algorithm which can significantly improve accuracy of transcript abundance estimates.

required
False

mask_file

label
Mask file

type
`data:annotation:gtf`

description
Ignore all reads that could have come from transcripts in this GTF file. We recommend including any annotated rRNA, mitochondrial transcripts other abundant transcripts you wish to ignore in your analysis in this file. Due to variable efficiency of mRNA enrichment methods and rRNA depletion kits, masking these transcripts often improves the overall robustness of transcript abundance estimates.

required
False

library_type

label
Library type

type
`basic:string`

description
In cases where Cufflinks cannot determine the platform and protocol used to generate input reads, you can supply this information manually, which will allow Cufflinks to infer source strand information with certain protocols. The available options are listed below. For paired-end data, we currently

only support protocols where reads are point towards each other: fr-unstranded - Reads from the left-most end of the fragment (in transcript coordinates) map to the transcript strand, and the right-most end maps to the opposite strand; fr-firststrand - Same as above except we enforce the rule that the right-most end of the fragment (in transcript coordinates) is the first sequenced (or only sequenced for single-end reads). Equivalently, it is assumed that only the strand generated during first strand synthesis is sequenced; fr-secondstrand - Same as above except we enforce the rule that the left-most end of the fragment (in transcript coordinates) is the first sequenced (or only sequenced for single-end reads). Equivalently, it is assumed that only the strand generated during second strand synthesis is sequenced.

default
`fr-unstranded`

choices

- fr-unstranded: `fr-unstranded`
- fr-firststrand: `fr-firststrand`
- fr-secondstrand: `fr-secondstrand`

annotation_usage

label
Instruct Cufflinks how to use the provided annotation (GFF/GTF) file

type
`basic:string`

description
GTF-guide - tells Cufflinks to use the supplied reference annotation (GFF) to guide RABT assembly. Reference transcripts will be tiled with faux-reads to provide additional information in assembly. Output will include all reference transcripts as well as any novel genes and isoforms that are assembled. –GTF - tells Cufflinks to use the supplied reference annotation (a GFF file) to estimate isoform expression. It will not assemble novel transcripts, and the program will ignore alignments not structurally compatible with any reference transcript.

default
`--GTF-guide`

choices

- Use supplied reference annotation to guide RABT assembly (–GTF-guide): `--GTF-guide`
- Use supplied reference annotation to estimate isoform expression (–GTF): `--GTF`

multi_read_correct

label
Do initial estimation procedure to more accurately weight reads with multiple genome mappings

type
`basic:boolean`

description
Run an initial estimation procedure that weights reads mapping to multiple locations more accurately.

default
`False`

Output results transcripts

label
Assembled transcript isoforms

type
basic:file

isoforms_fpkm_tracking

label
Isoforms FPKM tracking

type
basic:file

genes_fpkm_tracking

label
Genes FPKM tracking

type
basic:file

skipped_loci

label
Skipped loci

type
basic:file

source

label
Gene ID database

type
basic:string

species

label
Species

type
basic:string

build

label
Build

type
basic:string

Cuffmerge

data:annotation:cuffmergecuffmerge (*list:data:cufflinks:cufflinks* **expressions**, *list:data:annotation:gtf* **gtf**, *data:annotation* **gff**, *data:seq:nucleotide* **genome**, *basic:integer* **threads**) [Source: v2.2.0]

Cufflinks includes a script called Cuffmerge that you can use to merge together several Cufflinks assemblies. It also handles running Cuffcompare for you, and automatically filters a number of transfrags that are probably artfifacts. The main purpose of Cuffmerge is to make it easier to make an assembly GTF file suitable for use with Cuffdiff. See [here](http://cole-trapnell-lab.github.io/cufflinks/cuffmerge/) for more information.

Input arguments expressions

label
Cufflinks transcripts (GTF)

type
`list:data:cufflinks:cufflinks`

required
False

gtf

label
Annotation files (GTF)

type
`list:data:annotation:gtf`

description
Annotation files you wish to merge together with Cufflinks produced annotation files (e.g. upload Cufflinks annotation GTF file)

required
False

gff

label
Reference annotation (GTF/GFF3)

type
`data:annotation`

description
An optional "reference" annotation GTF. The input assemblies are merged together with the reference GTF and included in the final output.

required
False

genome

label
Reference genome

type
`data:seq:nucleotide`

description
This argument should point to the genomic DNA sequences for the reference. If a directory, it should contain one fasta file per contig. If a multifasta file, all contigs should be present. The merge script will pass this option to cuffcompare, which will use the sequences to assist in classifying transfrags and excluding artifacts (e.g. repeats). For example, Cufflinks transcripts consisting mostly of lower-case bases are classified as repeats. Note that <seq_dir> must contain one fasta file per reference chromosome, and each file must be named after the chromosome, and have a .fa or .fasta extension

required
False

threads

label
Use this many processor threads

type
`basic:integer`

description
Use this many threads to align reads. The default is 1.

default
1

Output results annot

label
Merged GTF file

type
`basic:file`

source

label
Gene ID database

type
`basic:string`

species

label
Species

type
`basic:string`

build

label
Build

type
`basic:string`

Cuffnorm

`data:cuffnormcuffnorm` (*list:data:cufflinks:cuffquant* **cuffquant**, *data:annotation* **annotation**, *basic:boolean* **useERCC**) [Source: v2.5.0]

Cufflinks includes a program, Cuffnorm, that you can use to generate tables of expression values that are properly normalized for library size. Cuffnorm takes a GTF2/GFF3 file of transcripts as input, along with two or more SAM, BAM, or CXB files for two or more samples. See [here](http://cole-trapnell-lab.github.io/cufflinks/cuffnorm/) for more information.

Replicate relation needs to be defined for Cuffnorm to account for replicates. If the replicate relation is not defined, each sample will be treated individually.

Input arguments cuffquant

label
Cuffquant expression file

type
`list:data:cufflinks:cuffquant`

annotation

label
Annotation (GTF/GFF3)

type
`data:annotation`

description
A transcript annotation file produced by cufflinks, cuffcompare, or other source.

useERCC

label
ERCC spike-in normalization

type
`basic:boolean`

description
Use ERRCC spike-in controls for normalization.

default
`False`

Output results genes_count

label
Genes count

type
`basic:file`

genes_fpkm

label
Genes FPKM

type
`basic:file`

genes_attr

label
Genes attr table

type
`basic:file`

isoform_count

label
Isoform count

type
`basic:file`

isoform_fpkm

label
Isoform FPKM

type
`basic:file`

isoform_attr

label
Isoform attr table

type
`basic:file`

cds_count

label
CDS count

type
`basic:file`

cds_fpkm

label
CDS FPKM

type
`basic:file`

cds_attr

label
CDS attr table

type
`basic:file`

tss_groups_count

label
TSS groups count

type
`basic:file`

tss_groups_fpkm

label
TSS groups FPKM

type
`basic:file`

tss_attr

label
TSS attr table

type
`basic:file`

run_info

label
Run info

type
`basic:file`

raw_scatter

label
FPKM exp scatter plot

type
basic:file

boxplot

label
Boxplot

type
basic:file

fpkm_exp_raw

label
FPKM exp raw

type
basic:file

replicate_correlations

label
Replicate correlatios plot

type
basic:file

fpkm_means

label
FPKM means

type
basic:file

exp_fpkm_means

label
Exp FPKM means

type
basic:file

norm_scatter

label
FKPM exp scatter normalized plot

type
basic:file

required
False

fpkm_exp_norm

label
FPKM exp normalized

type
basic:file

required
False

spike_raw

label
Spike raw

type
`basic:file`

required
False

spike_norm

label
Spike normalized

type
`basic:file`

required
False

R_data

label
All R normalization data

type
`basic:file`

source

label
Gene ID database

type
`basic:string`

species

label
Species

type
`basic:string`

build

label
Build

type
`basic:string`

Cuffquant 2.2

`data:cufflinks:cuffquantcuffquant` (*data:alignment:bam* **alignment**, *data:annotation* **annotation**, *data:seq:nucleotide* **genome**, *data:annotation:gtf* **mask_file**, *basic:string* **library_type**, *basic:boolean* **multi_read_correct**) [Source: v2.3.1]

Cuffquant allows you to compute the gene and transcript expression profiles and save these profiles to files that you can analyze later with Cuffdiff or Cuffnorm. See [here](http://cole-trapnell-lab.github.io/cufflinks/manual/) for more information.

Input arguments alignment

label
Aligned reads

type
`data:alignment:bam`

annotation

label
Annotation (GTF/GFF3)

type
`data:annotation`

genome

label
Run bias detection and correction algorithm

type
`data:seq:nucleotide`

description
Provide Cufflinks with a multifasta file (genome file) via this option to instruct it to run a bias detection and correction algorithm which can significantly improve accuracy of transcript abundance estimates.

required
False

mask_file

label
Mask file

type
`data:annotation:gtf`

description
Ignore all reads that could have come from transcripts in this GTF file. We recommend including any annotated rRNA, mitochondrial transcripts other abundant transcripts you wish to ignore in your analysis in this file. Due to variable efficiency of mRNA enrichment methods and rRNA depletion kits, masking these transcripts often improves the overall robustness of transcript abundance estimates.

required
False

library_type

label
Library type

type
`basic:string`

description
In cases where Cufflinks cannot determine the platform and protocol used to generate input reads, you can supply this information manually, which will allow Cufflinks to infer source strand information with certain protocols. The available options are listed below. For paired-end data, we currently only support protocols where reads are point towards each other: fr-unstranded - Reads from the left-most end of the fragment (in transcript coordinates) map to the transcript strand, and the right-most

end maps to the opposite strand; fr-firststrand - Same as above except we enforce the rule that the right-most end of the fragment (in transcript coordinates) is the first sequenced (or only sequenced for single-end reads). Equivalently, it is assumed that only the strand generated during first strand synthesis is sequenced; fr-secondstrand - Same as above except we enforce the rule that the left-most end of the fragment (in transcript coordinates) is the first sequenced (or only sequenced for single-end reads). Equivalently, it is assumed that only the strand generated during second strand synthesis is sequenced.

default
`fr-unstranded`

choices

- fr-unstranded: `fr-unstranded`
- fr-firststrand: `fr-firststrand`
- fr-secondstrand: `fr-secondstrand`

multi_read_correct

label
Do initial estimation procedure to more accurately weight reads with multiple genome mappings

type
`basic:boolean`

description
Run an initial estimation procedure that weights reads mapping to multiple locations more accurately.

default
`False`

Output results cxb

label
Abundances (.cxb)

type
`basic:file`

source

label
Gene ID database

type
`basic:string`

species

label
Species

type
`basic:string`

build

label
Build

type
`basic:string`

Cuffquant results

`data:cufflinks:cuffquantupload-cxb` (*basic:file* **src**, *basic:string* **source**, *basic:string* **species**, *basic:string* **build**, *basic:string* **feature_type**) [Source: v1.3.3]

Upload Cuffquant results file (.cxb)

Input arguments src

label
Cuffquant file

type
`basic:file`

description
Upload Cuffquant results file. Supported extention: *.cxb

required
True

validate_regex
`\.(cxb)$`

source

label
Gene ID database

type
`basic:string`

choices

- AFFY: `AFFY`
- DICTYBASE: `DICTYBASE`
- ENSEMBL: `ENSEMBL`
- NCBI: `NCBI`
- UCSC: `UCSC`

species

label
Species

type
`basic:string`

description
Species latin name.

choices

- Homo sapiens: `Homo sapiens`
- Mus musculus: `Mus musculus`
- Rattus norvegicus: `Rattus norvegicus`
- Dictyostelium discoideum: `Dictyostelium discoideum`
- Odocoileus virginianus texanus: `Odocoileus virginianus texanus`

- Solanum tuberosum: `Solanum tuberosum`

build

label
Build

type
`basic:string`

feature_type

label
Feature type

type
`basic:string`

default
`gene`

choices

- gene: `gene`
- transcript: `transcript`
- exon: `exon`

Output results cxb

label
Cuffquant results

type
`basic:file`

source

label
Gene ID database

type
`basic:string`

species

label
Species

type
`basic:string`

build

label
Build

type
`basic:string`

feature_type

label
Feature type

type
`basic:string`

Cut & Run

`data:workflow:cutnrunworkflow-cutnrun` (*data:reads:fastq:paired* **reads**, *basic:integer* **quality**, *basic:integer* **nextseq**, *basic:string* **phred**, *basic:integer* **min_length**, *basic:integer* **max_n**, *basic:boolean* **retain_unpaired**, *basic:integer* **unpaired_len_1**, *basic:integer* **unpaired_len_2**, *basic:integer* **clip_r1**, *basic:integer* **clip_r2**, *basic:integer* **three_prime_r1**, *basic:integer* **three_prime_r2**, *list:basic:string* **adapter**, *list:basic:string* **adapter_2**, *data:seq:nucleotide* **adapter_file_1**, *data:seq:nucleotide* **adapter_file_2**, *basic:string* **universal_adapter**, *basic:integer* **stringency**, *basic:decimal* **error_rate**, *basic:integer* **trim_5**, *basic:integer* **trim_3**, *data:index:bowtie2* **genome**, *basic:string* **mode**, *basic:string* **speed**, *basic:boolean* **discordantly**, *basic:boolean* **rep_se**, *basic:integer* **minins**, *basic:integer* **maxins**, *basic:boolean* **no_overlap**, *basic:boolean* **dovetail**, *basic:boolean* **no_unal**, *data:index:bowtie2* **genome**, *basic:string* **mode**, *basic:string* **speed**, *basic:boolean* **discordantly**, *basic:boolean* **rep_se**, *basic:integer* **minins**, *basic:integer* **maxins**, *basic:boolean* **no_overlap**, *basic:boolean* **dovetail**, *basic:boolean* **no_unal**, *basic:string* **format**, *basic:decimal* **pvalue**, *basic:string* **duplicates**, *basic:boolean* **bedgraph**, *basic:integer* **min_frag_length**, *basic:integer* **max_frag_length**, *basic:decimal* **scale**) [Source: v1.6.0]

Analysis of samples processed for high resolution mapping of DNA binding sites using targeted nuclease strategy. The process is named CUT&RUN which stands for Cleavage Under Target and Release Using Nuclease. Workflow includes steps of trimming the reads with trimgalore, aligning them using bowtie2 to target species genome as well as a spike-in genome. Aligned reads are processed to produce bigwig files to be viewed in a genome browser. Peaks are called using MACS2. Lenght-selection of reads is performed using alignmentSieve tool from the deeptools package.

Input arguments reads

label
Input reads

type
`data:reads:fastq:paired`

options_trimming.quality_trim.quality

label
Quality cutoff

type
`basic:integer`

description
Trim low-quality ends from reads based on Phred score.

required
False

options_trimming.quality_trim.nextseq

label
NextSeq/NovaSeq trim cutoff

type
`basic:integer`

description
NextSeq/NovaSeq-specific quality trimming. Trims also dark cycles appearing as high-quality G bases. This will set a specific quality cutoff, but qualities of G bases are ignored. This can not be used with Quality cutoff and will override it.

required
False

options_trimming.quality_trim.phred

label
Phred score encoding

type
`basic:string`

description
Use either ASCII+33 quality scores as Phred scores (Sanger/Illumina 1 .9+ encoding) or ASCII+64 quality scores (Illumina 1.5 encoding) for quality trimming.

default
`--phred33`

choices

- ASCII+33: `--phred33`
- ASCII+64: `--phred64`

options_trimming.quality_trim.min_length

label
Minimum length after trimming

type
`basic:integer`

description
Discard reads that became shorter than selected length because of either quality or adapter trimming. Both reads of a read-pair need to be longer than specified length to be printed out to validated paired-end files. If only one read became too short there is the possibility of keeping such unpaired single-end reads with Retain unpaired. A value of 0 disables filtering based on length.

default
`20`

options_trimming.quality_trim.max_n

label
Maximum number of Ns

type
`basic:integer`

description
Read exceeding this limit will result in the entire pair being removed from the trimmed output files.

required
False

options_trimming.quality_trim.retain_unpaired

label
Retain unpaired reads after trimming

type
`basic:boolean`

description
If only one of the two paired-end reads "became too short, the longer read will be written.

default
`False`

options_trimming.quality_trim.unpaired_len_1

label
Unpaired read length cutoff of mate 1

type
`basic:integer`

hidden
!quality_trim.retain_unpaired

default
`35`

options_trimming.quality_trim.unpaired_len_2

label
Unpaired read length cutoff for mate 2

type
`basic:integer`

hidden
!quality_trim.retain_unpaired

default
`35`

options_trimming.quality_trim.clip_r1

label
Trim bases from 5' end of read 1

type
`basic:integer`

description
This may be useful if the qualities were very poor, or if there is some sort of unwanted bias at the 5' end.

required
False

options_trimming.quality_trim.clip_r2

label
Trim bases from 5' end of read 2

type
`basic:integer`

description
This may be useful if the qualities were very poor, or if there is some sort of unwanted bias at the 5' end. For paired-end bisulfite sequencing, it is recommended to remove the first few bp because the end-repair reaction may introduce a bias towards low methylation.

required
False

options_trimming.quality_trim.three_prime_r1

label
Trim bases from 3' end of read 1

type
`basic:integer`

description
Remove bases from the 3' end of read 1 after adapter/quality trimming has been performed. This may remove some unwanted bias from the 3' end that is not directly related to adapter sequence or basecall quality.

required
False

options_trimming.quality_trim.three_prime_r2

label
Trim bases from 3' end of read 2

type
`basic:integer`

description
Remove bases from the 3' end of read 2 after adapter/quality trimming has been performed. This may remove some unwanted bias from the 3' end that is not directly related to adapter sequence or basecall quality.

required
False

options_trimming.adapter_trim.adapter

label
Read 1 adapter sequence

type
`list:basic:string`

description
Adapter sequences to be trimmed. Also see universal adapters field for predefined adapters. This is mutually exclusive with read 1 adapters file and universal adapters.

required
False

options_trimming.adapter_trim.adapter_2

label
Read 2 adapter sequence

type
`list:basic:string`

description
Optional adapter sequence to be trimmed off read 2 of paired-end files. This is mutually exclusive with read 2 adapters file and universal adapters.

required
False

options_trimming.adapter_trim.adapter_file_1

label
Read 1 adapters file

type
`data:seq:nucleotide`

description
This is mutually exclusive with read 1 adapters and universal adapters.

required
False

options_trimming.adapter_trim.adapter_file_2

label
Read 2 adapters file

type
`data:seq:nucleotide`

description
This is mutually exclusive with read 2 adapters and universal adapters.

required
False

options_trimming.adapter_trim.universal_adapter

label
Universal adapters

type
`basic:string`

description
Instead of default detection use specific adapters. Use 13bp of the Illumina universal adapter, 12bp of the Nextera adapter or 12bp of the Illumina Small RNA 3' Adapter. Selecting to trim smallRNA adapters will also lower the length value to 18bp. If the smallRNA libraries are paired-end then read 2 adapter will be set to the Illumina small RNA 5' adapter automatically (GATCGTCGGACT) unless defined explicitly. This is mutually exclusive with manually defined adapters and adapter files.

required
False

choices

- Illumina: `--illumina`
- Nextera: `--nextera`

- Illumina small RNA: `--small_rna`

options_trimming.adapter_trim.stringency

label
Overlap with adapter sequence required to trim

type
`basic:integer`

description
Defaults to a very stringent setting of 1, i.e. even a single base pair of overlapping sequence will be trimmed of the 3' end of any read.

default
`1`

options_trimming.adapter_trim.error_rate

label
Maximum allowed error rate

type
`basic:decimal`

description
Number of errors divided by the length of the matching region. Default value of 0.1.

default
`0.1`

options_trimming.hard_trim.trim_5

label
Hard trim sequence from 3' end

type
`basic:integer`

description
Instead of performing adapter-/quality trimming, this option will simply hard-trim sequences to bp from the 3' end. This is incompatible with other hard trimming options.

required
False

options_trimming.hard_trim.trim_3

label
Hard trim sequences from 5' end

type
`basic:integer`

description
Instead of performing adapter-/quality trimming, this option will simply hard-trim sequences to bp from the 5' end. This is incompatible with other hard trimming options.

required
False

options_aln_species.genome

label
Species genome

type
`data:index:bowtie2`

options_aln_species.mode

label
Alignment mode

type
`basic:string`

description
End to end: Bowtie 2 requires that the entire read align from one end to the other, without any trimming (or "soft clipping") of characters from either end.

Local: Bowtie 2 does not require that the entire read align from one end to the other. Rather, some characters may be omitted ("soft clipped") from the ends in order to achieve the greatest possible alignment score.

default
`--local`

choices

- end to end mode: `--end-to-end`
- local: `--local`

options_aln_species.speed

label
Speed vs. Sensitivity

type
`basic:string`

description
A quick setting for aligning fast or accurately. This option is a shortcut for parameters as follows:

For –end-to-end: –very-fast -D 5 -R 1 -N 0 -L 22 -i S,0,2.50 –fast -D 10 -R 2 -N 0 -L 22 -i S,0,2.50 –sensitive -D 15 -R 2 -N 0 -L 22 -i S,1,1.15 (default) –very-sensitive -D 20 -R 3 -N 0 -L 20 -i S,1,0.50

For –local: –very-fast-local -D 5 -R 1 -N 0 -L 25 -i S,1,2.00 –fast-local -D 10 -R 2 -N 0 -L 22 -i S,1,1.75 –sensitive-local -D 15 -R 2 -N 0 -L 20 -i S,1,0.75 (default) –very-sensitive-local -D 20 -R 3 -N 0 -L 20 -i S,1,0.50

default
`--very-sensitive`

choices

- Very fast: `--very-fast`
- Fast: `--fast`
- Sensitive: `--sensitive`
- Very sensitive: `--very-sensitive`

options_aln_species.discordantly

label
Report discordantly matched read

type
`basic:boolean`

description
If both mates have unique alignments, but the alignments do not match paired-end expectations (orientation and relative distance) then alignment will be reported. Useful for detecting structural variations.

default
`True`

options_aln_species.rep_se

label
Report single ended

type
`basic:boolean`

description
If paired alignment can not be found Bowtie2 tries to find alignments for the individual mates. Default is true (–no-mixed).

default
`True`

options_aln_species.minins

label
Minimal distance

type
`basic:integer`

description
The minimum fragment length (–minins) for valid paired-end alignments. Value 0 imposes no minimum.

default
`10`

options_aln_species.maxins

label
Maximal distance

type
`basic:integer`

description
The maximum fragment length (–maxins) for valid paired-end alignments.

default
`700`

options_aln_species.no_overlap

label
Not concordant when mates overlap

type
`basic:boolean`

description
When true, it is considered not concordant when mates overlap at all. Defaul is true (–no-overlap).

default
`False`

options_aln_species.dovetail

label
Dovetail

type
`basic:boolean`

description
If the mates "dovetail", that is if one mate alignment extends past the beginning of the other such that the wrong mate begins upstream, consider that to be concordant. Default: mates cannot dovetail in a concordant alignment. If true, parameter –dovetail is turned on.

default
`False`

options_aln_species.no_unal

label
Suppress SAM records for unaligned reads

type
`basic:boolean`

description
When true, suppress SAM records for unaligned reads. Default is true (–no-unal).

default
`True`

options_aln_spikein.genome

label
Spike-in genome

type
`data:index:bowtie2`

options_aln_spikein.mode

label
Alignment mode

type
`basic:string`

description
End to end: Bowtie 2 requires that the entire read align from one end to the other, without any trimming (or "soft clipping") of characters from either end.

Local: Bowtie 2 does not require that the entire read align from one end to the other. Rather, some characters may be omitted ("soft clipped") from the ends in order to achieve the greatest possible alignment score.

default
`--local`

choices

- end to end mode: `--end-to-end`
- local: `--local`

options_aln_spikein.speed

label
Speed vs. Sensitivity

type
`basic:string`

description
A quick setting for aligning fast or accurately. This option is a shortcut for parameters as follows:

For –end-to-end: –very-fast -D 5 -R 1 -N 0 -L 22 -i S,0,2.50 –fast -D 10 -R 2 -N 0 -L 22 -i S,0,2.50 –sensitive -D 15 -R 2 -N 0 -L 22 -i S,1,1.15 (default) –very-sensitive -D 20 -R 3 -N 0 -L 20 -i S,1,0.50

For –local: –very-fast-local -D 5 -R 1 -N 0 -L 25 -i S,1,2.00 –fast-local -D 10 -R 2 -N 0 -L 22 -i S,1,1.75 –sensitive-local -D 15 -R 2 -N 0 -L 20 -i S,1,0.75 (default) –very-sensitive-local -D 20 -R 3 -N 0 -L 20 -i S,1,0.50

default
`--very-sensitive`

choices

- Very fast: `--very-fast`
- Fast: `--fast`
- Sensitive: `--sensitive`
- Very sensitive: `--very-sensitive`

options_aln_spikein.discordantly

label
Report discordantly matched read

type
`basic:boolean`

description
If both mates have unique alignments, but the alignments do not match paired-end expectations (orientation and relative distance) then alignment will be reported. Useful for detecting structural variations.

default
`True`

options_aln_spikein.rep_se

label
Report single ended

type
`basic:boolean`

description
If paired alignment can not be found Bowtie2 tries to find alignments for the individual mates. Default is true (–no-mixed).

default
`True`

options_aln_spikein.minins

label
Minimal distance

type
`basic:integer`

description
The minimum fragment length (–minins) for valid paired-end alignments. Value 0 imposes no minimum.

default
`10`

options_aln_spikein.maxins

label
Maximal distance

type
`basic:integer`

description
The maximum fragment length (–maxins) for valid paired-end alignments.

default
`700`

options_aln_spikein.no_overlap

label
Not concordant when mates overlap

type
`basic:boolean`

description
When true, it is considered not concordant when mates overlap at all. Defaul is true (–no-overlap).

default
`True`

options_aln_spikein.dovetail

label
Dovetail

type
`basic:boolean`

description
If the mates “dovetail”, that is if one mate alignment extends past the beginning of the other such that the wrong mate begins upstream, consider that to be concordant. Default: mates cannot dovetail in a concordant alignment. If true, parameter –dovetail is turned on.

default
`False`

options_aln_spikein.no_unal

label
Suppress SAM records for unaligned reads

type
`basic:boolean`

description
When true, suppress SAM records for unaligned reads. Default is true (–no-unal).

default
`True`

options_pc.format

label
Format of tag file

type
`basic:string`

description
This specifies the format of input files. For paired-end data the format dicates how MACS2 will treat mates. If the selected format is BAM, MACS2 will only keep the left mate (5' end) tag. However, when format BAMPE is selected, MACS2 will use actual insert sizes of pairs of reads to build fragment pileup, instead of building bimodal distribution plus and minus strand reads to predict fragment size.

required
False

default
`BAMPE`

choices

- BAM: `BAM`
- BAMPE: `BAMPE`

options_pc.pvalue

label
P-value cutoff

type
`basic:decimal`

description
The p-value cutoff.

required
False

default
`0.001`

options_pc.duplicates

label
Number of duplicates

type
`basic:string`

description
It controls the MACS behavior towards duplicate tags at the exact same location – the same coordination and the same strand. The 'auto' option makes MACS calculate the maximum tags at the exact same location based on binomal distribution using 1e-5 as pvalue cutoff and the 'all' option keeps all the tags. If an integer is given, at most this number of tags will be kept at the same location. The default is to keep one tag at the same location.

default
`all`

choices

- 1: `1`
- auto: `auto`
- all: `all`

options_pc.bedgraph

label
Save fragment pileup and control lambda

type
`basic:boolean`

description
If this flag is on, MACS will store the fragment pileup, control lambda, -log10(pvalue) and -log10(qvalue) scores in bedGraph files. The bedGraph files will be stored in current directory named NAME+'_treat_pileup.bdg' for treatment data, NAME+'_control_lambda.bdg' for local lambda values from control, NAME+'_treat_pvalue.bdg' for Poisson pvalue scores (in -log10(pvalue) form), and NAME+'_treat_qvalue.bdg' for q-value scores from Benjamini-Hochberg-Yekutieli procedure.

default
`True`

options_sieve.min_frag_length

label
Minimum fragment length

type
`basic:integer`

description
The minimum fragment length needed for read/pair inclusion. This option is primarily useful in ATACseq experiments, for filtering mono- or di-nucleosome fragments. Default is 0.

default
`0`

options_sieve.max_frag_length

label
Maximum fragment length

type
`basic:integer`

description
The maximum fragment length needed for read/pair inclusion. A value of 0 indicates no limit. Default is 0.

default
`0`

options_scale.scale

label
Scale factor

type
`basic:decimal`

description
Magnitude of the scale factor. The scaling factor is calculated by dividing the scale with the number of features in BEDPE (scale/(number of features)).

default
`10000`

Output results

Cutadapt (3' mRNA-seq, single-end)

`data:reads:fastq:single:cutadapt:cutadapt-3prime-single` (*data:reads:fastq:single* **reads**, *basic:integer* **nextseq_trim**, *basic:integer* **quality_cutoff**, *basic:integer* **min_len**, *basic:integer* **min_overlap**, *basic:integer* **times**) [Source: v1.4.2]

Process 3' mRNA-seq datasets using Cutadapt tool.

Input arguments reads

label
Select sample(s)

type
`data:reads:fastq:single`

required
True

disabled
False

hidden
False

options.nextseq_trim

label
NextSeq/NovaSeq trim

type
`basic:integer`

description
NextSeq/NovaSeq-specific quality trimming. Trims also dark cycles appearing as high-quality G bases. This option is mutually exclusive with the use of standard quality-cutoff trimming and is suitable for the use with data generated by the recent Illumina machines that utilize two-color chemistry to encode the four bases.

required
True

disabled
False

hidden
False

default
`10`

options.quality_cutoff

label
Quality cutoff

type
`basic:integer`

description
Trim low-quality bases from 3' end of each read before adapter removal. The use of this option will override the use of NextSeq/NovaSeq trim option.

required
False

disabled
False

hidden
False

options.min_len

label
Discard reads shorter than specified minimum length.

type
`basic:integer`

required
True

disabled
False

hidden
False

default
`20`

options.min_overlap

label
Mimimum overlap

type
`basic:integer`

description
Minimum overlap between adapter and read for an adapter to be found.

required
True

disabled
False

hidden
False

default
`20`

options.times

label
Remove up to a specified number of adapters from each read.

type
`basic:integer`

required
True

disabled
False

hidden
False

default
`2`

Output results fastq

label
Reads file.

type
`list:basic:file`

required
True

disabled
False

hidden
False

report

label
Cutadapt report

type
`basic:file`

required
True

disabled
False

hidden
False

fastqc_url

label
Quality control with FastQC.

type
`list:basic:file:html`

required
True

disabled
False

hidden
False

fastqc_archive

label
Download FastQC archive.

type
`list:basic:file`

required
True

disabled
False

hidden
False

Cutadapt (Corall RNA-Seq, paired-end)

`data:reads:fastq:paired:cutadapt:cutadapt-corall-paired` (*data:reads:fastq:paired* **reads**, *basic:integer* **nextseq_trim**, *basic:integer* **quality_cutoff**, *basic:integer* **min_len**, *basic:integer* **min_overlap**) [Source: v1.3.2]

Pre-process reads obtained using CORALL Total RNA-Seq Library Prep Kit.

Trim UMI-tags from input reads and use Cutadapt to remove adapters and run QC filtering steps.

Input arguments reads

label
Select sample(s)

type
`data:reads:fastq:paired`

required
True

disabled
False

hidden
False

options.nextseq_trim

label
NextSeq/NovaSeq trim

type
`basic:integer`

description
NextSeq/NovaSeq-specific quality trimming. Trims also dark cycles appearing as high-quality G bases. This option is mutually exclusive with the use of standard quality-cutoff trimming and is suitable for the use with data generated by the recent Illumina machines that utilize two-color chemistry to encode the four bases.

required
True

disabled
False

hidden
False

default
`10`

options.quality_cutoff

label
Quality cutoff

type
`basic:integer`

description
Trim low-quality bases from 3' end of each read before adapter removal. The use of this option will override the use of NextSeq/NovaSeq trim option.

required
False

disabled
False

hidden
False

options.min_len

label
Minimum read length

type
`basic:integer`

required
True

disabled
False

hidden
False

default
`20`

options.min_overlap

label
Mimimum overlap

type
`basic:integer`

description
Minimum overlap between adapter and read for an adapter to be found.

required
True

disabled
False

hidden
False

default
`20`

Output results fastq

label
Remaining mate1 reads

type
`list:basic:file`

required
True

disabled
False

hidden
False

fastq2

label
Remaining mate2 reads

type
`list:basic:file`

required
True

disabled
False

hidden
False

report

label
Cutadapt report

type
`basic:file`

required
True

disabled
False

hidden
False

fastqc_url

label
Mate1 quality control with FastQC

type
`list:basic:file:html`

required
True

disabled
False

hidden
False

fastqc_url2

label
Mate2 quality control with FastQC

type
`list:basic:file:html`

required
True

disabled
False

hidden
False

fastqc_archive

label
Download mate1 FastQC archive

type
`list:basic:file`

required
True

disabled
False

hidden
False

fastqc_archive2

label
Download mate2 FastQC archive

type
`list:basic:file`

required
True

disabled
False

hidden
False

Cutadapt (Corall RNA-Seq, single-end)

`data:reads:fastq:single:cutadapt:cutadapt-corall-single` (*data:reads:fastq:single* **reads**, *basic:integer* **nextseq_trim**, *basic:integer* **quality_cutoff**, *basic:integer* **min_len**, *basic:integer* **min_overlap**) [Source: v1.4.2]

Pre-process reads obtained using CORALL Total RNA-Seq Library Prep Kit.

Trim UMI-tags from input reads and use Cutadapt to remove adapters and run QC filtering steps.

Input arguments reads

label
Select sample(s)

type
`data:reads:fastq:single`

required
True

disabled
False

hidden
False

options.nextseq_trim

label
NextSeq/NovaSeq trim

type
`basic:integer`

description
NextSeq/NovaSeq-specific quality trimming. Trims also dark cycles appearing as high-quality G bases. This option is mutually exclusive with the use of standard quality-cutoff trimming and is suitable for the use with data generated by the recent Illumina machines that utilize two-color chemistry to encode the four bases.

required
True

disabled
False

hidden
False

default
`10`

options.quality_cutoff

label
Quality cutoff

type
`basic:integer`

description
Trim low-quality bases from 3' end of each read before adapter removal. The use of this option will override the use of NextSeq/NovaSeq trim option.

required
False

disabled
False

hidden
False

options.min_len

label
Minimum read length

type
`basic:integer`

required
True

disabled
False

hidden
False

default
`20`

options.min_overlap

label
Mimimum overlap

type
`basic:integer`

description
Minimum overlap between adapter and read for an adapter to be found.

required
True

disabled
False

hidden
False

default
`20`

Output results fastq

label
Reads file

type
`list:basic:file`

required
True

disabled
False

hidden
False

report

label
Cutadapt report

type
`basic:file`

required
True

disabled
False

hidden
False

fastqc_url

label
Quality control with FastQC

type
`list:basic:file:html`

required
True

disabled
False

hidden
False

fastqc_archive

label
Download FastQC archive

type
`list:basic:file`

required
True

disabled
False

hidden
False

Cutadapt (paired-end)

`data:reads:fastq:paired:cutadaptcutadapt-paired` (*data:reads:fastq:paired* **reads**, *data:seq:nucleotide* **mate1_5prime_file**, *data:seq:nucleotide* **mate1_3prime_file**, *data:seq:nucleotide* **mate2_5prime_file**, *data:seq:nucleotide* **mate2_3prime_file**, *list:basic:string* **mate1_5prime_seq**, *list:basic:string* **mate1_3prime_seq**, *list:basic:string* **mate2_5prime_seq**, *list:basic:string* **mate2_3prime_seq**, *basic:integer* **times**, *basic:decimal* **error_rate**, *basic:integer* **min_overlap**, *basic:boolean* **match_read_wildcards**, *basic:boolean* **no_indels**, *basic:integer* **nextseq_trim**, *basic:integer* **leading**, *basic:integer* **trailing**, *basic:integer* **crop**, *basic:integer* **headcrop**, *basic:integer* **minlen**, *basic:integer* **maxlen**, *basic:integer* **max_n**, *basic:string* **pair_filter**) [Source: v2.7.2]

Cutadapt finds and removes adapter sequences, primers, poly-A tails and other types of unwanted sequence from high-throughput sequencing reads. More information about Cutadapt can be found [here](http://cutadapt.readthedocs.io/en/stable/).

Input arguments reads

label
Select sample(s)

type
`data:reads:fastq:paired`

adapters.mate1_5prime_file

label
5 prime adapter file for Mate 1

type
`data:seq:nucleotide`

required
False

adapters.mate1_3prime_file

label
3 prime adapter file for Mate 1

type
`data:seq:nucleotide`

required
False

adapters.mate2_5prime_file

label
5 prime adapter file for Mate 2

type
`data:seq:nucleotide`

required
False

adapters.mate2_3prime_file

label
3 prime adapter file for Mate 2

type
`data:seq:nucleotide`

required
False

adapters.mate1_5prime_seq

label
5 prime adapter sequence for Mate 1

type
`list:basic:string`

required
False

adapters.mate1_3prime_seq

label
3 prime adapter sequence for Mate 1

type
`list:basic:string`

required
False

adapters.mate2_5prime_seq

label
5 prime adapter sequence for Mate 2

type
`list:basic:string`

required
False

adapters.mate2_3prime_seq

label
3 prime adapter sequence for Mate 2

type
`list:basic:string`

required
False

adapters.times

label
Times

type
`basic:integer`

description
Remove up to COUNT adapters from each read.

default
`1`

adapters.error_rate

label
Error rate

type
`basic:decimal`

description
Maximum allowed error rate (no. of errors divided by the length of the matching region).

default
`0.1`

adapters.min_overlap

label
Minimal overlap

type
`basic:integer`

description
Minimum overlap for an adapter match.

default
`3`

adapters.match_read_wildcards

label
Match read wildcards

type
`basic:boolean`

description
Interpret IUPAC wildcards in reads.

default
`False`

adapters.no_indels

label
No indels

type
`basic:boolean`

description
Disable (disallow) insertions and deletions in adapters.

default
`False`

modify_reads.nextseq_trim

label
NextSeq-specific quality trimming

type
`basic:integer`

description
NextSeq-specific quality trimming (each read). Trims also dark cycles appearing as high-quality G bases. This option is mutually exclusive with the use of regular (-g) quality trimming.

required
False

modify_reads.leading

label
Quality on 5 prime

type
`basic:integer`

description
Remove low quality bases from 5 prime. Specifies the minimum quality required to keep a base.

required
False

modify_reads.trailing

label
Quality on 3 prime

type
`basic:integer`

description
Remove low quality bases from the 3 prime. Specifies the minimum quality required to keep a base.

required
False

modify_reads.crop

label
Crop

type
`basic:integer`

description
Cut the specified number of bases from the end of the reads.

required
False

modify_reads.headcrop

label
Headcrop

type
`basic:integer`

description
Cut the specified number of bases from the start of the reads.

required
False

filtering.minlen

label
Min length

type
`basic:integer`

description
Drop the read if it is below a specified.

required
False

filtering.maxlen

label
Max length

type
`basic:integer`

description
Drop the read if it is above a specified length.

required
False

filtering.max_n

label
Max numebr of N-s

type
`basic:integer`

description
Discard reads having more 'N' bases than specified.

required
False

filtering.pair_filter

label
Which of the reads have to match the filtering criterion

type
`basic:string`

description
Which of the reads in a paired-end read have to match the filtering criterion in order for the pair to be filtered.

default
`any`

choices

- Any of the reads in a paired-end read have to match the filtering criterion: `any`
- Both of the reads in a paired-end read have to match the filtering criterion: `both`

Output results fastq

label
Reads file (forward)

type
`list:basic:file`

fastq2

label
Reads file (reverse)

type
`list:basic:file`

report

label
Cutadapt report

type
`basic:file`

fastqc_url

label
Quality control with FastQC (forward)

type
`list:basic:file:html`

fastqc_url2

label
Quality control with FastQC (reverse)

type
`list:basic:file:html`

fastqc_archive

label
Download FastQC archive (forward)

type
`list:basic:file`

fastqc_archive2

label
Download FastQC archive (reverse)

type
`list:basic:file`

Cutadapt (single-end)

`data:reads:fastq:single:cutadaptcutadapt-single` (*data:reads:fastq:single* **reads**, *data:seq:nucleotide* **up_primers_file**, *data:seq:nucleotide* **down_primers_file**, *list:basic:string* **up_primers_seq**, *list:basic:string* **down_primers_seq**, *basic:integer* **polya_tail**, *basic:integer* **min_overlap**, *basic:integer* **nextseq_trim**, *basic:integer* **leading**, *basic:integer* **trailing**, *basic:integer* **crop**, *basic:integer* **headcrop**, *basic:integer* **minlen**, *basic:integer* **maxlen**, *basic:integer* **max_n**, *basic:boolean* **match_read_wildcards**, *basic:boolean* **no_indels**, *basic:integer* **times**, *basic:decimal* **error_rate**) [Source: v2.5.2]

Cutadapt finds and removes adapter sequences, primers, poly-A tails and other types of unwanted sequence from high-throughput sequencing reads. More information about Cutadapt can be found [here](http://cutadapt.readthedocs.io/en/stable/).

Input arguments reads

label
Select sample(s)

type
`data:reads:fastq:single`

adapters.up_primers_file

label
5 prime adapter file

type
`data:seq:nucleotide`

required
False

adapters.down_primers_file

label
3 prime adapter file

type
`data:seq:nucleotide`

required
False

adapters.up_primers_seq

label
5 prime adapter sequence

type
`list:basic:string`

required
False

adapters.down_primers_seq

label
3 prime adapter sequence

type
`list:basic:string`

required
False

adapters.polya_tail

label
Poly-A tail

type
`basic:integer`

description
Length of poly-A tail, example - AAAN -> 3, AAAAAN -> 5

required
False

adapters.min_overlap

label
Minimal overlap

type
`basic:integer`

description
Minimum overlap for an adapter match

default
3

modify_reads.nextseq_trim

label
NextSeq-specific quality trimming

type
`basic:integer`

description
NextSeq-specific quality trimming (each read). Trims also dark cycles appearing as high-quality G bases. This option is mutually exclusive with the use of regular (-g) quality trimming.

required
False

modify_reads.leading

label
Quality on 5 prime

type
`basic:integer`

description
Remove low quality bases from 5 prime. Specifies the minimum quality required to keep a base. This option is mutually exclusive with the use of NextSeq-specific quality trimming.

required
False

modify_reads.trailing

label
Quality on 3 prime

type
`basic:integer`

description
Remove low quality bases from the 3 prime. Specifies the minimum quality required to keep a base. This option is mutually exclusive with the use of NextSeq-specific quality trimming.

required
False

modify_reads.crop

label
Crop

type
`basic:integer`

description
Cut the read to a specified length by removing bases from the end

required
False

modify_reads.headcrop

label
Headcrop

type
`basic:integer`

description
Cut the specified number of bases from the start of the read

required
False

filtering.minlen

label
Min length

type
`basic:integer`

description
Drop the read if it is below a specified length

required
False

filtering.maxlen

label
Max length

type
`basic:integer`

description
Drop the read if it is above a specified length.

required
False

filtering.max_n

label
Max numebr of N-s

type
`basic:integer`

description
Discard reads having more 'N' bases than specified.

required
False

filtering.match_read_wildcards

label
Match read wildcards

type
`basic:boolean`

description
Interpret IUPAC wildcards in reads.

required
False

default
`False`

filtering.no_indels

label
No indels

type
`basic:boolean`

description
Disable (disallow) insertions and deletions in adapters.

default
`False`

filtering.times

label
Times

type
`basic:integer`

description
Remove up to COUNT adapters from each read.

default
`1`

filtering.error_rate

label
Error rate

type
`basic:decimal`

description
Maximum allowed error rate (no. of errors divided by the length of the matching region).

default
`0.1`

Output results fastq

label
Reads file

type
`list:basic:file`

report

label
Cutadapt report

type
`basic:file`

fastqc_url

label
Quality control with FastQC

type
`list:basic:file:html`

fastqc_archive

label
Download FastQC archive

type
`list:basic:file`

Cutadapt - STAR - StringTie (Corall, paired-end)

`data:workflow:rnaseq:corallworkflow-corall-paired` (*data:reads:fastq:paired* **reads**, *data:index:star* **star_index**, *data:annotation* **annotation**, *data:index:star* **rrna_reference**, *data:index:star* **globin_reference**, *basic:integer* **quality_cutoff**, *basic:integer* **n_reads**, *basic:integer* **seed**, *basic:decimal* **fraction**, *basic:boolean* **two_pass**, *basic:string* **feature_class**, *basic:string* **id_attribute**)[Source: v5.2.0]

RNA-seq pipeline optimized for the Lexogen Corall Total RNA-Seq Library Prep Kit.

UMI-sequences are extracted from the raw reads before the reads are trimmed and quality filtered using Cutadapt. Preprocessed reads are aligned by the STAR aligner and de-duplicated using UMI-tools. Gene abundance estimates are reported by the featureCounts tool.

QC operates on downsampled reads and includes alignment of input reads to the rRNA/globin reference sequences. The reported alignment rate is used to asses the rRNA/globin sequence depletion rate.

The analysis results and QC reports are summarized by the MultiQC.

Input arguments reads

label
Select sample(s)

type
`data:reads:fastq:paired`

star_index

label
Genome

type
`data:index:star`

description
Genome index prepared by STAR aligner indexing tool.

annotation

label
Annotation

type
`data:annotation`

description
Genome annotation file (GTF).

rrna_reference

label
Indexed rRNA reference sequence

type
`data:index:star`

description
Reference sequence index prepared by STAR aligner indexing tool.

globin_reference

label
Indexed Globin reference sequence

type
`data:index:star`

description
Reference sequence index prepared by STAR aligner indexing tool.

cutadapt.quality_cutoff

label
Reads quality cutoff

type
`basic:integer`

description
Trim low-quality bases from 3' end of each read before adapter removal. Use this option when processing the data generated by older Illumina machines. The use of this option will override the NextSeq/NovaSeq-specific trimming procedure which is enabled by default and is recommended for Illumina machines that utilize 2-color chemistry to encode the four bases.

required
False

downsampling.n_reads

label
Number of reads

type
`basic:integer`

default
`1000000`

downsampling.seed

label
Seed

type
`basic:integer`

default
`11`

downsampling.fraction

label
Fraction

type
`basic:decimal`

description
Use the fraction of reads in range [0.0, 1.0] from the orignal input file instead of the absolute number of reads. If set, this will override the "Number of reads" input parameter.

required
False

downsampling.two_pass

label
2-pass mode

type
`basic:boolean`

description
Enable two-pass mode when down-sampling. Two-pass mode is twice as slow but with much reduced memory.

default
`False`

quantification.feature_class

label
Feature class

type
`basic:string`

description
Feature class (3rd column in GTF/GFF3 file) to be used. All other features will be ignored.

default
`exon`

quantification.id_attribute

label
ID attribute

type
`basic:string`

description
GTF/GFF3 attribute to be used as feature ID. Several GTF/GFF3 lines with the same feature ID will be considered as parts of the same feature. The feature ID is used to identify the counts in the output table. In GTF files this is usually 'gene_id', in GFF3 files this is often 'ID', and 'transcript_id' is frequently a valid choice for both annotation formats.

default
`gene_id`

choices

- gene_id: `gene_id`
- transcript_id: `transcript_id`
- ID: `ID`
- geneid: `geneid`

Output results

Cutadapt - STAR - StringTie (Corall, single-end)

`data:workflow:rnaseq:corallworkflow-corall-single` (*data:reads:fastq:single* **reads**, *data:index:star* **star_index**, *data:annotation* **annotation**, *data:index:star* **rrna_reference**, *data:index:star* **globin_reference**, *basic:integer* **quality_cutoff**, *basic:integer* **n_reads**, *basic:integer* **seed**, *basic:decimal* **fraction**, *basic:boolean* **two_pass**, *basic:string* **feature_class**, *basic:string* **id_attribute**)[Source: v5.2.0]

RNA-seq pipeline optimized for the Lexogen Corall Total RNA-Seq Library Prep Kit.

UMI-sequences are extracted from the raw reads before the reads are trimmed and quality filtered using Cutadapt. Preprocessed reads are aligned by the STAR aligner and de-duplicated using UMI-tools. Gene abundance estimates are reported by the featureCounts tool.

QC operates on downsampled reads and includes alignment of input reads to the rRNA/globin reference sequences. The reported alignment rate is used to asses the rRNA/globin sequence depletion rate.

The analysis results and QC reports are summarized by the MultiQC.

Input arguments reads

label
Select sample(s)

type
`data:reads:fastq:single`

star_index

label
Genome

type
`data:index:star`

description
Genome index prepared by STAR aligner indexing tool.

annotation

label
Annotation

type
`data:annotation`

description
Genome annotation file (GTF).

rrna_reference

label
Indexed rRNA reference sequence

type
`data:index:star`

description
Reference sequence index prepared by STAR aligner indexing tool.

globin_reference

label
Indexed Globin reference sequence

type
`data:index:star`

description
Reference sequence index prepared by STAR aligner indexing tool.

cutadapt.quality_cutoff

label
Reads quality cutoff

type
`basic:integer`

description
Trim low-quality bases from 3' end of each read before adapter removal. Use this option when processing the data generated by older Illumina machines. The use of this option will override the NextSeq/NovaSeq-specific trimming procedure which is enabled by default and is recommended for Illumina machines that utilize 2-color chemistry to encode the four bases.

required
False

downsampling.n_reads

label
Number of reads

type
`basic:integer`

default
`1000000`

downsampling.seed

label
Seed

type
`basic:integer`

default
`11`

downsampling.fraction

label
Fraction

type
`basic:decimal`

description
Use the fraction of reads in range [0.0, 1.0] from the orignal input file instead of the absolute number of reads. If set, this will override the "Number of reads" input parameter.

required
False

downsampling.two_pass

label
2-pass mode

type
`basic:boolean`

description
Enable two-pass mode when down-sampling. Two-pass mode is twice as slow but with much reduced memory.

default
`False`

quantification.feature_class

label
Feature class

type
`basic:string`

description
Feature class (3rd column in GTF/GFF3 file) to be used. All other features will be ignored.

default
`exon`

quantification.id_attribute

label
ID attribute

type
`basic:string`

description
GTF/GFF3 attribute to be used as feature ID. Several GTF/GFF3 lines with the same feature ID will be considered as parts of the same feature. The feature ID is used to identify the counts in the output table. In GTF files this is usually 'gene_id', in GFF3 files this is often 'ID', and 'transcript_id' is frequently a valid choice for both annotation formats.

default
`gene_id`

choices

- gene_id: `gene_id`
- transcript_id: `transcript_id`
- ID: `ID`
- geneid: `geneid`

Output results

DESeq2

`data:differentialexpression:deseq2:differentialexpression-deseq2` (*list:data:expression* **case**, *list:data:expression* **control**, *basic:boolean* **create_sets**, *basic:decimal* **logfc**, *basic:decimal* **fdr**, *basic:boolean* **beta_prior**, *basic:boolean* **count**, *basic:integer* **min_count_sum**, *basic:boolean* **cook**, *basic:decimal* **cooks_cutoff**, *basic:boolean* **independent**, *basic:decimal* **alpha**) [Source: v3.6.0]

Run DESeq2 analysis.

The DESeq2 package estimates variance-mean dependence in count data from high-throughput sequencing assays and tests for differential expression based on a model using the negative binomial distribution. See [here](https://www.bioconductor.org/packages/release/bioc/manuals/DESeq2/man/DESeq2.pdf) and [here](http://bioconductor.org/packages/devel/bioc/vignettes/DESeq2/inst/doc/DESeq2.html) for more information.

Input arguments case

label
Case

type
`list:data:expression`

description
Case samples (replicates)

required
True

disabled
False

hidden
False

control

label
Control

type
`list:data:expression`

description
Control samples (replicates)

required
True

disabled
False

hidden
False

create_sets

label
Create gene sets

type
`basic:boolean`

description
After calculating differential gene expressions create gene sets for up-regulated genes, down-regulated genes and all genes.

required
True

disabled
False

hidden
False

default
`False`

logfc

label
Log2 fold change threshold for gene sets

type
`basic:decimal`

description
Genes above Log2FC are considered as up-regulated and genes below -Log2FC as down-regulated.

required
True

disabled
False

hidden
!create_sets

default
`1.0`

fdr

label
FDR threshold for gene sets

type
`basic:decimal`

required
True

disabled
False

hidden
!create_sets

default
`0.05`

options.beta_prior

label
Beta prior

type
`basic:boolean`

description
Whether or not to put a zero-mean normal prior on the non-intercept coefficients.

required
True

disabled
False

hidden
False

default
`False`

filter_options.count

label
Filter genes based on expression count

type
`basic:boolean`

required
True

disabled
False

hidden
False

default
`True`

filter_options.min_count_sum

label
Minimum gene expression count summed over all samples

type
`basic:integer`

description
Filter genes in the expression matrix input. Remove genes where the expression count sum over all samples is below the threshold.

required
True

disabled
False

hidden
!filter_options.count

default
`10`

filter_options.cook

label
Filter genes based on Cook's distance

type
`basic:boolean`

required
True

disabled
False

hidden
False

default
`False`

filter_options.cooks_cutoff

label
Threshold on Cook's distance

type
`basic:decimal`

description
If one or more samples have Cook's distance larger than the threshold set here, the p-value for the row is set to NA. If left empty, the default threshold of 0.99 quantile of the F(p, m-p) distribution is used, where p is the number of coefficients being fitted and m is the number of samples. This test excludes Cook's distance of samples belonging to experimental groups with only two samples.

required
False

disabled
False

hidden
!filter_options.cook

filter_options.independent

label
Apply independent gene filtering

type
`basic:boolean`

required
True

disabled
False

hidden
False

default
`True`

filter_options.alpha

label
Significance cut-off used for optimizing independent gene filtering

type
`basic:decimal`

description
The value should be set to adjusted p-value cut-off (FDR).

required
True

disabled
False

hidden
!filter_options.independent

default
`0.1`

Output results raw

label
Differential expression

type
`basic:file`

required
True

disabled
False

hidden
False

de_json

label
Results table (JSON)

type
`basic:json`

required
True

disabled
False

hidden
False

de_file

label
Results table (file)

type
`basic:file`

required
True

disabled
False

hidden
False

count_matrix

label
Count matrix

type
`basic:file`

required
True

disabled
False

hidden
False

count_matrix_normalized

label
Normalized count matrix (median of ratios)

type
`basic:file`

required
True

disabled
False

hidden
False

source

label
Gene ID database

type
`basic:string`

required
True

disabled
False

hidden
False

species

label
Species

type
`basic:string`

required
True

disabled
False

hidden
False

build

label
Build

type
`basic:string`

required
True

disabled
False

hidden
False

feature_type

label
Feature type

type
`basic:string`

required
True

disabled
False

hidden
False

Detect library strandedness

`data:strandednesslibrary-strandedness` (*data:reads:fastq* **reads**, *basic:integer* **read_number**, *data:index:salmon* **salmon_index**) [Source: v0.6.2]

This process uses the Salmon transcript quantification tool to automatically infer the NGS library strandedness. For more details, please see the Salmon [documentation](https://salmon.readthedocs.io/en/latest/library_type.html)

Input arguments reads

label
Sequencing reads

type
`data:reads:fastq`

description
Sequencing reads in .fastq format. Both single and paired-end libraries are supported

read_number

label
Number of input reads

type
`basic:integer`

description
Number of sequencing reads that are subsampled from each of the original .fastq files before library strand detection

default
`50000`

salmon_index

label
Transcriptome index file

type
`data:index:salmon`

description
Transcriptome index file created using the Salmon indexing tool. cDNA (transcriptome) sequences used for index file creation must be derived from the same species as the input sequencing reads to obtain the reliable analysis results

Output results strandedness

label
Library strandedness type

type
`basic:string`

description
The predicted library strandedness type. The codes U and IU indicate 'strand non-specific' library for single or paired-end reads, respectively. Codes SF and ISF correspond to the 'strand-specific forward' library, for the single or paired-end reads, respectively. For 'strand-specific reverse' library, the corresponding codes are SR and ISR. For more details, please see the Salmon [documentation](https://salmon.readthedocs.io/en/latest/library_type.html)

fragment_ratio

label
Compatible fragment ratio

type
`basic:decimal`

description
The ratio of fragments that support the predicted library strandedness type

log

label
Log file

type
`basic:file`

description
Analysis log file.

Dictyostelium expressions

data:expression:polyaexpression-dicty (*data:alignment:bam* **alignment**, *data:annotation:gff3* **gff**, *data:mappability:bcm* **mappable**) [Source: v1.4.2]

Dictyostelium-specific pipeline. Developed by Bioinformatics Laboratory, Faculty of Computer and Information Science, University of Ljubljana, Slovenia and Shaulsky Lab, Department of Molecular and Human Genetics, Baylor College of Medicine, Houston, TX, USA.

Input arguments alignment

label
Aligned sequence

type
`data:alignment:bam`

gff

label
Features (GFF3)

type
`data:annotation:gff3`

mappable

label
Mappability

type
`data:mappability:bcm`

Output results exp

label
Expression RPKUM (polyA)

type
`basic:file`

description
mRNA reads scaled by uniquely mappable part of exons.

rpkmpolya

label
Expression RPKM (polyA)

type
`basic:file`

description
mRNA reads scaled by exon length.

rc

label
Read counts (polyA)

type
`basic:file`

description
mRNA reads uniquely mapped to gene exons.

rpkum

label
Expression RPKUM

type
`basic:file`

description
Reads scaled by uniquely mappable part of exons.

rpkm

label
Expression RPKM

type
`basic:file`

description
Reads scaled by exon length.

rc_raw

label
Read counts (raw)

type
`basic:file`

description
Reads uniquely mapped to gene exons.

exp_json

label
Expression RPKUM (polyA) (json)

type
`basic:json`

exp_type

label
Expression Type (default output)

type
`basic:string`

source

label
Gene ID database

type
`basic:string`

species

label
Species

type
`basic:string`

build

label
Build

type
`basic:string`

feature_type

label
Feature type

type
`basic:string`

Differential Expression (table)

`data:differentialexpression:uploadupload-diffexp` (*basic:file* **src**, *basic:string* **gene_id**, *basic:string* **logfc**, *basic:string* **fdr**, *basic:string* **logodds**, *basic:string* **fwer**, *basic:string* **pvalue**, *basic:string* **stat**, *basic:string* **source**, *basic:string* **species**, *basic:string* **build**, *basic:string* **feature_type**, *list:data:expression* **case**, *list:data:expression* **control**) [Source: v1.5.1]

Upload Differential Expression table.

Input arguments src

label
Differential expression file

type
`basic:file`

description
Differential expression file. Supported file types: *.xls, *.xlsx, *.tab (tab-delimited file), *.diff. DE file must include columns with log2(fold change) and FDR or pval information. DE file must contain header row with column names. Accepts DESeq, DESeq2, edgeR and CuffDiff output files.

validate_regex
`\.(xls|xlsx|tab|tab.gz|diff|diff.gz)$`

gene_id

label
Gene ID label

type
`basic:string`

logfc

label
LogFC label

type
`basic:string`

fdr

label
FDR label

type
`basic:string`

required
False

logodds

label
LogOdds label

type
`basic:string`

required
False

fwer

label
FWER label

type
`basic:string`

required
False

pvalue

label
Pvalue label

type
`basic:string`

required
False

stat

label
Statistics label

type
`basic:string`

required
False

source

label
Gene ID database

type
`basic:string`

choices

- AFFY: `AFFY`
- DICTYBASE: `DICTYBASE`
- ENSEMBL: `ENSEMBL`
- NCBI: `NCBI`
- UCSC: `UCSC`

species

label
Species

type
`basic:string`

description
Species latin name.

choices

- Homo sapiens: `Homo sapiens`
- Mus musculus: `Mus musculus`
- Rattus norvegicus: `Rattus norvegicus`
- Dictyostelium discoideum: `Dictyostelium discoideum`
- Odocoileus virginianus texanus: `Odocoileus virginianus texanus`
- Solanum tuberosum: `Solanum tuberosum`

build

label
Build

type
`basic:string`

description
Genome build or annotation version.

feature_type

label
Feature type

type
`basic:string`

default
`gene`

choices

- gene: `gene`
- transcript: `transcript`

- exon: `exon`

case

label
Case

type
`list:data:expression`

description
Case samples (replicates)

required
False

control

label
Control

type
`list:data:expression`

description
Control samples (replicates)

required
False

Output results raw

label
Differential expression

type
`basic:file`

de_json

label
Results table (JSON)

type
`basic:json`

de_file

label
Results table (file)

type
`basic:file`

source

label
Gene ID database

type
`basic:string`

species

label
Species

type
basic:string

build

label
Build

type
basic:string

feature_type

label
Feature type

type
basic:string

Differential expression of shRNA

data:shrna:differentialexpression:differentialexpression-shrna (*data:file* **parameter_file**, *list:data:expression:shrna2quant:* **expression_data**) [Source: v1.3.0]

Performing differential expression on a list of objects.

Analysis starts by inputting a set of expression files (count matrices) and a parameter file. Parameter file is an xlsx file and consists of tabs: - `sample_key`: Should have column sample with exact sample name as input expression file(s), columns defining treatment and lastly a column which indicates replicate. - `contrasts`: Define groups which will be used to perform differential expression analysis. Model for DE uses these contrasts and replicate number. In R annotation, this would be ` ~ 1 + group + replicate`. Table should have two columns named `group_1` and `group_2`. - `overall_contrasts`: This is a layer "above" `contrasts`, where results from two contrasts are compared for lethal, beneficial and neutral species. Thresholds governing classification can be found in `classification_parameters` tab. - `classification_parameters`: This tab holds three columns, `threshold`, `value` and `description`. Only the first two are used in the workflow, description is for your benefit.

This process outputs DESeq2 results, classified results based on provided thresholds and counts of beneficial and lethal species.

Input arguments parameter_file

label
Excel parameter file (.xlsx)

type
data:file

description
Select .xlsx file which holds parameters for analysis. See [here](https://github.com/genialis/shRNAde/blob/master/inst/extdata/template_doDE_inputs.xlsx) for a template.

required
True

disabled
False

hidden
False

expression_data

label
List of expression files from shrna2quant

type
`list:data:expression:shrna2quant:`

required
True

disabled
False

hidden
False

Output results deseq_results

label
DESeq2 results

type
`basic:file`

required
True

disabled
False

hidden
False

class_results

label
Results classified based on thresholds provided by the user

type
`basic:file`

required
True

disabled
False

hidden
False

beneficial_counts

label
shRNAs considered as beneficial based on user input

type
`basic:file`

required
True

disabled
False

hidden
False

lethal_counts

label
shRNAs considered as lethal based on user input

type
`basic:file`

required
True

disabled
False

hidden
False

Ensembl Variant Effect Predictor

`data:variants:vcf:vep:ensembl-vep` (*data:variants:vcf* **vcf**, *data:vep:cache* **cache**, *data:seq:nucleotide* **ref_seq**, *basic:integer* **n_forks**) [Source: v2.1.0]

Run Ensembl-VEP.

VEP (Variant Effect Predictor) determines the effect of your variants (SNPs, insertions, deletions, CNVs or structural variants) on genes, transcripts, and protein sequence, as well as regulatory regions.

This process accepts VCF file and VEP cache directory to produce VCF file with annotated variants, its index and summary of the procces.

Input arguments vcf

label
Input VCF file

type
`data:variants:vcf`

required
True

disabled
False

hidden
False

cache

label
Cache directory for Ensembl-VEP

type
`data:vep:cache`

required
True

disabled
False

hidden
False

ref_seq

label
Reference sequence

type
`data:seq:nucleotide`

required
True

disabled
False

hidden
False

n_forks

label
Number of forks

type
`basic:integer`

description
Using forking enables VEP to run multiple parallel threads, with each thread processing a subset of your input. Forking can dramatically improve runtime.

required
True

disabled
False

hidden
False

default
`2`

Output results vcf

label
Annotated VCF file

type
`basic:file`

required
True

disabled
False

hidden
False

tbi

label
Tabix index

type
`basic:file`

required
True

disabled
False

hidden
False

summary

label
Summary of the analysis

type
`basic:file:html`

required
True

disabled
False

hidden
False

species

label
Species

type
`basic:string`

required
True

disabled
False

hidden
False

build

label
Build

type
`basic:string`

required
True

disabled
False

hidden
False

Ensembl-VEP cache directory

`data:vep:cache:upload-vep-cache` (*basic:file* **cache_file**, *basic:string* **species**, *basic:string* **build**, *basic:string* **release**) [Source: v1.1.0]

Import VEP cache directory.

Input arguments cache_file

label
Compressed cache directory

type
`basic:file`

required
True

disabled
False

hidden
False

species

label
Species

type
`basic:string`

description
Select a species name from the dropdown menu.

required
True

disabled
False

hidden
False

default
`Homo sapiens`

choices

- Homo sapiens: `Homo sapiens`
- Mus musculus: `Mus musculus`
- Rattus norvegicus: `Rattus norvegicus`

build

label
Genome build

type
`basic:string`

required
True

disabled
False

hidden
False

release

label
Cache release

type
`basic:string`

required
True

disabled
False

hidden
False

Output results cache

label
Cache directory

type
`basic:dir`

required
True

disabled
False

hidden
False

species

label
Species

type
`basic:string`

required
True

disabled
False

hidden
False

build

label
Build

type
`basic:string`

required
True

disabled
False

hidden
False

release

label
Cache release

type
`basic:string`

required
True

disabled
False

hidden
False

Expression Time Course

`data:etcetc-bcm` (*list:data:expression* **expressions**, *basic:boolean* **avg**) [Source: v1.2.2]

Select gene expression data and form a time course.

Input arguments expressions

label
RPKM expression profile

type
`list:data:expression`

required
True

avg

label
Average by time

type
`basic:boolean`

default
`True`

Output results etcfile

label
Expression time course file

type
`basic:file`

etc

label
Expression time course

type
`basic:json`

Expression aggregator

`data:aggregator:expressionexpression-aggregator` (*list:data:expression* **exps**, *basic:string* **group_by**, *data:aggregator:expression* **expr_aggregator**) [Source: v0.5.1]

Collect expression data from samples grouped by sample descriptor field. The Expression aggregator process should not be run in Batch Mode, as this will create redundant outputs. Rather, select multiple samples below for which you wish to aggregate the expression matrix.

Input arguments exps

label
Expressions

type
`list:data:expression`

group_by

label
Sample descriptor field

type
`basic:string`

expr_aggregator

label
Expression aggregator

type
`data:aggregator:expression`

required
False

Output results exp_matrix

label
Expression matrix

type
`basic:file`

box_plot

label
Box plot

type
`basic:json`

log_box_plot

label
Log box plot

type
`basic:json`

source

label
Gene ID database

type
`basic:string`

species

label
Species

type
`basic:string`

exp_type

label
Expression type

type
`basic:string`

Expression matrix

`data:expressionsetmergeexpressions` (*list:data:expression* **exps**, *list:basic:string* **genes**) [Source: v1.4.2]

Merge expression data to create an expression matrix where each column represents all the gene expression levels from a single experiment, and each row represents the expression of a gene across all experiments.

Input arguments exps

label
Gene expressions

type
`list:data:expression`

genes

label
Filter genes

type
`list:basic:string`

required
False

Output results expset

label
Expression set

type
`basic:file`

expset_type

label
Expression set type

type
`basic:string`

Expression time course

`data:etcupload-etc` (*basic:file* **src**)[Source: v1.4.1]

Upload Expression time course.

Input arguments src

label
Expression time course file (xls or tab)

type
`basic:file`

description
Expression time course

required
True

validate_regex
`\.(xls|xlsx|tab)$`

Output results etcfile

label
Expression time course file

type
`basic:file`

etc

label
Expression time course

type
`basic:json`

FASTA file

`data:seq:nucleotide:upload-fasta-nucl` (*basic:file* **src**, *basic:string* **species**, *basic:string* **build**)[Source: v3.2.0]

Import nucleotide sequence file in FASTA format.

FASTA file is a text-based format for representing nucleotide sequences, in which nucleotides are represented using single-letter codes. The uploaded FASTA file can hold multiple nucleotide sequences.

Input arguments src

label
Sequence file (FASTA)

type
`basic:file`

required
True

disabled
False

hidden
False

species

label
Species

type
`basic:string`

description
Select a species name from the dropdown menu or write a custom species name in the species field. For sequences that are not related to any particular species (e.g. adapters file), you can select the value Other.

required
True

disabled
False

hidden
False

choices

- Homo sapiens: `Homo sapiens`
- Mus musculus: `Mus musculus`
- Rattus norvegicus: `Rattus norvegicus`
- Macaca mulatta: `Macaca mulatta`
- Dictyostelium discoideum: `Dictyostelium discoideum`
- Other: `Other`

build

label
Genome build

type
`basic:string`

description
Enter a genome build information associated with the uploaded sequence(s).

required
True

disabled
False

hidden
False

Output results fastagz

label
FASTA file (compressed)

type
`basic:file`

required
True

disabled
False

hidden
False

fasta

label
FASTA file

type
`basic:file`

required
True

disabled
False

hidden
False

fai

label
FASTA file index

type
`basic:file`

required
True

disabled
False

hidden
False

fasta_dict

label
FASTA dictionary

type
`basic:file`

required
True

disabled
False

hidden
False

num_seqs

label
Number of sequences

type
`basic:integer`

required
True

disabled
False

hidden
False

species

label
Species

type
`basic:string`

required
True

disabled
False

hidden
False

build

label
Build

type
`basic:string`

required
True

disabled
False

hidden
False

FASTQ file (paired-end)

`data:reads:fastq:paired:upload-fastq-paired` (*list:basic:file* **src1**, *list:basic:file* **src2**, *basic:boolean* **merge_lanes**) [Source: v2.6.0]

Import paired-end reads in FASTQ format.

Import paired-end reads in FASTQ format, which is a text-based format for storing both a biological sequence (usually nucleotide sequence) and its corresponding quality scores.

Input arguments src1

label
Mate1

type
`list:basic:file`

description
Sequencing reads in FASTQ format. Supported extensions: .fastq.gz (preferred), .fq.* or .fastq.*

required
True

disabled
False

hidden
False

src2

label
Mate2

type
`list:basic:file`

description
Sequencing reads in FASTQ format. Supported extensions: .fastq.gz (preferred), .fq.* or .fastq.*

required
True

disabled
False

hidden
False

merge_lanes

label
Merge lanes

type
`basic:boolean`

description
Merge sample data split into multiple sequencing lanes into a single FASTQ file.

required
True

disabled
False

hidden
False

default
`False`

Output results fastq

label
Reads file (mate 1)

type
`list:basic:file`

required
True

disabled
False

hidden
False

fastq2

label
Reads file (mate 2)

type
`list:basic:file`

required
True

disabled
False

hidden
False

fastqc_url

label
Quality control with FastQC (Upstream)

type
`list:basic:file:html`

required
True

disabled
False

hidden
False

fastqc_url2

label
Quality control with FastQC (Downstream)

type
`list:basic:file:html`

required
True

disabled
False

hidden
False

fastqc_archive

label
Download FastQC archive (Upstream)

type
`list:basic:file`

required
True

disabled
False

hidden
False

fastqc_archive2

label
Download FastQC archive (Downstream)

type
`list:basic:file`

required
True

disabled
False

hidden
False

FASTQ file (single-end)

`data:reads:fastq:single:upload-fastq-single` (*list:basic:file* **src**, *basic:boolean* **merge_lanes**)[Source: v2.6.0]

Import single-end reads in FASTQ format.

Import single-end reads in FASTQ format, which is a text-based format for storing both a biological sequence (usually nucleotide sequence) and its corresponding quality scores.

Input arguments src

label
Reads

type
`list:basic:file`

description
Sequencing reads in FASTQ format. Supported extensions: .fastq.gz (preferred), .fq.* or .fastq.*

required
True

disabled
False

hidden
False

merge_lanes

label
Merge lanes

type
`basic:boolean`

description
Merge sample data split into multiple sequencing lanes into a single FASTQ file.

required
True

disabled
False

hidden
False

default
`False`

Output results fastq

label
Reads file

type
`list:basic:file`

required
True

disabled
False

hidden
False

fastqc_url

label
Quality control with FastQC

type
`list:basic:file:html`

required
True

disabled
False

hidden
False

fastqc_archive

label
Download FastQC archive

type
`list:basic:file`

required
True

disabled
False

hidden
False

Find similar genes

`data:similarexpression:find-similar` (*list:data:expression* **expressions**, *basic:string* **gene**, *basic:string* **distance**) [Source: v1.3.1]

Find genes with similar expression profile.

Find genes that have similar expression over time to the query gene.

Input arguments expressions

label
Time series relation

type
`list:data:expression`

description
Select time course to which the expressions belong to.

required
True

disabled
False

hidden
False

gene

label
Query gene

type
`basic:string`

description
Select a gene to which others are compared.

required
True

disabled
False

hidden
False

distance

label
Distance metric

type
`basic:string`

required
True

disabled
False

hidden
False

default
`spearman`

choices

- Euclidean: `euclidean`
- Spearman: `spearman`
- Pearson: `pearson`

Output results similar_genes

label
Similar genes

type
`basic:json`

required
True

disabled
False

hidden
False

source

label
Gene ID database

type
`basic:string`

required
True

disabled
False

hidden
False

species

label
Species

type
`basic:string`

required
True

disabled
False

hidden
False

build

label
Build

type
`basic:string`

required
True

disabled
False

hidden
False

feature_type

label
Feature type

type
`basic:string`

required
True

disabled
False

hidden
False

GAF file

`data:gaf:2:0upload-gaf` (*basic:file* **src**, *basic:string* **source**, *basic:string* **species**) [Source: v1.4.0]

GO annotation file (GAF v2.0) relating gene ID and associated GO terms

Input arguments src

label
GO annotation file (GAF v2.0)

type
`basic:file`

description
Upload GO annotation file (GAF v2.0) relating gene ID and associated GO terms

source

label
Gene ID database

type
`basic:string`

choices

- AFFY: `AFFY`
- DICTYBASE: `DICTYBASE`
- ENSEMBL: `ENSEMBL`
- MGI: `MGI`
- NCBI: `NCBI`
- UCSC: `UCSC`
- UniProtKB: `UniProtKB`

species

label
Species

type
`basic:string`

Output results gaf

label
GO annotation file (GAF v2.0)

type
`basic:file`

gaf_obj

label
GAF object

type
`basic:file`

source

label
Gene ID database

type
`basic:string`

species

label
Species

type
`basic:string`

GATK GenomicsDBImport

`data:genomicsdb:gatk-genomicsdb-import` (*list:data:variants:gvcf* **gvcfs**, *data:bed* **intervals**, *basic:boolean* **use_existing**, *data:genomicsdb* **existing_db**, *basic:integer* **batch_size**, *basic:boolean* **consolidate**, *basic:integer* **max_heap_size**, *basic:boolean* **use_cms_gc**) [Source: v1.3.0]

Import single-sample GVCFs into GenomicsDB before joint genotyping.

Input arguments gvcfs

label
Input data (GVCF)

type
`list:data:variants:gvcf`

required
True

disabled
False

hidden
False

intervals

label
Intervals file (.bed)

type
`data:bed`

description
Intervals file is required if a new database will be created.

required
False

disabled
False

hidden
False

use_existing

label
Add new samples to an existing GenomicsDB workspace

type
`basic:boolean`

required
True

disabled
False

hidden
False

default
`False`

existing_db

label
Select a GATK GenomicsDB object

type
`data:genomicsdb`

description
Instead of creating a new database the GVCFs are added to this database and a new GenomicsDB object is created.

required
False

disabled
False

hidden
!use_existing

advanced_options.batch_size

label
Batch size

type
`basic:integer`

description
Batch size controls the number of samples for which readers are open at once and therefore provides a way to minimize memory consumption. However, it can take longer to complete. Use the consolidate flag if more than a hundred batches were used. This will improve feature read time. batchSize=0 means no batching (i.e. readers for all samples will be opened at once).

required
True

disabled
False

hidden
False

default
`0`

advanced_options.consolidate

label
Consolidate

type
`basic:boolean`

description
Boolean flag to enable consolidation. If importing data in batches, a new fragment is created for each batch. In case thousands of fragments are created, GenomicsDB feature readers will try to open ~20x as many files. Also, internally GenomicsDB would consume more memory to maintain bookkeeping data from all fragments. Use this flag to merge all fragments into one. Merging can

potentially improve read performance, however overall benefit might not be noticeable as the top Java layers have significantly higher overheads. This flag has no effect if only one batch is used.

required
True

disabled
False

hidden
False

default
`False`

advanced_options.max_heap_size

label
Java maximum heap size in GB (Xmx)

type
`basic:integer`

description
Set the maximum Java heap size.

required
True

disabled
False

hidden
False

default
`28`

advanced_options.use_cms_gc

label
Use CMS Garbage Collector in Java

type
`basic:boolean`

description
The Concurrent Mark Sweep (CMS) implementation uses multiple garbage collector threads for garbage collection.

required
True

disabled
False

hidden
False

default
`True`

Output results database

label
GenomicsDB workspace

type
`basic:dir`

required
True

disabled
False

hidden
False

intervals

label
Intervals file

type
`basic:file`

required
True

disabled
False

hidden
False

species

label
Species

type
`basic:string`

required
True

disabled
False

hidden
False

build

label
Build

type
`basic:string`

required
True

disabled
False

hidden
False

GATK GenotypeGVCFs

`data:variants:vcf:genotypegvcfs:gatk-genotype-gvcfs` (*data:genomicsdb* **database**, *data:seq:nucleotide* **ref_seq**, *data:variants:vcf* **dbsnp**, *basic:integer* **n_jobs**, *basic:integer* **max_heap_size**) [Source: v2.3.0]

Consolidate GVCFs and run joint calling using GenotypeGVCFs tool.

Input arguments database

label
GATK GenomicsDB

type
`data:genomicsdb`

required
True

disabled
False

hidden
False

ref_seq

label
Reference sequence

type
`data:seq:nucleotide`

required
True

disabled
False

hidden
False

dbsnp

label
dbSNP file

type
`data:variants:vcf`

required
True

disabled
False

hidden
False

advanced_options.n_jobs

label
Number of concurent jobs

type
`basic:integer`

description
Use a fixed number of jobs for genotyping instead of determining it based on the number of available cores.

required
False

disabled
False

hidden
False

advanced_options.max_heap_size

label
Java maximum heap size in GB (Xmx)

type
`basic:integer`

description
Set the maximum Java heap size.

required
True

disabled
False

hidden
False

default
`28`

Output results vcf

label
GVCF file

type
`basic:file`

required
True

disabled
False

hidden
False

vcf_dir

label
Folder with split GVCFs

type
`basic:dir`

required
True

disabled
False

hidden
False

tbi

label
Tabix index

type
`basic:file`

required
True

disabled
False

hidden
False

species

label
Species

type
`basic:string`

required
True

disabled
False

hidden
False

build

label
Build

type
`basic:string`

required
True

disabled
False

hidden
False

GATK HaplotypeCaller (GVCF)

data:variants:gvcf:gatk-haplotypecaller-gvcf (*data:alignment:bam* **bam**, *data:seq:nucleotide* **ref_seq**, *data:bed* **intervals**, *basic:decimal* **contamination**) [Source: v1.3.0]

Run GATK HaplotypeCaller in GVCF mode.

Input arguments bam

label
Analysis ready BAM file

type
`data:alignment:bam`

required
True

disabled
False

hidden
False

ref_seq

label
Reference sequence

type
`data:seq:nucleotide`

required
True

disabled
False

hidden
False

options.intervals

label
Use intervals BED file to limit the analysis to the specified parts of the genome.

type
`data:bed`

required
False

disabled
False

hidden
False

options.contamination

label
Contamination fraction

type
`basic:decimal`

description
Fraction of contamination in sequencing data (for all samples) to aggressively remove.

required
True

disabled
False

hidden
False

default
`0`

Output results vcf

label
GVCF file

type
`basic:file`

required
True

disabled
False

hidden
False

tbi

label
Tabix index

type
`basic:file`

required
True

disabled
False

hidden
False

species

label
Species

type
`basic:string`

required
True

disabled
False

hidden
False

build

label
Build

type
`basic:string`

required
True

disabled
False

hidden
False

GATK MergeVcfs

`data:variants:vcf:mergevcfs:gatk-merge-vcfs` (*list:data:variants:vcf* **vcfs**, *data:seq:nucleotide* **ref_seq**, *basic:integer* **java_gc_threads**, *basic:integer* **max_heap_size**) [Source: v1.2.0]

Combine multiple variant files into a single variant file using GATK MergeVcfs.

Input arguments vcfs

label
Input data (VCFs)

type
`list:data:variants:vcf`

required
True

disabled
False

hidden
False

advanced_options.ref_seq

label
Reference sequence

type
`data:seq:nucleotide`

description
Optionally use a sequence dictionary file (.dict) if the input VCF does not contain a complete contig list.

required
False

disabled
False

hidden
False

advanced_options.java_gc_threads

label
Java ParallelGCThreads

type
`basic:integer`

description
Sets the number of threads used during parallel phases of the garbage collectors.

required
True

disabled
False

hidden
False

default
`2`

advanced_options.max_heap_size

label
Java maximum heap size (Xmx)

type
`basic:integer`

description
Set the maximum Java heap size (in GB).

required
True

disabled
False

hidden
False

default
`12`

Output results vcf

label
Merged VCF

type
`basic:file`

required
True

disabled
False

hidden
False

tbi

label
Tabix index

type
`basic:file`

required
True

disabled
False

hidden
False

species

label
Species

type
`basic:string`

required
True

disabled
False

hidden
False

build

label
Build

type
`basic:string`

required
True

disabled
False

hidden
False

GATK SelectVariants (multi-sample)

`data:variants:vcf:selectvariants:gatk-select-variants` (*data:variants:vcf* **vcf**, *data:bed* **intervals**, *list:basic:string* **select_type**, *basic:boolean* **exclude_filtered**, *data:seq:nucleotide* **ref_seq**, *basic:integer* **java_gc_threads**, *basic:integer* **max_heap_size**) [Source: v1.2.0]

Select a subset of variants based on various criteria using GATK SelectVariants.

This tool works with multi-sample VCF file as an input.

Input arguments vcf

label
Input data (VCF)

type
`data:variants:vcf`

required
True

disabled
False

hidden
False

intervals

label
Intervals file (.bed)

type
`data:bed`

description
One or more genomic intervals over which to operate. This can also be used to get data from a specific interval.

required
False

disabled
False

hidden
False

select_type

label
Select only a certain type of variants from the input file

type
`list:basic:string`

description
This argument selects particular kinds of variants out of a list. If left empty, there is no type selection and all variant types are considered for other selection criteria. Valid types are INDEL, SNP, MIXED, MNP, SYMBOLIC, NO_VARIATION. Can be specified multiple times.

required
False

disabled
False

hidden
False

exclude_filtered

label
Don't include filtered sites

type
`basic:boolean`

description
If this flag is enabled, sites that have been marked as filtered (i.e. have anything other than `.` or `PASS` in the FILTER field) will be excluded from the output.

required
True

disabled
False

hidden
False

default
`False`

advanced_options.ref_seq

label
Reference sequence

type
`data:seq:nucleotide`

required
False

disabled
False

hidden
False

advanced_options.java_gc_threads

label
Java ParallelGCThreads

type
`basic:integer`

description
Sets the number of threads used during parallel phases of the garbage collectors.

required
True

disabled
False

hidden
False

default
2

advanced_options.max_heap_size

label
Java maximum heap size (Xmx)

type
`basic:integer`

description
Set the maximum Java heap size (in GB).

required
True

disabled
False

hidden
False

default
`12`

Output results vcf

label
Selected variants (VCF)

type
`basic:file`

required
True

disabled
False

hidden
False

tbi

label
Tabix index

type
`basic:file`

required
True

disabled
False

hidden
False

species

label
Species

type
`basic:string`

required
True

disabled
False

hidden
False

build

label
Build

type
`basic:string`

required
True

disabled
False

hidden
False

GATK SelectVariants (single-sample)

`data:variants:vcf:selectvariants:single:gatk-select-variants-single` (*data:variants:vcf* **vcf**, *data:bed* **intervals**, *list:basic:string* **select_type**, *basic:boolean* **exclude_filtered**, *data:seq:nucleotide* **ref_seq**, *basic:integer* **java_gc_threads**, *basic:integer* **max_heap_size**) [Source: v1.1.0]

Select a subset of variants based on various criteria using GATK SelectVariants.

This tool works with single-sample VCF file as an input.

Input arguments vcf

label
Input data (VCF)

type
`data:variants:vcf`

required
True

disabled
False

hidden
False

intervals

label
Intervals file (.bed)

type
`data:bed`

description
One or more genomic intervals over which to operate. This can also be used to get data from a specific interval.

required
False

disabled
False

hidden
False

select_type

label
Select only a certain type of variants from the input file

type
`list:basic:string`

description
This argument selects particular kinds of variants out of a list. If left empty, there is no type selection and all variant types are considered for other selection criteria. Valid types are INDEL, SNP, MIXED, MNP, SYMBOLIC, NO_VARIATION. Can be specified multiple times.

required
False

disabled
False

hidden
False

exclude_filtered

label
Don't include filtered sites

type
`basic:boolean`

description
If this flag is enabled, sites that have been marked as filtered (i.e. have anything other than `.` or `PASS` in the FILTER field) will be excluded from the output.

required
True

disabled
False

hidden
False

default
`False`

advanced_options.ref_seq

label
Reference sequence

type
`data:seq:nucleotide`

required
False

disabled
False

hidden
False

advanced_options.java_gc_threads

label
Java ParallelGCThreads

type
`basic:integer`

description
Sets the number of threads used during parallel phases of the garbage collectors.

required
True

disabled
False

hidden
False

default
`2`

advanced_options.max_heap_size

label
Java maximum heap size (Xmx)

type
`basic:integer`

description
Set the maximum Java heap size (in GB).

required
True

disabled
False

hidden
False

default
`12`

Output results vcf

label
Selected variants (VCF)

type
`basic:file`

required
True

disabled
False

hidden
False

tbi

label
Tabix index

type
`basic:file`

required
True

disabled
False

hidden
False

species

label
Species

type
`basic:string`

required
True

disabled
False

hidden
False

build

label
Build

type
`basic:string`

required
True

disabled
False

hidden
False

GATK SplitNCigarReads

`data:alignment:bam:splitncigar:gatk-split-ncigar` (*data:alignment:bam* **bam**, *data:seq:nucleotide* **ref_seq**, *basic:integer* **java_gc_threads**, *basic:integer* **max_heap_size**) [Source: v1.2.0]

Splits reads that contain Ns in their cigar string.

Identifies all N cigar elements and creates k+1 new reads (where k is the number of N cigar elements). The first read includes the bases that are to the left of the first N element, while the part of the read that is to the right of the N (including the Ns) is hard clipped and so on for the rest of the new reads. Used for post-processing RNA reads aligned against the full reference.

Input arguments bam

label
Alignment BAM file

type
`data:alignment:bam`

required
True

disabled
False

hidden
False

ref_seq

label
Reference sequence FASTA file

type
`data:seq:nucleotide`

required
True

disabled
False

hidden
False

advanced.java_gc_threads

label
Java ParallelGCThreads

type
`basic:integer`

description
Sets the number of threads used during parallel phases of the garbage collectors.

required
True

disabled
False

hidden
False

default
2

advanced.max_heap_size

label
Java maximum heap size (Xmx)

type
basic:integer

description
Set the maximum Java heap size (in GB).

required
True

disabled
False

hidden
False

default
12

Output results bam

label
BAM file with reads split at N CIGAR elements

type
basic:file

required
True

disabled
False

hidden
False

bai

label
Index of BAM file

type
basic:file

required
True

disabled
False

hidden
False

stats

label
Alignment statistics

type
`basic:file`

required
True

disabled
False

hidden
False

species

label
Species

type
`basic:string`

required
True

disabled
False

hidden
False

build

label
Build

type
`basic:string`

required
True

disabled
False

hidden
False

GATK VariantFiltration (multi-sample)

`data:variants:vcf:variantfiltration:gatk-variant-filtration` (*data:variants:vcf* **vcf**, *data:seq:nucleotide* **ref_seq**, *list:basic:string* **filter_expressions**, *list:basic:string* **filter_name**, *list:basic:string* **genotype_filter_expressions**, *list:basic:string* **genotype_filter_name**, *data:variants:vcf* **mask**, *basic:string* **mask_name**, *basic:integer* **cluster**, *basic:integer* **window**, *basic:integer* **java_gc_threads**, *basic:integer* **max_heap_size**) [Source: v1.3.0]

Filter multi-sample variant calls based on INFO and/or FORMAT annotations.

This tool is designed for hard-filtering variant calls based on certain criteria. Records are hard-filtered by changing the value in the FILTER field to something other than PASS. Passing variants are annotated as PASS and failing variants are annotated with the name(s) of the filter(s) they failed. If you want to remove failing variants, use GATK SelectVariants process.

Input arguments vcf

label
Input data (VCF)

type
`data:variants:vcf`

required
True

disabled
False

hidden
False

ref_seq

label
Reference sequence

type
`data:seq:nucleotide`

required
True

disabled
False

hidden
False

filter_expressions

label
Expressions used with INFO fields to filter

type
`list:basic:string`

description
VariantFiltration accepts any number of JEXL expressions (so you can have two named filters by using –filter-name One –filter-expression 'X < 1' –filter-name Two –filter-expression 'X > 2'). It is preferable to use multiple expressions, each specifying an individual filter criteria, to a single compound expression that specifies multiple filter criteria. Input expressions one by one and press ENTER after each expression. Examples of filter expression: 'FS > 30', 'DP > 10'.

required
False

disabled
False

hidden
False

filter_name

label
Names to use for the list of filters

type
`list:basic:string`

description
This name is put in the FILTER field for variants that get filtered. Note that there must be a 1-to-1 mapping between filter expressions and filter names. Input expressions one by one and press ENTER after each name. Warning: filter names should be in the same order as filter expressions. Example: you specified filter expressions 'FS > 30' and 'DP > 10', now specify filter names 'FS' and 'DP'.

required
False

disabled
False

hidden
False

genotype_filter_expressions

label
Expressions used with FORMAT field to filter

type
`list:basic:string`

description
Similar to the INFO field based expressions, but used on the FORMAT (genotype) fields instead. VariantFiltration will add the sample-level FT tag to the FORMAT field of filtered samples (this does not affect the record's FILTER tag). One can filter normally based on most fields (e.g. 'GQ < 5.0'), but the GT (genotype) field is an exception. We have put in convenience methods so that one can now filter out hets ('isHet == 1'), refs ('isHomRef == 1'), or homs ('isHomVar == 1'). Also available are expressions isCalled, isNoCall, isMixed, and isAvailable, in accordance with the methods of the Genotype object. To filter by alternative allele depth, use the expression: 'AD.1 < 5'. This filter expression will filter all the samples in the multi-sample VCF file.

required
False

disabled
False

hidden
False

genotype_filter_name

label
Names to use for the list of genotype filters

type
`list:basic:string`

description
Similar to the INFO field based expressions, but used on the FORMAT (genotype) fields instead. Warning: filter names should be in the same order as filter expressions.

required
False

disabled
False

hidden
False

mask

label
Input mask

type
`data:variants:vcf`

description
Any variant which overlaps entries from the provided mask file will be filtered.

required
False

disabled
False

hidden
False

mask_name

label
The text to put in the FILTER field if a 'mask' is provided

type
`basic:string`

description
When using the mask file, the mask name will be annotated in the variant record.

required
False

disabled
!mask

hidden
False

advanced.cluster

label
Cluster size

type
`basic:integer`

description
The number of SNPs which make up a cluster. Must be at least 2.

required
True

disabled
False

hidden
False

default
`3`

advanced.window

label
Window size

type
`basic:integer`

description
The window size (in bases) in which to evaluate clustered SNPs.

required
True

disabled
False

hidden
False

default
`0`

advanced.java_gc_threads

label
Java ParallelGCThreads

type
`basic:integer`

description
Sets the number of threads used during parallel phases of the garbage collectors.

required
True

disabled
False

hidden
False

default
2

advanced.max_heap_size

label
Java maximum heap size (Xmx)

type
`basic:integer`

description
Set the maximum Java heap size (in GB).

required
True

disabled
False

hidden
False

default
12

Output results vcf

label
Filtered variants (VCF)

type
`basic:file`

required
True

disabled
False

hidden
False

tbi

label
Tabix index

type
`basic:file`

required
True

disabled
False

hidden
False

species

label
Species

type
`basic:string`

required
True

disabled
False

hidden
False

build

label
Build

type
`basic:string`

required
True

disabled
False

hidden
False

GATK VariantFiltration (single-sample)

`data:variants:vcf:variantfiltration:single:gatk-variant-filtration-single` (*data:variants:vcf* **vcf**, *data:seq:nucleotide* **ref_seq**, *list:basic:string* **filter_expressions**, *list:basic:string* **filter_name**, *list:basic:string* **genotype_filter_expressions**, *list:basic:string* **genotype_filter_name**, *data:variants:vcf* **mask**, *basic:string* **mask_name**, *basic:integer* **cluster**, *basic:integer* **window**, *basic:integer* **java_gc_threads**, *basic:integer* **max_heap_size**)[Source: v1.3.0]

Filter single-sample variant calls based on INFO and/or FORMAT annotations.

This tool is designed for hard-filtering variant calls based on certain criteria. Records are hard-filtered by changing the value in the FILTER field to something other than PASS. Passing variants are annotated as PASS and failing variants are annotated with the name(s) of the filter(s) they failed. If you want to remove failing variants, use GATK SelectVariants process.

Input arguments vcf

label
Input data (VCF)

type
`data:variants:vcf`

required
True

disabled
False

hidden
False

ref_seq

label
Reference sequence

type
`data:seq:nucleotide`

required
True

disabled
False

hidden
False

filter_expressions

label
Expressions used with INFO fields to filter

type
`list:basic:string`

description
VariantFiltration accepts any number of JEXL expressions (so you can have two named filters by using –filter-name One –filter-expression 'X < 1' –filter-name Two –filter-expression 'X > 2'). It is preferable to use multiple expressions, each specifying an individual filter criteria, to a single compound expression that specifies multiple filter criteria. Input expressions one by one and press ENTER after each expression. Examples of filter expression: 'FS > 30', 'DP > 10'.

required
False

disabled
False

hidden
False

filter_name

label
Names to use for the list of filters

type
`list:basic:string`

description
This name is put in the FILTER field for variants that get filtered. Note that there must be a 1-to-1 mapping between filter expressions and filter names. Input expressions one by one and press ENTER after each name. Warning: filter names should be in the same order as filter expressions. Example: you specified filter expressions 'FS > 30' and 'DP > 10', now specify filter names 'FS' and 'DP'.

required
False

disabled
False

hidden
False

genotype_filter_expressions

label
Expressions used with FORMAT field to filter

type
`list:basic:string`

description
Similar to the INFO field based expressions, but used on the FORMAT (genotype) fields instead. VariantFiltration will add the sample-level FT tag to the FORMAT field of filtered samples (this does not affect the record's FILTER tag). One can filter normally based on most fields (e.g. 'GQ < 5.0'), but the GT (genotype) field is an exception. We have put in convenience methods so that one can now filter out hets ('isHet == 1'), refs ('isHomRef == 1'), or homs ('isHomVar == 1'). Also available are expressions isCalled, isNoCall, isMixed, and isAvailable, in accordance with the methods of the Genotype object. To filter by alternative allele depth, use the expression: 'AD.1 < 5'.

required
False

disabled
False

hidden
False

genotype_filter_name

label
Names to use for the list of genotype filters

type
`list:basic:string`

description
Similar to the INFO field based expressions, but used on the FORMAT (genotype) fields instead. Warning: filter names should be in the same order as filter expressions.

required
False

disabled
False

hidden
False

mask

label
Input mask

type
`data:variants:vcf`

description
Any variant which overlaps entries from the provided mask file will be filtered.

required
False

disabled
False

hidden
False

mask_name

label
The text to put in the FILTER field if a 'mask' is provided

type
`basic:string`

description
When using the mask file, the mask name will be annotated in the variant record.

required
False

disabled
!mask

hidden
False

advanced.cluster

label
Cluster size

type
`basic:integer`

description
The number of SNPs which make up a cluster. Must be at least 2.

required
True

disabled
False

hidden
False

default
3

advanced.window

label
Window size

type
basic:integer

description
The window size (in bases) in which to evaluate clustered SNPs.

required
True

disabled
False

hidden
False

default
0

advanced.java_gc_threads

label
Java ParallelGCThreads

type
basic:integer

description
Sets the number of threads used during parallel phases of the garbage collectors.

required
True

disabled
False

hidden
False

default
2

advanced.max_heap_size

label
Java maximum heap size (Xmx)

type
basic:integer

description
Set the maximum Java heap size (in GB).

required
True

disabled
False

hidden
False

default
12

Output results vcf

label
Filtered variants (VCF)

type
basic:file

required
True

disabled
False

hidden
False

tbi

label
Tabix index

type
basic:file

required
True

disabled
False

hidden
False

species

label
Species

type
basic:string

required
True

disabled
False

hidden
False

build

label
Build

type
basic:string

required
True

disabled
False

hidden
False

GATK VariantsToTable

`data:variantstable:variants-to-table` (*data:variants:vcf* **vcf**, *list:basic:string* **vcf_fields**, *list:basic:string* **gf_fields**, *basic:boolean* **split_alleles**) [Source: v1.2.0]

Run GATK VariantsToTable.

This tool extracts specified fields for each variant in a VCF file to a tab-delimited table, which may be easier to work with than a VCF. For additional information, please see [manual page](https://gatk.broadinstitute.org/hc/en-us/articles/360036711531-VariantsToTable)

Input arguments vcf

label
Input VCF file

type
`data:variants:vcf`

required
True

disabled
False

hidden
False

vcf_fields

label
Select VCF fields

type
`list:basic:string`

description
The name of a standard VCF field or an INFO field to include in the output table. The field can be any standard VCF column (e.g. CHROM, ID, QUAL) or any annotation name in the INFO field (e.g. AC, AF).

required
True

disabled
False

hidden
False

default
`['CHROM', 'POS', 'ID', 'REF', 'ALT']`

advanced_options.gf_fields

label
Include FORMAT/sample-level fields

type
`list:basic:string`

required
True

disabled
False

hidden
False

default
`['GT', 'GQ']`

advanced_options.split_alleles

label
Split multi-allelic records into multiple lines

type
`basic:boolean`

description
By default, a variant record with multiple ALT alleles will be summarized in one line, with per alt-allele fields (e.g. allele depth) separated by commas.This may cause difficulty when the table is loaded by an R script, for example. Use this flag to write multi-allelic records on separate lines of output.

required
True

disabled
False

hidden
False

default
`True`

Output results tsv

label
Tab-delimited file with variants

type
`basic:file`

required
True

disabled
False

hidden
False

species

label
Species

type
`basic:string`

required
True

disabled
False

hidden
False

build

label
Build

type
`basic:string`

required
True

disabled
False

hidden
False

GATK filter variants (VQSR)

`data:variants:vcf:vqsr:gatk-vqsr` (*data:variants:vcf* **vcf**, *data:variants:vcf* **dbsnp**, *data:variants:vcf* **mills**, *data:variants:vcf* **axiom_poly**, *data:variants:vcf* **hapmap**, *data:variants:vcf* **omni**, *data:variants:vcf* **thousand_genomes**, *basic:boolean* **use_as_anno**, *list:basic:string* **indel_anno_fields**, *list:basic:string* **snp_anno_fields**, *basic:decimal* **indel_filter_level**, *basic:decimal* **snp_filter_level**, *basic:integer* **max_gaussians_indels**, *basic:integer* **max_gaussians_snps**)[Source: v1.2.0]

Filter WGS variants using Variant Quality Score Recalibration (VQSR) procedure.

Input arguments vcf

label
Input data (VCF)

type
`data:variants:vcf`

required
True

disabled
False

hidden
False

resource_files.dbsnp

label
dbSNP file

type
`data:variants:vcf`

required
True

disabled
False

hidden
False

resource_files.mills

label
Mills and 1000G gold standard indels

type
`data:variants:vcf`

required
False

disabled
False

hidden
False

resource_files.axiom_poly

label
1000G Axiom genotype data

type
`data:variants:vcf`

required
False

disabled
False

hidden
False

resource_files.hapmap

label
HapMap variants

type
`data:variants:vcf`

required
False

disabled
False

hidden
False

resource_files.omni

label
1000G Omni variants

type
`data:variants:vcf`

required
False

disabled
False

hidden
False

resource_files.thousand_genomes

label
1000G high confidence SNPs

type
`data:variants:vcf`

required
False

disabled
False

hidden
False

advanced_options.use_as_anno

label
–use-allele-specific-annotations

type
`basic:boolean`

required
True

disabled
False

hidden
False

default
`False`

advanced_options.indel_anno_fields

label
Annotation fields (INDEL filtering)

type
`list:basic:string`

required
True

disabled
False

hidden
False

default
`['FS', 'ReadPosRankSum', 'MQRankSum', 'QD', 'SOR', 'DP']`

advanced_options.snp_anno_fields

label
Annotation fields (SNP filtering)

type
`list:basic:string`

required
True

disabled
False

hidden
False

default
`['QD', 'MQRankSum', 'ReadPosRankSum', 'FS', 'MQ', 'SOR', 'DP']`

advanced_options.indel_filter_level

label
–truth-sensitivity-filter-level (INDELs)

type
`basic:decimal`

required
True

disabled
False

hidden
False

default
`99.0`

advanced_options.snp_filter_level

label
–truth-sensitivity-filter-level (SNPs)

type
`basic:decimal`

required
True

disabled
False

hidden
False

default
`99.7`

advanced_options.max_gaussians_indels

label
–max-gaussians (INDELs)

type
`basic:integer`

description
This parameter determines the maximum number of Gaussians that should be used when building a positive model using the variational Bayes algorithm. This parameter sets the expected number of clusters in modeling. If a dataset gives fewer distinct clusters, e.g. as can happen for smaller data, then the tool will tell you there is insufficient data with a No data found error message. In this case, try decrementing the –max-gaussians value.

required
True

disabled
False

hidden
False

default
`4`

advanced_options.max_gaussians_snps

label
–max-gaussians (SNPs)

type
`basic:integer`

description
This parameter determines the maximum number of Gaussians that should be used when building a positive model using the variational Bayes algorithm. This parameter sets the expected number of clusters in modeling. If a dataset gives fewer distinct clusters, e.g. as can happen for smaller data, then the tool will tell you there is insufficient data with a No data found error message. In this case, try decrementing the –max-gaussians value.

required
True

disabled
False

hidden
False

default
`6`

Output results vcf

label
GVCF file

type
`basic:file`

required
True

disabled
False

hidden
False

tbi

label
Tabix index

type
`basic:file`

required
True

disabled
False

hidden
False

species

label
Species

type
`basic:string`

required
True

disabled
False

hidden
False

build

label
Build

type
`basic:string`

required
True

disabled
False

hidden
False

GATK refine variants

data:variants:vcf:refinevariants:gatk-refine-variants (*data:variants:vcf* **vcf**, *data:seq:nucleotide* **ref_seq**, *data:variants:vcf* **vcf_pop**) [Source: v1.1.1]

Run GATK Genotype Refinement.

The goal of the Genotype Refinement workflow is to use additional data to improve the accuracy of genotype calls and to filter genotype calls that are not reliable enough for downstream analysis. In this sense it serves as an optional extension of the variant calling workflow, intended for researchers whose work requires high-quality identification of individual genotypes. For additional information, please see [manual page](https://gatk.broadinstitute.org/hc/en-us/articles/360035531432-Genotype-Refinement-workflow-for-germline-short-variants)

Input arguments vcf

label
The main input, as produced in the GATK VQSR process

type
`data:variants:vcf`

required
True

disabled
False

hidden
False

ref_seq

label
Reference sequence

type
`data:seq:nucleotide`

required
True

disabled
False

hidden
False

vcf_pop

label
Population-level variant set (VCF)

type
`data:variants:vcf`

required
False

disabled
False

hidden
False

Output results vcf

label
Refined multi-sample vcf

type
`basic:file`

required
True

disabled
False

hidden
False

tbi

label
Tabix index

type
`basic:file`

required
True

disabled
False

hidden
False

species

label
Species

type
`basic:string`

required
True

disabled
False

hidden
False

build

label
Build

type
`basic:string`

required
True

disabled
False

hidden
False

GATK4 (HaplotypeCaller)

`data:variants:vcf:gatk:hc:vc-gatk4-hc` (*data:alignment:bam* **alignment**, *data:seq:nucleotide* **genome**, *data:bed* **intervals_bed**, *data:variants:vcf* **dbsnp**, *basic:integer* **stand_call_conf**, *basic:integer* **mbq**, *basic:integer* **max_reads**, *basic:integer* **interval_padding**, *basic:boolean* **soft_clipped**, *basic:integer* **java_gc_threads**, *basic:integer* **max_heap_size**) [Source: v1.5.0]

GATK HaplotypeCaller Variant Calling.

Call germline SNPs and indels via local re-assembly of haplotypes.

The HaplotypeCaller is capable of calling SNPs and indels simultaneously via local de-novo assembly of haplotypes in an active region. In other words, whenever the program encounters a region showing signs of variation, it discards the existing mapping information and completely reassembles the reads in that region. This allows the HaplotypeCaller to be more accurate when calling regions that are traditionally difficult to call, for example when they contain different types of variants close to each other. It also makes the HaplotypeCaller much better at calling indels than position-based callers like UnifiedGenotyper.

Input arguments alignment

label
Analysis ready BAM file

type
`data:alignment:bam`

required
True

disabled
False

hidden
False

genome

label
Reference genome

type
`data:seq:nucleotide`

required
True

disabled
False

hidden
False

intervals_bed

label
Intervals (from BED file)

type
`data:bed`

description
Use this option to perform the analysis over only part of the genome.

required
False

disabled
False

hidden
False

dbsnp

label
dbSNP file

type
`data:variants:vcf`

description
Database of known polymorphic sites.

required
True

disabled
False

hidden
False

stand_call_conf

label
Min call confidence threshold

type
`basic:integer`

description
The minimum phred-scaled confidence threshold at which variants should be called.

required
True

disabled
False

hidden
False

default
`30`

mbq

label
Min Base Quality

type
`basic:integer`

description
Minimum base quality required to consider a base for calling.

required
True

disabled
False

hidden
False

default
`20`

max_reads

label
Max reads per aligment start site

type
`basic:integer`

description
Maximum number of reads to retain per alignment start position. Reads above this threshold will be downsampled. Set to 0 to disable.

required
True

disabled
False

hidden
False

default
`50`

advanced.interval_padding

label
Interval padding

type
`basic:integer`

description
Amount of padding (in bp) to add to each interval you are including. The recommended value is 100.

required
False

disabled
False

hidden
!intervals_bed

advanced.soft_clipped

label
Do not analyze soft clipped bases in the reads

type
`basic:boolean`

description
Suitable option for RNA-seq variant calling.

required
True

disabled
False

hidden
False

default
`False`

advanced.java_gc_threads

label
Java ParallelGCThreads

type
`basic:integer`

description
Sets the number of threads used during parallel phases of the garbage collectors.

required
True

disabled
False

hidden
False

default
`2`

advanced.max_heap_size

label
Java maximum heap size (Xmx)

type
`basic:integer`

description
Set the maximum Java heap size (in GB).

required
True

disabled
False

hidden
False

default
`12`

Output results vcf

label
VCF file

type
`basic:file`

required
True

disabled
False

hidden
False

tbi

label
Tabix index

type
`basic:file`

required
True

disabled
False

hidden
False

species

label
Species

type
`basic:string`

required
True

disabled
False

hidden
False

build

label
Build

type
`basic:string`

required
True

disabled
False

hidden
False

GEO import

`data:geo:geo-import` (*basic:string* **gse_accession**, *basic:boolean* **prefetch**, *basic:string* **max_size_prefetch**, *basic:integer* **min_spot_id**, *basic:integer* **max_spot_id**, *basic:integer* **min_read_len**, *basic:boolean* **clip**, *basic:boolean* **aligned**, *basic:boolean* **unaligned**, *basic:file* **mapping_file**, *basic:string* **source**, *basic:string* **build**) [Source: v2.7.2]

Import all runs from a GEO Series.

WARNING: Additional costs for storage and processing may be incurred if a very large data set is selected.

RNA-seq ChIP-Seq, ATAC-Seq and expression microarray datasets can be uploaded.

For RNA-Seq data sets this runs the SRA import process for each experiment (SRX) from the selected RNA-Seq GEO Series. The same procedure is followed for ChIP-Seq and ATAC-Seq data sets.

If GSE contains microarray data, it downloads individual samples and uploads them as microarray expression objects. Probe IDs can be mapped to the Ensembl IDs if the corresponding GPL platform is supported, otherwise, a custom mapping file should be provided. Currently supported platforms are: GPL74, GPL201, GPL96, GPL571, GPL97, GPL570, GPL91, GPL8300, GPL92, GPL93, GPL94, GPL95, GPL17586, GPL5175, GPL80, GPL6244, GPL16686, GPL15207, GPL1352, GPL11068, GPL26966, GPL6848, GPL14550, GPL17077, GPL16981, GPL13497, GPL6947, GPL10558, GPL6883, GPL13376,GPL6884, GPL6254.

In addition metadata table with sample information is created and uploaded to the same collection.

Input arguments gse_accession

label
GEO accession

type
`basic:string`

description
Enter a GEO series accession number.

required
True

disabled
False

hidden
False

advanced.prefetch

label
Prefetch SRA file

type
`basic:boolean`

required
True

disabled
False

hidden
False

default
`True`

advanced.max_size_prefetch

label
Maximum file size to download in KB

type
`basic:string`

description
A unit prefix can be used instead of a value in KB (e.g. 1024M or 1G).

required
True

disabled
False

hidden
False

default
`20G`

advanced.min_spot_id

label
Minimum spot ID

type
`basic:integer`

required
False

disabled
False

hidden
False

advanced.max_spot_id

label
Maximum spot ID

type
`basic:integer`

required
False

disabled
False

hidden
False

advanced.min_read_len

label
Minimum read length

type
`basic:integer`

required
False

disabled
False

hidden
False

advanced.clip

label
Clip adapter sequences

type
`basic:boolean`

required
True

disabled
False

hidden
False

default
`False`

advanced.aligned

label
Dump only aligned sequences

type
`basic:boolean`

required
True

disabled
False

hidden
False

default
`False`

advanced.unaligned

label
Dump only unaligned sequences

type
`basic:boolean`

required
True

disabled
False

hidden
False

default
`False`

advanced.mapping_file

label
File with probe ID mappings

type
`basic:file`

description
The file should be tab-separated and contain two columns with their column names. The first column should contain Gene IDs and the second one should contain probe names. Supported file extensions are .tab.*, .tsv.*, .txt.*

required
False

disabled
False

hidden
False

advanced.source

label
Gene ID source

type
`basic:string`

description
Gene ID source used for probe mapping is required when using a custom file.

required
False

disabled
False

hidden
False

choices

- AFFY: `AFFY`
- DICTYBASE: `DICTYBASE`
- ENSEMBL: `ENSEMBL`
- NCBI: `NCBI`
- UCSC: `UCSC`

advanced.build

label
Genome build

type
`basic:string`

description
Genome build of mapping file is required when using a custom file.

required
False

disabled
False

hidden
False

Output results

GFF3 file

`data:annotation:gff3upload-gff3` (*basic:file* **src**, *basic:string* **source**, *basic:string* **species**, *basic:string* **build**) [Source: v3.5.0]

Import a General Feature Format (GFF) file which is a file format used for describing genes and other features of DNA, RNA and protein sequences. See [here](https://useast.ensembl.org/info/website/upload/gff3.html) and [here](https://en.wikipedia.org/wiki/General_feature_format) for more information.

Input arguments src

label
Annotation (GFF3)

type
`basic:file`

description
Annotation in GFF3 format. Supported extensions are: .gff, .gff3 and .gtf

validate_regex
`\.(gff|gff3|gtf)(|\.gz|\.bz2|\.tgz|\.tar\.gz|\.tar\.bz2|\.zip|\.rar|\.7z)$`

source

label
Gene ID database

type
`basic:string`

choices

- DICTYBASE: `DICTYBASE`
- ENSEMBL: `ENSEMBL`
- NCBI: `NCBI`
- UCSC: `UCSC`

species

label
Species

type
`basic:string`

description
Species latin name.

choices

- Homo sapiens: `Homo sapiens`
- Mus musculus: `Mus musculus`
- Rattus norvegicus: `Rattus norvegicus`
- Dictyostelium discoideum: `Dictyostelium discoideum`

build

label
Build

type
`basic:string`

Output results annot

label
Uploaded GFF3 file

type
`basic:file`

annot_sorted

label
Sorted GFF3 file

type
`basic:file`

annot_sorted_idx_igv

label
IGV index for sorted GFF3

type
`basic:file`

annot_sorted_track_jbrowse

label
Jbrowse track for sorted GFF3

type
`basic:file`

source

label
Gene ID database

type
`basic:string`

species

label
Species

type
basic:string

build

label
Build

type
basic:string

GTF file

data:annotation:gtfupload-gtf (*basic:file* **src**, *basic:string* **source**, *basic:string* **species**, *basic:string* **build**) [Source: v3.5.0]

Import a Gene Transfer Format (GTF) file. It is a file format used to hold information about gene structure. It is a tab-delimited text format based on the general feature format (GFF), but contains some additional conventions specific to gene information. See [here](https://en.wikipedia.org/wiki/General_feature_format) for differences between GFF and GTF files.

Input arguments src

label
Annotation (GTF)

type
basic:file

description
Annotation in GTF format.

validate_regex
\.(gtf|gff)(|\.gz|\.bz2|\.tgz|\.tar\.gz|\.tar\.bz2|\.zip|\.rar|\.7z)$

source

label
Gene ID database

type
basic:string

choices

- DICTYBASE: DICTYBASE
- ENSEMBL: ENSEMBL
- NCBI: NCBI
- UCSC: UCSC

species

label
Species

type
basic:string

description
Species latin name.

choices

- Homo sapiens: `Homo sapiens`
- Mus musculus: `Mus musculus`
- Rattus norvegicus: `Rattus norvegicus`
- Dictyostelium discoideum: `Dictyostelium discoideum`

build

label
Build

type
`basic:string`

Output results annot

label
Uploaded GTF file

type
`basic:file`

annot_sorted

label
Sorted GTF file

type
`basic:file`

annot_sorted_idx_igv

label
IGV index for sorted GTF file

type
`basic:file`

required
False

annot_sorted_track_jbrowse

label
Jbrowse track for sorted GTF

type
`basic:file`

required
False

source

label
Gene ID database

type
`basic:string`

species

label
Species

type
`basic:string`

build

label
Build

type
`basic:string`

Gene set

`data:geneset:upload-geneset` (*basic:file* **src**, *basic:string* **source**, *basic:string* **species**) [Source: v1.3.2]

Upload a set of genes.

Provide one gene ID per line in a .tab, .tab.gz, or .txt file format.

Input arguments src

label
Gene set

type
`basic:file`

description
List of genes (.tab/.txt extension), one gene ID per line.

required
True

disabled
False

hidden
False

source

label
Gene ID source

type
`basic:string`

required
True

disabled
False

hidden
False

choices

- AFFY: `AFFY`

- DICTYBASE: `DICTYBASE`
- ENSEMBL: `ENSEMBL`
- NCBI: `NCBI`
- UCSC: `UCSC`

species

label
Species

type
`basic:string`

description
Species latin name.

required
True

disabled
False

hidden
False

choices

- Homo sapiens: `Homo sapiens`
- Mus musculus: `Mus musculus`
- Rattus norvegicus: `Rattus norvegicus`
- Dictyostelium discoideum: `Dictyostelium discoideum`
- Odocoileus virginianus texanus: `Odocoileus virginianus texanus`
- Solanum tuberosum: `Solanum tuberosum`

Output results geneset

label
Gene set

type
`basic:file`

required
True

disabled
False

hidden
False

geneset_json

label
Gene set (JSON)

type
`basic:json`

required
True

disabled
False

hidden
False

source

label
Gene ID source

type
`basic:string`

required
True

disabled
False

hidden
False

species

label
Species

type
`basic:string`

required
True

disabled
False

hidden
False

Gene set (create from Venn diagram)

`data:geneset:venn:create-geneset-venn` (*list:basic:string* **genes**, *basic:string* **source**, *basic:string* **species**, *basic:file* **venn**) [Source: v1.3.2]

Create a gene set from a Venn diagram.

Input arguments genes

label
Genes

type
`list:basic:string`

description
List of genes.

required
True

disabled
False

hidden
False

source

label
Gene ID source

type
`basic:string`

required
True

disabled
False

hidden
False

choices

- AFFY: `AFFY`
- DICTYBASE: `DICTYBASE`
- ENSEMBL: `ENSEMBL`
- NCBI: `NCBI`
- UCSC: `UCSC`

species

label
Species

type
`basic:string`

description
Species latin name.

required
True

disabled
False

hidden
False

choices

- Homo sapiens: `Homo sapiens`
- Mus musculus: `Mus musculus`
- Rattus norvegicus: `Rattus norvegicus`
- Dictyostelium discoideum: `Dictyostelium discoideum`
- Odocoileus virginianus texanus: `Odocoileus virginianus texanus`

- Solanum tuberosum: `Solanum tuberosum`

venn

label
Venn diagram

type
`basic:file`

description
JSON file of Venn diagram.

required
True

disabled
False

hidden
False

Output results geneset

label
Gene set

type
`basic:file`

required
True

disabled
False

hidden
False

geneset_json

label
Gene set (JSON)

type
`basic:json`

required
True

disabled
False

hidden
False

source

label
Gene ID source

type
`basic:string`

required
True

disabled
False

hidden
False

species

label
Species

type
`basic:string`

required
True

disabled
False

hidden
False

venn

label
Venn diagram

type
`basic:json`

required
True

disabled
False

hidden
False

Gene set (create)

`data:geneset:create-geneset` (*list:basic:string* **genes**, *basic:string* **source**, *basic:string* **species**) [Source: v1.3.2]

Create a gene set from a list of genes.

Input arguments genes

label
Genes

type
`list:basic:string`

description
List of genes.

required
True

disabled
False

hidden
False

source

label
Gene ID source

type
`basic:string`

required
True

disabled
False

hidden
False

choices

- AFFY: `AFFY`
- DICTYBASE: `DICTYBASE`
- ENSEMBL: `ENSEMBL`
- NCBI: `NCBI`
- UCSC: `UCSC`

species

label
Species

type
`basic:string`

description
Species latin name.

required
True

disabled
False

hidden
False

choices

- Homo sapiens: `Homo sapiens`
- Mus musculus: `Mus musculus`
- Rattus norvegicus: `Rattus norvegicus`
- Dictyostelium discoideum: `Dictyostelium discoideum`
- Odocoileus virginianus texanus: `Odocoileus virginianus texanus`

- Solanum tuberosum: `Solanum tuberosum`

Output results geneset

label
Gene set

type
`basic:file`

required
True

disabled
False

hidden
False

geneset_json

label
Gene set (JSON)

type
`basic:json`

required
True

disabled
False

hidden
False

source

label
Gene ID source

type
`basic:string`

required
True

disabled
False

hidden
False

species

label
Species

type
`basic:string`

required
True

disabled
False

hidden
False

HISAT2

`data:alignment:bam:hisat2alignment-hisat2` (*data:index:hisat2* **genome**, *data:reads:fastq* **reads**, *basic:boolean* **softclip**, *basic:integer* **noncansplice**, *basic:boolean* **cufflinks**) [Source: v2.6.1]

HISAT2 is a fast and sensitive alignment program for mapping next-generation sequencing reads (both DNA and RNA) to a population of genomes (as well as to a single reference genome). See [here](https://ccb.jhu.edu/software/hisat2/index.shtml) for more information.

Input arguments genome

label
Reference genome

type
`data:index:hisat2`

reads

label
Reads

type
`data:reads:fastq`

softclip

label
Disallow soft clipping

type
`basic:boolean`

default
`False`

spliced_alignments.noncansplice

label
Non-canonical splice sites penalty (optional)

type
`basic:integer`

description
Sets the penalty for each pair of non-canonical splice sites (e.g. non-GT/AG).

required
False

spliced_alignments.cufflinks

label
Report alignments tailored specifically for Cufflinks

type
basic:boolean

description
With this option, HISAT2 looks for novel splice sites with three signals (GT/AG, GC/AG, AT/AC), but all user-provided splice sites are used irrespective of their signals. HISAT2 produces an optional field, XS:A:[+-], for every spliced alignment.

default
False

Output results bam

label
Alignment file

type
basic:file

description
Position sorted alignment

bai

label
Index BAI

type
basic:file

stats

label
Statistics

type
basic:file

splice_junctions

label
Splice junctions

type
basic:file

unmapped_f

label
Unmapped reads (mate 1)

type
basic:file

required
False

unmapped_r

label
Unmapped reads (mate 2)

type
basic:file

required
False

species

label
Species

type
`basic:string`

build

label
Build

type
`basic:string`

HISAT2 genome index

`data:index:hisat2:hisat2-index` (*data:seq:nucleotide* **ref_seq**) [Source: v1.2.1]

Create HISAT2 genome index.

Input arguments ref_seq

label
Reference sequence (nucleotide FASTA)

type
`data:seq:nucleotide`

required
True

disabled
False

hidden
False

Output results index

label
HISAT2 index

type
`basic:dir`

required
True

disabled
False

hidden
False

fastagz

label
FASTA file (compressed)

type
`basic:file`

required
True

disabled
False

hidden
False

fasta

label
FASTA file

type
`basic:file`

required
True

disabled
False

hidden
False

fai

label
FASTA file index

type
`basic:file`

required
True

disabled
False

hidden
False

species

label
Species

type
`basic:string`

required
True

disabled
False

hidden
False

build

label
Build

type
`basic:string`

required
True

disabled
False

hidden
False

HMR

`data:wgbs:hmrhmr` (*data:wgbs:methcounts* **methcounts**) [Source: v1.4.0]

Identify hypo-methylated regions.

Input arguments methcounts

label
Methylation levels

type
`data:wgbs:methcounts`

description
Methylation levels data calculated using methcounts.

Output results hmr

label
Hypo-methylated regions

type
`basic:file`

tbi_jbrowse

label
Bed file index for Jbrowse

type
`basic:file`

species

label
Species

type
`basic:string`

build

label
Build

type
`basic:string`

Hierarchical clustering of time courses

`data:clustering:hierarchical:etc:clustering-hierarchical-etc` (*list:data:expression* **expressions**, *list:basic:string* **genes**, *basic:string* **gene_species**, *basic:string* **gene_source**, *basic:string* **distance**, *basic:string* **linkage**, *basic:boolean* **ordering**)[Source: v1.3.1]

Cluster gene expression time courses.

Hierarchical clustering of expression time courses.

Input arguments expressions

label
Time series relation

type
`list:data:expression`

description
Select time course to which the expressions belong to.

required
True

disabled
False

hidden
False

genes

label
Gene subset

type
`list:basic:string`

description
Select at least two genes or leave this field empty.

required
False

disabled
False

hidden
False

gene_species

label
Species

type
`basic:string`

description
Species to which the selected genes belong to. This field is required if gene subset is set.

required
False

disabled
False

hidden
!genes

choices

- Dictyostelium discoideum: `Dictyostelium discoideum`
- Homo sapiens: `Homo sapiens`
- Macaca mulatta: `Macaca mulatta`
- Mus musculus: `Mus musculus`
- Rattus norvegicus: `Rattus norvegicus`

gene_source

label
Gene ID database of selected genes

type
`basic:string`

description
This field is required if gene subset is set.

required
False

disabled
False

hidden
!genes

distance

label
Distance metric

type
`basic:string`

required
True

disabled
False

hidden
False

default
`spearman`

choices

- Euclidean: `euclidean`
- Spearman: `spearman`
- Pearson: `pearson`

linkage

label
Linkage method

type
`basic:string`

required
True

disabled
False

hidden
False

default
`average`

choices

- single: `single`
- average: `average`
- complete: `complete`

ordering

label
Use optimal ordering

type
`basic:boolean`

description
Results in a more intuitive tree structure, but may slow down the clustering on large datasets

required
True

disabled
False

hidden
False

default
`False`

Output results cluster

label
Hiearархical clustering

type
`basic:json`

required
True

disabled
False

hidden
False

source

label
Gene ID database

type
`basic:string`

required
True

disabled
False

hidden
False

species

label
Species

type
`basic:string`

required
True

disabled
False

hidden
False

build

label
Build

type
`basic:string`

required
True

disabled
False

hidden
False

feature_type

label
Feature type

type
`basic:string`

required
True

disabled
False

hidden
False

IDAT file

data:methylationarray:idat:upload-idat (*basic:file* **red_channel**, *basic:file* **green_channel**, *basic:string* **species**, *basic:string* **platform**) [Source: v1.1.1]

Upload Illumina methylation array raw IDAT data.

This import process accepts Illumina methylation array BeadChip raw files in IDAT format. Two input files, one for each of the Green and Red signal channels, are expected. The uploads of human (HM27, HM450, EPIC) and mouse (MM285) array types are supported.

Input arguments red_channel

label
Red channel IDAT file (*_Red.idat)

type
`basic:file`

required
True

disabled
False

hidden
False

green_channel

label
Green channel IDAT file (*_Grn.idat)

type
`basic:file`

required
True

disabled
False

hidden
False

species

label
Species

type
`basic:string`

description
Select a species name from the dropdown menu.

required
True

disabled
False

hidden
False

default
`Homo sapiens`

choices

- Homo sapiens: `Homo sapiens`
- Mus musculus: `Mus musculus`

platform

label
Protein ID database source

type
`basic:string`

description
Select a methylation array platform for human (HM450, HM27, EPIC) or mouse (MM285) samples.

required
True

disabled
False

hidden
False

default
`HM450`

choices

- HM450: `HM450`
- HM27: `HM27`
- EPIC: `EPIC`
- MM285: `MM285`

Output results red_channel

label
Red channel IDAT file

type
`basic:file`

required
True

disabled
False

hidden
False

green_channel

label
Green channel IDAT file

type
`basic:file`

required
True

disabled
False

hidden
False

species

label
Species

type
`basic:string`

required
True

disabled
False

hidden
False

platform

label
Platform

type
`basic:string`

required
True

disabled
False

hidden
False

MACS 1.4

data:chipseq:callpeak:macs14macs14 (*data:alignment:bam* **treatment**, *data:alignment:bam* **control**, *basic:string* **pvalue**) [Source: v3.5.1]

Model-based Analysis of ChIP-Seq (MACS 1.4) empirically models the length of the sequenced ChIP fragments, which tends to be shorter than sonication or library construction size estimates, and uses it to improve the spatial resolution of predicted binding sites. MACS also uses a dynamic Poisson distribution to effectively capture local biases in the genome sequence, allowing for more sensitive and robust prediction. See the [original paper](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2592715/) for more information.

Input arguments treatment

label
BAM File

type
`data:alignment:bam`

control

label
BAM Background File

type
`data:alignment:bam`

required
False

pvalue

label
P-value

type
`basic:string`

default
`1e-9`

choices

- 1e-9: `1e-9`
- 1e-6: `1e-6`

Output results peaks_bed

label
Peaks (BED)

type
`basic:file`

summits_bed

label
Summits (BED)

type
`basic:file`

peaks_xls

label
Peaks (XLS)

type
`basic:file`

wiggle

label
Wiggle

type
`basic:file`

control_bigwig

label
Control (bigWig)

type
`basic:file`

required
False

treat_bigwig

label
Treat (bigWig)

type
`basic:file`

peaks_bigbed_igv_ucsc

label
Peaks (bigBed)

type
`basic:file`

required
False

summits_bigbed_igv_ucsc

label
Summits (bigBed)

type
`basic:file`

required
False

peaks_tbi_jbrowse

label
JBrowse track peaks file

type
`basic:file`

summits_tbi_jbrowse

label
JBrowse track summits file

type
`basic:file`

model

label
Model

type
`basic:file`

required
False

neg_peaks

label
Negative peaks (XLS)

type
`basic:file`

required
False

species

label
Species

type
`basic:string`

build

label
Build

type
`basic:string`

MACS 2.0

`data:chipseq:callpeak:macs2:macs2-callpeak` (*data:alignment:bam* **case**, *data:alignment:bam* **control**, *data:bed* **promoter**, *basic:boolean* **tagalign**, *basic:integer* **q_threshold**, *basic:integer* **n_sub**, *basic:boolean* **tn5**, *basic:integer* **shift**, *basic:string* **format**, *basic:string* **duplicates**, *basic:string* **duplicates_prepeak**, *basic:decimal* **qvalue**, *basic:decimal* **pvalue**, *basic:decimal* **pvalue_prepeak**, *basic:integer* **cap_num**, *basic:integer* **mfold_lower**, *basic:integer* **mfold_upper**, *basic:integer* **slocal**, *basic:integer* **llocal**, *basic:integer* **extsize**, *basic:integer* **shift**, *basic:integer* **band_width**, *basic:boolean* **nolambda**, *basic:boolean* **fix_bimodal**, *basic:boolean* **nomodel**, *basic:boolean* **nomodel_prepeak**, *basic:boolean* **down_sample**, *basic:boolean* **bedgraph**, *basic:boolean* **spmr**, *basic:boolean* **call_summits**, *basic:boolean* **broad**, *basic:decimal* **broad_cutoff**) [Source: v4.8.1]

Call ChIP-Seq peaks with MACS 2.0.

Model-based Analysis of ChIP-Seq (MACS 2.0), is used to identify transcript factor binding sites. MACS 2.0 captures the influence of genome complexity to evaluate the significance of enriched ChIP regions, and MACS improves the spatial resolution of binding sites through combining the information of both sequencing tag position and orientation. It has also an option to link nearby peaks together in order to call broad peaks. See [here](https://github.com/taoliu/MACS/) for more information.

In addition to peak-calling, this process computes ChIP-Seq and ATAC-Seq QC metrics. Process returns a QC metrics report, fragment length estimation, and a deduplicated tagAlign file. QC report contains ENCODE 3 proposed QC metrics – [NRF](https://www.encodeproject.org/data-standards/terms/), [PBC bottlenecking coefficients, NSC, and RSC](https://genome.ucsc.edu/ENCODE/qualityMetrics.html#chipSeq).

Input arguments case

label
Case (treatment)

type
`data:alignment:bam`

required
True

disabled
False

hidden
False

control

label
Control (background)

type
`data:alignment:bam`

required
False

disabled
False

hidden
False

promoter

label
Promoter regions BED file

type
`data:bed`

description
BED file containing promoter regions (TSS+-1000bp for example). Needed to get the number of peaks and reads mapped to promoter regions.

required
False

disabled
False

hidden
False

tagalign

label
Use tagAlign files

type
`basic:boolean`

description
Use filtered tagAlign files as case (treatment) and control (background) samples. If extsize parameter is not set, run MACS using input's estimated fragment length.

required
True

disabled
False

hidden
False

default
`False`

prepeakqc_settings.q_threshold

label
Quality filtering threshold

type
`basic:integer`

required
True

disabled
False

hidden
False

default
`30`

prepeakqc_settings.n_sub

label
Number of reads to subsample

type
`basic:integer`

required
True

disabled
False

hidden
False

default
`15000000`

prepeakqc_settings.tn5

label
Tn5 shifting

type
`basic:boolean`

description
Tn5 transposon shifting. Shift reads on '+' strand by 4bp and reads on '-' strand by 5bp.

required
True

disabled
False

hidden
False

default
`False`

prepeakqc_settings.shift

label
User-defined cross-correlation peak strandshift

type
`basic:integer`

description
If defined, SPP tool will not try to estimate fragment length but will use the given value as fragment length.

required
False

disabled
False

hidden
False

settings.format

label
Format of tag file

type
`basic:string`

description
This specifies the format of input files. For paired-end data the format dictates how MACS2 will treat mates. If the selected format is BAM, MACS2 will only keep the left mate (5' end) tag. However, when format BAMPE is selected, MACS2 will use actual insert sizes of pairs of reads to build fragment pileup, instead of building bimodal distribution plus and minus strand reads to predict fragment size.

required
True

disabled
False

hidden
tagalign

default
`BAM`

choices

- BAM: `BAM`
- BAMPE: `BAMPE`

settings.duplicates

label
Number of duplicates

type
`basic:string`

description
It controls the MACS behavior towards duplicate tags at the exact same location – the same coordination and the same strand. The 'auto' option makes MACS calculate the maximum tags at the exact same location based on binomial distribution using 1e-5 as pvalue cutoff and the 'all' option keeps all the tags. If an integer is given, at most this number of tags will be kept at the same location. The default is to keep one tag at the same location.

required
False

disabled
False

hidden
tagalign

choices

- 1: `1`
- auto: `auto`
- all: `all`

settings.duplicates_prepeak

label
Number of duplicates

type
`basic:string`

description
It controls the MACS behavior towards duplicate tags at the exact same location – the same coordination and the same strand. The 'auto' option makes MACS calculate the maximum tags at the exact same location based on binomial distribution using 1e-5 as pvalue cutoff and the 'all' option keeps all the tags. If an integer is given, at most this number of tags will be kept at the same location. The default is to keep one tag at the same location.

required
True

disabled
False

hidden
!tagalign

default
`all`

choices

- 1: `1`
- auto: `auto`
- all: `all`

settings.qvalue

label
Q-value cutoff

type
`basic:decimal`

description
The q-value (minimum FDR) cutoff to call significant regions. Q-values are calculated from p-values using Benjamini-Hochberg procedure.

required
False

disabled
settings.pvalue && settings.pvalue_prepeak

hidden
False

settings.pvalue

label
P-value cutoff

type
`basic:decimal`

description
The p-value cutoff. If specified, MACS2 will use p-value instead of q-value cutoff.

required
False

disabled
settings.qvalue

hidden
tagalign

settings.pvalue_prepeak

label
P-value cutoff

type
`basic:decimal`

description
The p-value cutoff. If specified, MACS2 will use p-value instead of q-value cutoff.

required
True

disabled
settings.qvalue

hidden
!tagalign || settings.qvalue

default
`1e-05`

settings.cap_num

label
Cap number of peaks by taking top N peaks

type
`basic:integer`

description
To keep all peaks set value to 0.

required
True

disabled
settings.broad

hidden
False

default
`500000`

settings.mfold_lower

label
MFOLD range (lower limit)

type
`basic:integer`

description
This parameter is used to select the regions within MFOLD range of high-confidence enrichment ratio against background to build model. The regions must be lower than upper limit, and higher than the lower limit of fold enrichment. DEFAULT:10,30 means using all regions not too low (>10) and not too high (<30) to build paired-peaks model. If MACS can not find more than 100 regions to build model, it will use the –extsize parameter to continue the peak detection ONLY if –fix-bimodal is set.

required
False

disabled
False

hidden
False

settings.mfold_upper

label
MFOLD range (upper limit)

type
`basic:integer`

description
This parameter is used to select the regions within MFOLD range of high-confidence enrichment ratio against background to build model. The regions must be lower than upper limit, and higher than the lower limit of fold enrichment. DEFAULT:10,30 means using all regions not too low (>10) and not too high (<30) to build paired-peaks model. If MACS can not find more than 100 regions to build model, it will use the –extsize parameter to continue the peak detection ONLY if –fix-bimodal is set.

required
False

disabled
False

hidden
False

settings.slocal

label
Small local region

type
`basic:integer`

description
Slocal and llocal parameters control which two levels of regions will be checked around the peak

regions to calculate the maximum lambda as local lambda. By default, MACS considers 1000bp for small local region (–slocal), and 10000bps for large local region (–llocal) which captures the bias from a long range effect like an open chromatin domain. You can tweak these according to your project. Remember that if the region is set too small, a sharp spike in the input data may kill the significant peak.

required
False

disabled
False

hidden
False

settings.llocal

label
Large local region

type
`basic:integer`

description
Slocal and llocal parameters control which two levels of regions will be checked around the peak regions to calculate the maximum lambda as local lambda. By default, MACS considers 1000bp for small local region (–slocal), and 10000bps for large local region (–llocal) which captures the bias from a long range effect like an open chromatin domain. You can tweak these according to your project. Remember that if the region is set too small, a sharp spike in the input data may kill the significant peak.

required
False

disabled
False

hidden
False

settings.extsize

label
Extension size [–extsize]

type
`basic:integer`

description
While '–nomodel' is set, MACS uses this parameter to extend reads in 5'->3' direction to fix-sized fragments. For example, if the size of binding region for your transcription factor is 200 bp, and you want to bypass the model building by MACS, this parameter can be set as 200. This option is only valid when –nomodel is set or when MACS fails to build model and –fix-bimodal is on.

required
False

disabled
False

hidden
False

settings.shift

label
Shift

type
`basic:integer`

description
Note, this is NOT the legacy –shiftsize option which is replaced by –extsize! You can set an arbitrary shift in bp here. Please Use discretion while setting it other than default value (0). When –nomodel is set, MACS will use this value to move cutting ends (5') then apply –extsize from 5' to 3' direction to extend them to fragments. When this value is negative, ends will be moved toward 3'->5' direction, otherwise 5'->3' direction. Recommended to keep it as default 0 for ChIP-Seq datasets, or -1 * half of EXTSIZE together with –extsize option for detecting enriched cutting loci such as certain DNAseI-Seq datasets. Note, you can't set values other than 0 if format is BAMPE for paired-end data. Default is 0.

required
False

disabled
False

hidden
settings.format == 'BAMPE'

settings.band_width

label
Band width

type
`basic:integer`

description
The band width which is used to scan the genome ONLY for model building. You can set this parameter as the sonication fragment size expected from wet experiment. The previous side effect on the peak detection process has been removed. So this parameter only affects the model building.

required
False

disabled
False

hidden
False

settings.nolambda

label
Use background lambda as local lambda

type
`basic:boolean`

description
With this flag on, MACS will use the background lambda as local lambda. This means MACS will not consider the local bias at peak candidate regions.

required
True

disabled
False

hidden
False

default
`False`

settings.fix_bimodal

label
Turn on the auto paired-peak model process

type
`basic:boolean`

description
Turn on the auto paired-peak model process. If it's set, when MACS failed to build paired model, it will use the nomodel settings, the '–extsize' parameter to extend each tag. If set, MACS will be terminated if paired-peak model has failed.

required
True

disabled
False

hidden
False

default
`False`

settings.nomodel

label
Bypass building the shifting model [–nomodel]

type
`basic:boolean`

description
While on, MACS will bypass building the shifting model.

required
True

disabled
False

hidden
tagalign

default
`False`

settings.nomodel_prepeak

label
Bypass building the shifting model [–nomodel]

type
`basic:boolean`

description
While on, MACS will bypass building the shifting model.

required
True

disabled
False

hidden
!tagalign

default
`True`

settings.down_sample

label
Down-sample

type
`basic:boolean`

description
When set to true, random sampling method will scale down the bigger sample. By default, MACS uses linear scaling. This option will make the results unstable and unreproducible since each time, random reads would be selected, especially the numbers (pileup, pvalue, qvalue) would change.

required
True

disabled
False

hidden
False

default
`False`

settings.bedgraph

label
Save fragment pileup and control lambda

type
`basic:boolean`

description
If this flag is on, MACS will store the fragment pileup, control lambda, -log10pvalue and -log10qvalue scores in bedGraph files. The bedGraph files will be stored in current directory named NAME+'_treat_pileup.bdg' for treatment data, NAME+'_control_lambda.bdg' for local lambda values from control, NAME+'_treat_pvalue.bdg' for Poisson pvalue scores (in -log10(pvalue) form), and NAME+'_treat_qvalue.bdg' for q-value scores from Benjamini-Hochberg-Yekutieli procedure.

required
True

disabled
False

hidden
False

default
`True`

settings.spmr

label
Save fragment pileup and control lambda

type
`basic:boolean`

required
True

disabled
settings.bedgraph === false

hidden
False

default
`True`

settings.call_summits

label
Call summits [–call-summits]

type
`basic:boolean`

description
MACS will now reanalyze the shape of signal profile (p or q-score depending on cutoff setting) to deconvolve subpeaks within each peak called from general procedure. It's highly recommended to detect adjacent binding events. While used, the output subpeaks of a big peak region will have the same peak boundaries, and different scores and peak summit positions.

required
True

disabled
False

hidden
False

default
`False`

settings.broad

label
Composite broad regions [–broad]

type
`basic:boolean`

description
When this flag is on, MACS will try to composite broad regions in BED12 (a gene-model-like format) by putting nearby highly enriched regions into a broad region with loose cutoff. The broad region is controlled by another cutoff through –broad-cutoff. The maximum length of broad region length is 4 times of d from MACS.

required
True

disabled
settings.call_summits === true

hidden
False

default
`False`

settings.broad_cutoff

label
Broad cutoff

type
`basic:decimal`

description
Cutoff for broad region. This option is not available unless –broad is set. If -p is set, this is a p-value cutoff, otherwise, it's a q-value cutoff. DEFAULT = 0.1

required
False

disabled
settings.call_summits === true || settings.broad !== true

hidden
False

Output results called_peaks

label
Called peaks

type
`basic:file`

required
True

disabled
False

hidden
False

narrow_peaks

label
Narrow peaks

type
`basic:file`

required
False

disabled
False

hidden
False

chip_qc

label
QC report

type
`basic:file`

required
False

disabled
False

hidden
False

case_prepeak_qc

label
Pre-peak QC report (case)

type
`basic:file`

required
True

disabled
False

hidden
False

case_tagalign

label
Filtered tagAlign (case)

type
`basic:file`

required
True

disabled
False

hidden
False

case_bam

label
Filtered BAM (case)

type
`basic:file`

required
True

disabled
False

hidden
False

case_bai

label
Filtered BAM index (case)

type
`basic:file`

required
True

disabled
False

hidden
False

control_prepeak_qc

label
Pre-peak QC report (control)

type
`basic:file`

required
False

disabled
False

hidden
False

control_tagalign

label
Filtered tagAlign (control)

type
`basic:file`

required
False

disabled
False

hidden
False

control_bam

label
Filtered BAM (control)

type
`basic:file`

required
False

disabled
False

hidden
False

control_bai

label
Filtered BAM index (control)

type
`basic:file`

required
False

disabled
False

hidden
False

narrow_peaks_bigbed_igv_ucsc

label
Narrow peaks (BigBed)

type
`basic:file`

required
False

disabled
False

hidden
False

summits

label
Peak summits

type
`basic:file`

required
False

disabled
False

hidden
False

summits_tbi_jbrowse

label
Peak summits tbi index for JBrowse

type
`basic:file`

required
False

disabled
False

hidden
False

summits_bigbed_igv_ucsc

label
Summits (bigBed)

type
`basic:file`

required
False

disabled
False

hidden
False

broad_peaks

label
Broad peaks

type
`basic:file`

required
False

disabled
False

hidden
False

gappedPeak

label
Broad peaks (bed12/gappedPeak)

type
`basic:file`

required
False

disabled
False

hidden
False

treat_pileup

label
Treatment pileup (bedGraph)

type
`basic:file`

required
False

disabled
False

hidden
False

treat_pileup_bigwig

label
Treatment pileup (bigWig)

type
`basic:file`

required
False

disabled
False

hidden
False

control_lambda

label
Control lambda (bedGraph)

type
`basic:file`

required
False

disabled
False

hidden
False

control_lambda_bigwig

label
Control lambda (bigwig)

type
`basic:file`

required
False

disabled
False

hidden
False

model

label
Model

type
`basic:file`

required
False

disabled
False

hidden
False

species

label
Species

type
`basic:string`

required
True

disabled
False

hidden
False

build

label
Build

type
`basic:string`

required
True

disabled
False

hidden
False

MACS2

`data:workflow:chipseq:macs2rose2workflow-macs2` (*data:alignment:bam* **case**, *data:alignment:bam* **control**, *data:bed* **promoter**, *basic:boolean* **tagalign**, *basic:integer* **q_threshold**, *basic:integer* **n_sub**, *basic:boolean* **tn5**, *basic:integer* **shift**, *basic:string* **duplicates**, *basic:string* **duplicates_prepeak**, *basic:decimal* **qvalue**, *basic:decimal* **pvalue**, *basic:decimal* **pvalue_prepeak**, *basic:integer* **cap_num**, *basic:integer* **mfold_lower**, *basic:integer* **mfold_upper**, *basic:integer* **slocal**, *basic:integer* **llocal**, *basic:integer* **extsize**, *basic:integer* **shift**, *basic:integer* **band_width**, *basic:boolean* **nolambda**, *basic:boolean* **fix_bimodal**, *basic:boolean* **nomodel**, *basic:boolean* **nomodel_prepeak**, *basic:boolean* **down_sample**, *basic:boolean* **bedgraph**, *basic:boolean* **spmr**, *basic:boolean* **call_summits**, *basic:boolean* **broad**, *basic:decimal* **broad_cutoff**, *data:bed* **blacklist**, *basic:boolean* **calculate_enrichment**, *basic:integer* **profile_window**, *basic:string* **shift_size**) [Source: v1.2.0]

Input arguments case

label
Case (treatment)

type
`data:alignment:bam`

control

label
Control (background)

type
`data:alignment:bam`

required
False

promoter

label
Promoter regions BED file

type
`data:bed`

description
BED file containing promoter regions (TSS+-1000 bp for example). Needed to get the number of peaks and reads mapped to promoter regions.

required
False

tagalign

label
Use tagAlign files

type
`basic:boolean`

description
Use filtered tagAlign files as case (treatment) and control (background) samples. If extsize parameter is not set, run MACS using input's estimated fragment length.

default
`False`

prepeakqc_settings.q_threshold

label
Quality filtering threshold

type
`basic:integer`

default
`30`

prepeakqc_settings.n_sub

label
Number of reads to subsample

type
`basic:integer`

default
`15000000`

prepeakqc_settings.tn5

label
Tn5 shifting

type
`basic:boolean`

description
Tn5 transposon shifting. Shift reads on "+" strand by 4 bp and reads on "-" strand by 5 bp.

default
`False`

prepeakqc_settings.shift

label
User-defined cross-correlation peak strandshift

type
`basic:integer`

description
If defined, SPP tool will not try to estimate fragment length but will use the given value as fragment length.

required
False

settings.duplicates

label
Number of duplicates

type
`basic:string`

description
It controls the MACS behavior towards duplicate tags at the exact same location – the same coordination and the same strand. The 'auto' option makes MACS calculate the maximum tags at the exact same location based on binomal distribution using 1e-5 as pvalue cutoff and the 'all' option keeps all the tags. If an integer is given, at most this number of tags will be kept at the same location. The default is to keep one tag at the same location.

required
False

hidden
tagalign

choices

- 1: `1`
- auto: `auto`
- all: `all`

settings.duplicates_prepeak

label
Number of duplicates

type
`basic:string`

description
It controls the MACS behavior towards duplicate tags at the exact same location – the same coordination and the same strand. The 'auto' option makes MACS calculate the maximum tags at the exact same location based on binomal distribution using 1e-5 as pvalue cutoff and the 'all' option keeps all the tags. If an integer is given, at most this number of tags will be kept at the same location. The default is to keep one tag at the same location.

required
False

hidden
!tagalign

default
`all`

choices

- 1: `1`
- auto: `auto`
- all: `all`

settings.qvalue

label
Q-value cutoff

type
`basic:decimal`

description
The q-value (minimum FDR) cutoff to call significant regions. Q-values are calculated from p-values using Benjamini-Hochberg procedure.

required
False

disabled
settings.pvalue && settings.pvalue_prepeak

settings.pvalue

label
P-value cutoff

type
`basic:decimal`

description
The p-value cutoff. If specified, MACS2 will use p-value instead of q-value cutoff.

required
False

disabled
settings.qvalue

hidden
tagalign

settings.pvalue_prepeak

label
P-value cutoff

type
`basic:decimal`

description
The p-value cutoff. If specified, MACS2 will use p-value instead of q-value cutoff.

disabled
settings.qvalue

hidden
!tagalign || settings.qvalue

default
`1e-05`

settings.cap_num

label
Cap number of peaks by taking top N peaks

type
`basic:integer`

description
To keep all peaks set value to 0.

disabled
settings.broad

default
`500000`

settings.mfold_lower

label
MFOLD range (lower limit)

type
`basic:integer`

description
This parameter is used to select the regions within MFOLD range of high-confidence enrichment ratio against background to build model. The regions must be lower than upper limit, and higher than the lower limit of fold enrichment. DEFAULT:10,30 means using all regions not too low (>10) and not too high (<30) to build paired-peaks model. If MACS can not find more than 100 regions to build model, it will use the –extsize parameter to continue the peak detection ONLY if –fix-bimodal is set.

required
False

settings.mfold_upper

label
MFOLD range (upper limit)

type
`basic:integer`

description
This parameter is used to select the regions within MFOLD range of high-confidence enrichment ratio against background to build model. The regions must be lower than upper limit, and higher than the lower limit of fold enrichment. DEFAULT:10,30 means using all regions not too low (>10) and not too high (<30) to build paired-peaks model. If MACS can not find more than 100 regions to build model, it will use the –extsize parameter to continue the peak detection ONLY if –fix-bimodal is set.

required
False

settings.slocal

label
Small local region

type
`basic:integer`

description
Slocal and llocal parameters control which two levels of regions will be checked around the peak regions to calculate the maximum lambda as local lambda. By default, MACS considers 1000 bp for small local region (–slocal), and 10000 bp for large local region (–llocal) which captures the bias from a long range effect like an open chromatin domain. You can tweak these according to your project. Remember that if the region is set too small, a sharp spike in the input data may kill the significant peak.

required
False

settings.llocal

label
Large local region

type
`basic:integer`

description
Slocal and llocal parameters control which two levels of regions will be checked around the peak regions to calculate the maximum lambda as local lambda. By default, MACS considers 1000 bp for small local region (–slocal), and 10000 bp for large local region (–llocal) which captures the bias from a long range effect like an open chromatin domain. You can tweak these according to your project. Remember that if the region is set too small, a sharp spike in the input data may kill the significant peak.

required
False

settings.extsize

label
extsize

type
`basic:integer`

description
While '–nomodel' is set, MACS uses this parameter to extend reads in 5'->3' direction to fix-sized fragments. For example, if the size of binding region for your transcription factor is 200 bp, and you want to bypass the model building by MACS, this parameter can be set as 200. This option is only valid when –nomodel is set or when MACS fails to build model and –fix-bimodal is on.

required
False

settings.shift

label
Shift

type
`basic:integer`

description
Note, this is NOT the legacy –shiftsize option which is replaced by –extsize! You can set an arbitrary shift in bp here. Please Use discretion while setting it other than default value (0). When –nomodel is set, MACS will use this value to move cutting ends (5') then apply –extsize from 5' to 3' direction to extend them to fragments. When this value is negative, ends will be moved toward 3'->5' direction, otherwise 5'->3' direction. Recommended to keep it as default 0 for ChIP-Seq datasets, or -1 * half of EXTSIZE together with –extsize option for detecting enriched cutting loci such as certain DNAseI-Seq datasets. Note, you can't set values other than 0 if format is BAMPE for paired-end data. Default is 0.

required
False

settings.band_width

label
Band width

type
`basic:integer`

description
The band width which is used to scan the genome ONLY for model building. You can set this parameter as the sonication fragment size expected from wet experiment. The previous side effect on the peak detection process has been removed. So this parameter only affects the model building.

required
False

settings.nolambda

label
Use backgroud lambda as local lambda

type
`basic:boolean`

description
With this flag on, MACS will use the background lambda as local lambda. This means MACS will not consider the local bias at peak candidate regions.

default
`False`

settings.fix_bimodal

label
Turn on the auto paired-peak model process

type
`basic:boolean`

description
Turn on the auto paired-peak model process. If it's set, when MACS failed to build paired model, it will use the nomodel settings, the '–extsize' parameter to extend each tag. If set, MACS will be terminated if paired-peak model is failed.

default
`False`

settings.nomodel

label
Bypass building the shifting model

type
`basic:boolean`

description
While on, MACS will bypass building the shifting model.

hidden
tagalign

default
`False`

settings.nomodel_prepeak

label
Bypass building the shifting model

type
basic:boolean

description
While on, MACS will bypass building the shifting model.

hidden
!tagalign

default
True

settings.down_sample

label
Down-sample

type
basic:boolean

description
When set to true, random sampling method will scale down the bigger sample. By default, MACS uses linear scaling. This option will make the results unstable and irreproducible since each time, random reads would be selected, especially the numbers (pileup, pvalue, qvalue) would change.

default
False

settings.bedgraph

label
Save fragment pileup and control lambda

type
basic:boolean

description
If this flag is on, MACS will store the fragment pileup, control lambda, -log10pvalue and -log10qvalue scores in bedGraph files. The bedGraph files will be stored in current directory named NAME+'_treat_pileup.bdg' for treatment data, NAME+'_control_lambda.bdg' for local lambda values from control, NAME+'_treat_pvalue.bdg' for Poisson pvalue scores (in -log10(pvalue) form), and NAME+'_treat_qvalue.bdg' for q-value scores from Benjamini-Hochberg-Yekutieli procedure.

default
True

settings.spmr

label
Save signal per million reads for fragment pileup profiles

type
basic:boolean

disabled
settings.bedgraph === false

default
True

settings.call_summits

label
Call summits

type
`basic:boolean`

description
MACS will now reanalyze the shape of signal profile (p or q-score depending on cutoff setting) to deconvolve subpeaks within each peak called from general procedure. It's highly recommended to detect adjacent binding events. While used, the output subpeaks of a big peak region will have the same peak boundaries, and different scores and peak summit positions.

default
`False`

settings.broad

label
Composite broad regions

type
`basic:boolean`

description
When this flag is on, MACS will try to composite broad regions in BED12 (a gene-model-like format) by putting nearby highly enriched regions into a broad region with loose cutoff. The broad region is controlled by another cutoff through –broad-cutoff. The maximum length of broad region length is 4 times of d from MACS.

disabled
settings.call_summits === true

default
`False`

settings.broad_cutoff

label
Broad cutoff

type
`basic:decimal`

description
Cutoff for broad region. This option is not available unless –broad is set. If -p is set, this is a p-value cutoff, otherwise, it's a q-value cutoff. DEFAULT = 0.1

required
False

disabled
settings.call_summits === true || settings.broad !== true

chipqc_settings.blacklist

label
Blacklist regions

type
`data:bed`

description
BED file containing genomic regions that should be excluded from the analysis.

required
False

chipqc_settings.calculate_enrichment

label
Calculate enrichment

type
`basic:boolean`

description
Calculate enrichment of signal in known genomic annotation. By default annotation is provided from the TranscriptDB package specified by genome build which should match one of the supported annotations (hg19, hg38, hg18, mm10, mm9, rn4, ce6, dm3). If annotation is not supported the analysis is skipped.

default
`False`

chipqc_settings.profile_window

label
Window size

type
`basic:integer`

description
An integer indicating the width of the window used for peak profiles. Peaks will be centered on their summits and include half of the window size upstream and half downstream of this point.

default
`400`

chipqc_settings.shift_size

label
Shift size

type
`basic:string`

description
Vector of values to try when computing optimal shift sizes. It should be specified as consecutive numbers vector with start:end

default
`1:300`

Output results

MACS2 - ROSE2

`data:workflow:chipseq:macs2rose2workflow-macs-rose` (*data:alignment:bam* **case**, *data:alignment:bam* **control**, *data:bed* **promoter**, *basic:boolean* **tagalign**, *basic:integer* **q_threshold**, *basic:integer* **n_sub**, *basic:boolean* **tn5**, *basic:integer* **shift**, *basic:string* **duplicates**, *basic:string* **duplicates_prepeak**, *basic:decimal* **qvalue**, *basic:decimal* **pvalue**, *basic:decimal* **pvalue_prepeak**, *basic:integer* **cap_num**, *basic:integer* **mfold_lower**, *basic:integer* **mfold_upper**, *basic:integer* **slocal**, *basic:integer* **llocal**, *basic:integer* **extsize**, *basic:integer* **shift**, *basic:integer* **band_width**, *basic:boolean* **nolambda**, *basic:boolean* **fix_bimodal**, *basic:boolean* **nomodel**, *basic:boolean* **nomodel_prepeak**, *basic:boolean* **down_sample**, *basic:boolean* **bedgraph**, *basic:boolean* **spmr**, *basic:boolean* **call_summits**, *basic:boolean* **broad**, *basic:decimal* **broad_cutoff**, *basic:boolean* **use_filtered_bam**, *basic:integer* **tss**, *basic:integer* **stitch**, *data:bed* **mask**, *data:bed* **blacklist**, *basic:boolean* **calculate_enrichment**, *basic:integer* **profile_window**, *basic:string* **shift_size**) [Source: v1.4.0]

Input arguments case

label
Case (treatment)

type
`data:alignment:bam`

control

label
Control (background)

type
`data:alignment:bam`

required
False

promoter

label
Promoter regions BED file

type
`data:bed`

description
BED file containing promoter regions (TSS+-1000 bp for example). Needed to get the number of peaks and reads mapped to promoter regions.

required
False

tagalign

label
Use tagAlign files

type
`basic:boolean`

description
Use filtered tagAlign files as case (treatment) and control (background) samples. If extsize parameter is not set, run MACS using input's estimated fragment length.

default
`False`

prepeakqc_settings.q_threshold

label
Quality filtering threshold

type
`basic:integer`

default
`30`

prepeakqc_settings.n_sub

label
Number of reads to subsample

type
`basic:integer`

default
`15000000`

prepeakqc_settings.tn5

label
Tn5 shifting

type
`basic:boolean`

description
Tn5 transposon shifting. Shift reads on "+" strand by 4 bp and reads on "-" strand by 5 bp.

default
`False`

prepeakqc_settings.shift

label
User-defined cross-correlation peak strandshift

type
`basic:integer`

description
If defined, SPP tool will not try to estimate fragment length but will use the given value as fragment length.

required
False

settings.duplicates

label
Number of duplicates

type
`basic:string`

description
It controls the MACS behavior towards duplicate tags at the exact same location – the same coordination and the same strand. The 'auto' option makes MACS calculate the maximum tags at the exact same location based on binomal distribution using 1e-5 as pvalue cutoff and the 'all' option keeps all the tags. If an integer is given, at most this number of tags will be kept at the same location. The default is to keep one tag at the same location.

required
False

hidden
tagalign

choices

- 1: `1`
- auto: `auto`
- all: `all`

settings.duplicates_prepeak

label
Number of duplicates

type
`basic:string`

description
It controls the MACS behavior towards duplicate tags at the exact same location – the same coordination and the same strand. The 'auto' option makes MACS calculate the maximum tags at the exact same location based on binomal distribution using 1e-5 as pvalue cutoff and the 'all' option keeps all the tags. If an integer is given, at most this number of tags will be kept at the same location. The default is to keep one tag at the same location.

required
False

hidden
!tagalign

default
`all`

choices

- 1: `1`

- auto: `auto`
- all: `all`

settings.qvalue

label
Q-value cutoff

type
`basic:decimal`

description
The q-value (minimum FDR) cutoff to call significant regions. Q-values are calculated from p-values using Benjamini-Hochberg procedure.

required
False

disabled
settings.pvalue && settings.pvalue_prepeak

settings.pvalue

label
P-value cutoff

type
`basic:decimal`

description
The p-value cutoff. If specified, MACS2 will use p-value instead of q-value cutoff.

required
False

disabled
settings.qvalue

hidden
tagalign

settings.pvalue_prepeak

label
P-value cutoff

type
`basic:decimal`

description
The p-value cutoff. If specified, MACS2 will use p-value instead of q-value cutoff.

disabled
settings.qvalue

hidden
!tagalign || settings.qvalue

default
`1e-05`

settings.cap_num

label
Cap number of peaks by taking top N peaks

type
`basic:integer`

description
To keep all peaks set value to 0.

disabled
settings.broad

default
`500000`

settings.mfold_lower

label
MFOLD range (lower limit)

type
`basic:integer`

description
This parameter is used to select the regions within MFOLD range of high-confidence enrichment ratio against background to build model. The regions must be lower than upper limit, and higher than the lower limit of fold enrichment. DEFAULT:10,30 means using all regions not too low (>10) and not too high (<30) to build paired-peaks model. If MACS can not find more than 100 regions to build model, it will use the –extsize parameter to continue the peak detection ONLY if –fix-bimodal is set.

required
False

settings.mfold_upper

label
MFOLD range (upper limit)

type
`basic:integer`

description
This parameter is used to select the regions within MFOLD range of high-confidence enrichment ratio against background to build model. The regions must be lower than upper limit, and higher than the lower limit of fold enrichment. DEFAULT:10,30 means using all regions not too low (>10) and not too high (<30) to build paired-peaks model. If MACS can not find more than 100 regions to build model, it will use the –extsize parameter to continue the peak detection ONLY if –fix-bimodal is set.

required
False

settings.slocal

label
Small local region

type
`basic:integer`

description
Slocal and llocal parameters control which two levels of regions will be checked around the peak

regions to calculate the maximum lambda as local lambda. By default, MACS considers 1000 bp for small local region (–slocal), and 10000 bp for large local region (–llocal) which captures the bias from a long range effect like an open chromatin domain. You can tweak these according to your project. Remember that if the region is set too small, a sharp spike in the input data may kill the significant peak.

required
False

settings.llocal

label
Large local region

type
`basic:integer`

description
Slocal and llocal parameters control which two levels of regions will be checked around the peak regions to calculate the maximum lambda as local lambda. By default, MACS considers 1000 bp for small local region (–slocal), and 10000 bp for large local region (–llocal) which captures the bias from a long range effect like an open chromatin domain. You can tweak these according to your project. Remember that if the region is set too small, a sharp spike in the input data may kill the significant peak.

required
False

settings.extsize

label
extsize

type
`basic:integer`

description
While '–nomodel' is set, MACS uses this parameter to extend reads in 5'->3' direction to fix-sized fragments. For example, if the size of binding region for your transcription factor is 200 bp, and you want to bypass the model building by MACS, this parameter can be set as 200. This option is only valid when –nomodel is set or when MACS fails to build model and –fix-bimodal is on.

required
False

settings.shift

label
Shift

type
`basic:integer`

description
Note, this is NOT the legacy –shiftsize option which is replaced by –extsize! You can set an arbitrary shift in bp here. Please Use discretion while setting it other than default value (0). When –nomodel is set, MACS will use this value to move cutting ends (5') then apply –extsize from 5' to 3' direction to extend them to fragments. When this value is negative, ends will be moved toward 3'->5' direction, otherwise 5'->3' direction. Recommended to keep it as default 0 for ChIP-Seq datasets, or -1 * half of EXTSIZE together with –extsize option for detecting enriched cutting loci such as certain DNAseI-

Seq datasets. Note, you can't set values other than 0 if format is BAMPE for paired-end data. Default is 0.

required
False

settings.band_width

label
Band width

type
`basic:integer`

description
The band width which is used to scan the genome ONLY for model building. You can set this parameter as the sonication fragment size expected from wet experiment. The previous side effect on the peak detection process has been removed. So this parameter only affects the model building.

required
False

settings.nolambda

label
Use backgroud lambda as local lambda

type
`basic:boolean`

description
With this flag on, MACS will use the background lambda as local lambda. This means MACS will not consider the local bias at peak candidate regions.

default
`False`

settings.fix_bimodal

label
Turn on the auto paired-peak model process

type
`basic:boolean`

description
Turn on the auto paired-peak model process. If it's set, when MACS failed to build paired model, it will use the nomodel settings, the '–extsize' parameter to extend each tag. If set, MACS will be terminated if paired-peak model has failed.

default
`False`

settings.nomodel

label
Bypass building the shifting model

type
`basic:boolean`

description
While on, MACS will bypass building the shifting model.

hidden
tagalign

default
`False`

settings.nomodel_prepeak

label
Bypass building the shifting model

type
`basic:boolean`

description
While on, MACS will bypass building the shifting model.

hidden
!tagalign

default
`True`

settings.down_sample

label
Down-sample

type
`basic:boolean`

description
When set to true, random sampling method will scale down the bigger sample. By default, MACS uses linear scaling. This option will make the results unstable and irreproducible since each time, random reads would be selected, especially the numbers (pileup, pvalue, qvalue) would change.

default
`False`

settings.bedgraph

label
Save fragment pileup and control lambda

type
`basic:boolean`

description
If this flag is on, MACS will store the fragment pileup, control lambda, -log10pvalue and -log10qvalue scores in bedGraph files. The bedGraph files will be stored in current directory named NAME+'_treat_pileup.bdg' for treatment data, NAME+'_control_lambda.bdg' for local lambda values from control, NAME+'_treat_pvalue.bdg' for Poisson pvalue scores (in -log10(pvalue) form), and NAME+'_treat_qvalue.bdg' for q-value scores from Benjamini-Hochberg-Yekutieli procedure.

default
`True`

settings.spmr

label
Save signal per million reads for fragment pileup profiles

type
`basic:boolean`

disabled
settings.bedgraph === false

default
`True`

settings.call_summits

label
Call summits

type
`basic:boolean`

description
MACS will now reanalyze the shape of signal profile (p or q-score depending on cutoff setting) to deconvolve subpeaks within each peak called from general procedure. It's highly recommended to detect adjacent binding events. While used, the output subpeaks of a big peak region will have the same peak boundaries, and different scores and peak summit positions.

default
`False`

settings.broad

label
Composite broad regions

type
`basic:boolean`

description
When this flag is on, MACS will try to composite broad regions in BED12 (a gene-model-like format) by putting nearby highly enriched regions into a broad region with loose cutoff. The broad region is controlled by another cutoff through –broad-cutoff. The maximum length of broad region length is 4 times of d from MACS.

disabled
settings.call_summits === true

default
`False`

settings.broad_cutoff

label
Broad cutoff

type
`basic:decimal`

description
Cutoff for broad region. This option is not available unless –broad is set. If -p is set, this is a p-value cutoff, otherwise, it's a q-value cutoff. DEFAULT = 0.1

required
False

disabled
settings.call_summits === true || settings.broad !== true

rose_settings.use_filtered_bam

label
Use Filtered BAM File

type
`basic:boolean`

description
Use filtered BAM file from a MACS2 object to rank enhancers by.

default
`False`

rose_settings.tss

label
TSS exclusion

type
`basic:integer`

description
Enter a distance from TSS to exclude. 0 = no TSS exclusion

default
`0`

rose_settings.stitch

label
Stitch

type
`basic:integer`

description
Enter a max linking distance for stitching. If not given, optimal stitching parameter will be determined automatically.

required
False

rose_settings.mask

label
Masking BED file

type
`data:bed`

description
Mask a set of regions from analysis. Provide a BED of masking regions.

required
False

chipqc_settings.blacklist

label
Blacklist regions

type
`data:bed`

description
BED file containing genomic regions that should be excluded from the analysis.

required
False

chipqc_settings.calculate_enrichment

label
Calculate enrichment

type
`basic:boolean`

description
Calculate enrichment of signal in known genomic annotation. By default annotation is provided from the TranscriptDB package specified by genome build which should match one of the supported annotations (hg19, hg38, hg18, mm10, mm9, rn4, ce6, dm3). If annotation is not supported the analysis is skipped.

default
`False`

chipqc_settings.profile_window

label
Window size

type
`basic:integer`

description
An integer indicating the width of the window used for peak profiles. Peaks will be centered on their summits and include half of the window size upstream and half downstream of this point.

default
`400`

chipqc_settings.shift_size

label
Shift size

type
`basic:string`

description
Vector of values to try when computing optimal shift sizes. It should be specified as consecutive numbers vector with start:end

default
`1:300`

Output results

ML-ready expression

`data:ml:table:expressions:upload-ml-expression` (*basic:file* **exp**, *basic:string* **source**, *basic:string* **species**, *data:ml:space* **reference_space**) [Source: v1.0.2]

Upload ML-ready expression matrix.

Input arguments exp

label
Transformed expressions

type
`basic:file`

description
A TAB separated file containing transformed expression values with sample IDs for index (first column with label sample_id) and ENSEMBL IDs (recommended but not required) for the column names.

required
True

disabled
False

hidden
False

source

label
Feature source

type
`basic:string`

required
True

disabled
False

hidden
False

choices

- AFFY: `AFFY`
- DICTYBASE: `DICTYBASE`
- ENSEMBL: `ENSEMBL`
- NCBI: `NCBI`
- UCSC: `UCSC`

species

label
Species

type
`basic:string`

description
Species latin name.

required
True

disabled
False

hidden
False

choices

- Homo sapiens: `Homo sapiens`
- Mus musculus: `Mus musculus`
- Rattus norvegicus: `Rattus norvegicus`
- Dictyostelium discoideum: `Dictyostelium discoideum`

reference_space

label
Reference space of ML-ready data

type
`data:ml:space`

required
True

disabled
False

hidden
False

Output results exp

label
Transformed expressions

type
`basic:file`

required
True

disabled
False

hidden
False

source

label
Feature source

type
`basic:string`

required
True

disabled
False

hidden
False

species

label
Species

type
`basic:string`

required
True

disabled
False

hidden
False

Map microarray probes

`data:microarray:mapping:map-microarray-probes` (*list:data:microarray:normalized* **expressions**, *basic:file* **mapping_file**, *basic:string* **source**, *basic:string* **build**) [Source: v1.1.1]

Map microarray probes to Gene IDs.

Mapping can be done automatically or using a custom mapping file. For automatic probe mapping all 'Normalized expression' objects should have a GEO platform ID. If the platform is supported the provided probe IDs will be mapped to the corresponding Ensembl IDs. Currently supported platforms are: GPL74, GPL201, GPL96, GPL571, GPL97, GPL570, GPL91, GPL8300, GPL92, GPL93, GPL94, GPL95, GPL17586, GPL5175, GPL80, GPL6244, GPL16686, GPL15207, GPL1352, GPL11068, GPL26966, GPL6848, GPL14550, GPL17077, GPL16981, GPL13497, GPL6947, GPL10558, GPL6883, GPL13376,GPL6884, GPL6254.

Input arguments expressions

label
Normalized expressions

type
`list:data:microarray:normalized`

required
True

disabled
False

hidden
False

mapping_file

label
File with probe ID mappings

type
basic:file

description
The file should be tab-separated and contain two columns with their column names. The first column should contain Gene IDs and the second one should contain probe names. Supported file extensions are .tab.*, .tsv.*, .txt.*

required
False

disabled
False

hidden
False

source

label
Gene ID source

type
basic:string

description
Gene ID source used for probe mapping is required when using a custom file.

required
False

disabled
False

hidden
False

choices

- AFFY: AFFY
- DICTYBASE: DICTYBASE
- ENSEMBL: ENSEMBL
- NCBI: NCBI
- UCSC: UCSC

build

label
Genome build

type
basic:string

description
Genome build of mapping file is required when using a custom file.

required
False

disabled
False

hidden
False

Output results mapped_exp

label
Mapped expressions

type
`basic:file`

required
True

disabled
False

hidden
False

probe_mapping

label
Probe to transcript mapping used

type
`basic:string`

required
True

disabled
False

hidden
False

mapping

label
Mapping file

type
`basic:file`

required
True

disabled
False

hidden
False

platform

label
Microarray platform type

type
`basic:string`

required
True

disabled
False

hidden
False

platform_id

label
GEO platform ID

type
`basic:string`

required
False

disabled
False

hidden
False

Mappability

`data:mappability:bcmmappability-bcm` (*data:index:bowtie* **genome**, *data:annotation:gff3* **gff**, *basic:integer* **length**) [Source: v3.1.2]

Compute genome mappability. Developed by Bioinformatics Laboratory, Faculty of Computer and Information Science, University of Ljubljana, Slovenia and Shaulsky's Lab, Department of Molecular and Human Genetics, Baylor College of Medicine, Houston, TX, USA.

Input arguments genome

label
Reference genome

type
`data:index:bowtie`

gff

label
General feature format

type
`data:annotation:gff3`

length

label
Read length

type
`basic:integer`

default
`50`

Output results mappability

label
Mappability

type
`basic:file`

Mappability info

data:mappability:bcmupload-mappability (*basic:file* **src**)[Source: v1.2.3]

Upload mappability information.

Input arguments src

label
Mappability file

type
`basic:file`

description
Mappability file: 2 column tab separated

validate_regex
`\.(tab)(|\.gz|\.bz2|\.tgz|\.tar\.gz|\.tar\.bz2|\.zip|\.rar|\.7z)$`

Output results mappability

label
Uploaded mappability

type
`basic:file`

MarkDuplicates

data:alignment:bam:markduplicate:markduplicates (*data:alignment:bam* **bam**, *basic:boolean* **skip**, *basic:boolean* **remove_duplicates**, *basic:string* **validation_stringency**, *basic:string* **assume_sort_order**, *basic:integer* **java_gc_threads**, *basic:integer* **max_heap_size**)[Source: v1.7.0]

Remove duplicate reads from BAM file.

Tool from Picard, wrapped by GATK4. See GATK MarkDuplicates for more information.

Input arguments bam

label
Alignment BAM file

type
`data:alignment:bam`

required
True

disabled
False

hidden
False

skip

label
Skip MarkDuplicates step

type
`basic:boolean`

description
MarkDuplicates step can be skipped.

required
True

disabled
False

hidden
False

default
`False`

remove_duplicates

label
Remove duplicates

type
`basic:boolean`

description
If true do not write duplicates to the output file instead of writing them with appropriate flags set.

required
True

disabled
False

hidden
False

default
`False`

validation_stringency

label
Validation stringency

type
`basic:string`

description
Validation stringency for all SAM files read by this program. Setting stringency to SILENT can improve performance when processing a BAM file in which variable-length data (read, qualities, tags) do not otherwise need to be decoded. Default is STRICT.

required
True

disabled
False

hidden
False

default
`STRICT`

choices

- STRICT: `STRICT`
- LENIENT: `LENIENT`
- SILENT: `SILENT`

assume_sort_order

label
Assume sort order

type
`basic:string`

description
If not null (default), assume that the input file has this order even if the header says otherwise.Possible values are unsorted, queryname, coordinate and unknown.

required
True

disabled
False

hidden
False

default

choices

- as in BAM header (default):
- unsorted: `unsorted`
- queryname: `queryname`
- coordinate: `coordinate`
- duplicate: `duplicate`
- unknown: `unknown`

advanced.java_gc_threads

label
Java ParallelGCThreads

type
`basic:integer`

description
Sets the number of threads used during parallel phases of the garbage collectors.

required
True

disabled
False

hidden
False

default
2

advanced.max_heap_size

label
Java maximum heap size (Xmx)

type
basic:integer

description
Set the maximum Java heap size (in GB).

required
True

disabled
False

hidden
False

default
12

Output results bam

label
Marked duplicates BAM file

type
basic:file

required
True

disabled
False

hidden
False

bai

label
Index of marked duplicates BAM file

type
basic:file

required
True

disabled
False

hidden
False

stats

label
Alignment statistics

type
`basic:file`

required
True

disabled
False

hidden
False

species

label
Species

type
`basic:string`

required
True

disabled
False

hidden
False

build

label
Build

type
`basic:string`

required
True

disabled
False

hidden
False

metrics_file

label
Metrics from MarkDuplicate process

type
`basic:file`

required
True

disabled
False

hidden
False

Merge Expressions (ETC)

`data:expressionset:etcmergeetc` (*list:data:etc* **exps**, *list:basic:string* **genes**)[Source: v1.2.4]

Merge Expression Time Course (ETC) data.

Input arguments exps

label
Expression Time Course (ETC)

type
`list:data:etc`

genes

label
Filter genes

type
`list:basic:string`

required
False

Output results expset

label
Expression set

type
`basic:file`

expset_type

label
Expression set type

type
`basic:string`

Merge FASTQ (paired-end)

`data:mergereads:paired:merge-fastq-paired` (*list:data:reads:fastq:paired:* **reads**)[Source: v2.2.2]

Merge paired-end FASTQs into one sample.

Samples are merged based on the defined replicate group relations and then uploaded as separate samples.

Input arguments reads

label
Select relations

type
`list:data:reads:fastq:paired:`

description
Define and select Replicate relations.

required
True

disabled
False

hidden
False

Output results

Merge FASTQ (single-end)

`data:mergereads:single:merge-fastq-single` (*list:data:reads:fastq:single:* **reads**) [Source: v2.2.2]

Merge single-end FASTQs into one sample.

Samples are merged based on the defined replicate group relations and then uploaded as separate samples.

Input arguments reads

label
Select relations

type
`list:data:reads:fastq:single:`

description
Define and select replicate relations.

required
True

disabled
False

hidden
False

Output results

Metadata table

`data:metadata:upload-metadata` (*basic:file* **src**) [Source: v1.1.1]

Upload metadata file where more than one row can match to a single sample.

The uploaded metadata table represents one-to-many (1:n) relation to samples in the working collection. Metadata table must contain a column with one of the following headers: “Sample ID”, “Sample name” or “Sample slug”.

Input arguments src

label
Table with metadata

type
`basic:file`

description
The metadata table should use one of the following extensions: .csv, .tab, .tsv, .xlsx, .xls

required
True

disabled
False

hidden
False

Output results table

label
Uploaded table

type
`basic:file`

required
True

disabled
False

hidden
False

n_samples

label
Number of samples

type
`basic:integer`

required
True

disabled
False

hidden
False

Metadata table (one-to-one)

`data:metadata:unique:upload-metadata-unique` (*basic:file* **src**) [Source: v1.1.1]

Upload metadata file where each row corresponds to a single sample.

The uploaded metadata table represents one-to-one (1:1) relation to samples in the working collection. Metadata table must contain a column with one of the following headers: “Sample ID”, “Sample name” or “Sample slug”.

Input arguments src

label
Table with metadata

type
`basic:file`

description
The metadata table should use one of the following extensions: .csv, .tab, .tsv, .xlsx, .xls

required
True

disabled
False

hidden
False

Output results table

label
Uploaded table

type
`basic:file`

required
True

disabled
False

hidden
False

n_samples

label
Number of samples

type
`basic:integer`

required
True

disabled
False

hidden
False

MultiQC

data:multiqc:multiqc (*list:data:* **data**, *basic:boolean* **dirs**, *basic:integer* **dirs_depth**, *basic:boolean* **fullnames**, *basic:boolean* **config**, *basic:string* **cl_config**) [Source: v1.22.0]

Aggregate results from bioinformatics analyses across many samples into a single report.

[MultiQC](http://www.multiqc.info) searches a given directory for analysis logs and compiles a HTML report. It's a general purpose tool, perfect for summarising the output from numerous bioinformatics tools.

Input arguments data

label
Input data

type
`list:data:`

required
True

disabled
False

hidden
False

advanced.dirs

label
–dirs

type
`basic:boolean`

description
Prepend directory to sample names.

required
True

disabled
False

hidden
False

default
`True`

advanced.dirs_depth

label
–dirs-depth

type
`basic:integer`

description
Prepend a specified number of directories to sample names. Enter a negative number (default) to take from start of path.

required
True

disabled
False

hidden
False

default
`-1`

advanced.fullnames

label
–fullnames

type
`basic:boolean`

description
Disable the sample name cleaning (leave as full file name).

required
True

disabled
False

hidden
False

default
`False`

advanced.config

label
Use configuration file

type
`basic:boolean`

description
Use Genialis configuration file for MultiQC report.

required
True

disabled
False

hidden
False

default
`True`

advanced.cl_config

label
–cl-config

type
`basic:string`

description
Enter text with command-line configuration options to override the defaults (e.g. custom_logo_url: https://www.genialis.com).

required
False

disabled
False

hidden
False

Output results report

label
MultiQC report

type
`basic:file:html`

required
True

disabled
False

hidden
False

report_data

label
Report data

type
`basic:dir`

required
True

disabled
False

hidden
False

OBO file

`data:ontology:oboupload-obo` (*basic:file* **src**)[Source: v1.4.0]

Upload gene ontology in OBO format.

Input arguments src

label
Gene ontology (OBO)

type
`basic:file`

description
Gene ontology in OBO format.

required
True

validate_regex
`\.obo(|\.gz|\.bz2|\.tgz|\.tar\.gz|\.tar\.bz2|\.zip|\.rar|\.7z)$`

Output results obo

label
Ontology file

type
`basic:file`

obo_obj

label
OBO object

type
`basic:file`

PCA

data:pcapca (*list:data:expression* **exps**, *list:basic:string* **genes**, *basic:string* **source**, *basic:string* **species**) [Source: v2.4.2]

Principal component analysis (PCA)

Input arguments exps

label
Expressions

type
`list:data:expression`

genes

label
Gene subset

type
`list:basic:string`

required
False

source

label
Gene ID database of selected genes

type
`basic:string`

description
This field is required if gene subset is set.

required
False

species

label
Species

type
`basic:string`

description
Species latin name. This field is required if gene subset is set.

required
False

choices

- Homo sapiens: `Homo sapiens`
- Mus musculus: `Mus musculus`
- Rattus norvegicus: `Rattus norvegicus`
- Dictyostelium discoideum: `Dictyostelium discoideum`
- Odocoileus virginianus texanus: `Odocoileus virginianus texanus`

- Solanum tuberosum: `Solanum tuberosum`

Output results pca

label
PCA

type
`basic:json`

Picard AlignmentSummary

`data:picard:summary:alignment-summary` (*data:alignment:bam* **bam**, *data:seq:nucleotide* **genome**, *data:seq:nucleotide* **adapters**, *basic:string* **validation_stringency**, *basic:integer* **insert_size**, *basic:string* **pair_orientation**, *basic:boolean* **bisulfite**, *basic:boolean* **assume_sorted**)[Source: v2.3.0]

Produce a summary of alignment metrics from BAM file.

Tool from Picard, wrapped by GATK4. See GATK CollectAlignmentSummaryMetrics for more information.

Input arguments bam

label
Alignment BAM file

type
`data:alignment:bam`

required
True

disabled
False

hidden
False

genome

label
Genome

type
`data:seq:nucleotide`

required
True

disabled
False

hidden
False

adapters

label
Adapter sequences

type
`data:seq:nucleotide`

required
False

disabled
False

hidden
False

validation_stringency

label
Validation stringency

type
`basic:string`

description
Validation stringency for all SAM files read by this program. Setting stringency to SILENT can improve performance when processing a BAM file in which variable-length data (read, qualities, tags) do not otherwise need to be decoded. Default is STRICT.

required
True

disabled
False

hidden
False

default
`STRICT`

choices

- STRICT: `STRICT`
- LENIENT: `LENIENT`
- SILENT: `SILENT`

insert_size

label
Maximum insert size

type
`basic:integer`

required
True

disabled
False

hidden
False

default
`100000`

pair_orientation

label
Pair orientation

type
`basic:string`

required
True

disabled
False

hidden
False

default
`null`

choices

- Unspecified: `null`
- FR: `FR`
- RF: `RF`
- TANDEM: `TANDEM`

bisulfite

label
BAM file consists of bisulfite sequenced reads

type
`basic:boolean`

required
True

disabled
False

hidden
False

default
`False`

assume_sorted

label
Sorted BAM file

type
`basic:boolean`

description
If true the sort order in the header file will be ignored.

required
True

disabled
False

hidden
False

default
`False`

Output results report

label
Alignement metrics

type
`basic:file`

required
True

disabled
False

hidden
False

species

label
Species

type
`basic:string`

required
True

disabled
False

hidden
False

build

label
Build

type
`basic:string`

required
True

disabled
False

hidden
False

Picard CollectRrbsMetrics

`data:picard:rrbs:rrbs-metrics` (*data:alignment:bam* **bam**, *data:seq:nucleotide* **genome**, *basic:integer* **min_quality**, *basic:integer* **next_base_quality**, *basic:integer* **min_lenght**, *basic:decimal* **mismatch_rate**, *basic:string* **validation_stringency**, *basic:boolean* **assume_sorted**)[Source: v2.3.0]

Produce metrics for RRBS data based on the methylation status.

This tool uses reduced representation bisulfite sequencing (Rrbs) data to determine cytosine methylation status across all reads of a genomic DNA sequence.

Tool is wrapped by GATK4. See GATK CollectRrbsMetrics for more information.

Input arguments bam

label
Alignment BAM file

type
`data:alignment:bam`

required
True

disabled
False

hidden
False

genome

label
Genome

type
`data:seq:nucleotide`

required
True

disabled
False

hidden
False

min_quality

label
Threshold for base quality of a C base before it is considered

type
`basic:integer`

required
True

disabled
False

hidden
False

default
`20`

next_base_quality

label
Threshold for quality of a base next to a C before the C base is considered

type
`basic:integer`

required
True

disabled
False

hidden
False

default
`10`

min_lenght

label
Minimum read length

type
`basic:integer`

required
True

disabled
False

hidden
False

default
`5`

mismatch_rate

label
Maximum fraction of mismatches in a read to be considered (Range: 0 and 1)

type
`basic:decimal`

required
True

disabled
False

hidden
False

default
`0.1`

validation_stringency

label
Validation stringency

type
`basic:string`

description
Validation stringency for all SAM files read by this program. Setting stringency to SILENT can improve performance when processing a BAM file in which variable-length data (read, qualities, tags) do not otherwise need to be decoded. Default is STRICT.

required
True

disabled
False

hidden
False

default
`STRICT`

choices

- STRICT: `STRICT`
- LENIENT: `LENIENT`
- SILENT: `SILENT`

assume_sorted

label
Sorted BAM file

type
`basic:boolean`

description
If true the sort order in the header file will be ignored.

required
True

disabled
False

hidden
False

default
`False`

Output results report

label
RRBS summary metrics

type
`basic:file`

required
True

disabled
False

hidden
False

detailed_report

label
Detailed RRBS report

type
`basic:file`

required
True

disabled
False

hidden
False

plot

label
QC plots

type
`basic:file`

required
True

disabled
False

hidden
False

species

label
Species

type
`basic:string`

required
True

disabled
False

hidden
False

build

label
Build

type
`basic:string`

required
True

disabled
False

hidden
False

Picard InsertSizeMetrics

`data:picard:insert:insert-size` (*data:alignment:bam* **bam**, *data:seq:nucleotide* **genome**, *basic:decimal* **minimum_fraction**, *basic:boolean* **include_duplicates**, *basic:decimal* **deviations**, *basic:string* **validation_stringency**, *basic:boolean* **assume_sorted**) [Source: v2.3.0]

Collect metrics about the insert size of a paired-end library.

Tool from Picard, wrapped by GATK4. See GATK CollectInsertSizeMetrics for more information.

Input arguments bam

label
Alignment BAM file

type
`data:alignment:bam`

required
True

disabled
False

hidden
False

genome

label
Genome

type
`data:seq:nucleotide`

required
True

disabled
False

hidden
False

minimum_fraction

label
Minimum fraction of reads in a category to be considered

type
`basic:decimal`

description
When generating the histogram, discard any data categories (out of FR, TANDEM, RF) that have fewer than this fraction of overall reads (Range: 0 and 0.5).

required
True

disabled
False

hidden
False

default
`0.05`

include_duplicates

label
Include reads marked as duplicates in the insert size histogram

type
`basic:boolean`

required
True

disabled
False

hidden
False

default
`False`

deviations

label
Deviations limit

type
`basic:decimal`

description
Generate mean, standard deviation and plots by trimming the data down to MEDIAN + DEVIATIONS*MEDIAN_ABSOLUTE_DEVIATION. This is done because insert size data typically includes enough anomalous values from chimeras and other artifacts to make the mean and standard deviation grossly misleading regarding the real distribution.

required
True

disabled
False

hidden
False

default
`10.0`

validation_stringency

label
Validation stringency

type
`basic:string`

description
Validation stringency for all SAM files read by this program. Setting stringency to SILENT can improve performance when processing a BAM file in which variable-length data (read, qualities, tags) do not otherwise need to be decoded. Default is STRICT.

required
True

disabled
False

hidden
False

default
`STRICT`

choices

- STRICT: `STRICT`
- LENIENT: `LENIENT`
- SILENT: `SILENT`

assume_sorted

label
Sorted BAM file

type
`basic:boolean`

description
If True, the sort order in the header file will be ignored.

required
True

disabled
False

hidden
False

default
`False`

Output results report

label
Insert size metrics

type
`basic:file`

required
True

disabled
False

hidden
False

plot

label
Insert size histogram

type
`basic:file`

required
True

disabled
False

hidden
False

species

label
Species

type
`basic:string`

required
True

disabled
False

hidden
False

build

label
Build

type
`basic:string`

required
True

disabled
False

hidden
False

Picard WGS Metrics

`data:picard:wgsmetrics:wgs-metrics` (*data:alignment:bam* **bam**, *data:seq:nucleotide* **genome**, *basic:integer* **read_length**, *basic:boolean* **create_histogram**, *basic:integer* **min_map_quality**, *basic:integer* **min_quality**, *basic:integer* **coverage_cap**, *basic:integer* **accumulation_cap**, *basic:boolean* **count_unpaired**, *basic:integer* **sample_size**, *basic:string* **validation_stringency**) [Source: v2.4.0]

Collect metrics about coverage of whole genome sequencing.

Tool from Picard, wrapped by GATK4. See GATK CollectWgsMetrics for more information.

Input arguments bam

label
Alignment BAM file

type
`data:alignment:bam`

required
True

disabled
False

hidden
False

genome

label
Genome

type
`data:seq:nucleotide`

required
True

disabled
False

hidden
False

read_length

label
Average read length

type
`basic:integer`

required
True

disabled
False

hidden
False

default
`150`

create_histogram

label
Include data for base quality histogram in the metrics file

type
`basic:boolean`

required
True

disabled
False

hidden
False

default
`False`

options.min_map_quality

label
Minimum mapping quality for a read to contribute coverage

type
`basic:integer`

required
True

disabled
False

hidden
False

default
`20`

options.min_quality

label
Minimum base quality for a base to contribute coverage

type
`basic:integer`

description
N bases will be treated as having a base quality of negative infinity and will therefore be excluded from coverage regardless of the value of this parameter.

required
True

disabled
False

hidden
False

default
`20`

options.coverage_cap

label
Maximum coverage cap

type
`basic:integer`

description
Treat positions with coverage exceeding this value as if they had coverage at this set value.

required
True

disabled
False

hidden
False

default
`250`

options.accumulation_cap

label
Ignore positions with coverage above this value

type
`basic:integer`

description
At positions with coverage exceeding this value, completely ignore reads that accumulate beyond this value

required
True

disabled
False

hidden
False

default
`100000`

options.count_unpaired

label
Count unpaired reads and paired reads with one end unmapped

type
`basic:boolean`

required
True

disabled
False

hidden
False

default
`False`

options.sample_size

label
Sample Size used for Theoretical Het Sensitivity sampling

type
`basic:integer`

required
True

disabled
False

hidden
False

default
`10000`

options.validation_stringency

label
Validation stringency

type
`basic:string`

description
Validation stringency for all SAM files read by this program. Setting stringency to SILENT can improve performance when processing a BAM file in which variable-length data (read, qualities, tags) do not otherwise need to be decoded. Default is STRICT.

required
True

disabled
False

hidden
False

default
`STRICT`

choices

- STRICT: `STRICT`
- LENIENT: `LENIENT`
- SILENT: `SILENT`

Output results report

label
WGS metrics report

type
`basic:file`

required
True

disabled
False

hidden
False

species

label
Species

type
`basic:string`

required
True

disabled
False

hidden
False

build

label
Build

type
`basic:string`

required
True

disabled
False

hidden
False

Pre-peakcall QC

`data:prepeakqcqc-prepeak` (*data:alignment:bam* **alignment**, *basic:integer* **q_treshold**, *basic:integer* **n_sub**, *basic:boolean* **tn5**, *basic:integer* **shift**) [Source: v0.5.2]

ChIP-Seq and ATAC-Seq QC metrics. Process returns a QC metrics report, fragment length estimation, and a deduplicated tagAlign file. Both fragment length estimation and the tagAlign file can be used as inputs in MACS 2.0. QC report contains ENCODE 3 proposed QC metrics – [NRF, PBC bottlenecking coefficients](https://www.encodeproject.org/data-standards/terms/), [NSC, and RSC](https://genome.ucsc.edu/ENCODE/qualityMetrics.html#chipSeq).

Input arguments alignment

label
Aligned reads

type
`data:alignment:bam`

q_treshold

label
Quality filtering treshold

type
`basic:integer`

default
`30`

n_sub

label
Number of reads to subsample

type
`basic:integer`

default
`15000000`

tn5

label
Tn5 shifting

type
`basic:boolean`

description
Tn5 transposon shifting. Shift reads on "+" strand by 4bp and reads on "-" strand by 5bp.

default
`False`

shift

label
User-defined cross-correlation peak strandshift

type
`basic:integer`

description
If defined, SPP tool will not try to estimate fragment length but will use the given value as fragment length.

required
False

Output results chip_qc

label
QC report

type
`basic:file`

tagalign

label
Filtered tagAlign

type
`basic:file`

fraglen

label
Fragnment length

type
`basic:integer`

species

label
Species

type
`basic:string`

build

label
Build

type
`basic:string`

Prepare GEO - ChIP-Seq

`data:other:geo:chipseqprepare-geo-chipseq` (*list:data:reads:fastq* **reads**, *list:data:chipseq:callpeak* **macs**, *basic:string* **name**) [Source: v2.1.3]

Prepare ChIP-seq data for GEO upload.

Input arguments reads

label
Reads

type
`list:data:reads:fastq`

description
List of reads objects. Fastq files will be used.

macs

label
MACS

type
`list:data:chipseq:callpeak`

description
List of MACS2 or MACS14 objects. BedGraph (MACS2) or Wiggle (MACS14) files will be used.

name

label
Collection name

type
`basic:string`

Output results tarball

label
GEO folder

type
`basic:file`

table

label
Annotation table

type
`basic:file`

Prepare GEO - RNA-Seq

`data:other:geo:rnaseqprepare-geo-rnaseq` (*list:data:reads:fastq* **reads**, *list:data:expression* **expressions**, *basic:string* **name**) [Source: v0.2.3]

Prepare RNA-Seq data for GEO upload.

Input arguments reads

label
Reads

type
`list:data:reads:fastq`

description
List of reads objects. Fastq files will be used.

expressions

label
Expressions

type
`list:data:expression`

description
Cuffnorm data object. Expression table will be used.

name

label
Collection name

type
`basic:string`

Output results tarball

label
GEO folder

type
`basic:file`

table

label
Annotation table

type
`basic:file`

QoRTs QC

`data:qorts:qc:qorts-qc` (*data:alignment:bam* **alignment**, *data:annotation:gtf* **annotation**, *basic:string* **stranded**, *data:index:salmon* **cdna_index**, *basic:integer* **n_reads**, *basic:integer* **maxPhredScore**, *basic:integer* **adjustPhredScore**) [Source: v1.8.0]

QoRTs QC analysis.

Input arguments alignment

label
Alignment

type
`data:alignment:bam`

required
True

disabled
False

hidden
False

annotation

label
GTF annotation

type
`data:annotation:gtf`

required
True

disabled
False

hidden
False

options.stranded

label
Assay type

type
`basic:string`

required
True

disabled
False

hidden
False

default
`non_specific`

choices

- Strand non-specific: `non_specific`
- Strand-specific forward: `forward`
- Strand-specific reverse: `reverse`
- Detect automatically: `auto`

options.cdna_index

label
cDNA index file

type
`data:index:salmon`

required
False

disabled
False

hidden
options.stranded != 'auto'

options.n_reads

label
Number of reads in subsampled alignment file

type
`basic:integer`

required
True

disabled
False

hidden
options.stranded != 'auto'

default
`5000000`

options.maxPhredScore

label
Max Phred Score

type
`basic:integer`

required
False

disabled
False

hidden
False

options.adjustPhredScore

label
Adjust Phred Score

type
`basic:integer`

required
False

disabled
False

hidden
False

Output results plot

label
QC multiplot

type
`basic:file`

required
False

disabled
False

hidden
False

summary

label
QC summary

type
`basic:file`

required
True

disabled
False

hidden
False

qorts_data

label
QoRTs report data

type
`basic:file`

required
True

disabled
False

hidden
False

QuantSeq workflow

`data:workflow:quant:featurecounts:workflow-quantseq` (*basic:string* **trimming_tool**, *data:reads:fastq* **reads**, *data:index:star* **genome**, *list:data:seq:nucleotide* **adapters**, *data:annotation* **annotation**, *basic:string* **assay_type**, *data:index:star* **rrna_reference**, *data:index:star* **globin_reference**, *basic:integer* **quality_cutoff**, *basic:integer* **n_reads**, *basic:integer* **seed**, *basic:decimal* **fraction**, *basic:boolean* **two_pass**, *basic:string* **quality_encoding_offset**, *basic:boolean* **ignore_bad_quality**) [Source: v5.1.0]

3' mRNA-Seq pipeline.

Reads are preprocessed by __BBDuk__ or __Cutadapt__ which removes adapters, trims reads for quality from the 3'-end, and discards reads that are too short after trimming. Preprocessed reads are aligned by __STAR__ aligner. For read-count quantification, the __FeatureCounts__ tool is used. QoRTs QC and Samtools idxstats tools are used to report alignment QC metrics.

QC steps include downsampling, QoRTs QC analysis and alignment of input reads to the rRNA/globin reference sequences. The reported alignment rate is used to assess the rRNA/globin sequence depletion rate.

Input arguments trimming_tool

label
Trimming tool

type
`basic:string`

description
Select the trimming tool. If you select BBDuk then please provide adapter sequences in fasta file(s). If you select Cutadapt as a trimming tool, pre-determined adapter sequences will be removed.

required
True

disabled
False

hidden
False

choices

- BBDuk: `bbduk`
- Cutadapt: `cutadapt`

reads

label
Input reads (FASTQ)

type
`data:reads:fastq`

description
Reads in FASTQ file, single or paired end.

required
True

disabled
False

hidden
False

genome

label
Indexed reference genome

type
`data:index:star`

description
Genome index prepared by STAR aligner indexing tool.

required
True

disabled
False

hidden
False

adapters

label
Adapters

type
`list:data:seq:nucleotide`

description
Provide a list of sequencing adapters files (.fasta) to be removed by BBDuk.

required
False

disabled
False

hidden
trimming_tool != 'bbduk'

annotation

label
Annotation

type
`data:annotation`

description
GTF and GFF3 annotation formats are supported.

required
True

disabled
False

hidden
False

assay_type

label
Assay type

type
`basic:string`

description
In strand-specific forward assay and single reads, the read has to be mapped to the same strand as the feature. For paired-end reads, the first read has to be on the same strand and the second read on the opposite strand. In strand-specific reverse assay these rules are reversed.

required
False

disabled
False

hidden
False

choices

- Strand-specific forward: `forward`
- Strand-specific reverse: `reverse`

rrna_reference

label
Indexed rRNA reference sequence

type
`data:index:star`

description
Reference sequence index prepared by STAR aligner indexing tool.

required
False

disabled
False

hidden
False

globin_reference

label
Indexed Globin reference sequence

type
`data:index:star`

description
Reference sequence index prepared by STAR aligner indexing tool.

required
False

disabled
False

hidden
False

cutadapt.quality_cutoff

label
Reads quality cutoff

type
`basic:integer`

description
Trim low-quality bases from 3' end of each read before adapter removal. The use of this option will override the use of NextSeq/NovaSeq-specific trim option.

required
False

disabled
False

hidden
False

downsampling.n_reads

label
Number of reads

type
`basic:integer`

description
Number of reads to include in subsampling.

required
True

disabled
False

hidden
False

default
`1000000`

downsampling.advanced.seed

label
Number of reads

type
`basic:integer`

description
Using the same random seed makes reads subsampling reproducible in different environments.

required
True

disabled
False

hidden
False

default
11

downsampling.advanced.fraction

label
Fraction

type
`basic:decimal`

description
Use the fraction of reads [0 - 1.0] from the orignal input file instead of the absolute number of reads. If set, this will override the'Number of reads' input parameter.

required
False

disabled
False

hidden
False

downsampling.advanced.two_pass

label
2-pass mode

type
`basic:boolean`

description
Enable two-pass mode when down-sampling. Two-pass mode is twice as slow but with much reduced memory.

required
True

disabled
False

hidden
False

default
`False`

preprocessing.quality_encoding_offset

label
Quality encoding offset

type
`basic:string`

description
Quality encoding offset for input FASTQ files.

required
True

disabled
False

hidden
False

default
`auto`

choices

- Sanger / Illumina 1.8+: `33`
- Illumina up to 1.3+, 1.5+: `64`
- Auto: `auto`

preprocessing.ignore_bad_quality

label
Ignore bad quality

type
`basic:boolean`

description
Don't crash if quality values appear to be incorrect.

required
True

disabled
False

hidden
False

default
`False`

Output results

Quantify shRNA species using bowtie2

data:expression:shrna2quantshrna-quant (*data:alignment:bam* **alignment**, *basic:integer* **readlengths**, *basic:integer* **alignscores**) [Source: v1.4.0]

Based on `bowtie2` output (.bam file) calculate number of mapped species. Input is limited to results from `bowtie2` since `YT:Z:` tag used to fetch aligned species is specific to this process. Result is a count matrix (successfully mapped reads) where species are in rows columns contain read specifics (count, species name, sequence, `AS:i:` tag value).

Input arguments alignment

label
Alignment

type
`data:alignment:bam`

required
True

readlengths

label
Species lengths threshold

type
`basic:integer`

description
Species with read lengths below specified threshold will be removed from final output. Default is no removal.

alignscores

label
Align scores filter threshold

type
`basic:integer`

description
Species with align score below specified threshold will be removed from final output. Default is no removal.

Output results exp

label
Normalized expression

type
`basic:file`

rc

label
Read counts

type
`basic:file`

required
False

exp_json

label
Expression (json)

type
`basic:json`

exp_type

label
Expression type

type
`basic:string`

source

label
Gene ID source

type
`basic:string`

species

label
Species

type
`basic:string`

build

label
Build

type
`basic:string`

feature_type

label
Feature type

type
`basic:string`

mapped_species

label
Mapped species

type
`basic:file`

RNA-SeQC

data:rnaseqc:qc:rnaseqc-qc (*data:alignment:bam* **alignment**, *data:annotation:gtf* **annotation**, *basic:integer* **mapping_quality**, *basic:integer* **base_mismatch**, *basic:integer* **offset**, *basic:integer* **window_size**, *basic:integer* **gene_length**, *basic:integer* **detection_threshold**, *basic:boolean* **exclude_chimeric**, *basic:string* **stranded**, *data:index:salmon* **cdna_index**, *basic:integer* **n_reads**) [Source: v2.0.0]

RNA-SeQC QC analysis.

An efficient new version of RNA-SeQC that computes a comprehensive set of metrics for characterizing samples processed by a wide range of protocols. It also quantifies gene- and exon-level expression, enabling effective quality control of large-scale RNA-seq datasets.

More information can be found in the [GitHub repository](https://github.com/getzlab/rnaseqc) and in the [original paper](https://academic.oup.com/bioinformatics/article/37/18/3048/6156810?login=false).

Input arguments alignment

label
Input aligned reads (BAM file)

type
`data:alignment:bam`

required
True

disabled
False

hidden
False

annotation

label
Annotation file (GTF)

type
`data:annotation:gtf`

description
The input GTF file containing features to check the bam against. The file should include gene_id in the attributes column for all entries. During the process the file is formatted so the transcript_id matches the gene_id. Exons are merged to remove overlaps and exon_id field is then matched with gene_id including the consecutive exon number.

required
True

disabled
False

hidden
False

rnaseqc_options.mapping_quality

label
Mapping quality [–mapping-quality]

type
basic:integer

description
Set the lower bound on read quality for exon coverage counting. Reads below this number are excluded from coverage metrics.

required
True

disabled
False

hidden
False

default
255

rnaseqc_options.base_mismatch

label
Base mismatch [–base-mismatch]

type
basic:integer

description
Set the maximum number of allowed mismatches between a read and the reference sequence. Reads with more than this number of mismatches are excluded from coverage metrics.

required
True

disabled
False

hidden
False

default
6

rnaseqc_options.offset

label
Offset [–offset]

type
basic:integer

description
Set the offset into the gene for the 3' and 5' windows in bias calculation. A positive value shifts the 3' and 5' windows towards each other, while a negative value shifts them apart.

required
True

disabled
False

hidden
False

default
150

rnaseqc_options.window_size

label
Window size [–window-size]

type
basic:integer

description
Set the offset into the gene for the 3’ and 5’ windows in bias calculation.

required
True

disabled
False

hidden
False

default
100

rnaseqc_options.gene_length

label
Window size [–gene-length]

type
basic:integer

description
Set the minimum size of a gene for bias calculation. Genes below this size are ignored in the calculation.

required
True

disabled
False

hidden
False

default
600

rnaseqc_options.detection_threshold

label
Detection threshold [–detection-threshold]

type
basic:integer

description
Number of counts on a gene to consider the gene ‘detected’. Additionally, genes below this limit are excluded from 3’ bias computation.

required
True

disabled
False

hidden
False

default
`5`

rnaseqc_options.exclude_chimeric

label
Exclude chimeric reads [–exclude-chimeric]

type
`basic:boolean`

description
Exclude chimeric reads from the read counts.

required
True

disabled
False

hidden
False

default
`False`

strand_detection_options.stranded

label
Assay type [–stranded]

type
`basic:string`

required
True

disabled
False

hidden
False

default
`non_specific`

choices

- Strand non-specific: `non_specific`
- Strand-specific reverse then forward: `reverse`
- Strand-specific forward then reverse: `forward`
- Detect automatically: `auto`

strand_detection_options.cdna_index

label
cDNA index file

type
`data:index:salmon`

required
False

disabled
False

hidden
strand_detection_options.stranded != 'auto'

strand_detection_options.n_reads

label
Number of reads in subsampled alignment file. Subsampled reads will be used in strandedness detection

type
`basic:integer`

required
True

disabled
False

hidden
strand_detection_options.stranded != 'auto'

default
`5000000`

Output results metrics

label
metrics

type
`basic:file`

required
True

disabled
False

hidden
False

RNA-Seq (Cuffquant)

`data:workflow:rnaseq:cuffquantworkflow-rnaseq-cuffquant` (*data:reads:fastq* **reads**, *data:index:hisat2* **genome**, *data:annotation* **annotation**)[Source: v2.1.0]

Input arguments reads

label
Input reads

type
`data:reads:fastq`

genome

label
genome

type
`data:index:hisat2`

annotation

label
Annotation file

type
`data:annotation`

Output results

RNA-seq Variant Calling Workflow

`data:workflow:rnaseq:variants:workflow-rnaseq-variantcalling` (*data:alignment:bam:star* **bam**, *data:reads:fastq* **reads**, *basic:boolean* **preprocessing**, *data:seq:nucleotide* **ref_seq**, *data:index:star* **genome**, *data:variants:vcf* **dbsnp**, *list:data:variants:vcf* **indels**, *data:bed* **intervals**, *data:variants:vcf* **clinvar**, *data:geneset* **geneset**, *list:basic:string* **mutations**, *list:data:seq:nucleotide* **adapters**, *list:basic:string* **custom_adapter_sequences**, *basic:integer* **kmer_length**, *basic:integer* **min_k**, *basic:integer* **hamming_distance**, *basic:integer* **maxns**, *basic:integer* **trim_quality**, *basic:integer* **min_length**, *basic:string* **quality_encoding_offset**, *basic:boolean* **ignore_bad_quality**, *basic:boolean* **two_pass_mode**, *basic:boolean* **out_unmapped**, *basic:string* **align_end_alignment**, *basic:string* **read_group**, *basic:integer* **stand_call_conf**, *basic:boolean* **soft_clipped**, *basic:integer* **interval_padding**, *list:basic:string* **filter_expressions**, *list:basic:string* **filter_name**, *list:basic:string* **genotype_filter_expressions**, *list:basic:string* **genotype_filter_name**, *data:variants:vcf* **mask**, *basic:string* **mask_name**, *basic:string* **filtering_options**, *list:basic:string* **vcf_fields**, *list:basic:string* **ann_fields**, *basic:boolean* **split_alleles**, *basic:boolean* **show_filtered**, *list:basic:string* **gf_fields**, *basic:boolean* **multiqc**, *basic:integer* **java_gc_threads**, *basic:integer* **max_heap_size**) [Source: v2.4.0]

Identify variants in RNA-seq data.

This pipeline follows GATK best practices recommendantions for variant calling with RNA-seq data.

The pipeline steps include read alignment (STAR), data cleanup (MarkDuplicates), splitting reads that contain Ns in their cigar string (SplitNCigarReads), base quality recalibration (BaseRecalibrator, ApplyBQSR), variant calling (HaplotypeCaller), variant filtering (VariantFiltration) and variant annotation (SnpEff). The last step of the pipeline is process Mutations table which prepares variants for ReSDK VariantTables.

There is also possibility to run the pipeline directly from BAM file. In this case, it is recommended that you use two-pass mode in STAR alignment as well as turn the option '–outSAMunmapped Within' on.

Input arguments bam

label
Input BAM file

type
`data:alignment:bam:star`

description
Input BAM file that was computed with STAR aligner. It is highly recommended that two-pass mode was used for the alignment as well as '–outSAMunmapped Within' option if you want to use BAM file as an input.

required
False

disabled
reads

hidden
False

reads

label
Input sample (FASTQ)

type
`data:reads:fastq`

description
Input data in FASTQ format.

required
False

disabled
bam

hidden
False

preprocessing

label
Perform reads processing with BBDuk

type
`basic:boolean`

description
If your reads have not been processed, set this to True.

required
True

disabled
bam

hidden
False

default
`True`

ref_seq

label
Reference FASTA sequence

type
`data:seq:nucleotide`

required
True

disabled
False

hidden
False

genome

label
Indexed reference genome

type
`data:index:star`

description
Genome index prepared by STAR aligner indexing tool.

required
False

disabled
bam

hidden
False

dbsnp

label
dbSNP file

type
`data:variants:vcf`

description
File with known variants.

required
True

disabled
False

hidden
False

indels

label
Known INDEL sites

type
`list:data:variants:vcf`

required
False

disabled
False

hidden
False

intervals

label
Intervals (from BED file)

type
`data:bed`

description
Use this option to perform the analysis over only part of the genome.

required
False

disabled
False

hidden
False

clinvar

label
ClinVar VCF file

type
`data:variants:vcf`

description
[ClinVar](https://www.ncbi.nlm.nih.gov/clinvar/) is a freely available, public archive of human genetic variants and interpretations of their significance to disease.

required
False

disabled
False

hidden
False

geneset

label
Gene set

type
`data:geneset`

description
Select a gene set with genes you are interested in. Only variants of genes in the selected gene set will be in the output.

required
False

disabled
mutations

hidden
False

mutations

label
Gene and its mutations

type
`list:basic:string`

description
Insert the gene you are interested in, together with mutations. First enter the name of the gene and then the mutations. Seperate gene from mutations with ':' and mutations with ','. Example of an input: 'KRAS: Gly12, Gly61'. Press enter after each input (gene + mutations). NOTE: Field only accepts three character amino acid symbols. If you use this option, the selected geneset will not be used for Mutations table process.

required
False

disabled
geneset

hidden
False

bbduk.adapters

label
Adapters

type
`list:data:seq:nucleotide`

description
Provide a list of sequencing adapters files (.fasta) to be removed by BBDuk.

required
False

disabled
False

hidden
False

bbduk.custom_adapter_sequences

label
Custom adapter sequences

type
`list:basic:string`

description
Custom adapter sequences can be specified by inputting them one by one and pressing Enter after each sequence.

required
False

disabled
False

hidden
False

default
`[]`

bbduk.kmer_length

label
K-mer length [k=]

type
`basic:integer`

description
Kmer length used for finding contaminants. Contaminants shorter than kmer length will not be found. Kmer length must be at least 1.

required
True

disabled
False

hidden
False

default
`23`

bbduk.min_k

label
Minimum k-mer length at right end of reads used for trimming [mink=]

type
`basic:integer`

required
True

disabled
bbduk.adapters.length === 0 && bbduk.custom_adapter_sequences.length === 0

hidden
False

default
`11`

bbduk.hamming_distance

label
Maximum Hamming distance for k-mers [hammingdistance=]

type
`basic:integer`

description
Hamming distance i.e. the number of mismatches allowed in the kmer.

required
True

disabled
False

hidden
False

default
`1`

bbduk.maxns

label
Max Ns after trimming [maxns=]

type
`basic:integer`

description
If non-negative, reads with more Ns than this (after trimming) will be discarded.

required
True

disabled
False

hidden
False

default
`-1`

bbduk.trim_quality

label
Average quality below which to trim region [trimq=]

type
`basic:integer`

description
Phred algorithm is used, which is more accurate than naive trimming.

required
True

disabled
False

hidden
False

default
`28`

bbduk.min_length

label
Minimum read length [minlength=]

type
`basic:integer`

description
Reads shorter than minimum read length after trimming are discarded.

required
True

disabled
False

hidden
False

default
`30`

bbduk.quality_encoding_offset

label
Quality encoding offset

type
`basic:string`

description
Quality encoding offset for input FASTQ files.

required
True

disabled
False

hidden
False

default
`auto`

choices

- Sanger / Illumina 1.8+: `33`
- Illumina up to 1.3+, 1.5+: `64`
- Auto: `auto`

bbduk.ignore_bad_quality

label
Ignore bad quality

type
`basic:boolean`

description
Don't crash if quality values appear to be incorrect.

required
True

disabled
False

hidden
False

default
`False`

alignment.two_pass_mode

label
Use two pass mode [–twopassMode]

type
`basic:boolean`

description
Use two-pass maping instead of first-pass only. In two-pass mode we first perform first-pass mapping, extract junctions, insert them into genome index, and re-map all reads in the second mapping pass.

required
True

disabled
False

hidden
False

default
`True`

alignment.out_unmapped

label
Output unmapped reads (SAM) [–outSAMunmapped Within]

type
`basic:boolean`

description
Output of unmapped reads in the SAM format.

required
True

disabled
False

hidden
False

default
`True`

alignment.align_end_alignment

label
Read ends alignment [–alignEndsType]

type
`basic:string`

description
Type of read ends alignment (default: Local). Local: standard local alignment with soft-clipping allowed. EndToEnd: force end-to-end read alignment, do not soft-clip. Extend5pOfRead1: fully extend only the 5p of the read1, all other ends: local alignment. Extend5pOfReads12: fully extend only the 5' of the both read1 and read2, all other ends use local alignment.

required
True

disabled
False

hidden
False

default
`Local`

choices

- Local: `Local`
- EndToEnd: `EndToEnd`
- Extend5pOfRead1: `Extend5pOfRead1`
- Extend5pOfReads12: `Extend5pOfReads12`

bam_processing.read_group

label
Replace read groups in BAM

type
`basic:string`

description
Replace read groups in a BAM file. This argument enables the user to replace all read groups in the INPUT file with a single new read group and assign all reads to this read group in the OUTPUT BAM file. Addition or replacement is performed using Picard's AddOrReplaceReadGroups tool. Input should take the form of -name=value delimited by a ";", e.g. "-ID=1;-LB=GENIALIS;-PL=ILLUMINA;-PU=BARCODE;-SM=SAMPLENAME1". See tool's documentation for more information on tag names. Note that PL, LB, PU and SM are required fields. See caveats of rewriting read groups in the documentation.

required
True

disabled
False

hidden
False

default
`-ID=1;-LB=GENIALIS;-PL=ILLUMINA;-PU=BARCODE;-SM=SAMPLENAME1`

haplotype_caller.stand_call_conf

label
Min call confidence threshold

type
`basic:integer`

description
The minimum phred-scaled confidence threshold at which variants should be called.

required
True

disabled
False

hidden
False

default
`20`

haplotype_caller.soft_clipped

label
Do not analyze soft clipped bases in the reads

type
`basic:boolean`

description
Suitable option for RNA-seq variant calling.

required
True

disabled
False

hidden
False

default
`True`

haplotype_caller.interval_padding

label
Interval padding

type
`basic:integer`

description
Amount of padding (in bp) to add to each interval you are including. The recommended value is 100. Set to 0 if you want to turn it off.

required
True

disabled
False

hidden
!intervals

default
`100`

variant_filtration.filter_expressions

label
Expressions used with INFO fields to filter

type
`list:basic:string`

description
VariantFiltration accepts any number of JEXL expressions (so you can have two named filters by using –filter-name One –filter-expression 'X < 1' –filter-name Two –filter-expression 'X > 2'). It is preferable to use multiple expressions, each specifying an individual filter criteria, to a single compound expression that specifies multiple filter criteria. Input expressions one by one and press ENTER after each expression. Examples of filter expression: 'FS > 30', 'DP > 10'.

required
True

disabled
False

hidden
False

default
`['FS > 30.0', 'QD < 2.0']`

variant_filtration.filter_name

label
Names to use for the list of filters

type
`list:basic:string`

description
This name is put in the FILTER field for variants that get filtered. Note that there must be a 1-to-1 mapping between filter expressions and filter names. Input expressions one by one and press ENTER after each name. Warning: filter names should be in the same order as filter expressions. Example: you specified filter expressions 'FS > 30' and 'DP > 10', now specify filter names 'FS' and 'DP'.

required
True

disabled
False

hidden
False

default
`['FS', 'QD']`

variant_filtration.genotype_filter_expressions

label
Expressions used with FORMAT field to filter

type
`list:basic:string`

description
Similar to the INFO field based expressions, but used on the FORMAT (genotype) fields instead. VariantFiltration will add the sample-level FT tag to the FORMAT field of filtered samples (this does not affect the record's FILTER tag). One can filter normally based on most fields (e.g. 'GQ

< 5.0'), but the GT (genotype) field is an exception. We have put in convenience methods so that one can now filter out hets ('isHet == 1'), refs ('isHomRef == 1'), or homs ('isHomVar == 1'). Also available are expressions isCalled, isNoCall, isMixed, and isAvailable, in accordance with the methods of the Genotype object. To filter by alternative allele depth, use the expression: 'AD.1 < 5'. This filter expression will filter all the samples in the multi-sample VCF file.

required
True

disabled
False

hidden
False

default
`['AD.1 < 5.0']`

variant_filtration.genotype_filter_name

label
Names to use for the list of genotype filters

type
`list:basic:string`

description
Similar to the INFO field based expressions, but used on the FORMAT (genotype) fields instead. Warning: filter names should be in the same order as filter expressions.

required
True

disabled
False

hidden
False

default
`['AD']`

variant_filtration.mask

label
Input mask

type
`data:variants:vcf`

description
Any variant which overlaps entries from the provided mask file will be filtered.

required
False

disabled
False

hidden
False

variant_filtration.mask_name

label
The text to put in the FILTER field if a 'mask' is provided

type
`basic:string`

description
When using the mask file, the mask name will be annotated in the variant record.

required
False

disabled
!variant_filtration.mask

hidden
False

snpeff.filtering_options

label
SnpEff filtering expressions

type
`basic:string`

description
Filter annotated VCF file using arbitraty expressions.Examples of filtering expressions: '(ANN[*].GENE = 'PSD3')' or '(REF = 'A')' or '(countHom() > 3) | ((exists INDEL) & (QUAL >= 20)) | (QUAL >= 30)'.For more information checkout the official documentation of [SnpSift](https://pcingola.github.io/SnpEff/ss_filter/)

required
False

disabled
False

hidden
False

mutations_table.vcf_fields

label
Select VCF fields

type
`list:basic:string`

description
The name of a standard VCF field or an INFO field to include in the output table. The field can be any standard VCF column (e.g. CHROM, ID, QUAL) or any annotation name in the INFO field (e.g. AC, AF). Required fields are CHROM, POS, ID, REF and ANN. If your variants file was annotated with clinvar information then fields CLNDN, CLNSIG and CLNSIGCONF might be of your interest.

required
True

disabled
False

hidden
False

default
`['CHROM', 'POS', 'ID', 'QUAL', 'REF', 'ALT', 'FILTER', 'ANN', 'CLNDN', 'CLNSIG']`

mutations_table.ann_fields

label
ANN fields to use

type
`list:basic:string`

description
Only use specific fields from the SnpEff ANN field. All available fields: Allele | Annotation | Annotation_Impact | Gene_Name | Gene_ID | Feature_Type | Feature_ID | Transcript_BioType | Rank | HGVS.c | HGVS.p | cDNA.pos / cDNA.length | CDS.pos / CDS.length | AA.pos / AA.length | Distance | ERRORS / WARNINGS / INFO' .Fields are seperated by '|'. For more information, follow this [link](https://pcingola.github.io/SnpEff/se_inputoutput/#ann-field-vcf-output-files).

required
True

disabled
False

hidden
False

default
`['Allele', 'Annotation', 'Annotation_Impact', 'Gene_Name', 'Feature_ID', 'HGVS.p']`

mutations_table.split_alleles

label
Split multi-allelic records into multiple lines

type
`basic:boolean`

description
By default, a variant record with multiple ALT alleles will be summarized in one line, with per alt-allele fields (e.g. allele depth) separated by commas.This may cause difficulty when the table is loaded by an R script, for example. Use this flag to write multi-allelic records on separate lines of output.

required
True

disabled
False

hidden
False

default
`True`

mutations_table.show_filtered

label
Include filtered records in the output

type
basic:boolean

description
Include filtered records in the output of the GATK VariantsToTable.

required
True

disabled
False

hidden
False

default
`True`

mutations_table.gf_fields

label
Include FORMAT/sample-level fields. Note: If you specify DP from genotype field, it will overwrite the original DP field. By default fields GT (genotype), AD (allele depth), DP (depth at the sample level), FT (sample-level filter) are included in the analysis.

type
list:basic:string

required
True

disabled
False

hidden
False

default
`['GT', 'AD', 'DP', 'FT']`

advanced.multiqc

label
Trigger MultiQC

type
basic:boolean

description
If the input for the pipeline is BAM file that has been computed by the RNA-seq gene expression pipeline, than MultiQC object already exists for this sample, so there is no need for an additional MultiQC process. If the input for this pipeline is FASTQ, than MultiQC cannot be disabled.

required
True

disabled
False

hidden
!bam

default
`False`

advanced.java_gc_threads

label
Java ParallelGCThreads

type
`basic:integer`

description
Sets the number of threads used during parallel phases of the garbage collectors.

required
True

disabled
False

hidden
False

default
`2`

advanced.max_heap_size

label
Java maximum heap size (Xmx)

type
`basic:integer`

description
Set the maximum Java heap size (in GB).

required
True

disabled
False

hidden
False

default
`12`

Output results

RNA-seq variant calling preprocess

`data:alignment:bam:rnaseqvc:rnaseq-vc-preprocess` (*data:alignment:bam* **bam**, *data:seq:nucleotide* **ref_seq**, *list:data:variants:vcf* **known_sites**, *basic:string* **read_group**, *basic:integer* **java_gc_threads**, *basic:integer* **max_heap_size**) [Source: v1.3.0]

Prepare BAM file from STAR aligner for HaplotypeCaller.

This process includes steps MarkDuplicates, SplitNCigarReads, read-group assignment and base quality recalibration (BQSR).

Input arguments bam

label
Alignment BAM file from STAR alignment

type
`data:alignment:bam`

required
True

disabled
False

hidden
False

ref_seq

label
Reference sequence FASTA file

type
`data:seq:nucleotide`

required
True

disabled
False

hidden
False

known_sites

label
List of known sites of variation

type
`list:data:variants:vcf`

description
One or more databases of known polymorphic sites used to exclude regions around known polymorphisms from analysis.

required
True

disabled
False

hidden
False

read_group

label
Replace read groups in BAM

type
`basic:string`

description
Replace read groups in a BAM file. This argument enables the user to replace all read groups in

the INPUT file with a single new read group and assign all reads to this read group in the OUTPUT BAM file. Addition or replacement is performed using GATK AddOrReplaceReadGroups tool. Input should take the form of -name=value delimited by a ";", e.g. "-ID=1;-LB=GENIALIS;-PL=ILLUMINA;-PU=BARCODE;-SM=SAMPLENAME1". See tool's documentation for more information on tag names. Note that PL, LB, PU and SM are require fields. See caveats of rewriting read groups in the documentation.

required
True

disabled
False

hidden
False

default
`-ID=1;-LB=GENIALIS;-PL=ILLUMINA;-PU=BARCODE;-SM=SAMPLENAME1`

advanced.java_gc_threads

label
Java ParallelGCThreads

type
`basic:integer`

description
Sets the number of threads used during parallel phases of the garbage collectors.

required
True

disabled
False

hidden
False

default
`2`

advanced.max_heap_size

label
Java maximum heap size (Xmx)

type
`basic:integer`

description
Set the maximum Java heap size (in GB).

required
True

disabled
False

hidden
False

default
`12`

Output results bam

label
Preprocessed BAM file

type
`basic:file`

required
True

disabled
False

hidden
False

bai

label
Index of BAM file

type
`basic:file`

required
True

disabled
False

hidden
False

stats

label
Alignment statistics

type
`basic:file`

required
True

disabled
False

hidden
False

species

label
Species

type
`basic:string`

required
True

disabled
False

hidden
False

build

label
Build

type
`basic:string`

required
True

disabled
False

hidden
False

metrics_file

label
Metrics from MarkDuplicate process

type
`basic:file`

required
True

disabled
False

hidden
False

ROSE2

`data:chipseq:rose2:rose2` (*data:chipseq:callpeak* **input_macs**, *data:bed* **input_upload**, *basic:boolean* **use_filtered_bam**, *data:alignment:bam* **rankby**, *data:alignment:bam* **control**, *basic:integer* **tss**, *basic:integer* **stitch**, *data:bed* **mask**) [Source: v5.2.1]

Run ROSE2.

Rank Ordering of Super-Enhancers algorithm (ROSE2) takes the acetylation peaks called by a peak caller (MACS, MACS2...) and based on the in-between distances and the acetylation signal at the peaks judges whether they can be considered super-enhancers. The ranked values are plotted and by locating the inflection point in the resulting graph, super-enhancers are assigned. See [here](http://younglab.wi.mit.edu/super_enhancer_code.html) for more information.

Input arguments input_macs

label
BED/narrowPeak file (MACS results)

type
`data:chipseq:callpeak`

required
False

disabled
False

hidden
input_upload

input_upload

label
BED file (Upload)

type
`data:bed`

required
False

disabled
False

hidden
input_macs || use_filtered_bam

use_filtered_bam

label
Use Filtered BAM File

type
`basic:boolean`

description
Use filtered BAM file from a MACS2 object to rank enhancers by. Only applicable if input is MACS2.

required
True

disabled
False

hidden
input_upload

default
`False`

rankby

label
BAM file

type
`data:alignment:bam`

description
BAM file to rank enhancers by.

required
False

disabled
False

hidden
use_filtered_bam

control

label
Control BAM File

type
`data:alignment:bam`

description
BAM file to rank enhancers by.

required
False

disabled
False

hidden
use_filtered_bam

tss

label
TSS exclusion

type
`basic:integer`

description
Enter a distance from TSS to exclude. 0 = no TSS exclusion.

required
True

disabled
False

hidden
False

default
`0`

stitch

label
Stitch

type
`basic:integer`

description
Enter a max linking distance for stitching. If not given, optimal stitching parameter will be determined automatically.

required
False

disabled
False

hidden
False

mask

label
Masking BED file

type
`data:bed`

description
Mask a set of regions from analysis. Provide a BED of masking regions.

required
False

disabled
False

hidden
False

Output results all_enhancers

label
All enhancers table

type
`basic:file`

required
True

disabled
False

hidden
False

enhancers_with_super

label
Super enhancers table

type
`basic:file`

required
True

disabled
False

hidden
False

plot_points

label
Plot points

type
`basic:file`

required
True

disabled
False

hidden
False

plot_panel

label
Plot panel

type
`basic:file`

required
True

disabled
False

hidden
False

enhancer_gene

label
Enhancer to gene

type
`basic:file`

required
True

disabled
False

hidden
False

enhancer_top_gene

label
Enhancer to top gene

type
`basic:file`

required
True

disabled
False

hidden
False

gene_enhancer

label
Gene to Enhancer

type
`basic:file`

required
True

disabled
False

hidden
False

stitch_parameter

label
Stitch parameter

type
`basic:file`

required
False

disabled
False

hidden
False

all_output

label
All output

type
`basic:file`

required
True

disabled
False

hidden
False

scatter_plot

label
Super-Enhancer plot

type
`basic:json`

required
True

disabled
False

hidden
False

species

label
Species

type
`basic:string`

required
True

disabled
False

hidden
False

build

label
Build

type
`basic:string`

required
True

disabled
False

hidden
False

Reads (QSEQ multiplexed, paired)

`data:multiplexed:qseq:pairedupload-multiplexed-paired` (*basic:file* **reads**, *basic:file* **reads2**, *basic:file* **barcodes**, *basic:file* **annotation**) [Source: v1.4.1]

Upload multiplexed NGS reds in QSEQ format.

Input arguments reads

label
Multiplexed upstream reads

type
`basic:file`

description
NGS reads in QSeq format. Supported extensions: .qseq.txt.bz2 (preferred), .qseq.* or .qseq.txt.*.

required
True

validate_regex
`((\.qseq|\.qseq\.txt)(\.gz|\.bz2|\.tgz|\.tar\.gz|\.tar\.bz2|\.zip|\.rar|\.7z))|(\.bz2)$`

reads2

label
Multiplexed downstream reads

type
`basic:file`

description
NGS reads in QSeq format. Supported extensions: .qseq.txt.bz2 (preferred), .qseq.* or .qseq.txt.*.

required
True

validate_regex
`((\.qseq|\.qseq\.txt)(\.gz|\.bz2|\.tgz|\.tar\.gz|\.tar\.bz2|\.zip|\.rar|\.7z))|(\.bz2)$`

barcodes

label
NGS barcodes

type
`basic:file`

description
Barcodes in QSeq format. Supported extensions: .qseq.txt.bz2 (preferred), .qseq.* or .qseq.txt.*.

required
True

validate_regex
`((\.qseq|\.qseq\.txt)(\.gz|\.bz2|\.tgz|\.tar\.gz|\.tar\.bz2|\.zip|\.rar|\.7z))|(\.bz2)$`

annotation

label
Barcode mapping

type
`basic:file`

description
A tsv file mapping barcodes to experiment name, e.g. "TCGCAGG\tHr00".

required
True

validate_regex
`(\.tsv)$`

Output results qseq_reads

label
Multiplexed upstream reads

type
`basic:file`

qseq_reads2

label
Multiplexed downstream reads

type
`basic:file`

qseq_barcodes

label
NGS barcodes

type
`basic:file`

annotation

label
Barcode mapping

type
`basic:file`

matched

label
Matched

type
`basic:string`

notmatched

label
Not matched

type
`basic:string`

badquality

label
Bad quality

type
`basic:string`

skipped

label
Skipped

type
`basic:string`

Reads (QSEQ multiplexed, single)

`data:multiplexed:qseq:singleupload-multiplexed-single` (*basic:file* **reads**, *basic:file* **barcodes**, *basic:file* **annotation**) [Source: v1.4.1]

Upload multiplexed NGS reds in QSEQ format.

Input arguments reads

label
Multiplexed NGS reads

type
`basic:file`

description
NGS reads in QSeq format. Supported extensions: .qseq.txt.bz2 (preferred), .qseq.* or .qseq.txt.*.

required
True

validate_regex
`(\.(qseq)(|\.txt)(|\.gz|\.bz2|\.tgz|\.tar\.gz|\.tar\.bz2|\.zip|\.rar|\.7z))|(\.bz2)$`

barcodes

label
NGS barcodes

type
`basic:file`

description
Barcodes in QSeq format. Supported extensions: .qseq.txt.bz2 (preferred), .qseq.* or .qseq.txt.*.

required
True

validate_regex
`(\.(qseq)(|\.txt)(|\.gz|\.bz2|\.tgz|\.tar\.gz|\.tar\.bz2|\.zip|\.rar|\.7z))|(\.bz2)$`

annotation

label
Barcode mapping

type
`basic:file`

description
A tsv file mapping barcodes to experiment name, e.g. "TCGCAGG\tHr00".

required
True

validate_regex
`(\.tsv)$`

Output results qseq_reads

label
Multiplexed NGS reads

type
`basic:file`

qseq_barcodes

label
NGS barcodes

type
`basic:file`

annotation

label
Barcode mapping

type
`basic:file`

matched

label
Matched

type
`basic:string`

notmatched

label
Not matched

type
`basic:string`

badquality

label
Bad quality

type
`basic:string`

skipped

label
Skipped

type
`basic:string`

Reads (scRNA 10x)

`data:screads:10x:upload-sc-10x` (*list:basic:file* **barcodes**, *list:basic:file* **reads**) [Source: v1.4.1]

Import 10x scRNA reads in FASTQ format.

Input arguments barcodes

label
Barcodes (.fastq.gz)

type
`list:basic:file`

required
True

disabled
False

hidden
False

reads

label
Reads (.fastq.gz)

type
`list:basic:file`

required
True

disabled
False

hidden
False

Output results barcodes

label
Barcodes

type
`list:basic:file`

required
True

disabled
False

hidden
False

reads

label
Reads

type
`list:basic:file`

required
True

disabled
False

hidden
False

fastqc_url_barcodes

label
Quality control with FastQC (Barcodes)

type
`list:basic:file:html`

required
True

disabled
False

hidden
False

fastqc_url_reads

label
Quality control with FastQC (Reads)

type
`list:basic:file:html`

required
True

disabled
False

hidden
False

Reverse complement FASTQ (paired-end)

`data:reads:fastq:paired:seqtk:seqtk-rev-complement-paired` (*data:reads:fastq:paired* **reads**, *basic:string* **select_mate**) [Source: v1.2.2]

Reverse complement paired-end FASTQ reads file using Seqtk.

Input arguments reads

label
Reads

type
`data:reads:fastq:paired`

required
True

disabled
False

hidden
False

select_mate

label
Select mate

type
`basic:string`

description
Select the which mate should be reverse complemented.

required
True

disabled
False

hidden
False

default
`Mate 1`

choices

- Mate 1: `Mate 1`
- Mate 2: `Mate 2`

- Both: `Both`

Output results fastq

label
Reverse complemented FASTQ file

type
`list:basic:file`

required
True

disabled
False

hidden
False

fastq2

label
Remaining mate

type
`list:basic:file`

required
True

disabled
False

hidden
False

fastqc_url

label
Quality control with FastQC (Mate 1)

type
`list:basic:file:html`

required
True

disabled
False

hidden
False

fastqc_archive

label
Download FastQC archive (Mate 1)

type
`list:basic:file`

required
True

disabled
False

hidden
False

fastqc_url2

label
Quality control with FastQC (Mate 2)

type
`list:basic:file:html`

required
True

disabled
False

hidden
False

fastqc_archive2

label
Download FastQC archive (Mate 2)

type
`list:basic:file`

required
True

disabled
False

hidden
False

Reverse complement FASTQ (single-end)

`data:reads:fastq:single:seqtk:seqtk-rev-complement-single` (*data:reads:fastq:single* **reads**) [Source: v1.3.2]

Reverse complement single-end FASTQ reads file using Seqtk.

Input arguments reads

label
Reads

type
`data:reads:fastq:single`

required
True

disabled
False

hidden
False

Output results fastq

label
Reverse complemented FASTQ file

type
`list:basic:file`

required
True

disabled
False

hidden
False

fastqc_url

label
Quality control with FastQC

type
`list:basic:file:html`

required
True

disabled
False

hidden
False

fastqc_archive

label
Download FastQC archive

type
`list:basic:file`

required
True

disabled
False

hidden
False

SAM header

`data:sam:headerupload-header-sam` (*basic:file* **src**)[Source: v1.2.3]

Upload a mapping file header in SAM format.

Input arguments src

label
Header (SAM)

type
`basic:file`

description
A mapping file header in SAM format.

validate_regex
`\.(sam)$`

Output results sam

label
Uploaded file

type
`basic:file`

SRA data

`data:sra:import-sra` (*list:basic:string* **sra_accession**, *basic:boolean* **prefetch**, *basic:string* **max_size_prefetch**, *basic:integer* **min_spot_id**, *basic:integer* **max_spot_id**, *basic:integer* **min_read_len**, *basic:boolean* **clip**, *basic:boolean* **aligned**, *basic:boolean* **unaligned**) [Source: v1.5.1]

Import reads from SRA.

Import single or paired-end reads from Sequence Read Archive (SRA) via an SRA accession number. SRA stores raw sequencing data and alignment information from high-throughput sequencing platforms.

Input arguments sra_accession

label
SRA accession(s)

type
`list:basic:string`

required
True

disabled
False

hidden
False

advanced.prefetch

label
Prefetch SRA file

type
`basic:boolean`

required
True

disabled
False

hidden
False

default
`True`

advanced.max_size_prefetch

label
Maximum file size to download in KB

type
`basic:string`

description
A unit prefix can be used instead of a value in KB (e.g. 1024M or 1G).

required
True

disabled
False

hidden
False

default
`20G`

advanced.min_spot_id

label
Minimum spot ID

type
`basic:integer`

required
False

disabled
False

hidden
False

advanced.max_spot_id

label
Maximum spot ID

type
`basic:integer`

required
False

disabled
False

hidden
False

advanced.min_read_len

label
Minimum read length

type
`basic:integer`

required
False

disabled
False

hidden
False

advanced.clip

label
Clip adapter sequences

type
`basic:boolean`

required
True

disabled
False

hidden
False

default
`False`

advanced.aligned

label
Dump only aligned sequences

type
`basic:boolean`

required
True

disabled
False

hidden
False

default
`False`

advanced.unaligned

label
Dump only unaligned sequences

type
`basic:boolean`

required
True

disabled
False

hidden
False

default
`False`

Output results

SRA data (paired-end)

`data:reads:fastq:paired:import-sra-paired` (*list:basic:string* **sra_accession**, *basic:boolean* **prefetch**, *basic:string* **max_size_prefetch**, *basic:integer* **min_spot_id**, *basic:integer* **max_spot_id**, *basic:integer* **min_read_len**, *basic:boolean* **clip**, *basic:boolean* **aligned**, *basic:boolean* **unaligned**) [Source: v1.6.1]

Import paired-end reads from SRA.

Import paired-end reads from Sequence Read Archive (SRA) via an SRA accession number. SRA stores raw sequencing data and alignment information from high-throughput sequencing platforms.

Input arguments sra_accession

label
SRA accession(s)

type
`list:basic:string`

required
True

disabled
False

hidden
False

advanced.prefetch

label
Prefetch SRA file

type
`basic:boolean`

required
True

disabled
False

hidden
False

default
`True`

advanced.max_size_prefetch

label
Maximum file size to download in KB

type
basic:string

description
A unit prefix can be used instead of a value in KB (e.g. 1024M or 1G).

required
True

disabled
False

hidden
False

default
20G

advanced.min_spot_id

label
Minimum spot ID

type
basic:integer

required
False

disabled
False

hidden
False

advanced.max_spot_id

label
Maximum spot ID

type
basic:integer

required
False

disabled
False

hidden
False

advanced.min_read_len

label
Minimum read length

type
basic:integer

required
False

disabled
False

hidden
False

advanced.clip

label
Clip adapter sequences

type
`basic:boolean`

required
True

disabled
False

hidden
False

default
`False`

advanced.aligned

label
Dump only aligned sequences

type
`basic:boolean`

required
True

disabled
False

hidden
False

default
`False`

advanced.unaligned

label
Dump only unaligned sequences

type
`basic:boolean`

required
True

disabled
False

hidden
False

default
`False`

Output results fastq

label
Reads file (mate 1)

type
`list:basic:file`

required
True

disabled
False

hidden
False

fastq2

label
Reads file (mate 2)

type
`list:basic:file`

required
True

disabled
False

hidden
False

fastqc_url

label
Quality control with FastQC (mate 1)

type
`list:basic:file:html`

required
True

disabled
False

hidden
False

fastqc_url2

label
Quality control with FastQC (mate 2)

type
`list:basic:file:html`

required
True

disabled
False

hidden
False

fastqc_archive

label
Download FastQC archive (mate 1)

type
`list:basic:file`

required
True

disabled
False

hidden
False

fastqc_archive2

label
Download FastQC archive (mate 2)

type
`list:basic:file`

required
True

disabled
False

hidden
False

SRA data (single-end)

`data:reads:fastq:single:import-sra-single` (*list:basic:string* **sra_accession**, *basic:boolean* **prefetch**, *basic:string* **max_size_prefetch**, *basic:integer* **min_spot_id**, *basic:integer* **max_spot_id**, *basic:integer* **min_read_len**, *basic:boolean* **clip**, *basic:boolean* **aligned**, *basic:boolean* **unaligned**) [Source: v1.6.1]

Import single-end reads from SRA.

Import single-end reads from Sequence Read Archive (SRA) via an SRA accession number. SRA stores raw sequencing data and alignment information from high-throughput sequencing platforms.

Input arguments sra_accession

label
SRA accession(s)

type
`list:basic:string`

required
True

disabled
False

hidden
False

advanced.prefetch

label
Prefetch SRA file

type
`basic:boolean`

required
True

disabled
False

hidden
False

default
`True`

advanced.max_size_prefetch

label
Maximum file size to download in KB

type
`basic:string`

description
A unit prefix can be used instead of a value in KB (e.g. 1024M or 1G).

required
True

disabled
False

hidden
False

default
`20G`

advanced.min_spot_id

label
Minimum spot ID

type
`basic:integer`

required
False

disabled
False

hidden
False

advanced.max_spot_id

label
Maximum spot ID

type
`basic:integer`

required
False

disabled
False

hidden
False

advanced.min_read_len

label
Minimum read length

type
`basic:integer`

required
False

disabled
False

hidden
False

advanced.clip

label
Clip adapter sequences

type
`basic:boolean`

required
True

disabled
False

hidden
False

default
`False`

advanced.aligned

label
Dump only aligned sequences

type
`basic:boolean`

required
True

disabled
False

hidden
False

default
`False`

advanced.unaligned

label
Dump only unaligned sequences

type
`basic:boolean`

required
True

disabled
False

hidden
False

default
`False`

Output results fastq

label
Reads file

type
`list:basic:file`

required
True

disabled
False

hidden
False

fastqc_url

label
Quality control with FastQC

type
`list:basic:file:html`

required
True

disabled
False

hidden
False

fastqc_archive

label
Download FastQC archive

type
`list:basic:file`

required
True

disabled
False

hidden
False

STAR

`data:alignment:bam:star:alignment-star` (*data:reads:fastq* **reads**, *data:index:star* **genome**, *data:annotation* **annotation**, *basic:boolean* **unstranded**, *basic:boolean* **noncannonical**, *basic:boolean* **gene_counts**, *basic:string* **feature_exon**, *basic:integer* **sjdb_overhang**, *basic:boolean* **chimeric**, *basic:integer* **chim_segment_min**, *basic:boolean* **quant_mode**, *basic:boolean* **single_end**, *basic:string* **out_filter_type**, *basic:integer* **out_multimap_max**, *basic:integer* **out_mismatch_max**, *basic:decimal* **out_mismatch_nl_max**, *basic:integer* **out_score_min**, *basic:decimal* **out_mismatch_nrl_max**, *basic:integer* **align_overhang_min**, *basic:integer* **align_sjdb_overhang_min**, *basic:integer* **align_intron_size_min**, *basic:integer* **align_intron_size_max**, *basic:integer* **align_gap_max**, *basic:string* **align_end_alignment**, *basic:boolean* **two_pass_mode**, *basic:boolean* **out_unmapped**, *basic:string* **out_sam_attributes**, *basic:string* **out_rg_line**, *list:basic:integer* **limit_buffer_size**, *basic:integer* **limit_sam_records**, *basic:integer* **limit_junction_reads**, *basic:integer* **limit_collapsed_junctions**, *basic:integer* **limit_inserted_junctions**) [Source: v5.1.0]

Align reads with STAR aligner.

Spliced Transcripts Alignment to a Reference (STAR) software is based on an alignment algorithm that uses sequential maximum mappable seed search in uncompressed suffix arrays followed by seed clustering and stitching procedure. In addition to unbiased de novo detection of canonical junctions, STAR can discover non-canonical splices and chimeric (fusion) transcripts, and is also capable of mapping full-length RNA sequences. More information can be found in the [STAR manual](https://github.com/alexdobin/STAR/blob/master/doc/STARmanual.pdf) and in the [original paper](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3530905/). The current version of STAR is 2.7.10b.

Input arguments reads

label
Input reads (FASTQ)

type
`data:reads:fastq`

required
True

disabled
False

hidden
False

genome

label
Indexed reference genome

type
`data:index:star`

description
Genome index prepared by STAR aligner indexing tool.

required
True

disabled
False

hidden
False

annotation

label
Annotation file (GTF/GFF3)

type
`data:annotation`

description
Insert known annotations into genome indices at the mapping stage.

required
False

disabled
False

hidden
False

unstranded

label
The data is unstranded [–outSAMstrandField intronMotif]

type
`basic:boolean`

description
For unstranded RNA-seq data, Cufflinks/Cuffdiff require spliced alignments with XS strand attribute, which STAR will generate with –outSAMstrandField intronMotif option. As required, the XS strand attribute will be generated for all alignments that contain splice junctions. The spliced alignments that have undefined strand (i.e. containing only non-canonical unannotated junctions) will be suppressed. If you have stranded RNA-seq data, you do not need to use any specific STAR options. Instead, you

need to run Cufflinks with the library option –library-type options. For example, cufflinks –library-type fr-firststrand should be used for the standard dUTP protocol, including Illumina's stranded TruSeq. This option has to be used only for Cufflinks runs and not for STAR runs.

required
True

disabled
False

hidden
False

default
`False`

noncannonical

label
Remove non-canonical junctions (Cufflinks compatibility)

type
`basic:boolean`

description
It is recommended to remove the non-canonical junctions for Cufflinks runs using –outFilterIntronMotifs RemoveNoncanonical.

required
True

disabled
False

hidden
False

default
`False`

gene_counts

label
Gene count [–quantMode GeneCounts]

type
`basic:boolean`

description
With this option set to True STAR will count the number of reads per gene while mapping. A read is counted if it overlaps (1nt or more) one and only one gene. Both ends of the paired-end read are checked for overlaps. The counts coincide with those produced by htseq-count with default parameters.

required
True

disabled
False

hidden
False

default
`False`

annotation_options.feature_exon

label
Feature type [–sjdbGTFfeatureExon]

type
`basic:string`

description
Feature type in GTF file to be used as exons for building transcripts.

required
True

disabled
False

hidden
False

default
`exon`

annotation_options.sjdb_overhang

label
Junction length [–sjdbOverhang]

type
`basic:integer`

description
This parameter specifies the length of the genomic sequence around the annotated junction to be used in constructing the splice junction database. Ideally, this length should be equal to the ReadLength-1, where ReadLength is the length of the reads. For instance, for Illumina 2x100b paired-end reads, the ideal value is 100-1=99. In the case of reads of varying length, the ideal value is max(ReadLength)-1. In most cases, the default value of 100 will work as well as the ideal value.

required
True

disabled
False

hidden
False

default
`100`

detect_chimeric.chimeric

label
Detect chimeric and circular alignments [–chimOutType SeparateSAMold]

type
`basic:boolean`

description
To switch on detection of chimeric (fusion) alignments (in addition to normal mapping), –chimSegmentMin should be set to a positive value. Each chimeric alignment consists of two segments.Each

segment is non-chimeric on its own, but the segments are chimeric to each other (i.e. the segments belong to different chromosomes, or different strands, or are far from each other). Both segments may contain splice junctions, and one of the segments may contain portions of both mates. –chimSegmentMin parameter controls the minimum mapped length of the two segments that is allowed. For example, if you have 2x75 reads and used –chimSegmentMin 20, a chimeric alignment with 130b on one chromosome and 20b on the other will be output, while 135 + 15 won't be.

required
True

disabled
False

hidden
False

default
`False`

detect_chimeric.chim_segment_min

label
Minimum length of chimeric segment [–chimSegmentMin]

type
`basic:integer`

required
True

disabled
!detect_chimeric.chimeric

hidden
False

default
`20`

t_coordinates.quant_mode

label
Output in transcript coordinates [–quantMode TranscriptomeSAM]

type
`basic:boolean`

description
With –quantMode TranscriptomeSAM option STAR will output alignments translated into transcript coordinates in the Aligned.toTranscriptome.out.bam file (in addition to alignments in genomic coordinates in Aligned.*.sam/bam files). These transcriptomic alignments can be used with various transcript quantification software that require reads to be mapped to transcriptome, such as RSEM or eXpress.

required
True

disabled
False

hidden
False

default
`False`

t_coordinates.single_end

label
Allow soft-clipping and indels [–quantTranscriptomeBan Singleend]

type
`basic:boolean`

description
By default, the output satisfies RSEM requirements: soft-clipping or indels are not allowed. Use –quantTranscriptomeBan Singleend to allow insertions, deletions and soft-clips in the transcriptomic alignments, which can be used by some expression quantification softwares (e.g. eXpress).

required
True

disabled
!t_coordinates.quant_mode

hidden
False

default
`False`

filtering.out_filter_type

label
Type of filtering [–outFilterType]

type
`basic:string`

description
Normal: standard filtering using only current alignment; BySJout: keep only those reads that contain junctions that passed filtering into SJ.out.tab.

required
True

disabled
False

hidden
False

default
`Normal`

choices

- Normal: `Normal`
- BySJout: `BySJout`

filtering.out_multimap_max

label
Maximum number of loci [–outFilterMultimapNmax]

type
`basic:integer`

description
Maximum number of loci the read is allowed to map to. Alignments (all of them) will be output only if the read maps to no more loci than this value. Otherwise no alignments will be output, and the read will be counted as 'mapped to too many loci' (default: 10).

required
False

disabled
False

hidden
False

filtering.out_mismatch_max

label
Maximum number of mismatches [–outFilterMismatchNmax]

type
`basic:integer`

description
Alignment will be output only if it has fewer mismatches than this value (default: 10). Large number (e.g. 999) switches off this filter.

required
False

disabled
False

hidden
False

filtering.out_mismatch_nl_max

label
Maximum no. of mismatches (map length) [–outFilterMismatchNoverLmax]

type
`basic:decimal`

description
Alignment will be output only if its ratio of mismatches to *mapped* length is less than or equal to this value (default: 0.3). The value should be between 0.0 and 1.0.

required
False

disabled
False

hidden
False

filtering.out_score_min

label
Minumum alignment score [–outFilterScoreMin]

type
`basic:integer`

description
Alignment will be output only if its score is higher than or equal to this value (default: 0).

required
False

disabled
False

hidden
False

filtering.out_mismatch_nrl_max

label
Maximum no. of mismatches (read length) [–outFilterMismatchNoverReadLmax]

type
`basic:decimal`

description
Alignment will be output only if its ratio of mismatches to *read* length is less than or equal to this value (default: 1.0). Using 0.04 for 2x100bp, the max number of mismatches is calculated as 0.04*200=8 for the paired read. The value should be between 0.0 and 1.0.

required
False

disabled
False

hidden
False

alignment.align_overhang_min

label
Minimum overhang [–alignSJoverhangMin]

type
`basic:integer`

description
Minimum overhang (i.e. block size) for spliced alignments (default: 5).

required
False

disabled
False

hidden
False

alignment.align_sjdb_overhang_min

label
Minimum overhang (sjdb) [–alignSJDBoverhangMin]

type
`basic:integer`

description
Minimum overhang (i.e. block size) for annotated (sjdb) spliced alignments (default: 3).

required
False

disabled
False

hidden
False

alignment.align_intron_size_min

label
Minimum intron size [–alignIntronMin]

type
`basic:integer`

description
Minimum intron size: the genomic gap is considered an intron if its length >= alignIntronMin, otherwise it is considered Deletion (default: 21).

required
False

disabled
False

hidden
False

alignment.align_intron_size_max

label
Maximum intron size [–alignIntronMax]

type
`basic:integer`

description
Maximum intron size, if 0, max intron size will be determined by (2pow(winBinNbits)*winAnchorDistNbins)(default: 0).

required
False

disabled
False

hidden
False

alignment.align_gap_max

label
Minimum gap between mates [–alignMatesGapMax]

type
`basic:integer`

description
Maximum gap between two mates, if 0, max intron gap will be determined by (2pow(winBinNbits)*winAnchorDistNbins) (default: 0).

required
False

disabled
False

hidden
False

alignment.align_end_alignment

label
Read ends alignment [–alignEndsType]

type
`basic:string`

description
Type of read ends alignment (default: Local). Local: standard local alignment with soft-clipping allowed. EndToEnd: force end-to-end read alignment, do not soft-clip. Extend5pOfRead1: fully extend only the 5p of the read1, all other ends: local alignment. Extend5pOfReads12: fully extend only the 5' of the both read1 and read2, all other ends use local alignment.

required
False

disabled
False

hidden
False

choices

- Local: `Local`
- EndToEnd: `EndToEnd`
- Extend5pOfRead1: `Extend5pOfRead1`
- Extend5pOfReads12: `Extend5pOfReads12`

two_pass_mapping.two_pass_mode

label
Use two pass mode [–twopassMode]

type
`basic:boolean`

description
Use two-pass maping instead of first-pass only. In two-pass mode we first perform first-pass mapping, extract junctions, insert them into genome index, and re-map all reads in the second mapping pass.

required
True

disabled
False

hidden
False

default
`False`

output_options.out_unmapped

label
Output unmapped reads (SAM) [–outSAMunmapped Within]

type
`basic:boolean`

description
Output of unmapped reads in the SAM format.

required
True

disabled
False

hidden
False

default
`False`

output_options.out_sam_attributes

label
Desired SAM attributes [–outSAMattributes]

type
`basic:string`

description
A string of desired SAM attributes, in the order desired for the output SAM.

required
True

disabled
False

hidden
False

default
`Standard`

choices

- Standard: `Standard`
- All: `All`
- NH HI NM MD: `NH HI NM MD`
- None: `None`

output_options.out_rg_line

label
SAM/BAM read group line [–outSAMattrRGline]

type
`basic:string`

description
The first word contains the read group identifier and must start with ID:, e.g. –outSAMattrRGline ID:xxx CN:yy ”DS:z z z” xxx will be added as RG tag to each output alignment. Any spaces in the tag values have to be double quoted. Comma separated RG lines correspons to different (comma separated) input files in –readFilesIn. Commas have to be surrounded by spaces, e.g. –outSAMattrRGline ID:xxx , ID:zzz ”DS:z z” , ID:yyy DS:yyyy.

required
False

disabled
False

hidden
False

limits.limit_buffer_size

label
Buffer size [–limitIObufferSize]

type
`list:basic:integer`

description
Maximum available buffers size (bytes) for input/output, per thread. Parameter requires two numbers - separate sizes for input and output buffers.

required
True

disabled
False

hidden
False

default
`[30000000, 50000000]`

limits.limit_sam_records

label
Maximum size of the SAM record [–limitOutSAMoneReadBytes]

type
`basic:integer`

description
Maximum size of the SAM record (bytes) for one read. Recommended value: >(2*(LengthMate1+LengthMate2+100)*outFilterMultimapNmax.

required
True

disabled
False

hidden
False

default
`100000`

limits.limit_junction_reads

label
Maximum number of junctions [–limitOutSJoneRead]

type
`basic:integer`

description
Maximum number of junctions for one read (including all multi-mappers).

required
True

disabled
False

hidden
False

default
`1000`

limits.limit_collapsed_junctions

label
Maximum number of collapsed junctions [–limitOutSJcollapsed]

type
`basic:integer`

required
True

disabled
False

hidden
False

default
`1000000`

limits.limit_inserted_junctions

label
Maximum number of junction to be inserted [–limitSjdbInsertNsj]

type
`basic:integer`

description
Maximum number of junction to be inserted to the genome on the fly at the mapping stage, including those from annotations and those detected in the 1st step of the 2-pass run.

required
True

disabled
False

hidden
False

default
`1000000`

Output results bam

label
Alignment file

type
`basic:file`

required
True

disabled
False

hidden
False

bai

label
BAM file index

type
`basic:file`

required
True

disabled
False

hidden
False

unmapped_1

label
Unmapped reads (mate 1)

type
`basic:file`

required
False

disabled
False

hidden
False

unmapped_2

label
Unmapped reads (mate 2)

type
`basic:file`

required
False

disabled
False

hidden
False

sj

label
Splice junctions

type
`basic:file`

required
True

disabled
False

hidden
False

chimeric

label
Chimeric alignments

type
`basic:file`

required
False

disabled
False

hidden
False

alignment_transcriptome

label
Alignment (transcriptome coordinates)

type
`basic:file`

required
False

disabled
False

hidden
False

gene_counts

label
Gene counts

type
`basic:file`

required
False

disabled
False

hidden
False

stats

label
Statistics

type
`basic:file`

required
True

disabled
False

hidden
False

species

label
Species

type
`basic:string`

required
True

disabled
False

hidden
False

build

label
Build

type
`basic:string`

required
True

disabled
False

hidden
False

STAR genome index

`data:index:star:alignment-star-index` (*data:seq:nucleotide* **ref_seq**, *data:annotation* **annotation**, *basic:string* **source**, *basic:string* **feature_exon**, *basic:integer* **sjdb_overhang**, *basic:integer* **genome_sa_string_len**, *basic:integer* **genome_chr_bin_size**, *basic:integer* **genome_sa_sparsity**) [Source: v4.0.0]

Generate STAR genome index.

Generate genome indices files from the supplied reference genome sequence and GTF files. The current version of STAR is 2.7.10b.

Input arguments ref_seq

label
Reference sequence (nucleotide FASTA)

type
`data:seq:nucleotide`

required
True

disabled
False

hidden
False

annotation

label
Annotation file (GTF/GFF3)

type
`data:annotation`

description
Insert known annotations into genome indices at the indexing stage.

required
False

disabled
False

hidden
False

source

label
Gene ID Database Source

type
`basic:string`

required
False

disabled
annotation

hidden
False

choices

- ENSEMBL: `ENSEMBL`
- NCBI: `NCBI`
- UCSC: `UCSC`

annotation_options.feature_exon

label
Feature type [–sjdbGTFfeatureExon]

type
`basic:string`

description
Feature type in GTF file to be used as exons for building transcripts.

required
True

disabled
False

hidden
False

default
`exon`

annotation_options.sjdb_overhang

label
Junction length [–sjdbOverhang]

type
`basic:integer`

description
This parameter specifies the length of the genomic sequence around the annotated junction to be used in constructing the splice junction database. Ideally, this length should be equal to the ReadLength-1, where ReadLength is the length of the reads. For instance, for Illumina 2x100b paired-end reads, the ideal value is 100-1=99. In case of reads of varying length, the ideal value is max(ReadLength)-1. In most cases, the default value of 100 will work as well as the ideal value.

required
True

disabled
False

hidden
False

default
`100`

advanced.genome_sa_string_len

label
Small genome adjustment [–genomeSAindexNbases]

type
`basic:integer`

description
For small genomes, the parameter –genomeSAindexNbases needs to be scaled down, with a typical value of min(14, log2(GenomeLength)/2 - 1). For example, for 1 megaBase genome, this is equal to 9, for 100 kiloBase genome, this is equal to 7.

required
False

disabled
False

hidden
False

advanced.genome_chr_bin_size

label
Bin size for genome storage [–genomeChrBinNbits]

type
`basic:integer`

description
If you are using a genome with a large (>5,000) number of references (chrosomes/scaffolds), you may need to reduce the –genomeChrBinNbits to reduce RAM consumption. The following scaling is recommended: –genomeChrBinNbits = min(18, log2(GenomeLength / NumberOfReferences)). For example, for 3 gigaBase genome with 100,000 chromosomes/scaffolds, this is equal to 15.

required
False

disabled
False

hidden
False

advanced.genome_sa_sparsity

label
Suffix array sparsity [–genomeSAsparseD]

type
`basic:integer`

description
Suffix array sparsity, i.e. distance between indices: use bigger numbers to decrease needed RAM at the cost of mapping speed reduction (integer > 0, default = 1).

required
False

disabled
False

hidden
False

Output results index

label
Indexed genome

type
`basic:dir`

required
True

disabled
False

hidden
False

fastagz

label
FASTA file (compressed)

type
`basic:file`

required
True

disabled
False

hidden
False

fasta

label
FASTA file

type
`basic:file`

required
True

disabled
False

hidden
False

fai

label
FASTA file index

type
`basic:file`

required
True

disabled
False

hidden
False

source

label
Gene ID source

type
`basic:string`

required
True

disabled
False

hidden
False

species

label
Species

type
`basic:string`

required
True

disabled
False

hidden
False

build

label
Build

type
`basic:string`

required
True

disabled
False

hidden
False

STAR-based gene quantification workflow

data:workflow:rnaseq:star:qc:workflow-bbduk-star-qc (*data:reads:fastq* **reads**,
data:index:star **genome**,
data:annotation **annotation**,
basic:string **assay_type**,
data:index:salmon **cdna_index**,
data:index:star **rrna_reference**,
data:index:star **globin_reference**,
list:data:seq:nucleotide **adapters**,
list:basic:string **custom_adapter_sequences**,
basic:integer **kmer_length**,
basic:integer **min_k**,
basic:integer **hamming_distance**,
basic:integer **maxns**,
basic:integer **trim_quality**,
basic:integer **min_length**,
basic:string **quality_encoding_offset**,
basic:boolean **ignore_bad_quality**,
basic:boolean **unstranded**,
basic:boolean **noncannonical**,
basic:boolean **chimeric**,
basic:integer **chim_segment_min**,
basic:boolean **quant_mode**,
basic:boolean **single_end**,
basic:string **out_filter_type**,
basic:integer **out_multimap_max**,
basic:integer **out_mismatch_max**,
basic:decimal **out_mismatch_nl_max**,
basic:integer **out_score_min**,
basic:decimal **out_mismatch_nrl_max**,
basic:integer **align_overhang_min**,
basic:integer **align_sjdb_overhang_min**,
basic:integer **align_intron_size_min**,
basic:integer **align_intron_size_max**,
basic:integer **align_gap_max**,
basic:string **align_end_alignment**,
basic:boolean **two_pass_mode**,
basic:boolean **out_unmapped**,
basic:string **out_sam_attributes**,
basic:string **out_rg_line**,
basic:integer **n_reads**, *basic:integer* **n_reads**,
basic:integer **seed**, *basic:decimal* **fraction**,
basic:boolean **two_pass**)[Source: v1.4.0]

STAR-based RNA-seq pipeline.

First, reads are preprocessed by __BBDuk__ which removes adapters, trims reads for quality from the 3'-end, and discards reads that are too short after trimming. Compared to similar tools, BBDuk is regarded for its computational efficiency. Next, preprocessed reads are aligned by __STAR__ aligner. At the time of implementation, STAR is considered a state-of-the-art tool that consistently produces accurate results from diverse sets of reads, and performs well even with default settings. STAR aligner counts and reports the number of aligned reads per gene while mapping. STAR version used is 2.7.10b. For more information see [this comparison of RNA-seq aligners](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5792058/).

rRNA contamination rate in the sample is determined using the STAR aligner. Quality-trimmed reads are downsampled (using __Seqtk__ tool) and aligned to the rRNA reference sequences. The alignment rate indicates the percentage of

the reads in the sample that are derived from the rRNA sequences. Final step of the workflow is QoRTs QC analysis with downsampled reads.

Input arguments reads

label
Reads (FASTQ)

type
`data:reads:fastq`

description
Reads in FASTQ file, single or paired end.

required
True

disabled
False

hidden
False

genome

label
Indexed reference genome

type
`data:index:star`

description
Genome index prepared by STAR aligner indexing tool.

required
True

disabled
False

hidden
False

annotation

label
Annotation

type
`data:annotation`

description
GTF and GFF3 annotation formats are supported.

required
True

disabled
False

hidden
False

assay_type

label
Assay type

type
`basic:string`

description
In strand non-specific assay a read is considered overlapping with a feature regardless of whether it is mapped to the same or the opposite strand as the feature. In strand-specific forward assay and single reads, the read has to be mapped to the same strand as the feature. For paired-end reads, the first read has to be on the same strand and the second read on the opposite strand. In strand-specific reverse assay these rules are reversed.

required
True

disabled
False

hidden
False

default
`non_specific`

choices

- Strand non-specific: `non_specific`
- Strand-specific forward: `forward`
- Strand-specific reverse: `reverse`
- Detect automatically: `auto`

cdna_index

label
Indexed cDNA reference sequence

type
`data:index:salmon`

description
Transcriptome index file created using the Salmon indexing tool. cDNA (transcriptome) sequences used for index file creation must be derived from the same species as the input sequencing reads to obtain the reliable analysis results.

required
False

disabled
False

hidden
assay_type != 'auto'

rrna_reference

label
Indexed rRNA reference sequence

type
`data:index:star`

description
Reference sequence index prepared by STAR aligner indexing tool.

required
True

disabled
False

hidden
False

globin_reference

label
Indexed Globin reference sequence

type
`data:index:star`

description
Reference sequence index prepared by STAR aligner indexing tool.

required
True

disabled
False

hidden
False

preprocessing.adapters

label
Adapters

type
`list:data:seq:nucleotide`

description
FASTA file(s) with adapters.

required
False

disabled
False

hidden
False

preprocessing.custom_adapter_sequences

label
Custom adapter sequences

type
`list:basic:string`

description
Custom adapter sequences can be specified by inputting them one by one and pressing Enter after each sequence.

required
False

disabled
False

hidden
False

default
`[]`

preprocessing.kmer_length

label
K-mer length [k=]

type
`basic:integer`

description
K-mer length used for finding contaminants. Contaminants shorter than k-mer length will not be found. K-mer length must be at least 1.

required
True

disabled
False

hidden
False

default
`23`

preprocessing.min_k

label
Minimum k-mer length at right end of reads used for trimming [mink=]

type
`basic:integer`

required
True

disabled
preprocessing.adapters.length === 0 && preprocessing.custom_adapter_sequences.length === 0

hidden
False

default
`11`

preprocessing.hamming_distance

label
Maximum Hamming distance for k-mers [hammingdistance=]

type
`basic:integer`

description
Hamming distance i.e. the number of mismatches allowed in the k-mer.

required
True

disabled
False

hidden
False

default
`1`

preprocessing.maxns

label
Max Ns after trimming [maxns=]

type
`basic:integer`

description
If non-negative, reads with more Ns than this (after trimming) will be discarded.

required
True

disabled
False

hidden
False

default
`-1`

preprocessing.trim_quality

label
Average quality below which to trim region [trimq=]

type
`basic:integer`

description
Phred algorithm is used, which is more accurate than naive trimming.

required
True

disabled
False

hidden
False

default
`10`

preprocessing.min_length

label
Minimum read length [minlength=]

type
`basic:integer`

description
Reads shorter than minimum read length after trimming are discarded.

required
True

disabled
False

hidden
False

default
`20`

preprocessing.quality_encoding_offset

label
Quality encoding offset [qin=]

type
`basic:string`

description
Quality encoding offset for input FASTQ files.

required
True

disabled
False

hidden
False

default
`auto`

choices

- Sanger / Illumina 1.8+: `33`
- Illumina up to 1.3+, 1.5+: `64`
- Auto: `auto`

preprocessing.ignore_bad_quality

label
Ignore bad quality [ignorebadquality]

type
`basic:boolean`

description
Don't crash if quality values appear to be incorrect.

required
True

disabled
False

hidden
False

default
`False`

alignment.unstranded

label
The data is unstranded [–outSAMstrandField intronMotif]

type
`basic:boolean`

description
For unstranded RNA-seq data, Cufflinks/Cuffdiff require spliced alignments with XS strand attribute, which STAR will generate with –outSAMstrandField intronMotif option. As required, the XS strand attribute will be generated for all alignments that contain splice junctions. The spliced alignments that have undefined strand (i.e. containing only non-canonical unannotated junctions) will be suppressed. If you have stranded RNA-seq data, you do not need to use any specific STAR options. Instead, you need to run Cufflinks with the library option –library-type options. For example, cufflinks –library-type fr-firststrand should be used for the standard dUTP protocol, including Illumina's stranded TruSeq. This option has to be used only for Cufflinks runs and not for STAR runs.

required
True

disabled
False

hidden
False

default
`False`

alignment.noncannonical

label
Remove non-canonical junctions (Cufflinks compatibility)

type
`basic:boolean`

description
It is recommended to remove the non-canonical junctions for Cufflinks runs using –outFilterIntronMotifs RemoveNoncanonical.

required
True

disabled
False

hidden
False

default
`False`

alignment.chimeric_reads.chimeric

label
Detect chimeric and circular alignments [–chimOutType SeparateSAMold]

type
`basic:boolean`

description
To switch on detection of chimeric (fusion) alignments (in addition to normal mapping), –chimSegmentMin should be set to a positive value. Each chimeric alignment consists of two segments. Each segment is non-chimeric on its own, but the segments are chimeric to each other (i.e. the segments belong to different chromosomes, or different strands, or are far from each other). Both segments may contain splice junctions, and one of the segments may contain portions of both mates. –chimSegmentMin parameter controls the minimum mapped length of the two segments that is allowed. For example, if you have 2x75 reads and used –chimSegmentMin 20, a chimeric alignment with 130b on one chromosome and 20b on the other will be output, while 135 + 15 won't be.

required
True

disabled
False

hidden
False

default
`False`

alignment.chimeric_reads.chim_segment_min

label
Minimum length of chimeric segment [–chimSegmentMin]

type
`basic:integer`

required
True

disabled
!alignment.chimeric_reads.chimeric

hidden
False

default
`20`

alignment.transcript_output.quant_mode

label
Output in transcript coordinates [–quantMode]

type
`basic:boolean`

description
With –quantMode TranscriptomeSAM option STAR will output alignments translated into transcript coordinates in the Aligned.toTranscriptome.out.bam file (in addition to alignments in genomic coordinates in Aligned.*.sam/bam files). These transcriptomic alignments can be used with various transcript quantification software that require reads to be mapped to transcriptome, such as RSEM or eXpress.

required
True

disabled
False

hidden
False

default
`False`

alignment.transcript_output.single_end

label
Allow soft-clipping and indels [–quantTranscriptomeBan Singleend]

type
`basic:boolean`

description
By default, the output satisfies RSEM requirements: soft-clipping or indels are not allowed. Use –quantTranscriptomeBan Singleend to allow insertions, deletions and soft-clips in the transcriptomic alignments, which can be used by some expression quantification softwares (e.g. eXpress).

required
True

disabled
!t_coordinates.quant_mode

hidden
False

default
`False`

alignment.filtering_options.out_filter_type

label
Type of filtering [–outFilterType]

type
`basic:string`

description
Normal: standard filtering using only current alignment; BySJout: keep only those reads that contain junctions that passed filtering into SJ.out.tab.

required
True

disabled
False

hidden
False

default
`Normal`

choices

- Normal: `Normal`

- BySJout: `BySJout`

alignment.filtering_options.out_multimap_max

label
Maximum number of loci [–outFilterMultimapNmax]

type
`basic:integer`

description
Maximum number of loci the read is allowed to map to. Alignments (all of them) will be output only if the read maps to no more loci than this value. Otherwise no alignments will be output, and the read will be counted as 'mapped to too many loci' (default: 10).

required
False

disabled
False

hidden
False

alignment.filtering_options.out_mismatch_max

label
Maximum number of mismatches [–outFilterMismatchNmax]

type
`basic:integer`

description
Alignment will be output only if it has fewer mismatches than this value (default: 10). Large number (e.g. 999) switches off this filter.

required
False

disabled
False

hidden
False

alignment.filtering_options.out_mismatch_nl_max

label
Maximum no. of mismatches (map length) [–outFilterMismatchNoverLmax]

type
`basic:decimal`

description
Alignment will be output only if its ratio of mismatches to *mapped* length is less than or equal to this value (default: 0.3). The value should be between 0.0 and 1.0.

required
False

disabled
False

hidden
False

alignment.filtering_options.out_score_min

label
Minimum alignment score [–outFilterScoreMin]

type
`basic:integer`

description
Alignment will be output only if its score is higher than or equal to this value (default: 0).

required
False

disabled
False

hidden
False

alignment.filtering_options.out_mismatch_nrl_max

label
Maximum no. of mismatches (read length) [–outFilterMismatchNoverReadLmax]

type
`basic:decimal`

description
Alignment will be output only if its ratio of mismatches to *read* length is less than or equal to this value (default: 1.0). Using 0.04 for 2x100bp, the max number of mismatches is calculated as 0.04*200=8 for the paired read. The value should be between 0.0 and 1.0.

required
False

disabled
False

hidden
False

alignment.alignment_options.align_overhang_min

label
Minimum overhang [–alignSJoverhangMin]

type
`basic:integer`

description
Minimum overhang (i.e. block size) for spliced alignments (default: 5).

required
False

disabled
False

hidden
False

alignment.alignment_options.align_sjdb_overhang_min

label
Minimum overhang (sjdb) [–alignSJDBoverhangMin]

type
`basic:integer`

description
Minimum overhang (i.e. block size) for annotated (sjdb) spliced alignments (default: 3).

required
False

disabled
False

hidden
False

alignment.alignment_options.align_intron_size_min

label
Minimum intron size [–alignIntronMin]

type
`basic:integer`

description
Minimum intron size: the genomic gap is considered an intron if its length >= alignIntronMin, otherwise it is considered Deletion (default: 21).

required
False

disabled
False

hidden
False

alignment.alignment_options.align_intron_size_max

label
Maximum intron size [–alignIntronMax]

type
`basic:integer`

description
Maximum intron size, if 0, max intron size will be determined by (2pow(winBinNbits)*winAnchorDistNbins)(default: 0).

required
False

disabled
False

hidden
False

alignment.alignment_options.align_gap_max

label
Minimum gap between mates [–alignMatesGapMax]

type
`basic:integer`

description
Maximum gap between two mates, if 0, max intron gap will be determined by (2pow(winBinNbits)*winAnchorDistNbins) (default: 0).

required
False

disabled
False

hidden
False

alignment.alignment_options.align_end_alignment

label
Read ends alignment [–alignEndsType]

type
`basic:string`

description
Type of read ends alignment (default: Local). Local: standard local alignment with soft-clipping allowed. EndToEnd: force end-to-end read alignment, do not soft-clip. Extend5pOfRead1: fully extend only the 5p of the read1, all other ends: local alignment. Extend5pOfReads12: fully extend only the 5' of the both read1 and read2, all other ends use local alignment.

required
True

disabled
False

hidden
False

default
`Local`

choices

- Local: `Local`
- EndToEnd: `EndToEnd`
- Extend5pOfRead1: `Extend5pOfRead1`
- Extend5pOfReads12: `Extend5pOfReads12`

alignment.two_pass_mapping.two_pass_mode

label
Use two pass mode [–twopassMode]

type
`basic:boolean`

description
Use two-pass maping instead of first-pass only. In two-pass mode we first perform first-pass mapping, extract junctions, insert them into genome index, and re-map all reads in the second mapping pass.

required
True

disabled
False

hidden
False

default
`True`

alignment.output_options.out_unmapped

label
Output unmapped reads (SAM) [–outSAMunmapped Within]

type
`basic:boolean`

description
Output of unmapped reads in the SAM format.

required
True

disabled
False

hidden
False

default
`True`

alignment.output_options.out_sam_attributes

label
Desired SAM attributes [–outSAMattributes]

type
`basic:string`

description
A string of desired SAM attributes, in the order desired for the output SAM.

required
True

disabled
False

hidden
False

default
`Standard`

choices

- Standard: `Standard`

- All: `All`
- NH HI NM MD: `NH HI NM MD`
- None: `None`

alignment.output_options.out_rg_line

label
SAM/BAM read group line [–outSAMattrRGline]

type
`basic:string`

description
The first word contains the read group identifier and must start with ID:, e.g. –outSAMattrRGline ID:xxx CN:yy ”DS:z z z” xxx will be added as RG tag to each output alignment. Any spaces in the tag values have to be double quoted. Comma separated RG lines corresponds to different (comma separated) input files in -readFilesIn. Commas have to be surrounded by spaces, e.g. -outSAMattrRGline ID:xxx , ID:zzz ”DS:z z” , ID:yyy DS:yyyy.

required
False

disabled
False

hidden
False

quantification.n_reads

label
Number of reads in subsampled alignment file for strandedness detection

type
`basic:integer`

description
Alignment (.bam) file subsample size. Increase the number of reads to make automatic detection more reliable. Decrease the number of reads to make automatic detection run faster.

required
True

disabled
False

hidden
assay_type != ‘auto’

default
`5000000`

downsampling.n_reads

label
Number of reads

type
`basic:integer`

description
Number of reads to include in downsampling.

required
True

disabled
False

hidden
False

default
`1000000`

downsampling.advanced.seed

label
Seed [-s]

type
`basic:integer`

description
Using the same random seed makes reads downsampling more reproducible in different environments.

required
True

disabled
False

hidden
False

default
`11`

downsampling.advanced.fraction

label
Fraction of reads used

type
`basic:decimal`

description
Use the fraction of reads [0.0 - 1.0] from the original input file instead of the absolute number of reads. If set, this will override the 'Number of reads' input parameter.

required
False

disabled
False

hidden
False

downsampling.advanced.two_pass

label
2-pass mode [-2]

type
`basic:boolean`

description
Enable two-pass mode when downsampling. Two-pass mode is twice as slow but with much reduced memory.

required
True

disabled
False

hidden
False

default
`False`

Output results

Salmon Index

`data:index:salmonsalmon-index` (*data:seq:nucleotide* **nucl**, *data:file* **decoys**, *basic:boolean* **gencode**, *basic:boolean* **keep_duplicates**, *basic:string* **source**, *basic:string* **species**, *basic:string* **build**, *basic:integer* **kmerlen**)[Source: v2.2.1]

Generate index files for Salmon transcript quantification tool.

Input arguments nucl

label
Nucleotide sequence

type
`data:seq:nucleotide`

description
A CDS sequence file in .FASTA format.

decoys

label
Decoys

type
`data:file`

description
Treat these sequences as decoys that may have sequence homologous to some known transcript.

required
False

gencode

label
Gencode

type
`basic:boolean`

description
This flag will expect the input transcript FASTA to be in GENCODE format, and will split the transcript name at the first '|' character. These reduced names will be used in the output and when looking for these transcripts in a gene to transcript GTF.

default
`False`

keep_duplicates

label
Keep duplicates

type
`basic:boolean`

description
This flag will disable the default indexing behavior of discarding sequence-identical duplicate transcripts. If this flag is passed, then duplicate transcripts that appear in the input will be retained and quantified separately.

default
`False`

source

label
Source of attribute ID

type
`basic:string`

choices

- DICTYBASE: `DICTYBASE`
- ENSEMBL: `ENSEMBL`
- NCBI: `NCBI`
- UCSC: `UCSC`

species

label
Species

type
`basic:string`

description
Species latin name.

choices

- Homo sapiens: `Homo sapiens`
- Mus musculus: `Mus musculus`
- Rattus norvegicus: `Rattus norvegicus`
- Dictyostelium discoideum: `Dictyostelium discoideum`

build

label
Genome build

type
`basic:string`

kmerlen

label
Size of k-mers

type
`basic:integer`

description
The size of k-mers that should be used for the quasi index. We find that a k of 31 seems to work well for reads of 75bp or longer, but you might consider a smaller k if you plan to deal with shorter reads.

default
`31`

Output results index

label
Salmon index

type
`basic:dir`

source

label
Source of attribute ID

type
`basic:string`

species

label
Species

type
`basic:string`

build

label
Build

type
`basic:string`

Samtools bedcov

data:bedcov:samtools-bedcov (*data:alignment:bam* **bam**, *data:bed* **bedfile**, *basic:integer* **min_read_qual**, *basic:boolean* **rm_del_ref_skips**, *basic:string* **output_option**) [Source: v1.2.0]

Samtools bedcov.

Reports the total read base count (i.e. the sum of per base read depths) for each genomic region specified in the supplied BED file. The regions are output as they appear in the BED file and are 0-based. The output is formatted as tab-delimited data, where the initial three columns indicate the chromosome, start, and end positions of the region. The subsequent column provides either the cumulative read base counts or the normalized sum of read base counts based on the length of each individual region (mean coverage).

For more information about samtools bedcov, click [here](https://www.htslib.org/doc/samtools-bedcov.html).

Input arguments bam

label
Input BAM file

type
`data:alignment:bam`

required
True

disabled
False

hidden
False

bedfile

label
Target BED file

type
`data:bed`

description
Target BED file with regions to extract.

required
True

disabled
False

hidden
False

advanced.min_read_qual

label
Minimum read mapping quality

type
`basic:integer`

description
Only count reads with mapping quality greater than or equal to [-Q]

required
False

disabled
False

hidden
False

advanced.rm_del_ref_skips

label
Skip deletions and ref skips

type
`basic:boolean`

description
Do not include deletions (D) and ref skips (N) in bedcov computation. [-j]

required
True

disabled
False

hidden
False

default
`False`

advanced.output_option

label
Metric by which to output coverage

type
`basic:string`

description
Opt for either displaying the cumulative read base counts or the normalized read base counts based on the length of each region. The latter approach is not part of samtools but implemented within the resolwe-bio process.

required
False

disabled
False

hidden
False

default
`sum`

choices

- Sum (default): `sum`
- Mean: `mean`

Output results coverage_report

label
Output coverage report

type
`basic:file`

required
True

disabled
False

hidden
False

Samtools coverage (multi-sample)

`data:samtoolscoverage:multi:samtools-coverage-multi` (*list:data:alignment:bam* **bam**, *basic:string* **region**, *basic:integer* **min_read_length**, *basic:integer* **min_mq**, *basic:integer* **min_bq**, *list:basic:string* **excl_flags**, *basic:integer* **depth**, *basic:boolean* **no_header**) [Source: v1.0.0]

Samtools coverage for multiple BAM files.

Computes the depth at each position or region and creates tabulated text.

For more information about samtools coverage, click [here](https://www.htslib.org/doc/samtools-coverage.html).

Input arguments bam

label
Input BAM files

type
`list:data:alignment:bam`

description
Select BAM file(s) for the analysis. Coverage information will be calculated from the merged alignments.

required
True

disabled
False

hidden
False

region

label
Region

type
`basic:string`

description
Region can be specified as: RNAME:STARTPOS-ENDPOS and all position coordinates are 1-based, where RNAME is the name of the contig. If the input BAM file was generated by General RNA-seq pipeline, you should use only chromosome numbers to subset the input file, e.g. 3:30293-39103.

required
False

disabled
False

hidden
False

advanced.min_read_length

label
Minimum read length

type
`basic:integer`

description
Ignore reads shorter than specified number of base pairs.

required
False

disabled
False

hidden
False

advanced.min_mq

label
Minimum mapping quality

type
`basic:integer`

description
Minimum mapping quality for an alignment to be used.

required
False

disabled
False

hidden
False

advanced.min_bq

label
Minimum base quality

type
`basic:integer`

description
Minimum base quality for a base to be considered.

required
False

disabled
False

hidden
False

advanced.excl_flags

label
Filter flags

type
`list:basic:string`

description
Filter flags: skip reads with mask bits set. Press ENTER after each flag.

required
True

disabled
False

hidden
False

default
`['UNMAP', 'SECONDARY', 'QCFAIL', 'DUP']`

advanced.depth

label
Maximum allowed coverage depth

type
`basic:integer`

description
If 0, depth is set to the maximum integer value effectively removing any depth limit.

required
True

disabled
False

hidden
False

default
`1000000`

advanced.no_header

label
No header

type
`basic:boolean`

description
Do not output header.

required
True

disabled
False

hidden
False

default
`False`

Output results table

label
Output coverage table

type
`basic:file`

required
True

disabled
False

hidden
False

build

label
Build

type
`basic:string`

required
True

disabled
False

hidden
False

species

label
Species

type
`basic:string`

required
True

disabled
False

hidden
False

Samtools coverage (single-sample)

`data:samtoolscoverage:single:samtools-coverage-single` (*data:alignment:bam* **bam**,
basic:string **region**,
basic:integer **min_read_length**,
basic:integer **min_mq**,
basic:integer **min_bq**,
list:basic:string **excl_flags**,
basic:integer **depth**,
basic:boolean **no_header**) [Source: v1.0.0]

Samtools coverage for a single BAM file.

Computes the depth at each position or region and creates tabulated text.

For more information about samtools coverage, click [here](https://www.htslib.org/doc/samtools-coverage.html).

Input arguments bam

label
Input BAM file

type
`data:alignment:bam`

description
Select BAM file for the analysis

required
True

disabled
False

hidden
False

region

label
Region

type
`basic:string`

description
Region can be specified as: RNAME:STARTPOS-ENDPOS and all position coordinates are 1-based, where RNAME is the name of the contig. If the input BAM file was generated by General RNA-seq pipeline, you should use only chromosome numbers to subset the input file, e.g. 3:30293-39103.

required
False

disabled
False

hidden
False

advanced.min_read_length

label
Minimum read length

type
`basic:integer`

description
Ignore reads shorter than specified number of base pairs.

required
False

disabled
False

hidden
False

advanced.min_mq

label
Minimum mapping quality

type
`basic:integer`

description
Minimum mapping quality for an alignment to be used.

required
False

disabled
False

hidden
False

advanced.min_bq

label
Minimum base quality

type
`basic:integer`

description
Minimum base quality for a base to be considered.

required
False

disabled
False

hidden
False

advanced.excl_flags

label
Filter flags

type
`list:basic:string`

description
Filter flags: skip reads with mask bits set. Press ENTER after each flag.

required
True

disabled
False

hidden
False

default
`['UNMAP', 'SECONDARY', 'QCFAIL', 'DUP']`

advanced.depth

label
Maximum allowed coverage depth

type
`basic:integer`

description
If 0, depth is set to the maximum integer value effectively removing any depth limit.

required
True

disabled
False

hidden
False

default
`1000000`

advanced.no_header

label
No header

type
`basic:boolean`

description
Do not output header.

required
True

disabled
False

hidden
False

default
`False`

Output results table

label
Output coverage table

type
`basic:file`

required
True

disabled
False

hidden
False

build

label
Build

type
`basic:string`

required
True

disabled
False

hidden
False

species

label
Species

type
`basic:string`

required
True

disabled
False

hidden
False

Samtools fastq (paired-end)

`data:reads:fastq:paired:bamtofastq:bamtofastq-paired` (*data:alignment:bam* **bam**)[Source: v1.3.2]

Convert aligned reads in BAM format to paired-end FASTQ files format.

Input arguments bam

label
BAM file

type
`data:alignment:bam`

required
True

disabled
False

hidden
False

Output results fastq

label
Remaining mate1 reads

type
`list:basic:file`

required
True

disabled
False

hidden
False

fastq2

label
Remaining mate2 reads

type
`list:basic:file`

required
True

disabled
False

hidden
False

fastqc_url

label
Mate1 quality control with FastQC

type
`list:basic:file:html`

required
True

disabled
False

hidden
False

fastqc_url2

label
Mate2 quality control with FastQC

type
`list:basic:file:html`

required
True

disabled
False

hidden
False

fastqc_archive

label
Download mate1 FastQC archive

type
`list:basic:file`

required
True

disabled
False

hidden
False

fastqc_archive2

label
Download mate2 FastQC archive

type
`list:basic:file`

required
True

disabled
False

hidden
False

Samtools idxstats

`data:samtools:idxstats:samtools-idxstats` (*data:alignment:bam* **alignment**)[Source: v1.4.2]

Retrieve and print stats in the index file.

Input arguments alignment

label
Alignment

type
`data:alignment:bam`

required
True

disabled
False

hidden
False

Output results report

label
Samtools idxstats report

type
`basic:file`

required
True

disabled
False

hidden
False

Samtools view

`data:alignment:bam:samtools:samtools-view` (*data:alignment:bam* **bam**, *basic:string* **region**, *data:bed* **bedfile**, *basic:boolean* **include_header**, *basic:boolean* **only_header**, *basic:decimal* **subsample**, *basic:integer* **subsample_seed**, *basic:integer* **threads**)[Source: v1.0.1]

Samtools view.

With no options or regions specified, saves all alignments in the specified input alignment file in BAM format to standard output also in BAM format.

You may specify one or more space-separated region specifications to restrict output to only those alignments which overlap the specified region(s). For more information about samtools view, click [here](https://www.htslib.org/doc/samtools-view.html).

Input arguments bam

label
Input BAM file

type
`data:alignment:bam`

required
True

disabled
False

hidden
False

region

label
Region

type
`basic:string`

description
Region can be specified as: RNAME:STARTPOS-ENDPOS and all position coordinates are 1-based,

where RNAME is the name of the contig. If the input BAM file was generated by General RNA-seq pipeline, you should use only chromosome numbers to subset the input file, e.g. 3:30293-39103.

required
False

disabled
False

hidden
bedfile

bedfile

label
Target BED file

type
`data:bed`

description
Target BED file with regions to extract.If the input BAM file was generated by General RNA-seq pipeline, you should use only chromosome numbers to subset the input file, e.g. 3:30292-39103.

required
False

disabled
False

hidden
region

advanced.include_header

label
Include the header in the output

type
`basic:boolean`

required
True

disabled
advanced.only_header

hidden
False

default
`True`

advanced.only_header

label
Output the header only

type
`basic:boolean`

description
Selecting this option overrides all other options.

required
True

disabled
advanced.include_header

hidden
False

default
`False`

advanced.subsample

label
Fraction of the input alignments

type
`basic:decimal`

description
Output only a proportion of the input alignments, as specified by 0.0 FLOAT 1.0, which gives the fraction of templates/pairs to be kept. This subsampling acts in the same way on all of the alignment records in the same template or read pair, so it never keeps a read but not its mate.

required
False

disabled
False

hidden
False

advanced.subsample_seed

label
Subsampling seed

type
`basic:integer`

description
Subsampling seed used to influence which subset of reads is kept. When subsampling data that has previously been subsampled, be sure to use a different seed value from those used previously; otherwise more reads will be retained than expected.

required
True

disabled
False

hidden
!advanced.subsample

default
11

advanced.threads

label
Number of threads

type
`basic:integer`

description
Number of BAM compression threads to use in addition to main thread.

required
True

disabled
False

hidden
False

default
2

Output results bam

label
Output BAM file

type
`basic:file`

required
True

disabled
False

hidden
False

bai

label
Ouput index file

type
`basic:file`

required
True

disabled
False

hidden
False

stats

label
Alignment statistics

type
`basic:file`

required
False

disabled
False

hidden
False

build

label
Build

type
`basic:string`

required
True

disabled
False

hidden
False

species

label
Species

type
`basic:string`

required
True

disabled
False

hidden
False

Secondary hybrid BAM file

`data:alignment:bam:secondaryupload-bam-secondary` (*data:alignment:bam* **bam**, *basic:file* **src**, *basic:string* **species**, *basic:string* **build**)[Source: v0.10.0]

Upload a secondary mapping file in BAM format.

Input arguments bam

label
Hybrid bam

type
`data:alignment:bam`

description
Secondary bam will be appended to the same sample where hybrid bam is.

required
False

src

label
Mapping (BAM)

type
`basic:file`

description
A mapping file in BAM format. The file will be indexed on upload, so additional BAI files are not required.

validate_regex
`\.(bam)$`

species

label
Species

type
`basic:string`

description
Species latin name.

choices

- Drosophila melanogaster: `Drosophila melanogaster`
- Mus musculus: `Mus musculus`

build

label
Build

type
`basic:string`

Output results bam

label
Uploaded file

type
`basic:file`

bai

label
Index BAI

type
`basic:file`

stats

label
Alignment statistics

type
`basic:file`

species

label
Species

type
`basic:string`

build

label
Build

type
`basic:string`

Single cell BAM file and index

`data:alignment:bam:scseq:upload-bam-scseq-indexed` (*basic:file* **src**, *basic:file* **src2**, *data:screads:* **reads**, *basic:string* **species**, *basic:string* **build**) [Source: v1.4.1]

Import scSeq BAM file and index.

Input arguments src

label
Mapping (BAM)

type
`basic:file`

description
A mapping file in BAM format.

required
True

disabled
False

hidden
False

src2

label
BAM index (*.bam.bai file)

type
`basic:file`

description
An index file of a BAM mapping file (ending with bam.bai).

required
True

disabled
False

hidden
False

reads

label
Single cell fastq reads

type
`data:screads:`

required
True

disabled
False

hidden
False

species

label
Species

type
`basic:string`

description
Species latin name.

required
True

disabled
False

hidden
False

build

label
Build

type
`basic:string`

required
True

disabled
False

hidden
False

Output results bam

label
Uploaded BAM

type
`basic:file`

required
True

disabled
False

hidden
False

bai

label
Index BAI

type
`basic:file`

required
True

disabled
False

hidden
False

stats

label
Alignment statistics

type
`basic:file`

required
True

disabled
False

hidden
False

build

label
Build

type
`basic:string`

required
True

disabled
False

hidden
False

species

label
Species

type
`basic:string`

required
True

disabled
False

hidden
False

Spike-ins quality control

data:spikeinsspikein-qc (*list:data:expression* **samples**, *basic:string* **mix**) [Source: v1.4.1]

Plot spike-ins measured abundances for samples quality control. The process will output graphs showing the correlation between known concentration of ERCC spike-ins and sample's measured abundance.

Input arguments samples

label
Expressions with spike-ins

type
`list:data:expression`

mix

label
Spike-ins mix

type
`basic:string`

description
Select spike-ins mix.

choices

- ERCC Mix 1: `ercc_mix1`
- ERCC Mix 2: `ercc_mix2`
- SIRV-Set 3: `sirv_set3`

Output results plots

label
Plot figures

type
`list:basic:file`

required
False

report

label
HTML report with results

type
`basic:file:html`

required
False

hidden
True

report_zip

label
ZIP file contining HTML report with results

type
`basic:file`

required
False

Subsample FASTQ (paired-end)

`data:reads:fastq:paired:seqtk:seqtk-sample-paired` (*data:reads:fastq:paired* **reads**, *basic:integer* **n_reads**, *basic:integer* **seed**, *basic:decimal* **fraction**, *basic:boolean* **two_pass**) [Source: v1.5.2]

Subsample reads from FASTQ files (paired-end).

[Seqtk](https://github.com/lh3/seqtk) is a fast and lightweight tool for processing sequences in the FASTA or FASTQ format. The Seqtk "sample" command enables subsampling of the large FASTQ file(s).

Input arguments reads

label
Reads

type
`data:reads:fastq:paired`

required
True

disabled
False

hidden
False

n_reads

label
Number of reads

type
`basic:integer`

required
True

disabled
False

hidden
False

default
`1000000`

advanced.seed

label
Seed

type
`basic:integer`

required
True

disabled
False

hidden
False

default
11

advanced.fraction

label
Fraction

type
`basic:decimal`

description
Use the fraction of reads [0 - 1.0] from the orignal input file instead of the absolute number of reads. If set, this will override the 'Number of reads' input parameter.

required
False

disabled
False

hidden
False

advanced.two_pass

label
2-pass mode

type
`basic:boolean`

description
Enable two-pass mode when down-sampling. Two-pass mode is twice as slow but with much reduced memory.

required
True

disabled
False

hidden
False

default
`False`

Output results fastq

label
Remaining mate 1 reads

type
`list:basic:file`

required
True

disabled
False

hidden
False

fastq2

label
Remaining mate 2 reads

type
`list:basic:file`

required
True

disabled
False

hidden
False

fastqc_url

label
Mate 1 quality control with FastQC

type
`list:basic:file:html`

required
True

disabled
False

hidden
False

fastqc_url2

label
Mate 2 quality control with FastQC

type
`list:basic:file:html`

required
True

disabled
False

hidden
False

fastqc_archive

label
Download mate 1 FastQC archive

type
`list:basic:file`

required
True

disabled
False

hidden
False

fastqc_archive2

label
Download mate 2 FastQC archive

type
`list:basic:file`

required
True

disabled
False

hidden
False

Subsample FASTQ (single-end)

`data:reads:fastq:single:seqtk:seqtk-sample-single` (*data:reads:fastq:single* **reads**, *basic:integer* **n_reads**, *basic:integer* **seed**, *basic:decimal* **fraction**, *basic:boolean* **two_pass**) [Source: v1.5.2]

Subsample reads from FASTQ file (single-end).

[Seqtk](https://github.com/lh3/seqtk) is a fast and lightweight tool for processing sequences in the FASTA or FASTQ format. The Seqtk "sample" command enables subsampling of the large FASTQ file(s).

Input arguments reads

label
Reads

type
`data:reads:fastq:single`

required
True

disabled
False

hidden
False

n_reads

label
Number of reads

type
`basic:integer`

required
True

disabled
False

hidden
False

default
`1000000`

advanced.seed

label
Seed

type
`basic:integer`

required
True

disabled
False

hidden
False

default
`11`

advanced.fraction

label
Fraction

type
`basic:decimal`

description
Use the fraction of reads [0 - 1.0] from the original input file instead of the absolute number of reads. If set, this will override the 'Number of reads' input parameter.

required
False

disabled
False

hidden
False

advanced.two_pass

label
2-pass mode

type
`basic:boolean`

description
Enable two-pass mode when down-sampling. Two-pass mode is twice as slow but with much reduced memory.

required
True

disabled
False

hidden
False

default
`False`

Output results fastq

label
Remaining reads

type
`list:basic:file`

required
True

disabled
False

hidden
False

fastqc_url

label
Quality control with FastQC

type
`list:basic:file:html`

required
True

disabled
False

hidden
False

fastqc_archive

label
Download FastQC archive

type
`list:basic:file`

required
True

disabled
False

hidden
False

Subsample FASTQ and BWA Aln (paired-end)

`data:workflow:chipseq:seqtkbwaalnworkflow-subsample-bwa-aln-paired` (*data:reads:fastq:paired* **reads**, *data:index:bwa* **genome**, *basic:integer* **n_reads**, *basic:integer* **seed**, *basic:decimal* **fraction**, *basic:boolean* **two_pass**, *basic:integer* **q**, *basic:boolean* **use_edit**, *basic:integer* **edit_value**, *basic:decimal* **fraction**, *basic:boolean* **seeds**, *basic:integer* **seed_length**, *basic:integer* **seed_dist**) [Source: v1.1.0]

Input arguments reads

label
Reads

type
`data:reads:fastq:paired`

genome

label
Reference genome

type
`data:index:bwa`

downsampling.n_reads

label
Number of reads

type
`basic:integer`

default
`10000000`

downsampling.advanced.seed

label
Seed

type
`basic:integer`

default
11

downsampling.advanced.fraction

label
Fraction

type
`basic:decimal`

description
Use the fraction of reads [0 - 1.0] from the original input file instead of the absolute number of reads. If set, this will override the "Number of reads" input parameter.

required
False

downsampling.advanced.two_pass

label
2-pass mode

type
`basic:boolean`

description
Enable two-pass mode when down-sampling. Two-pass mode is twice as slow but with much reduced memory.

default
`True`

alignment.q

label
Quality threshold

type
`basic:integer`

description
Parameter for dynamic read trimming.

default
`5`

alignment.use_edit

label
Use maximum edit distance (excludes fraction of missing alignments)

type
`basic:boolean`

default
`False`

alignment.edit_value

label
Maximum edit distance

type
`basic:integer`

hidden
!use_edit

default
`5`

alignment.fraction

label
Fraction of missing alignments

type
`basic:decimal`

description
The fraction of missing alignments given 2% uniform base error rate. The maximum edit distance is automatically chosen for different read lengths.

hidden
use_edit

default
`0.04`

alignment.seeds

label
Use seeds

type
`basic:boolean`

default
`True`

alignment.seed_length

label
Seed length

type
`basic:integer`

description
Take the first X subsequence as seed. If X is larger than the query sequence, seeding will be disabled. For long reads, this option is typically ranged from 25 to 35 for value 2 in seed maximum edit distance.

hidden
!seeds

default
`32`

alignment.seed_dist

label
Seed maximum edit distance

type
`basic:integer`

hidden
!seeds

default
`2`

Output results

Subsample FASTQ and BWA Aln (single-end)

`data:workflow:chipseq:seqtkbwaalnworkflow-subsample-bwa-aln-single` (*data:reads:fastq:single* **reads**, *data:index:bwa* **genome**, *basic:integer* **n_reads**, *basic:integer* **seed**, *basic:decimal* **fraction**, *basic:boolean* **two_pass**, *basic:integer* **q**, *basic:boolean* **use_edit**, *basic:integer* **edit_value**, *basic:decimal* **fraction**, *basic:boolean* **seeds**, *basic:integer* **seed_length**, *basic:integer* **seed_dist**) [Source: v1.1.0]

Input arguments reads

label
Reads

type
`data:reads:fastq:single`

genome

label
Reference genome

type
`data:index:bwa`

downsampling.n_reads

label
Number of reads

type
`basic:integer`

default
`10000000`

downsampling.advanced.seed

label
Seed

type
`basic:integer`

default
11

downsampling.advanced.fraction

label
Fraction

type
`basic:decimal`

description
Use the fraction of reads [0 - 1.0] from the original input file instead of the absolute number of reads. If set, this will override the "Number of reads" input parameter.

required
False

downsampling.advanced.two_pass

label
2-pass mode

type
`basic:boolean`

description
Enable two-pass mode when down-sampling. Two-pass mode is twice as slow but with much reduced memory.

default
`True`

alignment.q

label
Quality threshold

type
`basic:integer`

description
Parameter for dynamic read trimming.

default
`5`

alignment.use_edit

label
Use maximum edit distance (excludes fraction of missing alignments)

type
`basic:boolean`

default
`False`

alignment.edit_value

label
Maximum edit distance

type
`basic:integer`

hidden
!use_edit

default
`5`

alignment.fraction

label
Fraction of missing alignments

type
`basic:decimal`

description
The fraction of missing alignments given 2% uniform base error rate. The maximum edit distance is automatically chosen for different read lengths.

hidden
use_edit

default
`0.04`

alignment.seeds

label
Use seeds

type
`basic:boolean`

default
`True`

alignment.seed_length

label
Seed length

type
`basic:integer`

description
Take the first X subsequence as seed. If X is larger than the query sequence, seeding will be disabled. For long reads, this option is typically ranged from 25 to 35 for value 2 in seed maximum edit distance.

hidden
!seeds

default
`32`

alignment.seed_dist

label
Seed maximum edit distance

type
`basic:integer`

hidden
!seeds

default
`2`

Output results

Test basic fields

`data:test:fieldstest-basic-fields` (*basic:boolean* **boolean**, *basic:date* **date**, *basic:datetime* **datetime**, *basic:decimal* **decimal**, *basic:integer* **integer**, *basic:string* **string**, *basic:text* **text**, *basic:url:download* **url_download**, *basic:url:view* **url_view**, *basic:string* **string2**, *basic:string* **string3**, *basic:string* **string4**, *basic:string* **string5**, *basic:string* **string6**, *basic:string* **string7**, *basic:string* **tricky2**) [Source: v1.2.4]

Test with all basic input fields whose values are printed by the processor and returned unmodified as output fields.

Input arguments boolean

label
Boolean

type
`basic:boolean`

default
`True`

date

label
Date

type
`basic:date`

default
`2013-12-31`

datetime

label
Date and time

type
`basic:datetime`

default
`2013-12-31 23:59:59`

decimal

label
Decimal

type
`basic:decimal`

default
`-123.456`

integer

label
Integer

type
`basic:integer`

default
`-123`

string

label
String

type
`basic:string`

default
`Foo b-a-r.gz 1.23`

text

label
Text

type
`basic:text`

default
`Foo bar in 3 lines.`

url_download

label
URL download

type
`basic:url:download`

default
`{'url': 'http://www.w3.org/TR/1998/REC-html40-19980424/html40.pdf'}`

url_view

label
URL view

type
`basic:url:view`

default
`{'name': 'Something', 'url': 'http://www.something.com/'}`

group.string2

label
String 2 required

type
`basic:string`

description
String 2 description.

required
True

disabled
false

hidden
false

placeholder
`Enter string`

group.string3

label
String 3 disabled

type
`basic:string`

description
String 3 description.

disabled
true

default
`disabled`

group.string4

label
String 4 hidden

type
`basic:string`

description
String 4 description.

hidden
True

default
`hidden`

group.string5

label
String 5 choices

type
`basic:string`

description
String 5 description.

hidden
False

default
`choice_2`

choices

- Choice 1: `choice_1`
- Choice 2: `choice_2`
- Choice 3: `choice_3`

group.string6

label
String 6 regex only “Aa”

type
`basic:string`

default
`AAaAaaa`

validate_regex
`^[aA]*$`

group.string7

label
String 7 optional choices

type
`basic:string`

description
String 7 description.

required
False

hidden
False

default
`choice_2`

choices

- Choice 1: `choice_1`
- Choice 2: `choice_2`
- Choice 3: `choice_3`

tricky.tricky1.tricky2

label
Tricky 2

type
`basic:string`

default
`true`

Output results output

label
Result

type
`basic:url:view`

out_boolean

label
Boolean

type
`basic:boolean`

out_date

label
Date

type
`basic:date`

out_datetime

label
Date and time

type
`basic:datetime`

out_decimal

label
Decimal

type
`basic:decimal`

out_integer

label
Integer

type
`basic:integer`

out_string

label
String

type
`basic:string`

out_text

label
Text

type
`basic:text`

out_url_download

label
URL download

type
`basic:url:download`

out_url_view

label
URL view

type
`basic:url:view`

out_group.string2

label
String 2 required

type
`basic:string`

description
String 2 description.

out_group.string3

label
String 3 disabled

type
`basic:string`

description
String 3 description.

out_group.string4

label
String 4 hidden

type
`basic:string`

description
String 4 description.

out_group.string5

label
String 5 choices

type
`basic:string`

description
String 5 description.

out_group.string6

label
String 6 regex only “Aa”

type
`basic:string`

out_group.string7

label
String 7 optional choices

type
`basic:string`

out_tricky.tricky1.tricky2

label
Tricky 2

type
`basic:string`

Test disabled inputs

`data:test:disabledtest-disabled` (*basic:boolean* **broad**, *basic:integer* **broad_width**, *basic:string* **width_label**, *basic:integer* **if_and_condition**) [Source: v1.2.4]

Test disabled input fields.

Input arguments broad

label
Broad peaks

type
`basic:boolean`

default
`False`

broad_width

label
Width of peaks

type
`basic:integer`

disabled
broad === false

default
`5`

width_label

label
Width label

type
`basic:string`

disabled
broad === false

default
`FD`

if_and_condition

label
If width is 5 and label FDR

type
`basic:integer`

disabled
broad_width == 5 && width_label == 'FDR'

default
`5`

Output results output

label
Result

type
`basic:string`

Test hidden inputs

data:test:hiddentest-hidden (*basic:boolean* **broad**, *basic:integer* **broad_width**, *basic:integer* **parameter1**, *basic:integer* **parameter2**, *basic:integer* **broad_width2**) [Source: v1.2.4]

Test hidden input fields

Input arguments broad

label
Broad peaks

type
`basic:boolean`

default
`False`

broad_width

label
Width of peaks

type
`basic:integer`

hidden
broad === false

default
`5`

parameters_broad_f.parameter1

label
parameter1

type
`basic:integer`

default
`10`

parameters_broad_f.parameter2

label
parameter2

type
`basic:integer`

default
`10`

parameters_broad_t.broad_width2

label
Width of peaks2

type
`basic:integer`

default
5

Output results output

label
Result

type
`basic:string`

Test select controler

`data:test:resulttest-list` (*data:test:result* **single**, *list:data:test:result* **multiple**) [Source: v1.2.4]

Test with all basic input fields whose values are printed by the processor and returned unmodified as output fields.

Input arguments single

label
Single

type
`data:test:result`

multiple

label
Multiple

type
`list:data:test:result`

Output results output

label
Result

type
`basic:string`

Test sleep progress

`data:test:resulttest-sleep-progress` (*basic:integer* **t**) [Source: v1.2.4]

Test for the progress bar by sleeping 5 times for the specified amount of time.

Input arguments t

label
Sleep time

type
`basic:integer`

default
5

Output results output

label
Result

type
`basic:string`

Trim Galore (paired-end)

`data:reads:fastq:paired:trimgalore:trimgalore-paired` (*data:reads:fastq:paired* **reads**, *list:basic:string* **adapter**, *list:basic:string* **adapter_2**, *data:seq:nucleotide* **adapter_file_1**, *data:seq:nucleotide* **adapter_file_2**, *basic:string* **universal_adapter**, *basic:integer* **stringency**, *basic:decimal* **error_rate**, *basic:integer* **quality**, *basic:integer* **nextseq**, *basic:string* **phred**, *basic:integer* **min_length**, *basic:integer* **max_n**, *basic:boolean* **retain_unpaired**, *basic:integer* **unpaired_len_1**, *basic:integer* **unpaired_len_2**, *basic:integer* **clip_r1**, *basic:integer* **clip_r2**, *basic:integer* **three_prime_r1**, *basic:integer* **three_prime_r2**, *basic:integer* **trim_5**, *basic:integer* **trim_3**) [Source: v1.3.2]

Process paired-end sequencing reads with Trim Galore.

Trim Galore is a wrapper script that makes use of the publicly available adapter trimming tool Cutadapt and FastQC for quality control once the trimming process has completed.

Low-quality ends are trimmed from reads in addition to adapter removal in a single pass. If no sequence was supplied, Trim Galore will attempt to auto-detect the adapter which has been used. For this it will analyse the first 1 million sequences of the first specified file and attempt to find the first 12 or 13bp of the following standard adapters: Illumina: AGATCGGAAGAGC, Small RNA: TGGAATTCTCGG, Nextera: CTGTCTCTTATA.

If no adapter contamination can be detected within the first 1 million sequences, or in case of a tie between several different adapters, Trim Galore defaults to illumina adapters.

For additional information see official [user guide](https://github.com/FelixKrueger/TrimGalore/blob/master/Docs/Trim_Galore_User_G

Input arguments reads

label
Select paired-end reads

type
`data:reads:fastq:paired`

required
True

disabled
False

hidden
False

adapter_trim.adapter

label
Read 1 adapter sequence

type
`list:basic:string`

description
Adapter sequences to be trimmed. Also see universal adapters field for predefined adapters. This is mutually exclusive with read 1 adapters file and universal adapters.

required
False

disabled
False

hidden
False

default
`[]`

adapter_trim.adapter_2

label
Read 2 adapter sequence

type
`list:basic:string`

description
Optional adapter sequence to be trimmed off read 2 of paired-end files. This is mutually exclusive with read 2 adapters file and universal adapters.

required
False

disabled
False

hidden
False

default
`[]`

adapter_trim.adapter_file_1

label
Read 1 adapters file

type
`data:seq:nucleotide`

description
This is mutually exclusive with read 1 adapters and universal adapters.

required
False

disabled
False

hidden
False

adapter_trim.adapter_file_2

label
Read 2 adapters file

type
`data:seq:nucleotide`

description
This is mutually exclusive with read 2 adapters and universal adapters.

required
False

disabled
False

hidden
False

adapter_trim.universal_adapter

label
Universal adapters

type
`basic:string`

description
Instead of default detection use specific adapters. Use 13bp of the Illumina universal adapter, 12bp of the Nextera adapter or 12bp of the Illumina Small RNA 3' Adapter. Selecting to trim smallRNA adapters will also lower the length value to 18bp. If the smallRNA libraries are paired-end then read 2 adapter will be set to the Illumina small RNA 5' adapter automatically (GATCGTCGGACT) unless defined explicitly. This is mutually exclusive with manually defined adapters and adapter files.

required
False

disabled
False

hidden
False

choices

- Illumina: `--illumina`
- Nextera: `--nextera`
- Illumina small RNA: `--small_rna`

adapter_trim.stringency

label
Overlap with adapter sequence required to trim

type
`basic:integer`

description
Defaults to a very stringent setting of 1, i.e. even a single base pair of overlapping sequence will be trimmed of the 3' end of any read.

required
True

disabled
False

hidden
False

default
`1`

adapter_trim.error_rate

label
Maximum allowed error rate

type
`basic:decimal`

description
Number of errors divided by the length of the matching region

required
True

disabled
False

hidden
False

default
`0.1`

quality_trim.quality

label
Quality cutoff

type
`basic:integer`

description
Trim low-quality ends from reads based on phred score.

required
True

disabled
False

hidden
False

default
`20`

quality_trim.nextseq

label
NextSeq/NovaSeq trim cutoff

type
`basic:integer`

description
NextSeq/NovaSeq-specific quality trimming. Trims also dark cycles appearing as high-quality G bases. This will set a specific quality cutoff, but qualities of G bases are ignored. This can not be used with Quality cutoff and will override it.

required
False

disabled
False

hidden
False

quality_trim.phred

label
Phred score encoding

type
`basic:string`

description
Use either ASCII+33 quality scores as Phred scores (Sanger/Illumina 1.9+ encoding) or ASCII+64 quality scores (Illumina 1.5 encoding) for quality trimming

required
True

disabled
False

hidden
False

default
`--phred33`

choices

- ASCII+33: `--phred33`
- ASCII+64: `--phred64`

quality_trim.min_length

label
Minimum length after trimming

type
`basic:integer`

description
Discard reads that became shorter than selected length because of either quality or adapter trimming. Both reads of a read-pair need to be longer than specified length to be printed out to validated paired-end files. If only one read became too short there is the possibility of keeping such unpaired single-end reads with Retain unpaired. A value of 0 disables filtering based on length.

required
True

disabled
False

hidden
False

default
`20`

quality_trim.max_n

label
Maximum number of Ns

type
`basic:integer`

description
Read exceeding this limit will result in the entire pair being removed from the trimmed output files.

required
False

disabled
False

hidden
False

quality_trim.retain_unpaired

label
Retain unpaired reads after trimming

type
`basic:boolean`

description
If only one of the two paired-end reads became too short, the longer read will be written.

required
True

disabled
False

hidden
False

default
`False`

quality_trim.unpaired_len_1

label
Unpaired read length cutoff for mate 1

type
`basic:integer`

required
True

disabled
False

hidden
!quality_trim.retain_unpaired

default
35

quality_trim.unpaired_len_2

label
Unpaired read length cutoff for mate 2

type
basic:integer

required
True

disabled
False

hidden
!quality_trim.retain_unpaired

default
35

quality_trim.clip_r1

label
Trim bases from 5' end of read 1

type
basic:integer

description
This may be useful if the qualities were very poor, or if there is some sort of unwanted bias at the 5' end.

required
False

disabled
False

hidden
False

quality_trim.clip_r2

label
Trim bases from 5' end of read 2

type
basic:integer

description
This may be useful if the qualities were very poor, or if there is some sort of unwanted bias at the 5' end. For paired-end bisulfite sequencing, it is recommended to remove the first few bp because the end-repair reaction may introduce a bias towards low methylation.

required
False

disabled
False

hidden
False

quality_trim.three_prime_r1

label
Trim bases from 3' end of read 1

type
`basic:integer`

description
Remove bases from the 3' end of read 1 after adapter/quality trimming has been performed. This may remove some unwanted bias from the 3' end that is not directly related to adapter sequence or basecall quality.

required
False

disabled
False

hidden
False

quality_trim.three_prime_r2

label
Trim bases from 3' end of read 2

type
`basic:integer`

description
Remove bases from the 3' end of read 2 after adapter/quality trimming has been performed. This may remove some unwanted bias from the 3' end that is not directly related to adapter sequence or basecall quality.

required
False

disabled
False

hidden
False

hard_trim.trim_5

label
Hard trim sequences from 3' end

type
`basic:integer`

description
Instead of performing adapter-/quality trimming, this option will simply hard-trim sequences to bp from the 3' end. This is incompatible with other hard trimming options.

required
False

disabled
False

hidden
False

hard_trim.trim_3

label
Hard trim sequences from 5' end

type
`basic:integer`

description
Instead of performing adapter-/quality trimming, this option will simply hard-trim sequences to bp from the 5' end. This is incompatible with other hard trimming options.

required
False

disabled
False

hidden
False

Output results fastq

label
Remaining mate 1 reads

type
`list:basic:file`

required
True

disabled
False

hidden
False

fastq2

label
Remaining mate 2 reads

type
`list:basic:file`

required
True

disabled
False

hidden
False

report

label
Trim galore report

type
`basic:file`

required
False

disabled
False

hidden
False

fastqc_url

label
Mate 1 quality control with FastQC

type
`list:basic:file:html`

required
True

disabled
False

hidden
False

fastqc_url2

label
Mate 2 quality control with FastQC

type
`list:basic:file:html`

required
True

disabled
False

hidden
False

fastqc_archive

label
Download mate 1 FastQC archive

type
`list:basic:file`

required
True

disabled
False

hidden
False

fastqc_archive2

label
Download mate 2 FastQC archive

type
`list:basic:file`

required
True

disabled
False

hidden
False

Trimmomatic (paired-end)

`data:reads:fastq:paired:trimmomatictrimmomatic-paired` (*data:reads:fastq:paired* **reads**, *data:seq:nucleotide* **adapters**, *basic:integer* **seed_mismatches**, *basic:integer* **simple_clip_threshold**, *basic:integer* **palindrome_clip_threshold**, *basic:integer* **min_adapter_length**, *basic:boolean* **keep_both_reads**, *basic:integer* **window_size**, *basic:integer* **required_quality**, *basic:integer* **target_length**, *basic:decimal* **strictness**, *basic:integer* **leading**, *basic:integer* **trailing**, *basic:integer* **crop**, *basic:integer* **headcrop**, *basic:integer* **minlen**, *basic:integer* **average_quality**)[Source: v2.5.2]

Trimmomatic performs a variety of useful trimming tasks including removing adapters for Illumina paired-end and single-end data. FastQC is performed for quality control checks on trimmed raw sequence data, which are the output of Trimmomatic. See [Trimmomatic official website](http://www.usadellab.org/cms/?page=trimmomatic), the [introductory paper](https://www.ncbi.nlm.nih.gov/pubmed/24695404), and the [FastQC official website](https://www.bioinformatics.babraham.ac.uk/projects/fastqc/) for more information.

Input arguments reads

label
Reads

type
`data:reads:fastq:paired`

illuminaclip.adapters

label
Adapter sequences

type
`data:seq:nucleotide`

description
Adapter sequence in FASTA format that will be removed from the read. This field as well as 'Seed mismatches', 'Simple clip threshold' and 'Palindrome clip threshold' parameters are needed to perform Illuminacliping. 'Minimum adapter length' and 'Keep both reads' are optional parameters.

required
False

illuminaclip.seed_mismatches

label
Seed mismatches

type
`basic:integer`

description
Specifies the maximum mismatch count which will still allow a full match to be performed. This field as well as 'Adapter sequence', 'Simple clip threshold' and 'Palindrome clip threshold' parameters are needed to perform Illuminacliping.

required
False

disabled
!illuminaclip.adapters

illuminaclip.simple_clip_threshold

label
Simple clip threshold

type
`basic:integer`

description
Specifies how accurate the match between any adapter etc. sequence must be against a read. This field as well as 'Adapter sequence', 'Seed mismatches' and 'Palindrome clip threshold' parameters are needed to perform Illuminacliping.

required
False

disabled
!illuminaclip.adapters

illuminaclip.palindrome_clip_threshold

label
Palindrome clip threshold

type
`basic:integer`

description
Specifies how accurate the match between the two 'adapter ligated' reads must be for PE palindrome read alignment. This field as well as 'Adapter sequence', 'Simple clip threshold' and 'Seed mismatches' parameters are needed to perform Illuminacliping.

required
False

disabled
!illuminaclip.adapters

illuminaclip.min_adapter_length

label
Minimum adapter length

type
`basic:integer`

description
In addition to the alignment score, palindrome mode can verify that a minimum length of adapter has been detected. If unspecified, this defaults to 8 bases, for historical reasons. However, since palindrome mode has a very low false positive rate, this can be safely reduced, even down to 1, to allow shorter adapter fragments to be removed. This field is optional for preforming Illuminaclip. 'Adapter sequences', 'Seed mismatches', 'Simple clip threshold' and 'Palindrome clip threshold' are also needed in order to use this parameter.

disabled
!illuminaclip.seed_mismatches && !illuminaclip.simple_clip_threshold && !illuminaclip.palindrome_clip_threshold

default
`8`

illuminaclip.keep_both_reads

label
Keep both reads

type
`basic:boolean`

description
After read-though has been detected by palindrome mode, and the adapter sequence removed, the reverse read contains the same sequence information as the forward read, albeit in reverse complement. For this reason, the default behaviour is to entirely drop the reverse read.By specifying this parameter, the reverse read will also be retained, which may be useful e.g. if the downstream tools cannot handle a combination of paired and unpaired reads. This field is optional for preforming Illuminaclip. 'Adapter sequence', 'Seed mismatches', 'Simple clip threshold', 'Palindrome clip threshold' and also 'Minimum adapter length' are needed in order to use this parameter.

required
False

disabled
!illuminaclip.seed_mismatches && !illuminaclip.simple_clip_threshold && !illuminaclip.palindrome_clip_threshold && !illuminaclip.min_adapter_length

slidingwindow.window_size

label
Window size

type
`basic:integer`

description
Specifies the number of bases to average across. This field as well as 'Required quality' are needed to perform a 'Sliding window' trimming (cutting once the average quality within the window falls below a threshold).

required
False

slidingwindow.required_quality

label
Required quality

type
`basic:integer`

description
Specifies the average quality required. This field as well as 'Window size' are needed to perform a 'Sliding window' trimming (cutting once the average quality within the window falls below a threshold).

required
False

maxinfo.target_length

label
Target length

type
`basic:integer`

description
This specifies the read length which is likely to allow the location of the read within the target sequence to be determined. This field as well as 'Strictness' are needed to perform 'Maxinfo' feature (an adaptive quality trimmer which balances read length and error rate to maximise the value of each read).

required
False

maxinfo.strictness

label
Strictness

type
`basic:decimal`

description
This value, which should be set between 0 and 1, specifies the balance between preserving as much read length as possible vs. removal of incorrect bases. A low value of this parameter (<0.2) favours longer reads, while a high value (>0.8) favours read correctness. This field as well as 'Target length' are needed to perform 'Maxinfo' feature (an adaptive quality trimmer which balances read length and error rate to maximise the value of each read).

required
False

trim_bases.leading

label
Leading quality

type
`basic:integer`

description
Remove low quality bases from the beginning. Specifies the minimum quality required to keep a base.

required
False

trim_bases.trailing

label
Trailing

type
`basic:integer`

description
Remove low quality bases from the end. Specifies the minimum quality required to keep a base.

required
False

trim_bases.crop

label
Crop

type
`basic:integer`

description
Cut the read to a specified length by removing bases from the end.

required
False

trim_bases.headcrop

label
Headcrop

type
`basic:integer`

description
Cut the specified number of bases from the start of the read.

required
False

reads_filtering.minlen

label
Minimum length

type
`basic:integer`

description
Drop the read if it is below a specified length.

required
False

reads_filtering.average_quality

label
Average quality

type
`basic:integer`

description
Drop the read if the average quality is below the specified level.

required
False

Output results fastq

label
Reads file (mate 1)

type
`list:basic:file`

fastq_unpaired

label
Reads file

type
`basic:file`

required
False

fastq2

label
Reads file (mate 2)

type
`list:basic:file`

fastq2_unpaired

label
Reads file

type
`basic:file`

required
False

fastqc_url

label
Quality control with FastQC (Upstream)

type
`list:basic:file:html`

fastqc_url2

label
Quality control with FastQC (Downstream)

type
`list:basic:file:html`

fastqc_archive

label
Download FastQC archive (Upstream)

type
`list:basic:file`

fastqc_archive2

label
Download FastQC archive (Downstream)

type
`list:basic:file`

Trimmomatic (single-end)

`data:reads:fastq:single:trimmomatictrimmomatic-single` (*data:reads:fastq:single* **reads**, *data:seq:nucleotide* **adapters**, *basic:integer* **seed_mismatches**, *basic:integer* **simple_clip_threshold**, *basic:integer* **window_size**, *basic:integer* **required_quality**, *basic:integer* **target_length**, *basic:decimal* **strictness**, *basic:integer* **leading**, *basic:integer* **trailing**, *basic:integer* **crop**, *basic:integer* **headcrop**, *basic:integer* **minlen**, *basic:integer* **average_quality**) [Source: v2.5.2]

Trimmomatic performs a variety of useful trimming tasks including removing adapters for Illumina paired-end and single-end data. FastQC is performed for quality control checks on trimmed raw sequence data, which are the output of Trimmomatic. See [Trimmomatic official website](http://www.usadellab.org/cms/?page=trimmomatic), the [introductory paper](https://www.ncbi.nlm.nih.gov/pubmed/24695404), and the [FastQC official website](https://www.bioinformatics.babraham.ac.uk/projects/fastqc/) for more information.

Input arguments reads

label
Reads

type
`data:reads:fastq:single`

illuminaclip.adapters

label
Adapter sequences

type
`data:seq:nucleotide`

description
Adapter sequence in FASTA format that will be removed from the read. This field as well as 'Seed mismatches' and 'Simple clip threshold' parameters are needed to perform Illuminacliping.

required
False

illuminaclip.seed_mismatches

label
Seed mismatches

type
`basic:integer`

description
Specifies the maximum mismatch count which will still allow a full match to be performed. This field as well as 'Adapter sequences' and 'Simple clip threshold' parameter are needed to perform Illuminacliping.

required
False

disabled
!illuminaclip.adapters

illuminaclip.simple_clip_threshold

label
Simple clip threshold

type
`basic:integer`

description
Specifies how accurate the match between any adapter etc. sequence must be against a read. This field as well as 'Adapter sequences' and 'Seed mismatches' parameter are needed to perform Illuminacliping.

required
False

disabled
!illuminaclip.adapters

slidingwindow.window_size

label
Window size

type
`basic:integer`

description
Specifies the number of bases to average across. This field as well as 'Required quality' are needed to perform a 'Sliding window' trimming (cutting once the average quality within the window falls below a threshold).

required
False

slidingwindow.required_quality

label
Required quality

type
`basic:integer`

description
Specifies the average quality required in window size. This field as well as 'Window size' are needed to perform a 'Sliding window' trimming (cutting once the average quality within the window falls below a threshold).

required
False

maxinfo.target_length

label
Target length

type
`basic:integer`

description
This specifies the read length which is likely to allow the location of the read within the target sequence to be determined. This field as well as 'Strictness' are needed to perform 'Maxinfo' feature (an adaptive quality trimmer which balances read length and error rate to maximise the value of each read).

required
False

maxinfo.strictness

label
Strictness

type
`basic:decimal`

description
This value, which should be set between 0 and 1, specifies the balance between preserving as much read length as possible vs. removal of incorrect bases. A low value of this parameter (<0.2) favours longer reads, while a high value (>0.8) favours read correctness. This field as well as 'Target length' are needed to perform 'Maxinfo' feature (an adaptive quality trimmer which balances read length and error rate to maximise the value of each read).

required
False

trim_bases.leading

label
Leading quality

type
`basic:integer`

description
Remove low quality bases from the beginning, if below a threshold quality.

required
False

trim_bases.trailing

label
Trailing quality

type
`basic:integer`

description
Remove low quality bases from the end, if below a threshold quality.

required
False

trim_bases.crop

label
Crop

type
`basic:integer`

description
Cut the read to a specified length by removing bases from the end.

required
False

trim_bases.headcrop

label
Headcrop

type
`basic:integer`

description
Cut the specified number of bases from the start of the read.

required
False

reads_filtering.minlen

label
Minimum length

type
`basic:integer`

description
Drop the read if it is below a specified length.

required
False

reads_filtering.average_quality

label
Average quality

type
`basic:integer`

description
Drop the read if the average quality is below the specified level.

required
False

Output results fastq

label
Reads file

type
`list:basic:file`

fastqc_url

label
Quality control with FastQC

type
`list:basic:file:html`

fastqc_archive

label
Download FastQC archive

type
`list:basic:file`

UMI-tools dedup

`data:alignment:bam:umitools:dedup:umi-tools-dedup` (*data:alignment:bam* **alignment**)[Source: v1.5.1]

Deduplicate reads using UMI and mapping coordinates.

Input arguments alignment

label
Alignment

type
`data:alignment:bam`

required
True

disabled
False

hidden
False

Output results bam

label
Clipped BAM file

type
`basic:file`

required
True

disabled
False

hidden
False

bai

label
Index of clipped BAM file

type
`basic:file`

required
True

disabled
False

hidden
False

stats

label
Alignment statistics

type
`basic:file`

required
True

disabled
False

hidden
False

dedup_log

label
Deduplication log

type
`basic:file`

required
True

disabled
False

hidden
False

dedup_stats

label
Deduplication stats

type
`basic:file`

required
True

disabled
False

hidden
False

species

label
Species

type
`basic:string`

required
True

disabled
False

hidden
False

build

label
Build

type
`basic:string`

required
True

disabled
False

hidden
False

Upload microarray expression (unmapped)

`data:microarray:normalized:upload-microarray-expression` (*basic:file* **exp**, *basic:string* **exp_type**, *basic:string* **platform**, *basic:string* **platform_id**, *basic:string* **species**) [Source: v1.1.1]

Import unmapped microarray expression data.

Input arguments exp

label
Normalized expression

type
`basic:file`

description
Normalized expression file with the original probe IDs. Supported file extensions are .tab.*, .tsv.*, .txt.*

required
True

disabled
False

hidden
False

exp_type

label
Normalization type

type
`basic:string`

required
True

disabled
False

hidden
False

platform

label
Microarray platform name

type
`basic:string`

required
True

disabled
False

hidden
False

platform_id

label
GEO platform ID

type
`basic:string`

description
Platform ID according to the GEO database. This can be used in following steps to automatically map probe IDs to genes.

required
False

disabled
False

hidden
False

species

label
Species

type
`basic:string`

description
Select a species name from the dropdown menu or write a custom species name in the species field

required
True

disabled
False

hidden
False

choices

- Homo sapiens: `Homo sapiens`
- Mus musculus: `Mus musculus`
- Rattus norvegicus: `Rattus norvegicus`
- Macaca mulatta: `Macaca mulatta`
- Dictyostelium discoideum: `Dictyostelium discoideum`

Output results exp

label
Uploaded normalized expression

type
`basic:file`

required
True

disabled
False

hidden
False

exp_type

label
Normalization type

type
`basic:string`

required
True

disabled
False

hidden
False

platform

label
Microarray platform type

type
basic:string

required
True

disabled
False

hidden
False

platform_id

label
GEO platform ID

type
basic:string

required
False

disabled
False

hidden
False

species

label
Species

type
basic:string

required
True

disabled
False

hidden
False

Upload proteomics sample

`data:proteomics:massspectrometry:upload-proteomics-sample` (*basic:file* **src**, *basic:string* **species**, *basic:string* **source**) [Source: v1.2.1]

Upload a mass spectrometry proteomics sample data file.

The input 5-column tab-delimited file with the .txt suffix is expected to contain a header line with the following metadata column names: "Uniprot ID", "Gene symbol", "Protein name" and "Number of peptides". The fifth column contains the sample data.

Input arguments src

label
Table containing mass spectrometry data (.txt)

type
`basic:file`

required
True

disabled
False

hidden
False

species

label
Species

type
`basic:string`

description
Select a species name from the dropdown menu or write a custom species name in the species field.

required
True

disabled
False

hidden
False

choices

- Homo sapiens: `Homo sapiens`
- Mus musculus: `Mus musculus`
- Rattus norvegicus: `Rattus norvegicus`

source

label
Protein ID database source

type
`basic:string`

required
True

disabled
False

hidden
False

default
`UniProtKB`

choices

- UniProtKB: `UniProtKB`

Output results table

label
Uploaded table

type
`basic:file`

required
True

disabled
False

hidden
False

species

label
Species

type
`basic:string`

required
True

disabled
False

hidden
False

source

label
Source

type
`basic:string`

required
True

disabled
False

hidden
False

Upload proteomics sample set

`data:proteomics:sampleset:upload-proteomics-sample-set` (*basic:file* **src**, *basic:string* **species**, *basic:string* **source**) [Source: v1.2.1]

Upload a mass spectrometry proteomics sample set file.

The input multi-sample tab-delimited file with the .txt suffix is expected to contain a header line with the following meta-data column names: "Uniprot ID", "Gene symbol", "Protein name" and "Number of peptides". Each additional column in the input file should contain data for a single sample.

Input arguments src

label
Table containing mass spectrometry data (.txt)

type
`basic:file`

required
True

disabled
False

hidden
False

species

label
Species

type
`basic:string`

description
Select a species name from the dropdown menu or write a custom species name in the species field.

required
True

disabled
False

hidden
False

choices

- Homo sapiens: `Homo sapiens`
- Mus musculus: `Mus musculus`
- Rattus norvegicus: `Rattus norvegicus`

source

label
Protein ID database source

type
`basic:string`

required
True

disabled
False

hidden
False

default
`UniProtKB`

choices

- UniProtKB: `UniProtKB`

Output results table

label
Uploaded table

type
`basic:file`

required
True

disabled
False

hidden
False

species

label
Species

type
`basic:string`

required
True

disabled
False

hidden
False

source

label
Source

type
`basic:string`

required
True

disabled
False

hidden
False

VCF file

`data:variants:vcfupload-variants-vcf` (*basic:file* **src**, *basic:string* **species**, *basic:string* **build**) [Source: v2.3.0]

Upload variants in VCF format.

Input arguments src

label
Variants (VCF)

type
`basic:file`

description
Variants in VCF format.

required
True

validate_regex
`\.(vcf)(|\.gz|\.bz2|\.tgz|\.tar\.gz|\.tar\.bz2|\.zip|\.rar|\.7z)$`

species

label
Species

type
`basic:string`

description
Species latin name.

choices

- Homo sapiens: `Homo sapiens`
- Mus musculus: `Mus musculus`
- Rattus norvegicus: `Rattus norvegicus`
- Dictyostelium discoideum: `Dictyostelium discoideum`
- Odocoileus virginianus texanus: `Odocoileus virginianus texanus`
- Solanum tuberosum: `Solanum tuberosum`

build

label
Genome build

type
`basic:string`

Output results vcf

label
Uploaded file

type
`basic:file`

tbi

label
Tabix index

type
`basic:file`

species

label
Species

type
`basic:string`

build

label
Build

type
`basic:string`

Variant calling (CheMut)

`data:variants:vcf:chemut:vc-chemut` (*data:seq:nucleotide* **genome**, *list:data:alignment:bam* **parental_strains**, *list:data:alignment:bam* **mutant_strains**, *basic:boolean* **base_recalibration**, *data:variants:vcf* **known_sites**, *list:data:variants:vcf* **known_indels**, *basic:string* **PL**, *basic:string* **LB**, *basic:string* **PU**, *basic:string* **CN**, *basic:date* **DT**, *data:bed* **intervals**, *basic:integer* **ploidy**, *basic:integer* **stand_call_conf**, *basic:integer* **mbq**, *basic:integer* **max_reads**, *basic:integer* **java_gc_threads**, *basic:integer* **max_heap_size**) [Source: v3.0.1]

CheMut varint calling using multiple BAM input files.

Input arguments genome

label
Reference genome

type
`data:seq:nucleotide`

required
True

disabled
False

hidden
False

parental_strains

label
Parental strains

type
`list:data:alignment:bam`

required
True

disabled
False

hidden
False

mutant_strains

label
Mutant strains

type
`list:data:alignment:bam`

required
True

disabled
False

hidden
False

base_recalibration

label
Do variant base recalibration

type
`basic:boolean`

required
True

disabled
False

hidden
False

default
`False`

known_sites

label
dbSNP file

type
`data:variants:vcf`

description
Database of known polymorphic sites.

required
False

disabled
False

hidden
False

known_indels

label
Known indels

type
`list:data:variants:vcf`

required
False

disabled
False

hidden
!base_recalibration

reads_info.PL

label
Platform/technology

type
`basic:string`

description
Platform/technology used to produce the reads.

required
True

disabled
False

hidden
False

default
`Illumina`

choices

- Capillary: `Capillary`
- Ls454: `Ls454`
- Illumina: `Illumina`
- SOLiD: `SOLiD`
- Helicos: `Helicos`
- IonTorrent: `IonTorrent`
- Pacbio: `Pacbio`

reads_info.LB

label
Library

type
`basic:string`

required
True

disabled
False

hidden
False

default
`x`

reads_info.PU

label
Platform unit

type
`basic:string`

description
Platform unit (e.g. flowcell-barcode.lane for Illumina or slide for SOLiD). Unique identifier.

required
True

disabled
False

hidden
False

default
`x`

reads_info.CN

label
Sequencing center

type
`basic:string`

description
Name of sequencing center producing the read.

required
True

disabled
False

hidden
False

default
`x`

reads_info.DT

label
Date

type
`basic:date`

description
Date the run was produced.

required
True

disabled
False

hidden
False

default
`2017-01-01`

hc.intervals

label
Intervals (from BED file)

type
`data:bed`

description
Use this option to perform the analysis over only part of the genome.

required
False

disabled
False

hidden
False

hc.ploidy

label
Sample ploidy

type
`basic:integer`

description
Ploidy (number of chromosomes) per sample. For pooled data, set to (Number of samples in each pool * Sample Ploidy).

required
True

disabled
False

hidden
False

default
`2`

hc.stand_call_conf

label
Min call confidence threshold

type
`basic:integer`

description
The minimum phred-scaled confidence threshold at which variants should be called.

required
True

disabled
False

hidden
False

default
`30`

hc.mbq

label
Min Base Quality

type
`basic:integer`

description
Minimum base quality required to consider a base for calling.

required
True

disabled
False

hidden
False

default
`10`

hc.max_reads

label
Max reads per alignment start site

type
`basic:integer`

description
Maximum number of reads to retain per alignment start position. Reads above this threshold will be downsampled. Set to 0 to disable.

required
True

disabled
False

hidden
False

default
`50`

advanced.java_gc_threads

label
Java ParallelGCThreads

type
`basic:integer`

description
Sets the number of threads used during parallel phases of the garbage collectors.

required
True

disabled
False

hidden
False

default
`2`

advanced.max_heap_size

label
Java maximum heap size (Xmx)

type
`basic:integer`

description
Set the maximum Java heap size (in GB).

required
True

disabled
False

hidden
False

default
`12`

Output results vcf

label
Called variants file

type
`basic:file`

required
True

disabled
False

hidden
False

tbi

label
Tabix index

type
`basic:file`

required
True

disabled
False

hidden
False

species

label
Species

type
`basic:string`

required
True

disabled
False

hidden
False

build

label
Build

type
`basic:string`

required
True

disabled
False

hidden
False

Variant filtering (CheMut)

`data:variants:vcf:filtering:filtering-chemut` (*data:variants:vcf* **variants**, *basic:string* **analysis_type**, *basic:string* **parental_strain**, *basic:string* **mutant_strain**, *data:seq:nucleotide* **genome**, *basic:integer* **read_depth**) [Source: v1.8.2]

Filtering and annotation of Variant Calling (CheMut).

Filtering and annotation of Variant Calling data - Chemical mutagenesis in _Dictyostelium discoideum_.

Input arguments variants

label
Variants file (VCF)

type
`data:variants:vcf`

required
True

disabled
False

hidden
False

analysis_type

label
Analysis type

type
`basic:string`

description
Choice of the analysis type. Use 'SNV' or 'INDEL' options. Choose options SNV_CHR2 or INDEL_CHR2 to run the GATK analysis only on the diploid portion of CHR2 (-ploidy 2 -L chr2:2263132-3015703).

required
True

disabled
False

hidden
False

default
`snv`

choices

- SNV: `snv`
- INDEL: `indel`
- SNV_CHR2: `snv_chr2`
- INDEL_CHR2: `indel_chr2`

parental_strain

label
Parental strain prefix

type
`basic:string`

required
True

disabled
False

hidden
False

default
`parental`

mutant_strain

label
Mutant strain prefix

type
`basic:string`

required
True

disabled
False

hidden
False

default
`mut`

genome

label
Reference genome

type
`data:seq:nucleotide`

required
True

disabled
False

hidden
False

read_depth

label
Read Depth Cutoff

type
`basic:integer`

required
True

disabled
False

hidden
False

default
`5`

Output results summary

label
Summary

type
`basic:file`

description
Summarize the input parameters and results.

required
True

disabled
False

hidden
False

vcf

label
Variants

type
`basic:file`

description
A genome VCF file of variants that passed the filters.

required
True

disabled
False

hidden
False

tbi

label
Tabix index

type
`basic:file`

required
True

disabled
False

hidden
False

variants_filtered

label
Variants filtered

type
`basic:file`

description
A data frame of variants that passed the filters.

required
False

disabled
False

hidden
False

variants_filtered_alt

label
Variants filtered (multiple alt. alleles)

type
`basic:file`

description
A data frame of variants that contain more than two alternative alleles. These variants are likely to be false positives.

required
False

disabled
False

hidden
False

gene_list_all

label
Gene list (all)

type
`basic:file`

description
Genes that are mutated at least once.

required
False

disabled
False

hidden
False

gene_list_top

label
Gene list (top)

type
`basic:file`

description
Genes that are mutated at least twice.

required
False

disabled
False

hidden
False

mut_chr

label
Mutations (by chr)

type
`basic:file`

description
List mutations in individual chromosomes.

required
False

disabled
False

hidden
False

mut_strain

label
Mutations (by strain)

type
`basic:file`

description
List mutations in individual strains.

required
False

disabled
False

hidden
False

strain_by_gene

label
Strain (by gene)

type
`basic:file`

description
List mutants that carry mutations in individual genes.

required
False

disabled
False

hidden
False

species

label
Species

type
`basic:string`

required
True

disabled
False

hidden
False

build

label
Build

type
`basic:string`

required
True

disabled
False

hidden
False

WALT

`data:alignment:bam:waltwalt` (*data:index:walt* **genome**, *data:reads:fastq* **reads**, *basic:boolean* **rm_dup**, *basic:integer* **optical_distance**, *basic:integer* **mismatch**, *basic:integer* **number**, *basic:string* **spikein_name**, *basic:boolean* **filter_spikein**) [Source: v3.7.2]

WALT (Wildcard ALignment Tool) is a read mapping program for bisulfite sequencing in DNA methylation studies.

Input arguments genome

label
Reference genome

type
`data:index:walt`

reads

label
Reads

type
`data:reads:fastq`

rm_dup

label
Remove duplicates

type
`basic:boolean`

default
`True`

optical_distance

label
Optical duplicate distance

type
`basic:integer`

description
The maximum offset between two duplicate clusters in order to consider them optical duplicates. Suggested settings of 100 for HiSeq style platforms or about 2500 for NovaSeq ones. Default is 0 to not look for optical duplicates.

disabled
!rm_dup

default
`0`

mismatch

label
Maximum allowed mismatches

type
`basic:integer`

required
False

number

label
Number of reads to map in one loop

type
`basic:integer`

description
Sets the number of reads to mapping in each loop. Larger number results in program taking more memory. This is especially evident in paired-end mapping.

required
False

spikein_options.spikein_name

label
Chromosome name of unmethylated control sequence

type
`basic:string`

description
Specifies the name of unmethylated control sequence which is output as a separate alignment file. It is recomended to remove duplicates to reduce any bias introduced by incomplete conversion on PCR duplicate reads.

required
False

spikein_options.filter_spikein

label
Remove control/spike-in sequences.

type
`basic:boolean`

description
Remove unmethylated control reads in the final alignment based on the provided name. It is recomended to remove any reads that are not naturally occuring in the sample (e.g. lambda virus spike-in).

disabled
!spikein_options.spikein_name

default
`False`

Output results bam

label
Alignment file (BAM)

type
`basic:file`

description
Position sorted alignment in .bam format

bai

label
Index BAI

type
`basic:file`

stats

label
Statistics

type
`basic:file`

mr

label
Alignment file (MR)

type
`basic:file`

description
Position sorted alignment in .mr format.

duplicates_report

label
Removed duplicates statistics

type
`basic:file`

required
False

unmapped

label
Unmapped reads

type
`basic:file`

required
False

spikein_mr

label
Alignment file of unmethylated control reads

type
`basic:file`

required
False

species

label
Species

type
`basic:string`

build

label
Build

type
`basic:string`

WALT genome index

`data:index:walt:walt-index` (*data:seq:nucleotide* **ref_seq**)[Source: v1.2.1]

Create WALT genome index.

Input arguments ref_seq

label
Reference sequence (nucleotide FASTA)

type
`data:seq:nucleotide`

required
True

disabled
False

hidden
False

Output results index

label
WALT index

type
`basic:dir`

required
True

disabled
False

hidden
False

fastagz

label
FASTA file (compressed)

type
`basic:file`

required
True

disabled
False

hidden
False

fasta

label
FASTA file

type
`basic:file`

required
True

disabled
False

hidden
False

fai

label
FASTA file index

type
`basic:file`

required
True

disabled
False

hidden
False

species

label
Species

type
basic:string

required
True

disabled
False

hidden
False

build

label
Build

type
basic:string

required
True

disabled
False

hidden
False

WGBS (paired-end)

`data:workflow:wgbsworkflow-wgbs-paired` (*data:reads:fastq:paired* **reads**, *data:index:walt* **walt_index**, *data:seq:nucleotide* **ref_seq**, *basic:string* **validation_stringency**, *data:seq:nucleotide* **adapters**, *basic:integer* **seed_mismatches**, *basic:integer* **simple_clip_threshold**, *basic:integer* **min_adapter_length**, *basic:integer* **palindrome_clip_threshold**, *basic:boolean* **keep_both_reads**, *basic:integer* **leading**, *basic:integer* **trailing**, *basic:integer* **crop**, *basic:integer* **headcrop**, *basic:integer* **minlen**, *basic:boolean* **rm_dup**, *basic:integer* **optical_distance**, *basic:integer* **mismatch**, *basic:integer* **number**, *basic:string* **spikein_name**, *basic:boolean* **filter_spikein**, *basic:boolean* **skip**, *data:seq:nucleotide* **sequence**, *basic:boolean* **count_all**, *basic:integer* **read_length**, *basic:decimal* **max_mismatch**, *basic:boolean* **a_rich**, *basic:boolean* **cpgs**, *basic:boolean* **symmetric_cpgs**, *data:seq:nucleotide* **adapters**, *basic:integer* **insert_size**, *basic:string* **pair_orientation**, *basic:integer* **read_length**, *basic:integer* **min_map_quality**, *basic:integer* **min_quality**, *basic:integer* **coverage_cap**, *basic:integer* **accumulation_cap**, *basic:integer* **sample_size**, *basic:integer* **min_quality**, *basic:integer* **next_base_quality**, *basic:integer* **min_lenght**, *basic:decimal* **mismatch_rate**, *basic:decimal* **minimum_fraction**, *basic:boolean* **include_duplicates**, *basic:decimal* **deviations**)[Source: v2.2.0]

This WGBS pipeline is comprised of trimming, alignment, computation of methylation levels, identification of hypo-methylated regions (HMRs) and additional QC steps.

First, reads are trimmed to remove adapters or kit specific artifacts. Reads are then aligned by __WALT__ aligner. [WALT (Wildcard ALignment Tool)](https://github.com/smithlabcode/walt) is fast and accurate read mapping for bisulfite sequencing. Then, methylation level at each genomic cytosine is calculated using __methcounts__. Finally, hypo-methylated regions are identified using __hmr__. Both methcounts and hmr are part of [MethPipe](http://smithlabresearch.org/software/methpipe/) package.

QC steps are based on [Picard](http://broadinstitute.github.io/picard/) and include high level metrics about the alignment, WGS performance and summary statistics from bisulfite sequencing. Final QC reports are summarized by MultiQC.

Input arguments reads

label
Select sample(s)

type
`data:reads:fastq:paired`

walt_index

label
Walt index

type
`data:index:walt`

ref_seq

label
Reference sequence

type
`data:seq:nucleotide`

validation_stringency

label
Validation stringency

type
`basic:string`

description
Validation stringency for all SAM files read by this program. Setting stringency to SILENT can improve performance when processing a BAM file in which variable-length data (read, qualities, tags) do not otherwise need to be decoded. Default is STRICT.

default
`STRICT`

choices

- STRICT: `STRICT`
- LENIENT: `LENIENT`
- SILENT: `SILENT`

adapter_trimming.adapters

label
Adapter sequences

type
`data:seq:nucleotide`

description
Adapter sequence in FASTA format that will be removed from the read. This field as well as 'Seed mismatches', 'Simple clip threshold' and 'Palindrome clip threshold' parameters are needed to perform adapter trimming. 'Minimum adapter length' and 'Keep both reads' are optional parameters.

required
False

adapter_trimming.seed_mismatches

label
Seed mismatches

type
`basic:integer`

description
Specifies the maximum mismatch count which will still allow a full match to be performed. This field is required to perform adapter trimming.

required
False

disabled
!adapter_trimming.adapters

adapter_trimming.simple_clip_threshold

label
Simple clip threshold

type
`basic:integer`

description
Specifies how accurate the match between any adapter etc. sequence must be against a read. This field is required to perform adapter trimming.

required
False

disabled
!adapter_trimming.adapters

adapter_trimming.min_adapter_length

label
Minimum adapter length

type
`basic:integer`

description
In addition to the alignment score, palindrome mode can verify that a minimum length of adapter has been detected. If unspecified, this defaults to 8 bases, for historical reasons. However, since palindrome mode has a very low false positive rate, this can be safely reduced, even down to 1, to allow shorter adapter fragments to be removed.

disabled
!adapter_trimming.seed_mismatches && !adapter_trimming.simple_clip_threshold && !adapter_trimming.palindrome_clip_threshold

default
8

adapter_trimming.palindrome_clip_threshold

label
Palindrome clip threshold

type
`basic:integer`

description
Specifies how accurate the match between the two 'adapter ligated' reads must be for PE palindrome read alignment. This field is required to perform adapter trimming.

required
False

disabled
!adapter_trimming.adapters

adapter_trimming.keep_both_reads

label
Keep both reads

type
`basic:boolean`

description
After read-though has been detected by palindrome mode, and the adapter sequence removed, the reverse read contains the same sequence information as the forward read, albeit in reverse complement. For this reason, the default behaviour is to entirely drop the reverse read. By specifying this parameter, the reverse read will also be retained, which may be useful e.g. if the downstream tools cannot handle a combination of paired and unpaired reads. This field is optional for preforming adapter trimming.

required
False

disabled
!adapter_trimming.seed_mismatches && !adapter_trimming.simple_clip_threshold && !adapter_trimming.palindrome_clip_threshold && !adapter_trimming.min_adapter_length

trimming_filtering.leading

label
Leading quality

type
`basic:integer`

description
Remove low quality bases from the beginning, if below a threshold quality.

required
False

trimming_filtering.trailing

label
Trailing quality

type
`basic:integer`

description
Remove low quality bases from the end, if below a threshold quality.

required
False

trimming_filtering.crop

label
Crop

type
`basic:integer`

description
Cut the read to a specified length by removing bases from the end.

required
False

trimming_filtering.headcrop

label
Headcrop

type
`basic:integer`

description
Cut the specified number of bases from the start of the read.

required
False

trimming_filtering.minlen

label
Minimum length

type
`basic:integer`

description
Drop the read if it is below a specified length.

required
False

alignment.rm_dup

label
Remove duplicates

type
`basic:boolean`

default
`True`

alignment.optical_distance

label
Optical duplicate distance

type
`basic:integer`

description
The maximum offset between two duplicate clusters in order to consider them optical duplicates. Suggested settings of 100 for HiSeq style platforms or about 2500 for NovaSeq ones. Default is 0 to not look for optical duplicates.

disabled
!alignment.rm_dup

default
`0`

alignment.mismatch

label
Maximum allowed mismatches

type
`basic:integer`

default
`6`

alignment.number

label
Number of reads to map in one loop

type
`basic:integer`

description
Sets the number of reads to mapping in each loop. Larger number results in program taking more memory. This is especially evident in paired-end mapping.

required
False

alignment.spikein_name

label
Chromosome name of unmethylated control sequence

type
`basic:string`

description
Specifies the name of unmethylated control sequence which is output as a separate alignment file. It is recomended to remove duplicates to reduce any bias introduced by incomplete conversion on PCR duplicate reads.

required
False

alignment.filter_spikein

label
Remove control/spike-in sequences.

type
`basic:boolean`

description
Remove unmethylated control reads in the final alignment based on the provided name. It is recomended to remove any reads that are not naturally occuring in the sample (e.g. lambda virus spike-in).

disabled
!alignment.spikein_name

default
`False`

bsrate.skip

label
Skip Bisulfite conversion rate step

type
`basic:boolean`

description
Bisulfite conversion rate step can be skipped. If separate alignment file for unmethylated control sequence is not produced during the alignment this process will fail.

disabled
!alignment.spikein_name

default
`True`

bsrate.sequence

label
Unmethylated control sequence

type
`data:seq:nucleotide`

required
False

disabled
bsrate.skip

bsrate.count_all

label
Count all cytosines including CpGs

type
`basic:boolean`

disabled
bsrate.skip

default
`True`

bsrate.read_length

label
Average read length

type
`basic:integer`

default
`150`

bsrate.max_mismatch

label
Maximum fraction of mismatches

type
`basic:decimal`

required
False

disabled
bsrate.skip

bsrate.a_rich

label
Reads are A-rich

type
`basic:boolean`

disabled
bsrate.skip

default
`False`

methcounts.cpgs

label
Only CpG context sites

type
`basic:boolean`

description
Output file will contain methylation data for CpG context sites only. Choosing this option will result in CpG content report only.

disabled
methcounts.symmetric_cpgs

default
`False`

methcounts.symmetric_cpgs

label
Merge CpG pairs

type
`basic:boolean`

description
Merging CpG pairs results in symmetric methylation levels. Methylation is usually symmetric (cytosines at CpG sites were methylated on both DNA strands). Choosing this option will only keep the CpG sites data.

disabled
methcounts.cpgs

default
`True`

summary.adapters

label
Adapter sequences

type
`data:seq:nucleotide`

required
False

summary.insert_size

label
Maximum insert size

type
`basic:integer`

default
`100000`

summary.pair_orientation

label
Pair orientation

type
`basic:string`

default
`null`

choices

- Unspecified: `null`
- FR: `FR`
- RF: `RF`
- TANDEM: `TANDEM`

wgs_metrics.read_length

label
Average read length

type
`basic:integer`

default
`150`

wgs_metrics.min_map_quality

label
Minimum mapping quality for a read to contribute coverage

type
`basic:integer`

default
`20`

wgs_metrics.min_quality

label
Minimum base quality for a base to contribute coverage

type
`basic:integer`

description
N bases will be treated as having a base quality of negative infinity and will therefore be excluded from coverage regardless of the value of this parameter.

default
`20`

wgs_metrics.coverage_cap

label
Maximum coverage cap

type
`basic:integer`

description
Treat positions with coverage exceeding this value as if they had coverage at this set value.

default
`250`

wgs_metrics.accumulation_cap

label
Ignore positions with coverage above this value

type
`basic:integer`

description
At positions with coverage exceeding this value, completely ignore reads that accumulate beyond this value

default
`100000`

wgs_metrics.sample_size

label
Sample Size used for Theoretical Het Sensitivity sampling

type
`basic:integer`

default
`10000`

rrbs_metrics.min_quality

label
Threshold for base quality of a C base before it is considered

type
`basic:integer`

default
`20`

rrbs_metrics.next_base_quality

label
Threshold for quality of a base next to a C before the C base is considered

type
`basic:integer`

default
`10`

rrbs_metrics.min_lenght

label
Minimum read length

type
`basic:integer`

default
`5`

rrbs_metrics.mismatch_rate

label
Maximum fraction of mismatches in a read to be considered (Between 0 and 1)

type
`basic:decimal`

default
`0.1`

insert.minimum_fraction

label
Minimum fraction of reads in a category to be considered

type
`basic:decimal`

description
When generating the histogram, discard any data categories (out of FR, TANDEM, RF) that have fewer than this fraction of overall reads (Range: 0 and 0.5).

default
`0.05`

insert.include_duplicates

label
Include reads marked as duplicates in the insert size histogram

type
`basic:boolean`

default
`False`

insert.deviations

label
Deviations limit

type
`basic:decimal`

description
Generate mean, standard deviation and plots by trimming the data down to MEDIAN + DEVIATIONS*MEDIAN_ABSOLUTE_DEVIATION. This is done because insert size data typically includes enough anomalous values from chimeras and other artifacts to make the mean and standard deviation grossly misleading regarding the real distribution.

default
`10.0`

Output results

WGBS (single-end)

`data:workflow:wgbsworkflow-wgbs-single` (*data:reads:fastq:single* **reads**, *data:index:walt* **walt_index**, *data:seq:nucleotide* **ref_seq**, *basic:string* **validation_stringency**, *data:seq:nucleotide* **adapters**, *basic:integer* **seed_mismatches**, *basic:integer* **simple_clip_threshold**, *basic:integer* **leading**, *basic:integer* **trailing**, *basic:integer* **crop**, *basic:integer* **headcrop**, *basic:integer* **minlen**, *basic:boolean* **rm_dup**, *basic:integer* **optical_distance**, *basic:integer* **mismatch**, *basic:integer* **number**, *basic:string* **spikein_name**, *basic:boolean* **filter_spikein**, *basic:boolean* **skip**, *data:seq:nucleotide* **sequence**, *basic:boolean* **count_all**, *basic:integer* **read_length**, *basic:decimal* **max_mismatch**, *basic:boolean* **a_rich**, *basic:boolean* **cpgs**, *basic:boolean* **symmetric_cpgs**, *data:seq:nucleotide* **adapters**, *basic:integer* **insert_size**, *basic:string* **pair_orientation**, *basic:integer* **read_length**, *basic:integer* **min_map_quality**, *basic:integer* **min_quality**, *basic:integer* **coverage_cap**, *basic:integer* **accumulation_cap**, *basic:integer* **sample_size**, *basic:integer* **min_quality**, *basic:integer* **next_base_quality**, *basic:integer* **min_lenght**, *basic:decimal* **mismatch_rate**) [Source: v2.2.0]

This WGBS pipeline is comprised of trimming, alignment, computation of methylation levels, identification of hypomethylated regions (HMRs) and additional QC steps.

First, reads are trimmed to remove adapters or kit specific artifacts. Reads are then aligned by __WALT__ aligner. [WALT (Wildcard ALignment Tool)](https://github.com/smithlabcode/walt) is fast and accurate read mapping for bisulfite sequencing. Then, methylation level at each genomic cytosine is calculated using __methcounts__. Finally, hypo-methylated regions are identified using __hmr__. Both methcounts and hmr are part of [MethPipe](http://smithlabresearch.org/software/methpipe/) package.

QC steps are based on [Picard](http://broadinstitute.github.io/picard/) and include high level metrics about the alignment, WGS performance and summary statistics from bisulfite sequencing. Final QC reports are summarized by Mul-

tiQC.

Input arguments reads

label
Select sample(s)

type
`data:reads:fastq:single`

walt_index

label
Walt index

type
`data:index:walt`

ref_seq

label
Reference sequence

type
`data:seq:nucleotide`

validation_stringency

label
Validation stringency

type
`basic:string`

description
Validation stringency for all SAM files read by this program. Setting stringency to SILENT can improve performance when processing a BAM file in which variable-length data (read, qualities, tags) do not otherwise need to be decoded. Default is STRICT.

default
`STRICT`

choices

- STRICT: `STRICT`
- LENIENT: `LENIENT`
- SILENT: `SILENT`

adapter_trimming.adapters

label
Adapter sequences

type
`data:seq:nucleotide`

description
Adapter sequence in FASTA format that will be removed from the read. This field as well as 'Seed mismatches' and 'Simple clip threshold' parameters are needed to perform adapter trimming.

required
False

adapter_trimming.seed_mismatches

label
Seed mismatches

type
`basic:integer`

description
Specifies the maximum mismatch count which will still allow a full match to be performed. This field is required to perform adapter trimming.

required
False

disabled
!adapter_trimming.adapters

adapter_trimming.simple_clip_threshold

label
Simple clip threshold

type
`basic:integer`

description
Specifies how accurate the match between any adapter etc. sequence must be against a read. This field is required to perform adapter trimming.

required
False

disabled
!adapter_trimming.adapters

trimming_filtering.leading

label
Leading quality

type
`basic:integer`

description
Remove low quality bases from the beginning, if below a threshold quality.

required
False

trimming_filtering.trailing

label
Trailing quality

type
`basic:integer`

description
Remove low quality bases from the end, if below a threshold quality.

required
False

trimming_filtering.crop

label
Crop

type
`basic:integer`

description
Cut the read to a specified length by removing bases from the end.

required
False

trimming_filtering.headcrop

label
Headcrop

type
`basic:integer`

description
Cut the specified number of bases from the start of the read.

required
False

trimming_filtering.minlen

label
Minimum length

type
`basic:integer`

description
Drop the read if it is below a specified length.

required
False

alignment.rm_dup

label
Remove duplicates

type
`basic:boolean`

default
`True`

alignment.optical_distance

label
Optical duplicate distance

type
`basic:integer`

description
The maximum offset between two duplicate clusters in order to consider them optical duplicates. Suggested settings of 100 for HiSeq style platforms or about 2500 for NovaSeq ones. Default is 0 to not look for optical duplicates.

disabled
!alignment.rm_dup

default
`0`

alignment.mismatch

label
Maximum allowed mismatches

type
`basic:integer`

default
6

alignment.number

label
Number of reads to map in one loop

type
`basic:integer`

description
Sets the number of reads to mapping in each loop. Larger number results in program taking more memory. This is especially evident in paired-end mapping.

required
False

alignment.spikein_name

label
Chromosome name of unmethylated control sequence

type
`basic:string`

description
Specifies the name of unmethylated control sequence which is output as a separate alignment file. It is recomended to remove duplicates to reduce any bias introduced by incomplete conversion on PCR duplicate reads.

required
False

alignment.filter_spikein

label
Remove control/spike-in sequences.

type
`basic:boolean`

description
Remove unmethylated control reads in the final alignment based on the provided name. It is recomended to remove any reads that are not naturally occuring in the sample (e.g. lambda virus spike-in).

disabled
!alignment.spikein_name

default
`False`

bsrate.skip

label
Skip Bisulfite conversion rate step

type
`basic:boolean`

description
Bisulfite conversion rate step can be skipped. If separate alignment file for unmethylated control sequence is not produced during the alignment this process will fail.

disabled
!alignment.spikein_name

default
`True`

bsrate.sequence

label
Unmethylated control sequence

type
`data:seq:nucleotide`

required
False

disabled
bsrate.skip

bsrate.count_all

label
Count all cytosines including CpGs

type
`basic:boolean`

disabled
bsrate.skip

default
`True`

bsrate.read_length

label
Average read length

type
`basic:integer`

default
`150`

bsrate.max_mismatch

label
Maximum fraction of mismatches

type
`basic:decimal`

required
False

disabled
bsrate.skip

bsrate.a_rich

label
Reads are A-rich

type
`basic:boolean`

disabled
bsrate.skip

default
`False`

methcounts.cpgs

label
Only CpG context sites

type
`basic:boolean`

description
Output file will contain methylation data for CpG context sites only. Choosing this option will result in CpG content report only.

disabled
methcounts.symmetric_cpgs

default
`False`

methcounts.symmetric_cpgs

label
Merge CpG pairs

type
`basic:boolean`

description
Merging CpG pairs results in symmetric methylation levels. Methylation is usually symmetric (cytosines at CpG sites were methylated on both DNA strands). Choosing this option will only keep the CpG sites data.

disabled
methcounts.cpgs

default
`True`

summary.adapters

label
Adapter sequences

type
`data:seq:nucleotide`

required
False

summary.insert_size

label
Maximum insert size

type
`basic:integer`

default
`100000`

summary.pair_orientation

label
Pair orientation

type
`basic:string`

default
`null`

choices

- Unspecified: `null`
- FR: `FR`
- RF: `RF`
- TANDEM: `TANDEM`

wgs_metrics.read_length

label
Average read length

type
`basic:integer`

default
`150`

wgs_metrics.min_map_quality

label
Minimum mapping quality for a read to contribute coverage

type
`basic:integer`

default
`20`

wgs_metrics.min_quality

label
Minimum base quality for a base to contribute coverage

type
`basic:integer`

description
N bases will be treated as having a base quality of negative infinity and will therefore be excluded from coverage regardless of the value of this parameter.

default
`20`

wgs_metrics.coverage_cap

label
Maximum coverage cap

type
`basic:integer`

description
Treat positions with coverage exceeding this value as if they had coverage at this set value.

default
`250`

wgs_metrics.accumulation_cap

label
Ignore positions with coverage above this value

type
`basic:integer`

description
At positions with coverage exceeding this value, completely ignore reads that accumulate beyond this value

default
`100000`

wgs_metrics.sample_size

label
Sample Size used for Theoretical Het Sensitivity sampling

type
`basic:integer`

default
`10000`

rrbs_metrics.min_quality

label
Threshold for base quality of a C base before it is considered

type
`basic:integer`

default
`20`

rrbs_metrics.next_base_quality

label
Threshold for quality of a base next to a C before the C base is considered

type
basic:integer

default
10

rrbs_metrics.min_lenght

label
Minimum read length

type
basic:integer

default
5

rrbs_metrics.mismatch_rate

label
Maximum fraction of mismatches in a read to be considered (Between 0 and 1)

type
basic:decimal

default
0.1

Output results

WGS (paired-end) analysis

`data:workflow:wgsworkflow-wgs-paired` (*data:reads:fastq:paired* **reads**, *data:index:bwa* **bwa_index**, *data:seq:nucleotide* **ref_seq**, *list:data:variants:vcf* **known_sites**, *data:variants:vcf* **hc_dbsnp**, *basic:string* **validation_stringency**, *data:seq:nucleotide* **adapters**, *basic:integer* **seed_mismatches**, *basic:integer* **simple_clip_threshold**, *basic:integer* **min_adapter_length**, *basic:integer* **palindrome_clip_threshold**, *basic:integer* **leading**, *basic:integer* **trailing**, *basic:integer* **minlen**, *basic:integer* **seed_l**, *basic:integer* **band_w**, *basic:decimal* **re_seeding**, *basic:boolean* **m**, *basic:integer* **match**, *basic:integer* **mismatch**, *basic:integer* **gap_o**, *basic:integer* **gap_e**, *basic:integer* **clipping**, *basic:integer* **unpaired_p**, *basic:integer* **report_tr**, *basic:boolean* **skip**, *basic:boolean* **remove_duplicates**, *basic:string* **assume_sort_order**, *basic:string* **read_group**, *data:seq:nucleotide* **adapters**, *basic:integer* **max_insert_size**, *basic:string* **pair_orientation**, *basic:integer* **read_length**, *basic:integer* **min_map_quality**, *basic:integer* **min_quality**, *basic:integer* **coverage_cap**, *basic:integer* **accumulation_cap**, *basic:integer* **sample_size**, *basic:decimal* **minimum_fraction**, *basic:boolean* **include_duplicates**, *basic:decimal* **deviations**, *basic:integer* **stand_call_conf**, *basic:integer* **mbq**) [Source: v2.1.0]

Whole genome sequencing pipeline analyses paired-end whole genome sequencing data. It consists of trimming, aligning, marking of duplicates, Picard metrics, recalibration of base quality scores and finally, calling of variants.

The tools used are Trimmomatic which performs trimming. Aligning is performed using BWA (mem). Marking of duplicates (MarkDuplicates), Picard metrics (AlignmentSummaryMetrics, CollectWgsMetrics and InsertSizeMetrics), recalibration of base quality scores (ApplyBQSR) and calling of variants (HaplotypeCaller) is done using GATK4 bundle of bioinformatics tools. Result is a file of called variants (VCF).

Input arguments reads

label
Raw untrimmed reads

type
`data:reads:fastq:paired`

description
Raw paired-end reads.

bwa_index

label
Genome index (BWA)

type
`data:index:bwa`

description
BWA genome index.

ref_seq

label
Reference genome sequence

type
`data:seq:nucleotide`

known_sites

label
Known sites of variation used in BQSR

type
`list:data:variants:vcf`

description
Known sites of variation as a VCF file.

hc_dbsnp

label
dbSNP for GATK4's HaplotypeCaller

type
`data:variants:vcf`

description
dbSNP database of variants for variant calling.

validation_stringency

label
Validation stringency

type
`basic:string`

description
Validation stringency for all BAM files read by this program. Setting stringency to SILENT can improve performance when processing a BAM file in which variable-length data (read, qualities, tags) do not otherwise need to be decoded. Default is STRICT.

default
`STRICT`

choices

- STRICT: `STRICT`
- LENIENT: `LENIENT`
- SILENT: `SILENT`

advanced.trimming.adapters

label
Adapter sequences

type
`data:seq:nucleotide`

description
Adapter sequence in FASTA format that will be removed from the read. This field as well as 'Seed mismatches', 'Simple clip threshold' and 'Palindrome clip threshold' parameters are needed to perform adapter trimming. 'Minimum adapter length' and 'Keep both reads' are optional parameters.

required
False

advanced.trimming.seed_mismatches

label
Seed mismatches

type
`basic:integer`

description
Specifies the maximum mismatch count which will still allow a full match to be performed. This field is required to perform adapter trimming.

required
False

disabled
!advanced.trimming.adapters

advanced.trimming.simple_clip_threshold

label
Simple clip threshold

type
`basic:integer`

description
Specifies how accurate the match between any adapter etc. sequence must be against a read. This field is required to perform adapter trimming.

required
False

disabled
!advanced.trimming.adapters

advanced.trimming.min_adapter_length

label
Minimum adapter length

type
`basic:integer`

description
In addition to the alignment score, palindrome mode can verify that a minimum length of adapter has been detected. If unspecified, this defaults to 8 bases, for historical reasons. However, since palindrome mode has a very low false positive rate, this can be safely reduced, even down to 1, to allow shorter adapter fragments to be removed.

disabled
!advanced.trimming.seed_mismatches && !advanced.trimming.simple_clip_threshold && !advanced.trimming.palindrome_clip_threshold

default
8

advanced.trimming.palindrome_clip_threshold

label
Palindrome clip threshold

type
`basic:integer`

description
Specifies how accurate the match between the two 'adapter ligated' reads must be for PE palindrome read alignment. This field is required to perform adapter trimming.

required
False

disabled
!advanced.trimming.adapters

advanced.trimming.leading

label
Leading quality

type
`basic:integer`

description
Remove low quality bases from the beginning, if below a threshold quality.

required
False

advanced.trimming.trailing

label
Trailing quality

type
`basic:integer`

description
Remove low quality bases from the end, if below a threshold quality.

required
False

advanced.trimming.minlen

label
Minimum length

type
`basic:integer`

description
Drop the read if it is below a specified length.

required
False

advanced.align.seed_l

label
Minimum seed length

type
`basic:integer`

description
Minimum seed length. Matches shorter than minimum seed length will be missed. The alignment speed is usually insensitive to this value unless it significantly deviates from 20.

default
`19`

advanced.align.band_w

label
Band width

type
`basic:integer`

description
Gaps longer than this will not be found.

default
`100`

advanced.align.re_seeding

label
Re-seeding factor

type
`basic:decimal`

description
Trigger re-seeding for a MEM longer than minSeedLen*FACTOR. This is a key heuristic parameter for tuning the performance. Larger value yields fewer seeds, which leads to faster alignment speed but lower accuracy.

default
`1.5`

advanced.align.m

label
Mark shorter split hits as secondary

type
`basic:boolean`

description
Mark shorter split hits as secondary (for Picard compatibility)

default
`False`

advanced.align.scoring.match

label
Score of a match

type
`basic:integer`

default
1

advanced.align.scoring.mismatch

label
Mismatch penalty

type
`basic:integer`

default
4

advanced.align.scoring.gap_o

label
Gap open penalty

type
`basic:integer`

default
6

advanced.align.scoring.gap_e

label
Gap extension penalty

type
`basic:integer`

default
1

advanced.align.scoring.clipping

label
Clipping penalty

type
`basic:integer`

description
Clipping is applied if final alignment score is smaller than (best score reaching the end of query) - (Clipping penalty)

default
`5`

advanced.align.scoring.unpaired_p

label
Penalty for an unpaired read pair

type
`basic:integer`

description
Affinity to force pair. Score: scoreRead1+ scoreRead2-Penalty

default
`9`

advanced.align.report_tr

label
Report threshold score

type
`basic:integer`

description
Don't output alignment with score lower than defined number. This option only affects output.

default
`30`

advanced.markduplicates.skip

label
Skip GATK's MarkDuplicates step

type
`basic:boolean`

default
`False`

advanced.markduplicates.remove_duplicates

label
Remove found duplicates

type
`basic:boolean`

default
`False`

advanced.markduplicates.assume_sort_order

label
Assume sort oder

type
`basic:string`

default

choices

- as in BAM header (default):
- unsorted: `unsorted`
- queryname: `queryname`
- coordinate: `coordinate`
- duplicate: `duplicate`
- unknown: `unknown`

advanced.bqsr.read_group

label
Read group (@RG)

type
`basic:string`

description
This argument enables the user to replace all read groups in the INPUT file with a single new read group and assign all reads to this read group in the OUTPUT BAM file. Addition or replacement is performed using Picard's AddOrReplaceReadGroups tool. Input should take the form of -name=value delimited by a \t, e.g. "-ID=1\t-PL=Illumina\t-SM=sample_1". See AddOrReplaceReadGroups documentation for more information on tag names. Note that PL, LB, PU and SM are required fields.

default
`-LB=NA;-PL=NA;-PU=NA;-SM=sample`

advanced.summary.adapters

label
Adapter sequences

type
`data:seq:nucleotide`

required
False

advanced.summary.max_insert_size

label
Maximum insert size

type
`basic:integer`

default
`100000`

advanced.summary.pair_orientation

label
Pair orientation

type
`basic:string`

default
`null`

choices

- Unspecified: `null`
- FR: `FR`
- RF: `RF`
- TANDEM: `TANDEM`

advanced.wgs_metrics.read_length

label
Average read length

type
`basic:integer`

default
`150`

advanced.wgs_metrics.min_map_quality

label
Minimum mapping quality for a read to contribute coverage

type
`basic:integer`

default
`20`

advanced.wgs_metrics.min_quality

label
Minimum base quality for a base to contribute coverage

type
`basic:integer`

description
N bases will be treated as having a base quality of negative infinity and will therefore be excluded from coverage regardless of the value of this parameter.

default
`20`

advanced.wgs_metrics.coverage_cap

label
Maximum coverage cap

type
`basic:integer`

description
Treat positions with coverage exceeding this value as if they had coverage at this set value.

default
`250`

advanced.wgs_metrics.accumulation_cap

label
Ignore positions with coverage above this value

type
`basic:integer`

description
At positions with coverage exceeding this value, completely ignore reads that accumulate beyond this value.

default
`100000`

advanced.wgs_metrics.sample_size

label
Sample Size used for Theoretical Het Sensitivity sampling

type
`basic:integer`

default
`10000`

advanced.insert_size.minimum_fraction

label
Minimum fraction of reads in a category to be considered

type
`basic:decimal`

description
When generating the histogram, discard any data categories (out of FR, TANDEM, RF) that have fewer than this fraction of overall reads (Range: 0 and 0.5).

default
`0.05`

advanced.insert_size.include_duplicates

label
Include reads marked as duplicates in the insert size histogram

type
`basic:boolean`

default
`False`

advanced.insert_size.deviations

label
Deviations limit

type
`basic:decimal`

description
Generate mean, standard deviation and plots by trimming the data down to MEDIAN + DEVIATIONS * MEDIAN_ABSOLUTE_DEVIATION. This is done because insert size data typically includes enough anomalous values from chimeras and other artifacts to make the mean and standard deviation grossly misleading regarding the real distribution.

default
`10.0`

advanced.hc.stand_call_conf

label
Min call confidence threshold

type
`basic:integer`

description
The minimum phred-scaled confidence threshold at which variants should be called.

default
`20`

advanced.hc.mbq

label
Min Base Quality

type
`basic:integer`

description
Minimum base quality required to consider a base for calling.

default
`20`

Output results

WGS analysis (GVCF)

`data:workflow:wgs:gvcf:workflow-wgs-gvcf` (*data:reads:fastq:paired* **reads**,
data:alignment:bam **aligned_reads**,
data:seq:nucleotide **ref_seq**,
data:index:bwamem2 **bwa_index**,
list:data:variants:vcf **known_sites**,
basic:boolean **enable_trimming**,
data:seq:nucleotide **adapters**,
basic:integer **seed_mismatches**,
basic:integer **simple_clip_threshold**,
basic:integer **min_adapter_length**,
basic:integer **palindrome_clip_threshold**,
basic:integer **leading**, *basic:integer* **trailing**,
basic:integer **minlen**, *data:bed* **intervals**,
basic:integer **contamination**, *data:seq:nucleotide* **adapters**,
basic:integer **max_insert_size**,
basic:string **pair_orientation**, *basic:integer* **read_length**,
basic:integer **min_map_quality**,
basic:integer **min_quality**, *basic:integer* **coverage_cap**,
basic:integer **accumulation_cap**,
basic:integer **sample_size**,
basic:decimal **minimum_fraction**,
basic:boolean **include_duplicates**,
basic:decimal **deviations**)[Source: v2.3.0]

Whole genome sequencing pipeline (GATK GVCF).

The pipeline follows GATK best practices recommendations and prepares single-sample paired-end sequencing data for a joint-genotyping step.

The pipeline steps include read trimming (Trimmomatic), read alignment (BWA-MEM2), marking of duplicates (Picard MarkDuplicates), recalibration of base quality scores (ApplyBQSR) and calling of variants (GATK HaplotypeCaller in GVCF mode). The QC reports (FASTQC report, Picard AlignmentSummaryMetrics, CollectWgsMetrics and InsertSizeMetrics) are summarized using MultiQC.

Input arguments reads

label
Input sample (FASTQ)

type
`data:reads:fastq:paired`

description
Input data in FASTQ format. This input type allows for optional read trimming procedure and is mutually exclusive with the BAM input file type.

required
False

disabled
aligned_reads

hidden
False

aligned_reads

label
Input sample (BAM)

type
`data:alignment:bam`

description
Input data in BAM format. This input file type is mutually exclusive with the FASTQ input file type and does not allow for read trimming procedure.

required
False

disabled
reads

hidden
False

ref_seq

label
Reference sequence

type
`data:seq:nucleotide`

required
True

disabled
False

hidden
False

bwa_index

label
BWA genome index

type
`data:index:bwamem2`

required
True

disabled
False

hidden
False

known_sites

label
Known sites of variation (VCF)

type
`list:data:variants:vcf`

required
True

disabled
False

hidden
False

trimming_options.enable_trimming

label
Trim and quality filter input data

type
`basic:boolean`

description
Enable or disable adapter trimming and QC filtering procedure.

required
True

disabled
False

hidden
False

default
`False`

trimming_options.adapters

label
Adapter sequences

type
`data:seq:nucleotide`

description
Adapter sequences in FASTA format that will be removed from the reads.

required
False

disabled
!trimming_options.enable_trimming

hidden
False

trimming_options.seed_mismatches

label
Seed mismatches

type
`basic:integer`

description
Specifies the maximum mismatch count which will still allow a full match to be performed. This field is required to perform adapter trimming.

required
False

disabled
!trimming_options.adapters

hidden
False

trimming_options.simple_clip_threshold

label
Simple clip threshold

type
`basic:integer`

description
Specifies how accurate the match between any adapter sequence must be against a read. This field is required to perform adapter trimming.

required
False

disabled
!trimming_options.adapters

hidden
False

trimming_options.min_adapter_length

label
Minimum adapter length

type
`basic:integer`

description
In addition to the alignment score, palindrome mode can verify that a minimum length of adapter has been detected. If unspecified, this defaults to 8 bases, for historical reasons. However, since palindrome mode has a very low false positive rate, this can be safely reduced, even down to 1, to allow shorter adapter fragments to be removed.

required
True

disabled
!trimming_options.seed_mismatches && !trimming_options.simple_clip_threshold && !trimming_options.palindrome_clip_threshold

hidden
False

default
8

trimming_options.palindrome_clip_threshold

label
Palindrome clip threshold

type
`basic:integer`

description
Specifies how accurate the match between the two adapter ligated reads must be for PE palindrome read alignment. This field is required to perform adapter trimming.

required
False

disabled
!trimming_options.adapters

hidden
False

trimming_options.leading

label
Leading quality

type
`basic:integer`

description
Remove low quality bases from the beginning, if below a threshold quality.

required
False

disabled
!trimming_options.enable_trimming

hidden
False

trimming_options.trailing

label
Trailing quality

type
`basic:integer`

description
Remove low quality bases from the end, if below a threshold quality.

required
False

disabled
!trimming_options.enable_trimming

hidden
False

trimming_options.minlen

label
Minimum length

type
`basic:integer`

description
Drop the read if it is below a specified length.

required
False

disabled
!trimming_options.enable_trimming

hidden
False

gatk_options.intervals

label
Intervals BED file

type
`data:bed`

description
Use intervals BED file to limit the analysis to the specified parts of the genome.

required
False

disabled
False

hidden
False

gatk_options.contamination

label
Contamination fraction

type
`basic:integer`

description
Fraction of contamination in sequencing data (for all samples) to aggressively remove.

required
True

disabled
False

hidden
False

default
`0`

alignment_summary.adapters

label
Adapter sequences

type
`data:seq:nucleotide`

required
False

disabled
False

hidden
False

alignment_summary.max_insert_size

label
Maximum insert size

type
`basic:integer`

required
True

disabled
False

hidden
False

default
`100000`

alignment_summary.pair_orientation

label
Pair orientation

type
`basic:string`

required
True

disabled
False

hidden
False

default
`null`

choices

- Unspecified: `null`
- FR: `FR`
- RF: `RF`
- TANDEM: `TANDEM`

wgs_metrics.read_length

label
Average read length

type
`basic:integer`

required
True

disabled
False

hidden
False

default
`150`

wgs_metrics.min_map_quality

label
Minimum mapping quality for a read to contribute coverage

type
`basic:integer`

required
True

disabled
False

hidden
False

default
`20`

wgs_metrics.min_quality

label
Minimum base quality for a base to contribute coverage

type
`basic:integer`

description
N bases will be treated as having a base quality of negative infinity and will therefore be excluded from coverage regardless of the value of this parameter.

required
True

disabled
False

hidden
False

default
`20`

wgs_metrics.coverage_cap

label
Maximum coverage cap

type
`basic:integer`

description
Treat positions with coverage exceeding this value as if they had coverage at this set value.

required
True

disabled
False

hidden
False

default
`250`

wgs_metrics.accumulation_cap

label
Ignore positions with coverage above this value

type
`basic:integer`

description
At positions with coverage exceeding this value, completely ignore reads that accumulate beyond this value.

required
True

disabled
False

hidden
False

default
`100000`

wgs_metrics.sample_size

label
Sample size used for Theoretical Het Sensitivity sampling

type
`basic:integer`

required
True

disabled
False

hidden
False

default
`10000`

insert_size.minimum_fraction

label
Minimum fraction of reads in a category to be considered

type
`basic:decimal`

description
When generating the histogram, discard any data categories (out of FR, TANDEM, RF) that have fewer than this fraction of overall reads (Range: 0 and 0.5).

required
True

disabled
False

hidden
False

default
`0.05`

insert_size.include_duplicates

label
Include reads marked as duplicates in the insert size histogram

type
`basic:boolean`

required
True

disabled
False

hidden
False

default
`False`

insert_size.deviations

label
Deviations limit

type
`basic:decimal`

description
Generate mean, standard deviation and plots by trimming the data down to MEDIAN + DEVIATIONS * MEDIAN_ABSOLUTE_DEVIATION. This is done because insert size data typically includes enough anomalous values from chimeras and other artifacts to make the mean and standard deviation grossly misleading regarding the real distribution.

required
True

disabled
False

hidden
False

default
`10.0`

Output results

WGS preprocess data with bwa-mem2

`data:alignment:bam:wgsbwa2:wgs-preprocess-bwa2` (*data:reads:fastq:paired* **reads**, *data:alignment:bam* **aligned_reads**, *data:seq:nucleotide* **ref_seq**, *data:index:bwamem2* **bwa_index**, *list:data:variants:vcf* **known_sites**, *basic:integer* **pixel_distance**, *basic:integer* **n_jobs**) [Source: v1.4.0]

Prepare analysis ready BAM file.

This process follows GATK best practices procedure to prepare analysis-ready BAM file. The steps included are read alignment using BWA MEM2, marking of duplicates (Picard MarkDuplicates), BAM sorting, read-group assignment and base quality score recalibration (BQSR).

Input arguments reads

label
Input sample (FASTQ)

type
`data:reads:fastq:paired`

required
False

disabled
False

hidden
False

aligned_reads

label
Input sample (BAM)

type
`data:alignment:bam`

required
False

disabled
False

hidden
False

ref_seq

label
Reference sequence

type
`data:seq:nucleotide`

required
True

disabled
False

hidden
False

bwa_index

label
BWA-MEM2 genome index

type
`data:index:bwamem2`

required
True

disabled
False

hidden
False

known_sites

label
Known sites of variation (VCF)

type
`list:data:variants:vcf`

required
True

disabled
False

hidden
False

advanced_options.pixel_distance

label
–OPTICAL_DUPLICATE_PIXEL_DISTANCE

type
`basic:integer`

description
Set the optical pixel distance, e.g. distance between clusters. Modify this parameter to ensure compatibility with older Illumina platforms.

required
True

disabled
False

hidden
False

default
`2500`

advanced_options.n_jobs

label
Number of concurent jobs

type
`basic:integer`

description
Use a fixed number of jobs for quality score recalibration of determining it based on the number of available cores.

required
False

disabled
False

hidden
False

Output results bam

label
Analysis ready BAM file

type
`basic:file`

required
True

disabled
False

hidden
False

bai

label
BAM file index

type
`basic:file`

required
True

disabled
False

hidden
False

stats

label
Alignment statistics

type
`basic:file`

required
True

disabled
False

hidden
False

species

label
Species

type
`basic:string`

required
True

disabled
False

hidden
False

build

label
Build

type
`basic:string`

required
True

disabled
False

hidden
False

metrics_file

label
Metrics from MarkDuplicate process

type
`basic:file`

required
True

disabled
False

hidden
False

Whole exome sequencing (WES) analysis

`data:workflow:wesworkflow-wes` (*data:reads:fastq:paired* **reads**, *data:index:bwa* **bwa_index**, *data:seq:nucleotide* **ref_seq**, *list:data:variants:vcf* **known_sites**, *data:bed* **intervals**, *data:variants:vcf* **hc_dbsnp**, *basic:string* **validation_stringency**, *data:seq:nucleotide* **adapters**, *basic:integer* **seed_mismatches**, *basic:integer* **simple_clip_threshold**, *basic:integer* **min_adapter_length**, *basic:integer* **palindrome_clip_threshold**, *basic:integer* **leading**, *basic:integer* **trailing**, *basic:integer* **minlen**, *basic:integer* **seed_l**, *basic:integer* **band_w**, *basic:boolean* **m**, *basic:decimal* **re_seeding**, *basic:integer* **match**, *basic:integer* **mismatch**, *basic:integer* **gap_o**, *basic:integer* **gap_e**, *basic:integer* **clipping**, *basic:integer* **unpaired_p**, *basic:integer* **report_tr**, *data:bedpe* **bedpe**, *basic:boolean* **skip**, *basic:boolean* **md_skip**, *basic:boolean* **md_remove_duplicates**, *basic:string* **md_assume_sort_order**, *basic:string* **read_group**, *basic:integer* **stand_call_conf**, *basic:integer* **mbq**) [Source: v3.1.0]

Whole exome sequencing pipeline analyzes Illumina panel data. It consists of trimming, aligning, soft clipping, (optional) marking of duplicates, recalibration of base quality scores and finally, calling of variants.

The tools used are Trimmomatic which performs trimming. Aligning is performed using BWA (mem). Soft clipping of Illumina primer sequences is done using bamclipper tool. Marking of duplicates (MarkDuplicates), recalibration of base quality scores (ApplyBQSR) and calling of variants (HaplotypeCaller) is done using GATK4 bundle of bioinformatics tools.

To successfully run this pipeline, you will need a genome (FASTA), paired-end (FASTQ) files, BEDPE file for bamclipper, known sites of variation (dbSNP) (VCF), dbSNP database of variations (can be the same as known sites of variation), intervals on which target capture was done (BED) and illumina adapter sequences (FASTA). Make sure that specified resources match the genome used in the alignment step.

Result is a file of called variants (VCF).

Input arguments reads

label
Raw untrimmed reads

type
`data:reads:fastq:paired`

description
Raw paired-end reads.

bwa_index

label
BWA genome index

type
`data:index:bwa`

description
Genome index used for the BWA alignment step.

ref_seq

label
Genome FASTA

type
`data:seq:nucleotide`

description
The selection of Genome FASTA should match the BWA index species and genome build type.

known_sites

label
Known sites of variation used in BQSR

type
`list:data:variants:vcf`

description
Known sites of variation as a VCF file.

intervals

label
Intervals

type
`data:bed`

description
Use intervals to narrow the analysis to defined regions. This usually help cutting down on process time.

hc_dbsnp

label
dbSNP for GATK4's HaplotypeCaller

type
`data:variants:vcf`

description
dbSNP database of variants for variant calling.

validation_stringency

label
Validation stringency for all SAM files read by this program. Setting stringency to SILENT can improve performance when processing a BAM file in which variable-length data (read, qualities, tags)

do not otherwise need to be decoded. Default is STRICT. This setting is used in BaseRecalibrator and ApplyBQSR processes.

type
`basic:string`

default
`STRICT`

choices

- STRICT: `STRICT`
- SILENT: `SILENT`
- LENIENT: `LENIENT`

advanced.trimming.adapters

label
Adapter sequences

type
`data:seq:nucleotide`

description
Adapter sequence in FASTA format that will be removed from the read. This field as well as 'Seed mismatches', 'Simple clip threshold' and 'Palindrome clip threshold' parameters are needed to perform Illuminacliping. 'Minimum adapter length' and 'Keep both reads' are optional parameters.

required
False

advanced.trimming.seed_mismatches

label
Seed mismatches

type
`basic:integer`

description
Specifies the maximum mismatch count which will still allow a full match to be performed. This field as well as 'Adapter sequence', 'Simple clip threshold' and 'Palindrome clip threshold' parameters are needed to perform Illuminacliping.

required
False

disabled
!advanced.trimming.adapters

advanced.trimming.simple_clip_threshold

label
Simple clip threshold

type
`basic:integer`

description
Specifies how accurate the match between any adapter etc. sequence must be against a read. This field as well as 'Adapter sequences' and 'Seed mismatches' parameter are needed to perform Illuminacliping.

required
False

disabled
!advanced.trimming.adapters

advanced.trimming.min_adapter_length

label
Minimum adapter length

type
`basic:integer`

description
In addition to the alignment score, palindrome mode can verify that a minimum length of adapter has been detected. If unspecified, this defaults to 8 bases, for historical reasons. However, since palindrome mode has a very low false positive rate, this can be safely reduced, even down to 1, to allow shorter adapter fragments to be removed. This field is optional for preforming Illuminaclip. 'Adapter sequences', 'Seed mismatches', 'Simple clip threshold' and 'Palindrome clip threshold' are also needed in order to use this parameter.

disabled
!advanced.trimming.seed_mismatches && !advanced.trimming.simple_clip_threshold && !advanced.trimming.palindrome_clip_threshold

default
`8`

advanced.trimming.palindrome_clip_threshold

label
Palindrome clip threshold

type
`basic:integer`

description
Specifies how accurate the match between the two 'adapter ligated' reads must be for PE palindrome read alignment. This field as well as 'Adapter sequence', 'Simple clip threshold' and 'Seed mismatches' parameters are needed to perform Illuminaclipping.

required
False

disabled
!advanced.trimming.adapters

advanced.trimming.leading

label
Leading quality

type
`basic:integer`

description
Remove low quality bases from the beginning, if below a threshold quality.

required
False

advanced.trimming.trailing

label
Trailing quality

type
`basic:integer`

description
Remove low quality bases from the end, if below a threshold quality.

required
False

advanced.trimming.minlen

label
Minimum length

type
`basic:integer`

description
Drop the read if it is below a specified length.

required
False

advanced.align.seed_l

label
Minimum seed length

type
`basic:integer`

description
Minimum seed length. Matches shorter than minimum seed length will be missed. The alignment speed is usually insensitive to this value unless it significantly deviates 20.

default
`19`

advanced.align.band_w

label
Band width

type
`basic:integer`

description
Gaps longer than this will not be found.

default
`100`

advanced.align.m

label
Mark shorter split hits as secondary

type
`basic:boolean`

description
Mark shorter split hits as secondary (for Picard compatibility)

default
`False`

advanced.align.re_seeding

label
Re-seeding factor

type
`basic:decimal`

description
Trigger re-seeding for a MEM longer than minSeedLen*FACTOR. This is a key heuristic parameter for tuning the performance. Larger value yields fewer seeds, which leads to faster alignment speed but lower accuracy.

default
`1.5`

advanced.align.scoring.match

label
Score of a match

type
`basic:integer`

default
`1`

advanced.align.scoring.mismatch

label
Mismatch penalty

type
`basic:integer`

default
`4`

advanced.align.scoring.gap_o

label
Gap open penalty

type
`basic:integer`

default
`6`

advanced.align.scoring.gap_e

label
Gap extension penalty

type
`basic:integer`

default
`1`

advanced.align.scoring.clipping

label
Clipping penalty

type
`basic:integer`

description
Clipping is applied if final alignment score is smaller than (best score reaching the end of query) - (Clipping penalty)

default
`5`

advanced.align.scoring.unpaired_p

label
Penalty for an unpaired read pair

type
`basic:integer`

description
Affinity to force pair. Score: scoreRead1+scoreRead2-Penalty

default
`9`

advanced.align.report_tr

label
Report threshold score

type
`basic:integer`

description
Don't output alignment with score lower than defined number. This option only affects output.

default
`30`

advanced.bamclipper.bedpe

label
BEDPE file used for clipping using Bamclipper

type
`data:bedpe`

description
BEDPE file used for clipping using Bamclipper tool.

required
False

advanced.bamclipper.skip

label
Skip Bamclipper step

type
`basic:boolean`

description
Use this option to skip Bamclipper step.

default
`False`

advanced.markduplicates.md_skip

label
Skip GATK's MarkDuplicates step

type
`basic:boolean`

default
`False`

advanced.markduplicates.md_remove_duplicates

label
Remove found duplicates

type
`basic:boolean`

default
`False`

advanced.markduplicates.md_assume_sort_order

label
Assume sort oder

type
`basic:string`

default

choices

- as in BAM header (default):
- unsorted: `unsorted`
- queryname: `queryname`
- coordinate: `coordinate`
- duplicate: `duplicate`
- unknown: `unknown`

advanced.bqsr.read_group

label
Read group (@RG)

type
`basic:string`

description
If BAM file has not been prepared using a @RG tag, you can add it here. This argument enables the user to replace all read groups in the INPUT file with a single new read group and assign all reads to this read group in the OUTPUT BAM file. Addition or replacement is performed using Picard's AddOrReplaceReadGroups tool. Input should take the form of -name=value delimited by a \t, e.g. "-ID=1\t-PL=Illumina\t-SM=sample_1". See AddOrReplaceReadGroups documentation for more information on tag names. Note that PL, LB, PU and SM are required fields. See caveats of rewriting read groups in the documentation linked above.

required
False

advanced.hc.stand_call_conf

label
Min call confidence threshold

type
`basic:integer`

description
The minimum phred-scaled confidence threshold at which variants should be called.

default
`20`

advanced.hc.mbq

label
Min Base Quality

type
`basic:integer`

description
Minimum base quality required to consider a base for calling.

default
`20`

Output results

Xengsort classify

`data:xengsort:classification:xengsort-classify` (*data:reads:fastq* **reads**, *data:xengsort:index* **index**, *basic:string* **upload_reads**, *basic:boolean* **merge_both**, *basic:decimal* **chunksize**) [Source: v1.0.0]

Classify xenograft reads with Xengsort.

Xengsort is an alignment free method for sorting reads from xenograft experiments. It classifies sequencing reads into five categories based on their origin: host, graft, both, neither, and ambiguous. Categories "host" and "graft" are for reads that can be clearly assigned to one of the species. Category "both" is for reads that match equally well to both references. Category "neither" is for reads that contain many k-mers that cannot be found in the key-value store; these could point to technical problems (primer dimers) or contamination of the sample with other species. Finally, category "ambiguous" is for reads that provide conflicting information. Such reads should not usually be seen; they could result from PCR hybrids between host and graft during library preparation.

Description of the method and evaluation on several datasets is provided in the [article](https://doi.org/10.1186/s13015-021-00181-w).

Input arguments reads

label
Reads

type
`data:reads:fastq`

required
True

disabled
False

hidden
False

index

label
Xengsort genome index

type
`data:xengsort:index`

required
True

disabled
False

hidden
False

upload_reads

label
Select reads to upload

type
`basic:string`

description
All read categories are returned in this process but only the ones selected are uploaded as separate FASTQ files. This should be used for categories of reads that will be used in further analyses.

required
True

disabled
False

hidden
False

default
`none`

choices

- none: `none`
- all: `all`
- graft: `graft`
- graft, both: `graft, both`
- graft, host: `graft, host`
- graft, host, both: `graft, host, both`

merge_both

label
Upload merged graft and both reads

type
`basic:boolean`

description
Merge graft reads with the reads that can originate from both genomes and upload it as graft reads. In any workflow, the latter reads, classified as both may pose problems, because one may not be able to decide on the species of origin due to ultra-conserved regions between species.

required
True

disabled
False

hidden
upload_reads == 'none'

default
`False`

advanced.chunksize

label
Chunk size in MB [–chunksize]

type
`basic:decimal`

description
Controll the memory usage by setting chunk size per thread.

required
True

disabled
False

hidden
False

default
`16.0`

Output results stats

label
Xengsort classification statistics

type
`basic:file`

required
True

disabled
False

hidden
False

host1

label
Host reads (mate 1)

type
`basic:file`

required
True

disabled
False

hidden
False

host2

label
Host reads (mate 2)

type
`basic:file`

required
False

disabled
False

hidden
False

graft1

label
Graft reads (mate 1)

type
`basic:file`

required
True

disabled
False

hidden
False

graft2

label
Graft reads (mate 2)

type
`basic:file`

required
False

disabled
False

hidden
False

both1

label
Both reads (mate 1)

type
`basic:file`

required
True

disabled
False

hidden
False

both2

label
Both reads (mate 2)

type
`basic:file`

required
False

disabled
False

hidden
False

neither1

label
Neither reads (mate 1)

type
`basic:file`

required
True

disabled
False

hidden
False

neither2

label
Neither reads (mate 2)

type
`basic:file`

required
False

disabled
False

hidden
False

ambiguous1

label
Ambiguous reads (mate 1)

type
`basic:file`

required
True

disabled
False

hidden
False

ambiguous2

label
Ambiguous reads (mate 2)

type
`basic:file`

required
False

disabled
False

hidden
False

graft_species

label
Graft species

type
`basic:string`

required
True

disabled
False

hidden
False

graft_build

label
Graft build

type
`basic:string`

required
True

disabled
False

hidden
False

host_species

label
Host species

type
`basic:string`

required
True

disabled
False

hidden
False

host_build

label
Host build

type
`basic:string`

required
True

disabled
False

hidden
False

Xengsort index

`data:xengsort:index:xengsort-index` (*list:data:seq:nucleotide* **graft_refs**, *list:data:seq:nucleotide* **host_refs**, *basic:integer* **n_kmer**, *basic:integer* **kmer_size**, *basic:boolean* **aligned_cache**, *basic:boolean* **fixed_hashing**, *basic:integer* **page_size**, *basic:decimal* **fill**) [Source: v1.0.1]

Build an index for sorting xenograft reads with Xengsort.

Xengsort is an alignment free method for sorting reads from xenograft experiments. Description of the method and evaluation on several datasets is provided in the [article](https://doi.org/10.1186/s13015-021-00181-w).

Input arguments graft_refs

label
Graft reference sequences (nucleotide FASTA)

type
`list:data:seq:nucleotide`

required
True

disabled
False

hidden
False

host_refs

label
Host reference sequences (nucleotide FASTA)

type
`list:data:seq:nucleotide`

required
True

disabled
False

hidden
False

n_kmer

label
Number of distinct k-mers [–nobjects]

type
`basic:integer`

description
The number of k-mers that will be stored in the hash table. This depends on the used reference genomes and must be estimated beforehand. If the number of distinct k-mers is known beforehand it should be specified. For all 25-mers in the human and mouse genome and transcriptome, this number is roughly 4,500,000,000. If this is not set, the number is estimated with ntCard tool and increased by two percent to account for errors.

required
False

disabled
False

hidden
False

advanced.kmer_size

label
k-mer size [–kmersize]

type
`basic:integer`

required
True

disabled
False

hidden
False

default
`25`

advanced.aligned_cache

label
Use power-of-two aligned pages [–aligned]

type
`basic:boolean`

description
Indicates whether each bucket should consume a number of bits that is a power of 2. Using –aligned ensures that each bucket stays within the same cache line, but may waste space (padding bits), yielding faster speed but larger space requirements. By default no bits are used for padding and buckets may cross cache line boundaries [–unaligned]. This is slightly slower, but may save a little or a lot of space.

required
True

disabled
False

hidden
False

default
`False`

advanced.fixed_hashing

label
Use fixed hash function [–hashfunctions]

type
`basic:boolean`

description
Hash function used to store the key-value pairs is defined by –hashfunction parameter. With this option selected a fixed hash function (linear945:linear9123641:linear349341847) is used. When this is not selected a different random functions are chosen each time. It is recommended to have them chosen randomly unless you need strictly reproducible behavior.

required
True

disabled
False

hidden
False

default
`True`

advanced.page_size

label
Number of elements stored in one bucket (or page) [–pagesize]

type
`basic:integer`

required
True

disabled
False

hidden
False

default
`4`

advanced.fill

label
Fill rate of the hash table [–fill]

type
`basic:decimal`

description
This determines the desired fill rate or load factor of the hash table. It should be set between 0.0 and 1.0. It is beneficial to leave part of the hash table empty for faster lookups. Together with the number of distinct k-mers [–nobjects], the number of slots in the table is calculated as ceil(nobjects/fill).

required
True

disabled
False

hidden
False

default
`0.88`

Output results index

label
Xengsort index

type
`basic:file`

required
True

disabled
False

hidden
False

stats

label
Xengsort statistics

type
`basic:file`

required
True

disabled
False

hidden
False

graft_species

label
Graft species

type
`basic:string`

required
True

disabled
False

hidden
False

graft_build

label
Graft build

type
`basic:string`

required
True

disabled
False

hidden
False

host_species

label
Host species

type
`basic:string`

required
True

disabled
False

hidden
False

host_build

label
Host build

type
`basic:string`

required
True

disabled
False

hidden
False

alignmentSieve

`data:alignment:bam:sieve:alignmentsieve` (*data:alignment:bam* **alignment**,
basic:integer **min_fragment_length**,
basic:integer **max_fragment_length**) [Source: v1.5.3]

Filter alignments of BAM files according to specified parameters.

Program is bundled with deeptools. See [documentation](https://deeptools.readthedocs.io/en/develop/content/tools/alignmentSieve.html
for more details.

Input arguments alignment

label
Alignment BAM file

type
`data:alignment:bam`

required
True

disabled
False

hidden
False

min_fragment_length

label
–minFragmentLength

type
`basic:integer`

description
The minimum fragment length needed for read/pair inclusion. This option is primarily useful in ATACseq experiments, for filtering mono- or di-nucleosome fragments. (Default: 0)

required
True

disabled
False

hidden
False

default
`0`

max_fragment_length

label
–maxFragmentLength

type
`basic:integer`

description
The maximum fragment length needed for read/pair inclusion. A value of 0 indicates no limit. (Default: 0)

required
True

disabled
False

hidden
False

default
`0`

Output results bam

label
Sieved BAM file

type
`basic:file`

required
True

disabled
False

hidden
False

bai

label
Index of sieved BAM file

type
`basic:file`

required
True

disabled
False

hidden
False

stats

label
Alignment statistics

type
`basic:file`

required
True

disabled
False

hidden
False

species

label
Species

type
`basic:string`

required
True

disabled
False

hidden
False

build

label
Build

type
`basic:string`

required
True

disabled
False

hidden
False

edgeR

`data:differentialexpression:edger:differentialexpression-edger` (*list:data:expression* **case**, *list:data:expression* **control**, *basic:integer* **count_filter**, *basic:boolean* **create_sets**, *basic:decimal* **logfc**, *basic:decimal* **fdr**) [Source: v1.7.0]

Run EdgeR analysis.

Empirical Analysis of Digital Gene Expression Data in R (edgeR). Differential expression analysis of RNA-seq expression profiles with biological replication. Implements a range of statistical methodology based on the negative binomial distributions, including empirical Bayes estimation, exact tests, generalized linear

models and quasi-likelihood tests. As well as RNA-seq, it be applied to differential signal analysis of other types of genomic data that produce counts, including ChIP-seq, Bisulfite-seq, SAGE and CAGE. See [here](https://www.bioconductor.org/packages/devel/bioc/vignettes/edgeR/inst/doc/edgeRUsersGuide.pdf) for more information.

Input arguments case

label
Case

type
`list:data:expression`

description
Case samples (replicates)

required
True

disabled
False

hidden
False

control

label
Control

type
`list:data:expression`

description
Control samples (replicates)

required
True

disabled
False

hidden
False

count_filter

label
Raw counts filtering threshold

type
`basic:integer`

description
Filter genes in the expression matrix input. Remove genes where the number of counts in all samples is below the threshold.

required
True

disabled
False

hidden
False

default
`10`

create_sets

label
Create gene sets

type
`basic:boolean`

description
After calculating differential gene expressions create gene sets for up-regulated genes, down-regulated genes and all genes.

required
True

disabled
False

hidden
False

default
`False`

logfc

label
Log2 fold change threshold for gene sets

type
`basic:decimal`

description
Genes above Log2FC are considered as up-regulated and genes below -Log2FC as down-regulated.

required
True

disabled
False

hidden
!create_sets

default
`1.0`

fdr

label
FDR threshold for gene sets

type
`basic:decimal`

required
True

disabled
False

hidden
!create_sets

default
`0.05`

Output results raw

label
Differential expression

type
`basic:file`

required
True

disabled
False

hidden
False

de_json

label
Results table (JSON)

type
`basic:json`

required
True

disabled
False

hidden
False

de_file

label
Results table (file)

type
`basic:file`

required
True

disabled
False

hidden
False

source

label
Gene ID database

type
basic:string

required
True

disabled
False

hidden
False

species

label
Species

type
basic:string

required
True

disabled
False

hidden
False

build

label
Build

type
basic:string

required
True

disabled
False

hidden
False

feature_type

label
Feature type

type
basic:string

required
True

disabled
False

hidden
False

methcounts

data:wgbs:methcountsmethcounts (*data:seq:nucleotide* **genome**, *data:alignment:bam:walt* **alignment**, *basic:boolean* **cpgs**, *basic:boolean* **symmetric_cpgs**) [Source: v3.3.0]

The methcounts program takes the mapped reads and produces the methylation level at each genomic cytosine, with the option to produce only levels for CpG-context cytosines.

Input arguments genome

label
Reference genome

type
`data:seq:nucleotide`

alignment

label
Mapped reads

type
`data:alignment:bam:walt`

description
WGBS alignment file in Mapped Read (.mr) format.

cpgs

label
Only CpG context sites

type
`basic:boolean`

description
Output file will contain methylation data for CpG context sites only. Choosing this option will result in CpG content report only.

disabled
symmetric_cpgs

default
`False`

symmetric_cpgs

label
Merge CpG pairs

type
`basic:boolean`

description
Merging CpG pairs results in symmetric methylation levels. Methylation is usually symmetric (cytosines at CpG sites were methylated on both DNA strands). Choosing this option will only keep the CpG sites data.

disabled
cpgs

default
`True`

Output results meth

label
Methylation levels

type
`basic:file`

stats

label
Statistics

type
`basic:file`

bigwig

label
Methylation levels BigWig file

type
`basic:file`

species

label
Species

type
`basic:string`

build

label
Build

type
`basic:string`

miRNA pipeline

`data:workflow:mirnaworkflow-mirna` (*data:reads:fastq:single* **reads**, *data:seq:nucleotide* **up_primers_file**, *data:seq:nucleotide* **down_primers_file**, *list:basic:string* **up_primers_seq**, *list:basic:string* **down_primers_seq**, *basic:integer* **min_overlap**, *basic:boolean* **show_advanced**, *basic:integer* **leading**, *basic:integer* **trailing**, *basic:integer* **minlen**, *basic:integer* **maxlen**, *basic:integer* **max_n**, *basic:boolean* **match_read_wildcards**, *basic:boolean* **no_indels**, *basic:decimal* **error_rate**, *data:index:bowtie2* **genome**, *basic:boolean* **show_alignment_options**, *basic:string* **mode**, *basic:string* **speed**, *basic:integer* **N**, *basic:integer* **L**, *basic:string* **rep_mode**, *basic:integer* **k_reports**, *data:annotation* **annotation**, *basic:string* **id_attribute**, *basic:string* **feature_class**, *basic:string* **normalization_type**, *basic:boolean* **allow_multi_overlap**, *basic:boolean* **count_multi_mapping_reads**, *basic:string* **assay_type**)[Source: v3.1.0]

Input arguments preprocessing.reads

label
Input miRNA reads.

type
`data:reads:fastq:single`

preprocessing.adapters.up_primers_file

label
5 prime adapter file

type
`data:seq:nucleotide`

required
False

preprocessing.adapters.down_primers_file

label
3 prime adapter file

type
`data:seq:nucleotide`

required
False

preprocessing.adapters.up_primers_seq

label
5 prime adapter sequence

type
`list:basic:string`

required
False

preprocessing.adapters.down_primers_seq

label
3 prime adapter sequence

type
`list:basic:string`

required
False

preprocessing.adapters.min_overlap

label
Minimal overlap

type
`basic:integer`

description
Minimum overlap for an adapter match. Default 5.

default
5

preprocessing.show_advanced

label
Show advanced preprocessing parameters

type
basic:boolean

default
False

preprocessing.trimming.leading

label
Quality on 5 prime

type
basic:integer

description
Remove low quality bases from 5 prime. Specifies the minimum quality required to keep a base. Default: 20.

hidden
!preprocessing.show_advanced

default
28

preprocessing.trimming.trailing

label
Quality on 3 prime

type
basic:integer

description
Remove low quality bases from the 3 prime. Specifies the minimum quality required to keep a base. Default: 20.

hidden
!preprocessing.show_advanced

default
28

preprocessing.filtering.minlen

label
Min length

type
basic:integer

description
Drop the read if it is below a specified length. Default: 15.

hidden
!preprocessing.show_advanced

default
15

preprocessing.filtering.maxlen

label
Max length

type
basic:integer

description
Drop the read if it is above a specified length. Default: 35.

hidden
!preprocessing.show_advanced

default
35

preprocessing.filtering.max_n

label
Max numebr of N-s

type
basic:integer

description
Discard reads having more 'N' bases than specified. Default: 1.

hidden
!preprocessing.show_advanced

default
1

preprocessing.filtering.match_read_wildcards

label
Match read wildcards

type
basic:boolean

description
Interpret IUPAC wildcards in reads.

hidden
!preprocessing.show_advanced

default
True

preprocessing.filtering.no_indels

label
No indels

type
basic:boolean

description
Disable (disallow) insertions and deletions in adapters.

hidden
!preprocessing.show_advanced

default
`True`

preprocessing.filtering.error_rate

label
Error rate

type
`basic:decimal`

description
Maximum allowed error rate (no. of errors divided by the length of the matching region). Default: 0.2.

hidden
!preprocessing.show_advanced

default
`0.2`

alignment.genome

label
Genome reference

type
`data:index:bowtie2`

description
Choose the genome reference against which to align reads.

alignment.show_alignment_options

label
Show alignment options

type
`basic:boolean`

default
`False`

alignment.alignment_options.mode

label
Alignment mode

type
`basic:string`

description
End to end: Bowtie 2 requires that the entire read align from one end to the other, without any trimming (or "soft clipping") of characters from either end. Local: Bowtie 2 does not require that the entire read align from one end to the other. Rather, some characters may be omitted ("soft clipped") from the ends in order to achieve the greatest possible alignment score. Default: –local (with sensitivity set to '–very-sensitive' for both options).

hidden
!alignment.show_alignment_options

default
`--local`

choices

- local: `--local`
- end to end mode: `--end-to-end`

alignment.alignment_options.speed

label
Sensitivity

type
`basic:string`

description
A quick parameter presetting for aligning accurately. This option is a shortcut for parameters as follows: For both alignment modes: –very-sensitive Same as: -D 20 -R 3 -N 0 -L 20 -i S,1,0.50

hidden
!alignment.show_alignment_options

default
`--very-sensitive`

alignment.alignment_options.N

label
Number of mismatches allowed in seed alignment (N)

type
`basic:integer`

description
Sets the number of mismatches allowed in seed. Can be set to 0 or 1. Default: 0

hidden
!alignment.show_alignment_options

default
`0`

alignment.alignment_options.L

label
Length of seed substrings (L)

type
`basic:integer`

description
Sets the length of the seed substrings to align during multiseed alignment. The –very-sensitive preset sets -L to 20 in –end-to-end and in –local mode. For miRNA, a shorter seed length is recommended. Default: -L 8

hidden
!alignment.show_alignment_options

default
`8`

alignment.alignment_options.rep_mode

label
Report mode

type
`basic:string`

description
Tool default mode: search for multiple alignments, report the best one; -k mode: search for one or more alignments, report each; -a mode: search for and report all alignments. Default: -k

hidden
!alignment.show_alignment_options

default
`k`

choices

- Tool default mode: `def`
- -k mode: `k`
- -a mode (very slow): `a`

alignment.alignment_options.k_reports

label
Number of reports (for -k mode only)

type
`basic:integer`

description
Searches for at most X distinct, valid alignments for each read. The search terminates when it can't find more distinct valid alignments, or when it finds X, whichever happens first. Default: 5

hidden
!alignment.show_alignment_options

default
`5`

quant_options.annotation

label
Annotation (GTF/GFF3)

type
`data:annotation`

quant_options.id_attribute

label
ID attribute

type
`basic:string`

description
GTF/GFF3 attribute to be used as feature ID. Several GTF/GFF3 lines with the same feature ID will be considered as parts of the same feature. The feature ID is used to identify the counts in the output table. In GTF files this is usually 'gene_id', in GFF3 files this is often 'ID', and 'transcript_id' is frequently a valid choice for both annotation formats. miRNA name refers to the miRBase GFF3 'Name' filed and is the default option.

default
`Name`

choices

- miRNA name: `Name`
- gene_id: `gene_id`
- transcript_id: `transcript_id`
- ID: `ID`
- geneid: `geneid`

quant_options.feature_class

label
Feature class

type
`basic:string`

description
Feature class (3rd column in GFF file) to be used, all features of other types are ignored. Default: miRNA.

default
`miRNA`

quant_options.normalization_type

label
Normalization type

type
`basic:string`

description
The default expression normalization type.

default
`CPM`

quant_options.allow_multi_overlap

label
Count multi-overlapping reads

type
`basic:boolean`

description
Assign reads to all their overlapping features or meta-features.

default
`True`

quant_options.count_multi_mapping_reads

label
Count multi-mapping reads

type
`basic:boolean`

description
For a multi-mapping read, all its reported alignments will be counted. The 'NH' tag in BAM input is used to detect multi-mapping reads.

default
`True`

assay_type

label
Assay type

type
`basic:string`

description
Indicate if strand-specific read counting should be performed. In strand non-specific assay a read is considered overlapping with a feature regardless of whether it is mapped to the same or the opposite strand as the feature. In strand-specific forward assay, the read has to be mapped to the same strand as the feature. In strand-specific reverse assay these rules are reversed.

choices

- Strand non-specific: `non_specific`
- Strand-specific forward: `forward`
- Strand-specific reverse: `reverse`

Output results

shRNA quantification

`data:workflow:trimalquantworkflow-trim-align-quant` (*data:reads:fastq:single* **reads**, *list:basic:string* **up_primers_seq**, *list:basic:string* **down_primers_seq**, *basic:decimal* **error_rate_5end**, *basic:decimal* **error_rate_3end**, *data:index:bowtie2* **genome**, *basic:string* **mode**, *basic:integer* **N**, *basic:integer* **L**, *basic:integer* **gbar**, *basic:string* **mp**, *basic:string* **rdg**, *basic:string* **rfg**, *basic:string* **score_min**, *basic:integer* **readlengths**, *basic:integer* **alignscores**) [Source: v1.1.0]

Input arguments reads

label
Untrimmed reads.

type
`data:reads:fastq:single`

description
First stage of shRNA pipeline. Trims 5' adapters, then 3' adapters using the same error rate setting, aligns reads to a reference library and quantifies species.

trimming_options.up_primers_seq

label
5’ adapter sequence

type
`list:basic:string`

description
A string of 5’ adapter sequence.

required
True

trimming_options.down_primers_seq

label
3’ adapter sequence

type
`list:basic:string`

description
A string of 3’ adapter sequence.

required
True

trimming_options.error_rate_5end

label
Error rate for 5’

type
`basic:decimal`

description
Maximum allowed error rate (no. of errors divided by the length of the matching region) for 5’ trimming.

required
False

default
`0.1`

trimming_options.error_rate_3end

label
Error rate for 3’

type
`basic:decimal`

description
Maximum allowed error rate (no. of errors divided by the length of the matching region) for 3’ trimming.

required
False

default
`0.1`

alignment_options.genome

label
Reference library

type
`data:index:bowtie2`

description
Choose the reference library against which to align reads.

alignment_options.mode

label
Alignment mode

type
`basic:string`

description
End to end: Bowtie 2 requires that the entire read align from one end to the other, without any trimming (or "soft clipping") of characters from either end. local: Bowtie 2 does not require that the entire read align from one end to the other. Rather, some characters may be omitted ("soft clipped") from the ends in order to achieve the greatest possible alignment score.

default
`--end-to-end`

choices

- end to end mode: `--end-to-end`
- local: `--local`

alignment_options.N

label
Number of mismatches allowed in seed alignment (N)

type
`basic:integer`

description
Sets the number of mismatches to allowed in a seed alignment during multiseed alignment. Can be set to 0 or 1. Setting this higher makes alignment slower (often much slower) but increases sensitivity. Default: 0.

required
False

alignment_options.L

label
Length of seed substrings (L)

type
`basic:integer`

description
Sets the length of the seed substrings to align during multiseed alignment. Smaller values make alignment slower but more sensitive. Default: the –sensitive preset is used by default for end-to-end alignment and –sensitive-local for local alignment. See documentation for details.

required
False

alignment_options.gbar

label
Disallow gaps within positions (gbar)

type
`basic:integer`

description
Disallow gaps within <int> positions of the beginning or end of the read. Default: 4.

required
False

alignment_options.mp

label
Maximal and minimal mismatch penalty (mp)

type
`basic:string`

description
Sets the maximum (MX) and minimum (MN) mismatch penalties, both integers. A number less than or equal to MX and greater than or equal to MN is subtracted from the alignment score for each position where a read character aligns to a reference character, the characters do not match, and neither is an N. If –ignore-quals is specified, the number subtracted quals MX. Otherwise, the number subtracted is MN + floor((MX-MN)(MIN(Q, 40.0)/40.0)) where Q is the Phred quality value. Default for MX, MN: 6,2.

required
False

alignment_options.rdg

label
Set read gap open and extend penalties (rdg)

type
`basic:string`

description
Sets the read gap open (<int1>) and extend (<int2>) penalties. A read gap of length N gets a penalty of <int1> + N * <int2>. Default: 5,3.

required
False

alignment_options.rfg

label
Set reference gap open and close penalties (rfg)

type
`basic:string`

description
Sets the reference gap open (<int1>) and extend (<int2>) penalties. A reference gap of length N gets a penalty of <int1> + N * <int2>. Default: 5,3.

required
False

alignment_options.score_min

label
Minimum alignment score needed for "valid" alignment (score-min)

type
`basic:string`

description
Sets a function governing the minimum alignment score needed for an alignment to be considered "valid" (i.e. good enough to report). This is a function of read length. For instance, specifying L,0,-0.6 sets the minimum-score function to f(x) = 0 + -0.6 * x, where x is the read length. The default in –end-to-end mode is L,-0.6,-0.6 and the default in –local mode is G,20,8.

required
False

quant_options.readlengths

label
Species lengths threshold

type
`basic:integer`

description
Species with read lengths below specified threshold will be removed from final output. Default is no removal.

quant_options.alignscores

label
Align scores filter threshold

type
`basic:integer`

description
Species with align score below specified threshold will be removed from final output. Default is no removal.

Output results

snpEff (General variant annotation) (multi-sample)

`data:variants:vcf:snpeff:snpeff` (*data:variants:vcf* **variants**, *basic:string* **database**, *data:variants:vcf* **dbsnp**, *basic:string* **filtering_options**, *list:data:geneset* **sets**, *list:basic:string* **extract_fields**, *basic:boolean* **one_per_line**) [Source: v1.1.1]

Annotate variants with SnpEff.

SnpEff is a variant annotation and effect prediction tool. It annotates and predicts the effects of genetic variants (such as amino acid changes).

This process also allows filtering of variants with ``SnpSift filter`` command and extracting specific fields from the VCF file with ``SnpSift extractFields`` command.

This tool works with multi-sample VCF file as an input.

Input arguments variants

label
Variants (VCF)

type
`data:variants:vcf`

required
True

disabled
False

hidden
False

database

label
snpEff database

type
`basic:string`

required
True

disabled
False

hidden
False

default
`GRCh38.99`

choices

- GRCh37.75: `GRCh37.75`
- GRCh38.99: `GRCh38.99`

dbsnp

label
Known variants

type
`data:variants:vcf`

description
List of known variants for annotation.

required
False

disabled
False

hidden
False

filtering_options

label
Filtering expressions

type
`basic:string`

description
Filter VCF file using arbitraty expressions.Examples of filtering expressions: '(ANN[*].GENE = 'PSD3')' or '(REF = 'A')' or '(countHom() > 3) | ((exists INDEL) & (QUAL >= 20)) | (QUAL >= 30)'.For more information checkout the official documentation of [SnpSift](https://pcingola.github.io/SnpEff/ss_filter/)

required
False

disabled
False

hidden
False

sets

label
Files with list of genes

type
`list:data:geneset`

description
Use list of genes, if you only want variants reported for them. Each file must have one string per line.

required
False

disabled
False

hidden
!filtering_options

extract_fields

label
Fields to extract

type
`list:basic:string`

description
Write fields you want to extract from annonated vcf file and press Enter after each one. Example of fields: `CHROM POS REF ALT 'ANN[*].GENE'`. For more information follow this [link](https://pcingola.github.io/SnpEff/ss_extractfields/).

required
False

disabled
False

hidden
False

advanced.one_per_line

label
One effect per line

type
`basic:boolean`

description
If there is more than one effect per variant, write them to seperate lines.

required
True

disabled
False

hidden
False

default
`False`

Output results vcf

label
Annotated variants (VCF)

type
`basic:file`

required
True

disabled
False

hidden
False

tbi

label
Index of annotated variants

type
`basic:file`

required
True

disabled
False

hidden
False

vcf_extracted

label
Extracted annotated variants (VCF)

type
`basic:file`

required
False

disabled
False

hidden
False

tbi_extracted

label
Index of extracted variants

type
`basic:file`

required
False

disabled
False

hidden
False

species

label
Species

type
`basic:string`

required
True

disabled
False

hidden
False

build

label
Build

type
`basic:string`

required
True

disabled
False

hidden
False

genes

label
SnpEff genes

type
`basic:file`

required
True

disabled
False

hidden
False

summary

label
Summary

type
`basic:file:html`

required
True

disabled
False

hidden
False

snpEff (General variant annotation) (single-sample)

`data:variants:vcf:snpeff:single:snpeff-single` (*data:variants:vcf* **variants**, *basic:string* **database**, *data:variants:vcf* **dbsnp**, *basic:string* **filtering_options**, *list:data:geneset* **sets**, *list:basic:string* **extract_fields**, *basic:boolean* **one_per_line**) [Source: v1.0.1]

Annotate variants with SnpEff.

SnpEff is a variant annotation and effect prediction tool. It annotates and predicts the effects of genetic variants (such as amino acid changes).

This process also allows filtering of variants with ``SnpSift filter`` command and extracting specific fields from the VCF file with ``SnpSift extractFields`` command.

This tool works with single-sample VCF file as an input.

Input arguments variants

label
Variants (VCF)

type
`data:variants:vcf`

required
True

disabled
False

hidden
False

database

label
snpEff database

type
`basic:string`

required
True

disabled
False

hidden
False

default
`GRCh38.99`

choices

- GRCh37.75: `GRCh37.75`
- GRCh38.99: `GRCh38.99`

dbsnp

label
Known variants

type
`data:variants:vcf`

description
List of known variants for annotation.

required
False

disabled
False

hidden
False

filtering_options

label
Filtering expressions

type
`basic:string`

description
Filter VCF file using arbitraty expressions.Examples of filtering expressions: '(ANN[*].GENE = 'PSD3')' or '(REF = 'A')' or '(countHom() > 3) | ((exists INDEL) & (QUAL >= 20)) | (QUAL >= 30)'.For more information checkout the official documentation of [SnpSift](https://pcingola.github.io/SnpEff/ss_filter/)

required
False

disabled
False

hidden
False

sets

label
Files with list of genes

type
`list:data:geneset`

description
Use list of genes, if you only want variants reported for them. Each file must have one string per line.

required
False

disabled
False

hidden
!filtering_options

extract_fields

label
Fields to extract

type
`list:basic:string`

description
Write fields you want to extract from annonated vcf file and press Enter after each one. Example of fields: `CHROM POS REF ALT 'ANN[*].GENE'`. For more information follow this [link](https://pcingola.github.io/SnpEff/ss_extractfields/).

required
False

disabled
False

hidden
False

advanced.one_per_line

label
One effect per line

type
`basic:boolean`

description
If there is more than one effect per variant, write them to seperate lines.

required
True

disabled
False

hidden
False

default
`False`

Output results vcf

label
Annotated variants (VCF)

type
`basic:file`

required
True

disabled
False

hidden
False

tbi

label
Index of annotated variants

type
`basic:file`

required
True

disabled
False

hidden
False

vcf_extracted

label
Extracted annotated variants (VCF)

type
`basic:file`

required
False

disabled
False

hidden
False

tbi_extracted

label
Index of extracted variants

type
`basic:file`

required
False

disabled
False

hidden
False

species

label
Species

type
`basic:string`

required
True

disabled
False

hidden
False

build

label
Build

type
`basic:string`

required
True

disabled
False

hidden
False

genes

label
SnpEff genes

type
`basic:file`

required
True

disabled
False

hidden
False

summary

label
Summary

type
`basic:file:html`

required
True

disabled
False

hidden
False

1.3 Descriptor schemas

When working with the biological data, it is recommended (and often required) to properly annotate samples. The annotation information attached to the samples includes information about *organism*, *source*, *cell type*, *library preparation protocols* and others.

The annotation fields associated with the samples or related sample files are defined in the descriptor schemas. This tutorial describes the descriptor schemas that are attached to the sample objects, raw sequencing reads and differential expressions files.

Other available descriptor schemas can be explored at the Resolwe-bio GitHub page. Customized descriptor schemas can be created using the Resolwe SDK.

1.3.1 Sample

When a new data object that represents a biological sample (i.e. fastq files, bam files) is uploaded to the database, the unannotated sample (presample) is automatically created. When annotation is attached to the presample object, this object is automatically converted to the annotated sample. To annotate the sample, we need to define a descriptor schema that will be used for the annotation. Together with the descriptor schema, we need to provide the annotations (descriptors) that populate the annotation fields defined in the descriptor shema. The details of this process are described in the Resolwe SDK documentation.

To annotate the sample in a GEO compliant way, we prepared the sample annotation schema. An example of the customized descriptor schema is also available.

1.3.2 Reads

To annotate raw sequencing reads we have prepared corresponding reads descriptor schema.

1.3.3 Differential expression

To define the default thresholds for `p-value`, `log fold change (FC)` and to describe which samples are used as cases and which as controls in the calculation of differential expression we have prepared diffexp descriptor schema.

1.4 Reference

1.4.1 Utilities

Test helper functions.

class resolwe_bio.utils.test.**BioProcessTestCase**(*methodName='runTest'*)

Base class for writing bioinformatics process tests.

It is a subclass of Resolwe's `ProcessTestCase` with some specific functions used for testing bioinformatics processes.

prepare_annotation(*fn='sp_test.gtf'*, *source='DICTYBASE'*, *species='Dictyostelium discoideum'*, *build='dd-05-2009'*)

Prepare annotation GTF.

prepare_annotation_gff(*fn='annotation dicty.gff.gz'*, *source='DICTYBASE'*, *species='Dictyostelium discoideum'*, *build='dd-05-2009'*)

Prepare annotation GFF3.

prepare_bam(*fn='sp_test.bam'*, *species='Dictyostelium discoideum'*, *build='dd-05-2009'*)

Prepare alignment BAM.

prepare_expression(*f_rc='exp_1_rc.tab.gz'*, *f_exp='exp_1_tpm.tab.gz'*, *f_type='TPM'*, *name='Expression'*, *source='DICTYBASE'*, *descriptor=None*, *feature_type='gene'*, *species='Dictyostelium discoideum'*, *build='dd-05-2009'*)

Prepare expression.

prepare_paired_reads(*mate1=['fw reads.fastq.gz']*, *mate2=['rw reads.fastq.gz']*)

Prepare NGS reads FASTQ.

prepare_reads(*fn=['reads.fastq.gz']*)

Prepare NGS reads FASTQ.

prepare_ref_seq(*fn='adapters.fasta'*, *species='Other'*, *build='Illumina adapters'*)

Prepare reference sequence FASTA.

setUp()

Initialize test files path and species annotation.

class resolwe_bio.utils.test.**KBBioProcessTestCase**(*methodName='runTest'*)

Class for bioinformatics process tests that use knowledge base.

It is based on *BioProcessTestCase* and Django's `LiveServerTestCase`. The latter launches a live Django server in a separate thread so that the tests may use it to query the knowledge base.

run_process(**args*, ***kwargs*)

Run processes in collection.

setUp()

Set up test gene information knowledge base, create collection.

resolwe_bio.utils.test.**skipDockerFailure**(*reason*)

Skip decorated tests due to failures when run in Docker.

Unless `TESTS_SKIP_DOCKER_FAILURES` Django setting is set to `False`. `reason` should describe why the test is being skipped.

resolwe_bio.utils.test.**skipUnlessLargeFiles**(**files*)

Skip decorated tests unless large files are available.

Parameters

***files** (*list*) – variable lenght files list, where each element represents a large file path relative to the `TEST_LARGE_FILES_DIR` directory

1.5 Change Log

All notable changes to this project are documented in this file. This project adheres to Semantic Versioning.

1.5.1 Unreleased

Added

Changed

- Make changes to input fields in `workflow-cutnrun-beta`

Fixed

1.5.2 55.1.0 - 2023-12-18

Added

- Add `samtools-bedcov` process
- Add `workflow-cutnrun-beta` workflow

Changed

- Use downsampled alignment for `rnaseqc-qc` process in `workflow-bbduk-star-featurecounts-qc` and `workflow-bbduk-star` workflows
- Unify the use of `resolwebio/common:4.1.1` Docker image version across processes
- Unify the use of `resolwebio/base:ubuntu-22.04-14112023` Docker image across processes
- Add normalized count matrix output to `differentialexpression-deseq2` process

Fixed

- Fix `qorts-qc` process so it can be used with hg19 annotation and improve error handling when files are missing

1.5.3 55.0.0 - 2023-11-13

Changed

- **BACKWARD INCOMPATIBLE:** Require Resolwe 38.x
- Change `rnaseqc-qc` data object label in `workflow-bbduk-star-featurecounts-qc`, `workflow-bbduk-star-qc` and `workflow-bbduk-salmon-qc` workflows
- Add `rnaseqc-qc` to `workflow-bbduk-star-featurecounts-qc`, `workflow-bbduk-star-qc` and `workflow-bbduk-salmon-qc`
- Add `qorts-qc` to `workflow-bbduk-star-featurecounts-qc`

Fixed

- Fixed stranded input options in `rnaseqc-qc` process
- Change GEO metadata test file for `geo-import` process and fix corresponding test function

1.5.4 54.0.0 - 2023-10-23

Changed

- **BACKWARD INCOMPATIBLE:** Require Resolwe 37.x
- Add `sample_annotation` Jinja expressions filter that accepts an annotation path and returns its value
- Update field paths for sample annotation in `geo-import` process
- Update species annotation in `alignment-star` process

Fixed

- Fix Cut&Run workflow to scale the correct input BAM file using the spike-in scaling factor

1.5.5 53.2.0 - 2023-09-18

Fixed

- Report gene body coverage by `rnaseqc-qc` and assigned reads by `star-quantification` for all samples in MultiQC

1.5.6 53.1.0 - 2023-08-15

Added

- Add Euclidean distance metric in `find-similar` and `clustering-hierarchical-etc` processes

Changed

- Change `star-quantification` process to include number of assigned reads in the summary report
- Change `MultiQC` report to include assigned reads from `star-quantification` process
- Change `workflow-bbduk-star-qc` workflow to include assigned reads by `star-quantification` in `MultiQC` report
- Bump storage requirements in processes `alignment-bwa-sw` and `alignment-bwa-aln`

1.5.7 53.0.0 - 2023-07-14

Added

- Add support for `Python 3.11`

Changed

- **BACKWARD INCOMPATIBLE:** Require Resolwe 36.x
- **BACKWARD INCOMPATIBLE:** Require Django 4.2
- Rename `filter_sense_rate` to `filter_rnaseqc_metrics` in `filter.py` and add additional fields for filtering
- Changed the ordering of exons in `format_ucsc` function of `rnaseq-qc` process and add collapse_only option for stranded protocols
- Change test files and modify test function for `rnaseqc-qc` process
- Bump `postgresql` and `redis` containers version
- Rename `workflow-rnaseq-variantcalling` from `RNA-seq Variant Calling` to `RNA-seq Variant Calling Workflow`
- Support optional triggering of MultiQC in `workflow-rnaseq-variantcalling`
- Add `mask` option to processes `gatk-variant-filtration` and `gatk-variant-filtration-single` and enable this option in `workflow-rnaseq-variantcalling`

1.5.8 52.1.0 - 2023-06-19

Added

- Add command `filter_features` to the listener that caches the returned `Feature` objects to speed up the queries
- Add `rnaseqc-qc` process including a function for parsing UCSC annotations and modify `multiqc` to accomodate its results

Changed

- Change test files for `multiqc`
- Modify `filter` command on the `Feature` class to use cache
- Explicitly set process resources in processes `alignment-bowtie` and `alignment-bowtie2`
- Improve warnings in `geo-import` process for series without raw data

Fixed

- Fix SRA parameters to pass zero values in `geo-import` workflow

1.5.9 52.0.0 - 2023-05-15

Changed

- **BACKWARD INCOMPATIBLE:** Require Resolwe 35.x
- Support the use of BAM file as an input in the workflow `workflow-rnaseq-variantcalling`
- Adjust assigned CPU core count requirements for processes
- Always use `Docker` default seccomp profile

Fixed

- Fix `min_read_len` parameter to pass zero values in `import-sra-single`, `import-sra-paired`, `import-sra`
- Prevent `bbduk-single` and `bbduk-paired` processes from stopping when the number of lanes exceeds the number of available cores
- Fix BBDuk settings in the workflow `workflow-rnaseq-variantcalling`

1.5.10 51.0.0 - 2023-04-15

Added

- Add option to filter by genotype fields in processes `gatk-variant-filtration` and `gatk-variant-filtration-single`
- Add option for 2-pass mode to workflow `workflow-bbduk-star-qc`

Changed

- **BACKWARD INCOMPATIBLE:** Require Resolwe 34.x
- Bump the STAR version in processes `alignment-star` and `alignment-star-index`. Deprecate test processes `alignment-star-new`, `alignment-star-index-new` and `workflow-bbduk-star-qc-new`
- Support filtering by genotype fields in the process `mutations-table` and change the default options for variant filtration in the workflow `workflow-rnaseq-variantcalling`
- Optimize resource usage for `SchedulingClass.INTERACTIVE` processes

Fixed

- Add output field `feature_type` to the process `star-quantification`

1.5.11 50.0.0 - 2023-03-13

Added

- Add option to compute gene counts in `alignment-star` process
- Add processes `alignment-star-new` and `alignment-star-index-new` that use STAR version 2.7.10b
- Add process `star-quantification`
- Add workflows `workflow-bbduk-star-qc` and `workflow-bbduk-star-qc-new`

Changed

- **BACKWARD INCOMPATIBLE:** Require Resolwe 33.x
- **BACKWARD INCOMPATIBLE:** Drop support for `Python <= 3.9`
- **BACKWARD INCOMPATIBLE:** Remove default ordering from Feature API endpoint
- Update ordering weights for full-text search on Feature API endpoint

1.5.12 49.0.0 - 2023-02-13

Added

- Add per-lane processing for processes `alignment-star`, `feature_counts`, `bbduk-single`, `bbduk-paired` and workflow `workflow-bbduk-star-featurecounts-qc`
- Add option of interval padding to process `vc-gatk4-hc`
- Add process `snpeff-single`

Changed

- **BACKWARD INCOMPATIBLE:** Remove comparison to reference in process `mutations-table`
- **BACKWARD INCOMPATIBLE:** Update workflow `workflow-rnaseq-variantcalling`:
  - merge with functionality from deprecated workflow `workflow-rnaseq-variantcalling-beta`
  - add processes `snpeff-single` and `mutations-table`
  - remove process `gatk-select-variants-single`
- Remove group nesting for QC fields in `general` descriptor schema
- Add memory limit to parallel GATK SplitNCigarReads step in process `rnaseq-vc-preprocess`
- Update process categories
- Remove `Show advanced options` checkbox from all processes and workflows

Fixed

- Enable ordering on knowledge-base endpoints
- Fix the check for genome builds of inputs in process `snpeff`

1.5.13 48.0.0 - 2022-11-14

Added

- Add `REDIS_CONNECTION_STRING` setting needed by the `Resolwe`
- Add process `samtools-view`
- Add process `samtools-coverage`

Changed

- **BACKWARD INCOMPATIBLE:** Require Resolwe 32.x
- Support GEO series with EBI samples in `geo-import` process

Fixed

- Output all detected input variants in `mutations-table` process
- Change `__` separators in field names to `_` in `general` descriptor schema
- Change choice values of the field `biomarkers_pdl1_tps_cat` in `general` descriptor schema

1.5.14 47.3.1 - 2022-10-18

Fixed

- Rename field `general_information` to `general` in `general` descriptor schema

1.5.15 47.3.0 - 2022-10-17

Changed

- Flatten General descriptor schema

Fixed

- Bump version of `rnaseq-vc-preprocess` process
- Fix sample naming in `multiqc` process to avoid leaving out data in the MultiQC report

1.5.16 47.2.0 - 2022-09-19

Added

- Add `dicty-time-series` time series descriptor schema

Changed

- Add options to calculate variance in abundance estimates in the workflow `workflow-bbduk-salmon-qc`
- Support geneset as input to process `mutations-table`
- Bump memory requirement to 64 GB and limit memory of parallel SplitNCigarReads step in process `rnaseq-vc-preprocess`
- Revert `workflow-rnaseq-variantcalling` to run individual data preprocess steps

1.5.17 47.1.0 - 2022-08-19

Added

- Add optional calculation of variance in abundance estimates in the process `salmon-quant`
- Add process `rnaseq-vc-preprocess`

1.5.18 47.0.0 - 2022-07-18

Added

- Add `general` descriptor schema

Changed

- **BACKWARD INCOMPATIBLE:** Require Resolwe 31.x
- Use all three fragment length estimates before failing due to negative the estimate in `macs2-callpeak` process

Fixed

- Fix ChIPQC plot rendering in `multiqc` process for samples containing file extensions in their name
- Update SRA url for fetching experiment metadata in `geo-import` process

1.5.19 46.0.0 - 2022-06-13

Added

- Add `xengsort-index` and `xengsort-classify` proceses

Changed

- **BACKWARD INCOMPATIBLE:** Generalize the `scale-bigwig` process and rename it to `calculate-bigwig`
- Use `resolwebio/wgbs:3.0.0` in `walt`, `methcounts`, `hmr` and `bs-conversion-rate` processes
- Use `resolwebio/chipseq:6.0.0` in `macs2-callpeak`, `macs14`, `qc-prepeak`, `chipseq-peakscore`, `chipseq-genescore` and `upload-bed` processes
- Change `merge-fastq-single` and `merge-fastq-paired` process type
- Use `resolwebio/chipseq:6.1.0` in `chipqc` process
- Use `resolwebio/methylation_arrays:1.1.0` in the `methylation-array-sesame` process
- Improve error reporting for invalid fragment length estimates and fix memory issues with MarkDuplicates in `macs2-callpeak` process
- Remove `make_report.py` script from resolwe-bio

Fixed

- Fix sporadically failing tests of `macs2-callpeak` by removing pipes in Plumbum commands
- Fix variants_filtered output in `filtering-chemut` process
- Fix typo in `alignment-star` process
- Remove unused tools bigwig_chroms_to_ucsc.py and check_bam_source.py

1.5.20 45.0.0 - 2022-05-13

Added

Changed

- **BACKWARD INCOMPATIBLE:** Update GATK to GATK4 in process `vc-chemut` and update the workflow `workflow-chemut`
- Rewrite the process `filtering-chemut` to Python
- Remove slamseq processes `alleyoop-collapse`, `alleyoop-rates`, `alleyoop-snpeval`, `alleyoop-summary`, `alleyoop-utr-rates`, `slam-count`, `slamdunk-all-paired` the workflow `workflow-slamdunk-paired` and related code in `multiqc`
- Use `resolwebio/common:3.1.0` in `upload-metadata-unique` and `upload-metadata` processes
- Use the parent Data object name for the data name of processes and workflows previously named after the sample name of the input file
- Remove Docker files from project

- Remove BigWig outputs created with `bamtobigwig.sh` script in processes `walt`, `alignment-bowtie`, `alignment-bowtie2`, `alignment-bwa-mem`, `alignment-bwa-sw`, `alignment-bwa-aln`, `alignment-bwa-mem2`, `alignment-hisat2`, `upload-bam`, `upload-bam-indexed`, `upload-bam-secondary`, `alignmentsieve`, `bamclipper`, `bqsr`, `markduplicates`, `bam-split`, `umi-tools-dedup` and workflow `workflow-cutnrun`

Fixed

- Update the process `mutations-table` so that it handles empty input VCF files

1.5.21 44.1.0 - 2022-04-15

Added

- Add processes `gatk-variant-filtration-single` and `gatk-select-variants-single`
- Add ExtendedCollectionFilter filter to allow filtering collections by samples containing given species, tissue type, outcome or treatment type
- Add process `reference-space` and `upload-ml-expression`
- Rewrite `macs2-callpeak` process to Python
- Add process `mutations-table`

Changed

- Specify tmp dir for GATK processes
- Attach workflow data objects to Samples
- Remove `workflow-accel` pipeline and related process: `align-bwa-trim`, `coveragebed`, `picard-pcrmetrics`, `upload-picard-pcrmetrics`, `upload-picard-pcrmetrics`, `vc-realign-recalibrate`, `vc-gatk-hc`, `lofreq`, `snpeff-legacy`, `amplicon-report`, `amplicon-table`, `upload-master-file`, `amplicon-archive-multi-report`, `upload-snpeff`
- Rewrite processes to Python and add `geneset` DescriptorSchema to Data made by processes:
  - `upload-geneset`
  - `create-geneset`
  - `create-geneset-venn`

Fixed

- Attach GATK VariantFiltration and SelectVariants output to the Sample object in `workflow-rnaseq-variantcalling` pipeline
- Set `Persistence` property to `TEMP` for processes `find-similar` and `clustering-hierarchical-etc`
- Fix input schema in pipeline `workflow-rnaseq-variantcalling`
- Fail gracefully when no relation labels are found in `merge-fastq-single` and `merge-fastq-paired` processes

1.5.22 44.0.0 - 2022-03-14

Added

- Add process `gatk-split-ncigar`
- Add process `gatk-variant-filtration`
- Add process `snpeff`
- Add workflow `workflow-rnaseq-variantcalling`
- Add support for `Python` 3.10

Changed

- **BACKWARD INCOMPATIBLE:** Require Resolwe 30.x
- **BACKWARD INCOMPATIBLE:** Change the slug of the process `snpeff` to `snpeff-legacy`
- **BACKWARD INCOMPATIBLE:** Deprecate process `upload-orange-metadata` in favour of processes `upload-metadata` and `upload-metadata-unique`
- Add parameter `--use-original-qualities` to `bqsr` process
- Add `fn_ignore_dirs` options to MultiQC configuration file in `resolwebio/common:3.0.1` Docker image
- Add parameter `--exclude-filtered` to `gatk-select-variants` process
- Normalize processes that use `resolwebio/dnaseq` Docker image to use the latest version 6.3.1 and modify tests as necessary
- Rewrite process `vc-gatk4-hc` to Python
- Prepare `resolwebio/rnaseq:6.0.0` Docker image:
  - Update Python package versions for Python 3.8
  - Pin R package versions
  - Bump genome-tools to 1.6.2
- Add Java memory settings to processes `bqsr` and `markduplicates`
- Update SnpEff version to 5.1 in `resolwebio/snpeff:2.1.0` Docker image
- Add additional file output with source ids and target ids to process `goenrichment`
- Normalize all processes that rely on `rnaseq` Docker image to use the latest `resolwebio/rnaseq:6.0.0` Docker image version

Fixed

- Fix `--cl-config` input option in MultiQC process. Use `resolwebio/common:3.0.1` Docker image with updated MultiQC configuration file to omit parsing the unwanted `tmp` folder
- Fix LISTENER_CONNECTION settings to work on Mac
- Add tool Tabix to `resolwebio/snpeff:2.1.1` Docker image

1.5.23 43.0.0 - 2022-02-14

Added

- Add bcftools version 1.14 to `resolwebio/common` Docker image

Changed

- **BACKWARD INCOMPATIBLE:** Rewrite processes `workflow-bbduk-star-featurecounts-qc-single` and `workflow-bbduk-star-featurecounts-qc-paired` to Python
- **BACKWARD INCOMPATIBLE:** Rewrite workflows `workflow-bbduk-star-fc-quant-single`, `workflow-bbduk-star-fc-quant-paired`, `workflow-cutadapt-star-fc-quant-single` and `workflow-cutadapt-star-fc-quant-wo-depletion-single` to Python
- **BACKWARD INCOMPATIBLE:** Rewrite workflows `workflow-bbduk-salmon-qc-single` and `workflow-bbduk-salmon-qc-paired` to Python
- Changes to `resolwebio/common:3.0.0` Docker image include pinning of R version and corresponding packages, fixed Python to 3.8, updated picard-tools to version 2.26.10, updated samtools to version 1.14 and updated MultiQC to version 1.11
- Normalize processes that use `resolwebio/common` Docker image to use the latest version 3.0.0 and modify tests as necessary
- Bump GATK version to 4.2.4.1 and pin R package versions in `resolwebio/dnaseq:6.3.0` Docker image
- **BACKWARD INCOMPATIBLE:** Bump Django requirement to version 3.2

Fixed

- Fix file import and process progress updates in `upload-fasta-nucl`
- Fix Ensembl-VEP installation in `resolwebio/dnaseq:6.3.1` Docker image

1.5.24 42.0.0 - 2022-01-14

Added

- Add an action for resolving pasted genes on Feature endpoint
- Make Knowledge base squashed migration reversable

Changed

- **BACKWARD INCOMPATIBLE:** Run `ensembl-vep` process offline and add mandatory reference sequence input
- **BACKWARD INCOMPATIBLE:** Remove Diagenode CATS RNA-seq pipeline and related tools (`rsem`, `index-fasta-nucl`, `workflow-custom-cutadapt-star-htseq-single`, `workflow-custom-cutadapt-star-htseq-paired`, `workflow-custom-cutadapt-star-rsem-single`, `workflow-custom-cutadapt-star-rsem-paired`, `cutadapt-custom-single`, `cutadapt-custom-paired`).

- **BACKWARD INCOMPATIBLE:** Remove HTSeq-count tool and related workflows (`htseq-count`, `htseq-count-raw`, `workflow-rnaseq-single`, `workflow-rnaseq-paired`, `workflow-bbduk-star-htseq`).
- **BACKWARD INCOMPATIBLE:** Remove redundant `wgs-preprocess` process
- **BACKWARD INCOMPATIBLE:** Unify Feature autocomplete and search endpoints into a single endpoint
- Rewrite `goenrichment` process to Python
- Rewrite process `basespace-file-import` to Python
- Change Ensembl-VEP version check in `ensembl-vep` process
- Rename `featureCounts` class to `FeatureCounts`
- Add scatter-gather approach for `BaseRecalibrator` and `ApplyBQSR` in `wgs-preprocess-bwa2` process

Fixed

- Fix build mismatch error message in `differentialexpression-deseq2`
- Fix how `self.progress` is called in `FeatureCounts`

1.5.25 41.0.0 - 2021-12-13

Added

- Add `gatk-select-variants` process

Changed

- **BACKWARD INCOMPATIBLE:** Rewrite `alignment-star` and `alignment-star-index` processes to Python
- **BACKWARD INCOMPATIBLE:** Rewrite processes `upload-expression` and `upload-expression-cuffnorm` to Python
- Rewrite processes `seqtk-sample-single` and `seqtk-sample-paired` to Python
- Rewrite `bbduk-single` and `bbduk-paired` processes to Python
- Rewrite processes `upload-fastq-single`, `upload-fastq-paired`, `files-to-fastq-single` and `files-to-fastq-paired` to Python
- Rewrite processes `clustering-hierarchical-samples` and `clustering-hierarchical-genes` to Python
- Add java memory setting and remove unused inputs in `gatk-genotype-gvcfs`
- Change the `independent` field to True by default in process `differentialexpression-deseq2` to match the behaviour of the R script
- Add `--fork` parameter in `ensembl-vep` process

1.5.26 40.0.0 - 2021-11-12

Added

- Add `gatk-merge-vcfs` process

Changed

- **BACKWARD INCOMPATIBLE:** Require Resolwe 29.x
- **BACKWARD INCOMPATIBLE:** Update `workflow-wgs-gvcf` to include BWA-MEM2-based preprocessing step. Support triggering the pipeline using a pre-aligned BAM input file
- Add `qin` and `ignorebadquality` BBDuk options to workflows `workflow-bbduk-salmon-qc-single`, `workflow-bbduk-salmon-qc-paired`, `workflow-bbduk-star-featurecounts-qc-single`, `workflow-bbduk-star-featurecounts-qc-paired`, `workflow-bbduk-star-fc-quant-single`, `workflow-bbduk-star-fc-quant-paired`, `workflow-bbduk-star-htseq`, `workflow-bbduk-star-htseq-paired`
- Replace ReSDK interface with Python API calls when accesing the gene KB in processes `salmon-quant`, `alleyoop-collapse`, `slam-count` and `mapped-microarray-expression`
- Use downsampled alignment for running QoRTs in 3' mRNA-Seq workflows `workflow-bbduk-star-fc-quant-single`, `workflow-bbduk-star-fc-quant-paired`, `workflow-cutadapt-star-fc-quant-single`, `workflow-cutadapt-star-fc-quant-wo-depletion-single`
- Replace `ANONYMOUS_USER_ID` with `ANONYMOUS_USER_NAME` in settings.py
- Add java memory settings in `gatk-genomicsdb-import`

Fixed

- Fix data name and advanced options in `variants-to-table` process

1.5.27 39.0.0 - 2021-10-19

Added

- Add `gatk-refine-variants` process
- Add ensembl-vep tool to the `resolwebio/dnaseq:6.2.0` Docker image
- Add `upload-vep-cache` process
- Add `ensembl-vep` process
- Add `variants-to-table` process

Changed

- **BACKWARD INCOMPATIBLE:** Update `merge-fastq-single` and `merge-fastq-paired` processes to use sample relations for merging FASTQ files
- **BACKWARD INCOMPATIBLE:** Rewrite `feature_counts` process to Python
- **BACKWARD INCOMPATIBLE:** Create a separate process `gatk-genomicsdb-import` for importing GVCFs into the database and use it as an input in the parallelised `gatk-genotype-gvcfs` process
- Support ChIP-Seq and ATAC-Seq data sets in `geo-import` process
- Replace the stdout/stdin file interface in SortSam / SetNmMdAndUqTags stage of the `wgs-preprocess-bwa2` with two distinct analysis steps

1.5.28 38.4.0 - 2021-09-14

Added

- Add BWA-mem2 to the `resolwebio/common` Docker image
- Add `bwamem2-index` process
- Add `bwamem2` process
- Add `wgs-preprocess-bwa2` process
- Add `upload-bwamem2-index` process

Changed

- Use `resolwebio/common:2.9.0` Docker image version in `resolwebio/dnaseq` Docker image
- Optimize CPU usage in process `gatk-haplotypecaller-gvcf`
- Make the read trimming step (trimmomatic) optional in the `workflow-wgs-gvcf` workflow
- Add aligned reads (BAM format) as an alternative input option in the `wgs-preprocess` process
- Set the requirements for number of cores from 20 to 4 and memory from 16 GB to 32 GB in `alignment-bwa-mem` process

Fixed

- Fix an edge case in `methylation-array-sesame` process where calling the `sesame.R` script using Plumbum was failing for some compressed IDAT inputs due to the file encoding issues

1.5.29 38.3.0 - 2021-08-16

Changed

- Replace Bedtools with Samtools for BAM to FASTQ file format conversion in `bamtofastq-paired` process
- Bump docker image version in `methylation-array-sesame` process
- Add `qin` and `ignorebadquality` options to `bbduk-single` and `bbduk-paired` processes

Fixed

- Use clean file name for gene sets from differential expressions
- Fix saving estimated counts output in `tximport_summarize.R` script
- Add `config.yaml` to methylation_arrays Dockerfile
- Use raw SigSet for performing QC in the SeSAMe pipeline

1.5.30 38.2.0 - 2021-07-13

Added

- Add more information about output to the `methylation-array-sesame` pipeline documentation
- Support filtering by `subject_information.sample_label`, `subject_information.subject_id`, `subject_information.batch`, `subject_information.group`, `disease_information.disease_type`, `disease_information.disease_status`, `immuno_oncology_treatment_type.io_drug`, `immuno_oncology_treatment_type.io_treatment`, `response_and_survival_analysis.confirmed_bor`, `response_and_survival_analysis.pfs_event`, `general.description`, `general.biosample_source`, and `general.biosample_treatment` fields in sample descriptor on API

Changed

- Improve automatic sample naming in the `geo-import` process

Fixed

- Fix stalled sam-to-bam conversion in `wgs-preprocess` process
- Return column betas to `methylation-array-sesame` pipeline output

1.5.31 38.1.1 - 2021-06-14

Changed

- Remove mapping of probe_ids to ENSEMBL ids and add extra variables in `methylation-array-sesame` process

1.5.32 38.1.0 - 2021-06-14

Added

- Add `wgs-preprocess` process
- Add `gatk-haplotypecaller-gvcf` process
- Add `workflow-wgs-gvcf` process
- Add `gatk-genotype-gvcfs` process
- Add `gatk-vqsr` process
- Add `bamtofastq-paired` process
- Add `methylation_array` docker image
- Add `methylation-array-sesame` process
- Add support for Python 3.9
- Support downloading knowledge base features and mappings from S3 bucket
- Cap process memory consumption at 10GB

Changed

- Bump GATK to version 4.2.0.0 in `resolwebio/dnaseq:6.0.0` Docker image
- Update `workflow-mirna`
- Add new parameters -maximumlength/-M and -no-indels in processes `cutadapt-single` and `cutadatp-paired`
- Add new `id_attribute` to `feature_counts` process

Fixed

- Remove some duplicated code in `test_probe_mapping`
- Rename FastQC output bundle in Trimmomatic processes so that the reports are correctly sorted/included in MultiQC reports
- Fix method signature for KB feature/mapping filtering

1.5.33 38.0.0 - 2021-05-17

Added

- Add bioservices python package to the `resolwebio/common:2.8.0` Docker image
- Add `upload-idat` process
- Add `upload-microarray-expression` and `mapped-microarray-expression` processes
- Add `map-microarray-probes` process

Changed

- **BACKWARD INCOMPATIBLE:** Support microarray expressions upload in `geo-import` process
- Trigger an error for microarray data in differential expression processes `differentialexpression-edger` and `differentialexpression-deseq2`

1.5.34 37.0.0 - 2021-04-19

Added

- Add GEOparse to the `resolwebio/common:2.7.0` Docker image
- Add fastq file validation in `import-sra-single` and `import-sra-paired` processes
- Add `geo-import` process

Changed

- **BACKWARD INCOMPATIBLE:** Require Resolwe 28.x
- Use `resolwebio/base:ubuntu-20.04` Docker image for building `resolwebio/sra-tools` Docker image. Include `dnaio` Python library in `resolwebio/sra-tools`.

Fixed

- Fix handling of non-sample data inputs in `multiqc` process

1.5.35 36.1.0 - 2021-03-15

Added

- Fail if wrong filtering arguments are used in KB Feature / Mapping search endpoints

Changed

- Use Amazon ECR when building `resolwebio/base` Docker images
- Use pinned version of the `resolwebio/base` Docker image for building `resolwebio/common` Docker image. Update versions of bioinformatic tools installed in the `resolwebio/common` image.
- Use only tagged versions of `resolwebio/base` Docker images in processes
- Save gene-level estimated counts to the `rc` output field in the `salmon-quant` process

Fixed

- Fix file import in processes `upload-multiplexed-single` and `upload-multiplexed-paired`
- Fix `import-sra-single` and `import-sra-paired` to correctly determine Illumina 1.5 and 1.3 quality encoding

1.5.36 36.0.0 - 2021-02-22

Changed

- **BACKWARD INCOMPATIBLE:** Require Resolwe 27.x
- Move docker images from Docker Hub to Amazon ECR

1.5.37 35.0.0 - 2021-01-20

Added

- Add OncXerna specific clinical descriptor schema `oncxerna_clinical`

Changed

- **BACKWARD INCOMPATIBLE:** Support new protocol in Resolwe 26.x

1.5.38 34.3.0 - 2020-12-14

Added

- Add initial general clinical descriptor schema `general_clinical`
- Add `id` field to `Feature` and `Mapping` serializers
- Add `resolwebio/base:ubuntu-20.04` Docker image

Changed

- Update the url for the Orange table example template in `upload-orange-metadata`

1.5.39 34.2.1 - 2020-11-17

Fixed

- Fix `macs2-callpeak` process version

1.5.40 34.2.0 - 2020-11-13

Added

- Add `upload-proteomics-sample` and `upload-proteomics-sample-set` processes for uploading custom tables holding proteomics data

Fixed

- Changed `scale-bigwig` output file field label to `bigwig file`
- Bump memory requirements in processes `import-sra`, `import-sra-single` and `import-sra-paired` to 8GB

1.5.41 34.1.0 - 2020-10-20

Added

- Add peakcalling to removed duplicates step in species' line of the `workflow-cutnrun` workflow

Fixed

- Add BigWig timeout and bin size parameters to `markduplicates`, `alignmentsieve` and `workflow-cutnrun`. Add bin size parameter to `alignment-bowtie2`.

1.5.42 34.0.0 - 2020-10-19

Added

- Added parameters `--normalizeUsing` and `--smoothLength` to script `bamtobigwig.sh` to be used in `bamCoverage` program
- Added parameters `--no-unal` and `--no-overlap` to process `alignment-bowtie`
- Add `alignmentsieve` process
- Add Trim Galore tool to `resolwebio/rnaseq:4.12.0`
- Add `trimgalore-paired` process
- Add `bedtools-bamtobed` and `scale-bigwig` processes
- Added BigWig timeout input parameter to `alignment-bowtie2` process
- Add workflow `workflow-cutnrun`
- Add `clustering-hierarchical-etc` process
- Add `find-similar` process

Changed

- **BACKWARD INCOMPATIBLE:** Require Resolwe 25.x
- **BACKWARD INCOMPATIBLE:** Rewrite `differentialexpression-deseq2` to Python
- Add format parameter to `macs2-callpeak`
- Rewrite `differentialexpression-edger` to Python
- Rewrite `cuffdiff` to Python
- Alignment processes `alignment-bowtie`, `alignment-bowtie2`, `alignment-star`, `alignment-bwa-mem`, `alignment-bwa-sw`, `alignment-bwa-aln`, `alignment-hisat2` and `walt` now issue a warning instead of an error when sample and genome species mismatch
- Support automated upload of gene sets in proceses `cuffdiff`, `differentialexpression-deseq2` and `differentialexpression-edger`
- Support the analysis of S. cerevisiea samples in `macs2-callpeak` process

1.5.43 33.0.0 - 2020-09-14

Added

- Add `resolwebio/sra-tools` Docker image
- Add `resolwebio/orange` Docker image
- Add `upload-orange-metadata` process

Changed

- **BACKWARD INCOMPATIBLE:** Require Resolwe 24.x
- **BACKWARD INCOMPATIBLE:** Include feature full names in full-text search
- Support automatic species annotation in alignment processes: `alignment-bowtie`, `alignment-bowtie2`, `alignment-bwa-mem`, `alignment-bwa-sw`, `alignment-bwa-aln`, `alignment-hisat2`, `alignment-star`, `walt`
- Pin `XML` R package to ensure compatibility with R 3.6.3 in `resolwebio/chipseq:4.1.3` Docker image
- Use `resolwebio/sra-tools:1.0.0` Docker image in processes `import-sra`, `import-sra-single` and `import-sra-paired`
- Optionally use sra-tools `prefetch` command when downloading and converting SRA files to FASTQ format

Fixed

- Bump Docker image version in `chipqc` process to fix enrichment heatmap plot

1.5.44 32.0.0 - 2020-08-17

Added

- Prepare `resolwebio/rnaseq:4.11.0` Docker image: Add rnanorm (1.3.0) RNA-seq normalization package. Use `resolwebio/common:1.6.0` Docker image as a base image. Pin `XML` R package to fix the image build issues. Install BBMap package from Google Drive.

Changed

- **BACKWARD INCOMPATIBLE:** Require Resolwe 23.x.
- **BACKWARD INCOMPATIBLE:** Use rnanorm Python package for TPM/CPM normalization of RNA-seq data in featureCounts and HTSeq-count tools
- Support Nanostring sample reports in MultiQC
- Support Nanostring analysis results in `differentialexpression-deseq2` process

Fixed

- Order results on autocomplete API endpoint in knowledge-base by relevance
- Support filtering by type on knowledge base Feature API
- Attach `rose2` Data object to the input sample

1.5.45 31.0.0 - 2020-07-10

Added

- Add Sample QC information fields to the `sample` descriptor schema

Changed

- **BACKWARD INCOMPATIBLE:** Disable editing capabilities of Knowledge Base API endpoints
- Bump Samtools to version 1.10 in `resolwebio/common:1.6.0` Docker image
- Migrate search for Knowledge Base enpoints from Elasticsearch to PostgreSQL
- Use `resolwebio/common:1.6.0` for the `resolwebio/wgbs:1.3.0` Docker image
- Support samtools markdup report in `walt` process when removing duplicates
- Support samtools markdup report from `walt` in MultiQC
- Support samtools markdup report in `workflow-wgbs-single` and in `workflow-wgbs-paired` workflows
- Bump memory requirements to 32GB in processes: `feature_counts`, `coveragebed`, `library-strandedness`, `qorts-qc`, `salmon-quant` and `vc-realign-recalibrate`
- Rename `workflow-slamdunk-paired` process

Fixed

- Fix read length estimation in `chipqc`

1.5.46 30.0.0 - 2020-06-15

Added

- Add `workflow-subsample-bwa-aln-single` and `workflow-subsample-bwa-aln-paired` workflows

Changed

- **BACKWARD INCOMPATIBLE:** Use Salmon 1.2.1 in `salmon-quant` and `salmon-index` processes
- Salmon quant 1.2.1 is not backwards compatible with indices generated with Salmon index prior to version 1.0.0, thus Salmon tool is updated to version 1.2.1 in processes that utilize Salmon to detect library strandedness type.
- Expose additional limit options in `alignment-star` process
- Bump SRA toolkit to 2.10.0 in `resolwebio/common:1.5.0` Docker image
- Use SRA tookit 2.10.0 in `import-sra`, `import-sra-single` and `import-sra-paired` processes
- Format floats to 2 decimal places in custom ChIP-seq pre/post-peak MultiQC reports

1.5.47 29.0.0 - 2020-05-18

Added

- Add filtered BAM output to `macs2-callpeak` process
- Add an option to use filtered BAM files from `macs2-callpeak` to `rose2`, `workflow-macs-rose`, and `macs2-rose2-batch`
- Add ChIPQC to the `resolwebio/chipseq:4.1.0` Docker image
- Add `chipqc` process

Changed

- **BACKWARD INCOMPATIBLE:** Require Resolwe 22.x
- **BACKWARD INCOMPATIBLE:** Remove processes `alignment-subread` and `subread-index`
- **BACKWARD INCOMPATIBLE:** Remove process `upload-genome`. Refactor processes and workflows that required `data:genome:fasta` type of object on the input to work with `data:seq:nucleotide` or dedicated aligner index files instead.
- Change `macs2-batch` and `macs2-rose2-batch` to use tagAlign files by default
- Bump Salmon to version 1.2.1 in `resolwebio/rnaseq:4.10.0` Docker image. Fix build issues affecting `jpeg` and `png` R packages.
- Support `chipqc` process outputs in MultiQC
- Support `chipqc` in `workflow-macs-rose`, `workflow-macs2`, `macs2-batch` and `macs2-rose2-batch` processes

- Bump memory requirements for process `upload-fasta-nucl` to 8 GB

Fixed

- Fix Data name in `bowtie-index`, `bowtie2-index`, `bwa-index`, `hisat2-index` and `walt-index`
- Fix filtering of empty VCF files in `lofreq` process

1.5.48 28.0.0 - 2020-04-10

Added

- Add `workflow-wgs-paired` workflow
- Add processes: `bowtie-index`, `bowtie2-index`, `bwa-index`, `hisat2-index`, `subread-index` and `walt-index`.
- Add `Dictyostelium purpureum` species choice to `sample` descriptor schema

Changed

- **BACKWARD INCOMPATIBLE:** Refactor `upload-fasta-nucl` process: `species` and `build` input information on FASTA file upload are now mandatory, while `source` input has been removed.
- **BACKWARD INCOMPATIBLE:** Change the `alignment-star-index` process type to `data:index:star`. The process now accepts only `upload-fasta-nucl` objects on input.
- Add trimming with Trimmomatic in `workflow-wgbs-single` and `workflow-wgbs-paired` workflows
- Make intervals an optional input in `bqsr` process
- Make intervals an optional input in `vc-gatk4-hc` process
- Bump memory requirements in `walt` process to 32 GB

Fixed

- Fix data type of adapters input field in `alignment-summary` process
- Fix handling of multiple adapters in `alignment-summary` process

1.5.49 27.0.0 - 2020-03-13

Added

- Add `merge-fastq-single` and `merge-fastq-paired` processes that merge multiple `data:reads:fastq` data objects into a single `data:reads:fastq` data object (and consequently a single sample)
- Add `bs-conversion-rate` process
- Add support for Python 3.8

Changed

- **BACKWARD INCOMPATIBLE:** Require Resolwe 21.x
- **BACKWARD INCOMPATIBLE:** Split `workflow-wgbs` into `workflow-wgbs-single` and `workflow-wgbs-paired` workflows
- Extend the `workflow-wgbs-single` and `workflow-wgbs-paired` with the `markduplicates`, `insert-size` and `bs-conversion-rate` QC processes
- Support detection and separation of control spike-in-derived reads from endogenous sequencing reads in `walt` process
- Replace duplicate-remover in `walt` to unify both (.mr and .bam) output alignment files
- Support `markduplicates` and `bs-conversion-rate` process outputs in `multiqc` reports
- Enable multiple SRR numbers as inputs in processes `import-sra`, `import-sra-single`, and `import-sra-paired`
- Bump memory requirements in `rrbs-metrics` process
- Improve process test input data for the `alignment-star` process
- Bump Bedtools to v2.29.2 in `resolwebio/common:1.3.2` Docker image

Fixed

- Fix Jbrowse track creation in `upload-genome` process. When gzip input was used in `prepare-refseqs.pl`, not all sequence chunks were created for some inputs.
- Fix `macs2-callpeak` process to work with paired-end reads when not using tagAlign files
- Fix `bed_file_corrections_genome_browsers.py` script to handle cases where the input file is empty

1.5.50 26.0.0 - 2020-02-14

Added

- Add `alignment-summary` process
- Add `insert-size` process
- Add `wgs-metrics` process
- Add `rrbs-metrics` process
- Add `workflow-macs2` workflow

Changed

- **BACKWARD INCOMPATIBLE:** Use featureCounts instead of Stringtie in the `workflow-corall-single` and `workflow-corall-paired` workflows
- **BACKWARD INCOMPATIBLE:** Remove `stringtie` and `upload-metabolic-pathway` processes
- **BACKWARD INCOMPATIBLE:** Refactor `walt` process to support Picard quality metrics and update `methcounts` process and to match the new outputs
- **BACKWARD INCOMPATIBLE:** Support MultiQC report in `wgbs` workflow

- Remove Stringtie tool from `resolwebio/rnaseq` Docker image
- Remove `resolwe/base:ubuntu-14.04` and `resolwe/base:ubuntu-17.10` Docker images
- Use pigz for output file compression in `bbduk-single` and `bbduk-paired` processes
- Use `resolwebio/rnaseq:4.9.0` Docker image in processes `bbduk-single`, `bbduk-paired`, `trimmomatic-single`, `trimmomatic-paired`, `alignment-bowtie`, `alignment-bowtie2`, `alignment-hisat2`, `alignment-subread`, `cuffmerge`, `pca`, `cuffdiff`, `differentialexpression-edger`, `cufflinks`, `cuffnorm`, `cuffquant`, `expression-aggregator`, `htseq-count`, `htseq-count-raw`, `index-fasta-nucl`, `rsem`, `upload-bam`, `upload-bam-indexed`, `upload-bam-secondary`, `upload-expression`, `upload-expression-cuffnorm`, `upload-expression-star`, `upload-genome`, `upload-gaf`, `upload-obo`, `upload-fasta-nucl`, `regtools-junctions-annotate`, `cutadapt-custom-single`, `cutadapt-custom-paired`, `bam-split`, `gff-to-gtf`, `spikein-qc`, `differentialexpression-shrna`, `feature_counts`, `salmon-index`, `salmon-quant`, `library-strandedness`, `qorts-qc`, `alignment-star`, `alignment-star-index`, `cutadapt-3prime-single`, `cutadapt-single`, `cutadapt-paired`, `differentialexpression-deseq2`, `cutadapt-corall-single`, `cutadapt-corall-paired`, `umi-tools-dedup` and `shrna-quant`.
- Use `resolwebio/common:1.3.1` Docker image in processes `amplicon-table`, `mergeexpressions`, `upload-bedpe`, `upload-bam-scseq-indexed`, `upload-diffexp`, `upload-etc`, `upload-sc-10x`, `upload-multiplexed-single`, `upload-multiplexed-paired`, `archive-samples`, `samtools-idxstats`, `seqtk-sample-single`, `seqtk-sample-paired`, `basespace-file-import`, `clustering-hierarchical-samples`, `clustering-hierarchical-genes`, `import-sra`, `import-sra-single`, `import-sra-paired`.
- Compute TPM values and map gene_ids to gene symbols in `alleyoop-collapse` process output
- Rewrite `multiqc` process to Python
- Save `lib_format_counts.json` in a separate output field in the `salmon-quant` process
- Use `resolwebio/common:1.3.1` as a base Docker image for the `resolwebio/wgbs:1.2.0` Docker image
- Support MultiQC reports in ChIP-seq workflows

Fixed

- Fix Mapping search for `source_id` / `target_id`
- Fix handling of input file names in processes: `cellranger-count`, `cutadapt-3prime-single`, `cutadapt-corall-single`, `cutadapt-corall-paired`, `salmon-quant`, `umi-tools-dedup`, `upload-sc-10x` and `upload-bam-scseq-indexed`
- Fix handling of chimeric alignments in `alignment-star`

1.5.51 25.1.0 - 2020-01-14

Added

Changed

- Extend the MultiQC report so that the Sample summary table is created for the compatible Data objects
- Bump CPU and memory requirements for the `alignment-bowtie2` process
- Move upload test files of differential expression to its own folder

Fixed

- Fix typo in `scheduling_class` variable in several Python processes
- Handle cases of improper tags passed to `read_group` argument of the `bqsr` process
- When processing differential expression files, a validation is performed for numeric columns

1.5.52 25.0.0 - 2019-12-17

Added

- Add `alleyoop-rates` process
- Add `alleyoop-utr-rates` process
- Add `alleyoop-summary` process
- Add `alleyoop-snpeval` process
- Add `alleyoop-collapse` process
- Add `slam-count` process
- Add `workflow-slamdunk-paired` workflow

Changed

- **BACKWARD INCOMPATIBLE:** Refactor `slamdunk-all-paired` process to support genome browser visualization and add additional output fields
- Append sample and genome reference information to the summary output file in the `filtering-chemut` process
- Bigwig output field in `bamclipper`, `bqsr` and `markduplicates` processes is no longer required
- Support Slamdunk/Alleyoop processes in MultiQC
- Enable sorting of files in `alignment-star` process using Samtools
- Support merging of multi-lane sequencing data into a single (pair) of FASTQ files in the `upload-fastq-single`, `upload-fastq-paired`, `files-to-fastq-single` and `files-to-fastq-paired` processes

1.5.53 24.0.0 - 2019-11-15

Added

- Add `resolwebio/slamdunk` Docker image
- Add Tabix (1.7-2) to `resolwebio/bamliquidator:1.2.0` Docker image
- Add `seqtk-rev-complement-single` and `seqtk-rev-complement-paired` process
- Add `slamdunk-all-paired` process

Changed

- **BACKWARD INCOMPATIBLE:** Require Resolwe 20.x
- Make BaseSpace file download more robust
- Bump `rose2` to 1.1.0, `bamliquidator` to 1.3.8, and use `resolwebio/base:ubuntu-18.04` Docker image as a base image in `resolwebio/bamliquidator:1.1.0` Docker image
- Use `resolwebio/bamliquidator:1.2.0` in `rose2` process
- Bump CPU, memory and Docker image (`resolwebio/rnaseq:4.9.0`) requirements in `alignment-bwa-mem`, `alignment-bwa-sw` and `alignment-bwa-aln` processes
- Use multi-threading option in Samtools commands in `alignment-bwa-mem`, `alignment-bwa-sw` and `alignment-bwa-aln` processes

1.5.54 23.1.1 - 2019-10-11

Changed

- Renamed `workflow-trim-align-quant` workflow to make the name more informative

1.5.55 23.1.0 - 2019-09-30

Added

- Add `Macaca mulatta` species choice to the `sample` descriptor schema
- Add `workflow-cutadapt-star-fc-quant-wo-depletion-single` process

Changed

- Test files improved for `workflow-wes`, `bamclipper`, `markduplicates` and `bqsr`
- Fix typo in `differentialexpression-shrna` process docstring

Fixed

- Fix transcript-to-gene_id mapping for Salmon expressions in `differentialexpression-deseq2` process. Transcript versions are now ignored when matching IDs using the transcript-to-gene_id mapping table.
- Fix `workflow-cutadapt-star-fc-quant-single` process description

1.5.56 23.0.0 - 2019-09-17

Changed

- Update order of QC reports in MultiQC configuration file. The updated configuration file is part of the `resolwebio/common:1.3.1` Docker image.
- Bump Jbrowse to version 1.16.6 in `resolwebio/rnaseq:4.9.0` Docker image
- Use JBrowse `generate-names.pl` script to index GTF/GFF3 features upon annotation file upload

- Support Salmon reports in MultiQC and expose `dirs_depth` parameter
- Expose transcript-level expression file in the `salmon-quant` process

Added

- Add `workflow-bbduk-salmon-qc-single` and `workflow-bbduk-salmon-qc-paired` workflows

Fixed

- Give process `upload-bedpe` access to network

1.5.57 22.0.0 - 2019-08-20

Changed

- **BACKWARD INCOMPATIBLE:** Require Resolwe 19.x
- **BACKWARD INCOMPATIBLE:** Unify `cutadapt-single` and `cutadapt-paired` process inputs and refactor to use Cutadapt v2.4
- Expose BetaPrior parameter in `differentialexpression-deseq2` process
- Install R from CRAN-maintained repositories in Docker images build from the `resolwebio/base:ubuntu-18.04` base image
- Prepare `resolwebio/common:1.3.0` Docker image:
  - Install R v3.6.1
  - Bump Resdk to v10.1.0
  - Install gawk package
  - Fix Docker image build issues
- Use `resolwebio/common:1.3.0` as a base image for `resolwebio/rnaseq:4.8.0`
- Update StringTie to v2.0.0 in `resolwebio/rnaseq:4.8.0`
- Support StringTie analysis results in DESeq2 tool

Added

- Add `cutadapt-3prime-single` process
- Add `workflow-cutadapt-star-fc-quant-single` process
- Add argument `skip` to `bamclipper` which enables skipping of the said process
- Add `cutadapt-corall-single` and `cutadapt-corall-paired` processes for pre-processing of reads obtained using Corall Total RNA-seq library prep kit
- Add `umi-tools-dedup` process
- Add `stringtie` process
- Add `workflow-corall-single` and `workflow-corall-paired` workflows optimized for Corall Total RNA-seq library prep kit data

Fixed

- Fix warning message in hierarchical clustering of genes. Incorrect gene names were reported in the warning message about removed genes. Computation of hierarchical clustering was correct.

1.5.58 21.0.1 - 2019-07-26

Changed

- Bump Cutadapt to v2.4 and use `resolwebio/common:1.2.0` as a base image in `resolwebio/rnaseq:4.6.0`

Added

- Add pigz package to `resolwebio/common:1.2.0` Docker image
- Add StringTie and UMI-tools to `resolwebio/rnaseq:4.7.0` Docker image

Fixed

- Fix `spikeins-qc` process to correctly handle the case where all expressions are without spikeins
- Fix an error in `macs2-callpeak` process that prevented correct reporting of build/species mismatch between inputs
- Support UCSC annotations in `feature_counts` process by assigning empty string gene_ids to the "unknown" gene

1.5.59 21.0.0 - 2019-07-16

Changed

- **BACKWARD INCOMPATIBLE:** Require Resolwe 18.x
- Bump the number of allocated CPU cores to 20 in `alignment-bwa-mem` process
- Bump memory requirements in `seqtk-sample-single` and `seqtk-sample-paired` processes
- Bump Salmon to v0.14.0 in `resolwebio/rnaseq:4.5.0` Docker image
- Expose additional inputs in `salmon-index` process
- Use `resolwebio/rnaseq:4.5.0` Docker image in processes that call Salmon tool (`library-strandedness`, `feature_counts` and `qorts-qc`)
- Implement dropdown menu for `upload-bedpe` process
- Add validation stringency parameter to `bqsr` process and propagate it to the `workflow-wes` as well
- Add LENIENT value to validation stringency parameter of the `markduplicates` process
- Improve performance of RPKUM normalization in `featureCounts` process

Added

- Add `salmon-quant` process

Fixed

- Fix genome upload process to correctly handle filenames with dots
- Fix merging of expressions in `archive-samples` process. Previously some genes were missing in the merged expression files. The genes that were present had expression values correctly assigned. The process was optimized for performance and now supports parallelization.

1.5.60 20.0.0 2019-06-19

Changed

- **BACKWARD INCOMPATIBLE:** Require Resolwe 17.x
- **BACKWARD INCOMPATIBLE:** Use Elasticsearch version 6.x
- **BACKWARD INCOMPATIBLE:** Bump Django requirement to version 2.2
- **BACKWARD INCOMPATIBLE:** Remove obsolete RNA-seq workflows `workflow-bbduk-star-featurecounts-single`, `workflow-bbduk-star-featurecounts-paired`, `workflow-cutadapt-star-featurecounts-single` and `workflow-cutadapt-star-featurecounts-paired`
- **BACKWARD INCOMPATIBLE:** Remove obsolete descriptor schemas: `rna-seq-bbduk-star-featurecounts`, `quantseq`, `rna-seq-cutadapt-star-featurecounts` and `kapa-rna-seq-bbduk-star-featurecounts`
- **BACKWARD INCOMPATIBLE:** In `upload-fasta-nucl` process, store compressed and uncompressed FASTA files in `fastagz` and `fasta` ouput fields, respectively
- Allow setting the Java memory usage flags for the QoRTs tool in `resolwebio/common:1.1.3` Docker image
- Use `resolwebio/common:1.1.3` Docker image as a base image for `resolwebio/rnaseq:4.4.2`
- Bump GATK4 version to 4.1.2.0 in `resolwebio/dnaseq:4.2.0`
- Use MultiQC configuration file and prepend directory name to sample names by default in `multiqc` process
- Bump `resolwebio/common` to 1.1.3 in `resolwebio/dnaseq:4.2.0`
- Process `vc-gatk4-hc` now also accepts BED files through parameter `intervals_bed`

Added

- Support Python 3.7
- Add Tabix (1.7-2) to `resolwebio/wgbs` docker image
- Add JBrowse index output to `hmr` process
- Add `bamclipper` tool and `parallel` package to `resolwebio/dnaseq:4.2.0` image
- Support `hg19_mm10` hybrid genome in `bam-split` process
- Support mappability-based normalization (RPKUM) in featureCounts
- Add BEDPE upload process

- Add `bamclipper` process
- Add `markduplicates` process
- Add `bqsr` (BaseQualityScoreRecalibrator) process
- Add whole exome sequencing (WES) pipeline

Fixed

- Fix building problems of `resolwebio/dnaseq` docker
- Fix handling of no-adapters input in workflows `workflow-bbduk-star-featurecounts-qc-single` and `workflow-bbduk-star-featurecounts-qc-paired`

1.5.61 19.0.1 2019-05-13

Fixed

- Use `resolwebio/rnaseq:4.4.2` Docker image that enforces the memory limit and bump memory requirements for `qorts-qc` process
- Bump memory requirements for `multiqc` process

1.5.62 19.0.0 2019-05-07

Changed

- Use Genialis fork of MultiQC 1.8.0b in `resolwebio/common:1.1.2`
- Support Samtools idxstats and QoRTs QC reports in `multiqc` process
- Support `samtools-idxstats` QC step in workflows:
  - `workflow-bbduk-star-featurecounts-qc-single`
  - `workflow-bbduk-star-featurecounts-qc-paired`
  - `workflow-bbduk-star-fc-quant-single`
  - `workflow-bbduk-star-fc-quant-paired`
- Simplify `cellranger-count` outputs folder structure
- Bump STAR aligner to version 2.7.0f in `resolwebio/rnaseq:4.4.1` Docker image
- Use `resolwebio/rnaseq:4.4.1` in `alignment-star` and `alignment-star-index` processes
- Save filtered count-matrix output file produced by DESeq2 differential expression process

Added

- Add `samtools-idxstats` process
- Improve `cellranger-count` and `cellranger-mkref` logging
- Add FastQC report to `upload-sc-10x` process

Fixed

- Fix `archive-samples` to work with `data:chipseq:callpeak:macs2` data objects when downloading only peaks without QC reports
- Fix parsing gene set files with empty lines to avoid saving gene sets with empty string elements

1.5.63 18.0.0 2019-04-16

Changed

- **BACKWARD INCOMPATIBLE:** Require Resolwe 16.x
- **BACKWARD INCOMPATIBLE:** Rename and improve descriptions of processes specific to CATS RNA-seq kits. Remove related `cutadapt-star-htseq` descriptor schema.
- **BACKWARD INCOMPATIBLE:** Remove `workflow-accel-gatk4` pipeline. Remove `amplicon-panel`, `amplicon-panel-advanced` and `amplicon-master-file` descriptor schemas.
- **BACKWARD INCOMPATIBLE:** Remove obsolete processes and descriptor schemas: `rna-seq-quantseq`, `bcm-workflow-rnaseq`, `bcm-workflow-chipseq`, `bcm-workflow-wgbs`, `dicty-align-reads`, `dicty-etc`, `affy` and `workflow-chip-seq`
- Expose additional parameters of `bowtie2` process
- Support strandedness auto detection in `qorts-qc` process

Added

- Add shRNAde (v1.0) R package to the `resolwebio/rnaseq:4.4.0` Docker image
- Add `resolwebio/scseq` Docker image
- Add shRNA differential expression process. This is a two-step process which trims, aligns and quantifies short hairpin RNA species. These are then used in a differential expression.
- Add `sc-seq` processes:
  - `cellranger-mkref`
  - `cellranger-count`
  - `upload-sc-10x`
  - `upload-bam-scseq-indexed`

Fixed

- Bump memory requirements in `seqtk-sample-single` and `seqtk-sample-paired` processes
- Fix `cellranger-count` html report
- Mark spliced-alignments with XS flags in `workflow-rnaseq-cuffquant`
- Fix whitespace handling in `cuffnorm` process

1.5.64 17.0.0 2019-03-19

Added

- Add `qorts-qc` (Quality of RNA-seq Tool-Set QC) process
- Add `workflow-bbduk-star-fc-quant-single` and `workflow-bbduk-star-fc-quant-paired` processes
- Add independent gene filtering and gene filtering based on Cook's distance in `DESeq2` differential expression process

Changed

- **BACKWARD INCOMPATIBLE**: Move gene filtering by expression count input to `filter.min_count_sum` in `DESeq2` differential expression process
- **BACKWARD INCOMPATIBLE:** Require Resolwe 15.x
- Update `resolwebio/common:1.1.0` Docker image:
  - add QoRTs (1.3.0) package
  - bump MultiQC to 1.7.0
  - bump Subread package to 1.6.3
- Expose `maxns` input parameter in `bbduk-single` and `bbduk-paired` processes. Make this parameter available in workflows `workflow-bbduk-star-featurecounts-qc-single`, `workflow-bbduk-star-featurecounts-qc-paired`, `workflow-bbduk-star-featurecounts-single` and `workflow-bbduk-star-featurecounts-paired`.
- Save CPM-normalized expressions in `feature_counts` process. Control the default expression normalization type (`exp_type`) using the `normalization_type` input.
- Bump MultiQC to version 1.7.0 in `multiqc` process
- Use `resolwebio/rnaseq:4.3.0` with Subread/featureCounts version 1.6.3 in `feature_counts` process

1.5.65 16.3.0 2019-02-19

Changed

- Bump STAR aligner version to 2.7.0c in `resolwebio/rnaseq:4.2.2`
- Processes `alignment-star` and `alignment-star-index` now use Docker image `resolwebio/rnaseq:4.2.2` which contains STAR version `2.7.0c`
- Persistence of `basespace-file-import` process changed from `RAW` to `TEMP`

Added

- Make `prepare-geo-chipseq` work with both `data:chipseq:callpeak:macs2` and `data:chipseq:callpeak:macs14` as inputs

Fixed

- Report correct total mapped reads and mapped reads percentage in prepeak QC report for `data:alignment:bam:bowtie2` inputs in `macs2-callpeak` process

1.5.66 16.2.0 2019-01-28

Changed

- Enable multithreading mode in `alignment-bwa-aln` and `alignment-bwa-sw`
- Lineary lower the timeout for BigWig calculation when running on multiple cores

Fixed

- Remove `pip --process-dependency-links` argument in testenv settings
- Fix walt getting killed when `sort` runs out of memory. The `sort` command buffer size was limited to the process memory limit.

1.5.67 16.1.0 2019-01-17

Changed

Added

- Add the `FASTQ` file validator script to the `upload-fastq-single`, `upload-fastq-paired`, `files-to-fastq-single` and `files-to-fastq-paired` processes
- Add `spikein-qc` process
- Add to `resolwebio/rnaseq:4.1.0` Docker image:
  - `dnaio` Python library
- Add to `resolwebio/rnaseq:4.2.0` Docker image:
  - ERCC table
  - common Genialis fonts and css file
  - spike-in QC report template
- Set `MPLBACKEND` environment variable to `Agg` in `resolwebio/common:1.0.1` Docker image

Fixed

- Fix the format of the output FASTQ file in the demultiplex.py script
- Fix NSC and RSC QC metric calculation for ATAC-seq and paired-end ChIP-seq samples in macs2-callpeak and qc-prepeak processes

1.5.68 16.0.0 2018-12-19

Changed

- **BACKWARD INCOMPATIBLE:** Require Resolwe 14.x
- **BACKWARD INCOMPATIBLE:** Remove obsolete processes findsimilar
- **BACKWARD INCOMPATIBLE:** Include ENCODE-proposed QC analysis metrics methodology in the macs2-callpeak process. Simplified MACS2 analysis inputs now allow the use of sample relations (treatment/background) concept to trigger multiple MACS2 jobs automatically using the macs2-batch or macs2-rose2-batch processes.
- **BACKWARD INCOMPATIBLE:** Update workflow-atac-seq inputs to match the updated macs2-callpeak process
- Use resolwebio/rnaseq:4.0.0 Docker image in alignment-star-index, bbduk-single, bbduk-paired, cuffdiff, cufflinks, cuffmerge, cuffnorm, cuffquant, cutadapt-custom-single, cutadapt-custom-paired, cutadapt-single, cutadapt-paired, differentialexpression-deseq2, differentialexpression-edger, expression-aggregator, feature_counts, goenrichment, htseq-count, htseq-count-raw, index-fasta-nucl, library-strandedness, pca, regtools-junctions-annotate, rsem, salmon-index, trimmomatic-single, trimmomatic-paired, upload-expression, upload-expression-cuffnorm, upload-expression-star, upload-fasta-nucl, upload-fastq-single, upload-fastq-paired, files-to-fastq-single, files-to-fastq-paired, upload-gaf, upload-genome, upload-gff3, upload-gtf and upload-obo
- Order statistical groups in expression aggregator output by sample descriptor field value
- Use resolwebio/biox:1.0.0 Docker image in etc-bcm, expression-dicty and mappability-bcm processes
- Use resolwebio/common:1.0.0 Docker image in amplicon-table, mergeexpressions, upload-diffexp, upload-etc, upload-multiplexed-single and upload-multiplexed-paired processes
- Use resolwebio/base:ubuntu-18.04 Docker image in create-geneset, create-geneset-venn, mergeetc, prepare-geo-chipseq, prepare-geo-rnaseq, upload-cxb, upload-geneset, upload-header-sam, upload-mappability, upload-snpeff and upload-picard-pcrmetrics processes
- Update GATK4 to version 4.0.11.0 in resolwebio/dnaseq:4.1.0 Docker image. Install and use JDK v8 by default to ensure compatibility with GATK4 package.
- Use resolwebio/dnaseq:4.1.0 Docker image in align-bwa-trim, coveragebed, filtering-chemut, lofreq, picard-pcrmetrics, upload-master-file, upload-variants-vcf and vc-gatk4-hc processes
- Expose reads quality filtering (q) parameter, reorganize inputs and rename the stats output file in alignment-bwa-aln process
- Use resolwebio/chipseq:4.0.0 Docker image in chipseq-genescore, chipseq-peakscore, macs14, upload-bed and qc-prepeak processes
- Use resolwebio/bamliquidator:1.0.0 Docker image in bamliquidator and bamplot processes

Added

- Add biosample source field to `sample` descriptor schema
- Add `background_pairs` Jinja expressions filter that accepts a list of data objects and orders them in a list of pairs (case, background) based on the background relation between corresponding samples
- Add `chipseq-bwa` descriptor schema. This schema specifies the default inputs for BWA ALN aligner process as defined in ENCODE ChIP-Seq experiments.
- Add support for MACS2 result files to MultiQC process
- Add `macs2-batch`, `macs2-rose2-batch` and `workflow-macs-rose` processes
- Add feature symbols to expressions in `archive-samples` process

Fixed

- Make ChIP-seq fields in `sample` descriptor schema visible when ChIPmentation assay type is selected
- Fix handling of whitespace in input BAM file name in script `detect_strandedness.sh`
- Set available memory for STAR aligner to 36GB. Limit the available memory for STAR aligner `--limitBAMsortRAM` parameter to 90% of the Docker requirements setting
- Set `bbduk-single` and `bbduk-paired` memory requirements to 8GB
- Fix wrong file path in `archive-samples` process

1.5.69 15.0.0 2018-11-20

Changed

- **BACKWARD INCOMPATIBLE:** Remove obsolete processes: `bsmap`, `mcall`, `coverage-garvan`, `igv`, `jbrowse-bed`, `jbrowse-gff3`, `jbrowse-gtf`, `jbrowse-bam-coverage`, `jbrowse-bam-coverage-normalized`, `jbrowse-refseq`, `fastq-mcf-single`, `fastq-mcf-paired`, `hsqutils-trim`, `prinseq-lite-single`, `prinseq-lite-paired`, `sortmerna-single`, `sortmerna-paired`, `bam-coverage`, `hsqutils-dedup`, `vc-samtools`, `workflow-heat-seq` and `alignment-tophat2`
- **BACKWARD INCOMPATIBLE:** Remove `jbrowse-bam-coverage` process step from the `workflow-accel` workflow. The bigwig coverage track is computed in `align-bwa-trim` process instead.
- **BACKWARD INCOMPATIBLE:** Remove `resolwebio/utils` Docker image. This image is replaced by the `resolwebio/common` image.
- **BACKWARD INCOMPATIBLE:** Use `resolwebio/common` Docker image as a base image for the `resolwebio/biox`, `resolwebio/chipseq`, `resolwebio/dnaseq` and `resolwebio/rnaseq` images
- **BACKWARD INCOMPATIBLE:** Remove `resolwebio/legacy` Docker image.
- Use sample name as the name of the data object in:
  - `alignment-bwa-aln`
  - `alignment-bowtie2`
  - `qc-prepeak`
  - `macs2-callpeak`
- Attach `macs2-callpeak`, `macs14` and `rose2` process data to the case/treatment sample

- Use `resolwebio/dnaseq:4.0.0` docker image in `align-bwa-trim` process
- Use `resolwebio/rnaseq:4.0.0` docker image in aligners: `alignment-bowtie`, `alignment-bowtie2`, `alignment-bwa-mem`, `alignment-bwa-sw`, `alignment-bwa-aln`, `alignment-hisat2`, `alignment-star` and `alignment-subread`.
- Set memory limits in `upload-genome`, `trimmomatic-single` and `trimmomatic-paired` processes
- Improve error messages in differential expression process `DESeq2`

Added

- Add `makedb (WALT 1.01)` - callable as `makedb-walt`, tool to create genome index for WALT aligner, to `resolwebio/rnaseq` docker image
- Add `resolwebio/wgbs` docker image including the following tools:
  - `MethPipe (3.4.3)`
  - `WALT (1.01)`
  - `wigToBigWig (kent-v365)`
- Add `resolwebio/common` Docker image. This image includes common bioinformatics utilities and can serve as a base image for other, specialized `resolwebio` Docker images: `resolwebio/biox`, `resolwebio/chipseq`, `resolwebio/dnaseq` and `resolwebio/rnaseq`.
- Add `shift` (user-defined cross-correlation peak strandshift) input to `qc-prepeak` process
- Add ATAC-seq workflow
- Compute index for `WALT` aligner on genome upload and support uploading the index together with the genome
- Add `Whole genome bisulfite sequencing` workflow and related WGBS processes:
  - `WALT`
  - `methcounts`
  - `HMR`
- Add bedClip to *resolwebio/chipseq:3.1.0* docker image
- Add `resolwebio/biox` Docker image. This image is based on the `resolwebio/common` image and includes Biox Python library for Dictyostelium RNA-Seq analysis support.
- Add `resolwebio/snpeff` Docker image. The image includes SnpEff (4.3K) tool.
- Add spike-in names, rRNA and globin RNA cromosome names in `resolwebio/common` image
- Add UCSC bedGraphtoBigWig tool for calculating BigWig in `bamtobigwig.sh` script. In `align-bwa-trim` processor set this option (that BigWig is calculated by UCSC tool instead of deepTools), because it is much faster for amplicon files. In other processors update the input parameters for `bamtobigwig.sh`: `alignment-bowtie`, `alignment-bowtie2`, `alignment-bwa-mem`, `alignment-bwa-sw`, `alignment-bwa-aln`, `alignment-hisat2`, `alignment-star` `alignment-subread`, `upload-bam`, `upload-bam-indexed` and `upload-bam-secondary`.
- In `bamtobigwig.sh` don't create BigWig when bam file was aligned on globin RNA or rRNA (this are QC steps and BigWig is not needed)

Fixed

- **BACKWARD INCOMPATIBLE:** Use user-specificed distance metric in hierarchical clustering
- Handle integer expression values in hierarchical clustering
- Fix Amplicon table gene hyperlinks for cases where multiple genes are associated with detected variant
- Handle empty gene name in expression files in PCA
- Fix PBC QC reporting in `qc-prepeak` process for a case where there are no duplicates in the input bam
- Fix `macs2-callpeak` process so that user defined fragment lenth has priority over the `qc-prepeak` estimated fragment length when shifting reads for post-peakcall QC
- Fix `macs2-callpeak` to prevent the extension of intervals beyond chromosome boundaries in MACS2 bedgraph outputs
- Fix warning message in hierarchical clustering of genes to display gene names

1.5.70 14.0.2 2018-10-23

Fixed

- Fix `htseq-count-raw` process to correctly map features with associated feature symbols.

1.5.71 14.0.1 2018-10-23

Fixed

- Handle missing gene expression in hierarchical clustering of genes. If one or more genes requested in gene filter are missing in selected expression files a warning is issued and hierarchical clustering of genes is computed with the rest of the genes instead of failing.
- Fix PCA computation for single sample case

1.5.72 14.0.0 2018-10-09

Changed

- **BACKWARD INCOMPATIBLE:** Require Resolwe 13.x
- **BACKWARD INCOMPATIBLE:** Remove `gsize` input from `macs2-callpeak` process and automate genome size selection
- **BACKWARD INCOMPATIBLE:** Set a new default `sample` and `reads` descriptor schema. Change slug from `sample2` to `sample`, modify group names, add `cell_type` field to the new `sample` descriptor schema, and remove the original `sample`, `sample-detailed`, and `reads-detailed` descriptor schemas.
- **BACKWARD INCOMPATIBLE:** Unify types of `macs14` and `macs2-callpeak` processes and make `rose2` work with both
- **BACKWARD INCOMPATIBLE:** Remove `replicates` input in `cuffnorm` process. Use sample relation information instead.
- Use `resolwebio/chipseq:3.0.0` docker image in the following processes:
  - `macs14`

  - macs2-callpeak
  - rose2

- Downgrade primerclip to old version (v171018) in resolwebio/dnaseq:3.3.0 docker image and move it to google drive.
- Move bam-split process to resolwebio/rnaseq:3.7.1 docker image
- Count unique and multimmaping reads in regtools-junctions-annotate process

Added

- Add qc-prepeak process that reports ENCODE3 accepted ChIP-seq and ATAC-seq QC metrics
- Add QC report to macs2-callpeak process
- Add combining ChIP-seq QC reports in archive-samples process
- Add detection of globin-derived reads as an additional QC step in the workflow-bbduk-star-featurecounts-qc-single and workflow-bbduk-star-featurecounts-qc-paired processes.
- Add mappings from ENSEMBL or NCBI to UCSC chromosome names and deepTools (v3.1.0) to resolwebio/dnaseq:3.3.0 docker image
- Add BigWig output field to following processors:
  - align-bwa-trim
  - upload-bam
  - upload-bam-indexed
  - upload-bam-secondary
- Add replicate_groups Jinja expressions filter that accepts a list of data objects and returns a list of labels determining replicate groups.
- Add 'Novel splice junctions in BED format' output to regtools-junctions-annotate process, so that user can visualize only novel splice juntions in genome browsers.

Fixed

- Fix handling of numerical feature_ids (NCBI source) in create_expression_set.py script
- Make chipseq-peakscore work with gzipped narrowPeak input from macs2-callpeak
- Use uncompressed FASTQ files as input to STAR aligner to prevent issues on (network) filesystems without FIFO support

1.5.73 13.0.0 2018-09-18

Changed

- **BACKWARD INCOMPATIBLE:** Require Resolwe 12.x
- **BACKWARD INCOMPATIBLE:** Remove obsolete processes: `assembler-abyss`, `cutadapt-amplicon`, `feature_location`, `microarray-affy-qc`, `reads-merge`, `reference_compatibility`, `transmart-expressions`, `upload-hmmer-db`, `upload-mappability-bigwig`, `upload-microarray-affy`.
- **BACKWARD INCOMPATIBLE:** Remove obsolete descriptor schema: `transmart`.
- **BACKWARD INCOMPATIBLE:** Remove tools which are not used by any process: `clustering_leaf_ordering.py`, `go_genesets.py`, `VCF_ad_extract.py`, `volcanoplot.py`, `xgff.py`, `xgtf2gff.py`.
- **BACKWARD INCOMPATIBLE:** Management command for inserting features and mappings requires PostgreSQL version 9.5 or newer
- Update the meta data like name, description, category, etc. of most of the processes
- Speed-up management command for inserting mappings
- Change location of cufflinks to Google Drive for resolwebio/rnaseq Docker build
- Calculate alignment statistics for the uploaded alignment (.bam) file in the `upload-bam`, `upload-bam-indexed` and `upload-bam-secondary` processes.
- Annotation (GTF/GFF3) file input is now optional for the creation of the STAR genome index files. Annotation file can be used at the alignment stage to supplement the genome indices with the set of known features.
- Trigger process warning instead of process error in the case when `bamtobigwig.sh` scripts detects an empty .bam file.
- Set the default reads length filtering parameter to 30 bp in the `rna-seq-bbduk-star-featurecounts` and `kapa-rna-seq-bbduk-star-featurecounts` experiment descriptor schema. Expand the kit selection choice options in the latter descriptor schema.

Added

- Add `MultiQC (1.6.0)` and `Seqtk (1.2-r94)` to the `resolwebio/utils:1.5.0` Docker image
- Add `sample2` descriptor schema which is the successor of the original `sample` and `reads` descriptor schemas
- Add bedToBigBed and Tabix to resolwebio/rnaseq:3.7.0 docker image
- Add `HS Panel` choice option to the `amplicon-master-file` descriptor schema
- Add MultiQC process
- Add process for the Seqtk tool `sample` sub-command. This process allows sub-sampling of `.fastq` files using either a fixed number of reads or the ratio of the input file.
- Add MultiQC analysis step to the `workflow-bbduk-star-featurecounts-single` and `workflow-bbduk-star-featurecounts-single` processes.
- Add `workflow-bbduk-star-featurecounts-qc-single` and `workflow-bbduk-star-featurecounts-qc-paired` processes which support MultiQC analysis, input reads down-sampling (using Seqtk) and rRNA sequence detection using STAR aligner.
- Add to `resolwebio/chipseq` Docker image:

  - bedtools (2.25.0-1)
  - gawk (1:4.1.3+dfsg-0.1)
  - picard-tools (1.113-2)
  - run_spp.R (1.2) (as spp)
  - SPP (1.14)
- Add `regtools-junctions-annotate` process that annotates novel splice junctions.
- Add `background` relation type to fixtures

Fixed

- Track `source` information in the `upload-fasta-nucl` process.
- When STAR aligner produces an empty alignment file, re-sort the alignment file to allow successful indexing of the output `.bam` file.
- Create a symbolic link to the alignment file in the `feature_counts` process, so that relative path is used in the quantification results. This prevent the FeatureCounts output to be listed as a separate sample in the MultiQC reports.
- Fix handling of expression objects in `archive-samples` process

1.5.74 12.0.0 - 2018-08-13

Changed

- **BACKWARD INCOMPATIBLE:** Require Resolwe 11.x
- **BACKWARD INCOMPATIBLE:** Use read count instead of sampling rate in strandedness detection
- **BACKWARD INCOMPATIBLE:** Remove `genome` input from `rose2` process and automate its selection
- **BACKWARD INCOMPATIBLE:** Refactor `cutadapt-paired` process
- **BACKWARD INCOMPATIBLE:** Improve leaf ordering performance in gene and sample hierarchical clustering. We now use exact leaf ordering which has been recently added to `scipy` instead of an approximate in-house solution based on nearest neighbor algorithm. Add informative warning and error messages to simplify troubleshooting with degenerate datasets.
- Remove `igvtools` from `resolwebio/utils` Docker image
- Improve helper text and labels in processes used for sequencing data upload
- Allow using custom adapter sequences in the `workflow-bbduk-star-featurecounts-single` and `workflow-bbduk-star-featurecounts-paired` processes
- Change chromosome names from ENSEMBL / NCBI to UCSC (example: “1” to “chr1”) in BigWig files. The purpose of this is to enable viewing BigWig files in UCSC genome browsers for files aligned with ENSEBML or NCBI genome. This change is done by adding script bigwig_chroms_to_ucsc.py to bamtobigwig.sh script.
- Reduce RAM requirement in SRA import processes

Added

- Add two-pass mode to `alignment-star` process
- Add `regtools (0.5.0)` to `resolwebio/rnaseq` Docker image
- Add KAPA experiment descriptor schema
- Add `resdk` Python 3 package to `resolwebio/utils` Docker image
- Add to `cutadapt-single` process an option to discard reads having more 'N' bases than specified.
- Add workflows for single-end `workflow-cutadapt-star-featurecounts-single` and paired-end reads `workflow-cutadapt-star-featurecounts-paired`. Both workflows consist of preprocessing with Cutadapt, alignment with STAR two pass mode and quantification with featureCounts.
- Add descriptor schema `rna-seq-cutadapt-star-featurecounts`

Fixed

- **BACKWARD INCOMPATIBLE:** Fix the `stitch` parameter handling in `rose2`
- fix `upload-gtf` to create JBrowse track only if GTF file is ok
- Pin `sra-toolkit` version to 2.9.0 in `resolwebio/utils` Docker image.
- Fix and improve `rose2` error messages
- Fail gracefully if bam file is empty when producing bigwig files
- Fail gracefully if there are no matches when mapping chromosome names

1.5.75 11.0.0 - 2018-07-17

Changed

- **BACKWARD INCOMPATIBLE:** Remove management command module
- **BACKWARD INCOMPATIBLE:** Remove filtering of genes with low expression in PCA analysis
- **BACKWARD INCOMPATIBLE:** Remove obsolete RNA-seq DSS process
- Expand error messages in `rose2` process
- Check for errors during download of FASTQ files and use `resolwebio/utils:1.3.0` Docker image in import SRA process
- Increase Feature's full name's max length to 350 to support a long full name of "Complement C3 Complement C3 beta chain C3-beta-c Complement C3 alpha chain C3a anaphylatoxin Acylation stimulating protein Complement C3b alpha' chain Complement C3c alpha' chain fragment 1 Complement C3dg fragment Complement C3g fragment Complement C3d fragment Complement C3f fragment Complement C3c alpha' chain fragment 2" in Ensembl

Added

- Add *exp_set* and *exp_set_json* output fields to expression processes:
  - `feature_counts`
  - `htseq-count`
  - `htseq-count-raw`
  - `rsem`
  - `upload-expression`
  - `upload-expression-cuffnorm`
  - `upload-expression-star`
- Add 'Masking BED file' input to `rose2` process which allows masking reagions from the analysis
- Add `filtering.outFilterMismatchNoverReadLmax` input to `alignment-star` process
- Add mappings from ENSEMBL or NCBI to UCSC chromosome names to `resolwebio/rnaseq:3.5.0` docker image

Fixed

- Fix peaks BigBed output in `macs14` process
- Remove duplicated forward of `alignIntronMax` input field in BBDuk - STAR - featureCounts workflow
- Make `cuffnorm` process attach correct expression data objects to samples
- Fix `upload-gtf` in a way that GTF can be shown in JBrowse. Because JBrowse works only with GFF files, input GTF is converted to GFF from which JBrowse track is created.

1.5.76 10.0.1 - 2018-07-06

Fixed

- Fix `bamtobigwig.sh` to timeout the `bamCoverage` calculation after defined time

1.5.77 10.0.0 - 2018-06-19

Added

- Add to `resolwebio/chipseq` Docker image:
  - `Bedops (v2.4.32)`
  - `Tabix (v1.8)`
  - `python3-pandas`
  - `bedGraphToBigWig (kent-v365)`
  - `bedToBigBed (kent-v365)`
- Add to `resolwebio/rnaseq:3.2.0` Docker image:
  - `genometools (1.5.9)`

  - igvtools (v2.3.98)
  - jbrowse (v1.12.0)
  - Bowtie (v1.2.2)
  - Bowtie2 (v2.3.4.1)
  - BWA (0.7.17-r1188)
  - TopHat (v2.1.1)
  - Picard Tools (v2.18.5)
  - bedGraphToBigWig (kent-v365)
- Add Debian package `file` to `resolwebio/rnaseq:3.3.0` Docker image
- Support filtering by type on feature API endpoint
- Add BigWig output field to following processes:
  - alignment-bowtie
  - alignment-bowtie2
  - alignment-tophat2
  - alignment-bwa-mem
  - alignment-bwa-sw
  - alignment-bwa-aln
  - alignment-hisat2
  - alignment-star
- Add Jbrowse track output field to `upload-genome` processor.
- Use `reslowebio/rnaseq` Docker image and add Jbrowse track and IGV sorting and indexing to following processes:
  - upload-gff3
  - upload-gtf
  - gff-to-gtf
- Add Tabix index for Jbrowse to `upload-bed` processor and use `reslowebio/rnaseq` Docker image
- Add BigWig, BigBed and JBrowse track outputs to `macs14` process
- Add Species and Build outputs to `rose2` process
- Add Species, Build, BigWig, BigBed and JBrowse track outputs to `macs2` process
- Add `scipy` (v1.1.0) Python 3 package to `resolwebio/utils` Docker image

Changed

- **BACKWARD INCOMPATIBLE:** Drop support for Python 3.4 and 3.5
- **BACKWARD INCOMPATIBLE:** Require Resolwe 10.x
- **BACKWARD INCOMPATIBLE:** Upgrade to Django Channels 2
- **BACKWARD INCOMPATIBLE:** Count fragments (or templates) instead of reads by default in `featureCounts` process and `BBDuk - STAR - featureCounts` pipeline. The change applies only to paired-end data.
- **BACKWARD INCOMPATIBLE:** Use `resolwebio/rnaseq:3.2.0` Docker image in the following processes that output reads:
  - `upload-fastq-single`
  - `upload-fastq-paired`
  - `files-to-fastq-single`
  - `files-to-fastq-paired`
  - `reads-merge`
  - `bbduk-single`
  - `bbduk-paired`
  - `cutadapt-single`
  - `cutadapt-paired`
  - `cutadapt-custom-single`
  - `cutadapt-custom-paired`
  - `trimmomatic-single`
  - `trimmomatic-paired`.

  This change unifies the version of `FastQC` tool (0.11.7) used for quality control of reads in the aforementioned processes. The new Docker image comes with an updated version of Cutadapt (1.16) which affects the following processes:
  - `cutadapt-single`
  - `cutadapt-paired`
  - `cutadapt-custom-single`
  - `cutadapt-custom-paired`.

  The new Docker image includes also an updated version of Trimmomatic (0.36) used in the following processes:
  - `upload-fastq-single`
  - `upload-fastq-paired`
  - `files-to-fastq-single`
  - `files-to-fastq-paired`
  - `trimmomatic-single`
  - `trimmomatic-paired`.

- **BACKWARD INCOMPATIBLE:** Change Docker image in `alignment-subread` from `resolwebio/legacy:1.0.0` with Subread (v1.5.1) to `resolwebio/rnaseq:3.2.0` with Subread (v1.6.0). `--multiMapping` option was added instead of `--unique_reads`. By default aligner report uniquely mapped reads only.
- Update `wigToBigWig` to kent-v365 version in `resolwebio/chipseq` Docker image
- Change paths in HTML amplicon report template in `resolwebio/dnaseq` Docker image
- Move assay type input in BBDuk - STAR - featureCounts pipeline descriptor schema to advanced options
- Use `resolwebio/rnaseq:3.2.0` Docker image with updated versions of tools instead of `resolwebio/legacy:1.0.0` Docker image in following processes:
  - `alignment-bowtie` with Bowtie (v1.2.2) instead of Bowtie (v1.1.2)
  - `alignment-bowtie2` with Bowtie2 (v2.3.4.1) instead of Bowtie2 (v2.2.6)
  - `alignment-tophat2` with TopHat (v2.1.1) instead of TopHat (v2.1.0)
  - `alignment-bwa-mem`, `alignment-bwa-sw` and ``alignment-bwa-aln` with BWA (v0.7.17-r1188) instead of BWA (v0.7.12-r1039)
  - `alignment-hisat2` with HISAT2 (v2.1.0) instead of HISAT2 (v2.0.3-beta)
  - `upload-genome`
- Use `resolwebio/base:ubuntu-18.04` Docker image as a base image in `resolwebio/utils` Docker image
- Update Python 3 packages in `resolwebio/utils` Docker image:
  - `numpy` (v1.14.4)
  - `pandas` (v0.23.0)
- Replace `bedgraphtobigwig` with `deepTools` in `resolwebio/rnaseq` Docker image, due to faster performance
- Use `resolwebio/rnaseq:3.3.0` Docker image in `alignment-star-index` with STAR (v2.5.4b)

Fixed

- Make management commands use a private random generator instance
- Fix output `covplot_html` of `coveragebed` process
- Fix process `archive-samples` and `amplicon-archive-multi-report` to correctly handle nested file paths
- Change `rose2` and `chipseq-peakscore` to work with `.bed` or `.bed.gz` input files
- Fix the `expression-aggregator` process so that it tracks the `species` of the input expression data
- Fix `bamtobigwig.sh` to use `deepTools` instead of `bedtools` with `bedgraphToBigWig` due to better time performance

1.5.78 9.0.0 - 2018-05-15

Changed

- **BACKWARD INCOMPATIBLE:** Simplify the `amplicon-report` process inputs by using Latex report template from the `resolwebio/latex` Docker image assets
- **BACKWARD INCOMPATIBLE:** Simplify the `coveragebed` process inputs by using Bokeh assets from the `resolwebio/dnaseq` Docker image
- **BACKWARD INCOMPATIBLE:** Require Resolwe 9.x
- Update `wigToBigWig` tool in `resolwebio/chipseq` Docker image
- Use `resolwebio/rnaseq:3.1.0` Docker image in the following processes:
  - `cufflinks`
  - `cuffnorm`
  - `cuffquant`
- Remove `differentialexpression-limma` process
- Use `resolwebio/rnaseq:3.1.0` docker image and expand error messages in:
  - `cuffdiff`
  - `differentialexpression-deseq2`
  - `differentialexpression-edger`
- Update `workflow-bbduk-star-htseq`
- Update `quantseq` descriptor schema
- Assert species and build in `htseq-count-normalized` process
- Set amplicon report template in `resolwebio/latex` Docker image to landscape mode

Added

- Support Python 3.6
- Add `template_amplicon_report.tex` to `resolwebio/latex` Docker image assets
- Add SnpEff tool and bokeh assets to `resolwebio/dnaseq` Docker image
- Add automated library strand detection to `feature_counts` quantification process
- Add FastQC option `nogroup` to `bbduk-single` and `bbduk-paired` processes
- Add CPM normalization to `htseq-count-raw` process
- Add `workflow-bbduk-star-htseq-paired`
- Add legend to amplicon report template in `resolwebio/latex` Docker image

Fixed

- Fix manual installation of packages in Docker images to handle dots and spaces in file names correctly
- Fix COSMIC url template in `amplicon-table` process
- Fix Create IGV session in Archive samples process
- Fix `source` tracking in `cufflinks` and `cuffquant` processes
- Fix amplicon master file validation script. Check and report error if duplicated amplicon names are included. Validation will now pass also for primer sequences in lowercase.
- Fix allele frequency (AF) calculation in `snpeff` process
- Fix bug in script for calculating FPKM. Because genes of raw counts from `featureCounts` were not lexicographically sorted, division of normalized counts was done with values from other, incorrect, genes. Results from `featureCounts`, but not `HTSeq-count` process, were affected.

1.5.79 8.1.0 - 2018-04-13

Changed

- Use the latest versions of the following Python packages in `resolwebio/rnaseq` docker image: Cutadapt 1.16, Apache Arrow 0.9.0, pysam 0.14.1, requests 2.18.4, appdirs 1.4.3, wrapt 1.10.11, PyYAML 3.12
- Bump tools version in `resolwebio/rnaseq` docker image:
  - Salmon to 0.9.1
  - FastQC to 0.11.7
- Generalize the no-extraction-needed use-case in `resolwebio/base` Docker image `download_and_verify` script

Added

- Add the following Python packages to `resolwebio/rnaseq` docker image: six 1.11.0, chardet 3.0.4, urllib3 1.22, idna 2.6, and certifi 2018.1.18
- Add `edgeR` R library to `resolwebio/rnaseq` docker image
- Add Bedtools to `resolwebio/rnaseq` docker image

Fixed

- Handle filenames with spaces in the following processes:
  - `alignment-star-index`
  - `alignment-tophat2`
  - `cuffmerge`
  - `index-fasta-nucl`
  - `upload-fasta-nucl`
- Fix COSMIC url template in (multisample) amplicon reports

1.5.80 8.0.0 - 2018-04-11

Changed

- **BACKWARD INCOMPATIBLE:** Refactor `trimmomatic-single`, `trimmomatic-paired`, `bbduk-single`, and `bbduk-paired` processes
- **BACKWARD INCOMPATIBLE:** Merge `align-bwa-trim` and `align-bwa-trim2` process functionality. Retain only the refactored process under slug `align-bwa-trim`
- **BACKWARD INCOMPATIBLE:** In processes handling VCF files, the output VCF files are stored in bgzip-compressed form. Tabix index is not referenced to an original VCF file anymore, but stored in a separate `tbi` output field
- **BACKWARD INCOMPATIBLE:** Remove an obsolete `workflow-accel-2` workflow
- **BACKWARD INCOMPATIBLE:** Use Elasticsearch version 5.x
- **BACKWARD INCOMPATIBLE:** Parallelize execution of the following processes:
  - `alignment-bowtie2`
  - `alignment-bwa-mem`
  - `alignment-hisat2`
  - `alignment-star`
  - `alignment-tophat2`
  - `cuffdiff`
  - `cufflinks`
  - `cuffquant`
- Require Resolwe 8.x
- Bump STAR aligner version in `resolwebio/rnaseq` docker image to 2.5.4b
- Bump Primerclip version in `resolwebio/dnaseq` docker image
- Use `resolwebio/dnaseq` Docker image in `picard-pcrmetrics` process
- Run `vc-realign-recalibrate` process using multiple cpu cores to optimize the processing time
- Use `resolwebio/rnaseq` Docker image in `alignment-star` process

Added

- Add CNVKit, LoFreq and GATK to `resolwebio/dnaseq` docker image
- Add BaseSpace files download tool
- Add process to import a file from BaseSpace
- Add process to convert files to single-end reads
- Add process to convert files to paired-end reads
- Add `vc-gatk4-hc` process which implements GATK4 HaplotypeCaller variant calling tool
- Add `workflow-accel-gatk4` pipeline that uses GATK4 HaplotypeCaller as an alternative to GATK3 used in `workflow-accel` pipeline
- Add `amplicon-master-file` descriptor schema

- Add `workflow-bbduk-star-featurecounts` pipeline
- Add `rna-seq-bbduk-star-featurecounts` RNA-seq descriptor schema

Fixed

- Fix iterative trimming in `bowtie` and `bowtie2` processes
- Fix `archive-samples` to use sample names for headers when merging expressions
- Improve `goea.py` tool to handle duplicated mapping results
- Handle filenames with spaces in the following processes:
  - `alignment-hisat2`
  - `alignment-bowtie`
  - `prepare-geo-chipseq`
  - `prepare-geo-rnaseq`
  - `cufflinks`
  - `cuffquant`

1.5.81 7.0.1 - 2018-03-27

Fixed

- Use name-ordered BAM file for counting reads in `HTSeq-count` process by default to avoid buffer overflow with large BAM files

1.5.82 7.0.0 - 2018-03-13

Changed

- **BACKWARD INCOMPATIBLE:** Remove Ubuntu 17.04 base Docker image since it has has reached its end of life and change all images to use the new ubuntu 17.10 base image
- **BACKWARD INCOMPATIBLE:** Require `species` and `build` inputs in the following processes:
  - `upload-genome`
  - `upload-gtf`
  - `upload-gff3`
  - `upload-bam`
  - `upload-bam-indexed`
- **BACKWARD INCOMPATIBLE:** Track `species` and `build` information in the following processes:
  - `cuffmerge`
  - alignment processes
  - variant calling processes
  - JBrowse processes

- **BACKWARD INCOMPATIBLE:** Track `species`, `build` and `feature_type` in the following processes:
  - `upload-expression-star`
  - quantification processes
  - differential expression processes
- **BACKWARD INCOMPATIBLE:** Track `species` in gene set (Venn) and `goenrichment` processes
- **BACKWARD INCOMPATIBLE:** Rename `genes_source` input to `source` in hierarchical clustering and PCA processes
- **BACKWARD INCOMPATIBLE:** Remove the following obsolete processes:
  - Dictyostelium-specific ncRNA quantification
  - `go-geneset`
  - bayseq differential expression
  - `cuffmerge-gtf-to-gff3`
  - `transdecoder`
  - `web-gtf-dictybase`
  - `upload-rmsk`
  - `snpdat`
- **BACKWARD INCOMPATIBLE:** Unify output fields of processes of type `data:annotation`
- **BACKWARD INCOMPATIBLE:** Rename the organism field names to species in `rna-seq` and `cutadapt-star-htseq` descriptor schemas
- **BACKWARD INCOMPATIBLE:** Rename the `genome_and_annotation` field name to `species` in `bcm-*` descriptor schemas and use the full species name for the `species` field values
- **BACKWARD INCOMPATIBLE:** Refactor `featureCounts` process
- **BACKWARD INCOMPATIBLE:** Change `import-sra` process to work with `resolwebio/utils` Docker image and refactor its inputs
- Require Resolwe 7.x
- Add environment export for Jenkins so that the manager will use a globally-unique channel name
- Set `scheduling_class` of gene and sample hierarchical clustering processes to `interactive`
- Change base Docker images of `resolwebio/rnaseq` and `resolwebio/dnaseq` to `resolwebio/base:ubuntu-18.04`
- Use the latest versions of the following Python packages in `resolwebio/rnaseq` Docker image: Cutadapt 1.15, Apache Arrow 0.8.0, pysam 0.13, and xopen 0.3.2
- Use the latest versions of the following Python packages in `resolwebio/dnaseq` Docker image: Bokeh 0.12.13, pandas 0.22.0, Matplotlib 2.1.2, six 1.11.0, PyYAML 3.12, Jinja2 2.10, NumPy 1.14.0, Tornado 4.5.3, and pytz 2017.3
- Use the latest version of `wigToBigWig` tool in `resolwebio/chipseq` Docker image
- Use `resolwebio/rnaseq:3.0.0` Docker image in `goenrichment`, `upload-gaf` and `upload-obo` processes
- Use `resolwebio/dnaseq:3.0.0` Docker image in `filtering_chemut` process
- Change `cuffnorm` process type to `data:cuffnorm`
- Set type of `coverage-garvan` process to `data:exomecoverage`

- Remove `gsize` input from `macs14` process and automate genome size selection
- Adjust `bam-split` process so it can be included in workflows
- Make ID attribute labels in `featureCounts` more informative
- Change 'source' to 'gene ID database' in labes and descriptions
- Change `archive-samples` process to create different IGV session files for `build` and `species`
- Expose advanced parameters in Chemical Mutagenesis workflow
- Clarify some descriptions in the `filtering_chemut` process and `chemut` workflow
- Change expected genome build formatting for hybrid genomes in `bam-split` process
- Set the `cooksCutoff` parameter to `FALSE` in `deseq.R` tool
- Rename 'Expressions (BCM)' to 'Dicty expressions'

Added

- Mechanism to override the manager's control channel prefix from the environment
- Add Ubuntu 17.10 and Ubuntu 18.04 base Docker images
- Add `resolwebio/utils` Docker image
- Add `BBMap`, `Trimmomatic`, `Subread`, `Salmon`, and `dexseq_prepare_annotation2` tools and `DEXSeq` and `loadSubread` R libraries to `resolwebio/rnaseq` Docker image
- Add abstract processes that ensure that all processes that inherit from them have the input and output fields that are defined in them:
  - `abstract-alignment`
  - `abstract-annotation`
  - `abstract-expression`
  - `abstract-differentialexpression`
  - `abstract-bed`
- Add miRNA workflow
- Add `prepare-geo-chipseq` and `prepare-geo-rnaseq` processes that produce a tarball with necessary data and folder structure for GEO upload
- Add `library-strandedness` process which uses the `Salmon` tool built-in functionality to detect the library strandedness information
- Add `species` and `genome build` output fields to `macs14` process
- Expose additional parameters in `alignment-star`, `cutadapt-single` and `cutadapt-paired` processes
- Add `merge expressions` to `archive-samples` process
- Add description of batch mode to Expression aggregator process
- Add error and warning messages to the `cuffnorm` process
- Add optional `species` input to hierarchical clustering and PCA processes
- Add Rattus norvegicus species choice to the `rna-seq` descriptor schema to allow running RNA-seq workflow for this species from the Recipes

Fixed

- Fix custom argument passing script for `Trimmomatic` in `resolwebio/rnaseq` Docker image
- Fix installation errors for `dexseq-prepare-annotation2` in `resolwebio/rnaseq` Docker image
- Fix `consensus_subreads` input option in Subread process
- Limit variant-calling process in the chemical mutagenesis workflow and the Picard tools run inside to 16 GB of memory to prevent them from crashing because they try to use too much memory
- The chemical mutagenesis workflow was erroneously categorized as `data:workflow:rnaseq:cuffquant` type. This is switched to `data:workflow:chemut` type.
- Fix handling of NA values in Differential expression results table. NA values were incorrectly replaced with value 0 instead of 1
- Fix `cuffnorm` process to work with samples containing dashes in their name and dispense prefixing sample names starting with numbers with 'X' in the `cuffnorm` normalization outputs
- Fix `cuffnorm` process' outputs to correctly track species and build information
- Fix typos and sync parameter description common to `featureCounts` and `miRNA` workflow

1.5.83 6.2.2 - 2018-02-21

Fixed

- Fix `cuffnorm` process to correctly use sample names as labels in output files and expand `cuffnorm` tests

1.5.84 6.2.1 - 2018-01-28

Changed

- Update description text of `cutadapt-star-htseq` descriptor schema to better describe the difference between gene/transcript-type analyses
- Speed-up management command for inserting mappings

1.5.85 6.2.0 - 2018-01-17

Added

- Add R, tabix, and CheMut R library to `resolwebio/dnaseq` Docker image
- Add `SRA Toolkit` to `resolwebio/rnaseq` Docker image

Changed

- Require Resolwe 6.x
- Extend pathway map with species and source field
- Move template and logo for multi-sample report into `resolwebio/latex` Docker image
- Refactor `amplicon-report` process to contain all relevant inputs for `amplicon-archive-multi-report`
- Refactor `amplicon-archive-multi-report`
- Use `resolwebio/dnaseq:1.2.0` Docker image in `filtering_chemut` process

Fixed

- Enable DEBUG setting in tests using Django's `LiveServerTestCase`
- Wait for ElasticSeach to index the data in `KBBioProcessTestCase`
- Remove unused parameters in TopHat (2.0.13) process and Chip-seq workflow

1.5.86 6.1.0 - 2017-12-12

Added

- Add `amplicon-archive-multi-report` process
- Add `upload-metabolic-pathway` process
- Add memory-optimized primerclip as a separate `align-bwa-trim2` process
- Add `workflow-accel-2` workflow

Changed

- Improve `PCA` process performance
- Use `resolwebio/chipseq:1.1.0` Docker image in `macs14` process
- Change formatting of `EFF[*].AA` column in `snpeff` process
- Save unmapped reads in `aligment-hisat2` process
- Turn off test profiling

Fixed

- Fix pre-sorting in `upload-master-file` process
- Revert `align-bwa-trim` process to use non-memory-optimized primerclip
- Fix file processing in `cutadapt-custom-paired` process

1.5.87 6.0.0 - 2017-11-28

Added

- Add AF filter to amplicon report
- Add number of samples to the output of expression aggregator
- Add `ChIP-Rx`, `ChIPmentation` and `eClIP` experiment types to `reads` descriptor schema
- Add `pandas` Python package to `resolwebio/latex` Docker image
- Add primerclip, samtools, picard-tools and bwa to `resolwebio/dnaseq` Docker image
- Add `cufflinks`, `RNASeqT` R library, `pyarrow` and `sklearn` Python packages to `resolwebio/rnaseq` Docker image
- Add `wigToBigWig` tool to `resolwebio/chipseq` Docker image

Changed

- **BACKWARD INCOMPATIBLE:** Drop Python 2 support, require Python 3.4 or 3.5
- **BACKWARD INCOMPATIBLE:** Make species part of the feature primary key
- **BACKWARD INCOMPATIBLE:** Substitute Python 2 with Python 3 in `resolwebio/rnaseq` Docker image. The processes to be updated to this version of the Docker image should also have their Python scripts updated to Python 3.
- Require Resolwe 5.x
- Set maximum RAM requirement in `bbduk` process
- Move *Assay type* input parameter in RNA-Seq descriptor schema from advanced options to regular options
- Use `resolwebio/rnaseq` Docker image in Cutadapt processes
- Use additional adapter trimming option in `cutadapt-custom-single/paired` processes
- Show antibody information in `reads` descriptor for `ChIP-Seq`, `ChIPmentation`, `ChIP-Rx`, `eClIP`, `MNase-Seq`, `MeDIP-Seq`, `RIP-Seq` and `ChIA-PET` experiment types
- Use `resolwebio/dnaseq` Docker image in `align-bwa-trim` process
- Refactor `resolwebio/chipseq` Docker image
- Use Resolwe's Test Runner for running tests and add ability to only run a partial test suite based on what proceses have Changed
- Configure Jenkins to only run a partial test suite when testing a pull request
- Make tests use the live Resolwe API host instead of external server

Fixed

- Fix merging multiple expressions in DESeq process
- Fix `resolwebio/rnaseq` Docker image's README
- Handle multiple ALT values in amplicon report
- Fix BAM file input in `rsem` process

1.5.88 5.0.1 - 2017-11-14

Fixed

- Update Features and Mappings ElasticSearch indices building to be compatible with Resolwe 4.0

1.5.89 5.0.0 - 2017-10-25

Added

- Add automatic headers extractor to `bam-split` process
- Add HTML amplicon plot in `coveragebed` process
- Add raw RSEM tool output to *rsem* process output
- Add support for transcript-level differential expression in `deseq2` process

Changed

- **BACKWARD INCOMPATIBLE:** Bump Django requirement to version 1.11.x
- **BACKWARD INCOMPATIBLE:** Make `BioProcessTestCase` non-transactional
- Require Resolwe 4.x
- Add the advanced options checkbox to the `rna-seq` descriptor schema
- Remove static amplicon plot from `coveragebed` and `amplicon-report` processes
- Update Dockerfile for `resolwebio/latex` with newer syntax and add some additional Python packages

1.5.90 4.2.0 - 2017-10-05

Added

- Add `resolwebio/base` Docker image based on Ubuntu 17.04
- Add `resolwebio/dnaseq` Docker image
- Add `DESeq2` tool to `resolwebio/rnaseq` docker image
- Add input filename regex validator for `upload-master-file` process

Changed

- Remove obsolete mongokey escape functionality
- Report novel splice-site junctions in HISAT2
- Use the latest stable versions of the following bioinformatics tools in `resolwebio/rnaseq` docker image: Cutadapt 1.14, FastQC 0.11.5, HTSeq 0.9.1, and SAMtools 1.5

1.5.91 4.1.0 - 2017-09-22

Added

- Add Mus musculus to all BCM workflows' schemas
- Add `bam-split` process with supporting processes `upload-bam-primary`, `upload-bam-secondary` and `upload-header-sam`

Changed

- Enable Chemut workflow and process tests

Fixed

- Fix chemut `intervals` input option

1.5.92 4.0.0 - 2017-09-14

Added

- New base and legacy Docker images for processes, which support non-root execution as implemented by Resolwe

Changed

- **BACKWARD INCOMPATIBLE:** Modify all processes to explicitly use the new Docker images
- **BACKWARD INCOMPATIBLE:** Remove `clustering-hierarchical-genes-etc` process
- Require Resolwe 3.x

1.5.93 3.2.0 2017-09-13

Added

- Add `index-fasta-nucl` and `rsem` process
- Add custom Cutadapt - STAR - RSEM workflow

1.5.94 3.1.0 2017-09-13

Added

- Add statistics of logarithmized expressions to `expression-aggregator`
- Add input field description to `cutadapt-star-htseq` descriptor schema
- Add `HISAT2` and `RSEM` tool to `resolwebio/rnaseq` docker image

Changed

- Remove `eXpress` tool from `resolwebio/rnaseq` docker image
- Use system packages of RNA-seq tools in `resolwebio/rnaseq` docker image
- Set `hisat2` process' memory resource requirement to 32GB
- Use `resolwebio/rnaseq` docker image in `hisat2` process

1.5.95 3.0.0 2017-09-07

Added

- Add custom Cutadapt - STAR - HT-seq workflow
- Add expression aggregator process
- Add `resolwebio/rnaseq` docker image
- Add `resolwebio/latex` docker image
- Add access to sample field of data objects in processes via `sample` filter

Changed

- **BACKWARD INCOMPATIBLE:** Remove `threads` input in STAR aligner process and replace it with the `cores` resources requirement
- **BACKWARD INCOMPATIBLE:** Allow upload of custom amplicon master files (make changes to `amplicon-panel` descriptor schema, `upload-master-file` and `amplicon-report` processes and `workflow-accel` workflow)
- **BACKWARD INCOMPATIBLE:** Remove `threads` input in `cuffnorm` process and replace it with the `cores` resources requirement
- Add sample descriptor to `prepare_expression` test function
- Prettify amplicon report

Fixed

- Fix `upload-expression-star` process to work with arbitrary file names
- Fix STAR aligner to work with arbitrary file names
- Fix `cuffnorm` group analysis to work correctly
- Do not crop Amplicon report title as this may result in malformed LaTeX command
- Escape LaTeX's special characters in `make_report.py` tool
- Fix validation error in `Test sleep progress` process

1.5.96 2.0.0 2017-08-25

Added

- Support bioinformatics process test case based on Resolwe's `TransactionProcessTestCase`
- Custom version of Resolwe's `with_resolwe_host` test decorator which skips the decorated tests on non-Linux systems
- Add optimal leaf ordering and simulated annealing to gene and sample hierarchical clustering
- Add `resolwebio/chipseq` docker image and use it in ChIP-Seq processes
- Add Odocoileus virginianus texanus (deer) organism to sample descriptor
- Add test for `import-sra` process
- Add RNA-seq DSS test
- Add Cutadapt and custom Cutadapt processes

Changed

- Require Resolwe 2.0.x
- Update processes to support new input sanitization introduced in Resolwe 2.0.0
- Improve variant table name in amplicon report
- Prepend `api/` to all URL patterns in the Django test project
- Set `hisat2` process' memory resource requirement to 16GB and cores resource requirement to 1
- Filter LoFreq output VCF files to remove overlapping indels
- Add *Non-canonical splice sites penalty*, *Disallow soft clipping* and *Report alignments tailored specifically for Cufflinks* parameters to `hisat2` process
- Remove `threads` input from `cuffquant` and `rna-seq` workfows
- Set core resource requirement in `cuffquant` process to 1

Fixed

- Correctly handle paired-end parameters in `featureCount`
- Fix `NaN` in explained variance in PCA. When PC1 alone explained more than 99% of variance, explained variance for PC2 was not returned
- Fix input sanitization error in `dss-rna-seq` process
- Fix gene source check in hierarchical clustering and PCA
- Enable network access for all import processes
- Fix RNA-seq DSS adapters bug
- Fix sample hierarchical clustering output for a single sample case

1.5.97 1.4.1 2017-07-20

Changed

- Optionally report all amplicons in Amplicon table

Fixed

- Remove remaining references to calling `pip` with `--process-dependency-links` argument

1.5.98 1.4.0 2017-07-04

Added

- Amplicon workflow
- Amplicon descriptor schemas
- Amplicon report generator
- Add Rattus norvegicus organism choice to sample schema
- Transforming form Phred 64 to Phred 33 when uploading fastq reads
- Add primertrim process
- RNA-Seq experiment descriptor schema
- iCount sample and reads descriptor schemas
- iCount demultiplexing and sample annotation
- ICount QC
- Add MM8, RN4 and RN6 options to rose2 process
- Add RN4 and RN6 options to bamplot process
- Archive-samples process
- Add bamliquidator
- CheMut workflow
- Dicty primary analysis descriptor schema

- IGV session to Archive-samples process
- Use Resolwe's field projection mixins for knowledge base endpoints
- `amplicon-table` process
- Add C. griseus organism choice to Sample descriptor schema
- Add S. tuberosum organism choice to Sample descriptor schema
- Add log2 to gene and sample hierarchical clustering
- Add new inputs to import SRA, add read type selection process
- Set memory resource requirement in jbrowse annotation gff3 and gtf processes to 16GB
- Set memory resource requirement in star alignment and index processes to 32GB
- Add C. elegans organism choice to Sample descriptor schema
- Add D. melanogaster organism choice to Sample descriptor schema
- Set core resource requirement in Bowtie process to 1
- Set memory resource requirement in amplicon BWA trim process to 32GB
- Add new master file choices to amplicon panel descriptor schema
- Add S. tuberosum organism choice to RNA-seq workflow
- Add Cutadapt process
- Add leaf ordering to gene and sample hierarchical clustering

Fixed

- Use new import paths in `resolwe.flow`
- Upload reads (paired/single) containing whitespace in the file name
- Fix reads filtering processes for cases where input read file names contain whitespace
- Add additional filtering option to STAR aligner
- Fix bbduk-star-htseq_count workflow
- Fix cuffnorm process: Use sample names as labels (boxplot, tables), remove group labels input, auto assign group labels, add outputs for Rscript output files which were only available compressed
- Derive output filenames in hisat2 from the first reads filename
- Correctly fetch KB features in `goea.py`
- Append JBrowse tracks to sample
- Replace the BAM MD tag in *align-bwa-trim* process to correct for an issue with the primerclip tool
- Fix typo in trimmomatic and bbduk processes
- Use re-import in *etc* and *hmmer_database* processes

Changed

- Support Resolwe test framework
- Run tests in parallel with Tox
- Use Resolwe's new `FLOW_DOCKER_COMMAND` setting in test project
- Always run Tox's `docs`, `linters` and `packaging` environments with Python 3
- Add `extra` Tox testing environment with a check that there are no large test files in `resolwe_bio/tests/files`
- Replace Travis CI with Genialis' Jenkins for running the tests
- Store compressed and uncompressed .fasta files in `data:genome:fasta` objects
- Change sample_geo descriptor schema to have strain option available for all organisms
- More readable rna-seq-quantseq schema, field stranded
- Remove obsolete Gene Info processes
- Change log2(fc) default from 2 to 1 in diffexp descriptor schema
- Change Efective genome size values to actual values in macs14 process
- Change variable names in bowtie processes
- Remove iClip processes, tools, files and tests

1.5.99 1.3.0 2017-01-28

Changed

- Add option to save expression JSON to file before saving it to Storage
- Update `upload-expression` process
- No longer treat `resolwe_bio/tools` as a Python package
- Move processes' test files to the `resolwe_bio/tests/files` directory to generalize and simplify handling of tests' files
- Update differential expression (DE) processors
- Update `generate_diffexpr_cuffdiff` django-admin command
- Save gene_id source to `output.source` for DE, expression and related objects
- Refactor `upload-diffexp` processor
- Update sample descriptor schema
- Remove obsolete descriptor schemas
- Add stitch parameter to rose2 processor
- Add filtering to DESeq2
- Set Docker Compose's project name to `resolwebio` to avoid name clashes
- GO enrichment analysis: map features using gene Knowledge base
- Add option to upload .gff v2 files with upload-gtf processor
- Replace Haystack with Resolwe Elastic Search API

- Require Resolwe 1.4.1+
- Update processes to be compatible with Resolwe 1.4.0

Added

- Process definition documentation style and text improvements
- Add `resolwe_bio.kb` app, Resolwe Bioinformatics Knowledge Base
- Add tests to ensure generators produce the same results
- Upload Gene sets (`data:geneset`)
- Add `generate_geneset` django-admin command
- Add `generate_diffexpr_deseq` django-admin command
- Add 'Generate GO gene sets' processor
- Add generic file upload processors
- Add upload processor for common image file types (.jpg/.tiff/.png/.gif)
- Add upload processor for tabular file formats (.tab/.tsv/.csv/.txt/.xls/.xlsx)
- Add Trimmomatic process
- Add featureCounts process
- Add Subread process
- Add process for hierarchical clusteing of samples
- Add gff3 to gtf file converter
- Add microarray data descriptor schema
- Add process for differential expression edgeR
- `BioCollectionFilter` and `BidDataFilter` to support filtering collections and data by samples on API
- Added processes for automatically downloading single and paired end SRA files from NCBI and converting them to FASTQ
- Added process for automatically downloading SRA files from NCBI and converting them to FASTQ
- Add HEAT-Seq pipeline tools
- Add HEAT-Seq workflow
- Add `create-geneset`, `create-geneset-venn` processors
- Add `source` filter to feature search endpoint
- Add bamplot process
- Add gene hiererhical clustering
- Add cuffquant workflow
- Support Django 1.10 and versionfield 0.5.0
- django-admin commands `insert_features` and `insert_mappings` for importing features and mappings to the Knowledge Base
- Add bsmap and mcall to analyse WGBS data
- Vaccinesurvey sample descriptor schema

- Add RNA-Seq single and paired-end workflow

Fixed

- Set `presample` to `False` for Samples created on Sample endpoint
- Fix FastQC report paths in processors
- Fix `htseq_count` and `featureCounts` for large files
- Fix `upload gtf annotation`
- Fix gene_id field type for differential expression storage objects
- Order data objects in `SampleViewSet`
- Fix sample hiererhical clustering
- Fix name in gff to gtf process
- Fix clustering to read expressed genes as strings
- Fix protocol labels in `rna-seq-quantseq` descriptor schema

1.5.100 1.2.1 2016-07-27

Changed

- Update `resolwe` requirement

1.5.101 1.2.0 2016-07-27

Changed

- Decorate all tests that currently fail on Docker with `skipDockerFailure`
- Require Resolwe's `master` git branch
- Put packaging tests in a separate Tox testing environment
- Rename DB user in test project
- Change PostgreSQL port in test project
- Add ROSE2 results parser
- Compute index for HISAT2 aligner on genome upload
- Updated Cuffquant/Cuffnorm tools
- Change ROSE2 enhancer rank plot labels
- Refactor processor syntax
- Move processes tests into `processes` subdirectory
- Split `sample` API endpoint to `sample` for annotated `Samples` and `presample` for unannotated `Samples`
- Rename test project's data and upload directories to `.test_data` and `.test_upload`
- Save fastq files to `lists:basic:file` field. Refactor related processors.
- Reference genome-index path when running aligners.

- Add pre-computed genome-index files when uploading reference fasta file.
- Include all necessary files for running the tests in source distribution
- Exclude tests from built/installed version of the package
- Move testing utilities from `resolwe_bio.tests.processes.utils` to `resolwe_bio.utils.test`
- Update Cuffdiff processor inputs and results table parsing
- Refactor processes to use the updated `resolwe.flow.executors.run` command
- Refactor STAR aligner - export expressions as separate objects

Fixed

- Make Tox configuration more robust to different developer environments
- Set `required: false` in processor input/output fields where necessary
- Add `Sample`'s `Data objects` to `Collection` when `Sample` is added
- Fixed/renamed Cufflinks processor field names

Added

- `skipDockerFailure` test decorator
- Expand documentation on running tests
- Use Travis CI to run the tests
- Add `Sample` model and corresponding viewset and filter
- Add docker-compose command for PostgreSQL
- API endpoint for adding `Samples` to `Collections`
- HISAT2 aligner
- Use Git Large File Storage (LFS) for large test files
- Test for `generate_samples` django-admin command
- django-admin command: `generate_diffexpr`

1.5.102 1.1.0 2016-04-18

Changed

- Remove obsolete utilities superseded by resolwe-runtime-utils
- Require Resolwe 1.1.0

Fixed

- Update sample descriptor schema
- Include all source files and supplementary package data in sdist

Added

- `flow_collection: sample` to processes
- MACS14 processor
- Initial Tox configuration for running the tests
- Tox tests for ensuring high-quality Python packaging
- ROSE2 processor
- django-admin command: `generate_samples`

1.5.103 1.0.0 2016-03-31

Changed

- Renamed assertFileExist to assertFileExists
- Restructured processes folder hierarchy
- Removed re-require and hard-coded tools' paths

Fixed

- Different line endings are correctly handled when opening gzipped files
- Fail gracefully if the field does not exist in assertFileExists
- Enabled processor tests (GO, Expression, Variant Calling)
- Enabled processor test (Upload reads with old Illumina QC encoding)
- Made Resolwe Bioinformatics work with Resolwe and Docker

Added

- Import expressions from tranSMART
- Limma differential expression (tranSMART)
- VC filtering tool (Chemical mutagenesis)
- Additional analysis options to Abyss assembler
- API endpoint for Sample
- Initial documentation

1.6 Contributing

1.6.1 Installing prerequisites

Make sure you have Python 3.6 installed on your system. If you don't have it yet, follow these instructions.

Resolwe Bioinformatics requires PostgreSQL (9.4+). Many Linux distributions already include the required version of PostgreSQL (e.g. Fedora 22+, Debian 8+, Ubuntu 15.04+) and you can simply install it via distribution's package manager. Otherwise, follow these instructions.

The pip tool will install all Resolwe Bioinformatics' dependencies from PyPI. Installing some (indirect) dependencies from PyPI will require having a C compiler (e.g. GCC) as well as Python development files installed on the system.

Note: The preferred way to install the C compiler and Python development files is to use your distribution's packages, if they exist. For example, on a Fedora/RHEL-based system, that would mean installing `gcc` and `python3-devel` packages.

Optional prerequisites

If you want to run or develop tests with large input or output files, then install the Git Large File Storage extension.

1.6.2 Preparing environment

Fork the main Resolwe Bioinformatics' git repository.

If you don't have Git installed on your system, follow these instructions.

Clone your fork (replace `<username>` with your GitHub account name) and change directory:

```
git clone https://github.com/<username>/resolwe-bio.git
cd resolwe-bio
```

Prepare Resolwe Bioinformatics for development:

```
pip install --pre -e .[docs,package,test]
```

Note: We recommend using pyvenv to create an isolated Python environment for Resolwe Bioinformatics.

1.6.3 Preparing database

Add a postgres user:

```
createuser -s -r postgres
```

1.6.4 Running tests

Manually

Change directory to the `tests` Django project:

```
cd tests
```

Run docker:

```
docker-compose up
```

Note: On Mac or Windows, Docker might complain about non-mounted volumes. You can edit volumes in *Docker => Preferences => File Sharing* The following volumes need to be shared:

- /private
- /tmp
- /var/folders

`/private` is shared by default. When you attempt to add `/var/folders` it might try to add `/private/var/folders` which will cause Docker complaining about overlapping volumes. Here's a workaround: Change `/private` to `/var/folders` and then add `/private` again.

To run the tests, use:

```
./manage.py test resolwe_bio --parallel 2
```

Note: If you don't specify the number of parallel test processes (i.e. you just use `--parallel`), Django will run one test process per each core available on the machine.

Warning: If you run Docker in a virtual machine (i.e. if you use MacOS or Windows) rather that directly on your machine, the virtual machine can become totally unresponsive if you set the number of parallel test processes too high. We recommend using at most `--parallel 2` in such cases.

To run a specific test, use:

```
./manage.py test resolwe_bio.tests.<module-name>.<class-name>.<method-name>
```

For example, to run the `test_macs14` test of the `ChipSeqProcessorTestCase` class in the `test_chipseq` module, use:

```
./manage.py test resolwe_bio.tests.processes.test_chipseq.ChipSeqProcessorTestCase.test_
↪macs14
```

Using Tox

To run the tests with Tox, use:

```
tox
```

To re-create the virtual environment before running the tests, use:

```
tox -r
```

To only run the tests with a specific Python version, use:

```
tox -e py<python-version>
```

For example, to only run the tests with Python 3.5, use

```
tox -e py35
```

Note: To see the list of available Python versions, see `tox.ini`.

Note: To control the number of test processes Django will run in parallel, set the `DJANGO_TEST_PROCESSES` environment variable.

Since running tests for all processes may take a long time, there is an option to run partial tests based on what files have been changed between HEAD and a specific commit (e.g. master). The Tox environments that run partial tests have the `-partial` suffix, e.g.:

```
tox -e py35-partial
```

To configure the commit against which the changes are compared you should set the `RESOLWE_TEST_ONLY_CHANGES_TO` environmental variable (it is set to master by default).

Running tests skipped on Docker

To run the tests that are skipped on Docker due to failures and errors, set the `RESOLWEBIO_TESTS_SKIP_DOCKER_FAILURES` environment variable to `no`.

For example, to run the skipped tests during a single test run, use:

```
RESOLWEBIO_TESTS_SKIP_DOCKER_FAILURES=no ./manage.py test resolwe_bio
```

To run the skipped tests for the whole terminal session, execute:

```
export RESOLWEBIO_TESTS_SKIP_DOCKER_FAILURES=no
```

and then run the tests as usual.

Running tests with large files

To run the tests with large input or output files, ensure you have the Git Large File Storage extension installed and run the tests as usual.

Adding tests with large files

If a test file is larger than 1 MiB, then put it in the `resolwe_bio/tests/files/large/` directory. Git Large File Storage (LFS) extension will automatically pick it up and treat it appropriately.

To ensure contributors without Git LFS or users using the source distribution can smoothly run the tests, decorate the tests using large files with the following:

```
@skipUnlessLargeFiles(<large-file1>, <large-file2>, ...)
```

where `<large-file1>`, `<large-file2>`, ... represent the names of large files used inside a particular test.

The decorator will ensure the test is skipped unless these files are present and represent real large files (not just Git LFS pointers).

1.6.5 Building documentation

```
python setup.py build_sphinx
```

Note: To build the documentation, you must use Python 3 (Python 2 is not supported).

1.6.6 Preparing release

Follow Resolwe's documentation on preparing a release. Resolwe code is automatically released to PyPI when tagged, but this is not supported in Resolwe Bioinformatics yet. After you have completed the first part, follow the steps below to release the code on PyPI.

Clean `build` directory:

```
python setup.py clean -a
```

Remove previous distributions in `dist` directory:

```
rm dist/*
```

Remove previous `egg-info` directory:

```
rm -r *.egg-info
```

Create source distribution:

```
python setup.py sdist
```

Build wheel:

```
python setup.py bdist_wheel
```

Upload distribution to PyPI:

```
twine upload dist/*
```

CHAPTER

TWO

INDICES AND TABLES

- genindex
- modindex
- search

PYTHON MODULE INDEX

r

INDEX

F

G

H

I

K

M

O

P

Q

R

S

T

U

V

W

X