

Package: BP4RNAseq (via r-universe)

October 7, 2026

Type Package

Title A babysitter's package for reproducible RNA-seq analysis

Version 1.23.0

Description An automated pipe for reproducible RNA-seq analysis with the minimal efforts from researchers. The package can process bulk RNA-seq data and single-cell RNA-seq data. You can only provide the taxa name and the accession id of RNA-seq data deposited in the National Center for Biotechnology Information (NCBI). After a cup of tea or longer, you will get formatted gene expression data as gene count and transcript count based on both alignment-based and alignment-free workflows.

License GPL-2

Encoding UTF-8

LazyData FALSE

Imports dplyr, fastqcr, stringr, tidyr, stats, utils, magrittr, reticulate

Depends R (>= 4.0.0)

SystemRequirements UNIX, SRA Toolkit=2.10.3, Entrez Direct=13.3, FastQC=v0.11.9, Cutadapt=2.10, datasets, SAMtools=1.9, HISAT2=2.2.0, StringTie=2.1.1, Salmon=1.2.1

RoxygenNote 7.1.1

biocViews GeneExpressionWorkflow

Suggests knitr, rmarkdown, testthat

VignetteBuilder knitr

Config/pak/sysreqs

cmake make libicu-dev libpng-dev libuv1-dev libxml2-dev libssl-dev python3 libx11-dev

Repository <https://bioc.r-universe.dev>

Date/Publication 2026-04-28 12:44:47 UTC

RemoteUrl <https://github.com/bioc/BP4RNAseq>

RemoteRef HEAD

RemoteSha c76cfd2328b23530eccd220b19e28865d84a49d6

Contents

adapter_trim	2
align_based_quan	3
align_free_quan	4
average	5
check_dep	6
down_Ref	6
down2quan	7
fastq2quan	8
qc_test	9
quality_trim	9
scRNA_quan	10
sra_download	11
sra2fastq	12
Index	13

adapter_trim	<i>Trim samples with adapter using cutadapt.</i>
--------------	--

Description

Trim samples with adapter using cutadapt based on the results from quality_test().

Usage

```
adapter_trim(sample, pair, threads = 4, scRNA = FALSE, cutadapt_add = NULL)
```

Arguments

- sample samples with adapter.
- pair 'single' for single-end (SE) or 'paired' for paired-end (PE) reads.
- threads the number of threads to be used. Default is 4.
- scRNA logic, whether single-cell RNA-seq is quantified or not. Default is FALSE.
- cutadapt_add additional parameters to customize cutadapt to trim adapter. Default is NULL.

Value

None

Examples

```
adapter_trim(sample = 'test_read_2.fastq', pair = 'paired', scRNA = TRUE)
```

align_based_quan	<i>alignment-based workflow</i>
------------------	---------------------------------

Description

alignment-based workflow

Usage

```
align_based_quan(
  pair,
  taxa,
  genome,
  annotation,
  novel_transcript = FALSE,
  threads = 4,
  hisat2_build_add = NULL,
  hisat2_add = NULL,
  stringtie_add = NULL
)
```

Arguments

pair	'single' for single-end (SE) or 'paired' for paired-end (PE) reads.
taxa	the scientific or common name of the organism.
genome	the reference genome.
annotation	the annotation file.
novel_transcript	logic, whether identifying novel transcripts is expected or not. Default is FALSE.
threads	the number of threads to be used. Default is 4.
hisat2_build_add	additional parameters to customize hisat2 build command. Default is NULL.
hisat2_add	additional parameters to customize hisat2 command. Default is NULL.
stringtie_add	additional parameters to customize stringtie command. Default is NULL.

Value

None

Examples

```
## Not run:
test_Homo_sapiens.fna <- system.file('extdata', 'test_Homo_sapiens.fna', package = 'BP4RNAseq')
test_Homo_sapiens.gff <- system.file('extdata', 'test_Homo_sapiens.gff', package = 'BP4RNAseq')
align_based_quan(pair = 'paired', taxa = 'Homo sapiens',
  genome = test_Homo_sapiens.fna,
```

```

annotation = test_Homo_sapiens.gff, novel_transcript = FALSE)

## End(Not run)

```

align_free_quan	<i>Alignment-free expression quantification with Salmon at gene and transcript levels.</i>
-----------------	--

Description

Alignment-free expression quantification with Salmon at gene and transcript levels.

Usage

```

align_free_quan(
  pair,
  genome,
  transcript,
  annotation,
  threads = 4,
  salmon_index_add = NULL,
  salmon_quan_add = NULL
)

```

Arguments

pair	'single' for single-end (SE) or 'paired' for paired-end (PE) reads.
genome	the reference genome.
transcript	the reference transcript
annotation	the annotation file.
threads	the number of threads to be used. Default is 4.
salmon_index_add	additional parameters to customize salmon index command. Default is NULL.
salmon_quan_add	additional parameters to customize salmon quan command. Default is NULL.

Value

None

Examples

```
align_free_quan(
  pair = 'paired', 'test_Homo_sapiens.fna',
  'test_transcript_Homo_sapiens.fna', 'test_Homo_sapiens.gff'
)
```

average

*Take averages with alignment-free and alignment-based quantification***Description**

Take averages with alignment-free and alignment-based quantification

Usage

```
average(based_gene, free_gene, based_transcript, free_transcript)
```

Arguments

based_gene	gene count matrix assessed based on alignment-based workflow
free_gene	gene count matrix assessed based on alignment-free workflow
based_transcript	transcript count matrix assessed based on alignment-based workflow
free_transcript	transcript count matrix assessed based on alignment-free workflow

Value

the average of gene expression quantification with alignment-free and alignment-based quantification workflow

Examples

```
align_based_gene <- system.file(
  'extdata', 'test_gene_alignment_based_quantification.csv',
  package = 'BP4RNAseq'
)
align_free_gene <- system.file(
  'extdata', 'test_gene_alignment_free_quantification.csv',
  package = 'BP4RNAseq'
)
align_based_transcript <- system.file(
  'extdata', 'test_transcript_alignment_based_quantification.csv',
  package = 'BP4RNAseq'
)
```

```

align_free_transcript <- system.file(
  'extdata', 'test_transcript_alignment_free_quantification.csv',
  package = 'BP4RNAseq'
)
align_based_gene <- read.csv(align_based_gene, header = TRUE)
align_free_gene <- read.csv(align_free_gene, header = TRUE)
align_based_transcript <- read.csv(align_based_transcript, header = TRUE)
align_free_transcript <- read.csv(align_free_transcript, header = TRUE)
average(
  align_based_gene, align_free_gene,
  align_based_transcript, align_free_transcript
)

```

check_dep	<i>Check the existence of dependancies.</i>
-----------	---

Description

Trim samples with adapter using cutadapt based on the results from quality_test().

Usage

```
check_dep(dependancy)
```

Arguments

dependancy the dependancy to check.

Value

TRUE if the dependancy exists, otherwise FALSE.

Examples

```
check_dep(dependancy = 'cutadapt')
```

down_Ref	<i>Download the latest reference genome, transcript and the corresponding annotation files from NCBI.</i>
----------	---

Description

Download the latest reference genome and the corresponding annotation files from NCBI using the NCBI datasets command-line tool.

Usage

```
down_Ref(taxa)
```

Arguments

taxa the scientific or common name of the organism.

Value

None

Examples

```
down_Ref(taxa = 'Drosophila melanogaster')
```

down2quan	<i>Download the RNA-seq samples from the NCBI and produce the quantifications of gene expression.</i>
-----------	---

Description

Download the RNA-seq samples from the NCBI and produce the quantifications of gene expression with alignment-based and alignment-free workflows.

Usage

```
down2quan(
  accession,
  dir = getwd(),
  taxa,
  novel_transcript = FALSE,
  threads = 4,
  scRNA = FALSE,
  protocol
)
```

Arguments

accession the bioproject accession code for the RNA-seq samples deposited in the NCBI.

dir the working directory. Default is the current working directory.

taxa the scientific or common name of the organism.

novel_transcript logic, whether identifying novel transcripts is expected or not. Default is FALSE.

threads the number of threads to be used. Default is 4.

scRNA logic, whether single-cell RNA-seq is quantified or not. Default is FALSE.

protocol the single-cell RNA sequencing protocol: dropseq, chromium, or chromiumV3.

Value

None

Examples

```
down2quan(
  accession = 'SRR11427582', dir = getwd(),
  taxa = 'Drosophila melanogaster', novel_transcript = FALSE, scRNA = FALSE
)
```

fastq2quan

*Produce the quantifications of gene expression based on the fastq files.***Description**

Produce the quantifications of gene expression based on the fastq files with alignment-based and alignment-free workflows.

Usage

```
fastq2quan(
  threads = 4,
  dir = getwd(),
  pair,
  taxa,
  novel_transcript = FALSE,
  scRNA = FALSE,
  protocol = NULL
)
```

Arguments

threads	the number of threads to be used. Default is 4.
dir	the working directory. Default is the current working directory.
pair	'single' for single-end (SE) reads or 'paired' for paired-end (PE) reads.
taxa	the scientific or common name of the organism.
novel_transcript	logic, whether identifying novel transcripts is expected or not. Default is FALSE.
scRNA	logic, whether single-cell RNA-seq is quantified or not. Default is FALSE.
protocol	the single-cell RNA sequencing protocol: dropseq, chromium, or chromiumV3.

Value

None

Examples

```
sra_download(accession = 'SRR11427582')
sra2fastq()
fastq2quan(pair = 'single', taxa = 'Drosophila melanogaster')
```

qc_test*Quality assessments of raw sequence data.*

Description

Per base sequence quality, per sequence quality scores and adapter content of raw sequence data are assessed and tested based on FastQC.

Usage

```
qc_test(threads = 4, scRNA = FALSE, fastqc_add = NULL)
```

Arguments

threads	the number of threads to be used. Default is 4.
scRNA	logic, whether single-cell RNA-seq is quantified or not. Default is FALSE.
fastqc_add	additional parameters to customize FASTQC for quality control. Default is NULL.

Value

Problematic samples' names

Examples

```
qc_test(scRNA = TRUE)
```

quality_trim*Trim poor quality samples using cutadapt.*

Description

Trim poor quality samples using cutadapt based on the results from quality_test().

Usage

```
quality_trim(
  per_base,
  per_seq,
  pair,
  threads = 4,
  scRNA = FALSE,
  cutadapt_add = NULL
)
```

Arguments

per_base	samples with poor per base sequence quality.
per_seq	samples with poor per sequence quality scores.
pair	'single' for single-end (SE) or 'paired' for paired-end (PE) reads.
threads	the number of threads to be used. Default is 4.
scRNA	logic, whether single-cell RNA-seq is quantified or not. Default is FALSE.
cutadapt_add	additional parameters to customize cutadapt to trim poor quality reads. Default is NULL

Value

None

Examples

```
quality_trim(per_base = 'test_read_2.fastq', per_seq = 'test_read_2.fastq',
  pair = 'paired', threads = 4, scRNA = TRUE)
```

scRNA_quan

scRNA expression quantification with Salmon.

Description

scRNA expression quantification with Salmon.

Usage

```
scRNA_quan(
  transcript,
  protocol,
  threads = 4,
  salmon_index_add = NULL,
  salmon_alevin_add = NULL
)
```

Arguments

transcript	the reference transcript
protocol	the single-cell RNA sequencing protocol: dropseq, chromium, or chromiumV3.
threads	the number of threads to be used. Default is 4.
salmon_index_add	additional parameters to customize salmon index command. Default is NULL.
salmon_alevin_add	additional parameters to customize salmon alevin command. Default is NULL.

Value

None

Examples

```
scRNA_quan(transcript = 'test_transcript_Homo_sapiens.fna', protocol = 'dropseq')
```

sra_download	<i>Download RNA-seq samples from the NCBI.</i>
--------------	--

Description

Download RNA-seq samples from the NCBI using SRA Toolkit and Entrez Direct based on the accession code.

Usage

```
sra_download(accession, dir = getwd())
```

Arguments

accession	the bioproject accession code for the RNA-seq samples deposited in the NCBI.
dir	the working directory. Default is the current working directory.

Value

the status of the download (0 for success, 1 for error).

Examples

```
sra_download(accession = 'SRR11427582')
```

sra2fastq	<i>Convert SRA format to fastq format and return the sequencing type, i.e., single-end (SE) or paired-end (PE) reads.</i>
-----------	---

Description

Convert all SRA files to fastq files and return the sequencing type, i.e., single-end (SE) or paired-end (PE) reads.

Usage

```
sra2fastq(threads = 4)
```

Arguments

threads	the number of threads to be used. Default is 4.
---------	---

Value

A string indicates single-end (SE) or paired-end (PE) reads.

Examples

```
sra_download(accession = 'SRR11427582')  
sra2fastq()
```

Index

adapter_trim, [2](#)
align_based_quan, [3](#)
align_free_quan, [4](#)
average, [5](#)

check_dep, [6](#)

down2quan, [7](#)
down_Ref, [6](#)

fastq2quan, [8](#)

qc_test, [9](#)
quality_trim, [9](#)

scRNA_quan, [10](#)
sra2fastq, [12](#)
sra_download, [11](#)