Package 'crisprBowtie'

April 7, 2026

Version 1.15.0

Date 2025-07-22

Title Bowtie-based alignment of CRISPR gRNA spacer sequences

Depends methods

Imports BiocGenerics, Biostrings, BSgenome, crisprBase (>= 0.99.15), Seqinfo, GenomicRanges, IRanges, Rbowtie, readr, stats, stringr, utils

Suggests BiocStyle, BSgenome.Hsapiens.UCSC.hg38, knitr, rmarkdown, testthat

biocViews CRISPR, FunctionalGenomics, Alignment

Description Provides a user-friendly interface to map on-targets and off-targets of CRISPR gRNA spacer sequences using bowtie. The alignment is fast, and can be performed using either commonly-used or custom CRISPR nucleases. The alignment can work with any reference or custom genomes. Both DNA- and RNA-targeting nucleases are supported.

License MIT + file LICENSE

Encoding UTF-8

RoxygenNote 7.2.1

VignetteBuilder knitr

BugReports https://github.com/crisprVerse/crisprBowtie/issues

URL https://github.com/crisprVerse/crisprBowtie

git_url https://git.bioconductor.org/packages/crisprBowtie

git_branch devel

git_last_commit 5838b33

git_last_commit_date 2025-10-29

Repository Bioconductor 3.23

Date/Publication 2026-04-07

Author Jean-Philippe Fortin [aut, cre]

Maintainer Jean-Philippe Fortin <fortin946@gmail.com>

Contents

`runBowtie` *Perform short sequence alignment with bowtie*

Description

Perform short sequence alignment with bowtie.

Usage

```
runBowtie(
  sequences,
  bowtie_index,
  bsgenome = NULL,
  n_mismatches = 0,
  all_alignments = TRUE,
  n_max_alignments = 1000,
  verbose = TRUE
)
```

Arguments

`sequences`	Character vector of DNA sequences.
`bowtie_index`	String specifying path to a bowtie index.
`bsgenome`	BSgenome object.
`n_mismatches`	Integer between 0 and 3 specifying maximum number of mismatches allowed between query sequences and target DNA. 0 by default.
`all_alignments`	Should all possible alignments be returned? TRUE by default.
`n_max_alignments`	Maximum number of alignments to return if `all_alignments` is FALSE. 1000 by default.
`verbose`	Should messages be printed to the console? TRUE by default.

Details

fasta <- system.file(package="crisprBowtie", "example/chr1.fa") outdir <- tempdir() Rbowtie::bowtie_build(fasta,outdir=outd force=TRUE, prefix="tempIndex")

`runBowtie` can be used to map short DNA sequences to a reference genome. To search for sequences while imposing constraints on PAM sequences (such as gRNA spacer sequences), see `runCrisprBowtie` instead.

Value

A data.frame of the alignments with the following columns:

- `query` — string specifying query DNA sequence
- `target` — string specifying target DNA sequence
- `chr` - string specifying chromosome name
- `pos` - string specifying genomic coordinate of the start of the target DNA sequence
- `strand` - string specifying strand ("+" or "-")
- `n_mismatches` - integer specifying number of mismatches between query and target sequences

Author(s)

Jean-Philippe Fortin

See Also

`runCrisprBowtie` to map gRNA spacer sequences.

Examples

```
fasta <- system.file(package="crisprBowtie", "example/chr1.fa")
outdir <- tempdir()
Rbowtie::bowtie_build(fasta,outdir=outdir, force=TRUE, prefix="tempIndex")
index <- file.path(outdir, "tempIndex")
seqs <- c("GGAAGT",
          "GTGGAC",
          "GTGTGC")
results <- runBowtie(seqs, bowtie_index=index, n_mismatches=2)
```

`runCrisprBowtie` *Perform CRISPR gRNA spacer alignment with bowtie*

Description

Perform CRISPR gRNA spacer alignment with bowtie.

Usage

```
runCrisprBowtie(
  spacers,
  mode = c("protospacer", "spacer"),
  bowtie_index = NULL,
  bsgenome = NULL,
  crisprNuclease = NULL,
  canonical = TRUE,
```

```
  ignore_pam = FALSE,
  n_mismatches = 0,
  all_alignments = TRUE,
  n_max_alignments = 1000,
  force_spacer_length = FALSE,
  rna_strict_directionality = TRUE,
  verbose = TRUE
)
```

Arguments

spacers	Character vector specifying gRNA spacer sequences. Sequences must all be of equal length.
mode	String specifying which alignment mode should be used: protospacer or spacer. For RNA-targeting nucleases such as CasRx, only the protospacer mode can be used.
bowtie_index	String specifying path to a bowtie index.
bsgenome	A BSgenome object. Must be provided if mode is "spacer". Ignore
crisprNuclease	A CrisprNuclease object.
canonical	Should only canonical PAM sequences be considered? TRUE by default.
ignore_pam	Should PAM sequences be ignore? If TRUE, all alignments are returned regardless of PAM tolerance. FALSE by default.
n_mismatches	Integer between 0 and 3 specifying maximum number of mismatches allowed between spacer sequences and target DNA. 0 by default.
all_alignments	Should all possible alignments be returned? TRUE by default.
n_max_alignments	Maximum number of alignments to return if all_alignments is FALSE. 1000 by default.
force_spacer_length	Should the spacer length be overwritten in the crisprNuclease object? FALSE by default.
rna_strict_directionality	Should only protospacers found in the original direction of the RNA be considered for RNA-targeting nucleases? TRUE by default.
verbose	Should messages be printed to the console? TRUE by default.

Details

When mode is "spacer", spacer sequences are aligned to the reference index without appending PAM sequences first. This requires the specification of a BSgenome object through the argument bsgenome to validate that the aligned spacer sequences are adjacent to valid PAM sequences.

When mode is "protospacer", sequences are aligned with all valid PAM sequences appended (spacer + PAM). The set of valid PAM sequences depend on the inputs canonical and ignore_pam. This is faster than the "spacer" mode if the number of possible PAM sequences is small (e.g. SpCas9).

For RNA-targeting nucleases, such as RfxCas13d (CasRx), the bowtie index should be built on a transcriptome. For such applications, only the "protospacer" mode can be used as there is no corresponding bsgenome package. The protospacer sequences searched in the reference index will be the reverse complement of the input spacer sequences.

Value

A data.frame of the spacer alignments with the following columns:

- `spacer` — string specifying gRNA spacer sequence
- `protospacer` — string specifying target protospacer sequence
- `pam` — string specifying target PAM sequence
- `chr` - string specifying chromosome name
- `pam_site` - string specifying genomic coordinate of the first nucleotide of the PAM sequence.
- `strand` - string specifying strand ("+" or "-")
- `n_mismatches` - integer specifying number of mismatches between spacer and protospacer sequences
- `canonical` - logical indicating whether or not PAM sequence is canonical.

Author(s)

Jean-Philippe Fortin

See Also

`runBowtie` to map general DNA sequences.

Examples

```
fasta <- system.file(package="crisprBowtie", "example/chr1.fa")
outdir <- tempdir()
Rbowtie::bowtie_build(fasta,outdir=outdir, force=TRUE, prefix="tempIndex")
index <- file.path(outdir, "tempIndex")
seqs <- c("GGAAATTCCCCCAGTGGCGC",
          "ACACAGCTGCGGACAGGGCC")
data(SpCas9, package="crisprBase")
results <- runCrisprBowtie(seqs,
                           bowtie_index=index,
                           n_mismatches=2,
                           crisprNuclease=SpCas9)
```

Index