Package 'motifmatchr'

November 12, 2026

Type Package

Title Fast Motif Matching in R

Version 1.34.0

Date 2025-07-22

Maintainer Alicia Schep <aschep@gmail.com>

Description Quickly find motif matches for many motifs and many sequences. Wraps C++ code from the MOODS motif calling library, which was developed by Pasi Rastas, Janne Korhonen, and Petri Martinmäki.

License GPL-3 + file LICENSE

Imports Matrix, Rcpp, methods, TFBSTools, Biostrings, BSgenome, S4Vectors, SummarizedExperiment, GenomicRanges, IRanges, Rsamtools, Seqinfo

Depends R (>= 3.3)

Suggests testthat, knitr, rmarkdown, BSgenome.Hsapiens.UCSC.hg19

biocViews MotifAnnotation

LinkingTo Rcpp, RcppArmadillo

SystemRequirements C++11

RoxygenNote 6.0.1

VignetteBuilder knitr

Encoding UTF-8

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

git_branch RELEASE_3_23

git_last_commit baa1e11

git_last_commit_date 2026-04-28

Repository Bioconductor 3.23

Date/Publication 2026-11-11

Author Alicia Schep [aut, cre],
Stanford University [cph]

Contents

example_motifs 2
matchMotifs 2
motifCounts 5
motifMatches 6
motifmatchr 7
motifmatchr_deprecated 7
motifScores 8
pwmType 9

Index **10**

`example_motifs` *example_motifs*

Description

A few example motifs from JASPAR 2016 for trying out motifmatchr

Usage

```
data(example_motifs)
```

Value

`PFMatrixList` of length 3

Examples

```
data(example_motifs, package = "motifmatchr")
```

`matchMotifs` *matchMotifs*

Description

Find motif matches

Usage

```
matchMotifs(pwms, subject, ...)

## S4 method for signature 'PWMatrixList,DNAStringSet'
matchMotifs(pwms, subject,
  genome = NULL, bg = c("subject", "genome", "even"), out = c("matches",
  "scores", "positions"), p.cutoff = 5e-05, w = 7, ranges = NULL)

## S4 method for signature 'PWMatrixList,character'
matchMotifs(pwms, subject, genome = NULL,
  bg = c("subject", "genome", "even"), out = c("matches", "scores",
  "positions"), p.cutoff = 5e-05, w = 7, ranges = NULL)

## S4 method for signature 'PWMatrixList,DNAString'
matchMotifs(pwms, subject, genome = NULL,
  bg = c("subject", "genome", "even"), out = c("matches", "scores",
  "positions"), p.cutoff = 5e-05, w = 7, ranges = NULL)

## S4 method for signature 'PWMatrixList,GenomicRanges'
matchMotifs(pwms, subject,
  genome = Seqinfo::genome(subject), bg = c("subject", "genome",
  "even"), out = c("matches", "scores", "positions"), p.cutoff = 5e-05,
  w = 7)

## S4 method for signature 'PWMatrixList,RangedSummarizedExperiment'
matchMotifs(pwms, subject,
  genome = Seqinfo::genome(subject), bg = c("subject", "genome",
  "even"), out = c("matches", "scores", "positions"), p.cutoff = 5e-05,
  w = 7)

## S4 method for signature 'PWMatrixList,BSgenomeViews'
matchMotifs(pwms, subject,
  bg = c("subject", "genome", "even"), out = c("matches", "scores",
  "positions"), p.cutoff = 5e-05, w = 7)

## S4 method for signature 'PFMatrixList,ANY'
matchMotifs(pwms, subject, ...)

## S4 method for signature 'PWMatrix,ANY'
matchMotifs(pwms, subject, ...)

## S4 method for signature 'PFMatrix,ANY'
matchMotifs(pwms, subject, ...)
```

Arguments

pwms	either PFMatrix, PFMatrixList, PWMatrix, PWMatrixList
subject	either GenomicRanges, DNAStringSet, DNAString, or character vector

...	additional arguments depending on inputs
genome	BSgenome object, DNAStringSet, or FaFile, or short string signifying genome build recognized by getBSgenome. Only required if subject is GenomicRanges or RangedSummarizedExperiment or if bg is set to "genome"
bg	background nucleotide frequencies. Default is to compute based on subject, i.e. the specific set of sequences being evaluated. See Details.
out	what to return? see return section
p.cutoff	p-value cutoff for returning motifs
w	parameter controlling size of window for filtration; default is 7
ranges	if subject is not GenomicRanges or RangedSummarizedExperiment, these ranges can be used to specify what ranges the input sequences correspond to. These ranges will be incorporated into the SummarizedExperiment output if out is "matches" or "scores" or will be used to give absolute positions of motifs if out is "positions"

Details

Background nucleotide frequencies can be set to "subject" to use the subject sequences or ranges for computing the nucleotide frequencies, "genome" for using the genomice frequencies (in which case a genome must be specified), "even" for using 0.25 for each base, or a numeric vector with A, C, G, and T frequencies.

Value

Either returns a SummarizedExperiment with a sparse matrix with values set to TRUE for a match (if out == 'matches'), a SummarizedExperiment with a matches matrix as well as matrices with the maximum motif score and total motif counts (if out == 'scores'), or a GenomicRangesList or a list of IRangesList with all the positions of matches (if out == 'positions')

Methods (by class)

- pwms = PWMatrixList,subject = DNAStringSet: PWMatrixList/DNAStringSet
- pwms = PWMatrixList,subject = character: PWMatrixList/character
- pwms = PWMatrixList,subject = DNAString: PWMatrixList/DNAString
- pwms = PWMatrixList,subject = GenomicRanges: PWMatrixList/GenomicRanges
- pwms = PWMatrixList,subject = RangedSummarizedExperiment: PWMatrixList/RangedSummarizedExperiment
- pwms = PWMatrixList,subject = BSgenomeViews: PWMatrixList/BSGenomeViews
- pwms = PFMatrixList,subject = ANY: PFMatrixList/ANY
- pwms = PWMatrix,subject = ANY: PWMatrix/ANY
- pwms = PFMatrix,subject = ANY: PFMatrix/ANY

Examples

```
data(example_motifs, package = "motifmatchr")

# Make a set of peaks
peaks <- GenomicRanges::GRanges(seqnames = c("chr1","chr2","chr2"),
                ranges = IRanges::IRanges(start = c(76585873,42772928,
                                                    100183786),
                                                    width = 500))

# Get motif matches for example motifs
motif_ix <- matchMotifs(example_motifs, peaks, genome = "BSgenome.Hsapiens.UCSC.hg19")
```

`motifCounts` *motifCounts*

Description

get motif counts from SummarizedExperiment object

Usage

```
motifCounts(object)

## S4 method for signature 'SummarizedExperiment'
motifCounts(object)
```

Arguments

`object`	SummarizedExperiment object with counts assay

Value

matrix with counts

Methods (by class)

- `SummarizedExperiment`: method for SummarizedExperiment

Examples

```
data(example_motifs, package = "motifmatchr")

# Make a set of peaks
peaks <- GenomicRanges::GRanges(seqnames = c("chr1","chr2","chr2"),
                ranges = IRanges::IRanges(start = c(76585873,42772928,
                                                    100183786),
                                                    width = 500))
```

```
# Get motif matches for example motifs
motif_ix <- matchMotifs(example_motifs, peaks,
                        genome = "BSgenome.Hsapiens.UCSC.hg19",
                        out = "scores")

motifCounts(motif_ix)
```

`motifMatches` *motifMatches*

Description

get motif matches from SummarizedExperiment object

Usage

```
motifMatches(object)

## S4 method for signature 'SummarizedExperiment'
motifMatches(object)
```

Arguments

`object`	SummarizedExperiment object with matches assay

Value

matrix with scores

Methods (by class)

- `SummarizedExperiment`: method for SummarizedExperiment

Examples

```
data(example_motifs, package = "motifmatchr")

# Make a set of peaks
peaks <- GenomicRanges::GRanges(seqnames = c("chr1","chr2","chr2"),
                ranges = IRanges::IRanges(start = c(76585873,42772928,
                                                    100183786),
                                                    width = 500))

# Get motif matches for example motifs
motif_ix <- matchMotifs(example_motifs, peaks,
                        genome = "BSgenome.Hsapiens.UCSC.hg19")

motifMatches(motif_ix)
```

`motifmatchr` *motifmatchr: Fast Motif Matching in R*

Description

The motifmatchr package is designed for analyzing many sequences and many motifs to find which sequences contain which motifs.

Details

motifmatchr uses the MOODS C++ library (developedby Pasi Rastas, Janne Korhonen, and Petri Martinmaki) internally for motif matching.

The primary method of motifmatchr is `matchMotifs`, which takes in motif PWMs/PFMs and genomic ranges or sequences and returns either which ranges/sequences match which motifs or the positions of the matches.

Compared with alternative motif matching functions available in Bioconductor (e.g. matchPWM in Biostrings or searchSeq in TFBSTools), motifmatchr is designed specifically for the use case of determining whether many different sequences/ranges contain many different motifs.

Author(s)

Alicia Schep

`motifmatchr_deprecated`

Deprecated functions in motifmatchr

Description

motifmatchr has moved functions and methods to camelCase from snake_case. The following functions have been deprecated and replaced with a different name:

- motif_matches is now `motifMatches`
- motif_counts is now `motifCounts`
- motif_scores is now `motifScores`
- match_motifs is now `matchMotifs`

Usage

```
motif_matches(...)

motif_counts(...)

motif_scores(...)

match_motifs(...)
```

Arguments

...	arguments passed to new function

Value

calls the replacement function

Author(s)

Alicia Schep

`motifScores` *motifScores*

Description

get motif scores from SummarizedExperiment object

Usage

```
motifScores(object)

## S4 method for signature 'SummarizedExperiment'
motifScores(object)
```

Arguments

`object`	SummarizedExperiment object with scores assay

Value

matrix with scores

Methods (by class)

- `SummarizedExperiment`: method for SummarizedExperiment

Examples

```
data(example_motifs, package = "motifmatchr")

# Make a set of peaks
peaks <- GenomicRanges::GRanges(seqnames = c("chr1","chr2","chr2"),
                ranges = IRanges::IRanges(start = c(76585873,42772928,
                                                    100183786),
                                                    width = 500))

# Get motif matches for example motifs
motif_ix <- matchMotifs(example_motifs, peaks,
```

```
                          genome = "BSgenome.Hsapiens.UCSC.hg19",
                          out = "scores")

motifScores(motif_ix)
```

---

pwmType *pwmType*

Description

Determines type of PWM

Usage

```
pwmType(pwm)
```

Arguments

pwm	PWMatrix object

Value

'log','log2', or 'frequency' depending on type of pwm

Examples

```
data(example_motifs, package = "motifmatchr")
pwmType(TFBSTools::toPWM(example_motifs[[1]]))
pwmType(TFBSTools::toPWM(example_motifs[[1]], type = "prob"))
```

Index

∗ **datasets**
example_motifs, 2
∗ **internal**
pwmType, 9

DNAString, *3*
DNAStringSet, *3, 4*

example_motifs, 2

FaFile, *4*

GenomicRanges, *3, 4*
GenomicRangesList, *4*
getBSgenome, *4*

IRangesList, *4*

match_motifs *(*motifmatchr_deprecated*)*, 7
matchMotifs, 2, *7*
matchMotifs,PFMatrix,ANY-method *(*matchMotifs*)*, 2
matchMotifs,PFMatrixList,ANY-method *(*matchMotifs*)*, 2
matchMotifs,PWMatrix,ANY-method *(*matchMotifs*)*, 2
matchMotifs,PWMatrixList,BSgenomeViews-method *(*matchMotifs*)*, 2
matchMotifs,PWMatrixList,character-method *(*matchMotifs*)*, 2
matchMotifs,PWMatrixList,DNAString-method *(*matchMotifs*)*, 2
matchMotifs,PWMatrixList,DNAStringSet-method *(*matchMotifs*)*, 2
matchMotifs,PWMatrixList,GenomicRanges-method *(*matchMotifs*)*, 2
matchMotifs,PWMatrixList,RangedSummarizedExperiment-method *(*matchMotifs*)*, 2
motif_counts *(*motifmatchr_deprecated*)*, 7
motif_matches *(*motifmatchr_deprecated*)*, 7
motif_scores *(*motifmatchr_deprecated*)*, 7
motifCounts, 5, *7*
motifCounts,SummarizedExperiment-method *(*motifCounts*)*, 5
motifMatches, 6, *7*
motifMatches,SummarizedExperiment-method *(*motifMatches*)*, 6
motifmatchr, 7
motifmatchr-package *(*motifmatchr*)*, 7
motifmatchr_deprecated, 7
motifScores, *7*, 8
motifScores,SummarizedExperiment-method *(*motifScores*)*, 8

PFMatrix, *3*
PFMatrixList, *2, 3*
PWMatrix, *3*
PWMatrixList, *3*
pwmType, 9

RangedSummarizedExperiment, *4*