

Package ‘MatrixRider’

January 27, 2025

Type Package

Title Obtain total affinity and occupancies for binding site matrices on a given sequence

Version 1.39.0

Date 2015-11-02

Author Elena Grassi

Maintainer Elena Grassi <grassi.e@gmail.com>

Description Calculates a single number for a whole sequence that reflects the propensity of a DNA binding protein to interact with it. The DNA binding protein has to be described with a PFM matrix, for example gotten from Jaspar.

biocViews GeneRegulation, Genetics, MotifAnnotation

License GPL-3

Depends R (>= 3.1.2)

Imports methods, TFBSTools, IRanges, XVector, Biostrings

Suggests RUnit, BiocGenerics, BiocStyle, JASPAR2014

LinkingTo IRanges, XVector, Biostrings, S4Vectors

NeedsCompilation yes

git_url <https://git.bioconductor.org/packages/MatrixRider>

git_branch devel

git_last_commit bddf4b1

git_last_commit_date 2024-10-29

Repository Bioconductor 3.21

Date/Publication 2025-01-27

Contents

MatrixRider-package	2
getSeqOccupancy	2

Index	4
--------------	----------

MatrixRider-package	<i>Calculate total affinity and occupancies for binding site matrices on a given sequence</i>
---------------------	-----------------------------------------------------------------------------------------------

Description

Calculates a single number for a whole sequence that reflects the propensity of a DNA binding protein to interact with it. The DNA binding protein has to be described with a PFM matrix, for example gotten from Jaspar.

Author(s)

Elena Grassi <elena.grassi@unito.it>

Examples

```
library(JASPAR2014)
library(TFBSTools)
library(Biostrings)
pfm <- getMatrixByID(JASPAR2014,"MA0004.1")
## The following sequence has a single perfect match
## thus it gives the same results with all cutoff values.
sequence <- DNASTring("CACGTG")
getSeqOccupancy(sequence, pfm, 0.1)
getSeqOccupancy(sequence, pfm, 1)
```

getSeqOccupancy	<i>Computes the total affinity or the occupancy at a given cutoff</i>
-----------------	-----------------------------------------------------------------------

Description

The affinity/occupancy is calculated on the given DNASTring for the given PFMatrix (or all the ones in the PFMatrix list given).

Usage

```
getSeqOccupancy(sequence, pfm, cutoff)
```

Arguments

sequence	A DNASTring object with the sequence for which affinity will be computed.
pfm	A PFMatrix or a PFMatrixList object with the matrixes whose affinity will be calculated. The background (bg,XMatrix-method) of the given pfm is used to perform affinity calculations.
cutoff	numeric(1); between 0 and 1 (included): 0 corresponds to total affinity (i.e. summing all the affinities) while 1 to summing only values corresponding to the perfect match for a given PFMatrix . See vignette for details on how scores are calculated. If MatrixRider is installed, open the vignette with <code>vignette("MatrixRider")</code> .

Value

numeric; the resulting total affinity calculated on the given fasta. If a [PFMatrixList](#) has been passed then a named numeric vector with the affinities for all the PFMs. The vignette has all the details on the calculations (such as PFM to PWM conversion and pseudocounts).

Examples

```
library(JASPAR2014)
library(TFBSTools)
library(Biostrings)
pfm <- getMatrixByID(JASPAR2014,"MA0004.1")
## The following sequence has a single perfect match
## thus it gives the same results with all cutoff values.
sequence <- DNASTring("CACGTG")
getSeqOccupancy(sequence, pfm, 0.1)
getSeqOccupancy(sequence, pfm, 1)

pfm2 <- getMatrixByID(JASPAR2014,"MA0005.1")
pfms <- PFMatrixList(pfm, pfm2)
names(pfms) <- c(name(pfm), name(pfm2))
## This calculates total affinity for both the PFMatrixes.
getSeqOccupancy(sequence, pfms, 0)
```

Index

* **MatrixRider**

MatrixRider-package, [2](#)

* **getSeqOccupancy**

getSeqOccupancy, [2](#)

bg, XMatrix-method, [2](#)

DNAStrng, [2](#)

getSeqOccupancy, [2](#)

getSeqOccupancy, DNAStrng, PFMatrix, numeric-method
(getSeqOccupancy), [2](#)

getSeqOccupancy, DNAStrng, PFMatrixList, numeric-method
(getSeqOccupancy), [2](#)

getSeqOccupancy-methods
(getSeqOccupancy), [2](#)

MatrixRider (MatrixRider-package), [2](#)

MatrixRider-package, [2](#)

PFMatrix, [2](#)

PFMatrixList, [2](#), [3](#)