Package 'JohnsonKinaseData'

July 16, 2024

Title Kinase PWMs based on data published by Johnson et al. 2023 (doi:10.1038/s41586-022-05575-3)

Version 1.0.0

Date 2023-09-19

Description The packages provides position specific weight matrices (PWMs) for 303 human serine/threonine kinases originally published in Johnson et al. 2023. It includes gene annotation for each kinase PWM and PWM matching scores for a set of 85603 curated human phosphosites which can be used to map a PWM score to its percentile rank. The package also includes basic functionality to score user provided phosphosites.

License MIT + file LICENSE

URL https://github.com/fgeier/JohnsonKinaseData/

BugReports https://support.bioconductor.org/t/JohnsonKinaseData

Imports ExperimentHub, BiocParallel, checkmate, dplyr, stats, stringr, tidyr, purrr, utils

Suggests knitr, BiocStyle, ExperimentHubData, testthat (>= 3.0.0), rmarkdown

biocViews ExperimentHub, Homo_sapiens_Data, Proteome

VignetteBuilder knitr

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.1

Config/testthat/edition 3

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

git_branch RELEASE_3_19

git_last_commit d9a7bcc

git_last_commit_date 2024-04-30

Repository Bioconductor 3.19

Date/Publication 2024-07-16

Author Florian Geier [aut, cre] (<https://orcid.org/0000-0002-9076-9264>)

Maintainer Florian Geier <florian.geier@unibas.ch>

Contents

`JohnsonKinaseData-package`
JohnsonKinaseData

Description

The **JohnsonKinaseData** package provides position specific weight matrices (PWMs) for 303 human serine/threonine kinases originally published in Johnson et al. 2023.

It also includes pre-computed PWM scores ("background scores") for a large collection of curated human phosphosites.

The package additionally offers functionality to match kinase PWMs against user-provided phosphopeptides and to rank PWM scores relative to the background scores ("percentile rank").

Author(s)

Florian Geier (florian.geier@unibas.ch)

References

Johnson, J.L., Yaron, T.M., Huntsman, E.M. et al. An atlas of substrate specificities for the human serine/threonine kinome. Nature 613, 759–766 (2023). https://doi.org/10.1038/s41586-022-05575-3

Yaffe, M., Leparc, G., Lai, J. et al. A motif-based profile scanning approach for genome-wide prediction of signaling pathways. Nat Biotechnol 19, 348–353 (2001). https://doi.org/10.1038/86737

See Also

Useful links:

- https://github.com/fgeier/JohnsonKinaseData/
- Report bugs at https://support.bioconductor.org/t/JohnsonKinaseData

getBackgroundScores *Get precomputed PWM scores for a large set of curated human phosphosites*

Description

The background scores are derived from matching each PWM to the 85'603 unique phosphosites published in Johnson et al. 2023. The data frame contains the log2-odds score per phosphosite and PWM.

Usage

```
getBackgroundScores()
```

Value

A data frame with log2-odds scores per phosphosite (rows) and PWMs (columns)

Examples

```
bg <- getBackgroundScores()
```

getKinaseAnnotation *Get annnotation data for all 303 human serine/threonine kinase PWMs*

Description

The annotation data records for each of the 303 human serine/threonine kinase PWMs originally published in Johnson et al. the PWM matrix name, gene symbol and description, Uniprot ID and Entrez ID as well as the kinase family.

Usage

```
getKinaseAnnotation()
```

Value

A data frame with columns MatrixName, GeneName, UniprotID, EntrezID, Description and KinaseFamily

Examples

```
anno <- getKinaseAnnotation()
```

`getKinasePWM` *Get a list of position specific weight matrices (PWMs)*

Description

The function returns a list of PWMs for the 303 human serine/threonine kinases originally published in Johnson et al. 2023. Each PWM stores the log2-odds score per amino acid (23 rows) and position (10 columns) in matrix format. Beside the 20 standard amino acids also phosphorylated serine, threonine and tyrosine residues are included.

Usage

```
getKinasePWM(includeSTfavorability = TRUE)
```

Arguments

`includeSTfavorability`
: Include serine vs. threonine favorability for the central phospho-acceptor?

Details

The central phospho-acceptor position of each PWM is either serine or threonine. By default, this position quantifies the favorability of serine over threonine. This favorability can be omitted when setting 'includeSTfavorability=FALSE' in which case the central position doesn't contribute to the PWM score.

Value

A named list of numeric matrices (PWMs).

References

Johnson, J.L., Yaron, T.M., Huntsman, E.M. et al. An atlas of substrate specificities for the human serine/threonine kinome. Nature 613, 759–766 (2023). https://doi.org/10.1038/s41586-022-05575-3

Examples

```
pwms <- getKinasePWM()
```

`getScoreMaps` *Map log2-odds score to percentile rank*

Description

For each kinase PWM, get a function that maps its log2-odds score to the percentile rank in the background score distribution. The percentile rank of a given score is the percentage of scores in corresponding background score distribution that are less than or equal to that score. The background score distribution per PWM is derived from matching each PWM to the 85'603 unique phosphosites published in Johnson et al. 2023.

Usage

```
getScoreMaps()
```

Details

The background sites used by Johnson et al. don't contain non-central phosphorylated residues (phospho-priming). Therefore any input sites which include phospho-priming will be capped to 100 percentile rank, if their PWM score exceeds the largest observed background score for that PWM.

Internally, stats::approxfun is used to linearly interpolate between the PWM score and its 0.1% - quantile in the distribution over background scores. This approximation allows for a lower memory footprint compared with the full set of background scores.

Value

A named list of functions, one for each kinase PWM. Each function is taking a vector of PWM log2-odds scores and maps them to a percentile rank in the range 0 to 100.

Examples

```
maps <- getScoreMaps()
```

`JohnsonKinaseAnnotation`

Get file path to local cache of annotation data for kinase PWMs

Description

Internal cache accessor function.

Usage

```
JohnsonKinaseAnnotation(metadata=FALSE)
```

Arguments

metadata	When 'metadata=FALSE' the full resource is loaded, when 'metadata=TRUE' just the metadata are displayed.

Value

A character vector with the file path to the local cache of the annotation data.

See Also

Use `getKinaseAnnotation()` to load the data.

Examples

```
anno <- JohnsonKinaseAnnotation()
```

JohnsonKinaseBackgroundQuantiles

Get file path to local cache of background score quantiles

Description

Internal cache accessor function.

Usage

```
JohnsonKinaseBackgroundQuantiles(metadata=FALSE)
```

Arguments

metadata	When 'metadata=FALSE' the full resource is loaded, when 'metadata=TRUE' just the metadata are displayed.

Value

A character vector with the file path to the local cache of background score quantiles.

See Also

Use `getScoreMaps()` to load the score mapping functions.

Examples

```
quants <- JohnsonKinaseBackgroundQuantiles()
```

JohnsonKinaseBackgroundScores *Get file path to local cache of background scores*

Description

Internal cache accessor function.

Usage

```
JohnsonKinaseBackgroundScores(metadata=FALSE)
```

Arguments

metadata	When 'metadata=FALSE' the full resource is loaded, when 'metadata=TRUE' just the metadata are displayed.

Value

A character vector with the file path to the local cache of the background scores.

See Also

Use `getBackgroundScores()` to load the data.

Examples

```
scores <- JohnsonKinaseBackgroundScores()
```

JohnsonKinasePWM *Get file path to local cache of kinase PWM data*

Description

Internal cache accessor function.

Usage

```
JohnsonKinasePWM(metadata=FALSE)
```

Arguments

metadata	When 'metadata=FALSE' the full resource is loaded, when 'metadata=TRUE' just the metadata are displayed.

Value

A character vector with the file path to the local cache of the PWM data.

See Also

Use `getKinasePWM()` to load the data.

Examples

```
data <- JohnsonKinasePWM()
```

`processPhosphopeptides` *processPhosphopetides*

Description

Process phospho-peptides to a common format used for PWM matching

Usage

```
processPhosphopeptides(
  sites,
  onlyCentralAcceptor = TRUE,
  allowPhosphoPriming = TRUE
)
```

Arguments

`sites`	Character vector with phospho-peptides
`onlyCentralAcceptor`	Process only the central phospho-acceptor residue?
`allowPhosphoPriming`	Allow phospho-acceptors at non-central positions? These should be indicated by the lower case letters s, t or y.

Details

Phosphorylated residues are recognized either by lower case letters (s, t or y) or the phosphorylated residue is followed by an asterisk (S*, T* or Y*).

If a peptide reports several phosphorylated residues, parameter `onlyCentralAcceptor` allows for two processing options: (1) By default, only the central phospho-acceptor of each phospho-peptide is considered. Here central is defined as the left-closest position to `floor(nchar(site)/2)+1`. (2) All phospho-acceptors are considered as central in which case the phospho-peptide is replicated and aligned for each of its phosphorylated residues. In this case the output sites are not in parallel to the input peptides.

In both cases, non-central phospho-acceptors are indicated by lower case letters (s, t, or y). These residues enable phospho-priming of the site. If phospho-priming is disabled (parameter `allowPhosphoPriming`) these residues are converted to upper case letters.

If a site does not follow the phosphorylation patterns described above, the central residue defined by position `floor(nchar(site)/2)+1` is considered the default phospho-acceptor site.

The input sites are truncated and/or padded such that the processed sites are of width 10 and have the central phospho-acceptor surrounded by 5 upstream and 4 downstream residues, as required for PWM matching.

A warning is raised if the central phospho-acceptor is not serine or threonine, as these sites are not covered by the Johnson PWMs.

Value

A tibble with columns: `sites`, `processed`, `acceptor`

Examples

```
procSites <- processPhosphopeptides(c("SAGLLS*DEDC", "RtEKGS*N", "EKGDSN__"))
```

`scorePhosphosites` *Match kinase PWMs to processed phosphosites*

Description

`scorePhosphosites` takes a list of kinase PWMs and a vector of processed phosphosites as input and returns a matrix of match scores per PWM and site.

Usage

```
scorePhosphosites(
  pwms,
  sites,
  scoreType = c("lod", "percentile"),
  BPPARAM = BiocParallel::SerialParam()
)
```

Arguments

`pwms`	List with kinase PWMs as returned by getKinasePWM.
`sites`	A character vector with phosphosites. Check processPhosphopeptides for the correct phosphosite format.
`scoreType`	Log2-odds score or percentile rank.
`BPPARAM`	A BiocParallelParam object specifying how parallelization should be performed.

Details

The match score is either the log2-odds score (`lod`) or the percentile rank (`percentile`) in the background score distribution.

Value

A numeric matrix of size `length(sites)` times `length(pwms)`.

See Also

getKinasePWM for getting a list of kinase PWMs, processPhosphopeptides for the correct phosphosite format, and getScoreMaps for mapping PWM scores to percentile ranks

Examples

```
score <- scorePhosphosites(getKinasePWM(), c("TGRRHTLAEV", "LISAVSPEIR"))
```

Index