Package 'RforProteomics'

October 16, 2018

Type Package

Title Companion package to the 'Using R and Bioconductor for proteomics data analysis' publication

Version 1.18.1

Maintainer Laurent Gatto <lg390@cam.ac.uk>

Depends MSnbase (>= 2.5.3)

Imports R.utils, Biobase, rpx, biocViews, BiocInstaller, interactiveDisplay, DT, AnnotationDbi, shiny

Suggests knitr, rmarkdown, BiocStyle, mzR, xcms, msdata, isobar, MALDIquant (>= 1.12), MALDIquantForeign, readBrukerFlexData, rTANDEM, synapter, synapterdata, IPPD, Rdisop, OrgMassSpecR, SummarizedExperiment, BRAIN, rols, hpar, GO.db, org.Hs.eg.db, e1071, biomaRt, RColorBrewer, ggplot2, reshape2, xtable, lattice, mzID, pRoloc, pRolocdata, MSGFplus, MSGFgui, MSnID, msmsTests, msmsEDA, DEP, corrplot, beanplot, Heatplus, gplots, VennDiagram, protViz, genefilter

Enhances cleaver

Description This package contains code to illustrate the 'Using R and Bioconductor for proteomics data analysis' and 'Visualisation of proteomics data using R and Bioconductor' manuscripts. The vignettes describe the code and data needed to reproduce the examples and figures described in the paper and functionality for proteomics visualisation. It also contain various function to discover R software for mass spectrometry and proteomics.

URL http://lgatto.github.com/RforProteomics/

biocViews ExperimentData, MassSpectrometryData, ReproducibleResearch

License Artistic-2.0

VignetteBuilder knitr

Author Laurent Gatto [aut, cre], Thomas Lin Pedersen [ctb], Sebastian Gibb [ctb], Vlad Petyuk [ctb]

RoxygenNote 5.0.1

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

git_branch RELEASE_3_7

git_last_commit 765757e

git_last_commit_date 2018-05-23

Date/Publication 2018-10-16

R topics documented:

`downloadData` *Download a file*

Description

Unless already present, downloads `src` in the `destdir` directory.

Usage

```
downloadData(src, destdir = ".", unpack = TRUE, ...)
```

Arguments

`src`	The url of the file to download.
`destdir`	The destination directory. Default is `"."`.
`unpack`	Should `src` be uncompressed? Default is `TRUE`.
`...`	Additional paramters passed to `download.file`.

Value

Invisible returns the full path of the downloaded file.

Author(s)

Laurent Gatto

getPackagesInBiocView *Packages in a biocView*

Description

Finds the package names that have a specific biocView.

Usage

```
getPackagesInBiocView(view, rep = c("BioCsoft", "BioCann", "BioCexp",
  "BioCextra"), biocVersion)
```

Arguments

view	The biocView of interest. For example "Proteomics".
rep	Repository of interest. One of "BioCsoft", "BioCann", "BioCexp" or "BioCextra".
biocVersion	A character with the Bioconductor version of interest. For example "2.14".

Value

An instance of class BiocView. NULL if the the biocView was not found.

Author(s)

Laurent Gatto

getPXD000001mzData *Download the PXD000001 mzTab file*

Description

Unless already present, downloads the PXD000001 mzData file in the destdir directory. The resulting file is named PRIDE_Exp_Complete_Ac_22134.xml

Usage

```
getPXD000001mzData(destdir = ".")
```

Arguments

destdir	A character with the destination folder.

Value

Invisibly returns the name of the downloaded file.

Author(s)

Laurent Gatto

`getPXD000001mzTab` *Download the PXD000001 mzTab file*

Description

Unless already present, downloads the PXD000001 mzTab file in the `destdir` directory. The resulting file is named `F063721.dat-mztab.txt`.

Usage

```
getPXD000001mzTab(destdir = ".")
```

Arguments

`destdir`	A `character` with the destination folder.

Value

Invisibly returns the name of the downloaded file.

Author(s)

Laurent Gatto

`getPXD000001mzXML` *Download the PXD000001 mzXML file*

Description

Unless already present, downloads the PXD000001 mzXML file in the `destdir` directory. The resulting file is named `TMT_Erwinia_1uLSike_Top10HCD_isol2_45stepped_60min_01.mzXML`.

Usage

```
getPXD000001mzXML(destdir = ".")
```

Arguments

`destdir`	A `character` with the destination folder.

Value

Invisibly returns the name of the downloaded file.

Author(s)

Laurent Gatto

getThermoHelaPRTC *Dowload Thermo Hela PRTC data*

Description

Downloads on of multiple Thermo Hela/PRTC data files.

Usage

```
getThermoHelaPRTC(src, destdir = ".")
```

Arguments

src	The name of the file to be downloaded. If missing, a vector of possible filenames is returned. If "all", all files are downloaded. Alternatively, a pattern can be used to grep the files from the output getThermoHelaPRTC() the files to be downloaded.
destdir	Destination directory. Default is ".".

Value

Invisibly return the path of the downloaded files.

Author(s)

Laurent Gatto

See Also

downloadData

Examples

```
getThermoHelaPRTC()
getThermoHelaPRTC("design")
## Not run:
getThermoHelaPRTC("all")

## End(Not run)
```

id *An* mzIdentML *file*

Description

This file has been generated by searching the raw mzXML file of the ProteomeXchange PXD000001
data set against the erwinia_carotovora.fasta using the MSGF+ search engine: java -jar ~/bin/MSGFPlus.20140
TMT_Erwinia_1uLSike_Top10HCD_isol2_45stepped_60min_01.mzXML -d erwinia_carotovora.fasta -inst 1

Examples

```
## source files to repeat the search
library("rpx")
px <- PXDataset("PXD000001")
pxfiles(px)

f <- dir(system.file("extdata", package = "RforProteomics"),
         pattern = "mzid", full.names=TRUE)
library("mzID")
id <- mzID(f)
id
```

`new_ions` *Create, analyse and detect ions*

Description

This is the constructor function to generate a set of ions that can later be analysed with 'analyse()' and detected with 'detect()'.

Usage

```
new_ions(npeaks = 10, mzrange = c(100, 1000), nimg = 100)

analyse(x, sleep = 0.1)

analyze(x, sleep = 0.1)

detect(x, new = FALSE)

spectrum(x, ...)
```

Arguments

`npeaks`	A 'numeric' scalar defining the number of unique peaks (M/Z values). Default is 10.
`mzrange`	A 'numeric' of length 2 defining the range of possible M/Z values. Default is 'c(100, 1000)'.
`nimg`	A 'numeric' scalar. When analysing the ions, their separation along their M/Z values will be split along a sequence of length 'nimg'. Default is 100.
`x`	An object of class 'ions'.
`sleep`	How much time to wait before producing the next plot.
`new`	A 'logical' scalar, indicating if the separated ions (last frame of calling 'analyse) should be plotting, or whether the detection should be overlaid. Default is 'FALSE', to add the plot on top of the opened device.
`...`	Additional arguments passed to [graphics::plot()].

Value

An object of class ‘ions‘.

‘analyse‘, ‘detect‘ and ‘spectrum‘ are used for their side effect or producing plots. They all invisibly return ‘NULL‘.

Author(s)

Laurent Gatto

Examples

```
set.seed(1L)
x <- new_ions(nimg = 5)
x
analyse(x)
detect(x)
spectrum(x)
```

---

packageDF *Package descriptions*

Description

Format a BiocView as a data.frame.

Usage

```
packageDF(x, nsub = TRUE, version = TRUE)
```

Arguments

x	An instance of class BiocView, as produced by getPackagesInBiocView.
nsub	A logical indicating "\n" are to be replaced by a space.
version	A logical specifying if the package version should be added.

Value

A data.frame with package information.

Author(s)

Laurent Gatto

proteomicsPackages *Proteomics and MS biocView packages*

Description

Searches for all the packages with the "Proteomics" (software), "MassSpectrometry" (software) and "MassSepctrometryData" (data) packages and return their names, titles and versions as a data.frame. The (unexported but documented) underlying functions are RforProteomics:::getPackagesInBiocView (to find relevant package) and RforProteomics:::packageDF (data.frame formatting).

Usage

```
proteomicsPackages(biocv, cache=FALSE)
massSpectrometryPackages(biocv, cache=FALSE)
massSpectrometryDataPackages(biocv, cache=FALSE)
```

Arguments

biocv	A character with the Bioconductor version to search for relevant packages. If missing, the running version is used.
cache	A logical indicating whether cached package information should be used. Default is FALSE. All except development versions are up-to-date.

Value

A data.frame with the respective package names, titles and versions.

Author(s)

Laurent Gatto

Examples

```
head(pp <- proteomicsPackages("3.0"))
ppc <- proteomicsPackages("3.0", cache = TRUE)
all.equal(pp, ppc)
```

qnt *PXD000001 example MSnSet*

Description

In this TMT 6-plex experiment, four exogenous proteins were spiked into an equimolar *Erwinia carotovora* lysate with varying proportions in each channel of quantitation; yeast enolase (ENO) at 10:5:2.5:1:2.5:10, bovine serum albumin (BSA) at 1:2.5:5:10:5:1, rabbit glycogen phosphorylase (PHO) at 2:2:2:2:1:1 and bovin cytochrome C (CYT) at 1:1:1:1:1:2. Proteins were then digested, differentially labelled with TMT reagents, fractionated by reverse phase nanoflow UPLC (nanoACQUITY, Waters), and analysed on an LTQ Orbitrap Velos mass spectrometer (Thermo Scientific). Files in multiple format will be used to illustrate the input/output capabilities that are available to the

proteomics audience. The companion package provides dedicated functions to directly download the data.

The data has been downloaded from the ProteomeXchange repository and imported into R as illustrated in the example. It is of class MSnSet. See also the MSnbase-demo vignette for more details.

Usage

```
data("qnt")
```

Format

An instance of class MSnSet

References

Laurent Gatto (2014). RforProteomcs: Companion package to the 'Using R and Bioconductor for proteomics data analysis' publicationR package version 1.3.1.

Gatto L, Christoforou A. Using R and Bioconductor for proteomics data analysis. Biochim Biophys Acta. 2013 May 18. doi:pii: S1570-9639(13)00186-6. 10.1016/j.bbapap.2013.04.032. [Epub ahead of print] PubMed PMID: 23692960.

Examples

```
## Not run:
    library("rpx")
    px1 <- PXDataset("PXD000001")
    mztab <- pxget(px1, "PXD000001_mztab.txt")
    library("MSnbase")
    qnt <- readMzTabData(mztab, what = "PEP")
    sampleNames(qnt) <- reporterNames(TMT6)
    qnt$conditions <- factor(c("A", "A", "B", "B", "B", "A"))
    qnt <- filterNA(qnt)

    selA <- qnt$conditions == "A"

    fData(qnt)$log2FC <-
        log(rowMeans(exprs(qnt)[, selA]), 2) -
            log(rowMeans(exprs(qnt)[, !selA]), 2)
    fData(qnt)$baseMean <- log(rowMeans(exprs(qnt)), 10)

## End(Not run)

library("RforProteomics")
library("MSnbase")
data(qnt)
class(qnt)
head(exprs(qnt))
head(fData(qnt))
```

RforProteomics *Opens the RforProteomics vignette*

Description

Opens the package vignettes.

Usage

```
RforProteomics()
```

Value

An instance of class `vignette`. Used for its side effect, opening the vignette.

Author(s)

Laurent Gatto

RProtVis *Opens the visualisation vignette*

Description

Opens the visualisation vignette

Usage

```
RProtVis()
```

Value

An instance of class `vignette`. Used for its side effect, opening the vignette.

Author(s)

Laurent Gatto

shinyMA *MA and expression plots in* shiny

Description

Starts an interactive shiny application that displays an MA plot and an expression plot side by side. The user can select features of interest on the MAplot and the respective intensities are displayed on the expression plot on the right.

The data has been prepared using the mzTab file from the ProteomeXchange spiked-in data PXD000001 (see qnt for details). Sample 1, 2, 6 and 3, 4, 5 have been arbitrarily chosen to define two groups.

Usage

```
shinyMA()
```

Value

Used for its side effects of starting a shiny application.

Author(s)

Laurent Gatto <lg390@cam.ac.uk>

References

The application is an adaptation of Michael Love's shinyMA app available on https://github.com/mikelove/shinyMA.

Examples

```
if (interactive())
    shinyMA()
```

Index