Package ‘MWASTools’

February 1, 2017

Type Package

Title MWASTools: an integrated pipeline to perform metabolome-wide association studies

Version 0.99.15

Date 2016-12-19

Author Andrea Rodriguez-Martinez, Joram M. Posma, Rafael Ayala, Ana L. Neves, Jeremy K. Nicholson, Marc-Emmanuel Dumas

Maintainer Andrea Rodriguez-Martinez <andrea.rodriguez-martinez13@imperial.ac.uk>, Rafael Ayala <r.ayala14@imperial.ac.uk>

Description MWAS provides a complete pipeline to perform metabolome-wide association studies. Key functionalities of the package include: quality control analysis of metabolomic data; MWAS using different association models (partial correlations; generalized linear models); model validation using non-parametric bootstrapping; visualization of MWAS results; NMR metabolite identification using STOCSY.

License CC BY-NC-ND 4.0

Depends R(>= 3.4)

Suggests RUnit, BiocGenerics, knitr, BiocStyle, rmarkdown

VignetteBuilder knitr

Imports glm2, ppcor, qvalue, car, boot, grid, ggplot2, gridExtra, igraph, SummarizedExperiment, graphics, stats, utils

biocViews Metabolomics, Lipidomics, Cheminformatics, SystemsBiology, QualityControl

NeedsCompilation no

LazyData true

Encoding UTF-8

R topics documented:

CV_filter ................................................................. 2
metabo_SE ............................................................ 3
MWAS_barplot ....................................................... 3
MWAS_bootstrapping ............................................... 4
MWAS_filter .......................................................... 5
Filter metabolic data by CV

Description

This function allows filtering a matrix of metabolic variables based on the coefficient of variation (CV) of each variable across the quality control (QC) samples. See also function "QC_CV()".

Usage

CV_filter(metabo_SE, CV_metabo, CV_th = 0.30)

Arguments

- metabo_SE: SummarizedExperiment object. See "MWAS_SummarizedExperiment()".
- CV_metabo: numeric vector containing the CVs of the metabolic variables. See function "QC_CV()".
- CV_th: numeric value indicating the CV threshold. Only features with CV below CV_th will be retained in the matrix.

Value

A SummarizedExperiment object containing the CV-filtered metabolic_data.

References


Examples

```r
## Load data
data(metabo_SE)

## Calculate CVs
CV_metabo <- QC_CV(metabo_SE)

## Filter metabolic_data by CV
```
metabo_SE

metabo_CVfiltered <- CV_filter(metabo_SE, CV_metabo, CV_th = 0.30)
metabo_CVfiltered2 <- CV_filter(metabo_SE, CV_metabo, CV_th = 0.15)

metabo_SE  
\textbf{NMR plasma metabolic profiles dataset}

\textbf{Description}

This SummarizedExperiment object contains the following information:

- An assay matrix containing the 1H NMR profiles (1.60 - 0.80 ppm) of 506 plasma samples from the FGENTCARD cohort and 10 identical quality control (QC) samples. The QC samples were prepared from a representative pool of the experimental samples, and were injected regularly throughout the run to ensure analytical reproducibility.

- A data.frame containing clinical information (age, gender, type II diabetes status and BMI) and sample class (i.e. experimental sample or QC sample) information for each sample row in the assay matrix.

\textbf{Usage}

data(metabo_SE)

\textbf{Format}

SummarizedExperiment

\textbf{Value}

SummarizedExperiment

\textbf{MWAS_barplot}

\textbf{Visualize MWAS results in a bar plot}

\textbf{Description}

This function creates a bar plot based on the output from "MWAS_stats()". This function is designed to visualize MWAS results in the case of discrete metabolic variables (e.g. target GC/MS metabolites).

\textbf{Usage}

MWAS_barplot(MWAS_matrix, alpha_th = 0.05, width = NULL, scale_color = c("darkgray", "cornflowerblue", "firebrick1"), legend_labs =c("unchanged","downregulated","upregulated"), ylab = "sign*log(pFDR)", size_yaxis = 12, size_ylab = 12, size_names = 10, angle_names = 45, sort = TRUE)
MWAS_bootstrapping

Arguments

MWAS_matrix numeric matrix resulting from the function "MWAS_stats()".
alpha_th numeric value indicating the significance threshold.
width numeric value indicating bar width.
scale_color character vector corresponding to the 3-color scale that will be used to represent the association results. The first color of the scale indicates "no change", the second color indicates "downregulation", and the third color indicates "upregulation".
legend_labs character vector containing the legend labels, according to scale_color.
ylab character vector specifying a title for the y-axis.
size_yaxis numeric value indicating the font size of y-axis title.
size_ylab numeric value indicating the font size of y-axis labels.
size_names numeric value indicating the font size of the metabolite ids displayed on the x-axis.
angle_names numeric value indicating the angle in which the metabolite ids will be displayed on the x-axis.
sort logical constant indicating whether the metabolites will be sorted based on MWAS results.

Value

A bar plot.

Examples

## Load data
data(targetMetabo_SE)

## Test for association between diabetes and target_metabolites
T2D_model <- MWAS_stats(targetMetabo_SE, disease_id = "T2D",
                        confounder_ids = c("Age", "Gender", "BMI"),
                        assoc_method = "logistic")

## Bar plot
MWAS_barplot(T2D_model)
MWAS_barplot(T2D_model, width = 0.7) # change bar width
MWAS_barplot(T2D_model, width = 0.7, angle_names = 90)

MWAS_bootstrapping

MWAS bootstrap resampling

Description

This function generates bootstrap replicates (non-parametric resampling) of a model testing for association between a given metabolite and a disease phenotype, and calculates the confidence interval of model coefficients.
**MWAS_bootstrapping**

Usage

```r
MWAS_bootstrapping (metabo_SE, metabolite_id, disease_id, confounder_ids = NULL, assoc_method, iterations = 10000)
```

Arguments

- `metabo_SE`: SummarizedExperiment object. See "MWAS_SummarizedExperiment()".
- `metabolite_id`: character vector corresponding to the id of the metabolite to be modeled.
- `disease_id`: character vector corresponding to the id of the response to be modeled.
- `confounder_ids`: optional character vector corresponding to the ids of covariates to be included in the model (e.g., age or gender).
- `assoc_method`: character constant indicating the association method that will be used. Possible values for assoc_method are: "pearson" (pearson correlation), "spearman" (spearman correlation), "kendall" (kendall correlation), "linear" (linear regression) or "logistic" (logistic regression).
- `iterations`: numeric value indicating the number of bootstrap replicates.

Value

A list with 3 elements, each list element reporting the following information: i) object of class "boot"; ii) summary of the previous object; iii) 95-confidence interval of the metabolite model coefficient. For more details, check the function "boot()" from the "boot" package.

**References**


**Examples**

```r
## Load data
data(targetMetabo_SE)

## Bootstrap model testing for association between diabetes (T2D) and 3OH-butyrate
MWAS_bootstrapping (targetMetabo_SE, metabolite_id = "3OH-butyrate", disease_id = "T2D", assoc_method = "logistic", iterations = 1000)
```

**MWAS_filter**

Filter MWAS results by p-value and/or CV

Description

This function allows filtering the output matrix from "MWAS_stats()", by p-value and/or coefficient of variation (CV).

Usage

```r
MWAS_filter(MWAS_matrix, type = "pvalue", alpha_th = 0.05, CV_th = 0.30)
```
Arguments

MWAS_matrix
numeric matrix generated by the function "MWAS_stats()".

type
character constant indicating the filtering criteria. If type = "pvalue", only metabolic variables with p-value below alpha_th will be retained in the MWAS_matrix. If type = "CV", only metabolic variables with CV below CV_th will be retained. If type = "all", only metabolic variables with CV below CV_th and p-value below alpha_th will be retained.

alpha_th
numeric value indicating the significance threshold.

CV_th
numeric value indicating the CV threshold.

Value

A numeric matrix corresponding to the filtered MWAS_matrix. The matrix has an additional column, which indicates the index of each metabolic variable in the original MWAS_matrix.

Examples

```r
## Load data
data(targetMetabo_SE)

## Test for association between diabetes and target_metabolites
T2D_model <- MWAS_stats (targetMetabo_SE, disease_id = "T2D",
                        assoc_method = "logistic")

## Filter T2D_model by p-value
pvalue_filter <- MWAS_filter(T2D_model, type = "pvalue", alpha_th = 0.001)

## Subset targetMetabo_SE based on pvalue_filter
index_features <- pvalue_filter[, 4]
targetMetabo_SE[index_features, ]
```

Description

This function allows visualizing MWAS results in a correlation-based metabolic network. The network is an undirected graph where the nodes represent the metabolites, and the edges represent a co-abundance relationship between pairs of nodes. Different node parameters (e.g. color, size) can be customized based on MWAS results.

Usage

```r
MWAS_network (metabo_SE, MWAS_matrix, alpha_th = 0.05, cor_th = 0.25,
               file_name = "MWAS", res_cor = 2)
```
Arguments

- `metabo_SE`: SummarizedExperiment object. See "MWAS_SummarizedExperiment()".
- `MWAS_matrix`: numeric matrix generated by the function "MWAS_stats()".
- `alpha_th`: numeric value indicating MWAS significance threshold.
- `cor_th`: numeric value indicating the co-abundance similarity threshold. Thus, two metabolites will be linked in the network if the absolute correlation (Pearson) between them exceeds `cor_th`.
- `file_name`: character string indicating the name given to the cytoscape files that will be exported to the working directory.
- `res_cor`: numeric value restricting the number of decimals of the correlation of coefficients used to build the edges of the network.

Value

A correlation based-metabolic network formalized as a weighted igraph object. This igraph object contains two node attributes: "score" and "color". "score" is a vector containing the MWAS score (-log10(pvalue)*estimate sign) of each metabolite. "color" is a vector indicating the color of each node based on MWAS results ("cornflowerblue": "downregulation", "gray":"no change", "firebrick1":"upregulation"). These attributes can be used to customize node parameters based on MWAS results. The function also exports a network file ("MWASNetwork.txt") and an attribute file ("MWASAttribute.txt") of MWAS scores, which can be imported into cytoscape to visualize the network.

References


Examples

```r
## Load data
data(targetMetabo_SE)

## Test for association between diabetes and target_metabolites
T2D_model <- MWAS_stats (targetMetabo_SE, disease_id = "T2D",
                        assoc_method = "logistic")

## Build correlation-based metabolic network
net_T2D <- MWAS_network(targetMetabo_SE, T2D_model, file_name = "MWAS_T2D",
                        cor_th = 0.30)

## Visualize network using the igraph package
# library(igraph)
# plot(net_T2D, vertex.size = abs(V(net_T2D)$score*6)) # node size based on scores
# plot(net_T2D, vertex.size = abs(V(net_T2D)$score*6),
#     edge.label = E(net_T2D)$weight) # show edge labels
```
MWAS_skylineNMR  

Visualize MWAS results in an NMR-skyline plot

Description

This function generates a 2-panel figure showing the results from "MWAS_stats()" applied to NMR data. The upper panel shows an NMR-skyline plot (comparable to a GWAS-Manhattan plot), where the chemical shifts are displayed along the x-axis and the -log10 p-values (sign-adjusted for the direction of the association) are displayed on the y-axis. The lower panel shows an NMR spectrum colored according to MWAS results.

Usage

MWAS_skylineNMR (metabo_SE, MWAS_matrix, ref_sample, alpha_th = 0.05, output = "all", xlab = "ppm", ylab1 = "sign*log(pFDR)", ylab2 = "intensity", pch = 20, marker_size = 1, scale_color = c("black", "cornflowerblue", "red"), size_lab = 12, size_axis = 12, xlim = NULL, ylim1 = NULL, ylim2 = NULL, guide_type = "legend", xbreaks = waiver(), xnames = waiver(), ybreaks1 = waiver(), ybreaks2 = waiver(), ynames1 = waiver(), ynames2 = waiver())

Arguments

metabo_SE  
SummarizedExperiment object. See "MWAS_SummarizedExperiment()".

MWAS_matrix  
numeric matrix resulting from the function "MWAS_stats()".

ref_sample  
character vector indicating the id of the sample that will be used to plot the NMR spectrum.

alpha_th  
numeric value indicating the significance threshold.

output  
character constant indicating the outcome of the function ("skyline", "spectrum" or "all"). If outcome = "all", both the skyline and the spectrum with be plotted in a 2-panel plot.

xlab  
character vector specifying a title for the x-axis.

ylab1  
character vector specifying a title for the y-axis of the upper panel.

ylab2  
character vector specifying a title for the y-axis of the lower panel.

pch  
value specifying the symbol used to represent each ppm value in the skyline plot. To see all possible symbols, check "plot()" options.

marker_size  
numeric value indicating the size of the symbol used to represent each ppm value in the skyline plot.

scale_color  
character vector corresponding to the 3-color scale that will be used to represent the association results. The first color of the scale indicates "no change", the second color indicates "downregulation", and the third color indicates "upregulation".

size_lab  
numeric value indicating the font size of x- and y-axis titles.

size_axis  
numeric value indicating the font size of x- and y-axis labels.

xlim  
numeric vector containing the minimum and maximum values of the x-axis. Notice that ppm is displayed in reverse scale (e.g. xlim = c(5, 2)).
**MWAS_stats**

ylim1 numeric vector containing the minimum and maximum values of the y-axis for the upper panel.

yylim2 numeric vector containing the minimum and maximum values of the y-axis for the lower panel.

guide_type character constant indicating the guide ("legend" or "none") that will be added to the plots.

xbreaks numeric vector indicating the positions of the breaks of the x-axis.

xnames character vector (same length as xbreaks) containing the labels of each break of the x-axis.

ybreaks1 numeric vector indicating the positions of the breaks of the y-axis for the upper panel.

ybreaks2 numeric vector indicating the positions of the breaks of the y-axis for the lower panel.

ynames1 character vector (same length as ybreaks1) containing the labels of each break of the y-axis for the upper panel.

ynames2 character vector (same length as ybreaks2) containing the labels of each break of the y-axis for the lower panel.

**Value**

By default, a plot with 2 panels, the upper panel showing an NMR-skyline plot and the lower panel showing an NMR spectrum colored based on MWAS results.

**References**


**Examples**

```r
## Load data
data(metabo_SE)

## Test for association between BMI and metabolic_data
BMI_model <- MWAS_stats (metabo_SE, disease_id = "BMI", assoc_method = "spearman", output = "pvalues")

## Create skyline plots
MWAS_skylineNMR (metabo_SE, BMI_model, ref_sample = "QC1")
MWAS_skylineNMR (metabo_SE, BMI_model, ref_sample = "QC1", pch = "*", marker_size = 3)
```

---

**MWAS_stats Metabolome-Wide Associations**

**Description**

This function tests for association between individual metabolites and a disease phenotype.
Usage

MWAS_stats (metabo_SE, disease_id, confounder_ids = NULL, assoc_method, mt_method = "BH", output = "pvalues", CV_metabo = NULL)

Arguments

- **metabo_SE**: SummarizedExperiment object. See "MWAS_SummarizedExperiment()".
- **disease_id**: character vector corresponding to the id of the response to be modeled.
- **confounder_ids**: optional character vector corresponding to the ids of the covariates to be included in the model (e.g. age or gender).
- **assoc_method**: character constant indicating the association method that will be used. Possible values for assoc_method are: "pearson" (Pearson correlation), "spearman" (Spearman correlation), "kendall" (Kendall correlation), "linear" (linear regression) or "logistic" (logistic regression).
- **mt_method**: character constant indicating the multiple-testing correction method that will be used. Possible values for mt_method are: "BH" (Benjamini and Hochberg), "bonferroni", "holm", "hochberg", "hommel", "BY" (Benjamini and Yekutieli), "qvalues", or "none".
- **output**: character constant indicating the output of the function. If output = "pvalues", p-values and estimates for each metabolic variable will be returned as a matrix. If output = "models", detailed information about the statistical model fitted for each metabolic variable will be returned.
- **CV_metabo**: optional numeric vector containing the coefficients of variation of the metabolic variables. This vector will be added as an additional column of the output matrix.

Value

By default, a matrix where each row contains the model coefficient estimate and the p-value obtained for each metabolic variable. When output = "models", the function returns a list, each list element containing detailed information about the statistic model fitted for each metabolic variable.

References


Examples

```r
# Load data
data(metabo_SE)
data(targetMetabo_SE)

# Test for association between BMI and metabolic_data
BMI_model <- MWAS_stats(metabo_SE, disease_id = "BMI", assoc_method = "spearman",
                          mt_method = "BH", output = "pvalues")

# Test for association between diabetes and target_metabolites (age-gender adjusted)
T2D_model <- MWAS_stats(targetMetabo_SE, disease_id = "T2D",
                        confounder_ids = c("Age", "Gender"),
                        assoc_method = "logistic", mt_method = "BY",
                        output = "pvalues")
```

MWAS_SummarizedExperiment

Create a SummarizeExperiment object

Description

This function formats the metabolic and clinical data into a SummarizedExperiment object.

Usage

```r
MWAS_SummarizedExperiment(metabo_matrix, clinical_matrix, sample_type)
```

Arguments

- `metabo_matrix` numeric matrix containing the metabolic data (e.g. NMR peak intensities or metabolite concentrations). The columns of the matrix must correspond to the metabolic variables and the rows to the samples. Column and row names must contain the metabolite ids (e.g. chemical shifts for NMR data) and the sample ids, respectively.

- `clinical_matrix` numeric matrix containing the clinical data (e.g. age, gender). The columns of the matrix must correspond to the phenotypic variables and the rows to the samples. Column and row names must contain the phenotype ids and the sample ids, respectively. For samples without clinical data (e.g. quality control (QC) samples), NA values must be used.

- `sample_type` numeric vector indicating sample type (i.e. experimental sample or QC sample). The vector must be coded as follows: experimental sample = 0, QC sample = 1. If QC samples are not available, all the elements of this vector must be 0.

Value

A SummarizedExperiment object.
References


Examples

```r
## Load data
data(metabo_SE)

## Get metabolic_data, clinical_data, and sample_type
library(SummarizedExperiment)
metabolic_data = t(as.matrix(metabolome(metabo_SE)$metabolic_data))
clinical_data = as.matrix(colData(metabo_SE)[, -5])
sample_type = as.vector(colData(metabo_SE)[, 5])

## Reconstruct SummarizedExperiment
data_SE = MWAS_SummarizedExperiment(metabolic_data, clinical_data, sample_type)
```

plot_spectraNMR  

Plot NMR spectra

Description

This function generates an NMR spectra plot, with the chemical shifts displayed along the x-axis, and the peak intensities displayed on the y-axis.

Usage

```r
plot_spectraNMR (metabo_SE, type = "l", lty = 1, xlab = "ppm", ylab = "intensity", xlim = NULL, ...)
```

Arguments

- `metabo_SE`: SummarizedExperiment object. See "MWAS_SummarizedExperiment()".
- `type`: character vector indicating the type of plot for each row of metabo_matrix. For all possible types, see "plot()".
- `lty`: character vector of line types. For all possible types, see "plot()".
- `xlab`: character vector specifying a title for the x-axis.
- `ylab`: character vector specifying a title for the y-axis.
- `xlim`: numeric vector containing the minimum and maximum values of the x axis. Notice that ppm is displayed in reverse scale (e.g. xlim = c(10, 0)).
- `...`: other arguments passed to "matplot()".

Value

An NMR spectra plot.
### QC.CV

**Examples**

```r
## Load data
data(metabo_SE)

## Plot first 2 spectra
plot_spectraNMR (metabo_SE[, 1:2])
plot_spectraNMR (metabo_SE[, 1:2], xlim = c(1.03, 0.85), main = "NMR spectra")
```

---

**Calculate coefficients of variation**

**Description**

This function calculates the coefficient of variation (CV) (\(|sd/mean|\)) of each metabolic feature across the quality control (QC) samples. The CV distribution is represented in a histogram. This function can be used to assess the reproducibility of individual metabolic features. Notice that CV = 0.30 and CV = 0.15 are the thresholds established by the FDA guidelines for biomarker discovery and quantification, respectively.

**Usage**

```r
QC.CV (metabo_SE, CV_th = 0.30, plot_hist = TRUE, hist_bw = 0.005, hist_col = "moccasin", size_lab = 12, size_axis = 12)
```

**Arguments**

- `metabo_SE` SummarizedExperiment object. See "MWAS.SummarizedExperiment()".
- `CV_th` numeric value indicating the CV threshold.
- `plot_hist` logical constant indicating whether a histogram showing CV distribution will be plotted.
- `hist_bw` numeric value indicating histogram bin width.
- `hist_col` character string indicating the color to be used to fill the histogram bars.
- `size_lab` numeric value indicating the font size of x- and y-axis titles.
- `size_axis` numeric value indicating the font size of x- and y-axis labels.

**Value**

A numeric vector containing the CV of each metabolic feature and a histogram showing CV distribution. In the histogram, CVs above 1 are set to 1.

**References**

Examples

```r
## Load data
data(metabo_SE)

## Calculate CVs
metabo_CV <- QC_CV(metabo_SE)
metabo_CV2 <- QC_CV(metabo_SE, hist_bw = 0.008, hist_col = "lightblue")
```

## Description

This function allows plotting a reference NMR spectrum colored based on the coefficient of variation (CV) of each NMR signal. See function "QC_CV()".

## Usage

```r
QC_CV_specNMR(metabo_SE, ref_sample, CV_th = 0.30, xlab = "ppm", ylab = "intensity", size_axis = 12, size_lab = 12, xlim = NULL, ylim = NULL, xbreaks = waiver(), xnames = waiver(), ybreaks = waiver(), ynames = waiver())
```

## Arguments

- `metabo_SE`: SummarizedExperiment object. See "MWAS_SummarizedExperiment()".
- `ref_sample`: character vector indicating the id of the sample that will be used to plot the NMR spectrum.
- `CV_th`: numeric value indicating the CV threshold. NMR signals with CV equal or above CV_th will be colored in red.
- `xlab`: character vector specifying a title for the x-axis.
- `ylab`: character vector specifying a title for the y-axis.
- `size_axis`: numeric vector indicating the font size of x- and y-axis labels.
- `size_lab`: numeric vector indicating the font size of x- and y-axis titles.
- `xlim`: numeric vector containing the minimum and maximum values of the x-axis. Notice that ppm is displayed in reverse scaled (e.g. xlim = c(10, 0)).
- `ylim`: numeric vector containing the minimum and maximum values of the y-axis.
- `xbreaks`: numeric vector indicating the positions of the breaks of the x-axis.
- `xnames`: character vector (same length as xbreaks) containing the labels of each break of the x-axis.
- `ybreaks`: numeric vector indicating the positions of the breaks of the y-axis.
- `ynames`: character vector (same length as ybreaks) containing the labels of each break of the y-axis.

## Value

An NMR spectrum plot colored based on the CV of each NMR signal.
Expressions

## Load data
data(metabo_SE)

## Plot NMR spectrum colored by CV
QC_CV_specNMR(metabo_SE, ref_sample = "QC1", CV_th = 0.30)
QC_CV_specNMR(metabo_SE, ref_sample = "QC1", CV_th = 0.30, xlim = c(1.1, 0.95))
QC_CV_specNMR(metabo_SE, ref_sample = "QC1", CV_th = 0.15)

---

**QC_PCA**  
Principal Component Analysis

**Description**

This function performs PCA on a matrix of metabolic data and returns the results as an object of class "prcomp". When quality control (QC) samples are available, "QC_PCA()" can be used to assess the stability and reproducibility of the dataset.

**Usage**

QC_PCA(metabo_SE, scale = FALSE, center = TRUE,...)

**Arguments**

- **metabo_SE**  
  SummarizedExperiment object. See "MWAS_SummarizedExperiment()".

- **scale**  
  logical constant indicating whether the metabolic variables will be scaled to have unit variance before the analysis. For more details, check "prcomp()".

- **center**  
  logical constant indicating whether the metabolic variables will be shifted to be zero-centered before the analysis. For more details, check "prcomp()".

- **...**  
  other arguments passed to "prcomp()".

**Value**

A list with class "prcomp". For more details, check "prcomp()".

**References**

Examples

```r
## Load data
data(metabo_SE)
data(targetMetabo_SE)

## PCA model using all metabolic data
PCA_model <- QC_PCA(metabo_SE)

## PCA model using target metabolites
PCA_subset <- QC_PCA(targetMetabo_SE)
```

Description

This function generates a PCA score plot colored based on sample type (i.e. experimental or quality control (QC) sample). The plots generated with this function can be used to assess analytical reproducibility and stability. If the dataset is reproducible, all quality control samples should appear clustered in the center of the Hotelling’s ellipse.

Usage

```r
QC_PCA_scoreplot (PCA_model, metabo_SE, plot_labels = FALSE, px = 1, py = 2,
                   CI_level = 0.95, pch = 20, xlim = NULL, ylim = NULL,
                   color_scale = c("cornflowerblue", "red"), grid = TRUE,...)
```

Arguments

- **PCA_model** "prcomp" object generated by the function "QC_PCA()".
- **metabo_SE** SummarizedExperiment object. See "MWAS_SummarizedExperiment()".
- **plot_labels** logical constant indicating whether the sample ids will be displayed in the score plot.
- **px** numeric value indicating the index of the principal component that will be displayed on the x-axis.
- **py** numeric value indicating the index of the principal component that will be displayed on the y-axis.
- **CI_level** numeric value indicating the confidence interval for the Hotelling’s ellipse.
- **pch** value specifying the symbol that will represent each sample in the score. To see all possible symbols, check "plot()" options.
- **xlim** numeric vector containing the minimum and maximum values of the x-axis.
- **ylim** numeric vector containing the minimum and maximum values of the y-axis.
- **color_scale** character vector corresponding to the 2-color scale that will be used to discriminate the experimental samples from the QC samples.
- **grid** logical constant indicating whether grid lines will be added to the plot.
- **...** other arguments passed to "plot()".
STOCSY_NMR

Value

A PCA score plot.

References


Examples

## Load data
data(metabo_SE)

## PCA model
PCA_model <- QC_PCA(metabo_SE)

## PCA score plots
QC_PCA_scoreplot(PCA_model, metabo_SE) # PC1 vs PC2
QC_PCA_scoreplot(PCA_model, metabo_SE, px = 3, py = 4) # PC3 vs PC4
QC_PCA_scoreplot(PCA_model, metabo_SE, plot_labels = TRUE) # show labels
QC_PCA_scoreplot(PCA_model, metabo_SE, CI_level = 0.80) # change CI

STOCSY_NMR

Statistical Total Correlation Spectroscopy - Academic use only

Description

This function calculates STOCSY between an NMR signal of interest and all the NMR variables, representing a useful tool for NMR molecular identification and assignment. The results are represented in a pseudo-NMR spectrum displaying the covariance (height) and the Pearson correlation coefficient (color) of all spectral variables with the variable of interest (driver signal).

Usage

STOCSY_NMR(metabo_SE, ppm_query, alpha_th = 0.05, xlab = "ppm", ylab = "covariance", size_lab = 12, size_axis = 12, xlim = NULL, ylim = NULL, xbreaks = waiver(), xnames = waiver(), ynames = waiver(), ybreaks = waiver())

Arguments

metabo_SE SummarizedExperiment object. See "MWAS_SummarizedExperiment()".
ppm_query numeric value (at least 2 decimals) corresponding to the driver ppm.
alpha_th numeric value indicating the significance threshold. NMR variables with BH-adjusted p-value above this threshold will be neglected.
xlab character vector specifying a title for the x-axis.
ylab character vector specifying a title for the y-axis.
size_lab numeric value indicating the font size of x- and y-axis titles.
size_axis numeric value indicating the font size of x- and y-axis labels.
xlim numeric vector containing the minimum and maximum values of the x-axis. Notice that ppm is displayed in reverse scale (e.g. xlim = c(2, 1)).
ylim numeric vector containing the minimum and maximum values of the y-axis.
xbreaks numeric vector indicating the positions of the breaks of the x-axis.
xnames character vector (same length as xbreaks) containing the labels of each break of the x-axis.
ybreaks numeric vector indicating the positions of the breaks of the y-axis.
ynames character vector (same length as ybreaks) containing the labels of each break of the y-axis.

Value

A plot displaying the Pearson correlation coefficient (color) and covariance (height) between all spectral variables and the driver signal.

References


Examples

```r
## Load data
data(targetMetabo_SE)

## STOCSY using 1.04 as driver signal
STOCSY_NMR(targetMetabo_SE, ppm_query = 1.04)
STOCSY_NMR(targetMetabo_SE, ppm_query = 1.04, alpha_th = 0, xlim = c(1.06, 1))
```

Description

This SummarizedExperiment object contains the following information:
- An assay matrix containing the levels of 8 target 1H NMR metabolites (lactate, 3-hydroxy-butyrate, leucine, valine, isoleucine, acetate, alanine and 1,5-anhydroglucitol) across the experimental samples and the quality control (QC) samples.
- A data.frame containing clinical information (age, gender, type II diabetes status and BMI) and sample class (i.e. experimental sample or QC sample) information for each sample row in the assay matrix.

Usage

```r
data(targetMetabo_SE)
```

Format

SummarizedExperiment
targetMetabo_SE

Value

SummarizedExperiment
Index

CV_filter, 2
metabo_SE, 3
MWAS_barplot, 3
MWAS_bootstrapping, 4
MWAS_filter, 5
MWAS_network, 6
MWAS.skylineNMR, 8
MWAS.stats, 9
MWAS.SummarizedExperiment, 11
plot_spectraNMR, 12
QC.CV, 13
QC.CV.specNMR, 14
QC.PCA, 15
QC.PCA.scoreplot, 16
STOCSY.NMR, 17
targetMetabo_SE, 18