Package ‘POMA’

July 4, 2021

Title  User-friendly Workflow for Metabolomics and Proteomics Data Analysis

Version 1.2.0

Description  A structured, reproducible and easy-to-use workflow for the visualization, pre-processing, exploratory data analysis, and statistical analysis of metabolomics and proteomics data. The main aim of POMA is to enable a flexible data cleaning and statistical analysis processes in one comprehensible and user-friendly R package. This package also has a Shiny app version that implements all POMA functions. See https://github.com/pcastellanoescuder/POMAShiny.

License  GPL-3

Encoding  UTF-8

LazyData true

biocViews  MassSpectrometry, Metabolomics, Proteomics, Software, Visualization, Preprocessing, Normalization, ReportWriting

Imports  broom, caret, clisymbols, ComplexHeatmap, crayon, dplyr, e1071, ggcorrplot, ggplot2, ggraph, ggrepel, glasso (>= 1.11), glmnet, impute, knitr, limma, magrittr, mixOmics, MSnbase (>= 2.12), patchwork, qpdf, randomForest, RankProd (>= 3.14), markdown, tibble, tidyR, vegan

Suggests  Biobase, BiocStyle, covr, plotly, tidyverse, testthat (>= 2.3.2)

Roxygen  list(markdown = TRUE)

RoxygenNote  7.1.1

Depends  R (>= 4.0)

VignetteBuilder knitr

URL  https://github.com/pcastellanoescuder/POMA

BugReports  https://github.com/pcastellanoescuder/POMA/issues
git_url  https://git.bioconductor.org/packages/POMA
PomaBoxplots

**Description**

PomaBoxplots() generates a boxplot for subjects or features. This boxplot can help in the comparison between pre and post normalized data and in the "validation" of the normalization process.
**Usage**

```r
PomaBoxplots(
  data,
  group = "samples",
  jitter = TRUE,
  feature_name = NULL,
  label_size = 10,
  legend_position = "bottom"
)
```

**Arguments**

- **data**: A MSnSet object. First pData column must be the subject group/type.
- **group**: Grouping factor for the plot. Options are "samples" and "features". Option "samples" (default) will create a boxplot for each sample and option "features" will create a boxplot of each variable.
- **jitter**: Logical. If it's TRUE (default), the boxplot will show all points.
- **feature_name**: A vector with the name/s of feature/s to plot. If it’s NULL (default) a boxplot of all features will be created.
- **label_size**: Numeric indicating the size of x-axis labels.
- **legend_position**: Character indicating the legend position. Options are "none", "top", "bottom", "left", and "right".

**Value**

A ggplot2 object.

**Author(s)**

Pol Castellano-Escuder

**Examples**

```r
data("st000284")

# samples
PomaBoxplots(st000284)

# features
PomaBoxplots(st000284, group = "features")

# concrete features
PomaBoxplots(st000284, group = "features",
             feature_name = c("ornithine", "orotate"))
```
Description

This function performs a classical multidimensional scaling (MDS) using all features in the data and computes a cluster analysis for \( k \) clusters. Then, the calculated clusters will be represented on a MDS plot.

Usage

```r
PomaClust(
  data,
  method = "euclidean",
  k = NA,
  k_max = 15,
  show_clusters = TRUE,
  labels = FALSE,
  show_group = FALSE
)
```

Arguments

- **data**: A MSnSet object. First `pData` column must be the subject group/type.
- **method**: Distance measure method to perform MDS. Options are "euclidean", "maximum", "manhattan", "canberra" and "minkowski". See `?dist()`.
- **k**: Number of clusters (default is `NA`). The optimum number of clusters will be used by default.
- **k_max**: Number of clusters among which the optimal one will be selected.
- **show_clusters**: Logical indicating if clusters should be plotted or not. If this parameter is set to FALSE the resultant plot will be a classical 2-dimension MDS plot.
- **labels**: Logical indicating if sample names should be plotted or not.
- **show_group**: Logical indicating if the original sample group from `pData` should be plotted instead of sample ID or not. Only works if `labels` is set to `TRUE`.

Value

A list with the results.

Author(s)

Pol Castellano-Escuder
Examples

```r
data("st000284")
PomaClust(st000284)
```

**Description**

This function returns different correlation plots (correlogram and network plots) and a table with all pairwise correlations in the data.

**Usage**

```r
PomaCorr(
  data,
  method = "pearson",
  shape = "square",
  type = "full",
  show_corr = FALSE,
  low = "#336B87",
  outline = "white",
  high = "#EA8620",
  label_size = 12,
  corr_type = "cor",
  coeff = 0.7
)
```

**Arguments**

- **data**: A MSnSet object. First pData column must be the subject group/type.
- **method**: Character indicating which correlation coefficient has to be computed. Options are "pearson" (default), "kendall" and "spearman".
- **shape**: Character indicating shape of correlogram. Options are "square" (default) and "circle".
- **type**: Character indicating type of correlogram. Options are "full" (default), "lower" or "upper".
- **show_corr**: Logical indicating if correlation coefficient for each pair of features should be plotted in correlogram or not (default = FALSE). Only recommended for a low number of features.
- **low**: Colour for low end of the gradient in correlogram.
- **outline**: Colour for the outline of the gradient in correlogram.
- **high**: Colour for high end of the gradient in correlogram.
PomaDensity

Description

PomaDensity() generates a density plot of not normalized and normalized MS data. This plot can help in the comparison between pre and post normalized data and in the "validation" of the normalization process.

label_size  Numeric indicating label size in correlogram.
corr_type  Type of network to be made with correlation matrix. Options are "cor" (for global correlations) and "glasso" (for gaussian graphical model). Default is "cor". See glasso R package for the second option.
coeff  Numeric indicating correlation coefficient. Edges with absolute weight below this value will be removed from the network. If "corr_type" is set to "glasso", this parameter indicates the regularization parameter for lasso (rho = 0 means no regularization). See glasso::glasso().

Value

A list with the results.

Author(s)

Pol Castellano-Escuder

References


Examples

data("st000284")

# pearson correlation
PomaCorr(st000284)$correlations
PomaCorr(st000284)$corrplot

# gaussian graphical model
# library(ggraph)
# PomaCorr(st000284, corr_type = "glasso")
PomaDensity

Usage

```r
PomaDensity(
  data,
  group = "samples",
  feature_name = NULL,
  legend_position = "bottom"
)
```

Arguments

data A MSnSet object. First pData column must be the subject group/type.
group Grouping factor for the plot. Options are "samples" and "features". Option "samples" (default) will create a density plot for each group and option "features" will create a density plot of each variable.
feature_name A vector with the name/s of feature/s to plot. If it's NULL (default) a density plot of all variables will be created.
legend_position Character indicating the legend position. Options are "none", "top", "bottom", "left", and "right".

Value

A ggplot2 object.

Author(s)

Pol Castellano-Escuder

Examples

```r
data("st000284")

# samples
PomaDensity(st000284)

# features
PomaDensity(st000284, group = "features")

# concrete features
PomaDensity(st000284, group = "features",
            feature_name = c("ornithine", "orotate"))
```
PomaEDA  

Automatic Exploratory Data Analysis PDF Report

Description

This function automatically generates a PDF report with different exploratory plots and tables from an MSnSet object.

Usage

PomaEDA(
  data,
  imputation = "knn",
  normalization = "log_pareto",
  clean_outliers = TRUE,
  coeff_outliers = 1.5,
  username = "Username"
)

Arguments

data A MSnSet object. First pData column must be the subject group/type.
imputation Imputation method. Options are "none", "half_min", "median", "mean", "min" and "knn" (default). If "none", all missing values will be replaced by zero.
normalization Normalization method. Options are "none", "auto_scaling", "level_scaling", "log_scaling", "log_transformation", "vast_scaling" and "log_pareto" (default).
clean_outliers Logical. If it's set to TRUE, outliers will be removed from EDA.
coeff_outliers This value corresponds to the classical $1.5 \times IQR$ formula to detect outliers. By changing this value, the permissiveness in outlier detection will change.
username This name will be included as a report subtitle.

Value

An exploratory data analysis PDF report.

Author(s)

Pol Castellano-Escuder
PomaHeatmap

Classical Heatmap

Description

This function returns a basic heatmap plot made with base R.

Usage

PomaHeatmap(
  data,
  sample_names = TRUE,
  feature_names = FALSE,
  show_legend = TRUE
)

Arguments

data A MSnSet object. First pData column must be the subject group/type.
sample_names Logical indicating if sample names should be plotted or not. Default is TRUE.
feature_names Logical indicating if feature names should be plotted or not. Default is FALSE.
show_legend Logical indicating if legend should be plotted or not. Default is TRUE.

Value

A heatmap.

Author(s)

Pol Castellano-Escuder

Examples

data("st000284")

st000284 %>%
  PomaNorm() %>%
  PomaHeatmap()
Description

PomaImpute() offers different methods to impute missing values in MS data.

Usage

PomaImpute(
  data,
  ZerosAsNA = FALSE,
  RemoveNA = TRUE,
  cutoff = 20,
  method = "knn"
)

Arguments

data: A MSnSet object. First pData column must be the subject group/type.
ZerosAsNA: Logical that indicates if the zeros in the data are missing values. Default is FALSE.
RemoveNA: Logical that indicates if those features with more than selected cutoff missing values in each group have to be removed. Default is TRUE.
cutoff: Numeric that indicates the percentage of missing values allowed in each group. If one of the groups have less missing values than selected cutoff value, these feature will not be removed.
method: Imputation method. Options are: "none", "half_min", "median", "mean", "min", "knn" and "rf". If "none", all missing values will be replaced by zero.

Value

A MSnSet object with cleaned data.

Author(s)

Pol Castellano-Escuder

References


Examples

data("st000336")

PomaImpute(st000336, method = "knn")
PomaLasso

Lasso, Ridge and Elasticnet Regularized Generalized Linear Models for Binary Outcomes

Description

PomaLasso() is an implementation of the lasso, ridge and elasticnet regression from glmnet package for binary outcomes.

Usage

```r
PomaLasso(
  data,
  alpha = 1,
  ntest = NULL,
  nfolds = 10,
  lambda = NULL,
  labels = FALSE
)
```

Arguments

data: A MSnSet object. First pData column must be the subject group/type.
alpha: Elasticnet mixing parameter. alpha = 1 is the lasso penalty and alpha = 0 is the ridge penalty. This value must be between 0 and 1.
test: Numeric indicating the percentage of observations that will be used as test set. Default is NULL (no test set).
nfolds: Number of folds for CV (default is 10). Although nfolds can be as large as the sample size (leave-one-out CV), it is not recommended for large datasets. Smallest value allowable is nfolds = 3.
lambda: A user supplied lambda sequence. Typical usage is to have the program compute its own lambda sequence based on nlambda and lambda.min.ratio. See ?glmnet::glmnet().
labels: Logical indicating if feature names should be plotted in coefficient plot or not. Default is FALSE.

Value

A list with all results including plots, data frames and the resulting prediction model.

Author(s)

Pol Castellano-Escuder
References

Examples

data("st000336")

# lasso
st000336 %>%
PomaImpute() %>%
PomaNorm() %>%
PomaOutliers() %>%
PomaLasso()

# elasticnet
st000336 %>%
PomaImpute() %>%
PomaNorm() %>%
PomaOutliers() %>%
PomaLasso(alpha = 0.5)

# ridge
st000336 %>%
PomaImpute() %>%
PomaNorm() %>%
PomaOutliers() %>%
PomaLasso(alpha = 0)

---

PomaLimma

Implementation of limma R Package on Mass Spectrometry Data

Description
PomaLimma() uses the classical limma package for MS data.

Usage

PomaLimma(
  data,
  contrast = NULL,
  covariates = FALSE,
  adjust = "fdr",
  cutoff = NULL
)
**PomaMSnSetClass**

Convert data frames to a MSnSet Object

**Description**

This function converts data frame objects to a MSnSet object.

**Usage**

PomaMSnSetClass(target, features)

---

**Arguments**

- **data**
  - A MSnSet object. First pData column must be the subject group/type.

- **contrast**
  - A character with the limma comparison. For example, "Group1-Group2" or "control-intervention".

- **covariates**
  - Logical. If it's set to TRUE all metadata variables stored in pData will be used as covariables. Default = FALSE.

- **adjust**
  - Multiple comparisons correction method. Options are: "fdr", "holm", "hochberg", "hommel", "bonferroni", "BH" and "BY".

- **cutoff**
  - Default is NULL. If this value is replaced for a numeric value, the resultant table will contains only those features with an adjusted p-value below selected value.

**Value**

A data frame with limma results.

**Author(s)**

Pol Castellano-Escuder

**References**


**Examples**

```r
data("st000284")

st000284 %>%
PomaNorm() %>%
PomaLimma(contrast = "Healthy-CRC", adjust = "fdr")
```
Arguments

- **target**: Metadata variables structured in columns. Sample ID must be the first column and group/type/treatment of the study must be the second column.
- **features**: Table of features. Each feature in one column.

Value

A MSnSet object.

Author(s)

Pol Castellano-Escuder

References


Examples

```r
data(iris)

# create target: two column (or more) data frame with IDs and Group factor
target <- data.frame(ID = 1:150, Group = iris$Species)

# create features: p column data frame (or matrix) with features
features <- iris[,1:4]

# create an MSnSet object with POMA
object <- PomaMSnSetClass(target = target, features = features)
```

Description

PomaMultivariate() allows users to perform different multivariate statistical analysis on MS data.

Usage

```r
PomaMultivariate(  
data,  
method = "pca",  
components = 5,  
center = FALSE,  
scale = FALSE,  
labels = FALSE,
)```
load_length = 1,
ellipse = TRUE,
validation = "Mfold",
folds = 5,
nrepeat = 10,
vip = 1.5,
num_features = 10,
legend_position = "bottom"
)

Arguments

- **data**: A MSnSet object. First pData column must be the subject group/type.
- **method**: A multivariate method. Options are: "pca", "plsda" and "splsda".
- **components**: Numeric. Number of components to include in the model. Default is 5.
- **center**: Logical. Indicates whether the variables should be shifted to be zero centered. Default is FALSE.
- **scale**: Logical. Indicates whether the variables should be scaled to have unit variance before the analysis takes place. Default is FALSE.
- **labels**: Logical indicating if sample names should be plotted or not.
- **load_length**: Numeric between 1 and 2. Define the length of biplot loadings. Default is 1.
- **ellipse**: Logical that indicates whether a 95% CI ellipse should be plotted in scores plot. Default is TRUE.
- **validation**: (Only for "plsda" and "splsda" methods) Validation method. Options are "Mfold" and "loo".
- **folds**: (Only for "plsda" and "splsda" methods) Numeric. Number of folds for Mfold validation method (default is 5). If the validation method is loo, this value will become to 1.
- **nrepeat**: (Only for "plsda" and "splsda" methods) Numeric. Number of iterations for the validation method selected.
- **vip**: (Only for "plsda" method) Numeric indicating VIP cutoff to select features that will be displayed in vip plot.
- **num_features**: (Only for "splsda" method) Numeric. Number of variables selected to discriminate groups.
- **legend_position**: Character indicating the legend position. Options are "none", "top", "bottom", "left", and "right".

Value

A list with all results for multivariate statistical analysis including plots and data frames.

Author(s)

Pol Castellano-Escuder
Examples

data("st000336")

# PCA
st000336 %>%
PomaImpute() %>%
PomaNorm() %>%
PomaOutliers() %>%
PomaMultivariate(method = "pca")

# PLSDA
st000336 %>%
PomaImpute() %>%
PomaNorm() %>%
PomaOutliers() %>%
PomaMultivariate(method = "plsda", vip = 1)

# sPLSDA
st000336 %>%
PomaImpute() %>%
PomaNorm() %>%
PomaOutliers() %>%
PomaMultivariate(method = "splsda")

PomaNorm

Collection of Normalization Methods for Mass Spectrometry Data

Description

PomaNorm() offers different methods to normalize MS data. This function contains both centering and scaling functions to normalize the data.

Usage

PomaNorm(data, method = "log_pareto", round = 3)

Arguments

data A MSnSet object. First pData column must be the subject group/type.
method Normalization method. Options are: "none", "auto_scaling", "level_scaling", "log_scaling", "log_transformation", "vast_scaling" and "log_pareto".
round Numeric. Number of decimal places (Default is 3).

Value

A MSnSet object with normalized data.
**PomaOddsRatio**

**Author(s)**
Pol Castellano-Escuder

**References**

**Examples**
```r
data("st000284")
PomaNorm(st000284, method = "log_pareto")```

---

**PomaOddsRatio**  
**Logistic Regression Model Odds Ratios**

**Description**
PomaOddsRatio() calculates the Odds Ratios for each feature from a logistic regression model using the binary outcome (group/type must be a binary factor) as a dependent variable.

**Usage**
PomaOddsRatio(data, feature_name = NULL, covariates = FALSE, showCI = TRUE)

**Arguments**
- `data`: A MSnSet object. First pData column must be the subject group/type.
- `feature_name`: A vector with the name/s of feature/s that will be used to fit the model. If it’s NULL (default), all variables will be included in the model.
- `covariates`: Logical that indicates if covariates will be included in logistic regression model. Default is FALSE.
- `showCI`: Logical that indicates if the 95% confidence intervals will be plotted. Default is TRUE.

**Value**
A data frame with the Odds Ratios for all features with their 95% confidence intervals and a ggplot2 object.

**Author(s)**
Pol Castellano-Escuder
Examples

data("st000336")

st000336 %>%
PomaImpute() %>%
PomaNorm() %>%
PomaOddsRatio(feature_name = c("glutamic_acid", "glutamine", "glycine", "histidine"))

---

PomaOutliers  Remove and Analyze Outliers

Description

This function allows users to analyze outliers by different plots and remove them from an MSnSet object.

Usage

PomaOutliers(
  data,
  do = "clean",
  method = "euclidean",
  type = "median",
  coef = 1.5,
  labels = FALSE
)

Arguments

data  A MSnSet object. First pData column must be the subject group/type.
do  Action to do. Options are "clean" (to remove detected outliers) and "analyze" (to analyze data outliers). Note that the output of this function will be different depending on this parameter.
method  Distance measure method to perform MDS. Options are "euclidean", "maximum", "manhattan", "canberra" and "minkowski". See ?dist().
type  Type of outliers analysis to perform. Options are "median" (default) and "centroid". See vegan::betadisper.
coef  This value corresponds to the classical $1.5\times IQR$ formula to detect outliers. By changing this value, the permissiveness in outlier detection will change.
labels  Logical indicating if sample IDs should to be plotted or not.

Value

A MSnSet object with cleaned data or different exploratory plots for the detailed analysis of outliers (depending on "do" parameter).
Author(s)

Pol Castellano-Escuder

Examples

data("st000336")

# clean outliers
st000336 %>%
PomaImpute() %>%
PomaNorm() %>%
PomaOutliers()

# analyze outliers
st000336 %>%
PomaImpute() %>%
PomaNorm() %>%
PomaOutliers(do = "analyze")

PomaRandForest

Classification Random Forest for Mass Spectrometry Data

Description

PomaRandForest() allows users to perform a classification Random Forest with a MS data matrix using the classical randomForest R package.

Usage

PomaRandForest(
  data,
  ntest = 20,
  ntree = 500,
  mtry = floor(sqrt(ncol(t(MSnbase::exprs(data))))),
  nodesize = 1,
  nvar = 20
)

Arguments

data A MSnSet object. First pData column must be the subject group/type.
ntest Numeric indicating the percentage of observations that will be used as test set. Default is 20% of observations.
ntree Number of trees to grow.
mtry Number of variables randomly sampled as candidates at each split. This value is set sqrt(p) (where p is number of variables in data) by default.
nodesize Minimum size of terminal nodes. By default is equal to 1.
nvar Number of variables to show in the Gini plot.
Value
A list with all results for Random Forest including plots and data frames.

Author(s)
Pol Castellano-Escuder

References

Examples
```r
data("st000336")
st000336 %>%
PomaImpute() %>%
PomaRandForest()
```

Description
PomaRankProd() performs the Rank Product method to identify differential feature concentration/intensity.

Usage
```r
PomaRankProd(
  data,
  logged = TRUE,
  logbase = 2,
  paired = NA,
  cutoff = 0.05,
  method = "pfp"
)
```

Arguments
- data: A MSnSet object. First `pData` column must be the subject group/type.
- logged: If "TRUE" (default) data have been previously log transformed.
- paired: Number of random pairs generated in the function, if set to NA (default), the odd integer closer to the square of the number of replicates is used.
cutoff

The pfp/pvalue threshold value used to select features.

method

If cutoff is provided, the method needs to be selected to identify features. "pfp" uses percentage of false prediction, which is a default setting. "pval" uses p-values which is less stringent than pfp.

Value

A list with all results for Rank Product analysis including tables and plots.

Author(s)

Pol Castellano-Escuder

References


Description

PomaUnivariate() allows users to perform different univariate statistical analysis on MS data.

Usage

```
PomaUnivariate(
  data,
  covariates = FALSE,
  method = "ttest",
  paired = FALSE,
  var_equal = FALSE,
  adjust = "fdr"
)
```
Arguments

- `data` A MSnSet object. First pData column must be the subject group/type.
- `covariates` Logical. If it's set to TRUE all metadata variables stored in pData will be used as covariates. Default = FALSE.
- `method` Univariate statistical method. Options are: "ttest", "anova", "mann" and "kruskal".
- `paired` Logical that indicates if the data is paired or not.
- `var_equal` Logical that indicates if the data variance is equal or not.
- `adjust` Multiple comparisons correction method. Options are: "fdr", "holm", "hochberg", "hommel", "bonferroni", "BH" and "BY".

Value

A data frame with results.

Author(s)

Pol Castellano-Escuder

Examples

```r
data("st000336")
data("st000284")

# ttest
st000336 %>%
  PomaImpute() %>%
  PomaNorm() %>%
  PomaOutliers() %>%
  PomaUnivariate(method = "ttest")

# ANOVA
st000284 %>%
  PomaImpute() %>%
  PomaNorm() %>%
  PomaOutliers() %>%
  PomaUnivariate(method = "anova")
```

Description

PomaVolcano() generates a volcano plot from the PomaUnivariate(method = "ttest") result. The data can't have negative values.
Usage

PomaVolcano(
data, 
pval = "raw", 
pval_cutoff = 0.05, 
adjust = "fdr", 
log2FC = 0.6, 
xlim = 2, 
labels = FALSE, 
paired = FALSE, 
var_equal = FALSE, 
interactive = FALSE, 
plot_title = TRUE
)

Arguments

data A MSnSet object. First pData column must be the subject group/type. Only for two group data!
pval Select a pvalue type to generate the volcano plot. Options are: "raw" and "adjusted".
pval_cutoff Numeric. Define the pvalue cutoff (horizontal line).
adjust Multiple comparisons correction method for t test result. Options are: "fdr", "holm", "hochberg", "hommel", "bonferroni", "BH" and "BY".
log2FC Numeric. Define the log2 fold change cutoff (vertical lines).
xlim Numeric. Define the limits for x axis.
labels Logical that indicates if selected labels will be plotted or not. Default is FALSE.
paired Logical that indicates if the data is paired or not.
var_equal Logical that indicates if the data variance is equal or not.
interactive Logical that indicates if an interactive plot will be plotted or not. Default is FALSE.
plot_title Logical that indicates if title will be plotted or not. Default is TRUE.

Value

A ggplot2 object.

Author(s)

Pol Castellano-Escuder

Examples

data("st000336")
st000336 %>%
Description

Colorectal cancer (CRC) is one of the most prevalent and deadly cancers in the world. Despite an expanding knowledge of its molecular pathogenesis during the past two decades, robust biomarkers to enable screening, surveillance, and therapy monitoring of CRC are still lacking. In this study, we present a targeted liquid chromatography-tandem mass spectrometry-based metabolic profiling approach for identifying biomarker candidates that could enable highly sensitive and specific CRC detection using human serum samples. In this targeted approach, 158 metabolites from 25 metabolic pathways of potential significance were monitored in 234 serum samples from three groups of patients (66 CRC patients, 76 polyp patients, and 92 healthy controls). Partial least squares-discriminant analysis (PLS-DA) models were established, which proved to be powerful for distinguishing CRC patients from both healthy controls and polyp patients. Receiver operating characteristic curves generated based on these PLS-DA models showed high sensitivities (0.96 and 0.89, respectively, for differentiating CRC patients from healthy controls or polyp patients); good specificities (0.80 and 0.88), and excellent areas under the curve (0.93 and 0.95) were also obtained. Monte Carlo cross validation (MCCV) was also applied, demonstrating the robust diagnostic power of this metabolic profiling approach.

Usage

st000284

Format

A MSnSet object: 224 samples, 113 metabolites, 4 covariables and 3 groups (CRC, Healthy and Polyp).

metabolites 113 serum metabolites.
covariables  Age at consent, Gender, Smoking Condition and Alcohol Consumption.

Source

https://www.metabolomicsworkbench.org/data/DRCCMetadata.php?Mode=Study&StudyID=ST000284&StudyType=MS&ResultType=1%20target=_blank

References

Targeted LC/MS of urine from boys with DMD and controls

Description

Duchenne Muscular Dystrophy (DMD) is an X-linked recessive form of muscular dystrophy that affects males via a mutation in the gene for the muscle protein, dystrophin. Progression of the disease results in severe muscle loss, ultimately leading to paralysis and death. Steroid therapy has been a commonly employed method for reducing the severity of symptoms. This study aims to quantify the urine levels of amino acids and organic acids in patients with DMD both with and without steroid treatment. Track the progression of DMD in patients who have provided multiple urine samples.

Usage

st000336

Format

A MSnSet object: 57 samples, 31 metabolites, 1 covariable and 2 groups (Controls and DMD).

- metabolites 31 urine metabolites.
- covariables Steroid status.

Source

https://www.metabolomicsworkbench.org/data/DRCCMetadata.php?Mode=Study&DataMode=AllData&StudyID=ST000336&StudyType=MS&ResultType=1#DataTabs
Index

* datasets
  st000284, 24
  st000336, 25
  PomaBoxplots, 2
  PomaClust, 4
  PomaCorr, 5
  PomaDensity, 6
  PomaEDA, 8
  PomaHeatmap, 9
  PomaImpute, 10
  PomaLasso, 11
  PomaLimma, 12
  PomaMSnSetClass, 13
  PomaMultivariate, 14
  PomaNorm, 16
  PomaOddsRatio, 17
  PomaOutliers, 18
  PomaRandForest, 19
  PomaRankProd, 20
  PomaUnivariate, 21
  PomaVolcano, 22

  st000284, 24
  st000336, 25