

Package: omicsTools (via r-universe)

June 19, 2024

Title Omics Data Process Toolbox

Version 1.1.2

Date 2023-07-04

Description Processing and analyzing omics data from genomics, transcriptomics, proteomics, and metabolomics platforms. It provides functions for preprocessing, normalization, visualization, and statistical analysis, as well as machine learning algorithms for predictive modeling. 'omicsTools' is an essential tool for researchers working with high-throughput omics data in fields such as biology, bioinformatics, and medicine. The QC-RLSC (quality control-based robust LOESS signal correction) algorithm is used for normalization. Dunn et al. (2011) <[doi:10.1038/nprot.2011.335](https://doi.org/10.1038/nprot.2011.335)>.

License AGPL (>= 3)

Imports bs4Dash, cli, config (>= 0.3.1), dbscan, dplyr, DT, ggplot2, golem (>= 0.3.5), magrittr, MASS, matrixStats, moments, outliers, progress, readr, shiny (>= 1.7.2), tibble, tidyr, UpSetR

Encoding UTF-8

LazyData true

RoxygenNote 7.3.1

Suggests spelling, testthat (>= 3.0.0), knitr, rmarkdown

VignetteBuilder knitr

Config/testthat/edition 3

URL <https://github.com/YaoxiangLi/omicsTools>

BugReports <https://github.com/YaoxiangLi/omicsTools/issues>

Repository <https://cheemalab.r-universe.dev>

RemoteUrl <https://github.com/cheemalab/omicstools>

RemoteRef HEAD

RemoteSha 3e1e4f869291d4ea7ff579300d3a1f002ce049ea

Contents

calculate_cooks_distance	2
calculate_lof	3
calculate_measures	3
convert_to_binary_matrix	4
createOmicsData	4
define_thresholds	5
ensure_enough_sets_for_upset	6
flag_anomalies	6
generate_data_with_anomalies	7
handle_missing_values	7
initialize_results_df	8
is_normalize	9
load_is_area	9
load_peak_area	10
nist_rsd	11
OmicsData-class	11
pieDraw	12
plot_distribution_measures	12
plot_sample_measures	13
prepare_upset_data	13
pvcaBF	14
pvcaDraw	14
qc_normalize	14
qc_rsd	15
run_app	16
Index	17

calculate_cooks_distance

Calculate Cook's Distance

Description

Calculate Cook's Distance

Usage

calculate_cooks_distance(values)

Arguments

values A numeric vector of values for which to calculate Cook's Distance.

Value

A numeric vector of Cook's Distance values.

Examples

```
values <- rnorm(100)
calculate_cooks_distance(values)
```

calculate_lof	<i>Calculate Local Outlier Factor (LOF)</i>
---------------	---

Description

Calculate Local Outlier Factor (LOF)

Usage

```
calculate_lof(values, k = 5)
```

Arguments

values	A numeric vector of values for which to calculate LOF.
k	The number of neighbors to use for calculating LOF. Defaults to 5.

Value

A numeric vector of LOF values.

Examples

```
values <- rnorm(100)
calculate_lof(values)
```

calculate_measures	<i>Calculate Measures for Each Feature</i>
--------------------	--

Description

Calculate Measures for Each Feature

Usage

```
calculate_measures(object)
```

Arguments

object	An OmicsData object.
--------	----------------------

Value

The OmicsData object with calculated measures for each feature.

Examples

```
omics_data <- createOmicsData()
omics_data <- calculate_measures(omics_data)
```

```
convert_to_binary_matrix
```

Convert to Binary Matrix for UpSetR

Description

Convert to Binary Matrix for UpSetR

Usage

```
convert_to_binary_matrix(upset_data)
```

Arguments

upset_data The data frame prepared for UpSet plot.

Value

A binary matrix for UpSetR.

Examples

```
omics_data <- createOmicsData()
omics_data <- calculate_measures(omics_data)
omics_data <- flag_anomalies(omics_data)
upset_data <- prepare_upset_data(omics_data)
upset_matrix <- convert_to_binary_matrix(upset_data)
```

```
createOmicsData
```

Constructor for OmicsData

Description

Constructor for OmicsData

Usage

```
createOmicsData(data = NULL, n_samples = 100, n_features = 2000)
```

Arguments

data	The data frame containing the data. If not provided, synthetic data will be generated.
n_samples	The number of samples to generate if data is not provided. Defaults to 100.
n_features	The number of features to generate if data is not provided. Defaults to 2000.

Value

An OmicsData object.

Examples

```
omics_data <- createOmicsData()
user_data <- generate_data_with_anomalies()
omics_data <- createOmicsData(data = user_data)
```

define_thresholds *Define Anomaly Thresholds*

Description

Define Anomaly Thresholds

Usage

```
define_thresholds(
  skewness = 2.5,
  kurtosis = 10,
  shapiro_p = 1e-10,
  cooks_distance = 1,
  lof = 15
)
```

Arguments

skewness	The threshold for skewness. Defaults to 2.5.
kurtosis	The threshold for kurtosis. Defaults to 10.
shapiro_p	The threshold for Shapiro-Wilk test p-value. Defaults to 1e-10.
cooks_distance	The threshold for Cook's Distance. Defaults to 1.
lof	The threshold for Local Outlier Factor. Defaults to 15.

Value

A list of thresholds for anomaly detection.

Examples

```
thresholds <- define_thresholds()
thresholds <- define_thresholds(skewness = 3, kurtosis = 8)
```

```
ensure_enough_sets_for_upset
```

Ensure There Are Enough Sets for UpSet Plot

Description

Ensure There Are Enough Sets for UpSet Plot

Usage

```
ensure_enough_sets_for_upset(upset_matrix)
```

Arguments

upset_matrix The binary matrix for UpSetR.

Examples

```
omics_data <- createOmicsData()
omics_data <- calculate_measures(omics_data)
omics_data <- flag_anomalies(omics_data)
upset_data <- prepare_upset_data(omics_data)
upset_matrix <- convert_to_binary_matrix(upset_data)
ensure_enough_sets_for_upset(upset_matrix)
```

```
flag_anomalies
```

Flag Anomalies

Description

Flag Anomalies

Usage

```
flag_anomalies(object)
```

Arguments

object An OmicsData object.

Value

The OmicsData object with flagged anomalies.

Examples

```
omics_data <- createOmicsData()
omics_data <- calculate_measures(omics_data)
omics_data <- flag_anomalies(omics_data)
```

```
generate_data_with_anomalies
```

Generate High-Dimensional Data with Anomalies

Description

Generate High-Dimensional Data with Anomalies

Usage

```
generate_data_with_anomalies(n_samples = 100, n_features = 2000)
```

Arguments

n_samples The number of samples to generate. Defaults to 100.
n_features The number of features to generate. Defaults to 2000.

Value

A data frame containing the generated data with anomalies.

Examples

```
data <- generate_data_with_anomalies()
```

```
handle_missing_values
```

Handle Missing Values in a Tibble

Description

This function filters features based on a missing value threshold and imputes missing values using various methods. It provides an interactive experience with CLI emojis and progress bars.

Usage

```
handle_missing_values(data, threshold = 0.2, imputation_method = "mean")
```

Arguments

<code>data</code>	A tibble containing the data with potential missing values.
<code>threshold</code>	A numeric value between 0 and 1 representing the maximum allowable proportion of missing values in a feature. Default is 0.20.
<code>imputation_method</code>	A character string indicating the method to use for imputation. Valid methods are "mean", "median", "mode", and "half_min". Default is "mean".

Value

A tibble with filtered features and imputed missing values.

Author(s)

Yaoxiang Li

Examples

```
data <- tibble::tibble(
  Feature1 = c(1, 2, NA, 4, 5),
  Feature2 = c(NA, 2, 3, 4, NA),
  Feature3 = c(1, NA, 3, NA, 5)
)
imputed_data <- data |> handle_missing_values(threshold = 0.20, imputation_method = "median")
print(imputed_data)
```

`initialize_results_df` *Initialize Results Data Frame*

Description

Initialize Results Data Frame

Usage

```
initialize_results_df(data)
```

Arguments

<code>data</code>	The data frame containing the generated data.
-------------------	---

Value

A data frame initialized with columns for anomaly measures.

Examples

```
data <- generate_data_with_anomalies()
anomaly_measures <- initialize_results_df(data)
```

is_normalize	<i>Normalize loaded tibble with IS tibble</i>
--------------	---

Description

Normalize loaded tibble with IS tibble

Usage

```
is_normalize(loaded_tibble, is_tibble)
```

Arguments

loaded_tibble A tibble loaded by 'load_peak_area' or similar function.

is_tibble A tibble loaded by 'load_is_area' or similar function.

Value

A tibble that is normalized by the IS tibble.

Author(s)

Yaoxiang Li <y1814@georgetown.edu>

Georgetown University, USA

License: GPL (>= 3)

load_is_area	<i>Load and clean IS area data</i>
--------------	------------------------------------

Description

Load and clean IS area data

Usage

```
load_is_area(is_area_path, delim = "\t", na = c("N/A", "Unknown"))
```

Arguments

is_area_path Character string specifying the path to the IS area data file.

delim Character string indicating the delimiter used in the file.

na Character vector of strings to be treated as NA. Defaults to c("N/A", "Unknown").

Value

A tibble containing the cleaned IS area data.

Author(s)

Yaoxiang Li <y1814@georgetown.edu>

Georgetown University, USA

License: GPL (>= 3)

load_peak_area	<i>Load and clean peak area data</i>
----------------	--------------------------------------

Description

Load and clean peak area data

Usage

```
load_peak_area(peak_area_path, delim = "\t", na = c("N/A", "Unknown"))
```

Arguments

peak_area_path Character string specifying the path to the peak area data file.

delim Character string indicating the delimiter used in the file.

na Character vector of strings to be treated as NA. Defaults to c("N/A", "Unknown").

Value

A tibble containing the cleaned peak area data.

Author(s)

Yaoxiang Li <y1814@georgetown.edu>

Georgetown University, USA

License: GPL (>= 3)

nist_rsd	<i>Calculate the relative standard deviation for NIST samples</i>
----------	---

Description

Calculate the relative standard deviation for NIST samples

Usage

```
nist_rsd(normalized_tibble, nist_name = "NIST")
```

Arguments

`normalized_tibble` A tibble normalized by 'is_normalize' or similar function.
`nist_name` Character string specifying the name of the NIST sample. Defaults to "NIST".

Value

A tibble containing the mean, standard deviation, and coefficient of variation for NIST samples.

Author(s)

Yaoxiang Li <y1814@georgetown.edu>
Georgetown University, USA
License: GPL (>= 3)

OmicsData-class	<i>OmicsData Class</i>
-----------------	------------------------

Description

A class to represent and analyze high-dimensional omics data.

Slots

`data` The data frame containing the generated data with anomalies.
`anomaly_measures` The data frame containing the calculated anomaly measures.
`thresholds` The list of thresholds for anomaly detection.

```
pieDraw          import ggplot2
```

Description

```
import ggplot2
```

Usage

```
pieDraw(pvcaobj)
```

```
plot_distribution_measures  
      Plot Distribution Measures
```

Description

Plot Distribution Measures

Usage

```
plot_distribution_measures(object)
```

Arguments

object An OmicsData object.

Examples

```
omics_data <- createOmicsData()  
omics_data <- calculate_measures(omics_data)  
omics_data <- flag_anomalies(omics_data)  
plot_distribution_measures(omics_data)
```

plot_sample_measures *Plot Sample Measures*

Description

Plot Sample Measures

Usage

```
plot_sample_measures(object)
```

Arguments

object An OmicsData object.

Examples

```
omics_data <- createOmicsData()
omics_data <- calculate_measures(omics_data)
omics_data <- flag_anomalies(omics_data)
plot_sample_measures(omics_data)
```

prepare_upset_data *Prepare Data for UpSet Plot*

Description

Prepare Data for UpSet Plot

Usage

```
prepare_upset_data(object)
```

Arguments

object An OmicsData object.

Value

A data frame prepared for UpSet plot.

Examples

```
omics_data <- createOmicsData()
omics_data <- calculate_measures(omics_data)
omics_data <- flag_anomalies(omics_data)
upset_data <- prepare_upset_data(omics_data)
```

pvcaBF	<i>import Biobase pcva</i>
--------	----------------------------

Description

```
import Biobase pcva
```

Usage

```
pvcaBF(df, sampleInfo, batch_effect, threshold)
```

pvcaDraw	<i>import ggplot2</i>
----------	-----------------------

Description

```
import ggplot2
```

Usage

```
pvcaDraw(pvcaobj)
```

qc_normalize	<i>QC-Normalize function</i>
--------------	------------------------------

Description

This function performs normalization on the input data matrix using the loess regression method. Normalization is done based on Quality Control (QC) samples in the data.

Usage

```
qc_normalize(data, qc_label = "QC")
```

Arguments

data	A data frame containing the sample data. The first column
qc_label	A string indicating the string should contain the sample identifiers, and the rest of the columns contain the peaks to be normalized. QC samples should be indicated in the sample identifiers with 'QC'.

Value

A data frame with the first column as the sample identifiers and the rest of the columns containing the normalized peak intensities.

Author(s)

Yaoxiang Li <yl1814@georgetown.edu>

Georgetown University, USA

License: GPL (>= 3)

Examples

```
# Load the CSV data
data_file <- system.file("extdata", "example2.csv", package = "omicsTools")
data <- readr::read_csv(data_file)
# Apply the qc_normalize function
normalized_data <- omicsTools::qc_normalize(data)

# Write the normalized data to a new CSV file
readr::write_csv(normalized_data, paste0(tempdir(), "/normalized_data.csv"))
```

qc_rsd

Calculate the relative standard deviation for QC samples

Description

Calculate the relative standard deviation for QC samples

Usage

```
qc_rsd(normalized_tibble, qc_name = "Pooled QC")
```

Arguments

normalized_tibble

A tibble normalized by 'is_normalize' or similar function.

qc_name

Character string specifying the name of the QC sample. Defaults to "Pooled QC".

Value

A tibble containing the mean, standard deviation, and coefficient of variation for QC samples.

Author(s)

Yaoxiang Li <yl1814@georgetown.edu>

Georgetown University, USA

License: GPL (>= 3)

`run_app`*Run the Shiny Application*

Description

Run the Shiny Application

Usage

```
run_app(  
  onStart = NULL,  
  options = list(),  
  enableBookmarking = NULL,  
  uiPattern = "/",  
  ...  
)
```

Arguments

<code>onStart</code>	A function that will be called before the app is actually run. This is only needed for <code>shinyAppObj</code> , since in the <code>shinyAppDir</code> case, a global <code>.R</code> file can be used for this purpose.
<code>options</code>	Named options that should be passed to the <code>runApp</code> call (these can be any of the following: "port", "launch.browser", "host", "quiet", "display.mode" and "test.mode"). You can also specify width and height parameters which provide a hint to the embedding environment about the ideal height/width for the app.
<code>enableBookmarking</code>	Can be one of "url", "server", or "disable". The default value, <code>NULL</code> , will respect the setting from any previous calls to <code>enableBookmarking()</code> . See <code>enableBookmarking()</code> for more information on bookmarking your app.
<code>uiPattern</code>	A regular expression that will be applied to each GET request to determine whether the <code>ui</code> should be used to handle the request. Note that the entire request path must match the regular expression in order for the match to be considered successful.
<code>...</code>	arguments to pass to <code>golem_opts</code> . See <code>'?golem::get_golem_options'</code> for more details.

Value

No return value, called for launch the application.

Index

`calculate_cooks_distance`, 2
`calculate_lof`, 3
`calculate_measures`, 3
`convert_to_binary_matrix`, 4
`createOmicsData`, 4

`define_thresholds`, 5

`enableBookmarking()`, 16
`ensure_enough_sets_for_upset`, 6

`flag_anomalies`, 6

`generate_data_with_anomalies`, 7

`handle_missing_values`, 7

`initialize_results_df`, 8
`is_normalize`, 9

`load_is_area`, 9
`load_peak_area`, 10

`nist_rsd`, 11

`OmicsData-class`, 11

`pieDraw`, 12
`plot_distribution_measures`, 12
`plot_sample_measures`, 13
`prepare_upset_data`, 13
`pvcaBF`, 14
`pvcaDraw`, 14

`qc_normalize`, 14
`qc_rsd`, 15

`run_app`, 16