Package ‘dispersionIndicators’

October 16, 2025

Title Indicators for the Analysis of Dispersion of Datasets with Batched and Ordered Samples

Version 0.1.1

Depends R (>= 4.1)

Description Provides methods for analyzing the dispersion of tabular datasets with batched and ordered samples. Based on convex hull or integrated covariance Mahalanobis, several indicators are implemented for inter and intra batch dispersion analysis. It is designed to facilitate robust statistical assessment of data variability, supporting applications in exploratory data analysis and quality control, for such datasets as the one found in metabololomics studies.
For more details see Salanon (2024) <doi:10.1016/j.chemolab.2024.105148> and Salanon (2025) <doi:10.1101/2025.08.01.668073>.

License MIT + file LICENSE

Encoding UTF-8

RoxygenNote 7.3.3

Suggests cli, pdftools, testthat (>= 3.0.0)

Imports corpcor, ggplot2 (>= 3.5.2), stats, utils

Config/testthat/edition 3

Collate convex_function.R icm_function.R

NeedsCompilation no

Author Brice Mulot [aut],
Elfried Salanon [ctb],
Etienne Jules [aut, cre],
INRAE (Institut national de recherche pour l'agriculture,
l'alimentation et l'environnement) [cph]

Maintainer Etienne Jules <`etienne.jules@inrae.fr`>

Repository CRAN

Date/Publication 2025-10-16 12:10:07 UTC

Contents

`calculate_convex_hull` *Calculate Convex Hulls for one variable*

Description

Calculate Convex Hulls for one variable

Usage

```
calculate_convex_hull(data, var_name, impute_method = c("mean", "median"))
```

Arguments

`data`	Data frame containing the 'batch', 'order' and variable 'value' columns.
`var_name`	Name of the variable to calculate convex hull for.
`impute_method`	One of "mean" or "median".

Value

A list of dataframes of convex hull.

`calculate_convex_indicators` *Calculate the intra/inter batch dispersion indicators and their ratio on convex hulls of a single variable.*

Description

Calculate the intra/inter batch dispersion indicators and their ratio on convex hulls of a single variable.

Usage

```
calculate_convex_indicators(hull_data_list, var_name)
```

Arguments

`hull_data_list`	list of data frames of convex hulls.
`var_name`	name of the variable.

Value

A data frame with the indicators values.

`compute_icm_distances` *Compute ICM (Integrated Covariance Mahalanobis) Distances*

Description

This function computes Mahalanobis distances in PCA-reduced space, with options for individual, intra-group, and inter-group comparisons. It supports batch-wise analysis and shrinkage covariance estimation for robustness.

Usage

```
compute_icm_distances(
  data,
  batch_col = NULL,
  mode = c("individual", "intra", "inter", "all"),
  variance_threshold = 0.95,
  center_method_individual = c("global", "batch"),
  center_method_inter = c("mean", "median"),
  ref_batch = NULL
)
```

Arguments

`data`	A data.frame containing numeric variables and optionally a batch/group column.
`batch_col`	Name of the column representing batch or group (optional).
`mode`	Mode of computation: "individual", "intra", "inter", or "all".
`variance_threshold`	Threshold for cumulative variance to retain in PCA (default: 0.95).
`center_method_individual`	Method for centering in "individual" mode: "global" or "batch" (default: "global").
`center_method_inter`	Method for centering in "inter" mode: "mean" or "median" (default: "mean").
`ref_batch`	Reference batch name to compute inter-batch distances (default: first batch).

Value

A list containing data.frames of computed distances depending on the selected mode(s).

Examples

```
data <- data.frame(matrix(rnorm(100*5), ncol = 5))
data$Batch <- rep(c("A", "B", "C", "D"), each = 25)
result <- compute_icm_distances(
 data,
 batch_col = "Batch",
 mode = "all",
 center_method_individual = "batch",
 center_method_inter = "mean"
)
print(result)
```

`compute_individual` *Computes Integrated Covariance Mahalanobis (ICM) distances for individuals, in PCA-reduced space, against either global or batch-wise references.*

Description

Computes Integrated Covariance Mahalanobis (ICM) distances for individuals, in PCA-reduced space, against either global or batch-wise references.

Usage

```
compute_individual(pc_data, ref = c("global", "batch"), batch_col)
```

Arguments

`pc_data`	PCA-reduced data frame.
`ref`	Reference type: "global" for global barycenter, "batch" for batch-wise barycenters.
`batch_col`	Name of the column representing batch or group.

Value

A data frame with Mahalanobis distances for each individual against the specified reference.

`compute_individual_batch` *Computes Integrated Covariance Mahalanobis (ICM) distances of all individuals in PCA-reduced space, against their batch-wise barycenter reference.*

Description

Computes Integrated Covariance Mahalanobis (ICM) distances of all individuals in PCA-reduced space, against their batch-wise barycenter reference.

Usage

```
compute_individual_batch(pc_data, batch_col)
```

Arguments

`pc_data`	PCA-reduced data frame.
`batch_col`	Name of the column representing batch or group.

Value

A data frame with Mahalanobis distances for each individual against their batch barycenter.

`compute_individual_global` *Computes Integrated Covariance Mahalanobis (ICM) distances of all individuals in PCA-reduced space, against their global barycenter reference.*

Description

Computes Integrated Covariance Mahalanobis (ICM) distances of all individuals in PCA-reduced space, against their global barycenter reference.

Usage

```
compute_individual_global(pc_data, batch_col)
```

Arguments

`pc_data`	PCA-reduced data frame.
`batch_col`	Name of the column representing batch or group.

Value

A data frame with Mahalanobis distances for each individual against the global barycenter.

`compute_inter` *Computes Integrated Covariance Mahalanobis (ICM) distances between batches barycenters in PCA-reduced space, using a reference bacth and either mean or median for center references.*

Description

Computes Integrated Covariance Mahalanobis (ICM) distances between batches barycenters in PCA-reduced space, using a reference bacth and either mean or median for center references.

Usage

```
compute_inter(
  pc_data,
  batch_col,
  ref_batch,
  center_method = c("mean", "median")
)
```

Arguments

`pc_data`	PCA-reduced data frame.
`batch_col`	Name of the column representing batch or group.
`ref_batch`	Name of the reference batch for distance computation.
`center_method`	Method for centering: "mean" or "median".

Value

A data frame with Mahalanobis distances for each batch against the reference.

`compute_inter_batch_dispersion` *Calculate the inter batch dispersion indicator on convex hulls of a single variable*

Description

Calculate the inter batch dispersion indicator on convex hulls of a single variable

Usage

```
compute_inter_batch_dispersion(hull_data_shoelace_list)
```

Arguments

`hull_data_shoelace_list`	named list of convex hulls data frames with an additional column of shoelace core

Value

value of inter batch dispersion.

`compute_intra` *Computes Integrated Covariance Mahalanobis (ICM) mean distances within each batch in PCA-reduced space, using median and mean for center references.*

Description

Computes Integrated Covariance Mahalanobis (ICM) mean distances within each batch in PCA-reduced space, using median and mean for center references.

Usage

```
compute_intra(pc_data, batch_col)
```

Arguments

`pc_data`	PCA-reduced data frame.
`batch_col`	Name of the column representing batch or group.

Value

A data frame with Mahalanobis distances mean for each batch.

`compute_intra_batch_dispersion`

Calculate the intra batch dispersion indicator on convex hulls of a single variable

Description

Calculate the intra batch dispersion indicator on convex hulls of a single variable

Usage

```
compute_intra_batch_dispersion(hull_data_shoelace_list)
```

Arguments

`hull_data_shoelace_list`	named list of convex hulls data frames with an additional column of shoelace core values, for each batch.

Value

value of intra batch dispersion.

`compute_ratio` *Calculate the intra/inter batch dispersion ratio indicator on convex hulls of a single variable.*

Description

Calculate the intra/inter batch dispersion ratio indicator on convex hulls of a single variable.

Usage

```
compute_ratio(intraB_disp, interB_disp)
```

Arguments

`intraB_disp`	value of intra batch dispersion indicator.
`interB_disp`	value of inter batch dispersion indicator.

Value

value of intra/inter batch dispersion ratio.

`compute_shoelace_core` *Compute the shoelace core for convex hulls of a single variable*

Description

Compute the shoelace core for convex hulls of a single variable

Usage

```
compute_shoelace_core(hull_data_list)
```

Arguments

`hull_data_list`	named list of data frames of convex hulls, for each batch.

Value

named list of dataframes of convex hull concatenated with a column of shoelace core values, for each batch.

`convex_analysis_of_variables` *Analyze a set of variables using convex hulls.*

Description

Analyze a set of variables using convex hulls.

Usage

```
convex_analysis_of_variables(
  data,
  variable_columns,
  batch_col = "batch",
  sample_order_col = "order",
  impute_if_needed = c("median", "mean"),
  mode = c("global", "batchwise")
)
```

Arguments

`data`	Data frame containing the data of multiple variable on multiple ordered and potentially batched sample.
`variable_columns`	Character vector of variable column names to analyse.
`batch_col`	Name of the column containing batch information.
`sample_order_col`	Name of the column containing the sample time order.
`impute_if_needed`	Method for imputing missing values, either "mean" or "median".
`mode`	Analysis mode, either "global" or "batchwise"

Value

A list containing the following elements:

- data: List of data frames for each variable.
- indicators: Data frame with convex hull indicators for each variable.
- convex_hulls: List of data frames of convex hulls for each varaible.

Examples

```
# Example usage on toy metabolomics data:
data <- data.frame(
  batch = rep(c("A","B","C"), each = 10),
  injectionOrder = rep(1:30, times = 1),
```

```
  metabolite1 = rnorm(30, mean = 100, sd = 10),
  metabolite2 = rnorm(30, mean = 200, sd = 20)
)
result <- convex_analysis_of_variables(
  data = data,
  variable_columns = c("metabolite1", "metabolite2"),
  batch_col = "batch",
  sample_order_col = "injectionOrder",
  impute_if_needed = "median",
  mode = "global"
)
plot_all_convex_hulls(
  target_file_path = file.path(tempdir(), "convex_hulls.pdf"),
  convex_analysis_res = result,
  show_points = TRUE,
  mode = "global"
)
```

`hull_data_list_check` *Function to check if hull_data_list is a valid list of data frames*

Description

Function to check if hull_data_list is a valid list of data frames

Usage

```
hull_data_list_check(hull_data_list, name)
```

Arguments

`hull_data_list`	List of data frames representing convex hulls.
`name`	Name of the hull_data_list for error messages.

Value

None. The function raises an error if the checks fail.

`plot_all_convex_hulls` *Plot all convex hulls for each variable in a PDF file.*

Description

Plot all convex hulls for each variable in a PDF file.

Usage

```
plot_all_convex_hulls(
  target_file_path,
  convex_analysis_res,
  show_points,
  mode = c("global", "batchwise")
)
```

Arguments

`target_file_path`	Path to the output PDF file.
`convex_analysis_res`	Result of the convex analysis containing data, convex hulls and indicators.
`show_points`	Boolean indicating whether to show points in the plot.
`mode`	Mode of the analysis, either "global" or "batchwise".

Value

None. The function saves the plots to a PDF file.

`plot_convex_hull` *Plot the convex hulls of a single variable.*

Description

Plot the convex hulls of a single variable.

Usage

```
plot_convex_hull(
  data,
  hull_data_list,
  var_name,
  show_points,
  label_prefix,
  indicators
)
```

Arguments

`data`	Data frame containing the batch, order and variable value columns.
`hull_data_list`	List of data frames of convex hulls.
`var_name`	Name of the variable.
`show_points`	Boolean indicating whether to show points.
`label_prefix`	Prefix for the plot title.
`indicators`	Data frame with the indicators values.

Value

A ggplot object.

`save_icm_distances_csv` *Save ICM Distances to CSV Files*

Description

Save ICM Distances to CSV Files

Usage

```
save_icm_distances_csv(distances, folder_path, prefix = "ICM")
```

Arguments

`distances`	A list containing data.frames of distances (result from `compute_icm_distances`)
`folder_path`	Path to the folder where files will be saved.
`prefix`	Prefix for the output file names.

Value

None. Saves files to folder_path.

`single_variable_df_check` *Function to check if a single variable data frame is valid*

Description

Function to check if a single variable data frame is valid

Usage

```
single_variable_df_check(df, name)
```

Arguments

`df`	Data frame containing 'batch', 'order', and 'value' columns.
`name`	Name of the data frame for error messages.

Value

None. The function raises an error if the checks fail.

Index