# Package 'CytoProfile'

December 6, 2025

**Title** Cytokine Profiling Analysis Tool

**Version** 0.2.3

**Description** Provides comprehensive cytokine profiling analysis through quality control using biologically meaningful cutoffs on raw cytokine measurements and by testing for distributional symmetry to recommend appropriate transformations. Offers exploratory data analysis with summary statistics, enhanced boxplots, and barplots, along with univariate and multivariate analytical capabilities for in-depth cytokine profiling such as Principal Component Analysis based on Andrzej Maćkiewicz and Waldemar Ratajczak (1993) <doi:10.1016/0098-3004(93)90090-R>, Sparse Partial Least Squares Discriminant Analysis based on Lê Cao K-A, Boitard S, and Besse P (2011) <doi:10.1186/1471-2105-12-253>, Random Forest based on Breiman, L. (2001) <doi:10.1023/A:1010933404324>, and Extreme Gradient Boosting based on Tianqi Chen and Carlos Guestrin (2016) <doi:10.1145/2939672.2939785>.

**Encoding** UTF-8

**RoxygenNote** 7.3.3

**URL** https://github.com/saraswatsh/CytoProfile, https://cytoprofile.cytokineprofile.org/

**Depends** R (>= 4.3)

**Imports** mixOmics, dplyr, tidyr, pROC, plot3D, caret, xgboost, randomForest, pheatmap, e1071, ggplot2, ggrepel, gridExtra, reshape2, lifecycle

**Suggests** spelling, BiocManager, testthat, knitr, rmarkdown, devtools, Ckmeans.1d.dp, prodlim

**NeedsCompilation** no

**License** GPL (>= 2)

**LazyData** true

**VignetteBuilder** knitr

**BugReports** https://github.com/saraswatsh/CytoProfile/issues

**Language** en-US

**Author** Shubh Saraswat [cre, aut, cph] (ORCID: <https://orcid.org/0009-0009-2359-1484>), Xiaohua Douglas Zhang [aut] (ORCID: <https://orcid.org/0000-0002-2486-7931>)

**Maintainer** Shubh Saraswat <shubh.saraswat00@gmail.com>

**Repository** CRAN

**Date/Publication** 2025-12-06 16:10:07 UTC

# Contents

cyt_anova . . . . . 2
cyt_bp . . . . . 3
cyt_bp2 . . . . . 4
cyt_dualflashplot . . . . . 5
cyt_errbp . . . . . 6
cyt_heatmap . . . . . 8
cyt_mint_splsda . . . . . 9
cyt_pca . . . . . 11
cyt_rf . . . . . 13
cyt_skku . . . . . 15
cyt_splsda . . . . . 16
cyt_ttest . . . . . 19
cyt_volc . . . . . 20
cyt_xgb . . . . . 21
ExampleData1 . . . . . 24
ExampleData2 . . . . . 25
ExampleData3 . . . . . 26
ExampleData4 . . . . . 27
ExampleData5 . . . . . 28

**Index** **31**

---

`cyt_anova` *ANOVA Analysis on Continuous Variables.*

---

## Description

This function performs an analysis of variance (ANOVA) for each continuous variable against every categorical predictor in the input data. Character columns are automatically converted to factors; all factor columns are used as predictors while numeric columns are used as continuous outcomes. For each valid predictor (i.e., with more than one level and no more than 10 levels), Tukey's Honest Significant Difference (HSD) test is conducted and the adjusted p-values for pairwise comparisons are extracted.

## Usage

```
cyt_anova(data, format_output = FALSE)
```

### Arguments

| | |
|---|---|
| data | A data frame or matrix containing both categorical and continuous variables. Character columns will be converted to factors and used as predictors, while numeric columns will be used as continuous outcomes. |
| format_output | Logical. If TRUE, returns the results as a tidy data frame instead of a list. Default is FALSE. |

### Value

If format_output is FALSE (default), a list of adjusted p-values from Tukey's HSD tests for each combination of continuous outcome and categorical predictor. List elements are named in the format "Outcome_Categorical". If format_output is TRUE, a data frame in a tidy format.

### Author(s)

Shubh Saraswat

### Examples

```
data("ExampleData1")
cyt_anova(ExampleData1[, c(1:2, 5:6)], format_output = TRUE)
```

---

## cyt_bp *Boxplots for Overall Comparisons by Continuous Variables.*

---

### Description

This function creates a PDF file containing box plots for the continuous variables in the provided data. If the number of columns in data exceeds bin.size, the function splits the plots across multiple pages.

### Usage

```
cyt_bp(data, pdf_title, bin_size = 25, y_lim = NULL, scale = NULL)
```

### Arguments

| | |
|---|---|
| data | A matrix or data frame containing the raw data to be plotted. |
| pdf_title | A string representing the name of the PDF file to be created. If set to NULL, the box plots are displayed on the current graphics device. Default is NULL. |
| bin_size | An integer specifying the maximum number of box plots to display on a single page. |
| y_lim | An optional numeric vector defining the y-axis limits for the plots. |
| scale | An optional character string. If set to "log2", numeric columns are log2-transformed. |

## Value

A PDF file containing the box plots for the continuous variables.

## Author(s)

Shubh Saraswat

## Examples

```
# Loading data
data.df <- ExampleData1
# Generate box plots for log2-transformed values to check for outliers:
cyt_bp(data.df[,-c(1:3)], pdf_title = NULL, scale = "log2")
```

---

# cyt_bp2 *Boxplot Function Enhanced for Specific Group Comparisons.*

---

## Description

This function generates a PDF file containing boxplots for each combination of numeric and factor variables in the provided data. It first converts any character columns to factors and checks that the data contains at least one numeric and one factor column. If the scale argument is set to "log2", all numeric columns are log2-transformed. The function then creates boxplots using ggplot2 for each numeric variable grouped by each factor variable.

## Usage

```
cyt_bp2(data, pdf_title, scale = NULL, y_lim = NULL)
```

## Arguments

| | |
|---|---|
| data | A matrix or data frame of raw data. |
| pdf_title | A string representing the title (and filename) of the PDF file. If NULL, the boxplots are displayed on the current graphics device. Defaults to NULL. |
| scale | Transformation option for continuous variables. Options are NULL (default) and "log2". When set to "log2", numeric columns are transformed using the log2 function. |
| y_lim | An optional numeric vector defining the y-axis limits for the plots. |

## Value

A PDF file containing the boxplots.

## Author(s)

Shubh Saraswat

## Examples

```
# Loading data
data_df <- ExampleData1[, -c(3, 5:28)]
data_df <- dplyr::filter(data_df, Group == "T2D", Treatment == "Unstimulated")
cyt_bp2(data_df, pdf_title = NULL, scale = "log2")
```

---

`cyt_dualflashplot` *Dual-flashlight Plot.*

---

## Description

This function reshapes the input data and computes summary statistics (mean and variance) for each variable grouped by a specified factor column. It then calculates the SSMD (Strictly Standardized Mean Difference) and log2 fold change between two groups (group1 and group2) and categorizes the effect strength as "Strong Effect", "Moderate Effect", or "Weak Effect". A dual flash plot is generated using ggplot2 where the x-axis represents the average log2 fold change and the y-axis represents the SSMD. Additionally, the function prints the computed statistics to the console.

## Usage

```
cyt_dualflashplot(
  data,
  group_var,
  group1,
  group2,
  ssmd_thresh = 1,
  log2fc_thresh = 1,
  top_labels = 15,
  verbose = FALSE
)
```

## Arguments

| | |
|---|---|
| `data` | A data frame containing the input data. |
| `group_var` | A string specifying the name of the grouping column in the data. |
| `group1` | A string representing the name of the first group for comparison. |
| `group2` | A string representing the name of the second group for comparison. |
| `ssmd_thresh` | A numeric threshold for the SSMD value used to determine significance. Default is 1. |
| `log2fc_thresh` | A numeric threshold for the log2 fold change used to determine significance. Default is 1. |
| `top_labels` | An integer specifying the number of top variables (based on absolute SSMD) to label in the plot. Default is 15. |
| `verbose` | A logical indicating whether to print the computed statistics to the console. Default is `FALSE`. |

## Value

A ggplot object representing the dual flash plot for the comparisons between group1 and group2.

## Author(s)

Xiaohua Douglas Zhang and Shubh Saraswat

## Examples

```
# Loading data
data_df <- ExampleData1[, -c(2:3)]

cyt_dualflashplot(
  data_df,
  group_var = "Group",
  group1 = "T2D",
  group2 = "ND",
  ssmd_thresh = -0.2,
  log2fc_thresh = 1,
  top_labels = 10,
  verbose = FALSE
)
```

---

# cyt_errbp *Error-bar Plot.*

---

## Description

This function generates an error-bar plot to visually compare different groups against a designated baseline group. It displays the central tendency (mean or median) as a bar and overlays error bars to represent the data's spread (e.g., standard deviation, MAD, or standard error). The plot can also include p-value and effect size labels (based on SSMD), presented either as symbols or numeric values, to highlight significant differences and the magnitude of effects.

## Usage

```
cyt_errbp(
  data,
  group_col = NULL,
  p_lab = FALSE,
  es_lab = FALSE,
  class_symbol = TRUE,
  x_lab = "",
  y_lab = "",
  title = "",
  log2 = FALSE,
  output_file = NULL
)
```

## Arguments

| | |
|---|---|
| data | A data frame containing the data for each group. It should include at least one numeric column for the measurements and a column specifying the group membership. |
| group_col | Character. The name of the column in data that specifies the group membership. |
| p_lab | Logical. If TRUE, p-values are displayed on the plot. Default is FALSE. |
| es_lab | Logical. If TRUE, effect sizes (SSMD) are displayed on the plot. Default is FALSE. |
| class_symbol | Logical. If TRUE, significance and effect size are represented using symbolic notation (e.g., *, **, >, «). If FALSE, numeric values are used. Default is TRUE. |
| x_lab | Character. Label for the x-axis. If not provided, defaults to the name of the group_col or "Group" if group_col is NULL. |
| y_lab | Character. Label for the y-axis. If not provided, defaults to "Value". |
| title | Character. Title of the plot. If not provided, a default title is generated based on the measured variables. |
| log2 | Logical. If TRUE, a log2 transformation (with a +1 offset) is applied to all numeric columns before analysis. Default is FALSE. |
| output_file | Character. The file path to save the plot as a PDF. If NULL, the plot is displayed but not saved. Default is NULL. |

## Details

The function performs the following steps:

1. Optionally applies a log2 transformation to numeric data.
2. Determines the baseline group (the first level of group_col).
3. Calculates summary statistics (sample size, mean, standard deviation) for each group and each numeric variable.
4. Performs t-tests to compare each group against the baseline for each numeric variable.
5. Computes effect sizes (SSMD) for each group compared to the baseline.
6. Generates a faceted error-bar plot, with one facet per numeric variable.
7. Optionally adds p-value and effect size labels to the plot.
8. Optionally saves the plot as a PDF.

## Value

An error-bar plot (a ggplot object) is produced and optionally saved as a PDF. If output_file is specified, the function returns returns the ggplot object.

## Author(s)

Xiaohua Douglas Zhang and Shubh Saraswat

## Examples

```
data <- ExampleData1

cyt_errbp(data[,c("Group", "CCL.20.MIP.3A", "IL.10")], group_col = "Group",
p_lab = TRUE, es_lab = TRUE, class_symbol = TRUE, x_lab = "Cytokines",
y_lab = "Concentrations in log2 scale", log2 = TRUE)
```

---

`cyt_heatmap` *Heat Map.*

---

## Description

This function creates a heatmap using the numeric columns from the provided data frame. It supports various scaling options and allows for row or column annotations. The heatmap is saved as a file, with the format determined by the file extension in `title`.

## Usage

```
cyt_heatmap(
  data,
  scale = c(NULL, "log2", "row_zscore", "col_zscore"),
  annotation_col = NULL,
  annotation_side = c("auto", "row", "col"),
  title = NULL
)
```

## Arguments

| | |
|---|---|
| `data` | A data frame containing the input data. Only numeric columns will be used to generate the heatmap. |
| `scale` | Character. An optional scaling option. Options are NULL (no scaling), "log2" (log2 transformation), "row_zscore" (z-score scaling by row), or "col_zscore" (z-score scaling by column). Default is NULL. |
| `annotation_col` | Character. An optional column name from `data` to be used for generating annotation colors. Default is NULL. |
| `annotation_side` | Character. Specifies whether the annotation should be applied to rows or columns. Options are "auto", "row", or "col". |
| `title` | Character. The title of the heatmap and the file name for saving the plot. The file extension (".pdf" or ".png") determines the output format. If `NULL`, the plot is generated on the current graphics device. Default is `NULL`. |

## Value

The function does not return a value. It saves the heatmap to a file.

## Author(s)

Shubh Saraswat

## Examples

```
# Load sample data
data("ExampleData1")
data_df <- ExampleData1
# Generate a heatmap with log2 scaling and annotation based on
# the "Group" column
cyt_heatmap(
  data = data_df[, -c(2:3)],
  scale = "log2",  # Optional scaling
  annotation_col = "Group",
  annotation_side = "auto",
  title = NULL
)
```

---

`cyt_mint_splsda` *Analyze data with Multivariate INTegration (MINT) Sparse Partial Least Squares Discriminant Analysis (sPLS-DA).*

---

## Description

This function performs a MINT (Multivariate INTegrative) sPLS-DA to handle batch effects by modeling a global biological signal across different studies or batches. If a second grouping column (`group_col2`) is provided, the analysis is stratified and performed for each level of that column.

## Usage

```
cyt_mint_splsda(
  data,
  group_col,
  batch_col,
  group_col2 = NULL,
  colors = NULL,
  pdf_title = NULL,
  ellipse = TRUE,
  bg = FALSE,
  var_num = 20,
  comp_num = 2,
  scale = NULL,
  cim = FALSE,
  roc = FALSE,
  verbose = FALSE
)
```

## Arguments

| | |
|---|---|
| data | A matrix or data frame containing the variables. Columns not specified by group_col, group_col2, or multilevel_col are assumed to be continuous variables for analysis. |
| group_col | A string specifying the first grouping column name that contains grouping information. If group_col2 is not provided, it will be used for both grouping and treatment. |
| batch_col | A string specifying the column that identifies the batch or study for each sample. |
| group_col2 | A string specifying the second grouping column name. Default is NULL. |
| colors | A vector of splsda_colors for the groups or treatments. If NULL, a random palette (using rainbow) is generated based on the number of groups. |
| pdf_title | A string specifying the file name for saving the PDF output. If set to NULL, the function runs in IDE plots pane. |
| ellipse | Logical. Whether to draw a 95\ Default is FALSE. |
| bg | Logical. Whether to draw the prediction background in the figures. Default is FALSE. |
| var_num | Numeric. The number of variables to be used in the PLS-DA model. |
| comp_num | Numeric. The number of components to calculate in the sPLS-DA model. Default is 2. |
| scale | Character. Option for data transformation; if set to "log2", a log2 transformation is applied to the continuous variables. Default is NULL. |
| cim | Logical. Whether to compute and plot the Clustered Image Map (CIM) heatmap. Default is FALSE. |
| roc | Logical. Whether to compute and plot the ROC curve for the model. Default is FALSE. |
| verbose | A logical value indicating whether to print additional informational output to the console. When TRUE, the function will display progress messages, and intermediate results when FALSE (the default), it runs quietly. |

## Details

When verbose is set to TRUE, additional information about the analysis and confusion matrices are printed to the console. These can be suppressed by keeping verbose = FALSE.

## Value

Plots consisting of the classification figures, ROC curves, correlation circle plots, and heatmaps.

## Author(s)

Shubh Saraswat

## References

Rohart F, Eslami A, Matigian, N, Bougeard S, Lê Cao K-A (2017). MINT: A multivariate integrative approach to identify a reproducible biomarker signature across multiple experiments and platforms. BMC Bioinformatics 18:128.

## Examples

```
# Loading ExampleData5 dataset with batch column
data_df <- ExampleData5[,-c(2,4)]
data_df <- dplyr::filter(data_df, Group != "ND")

cyt_mint_splsda(data_df, group_col = "Group",
 batch_col = "Batch", colors = c("black", "purple"),
 ellipse = TRUE, var_num = 25, comp_num = 2,
 scale = "log2", verbose = FALSE)
```

---

`cyt_pca` *Analyze Data with Principal Component Analysis (PCA) for Cytokines.*

---

## Description

This function performs Principal Component Analysis (PCA) on cytokine data and generates several types of plots, including:

- 2D PCA plots using mixOmics' `plotIndiv` function,
- 3D scatter plots (if `style` is "3d" or "3D" and `comp_num` is 3) via the plot3D package,
- Scree plots showing both individual and cumulative explained variance,
- Loadings plots, and
- Biplots and correlation circle plots.

The function optionally applies a log2 transformation to the numeric data and handles analyses based treatment groups.

## Usage

```
cyt_pca(
  data,
  group_col = NULL,
  group_col2 = NULL,
  colors = NULL,
  pdf_title,
  ellipse = FALSE,
  comp_num = 2,
  scale = NULL,
  pch_values = NULL,
  style = NULL
)
```

## Arguments

| | |
|---|---|
| data | A data frame containing cytokine data. It should include at least one column representing grouping information and optionally a second column representing treatment or stimulation. |
| group_col | A string specifying the column name that contains the first group information. If group_col2 is not provided, an overall analysis will be performed. |
| group_col2 | A string specifying the second grouping column. Default is NULL. |
| colors | A vector of colors corresponding to the groups. If set to NULL, a palette is generated using rainbow() based on the number of unique groups. |
| pdf_title | A string specifying the file name of the PDF where the PCA plots will be saved. If NULL, the plots are generated on the current graphics device. Default is NULL. |
| ellipse | Logical. If TRUE, a 95% confidence ellipse is drawn on the PCA individuals plot. Default is FALSE. |
| comp_num | Numeric. The number of principal components to compute and display. Default is 2. |
| scale | Character. If set to "log2", a log2 transformation is applied to the numeric cytokine measurements (excluding the grouping columns). Default is NULL. |
| pch_values | A vector of plotting symbols (pch values) to be used in the PCA plots. Default is NULL. |
| style | Character. If set to "3d" or "3D" and comp_num equals 3, a 3D scatter plot is generated using the plot3D package. Default is NULL. |

## Value

A PDF file containing the PCA plots is generated and saved.

## Author(s)

Shubh Saraswat

## Examples

```
# Load sample data
data <- ExampleData1[, -c(3,23)]
data_df <- dplyr::filter(data, Group != "ND" & Treatment != "Unstimulated")
# Run PCA analysis and save plots to a PDF file
cyt_pca(
  data = data_df,
  pdf_title = NULL,
  colors = c("black", "red2"),
  scale = "log2",
  comp_num = 3,
  pch_values = c(16, 4),
  style = "3D",
  group_col = "Group",
  group_col2 = "Treatment",
  ellipse = FALSE
```

```
)
```

---

cyt_rf *Run Random Forest Classification on Cytokine Data,*

---

## Description

This function trains and evaluates a Random Forest classification model on cytokine data. It includes feature importance visualization, cross- validation for feature selection, and performance metrics such as accuracy, sensitivity, and specificity. Optionally, for binary classification, the function also plots the ROC curve and computes the AUC.

## Usage

```
cyt_rf(
  data,
  group_col,
  ntree = 500,
  mtry = 5,
  train_fraction = 0.7,
  plot_roc = FALSE,
  k_folds = 5,
  step = 0.5,
  run_rfcv = TRUE,
  verbose = FALSE,
  seed = 123
)
```

## Arguments

| | |
|---|---|
| data | A data frame containing the cytokine data, with one column as the grouping variable and the rest as numerical features. |
| group_col | A string representing the name of the column with the grouping variable (the target variable for classification). |
| ntree | An integer specifying the number of trees to grow in the forest (default is 500). |
| mtry | An integer specifying the number of variables randomly selected at each split (default is 5). |
| train_fraction | A numeric value between 0 and 1 representing the proportion of data to use for training (default is 0.7). |
| plot_roc | A logical value indicating whether to plot the ROC curve and compute the AUC for binary classification (default is FALSE). |
| k_folds | An integer specifying the number of folds for cross-validation (default is 5). |
| step | A numeric value specifying the fraction of variables to remove at each step during cross-validation for feature selection (default is 0.5). |

| | |
|---|---|
| run_rfcv | A logical value indicating whether to run Random Forest cross-validation for feature selection (default is TRUE). |
| verbose | A logical value indicating whether to print additional informational output to the console. When TRUE, the function will display progress messages, and intermediate results when FALSE (the default), it runs quietly. |
| seed | An integer specifying the seed for reproducibility (default is 123). |

## Details

The function fits a Random Forest model to the provided data by splitting it into training and test sets. It calculates performance metrics such as accuracy, sensitivity, and specificity for both sets. For binary classification, it can also plot the ROC curve and compute the AUC. If run_rfcv is TRUE, cross-validation is performed to select the optimal number of features. If verbose is TRUE, the function prints additional information to the console, including training results, test results, and plots.

## Value

A list containing:

| | |
|---|---|
| model | The trained Random Forest model. |
| confusion_matrix | The confusion matrix of the test set predictions. |
| importance_plot | A ggplot object showing the variable importance plot based on Mean Decrease Gini. |
| rfcv_result | Results from Random Forest cross-validation for feature selection (if run_rfcv is TRUE). |
| importance_data | A data frame containing the variable importance based on the Gini index. |

## Author(s)

Shubh Saraswat

## Examples

```
data.df0 <- ExampleData1
data.df <- data.frame(data.df0[, 1:3], log2(data.df0[, -c(1:3)]))
data.df <- data.df[, -c(2:3)]
data.df <- dplyr::filter(data.df, Group != "ND")

cyt_rf(
  data = data.df, group_col = "Group", k_folds = 5, ntree = 1000,
  mtry = 4, run_rfcv = TRUE, plot_roc = TRUE, verbose = FALSE
)
```

cyt_skku *Distribution of the Data Set Shown by Skewness and Kurtosis.*

## Description

This function computes summary statistics — including sample size, mean, standard error, skewness, and kurtosis — for each numeric measurement column in a data set. If grouping columns are provided via `group_cols`, the function computes the metrics separately for each group defined by the combination of these columns (using the first element as the treatment variable and the second as the grouping variable, or the same column for both if only one is given). When no grouping columns are provided, the entire data set is treated as a single group ("Overall"). A log2 transformation (using a cutoff equal to one-tenth of the smallest positive value in the data) is applied to generate alternative metrics. Histograms showing the distribution of skewness and kurtosis for both raw and log2-transformed data are then generated and saved to a PDF if a file name is provided.

## Usage

```
cyt_skku(
  data,
  group_cols = NULL,
  pdf_title = NULL,
  print_res_raw = FALSE,
  print_res_log = FALSE
)
```

## Arguments

| | |
|---|---|
| `data` | A matrix or data frame containing the raw data. If `group_cols` is specified, the columns with names in `group_cols` are treated as grouping variables and all other columns are assumed to be numeric measurement variables. |
| `group_cols` | A character vector specifying the names of the grouping columns. When provided, the first element is treated as the treatment variable and the second as the group variable. If not provided, the entire data set is treated as one group. |
| `pdf_title` | A character string specifying the file name for the PDF file in which the histograms will be saved. If `NULL`, the histograms are displayed on the current graphics device. Default is `NULL`. |
| `print_res_raw` | Logical. If `TRUE`, the function returns and prints the computed summary statistics for the raw data. Default is `FALSE`. |
| `print_res_log` | Logical. If `TRUE`, the function returns and prints the computed summary statistics for the log2-transformed data. Default is `FALSE`. |

## Details

A cutoff is computed as one-tenth of the minimum positive value among all numeric measurement columns to avoid taking logarithms of zero. When grouping columns are provided, the function loops over unique grouping columns and computes the metrics for each measurement column within each subgroup. Without grouping columns, the entire data set is analyzed as one overall group.

## Value

The function generates histograms of skewness and kurtosis for both raw and log2-transformed data. Additionally, if either `printResRaw` and/or `printResLog` is `TRUE`, the function returns the corresponding summary statistics as a data frame or a list of data frames.

## Author(s)

Xiaohua Douglas Zhang and Shubh Saraswat

## Examples

```
# Example with grouping columns (e.g., "Group" and "Treatment")
data(ExampleData1)
cyt_skku(ExampleData1[, -c(2:3)], pdf_title = NULL,
  group_cols = c("Group")
)

# Example without grouping columns (analyzes the entire data set)
cyt_skku(ExampleData1[, -c(1:3)], pdf_title = NULL)
```

---

`cyt_splsda` *Analyze data with Sparse Partial Least Squares Discriminant Analysis (sPLS-DA).*

---

## Description

This function conducts Sparse Partial Least Squares Discriminant Analysis (sPLS-DA) on the provided data. It uses the specified `group_col` (and optionally `group_col2`) to define class labels while assuming the remaining columns contain continuous variables. The function supports a log2 transformation via the `scale` parameter and generates a series of plots, including classification plots, scree plots, loadings plots, and VIP score plots. Optionally, ROC curves are produced when `roc` is `TRUE`. Additionally, cross-validation is supported via LOOCV or Mfold methods. When both `group_col` and `group_col2` are provided and differ, the function analyzes each treatment level separately.

## Usage

```
cyt_splsda(
  data,
  group_col = NULL,
  group_col2 = NULL,
  multilevel_col = NULL,
  batch_col = NULL,
  ind_names = FALSE,
  colors = NULL,
  pdf_title = NULL,
  ellipse = FALSE,
```

```
  bg = FALSE,
  conf_mat = FALSE,
  var_num,
  cv_opt = NULL,
  fold_num = 5,
  scale = NULL,
  comp_num = 2,
  pch_values,
  style = NULL,
  roc = FALSE,
  verbose = FALSE,
  seed = 123
)
```

## Arguments

| | |
|---|---|
| data | A matrix or data frame containing the variables. Columns not specified by group_col or group_col2 are assumed to be continuous variables for analysis. |
| group_col | A string specifying the column name that contains the first group information. If group_col2 is not provided, an overall analysis will be performed. |
| group_col2 | A string specifying the second grouping column. Default is NULL. |
| multilevel_col | A string specifying the column name that identifies repeated measurements (e.g., patient or sample IDs). If provided, a multilevel analysis will be performed. Default is NULL. |
| batch_col | A string specifying the column that identifies the batch or study for each sample. |
| ind_names | If TRUE, the row names of the first (or second) data matrix is used as names. Default is FALSE. If a character vector is provided, these values will be used as names. If 'pch' is set this will overwrite the names as shapes. See ?mixOmics::plotIndiv for details. |
| colors | A vector of colors for the groups or treatments. If NULL, a random palette (using rainbow) is generated based on the number of groups. |
| pdf_title | A string specifying the file name for saving the PDF output. Default is NULL which generates figures in the current graphics device. |
| ellipse | Logical. Whether to draw a 95\ figures. Default is FALSE. |
| bg | Logical. Whether to draw the prediction background in the figures. Default is FALSE. |
| conf_mat | Logical. Whether to print the confusion matrix for the classifications. Default is FALSE. |
| var_num | Numeric. The number of variables to be used in the PLS-DA model. |
| cv_opt | Character. Option for cross-validation method: either "loocv" or "Mfold". Default is NULL. |
| fold_num | Numeric. The number of folds to use if cv_opt is "Mfold". Default is 5. |
| scale | Character. Option for data transformation; if set to "log2", a log2 transformation is applied to the continuous variables. Default is NULL. |

| | |
|---|---|
| comp_num | Numeric. The number of components to calculate in the sPLS-DA model. Default is 2. |
| pch_values | A vector of integers specifying the plotting characters (pch values) to be used in the plots. |
| style | Character. If set to "3D" or "3d" and comp_num equals 3, a 3D plot is generated using the plot3D package. Default is NULL. |
| roc | Logical. Whether to compute and plot the ROC curve for the model. Default is FALSE. |
| verbose | A logical value indicating whether to print additional informational output to the console. When TRUE, the function will display progress messages, and intermediate results when FALSE (the default), it runs quietly. |
| seed | An integer specifying the seed for reproducibility (default is 123). |

## Details

When verbose is set to TRUE, additional information about the analysis and confusion matrices are printed to the console. These can be suppressed by keeping verbose = FALSE.

## Value

Plots consisting of the classification figures, component figures with Variable of Importance in Projection (VIP) scores, and classifications based on VIP scores greater than 1. ROC curves and confusion matrices are also produced if requested.

## Author(s)

Xiaohua Douglas Zhang and Shubh Saraswat

## References

Lê Cao, K.-A., Boitard, S. and Besse, P. (2011). Sparse PLS Discriminant Analysis: biologically relevant feature selection and graphical displays for multiclass problems. *BMC Bioinformatics* **12**:253.

## Examples

```
# Loading Sample Data
data_df <- ExampleData1[,-c(3)]
data_df <- dplyr::filter(data_df, Group != "ND", Treatment != "Unstimulated")

cyt_splsda(data_df, pdf_title = NULL,
colors = c("black", "purple"), bg = FALSE, scale = "log2",
conf_mat = FALSE, var_num = 25, cv_opt = NULL, comp_num = 2,
pch_values = c(16, 4), style = NULL, ellipse = TRUE,
group_col = "Group", group_col2 = "Treatment", roc = FALSE, verbose = FALSE)
```

cyt_ttest *Two Sample T-test Comparisons.*

## Description

This function performs pairwise comparisons between two groups for each combination of a categorical predictor (with exactly two levels) and a continuous outcome variable. It first converts any character variables in `data` to factors and, if specified, applies a log2 transformation to the continuous variables. Depending on the value of `scale`, the function conducts either a two-sample t-test (if `scale = "log2"`) or a Mann-Whitney U test (if `scale` is `NULL`). The resulting p-values are printed and returned.

## Usage

```
cyt_ttest(data, scale = NULL, verbose = TRUE, format_output = FALSE)
```

## Arguments

| | |
|---|---|
| `data` | A matrix or data frame containing continuous and categorical variables. |
| `scale` | A character specifying a transformation for continuous variables. Options are `NULL` (default) and `"log2"`. When `scale = "log2"`, a log2 transformation is applied and a two-sample t-test is used; when `scale` is `NULL`, a Mann-Whitney U test is performed. |
| `verbose` | A logical indicating whether to print the p-values of the statistical tests. Default is `TRUE`. |
| `format_output` | Logical. If TRUE, returns the results as a tidy data frame. Default is `FALSE`. |

## Value

If `format_output` is FALSE, returns a list of p-values (named by Outcome and Categorical variable). If TRUE, returns a data frame in a tidy format.

## Author(s)

Shubh Saraswat

## Examples

```
data_df <- ExampleData1[, -c(3)]
data_df <- dplyr::filter(data_df, Group != "ND", Treatment != "Unstimulated")
# Test example
cyt_ttest(
 data_df[, c(1:2, 5:6)],
 scale = "log2",
 verbose = TRUE,
 format_output = TRUE
)
```

cyt_volc *Volcano Plot.*

## Description

This function subsets the numeric columns from the input data and compares them based on a selected grouping column. It computes the fold changes (as the ratio of means) and associated p-values (using two-sample t-tests) for each numeric variable between two groups. The results are log2-transformed (for fold change) and -log10-transformed (for p-values) to generate a volcano plot.

## Usage

```
cyt_volc(
  data,
  group_col,
  cond1 = NULL,
  cond2 = NULL,
  fold_change_thresh = 2,
  p_value_thresh = 0.05,
  top_labels = 10,
  verbose = FALSE
)
```

## Arguments

| | |
|---|---|
| data | A matrix or data frame containing the data to be analyzed. |
| group_col | A character string specifying the column name used for comparisons (e.g., group, treatment, or stimulation). |
| cond1 | A character string specifying the name of the first condition for comparison. Default is NULL. |
| cond2 | A character string specifying the name of the second condition for comparison. Default is NULL. |
| fold_change_thresh | A numeric threshold for the fold change. Default is 2. |
| p_value_thresh | A numeric threshold for the p-value. Default is 0.05. |
| top_labels | An integer specifying the number of top variables to label on the plot. Default is 10. |
| verbose | A logical indicating whether to print the computed statistics to the console. Default is FALSE. |

## Value

A list of volcano plots (as ggplot objects) for each pairwise comparison. Additionally, the function prints the data frame used for plotting (excluding the significance column) from the final comparison.

## Note

If cond1 and cond2 are not provided, the function automatically generates all possible pairwise combinations of groups from the specified group_col for comparisons.

## Author(s)

Xiaohua Douglas Zhang and Shubh Saraswat

## Examples

```
# Loading data
data_df <- ExampleData1[,-c(2:3)]

volc_plot <- cyt_volc(data_df, "Group", cond1 = "T2D", cond2 = "ND",
fold_change_thresh = 2.0, top_labels= 15)
print(volc_plot$`T2D vs ND`)
```

---

# cyt_xgb *Run XGBoost Classification on Cytokine Data.*

---

## Description

This function trains and evaluates an XGBoost classification model on cytokine data. It allows for hyperparameter tuning, cross-validation, and visualizes feature importance.

## Usage

```
cyt_xgb(
  data,
  group_col,
  train_fraction = 0.7,
  nrounds = 500,
  max_depth = 6,
  eta = lifecycle::deprecated(),
  learning_rate = 0.1,
  nfold = 5,
  cv = FALSE,
  objective = "multi:softprob",
  early_stopping_rounds = NULL,
  eval_metric = "mlogloss",
  gamma = lifecycle::deprecated(),
  min_split_loss = 0,
  colsample_bytree = 1,
  subsample = 1,
  min_child_weight = 1,
  top_n_features = 10,
  verbose = 1,
```

```
  plot_roc = FALSE,
  print_results = FALSE,
  seed = 123
)
```

## Arguments

| | |
|---|---|
| data | A data frame containing the cytokine data, with one column as the grouping variable and the rest as numerical features. |
| group_col | A string representing the name of the column with the grouping variable (i.e., the target variable for classification). |
| train_fraction | A numeric value between 0 and 1 representing the proportion of data to use for training (default is 0.7). |
| nrounds | An integer specifying the number of boosting rounds (default is 500). |
| max_depth | An integer specifying the maximum depth of the trees (default is 6). |
| eta | **[Deprecated]** Deprecated; use learning_rate instead. |
| learning_rate | A numeric value representing the learning rate (default is 0.1). This replaces the deprecated eta argument. |
| nfold | An integer specifying the number of folds for cross-validation (default is 5). |
| cv | A logical value indicating whether to perform cross-validation (default is FALSE). |
| objective | A string specifying the XGBoost objective function (default is "multi:softprob" for multi-class classification). |
| early_stopping_rounds | An integer specifying the number of rounds with no improvement to stop training early (default is NULL). |
| eval_metric | A string specifying the evaluation metric (default is "mlogloss"). |
| gamma | **[Deprecated]** Deprecated; use min_split_loss instead. |
| min_split_loss | A numeric value for the minimum loss reduction required to make a further partition (default is 0). This replaces the deprecated gamma argument. |
| colsample_bytree | A numeric value specifying the subsample ratio of columns when constructing each tree (default is 1). |
| subsample | A numeric value specifying the subsample ratio of the training instances (default is 1). |
| min_child_weight | A numeric value specifying the minimum sum of instance weight needed in a child (default is 1). |
| top_n_features | An integer specifying the number of top features to display in the importance plot (default is 10). |
| verbose | An integer specifying the verbosity of the training process (default is 1). |
| plot_roc | A logical value indicating whether to plot the ROC curve and calculate the AUC for binary classification (default is FALSE). |
| print_results | A logical value indicating whether to print the results of the model training and evaluation (default is FALSE). If set to TRUE, it will print the confusion matrix, and feature importance. |
| seed | An integer specifying the seed for reproducibility (default is 123). |

## Details

The function allows for training an XGBoost model on cytokine data, splitting the data into training and test sets. If cross-validation is enabled (`cv = TRUE`), it performs k-fold cross-validation and reports the confusion matrix and accuracy. The function also visualizes the top N important features using `xgb.ggplot.importance()`.

## Value

A list containing:

| | |
|---|---|
| `model` | The trained XGBoost model. |
| `confusion_matrix` | The confusion matrix of the test set predictions. |
| `importance` | The feature importance matrix for the top features. |
| `class_mapping` | A named vector showing the mapping from class labels to numeric values used for training. |
| `cv_results` | Cross-validation results, if cross-validation was performed (otherwise NULL). |
| `plot` | A ggplot object showing the feature importance plot. |

## Author(s)

Shubh Saraswat

## Examples

```
# Example usage:
data_df0 <- ExampleData1
data_df <- data.frame(data_df0[, 1:3], log2(data_df0[, -c(1:3)]))
data_df <- data_df[, -c(2:3)]
data_df <- dplyr::filter(data_df, Group != "ND")

cyt_xgb(
 data = data_df,
 group_col = "Group",
 nrounds = 500,
 max_depth = 4,
 min_split_loss = 0,
 learning_rate = 0.05,
 nfold = 5,
 cv = FALSE,
 objective = "multi:softprob",
 eval_metric = "auc",
 early_stopping_rounds = NULL,
 top_n_features = 10,
 verbose = 0,
 plot_roc = TRUE,
 print_results = FALSE)
```

ExampleData1 *Example Cytokine Profiling Data 1.*

## Description

Contains observed concentrations of cytokines and their respective treatment and groups, derived from:

## Usage

```
ExampleData1
```

## Format

A data frame with 297 rows and 29 columns:

**Group** Group assigned to the subjects.
**Treatment** Treatment received by subjects.
**Time** Time point of the measurement.
**IL.17F** Observed concentration of IL.17F cytokine.
**GM.CSF** Observed concentration of GM.CSF cytokine.
**IFN.G** Observed concentration of IFN.G cytokine.
**IL.10** Observed concentration of IL.10 cytokine.
**CCL.20.MIP.3A** Observed concentration of CCL.20.MIP.3A cytokine.
**IL.12.P70** Observed concentration of IL.12.P70 cytokine.
**IL.13** Observed concentration of IL.13 cytokine.
**IL.15** Observed concentration of IL.15 cytokine.
**IL.17A** Observed concentration of IL.17A cytokine.
**IL.22** Observed concentration of IL.22 cytokine.
**IL.9** Observed concentration of IL.9 cytokine.
**IL.1B** Observed concentration of IL.1B cytokine.
**IL.33** Observed concentration of IL.33 cytokine.
**IL.2** Observed concentration of IL.2 cytokine.
**IL.21** Observed concentration of IL.21 cytokine.
**IL.4** Observed concentration of IL.4 cytokine.
**IL.23** Observed concentration of IL.23 cytokine.
**IL.5** Observed concentration of IL.5 cytokine.
**IL.6** Observed concentration of IL.6 cytokine.
**IL.17E.IL.25** Observed concentration of IL.17E.IL.25 cytokine.
**IL.27** Observed concentration of IL.27 cytokine.
**IL.31** Observed concentration of IL.31 cytokine.
**TNF.A** Observed concentration of TNF.A cytokine.
**TNF.B** Observed concentration of TNF.B cytokine.
**IL.28A** Observed concentration of IL.28A cytokine.

## Source

Example data compiled for cytokine profiling.

## References

Pugh GH, Fouladvand S, SantaCruz-Calvo S, Agrawal M, Zhang XD, Chen J, Kern PA, Nikolajczyk BS. T cells dominate peripheral inflammation in a cross-sectional analysis of obesity-associated diabetes. *Obesity (Silver Spring)*. 2022;30(10): 1983–1994. doi:10.1002/oby.23528.

## Examples

```
data(ExampleData1)
```

---

`ExampleData2` *Example Cytokine Profiling Data 2.*

---

## Description

Contains observed concentrations of cytokines and their respective treatment and groups, derived from:

## Usage

```
ExampleData2
```

## Format

A data frame with 66 rows and 20 columns:

**Stimulation** Stimulation assigned to the subjects.

**Group** Group assigned to the subjects.

**IL.17F** Observed concentration of IL.17F cytokine.

**GM.CSF** Observed concentration of GM.CSF cytokine.

**IFN.G** Observed concentration of IFN.G cytokine.

**IL.10** Observed concentration of IL.10 cytokine.

**CCL.20** Observed concentration of CCL.20 cytokine.

**IL.12** Observed concentration of IL.12 cytokine.

**IL.13** Observed concentration of IL.13 cytokine.

**IL.17A** Observed concentration of IL.17A cytokine.

**IL.22** Observed concentration of IL.22 cytokine.

**IL.9** Observed concentration of IL.9 cytokine.

**IL.1B** Observed concentration of IL.1B cytokine.

**IL.2** Observed concentration of IL.2 cytokine.

**IL.21** Observed concentration of IL.21 cytokine.

**IL.4** Observed concentration of IL.4 cytokine.

**IL.5** Observed concentration of IL.5 cytokine.

**IL.6** Observed concentration of IL.6 cytokine.

**TNF.A** Observed concentration of TNF.A cytokine.

**TNF.B** Observed concentration of TNF.B cytokine.

## Source

Example data compiled for cytokine profiling.

## References

SantaCruz-Calvo S, Saraswat S, Hasturk H, Dawson DR, Zhang XD, Nikolajczyk BS. Periodontitis and Diabetes Differentially Affect Inflammation in Obesity. *J Dent Res.* 2024;103(12):1313-1322. doi:10.1177/00220345241280743

## Examples

```
data(ExampleData2)
```

---

`ExampleData3` *Example Cytokine Profiling Data 3.*

---

## Description

Contains observed concentrations of cytokines and their respective treatment and groups, derived from:

## Usage

```
ExampleData3
```

## Format

A data frame with 64 rows and 14 columns:

**Stimulation** Stimulation assigned to the subjects.

**Group** Group assigned to the subjects.

**GM.CSF** Observed concentration of GM.CSF cytokine.

**IFN.G** Observed concentration of IFN.G cytokine.

**IL.10** Observed concentration of IL.10 cytokine.

**CCL.20.MIP.3A** Observed concentration of CCL.20.MIP.3A cytokine.

**IL.12.P70** Observed concentration of IL.12.P70 cytokine.

**IL.13** Observed concentration of IL.13 cytokine.

**IL.15** Observed concentration of IL.15 cytokine.

**IL.9** Observed concentration of IL.9 cytokine.

**IL.1B** Observed concentration of IL.1B cytokine.

**IL.21** Observed concentration of IL.21 cytokine.

**IL.6** Observed concentration of IL.6 cytokine.

**TNF.A** Observed concentration of TNF.A cytokine.

## Source

Example data compiled for cytokine profiling.

## References

SantaCruz-Calvo S, Saraswat S, Hasturk H, Dawson DR, Zhang XD, Nikolajczyk BS. Periodontitis and Diabetes Differentially Affect Inflammation in Obesity. *J Dent Res.* 2024;103(12):1313-1322. doi:10.1177/00220345241280743

## Examples

```
data(ExampleData3)
```

---

`ExampleData4` *Example Cytokine Profiling Data 4.*

---

## Description

Contains observed concentrations of cytokines and their respective treatment and groups, derived from:

## Usage

```
ExampleData4
```

## Format

A data frame with 64 rows and 14 columns:

**Group** Group assigned to the subjects.

**Treatment** Treatment received by subjects.

**IL.17F** Observed concentration of IL.17F cytokine.

**GM.CSF** Observed concentration of GM.CSF cytokine.

**IFNg** Observed concentration of IFNg cytokine.

**IL.10** Observed concentration of IL.10 cytokine.

**CCL.20** Observed concentration of CCL.20 cytokine.

**IL.12** Observed concentration of IL.12 cytokine.

**IL.13** Observed concentration of IL.13 cytokine.

**IL.17A** Observed concentration of IL.17A cytokine.

**IL.22** Observed concentration of IL.22 cytokine.

**IL.9** Observed concentration of IL.9 cytokine.

**IL.2** Observed concentration of IL.2 cytokine.

**IL.21** Observed concentration of IL.21 cytokine.

**IL.4** Observed concentration of IL.4 cytokine.

**IL.23** Observed concentration of IL.23 cytokine.

**IL.5** Observed concentration of IL.5 cytokine.

**IL.6** Observed concentration of IL.6 cytokine.

**TNFa** Observed concentration of TNFa cytokine.

**TNFb** Observed concentration of TNFb cytokine.

## Source

Example data compiled for cytokine profiling.

## References

SantaCruz-Calvo, S., Saraswat, S., Kalantar, G. H., Zukowski, E., Marszalkowski, H., Javidan, A., Gholamrezaeinejad, F., Bharath, L. P., Kern, P. A., Zhang, X. D., & Nikolajczyk, B. S. (2024). A unique inflammaging profile generated by T cells from people with obesity is metformin resistant. *GeroScience*, 10.1007/s11357-024-01441-4. Advance online publication. https://doi.org/10.1007/s11357-024-01441-4

## Examples

```
data(ExampleData4)
```

---

`ExampleData5` *Example Cytokine Profiling Data 5.*

---

## Description

Contains observed concentrations of cytokines and their respective treatment and groups, derived from:

## Usage

```
ExampleData5
```

## Format

A data frame with 297 rows and 29 columns:

**Group** Group assigned to the subjects.

**Treatment** Treatment received by subjects.

**Batch** Batch number corresponding to the sample.

**Time** Time point of the measurement.

**IL.17F** Observed concentration of IL.17F cytokine.

**GM.CSF** Observed concentration of GM.CSF cytokine.

**IFN.G** Observed concentration of IFN.G cytokine.

**IL.10** Observed concentration of IL.10 cytokine.

**CCL.20.MIP.3A** Observed concentration of CCL.20.MIP.3A cytokine.

**IL.12.P70** Observed concentration of IL.12.P70 cytokine.

**IL.13** Observed concentration of IL.13 cytokine.

**IL.15** Observed concentration of IL.15 cytokine.

**IL.17A** Observed concentration of IL.17A cytokine.

**IL.22** Observed concentration of IL.22 cytokine.

**IL.9** Observed concentration of IL.9 cytokine.

**IL.1B** Observed concentration of IL.1B cytokine.

**IL.33** Observed concentration of IL.33 cytokine.

**IL.2** Observed concentration of IL.2 cytokine.

**IL.21** Observed concentration of IL.21 cytokine.

**IL.4** Observed concentration of IL.4 cytokine.

**IL.23** Observed concentration of IL.23 cytokine.

**IL.5** Observed concentration of IL.5 cytokine.

**IL.6** Observed concentration of IL.6 cytokine.

**IL.17E.IL.25** Observed concentration of IL.17E.IL.25 cytokine.

**IL.27** Observed concentration of IL.27 cytokine.

**IL.31** Observed concentration of IL.31 cytokine.

**TNF.A** Observed concentration of TNF.A cytokine.

**TNF.B** Observed concentration of TNF.B cytokine.

**IL.28A** Observed concentration of IL.28A cytokine.

## Note

The ExampleData5 dataset is the same data as ExampleData1 but with a new column "Batch" added to indicate the batch number corresponding to each sample. The "Batch" column was randomly generated to simulate batch effects in the data.

## Source

Example data compiled for cytokine profiling.

## References

Pugh GH, Fouladvand S, SantaCruz-Calvo S, Agrawal M, Zhang XD, Chen J, Kern PA, Nikolajczyk BS. T cells dominate peripheral inflammation in a cross-sectional analysis of obesity-associated diabetes. *Obesity (Silver Spring)*. 2022;30(10): 1983–1994. doi:10.1002/oby.23528.

## Examples

```
data(ExampleData5)
```

# Index

∗ **datasets**
ExampleData1, 24
ExampleData2, 25
ExampleData3, 26
ExampleData4, 27
ExampleData5, 28

cyt_anova, 2
cyt_bp, 3
cyt_bp2, 4
cyt_dualflashplot, 5
cyt_errbp, 6
cyt_heatmap, 8
cyt_mint_splsda, 9
cyt_pca, 11
cyt_rf, 13
cyt_skku, 15
cyt_splsda, 16
cyt_ttest, 19
cyt_volc, 20
cyt_xgb, 21

ExampleData1, 24
ExampleData2, 25
ExampleData3, 26
ExampleData4, 27
ExampleData5, 28