Package 'xmsPANDA'

July 22, 2016

Type Package

Title Predictive And Network Discovery Analysis (PANDA): R package for biomarker discovery and biomarker-driven network analysis for studying disease mechanisms using metabolomics

Version 1.0.6

Date 2016-07-22

Author Karan Uppal

Maintainer Karan Uppal <kuppal2@emory.edu>

Description This package includes functions to perform biomarker discovery for classification and regression scenarios and to perform correlation based network analysis (all metabolites and/or targeted metabolites eg: glucose metabolism pathway).

License GPL2.0

LazyLoad yes

Depends limma, statmod, rgl, gplots, snow, mixOmics, doSNOW, foreach, e1071, WGCNA, randomForest, party,qvalue,fdrtool, GeneNet, corpcor, earth, pROC, multcomp, RColorBrewer, nlme, pls, plsgenomics, igraph, ROCR, flashClust, data.table, dplyr,plyr

R topics documented:

`xmsPANDA-package` *xmsPANDA*

Description

R package for metabolomic biomarker discovery and network analysis.

Details

Package:	xmsPANDA
Type:	Package
Version:	1.0.5.1
Date:	2016-06-27
License:	gpl2.0
LazyLoad:	yes

Author(s)

Karan Uppal Maintainer: <kuppal2@emory.edu>

`data_preprocess` *data_preprocess*

Description

This function performs data transformation, normalization

Usage

```
data_preprocess(Xmat=NA,Ymat=NA,feature_table_file, parentoutput_dir, class_labe
num_replicates = 3,
feat.filt.thresh = NA, summarize.replicates = TRUE, summary.method = "mean",
all.missing.thresh=0.5,
group.missing.thresh = 0.7, log2transform = TRUE,
medcenter = TRUE, znormtransform = FALSE, quantile_norm = TRUE,
lowess_norm = FALSE,madscaling = FALSE, missing.val = 0, samplermindex = NA,
rep.max.missing.thresh = 0.5,summary.na.replacement = "zeros",featselmethod=NA)
```

Arguments

`Xmat`	R object for feature table. If this is given, then feature table can be set to NA.
`Ymat`	R object for response/class labels matrix. If this is given, then class can be set to NA.

`feature_table_file`
: Feature table that includes the mz, retention time, and measured intensity in each sample for each analyte. The first 2 columns should be the mz and time. The remaining columns should correspond to the samples in the class labels file with each column including the intensity profile of a sample. Full path required. Eg: C:/My Documents/test.txt The feature table should be in a tab-delimited format. An example of the input file is provided under the "example" folder.

`parentoutput_dir`
: Provide full path of the folder where you want the results to be written. Eg: C:/My Documents/ProjectA/results/

`class_labels_file`
: File with class labels information for each sample. Samples should be in the same order as in the feature table. Please use the same format as in the example folder.

`num_replicates`
: Number of technical replicates

`feat.filt.thresh`
: Percent Intensity Difference or Coefficient of variation threshold; feature filtering Use NA to skip this step.

`summarize.replicates`
: Do the technical replicates per sample need to be averaged or median summarized?

`summary.method`
: Method for summarizing the replicates. Options: "mean" or "median"

`summary.na.replacement`
: How should the missing values be represented? Options: "zeros", "halffeaturemin", "halfsamplemin","halfdatamin", "none" "zeros": replaces missing values by 0 "halfsamplemin": replaces missing value by one-half of the lowest signal intensity in the corresponding sample "halfdatamin": replaces missing value by one-half of the lowest signal intensity in the complete dataset "halffeaturemin": replaces missing value by one-half of the lowest signal intensity for the current feature "none": keeps missing values as NAs

  Users are recommended to perform imputation prior to performing biomarker discovery.

`missing.val` How are the missing values represented in the input data? Options: "0" or "NA"

`samplermindex`
: Column index of any additional or irrelevant columns to be deleted. Options: "NA" or list of column numbers. eg: c(1,3,4) Default=NA

`rep.max.missing.thresh`
: What propotion of replicates are allowed to have missing values during the averaging or median summarization step of each biological sample? If the number of replicates with missing values is greater than the defined threshold, then the summarized value is represented by the "missing.val" parameter. If the number of replicates with missing values is less than or equal to the defined threshold, then the summarized value is equal to the mean or the median of the non-missing values. Default: 0.5

`all.missing.thresh`
: What propotion of total number of samples should have an intensity? Default: 0.5

`group.missing.thresh`
: What propotion of samples in either of the two groups should have an intensity? If at least x for further analysis. Default: 0.7

`log2transform`
: Data transformation: Please refer to http://www.biomedcentral.com/1471-2164/7/142 Try different combinations; such as log2transform=TRUE, znormtransfrom=FALSE or log2transform=FALSE, znormtransfrom=TRUE

`medcenter`
: Median centering of metabolites

`znormtransform`
: Auto scaling; each metabolite will have a mean of 0 and unit variance

`quantile_norm`
: Performs quantile normalization. Normalization options: Please set only one of the options to be TRUE

`lowess_norm`
: Performs lowess normalization. Normalization options: Please set only one of the options to be TRUE

`madscaling`
: Performs median adjusted scale normalization. Normalization options: Please set only one of the options to be TRUE

Value

Pre-processed data matrix.

Author(s)

Karan Uppal <kuppal2@emory.edu>

`diffexp` *diffexp*

Description

This function performs biomarker discovery and generates a correlation network based on the metabolome-wide (and targeted) correlation analysis of the differentially expressed features. The "featselmethod" allows users to select the method for selecting discriminatory features. The function evaluates the k-fold cross-validation accuracy using Support Vector Machine, performs hierarchical clustering analysis, PCA analysis (R2/Q2 diagnostics), and generates boxplots for discriminatory features identified at each coefficient of variation across all samples (RSD) filtering threshold level. An optimization score that minimizes the number of false positives and increases the classification accuracy is used to select the best set of features. The best set is then used for correlation (complete or partial) based metabolome-wide network analysis. Additionally, users have the option to provide a list of mzs corresponding to chemicals of interest such as (phenylalanine, choline, etc). The function using the getVenn function in xMSanalyzer to find the mzs matching the target list based on a user defined mass search threshold (+/- ppm).

Usage

```
diffexp(Xmat = NA, Ymat = NA, feature_table_file, parentoutput_dir,
class_labels_file, num_replicates = 3, feat.filt.thresh = NA,
summarize.replicates = TRUE, summary.method = "mean",
summary.na.replacement = "zeros", missing.val = 0,
rep.max.missing.thresh = 0.5, all.missing.thresh = 0.1,
group.missing.thresh = 0.7, input.intensity.scale = "raw",
log2transform = TRUE, medcenter = FALSE,
```

```
znormtransform = FALSE, quantile_norm = TRUE,
lowess_norm = FALSE, madscaling = FALSE,
rsd.filt.list = seq(0, 75, 5), pairedanalysis = FALSE,
featselmethod = "limma", fdrthresh = 0.05, fdrmethod = "BH",
cor.method = "spearman", networktype = "complete",
abs.cor.thresh = 0.4, cor.fdrthresh = 0.05, kfold = 10,
pred.eval.method = "AUC", feat_weight = 1, globalcor = TRUE,
target.metab.file = NA, target.mzmatch.diff = 10,
target.rtmatch.diff = NA, max.cor.num = 100, samplermindex = NA,
pcacenter = FALSE, pcascale = FALSE, numtrees = 20000,
analysismode = "classification", net_node_colors = c("green", "red"),
net_legend = TRUE, svm_kernel = "radial", heatmap.col.opt = "RdBu",
sample.col.opt = "rainbow", alphacol = 0.3, rf_selmethod = "rawVIMsig",
pls_vip_thresh = 1, num_nodes = 2, max_varsel = 100, pls_ncomp = 5,
pca.stage2.eval = TRUE, scoreplot_legend = TRUE, pca.global.eval = TRUE,
rocfeatlist = seq(2, 11, 1), rocfeatincrement = TRUE, rocclassifier = "svm",
foldchangethresh = 2, wgcnarsdthresh = 20, WGCNAmodules = TRUE,
optselect = TRUE, max_comp_sel = 1, saveRda = TRUE,
legendlocation = "topleft", degree_rank_method = "diffK",
pca.cex.val = 4, pca.ellipse = FALSE, ellipse.conf.level = 0.95,
pls.permut.count = 100, svm.acc.tolerance = 5)
```

Arguments

`Xmat`	R object for feature table. If this is given, then feature table can be set to NA.
`Ymat`	R object for response/class labels matrix. If this is given, then class can be set to NA.
`feature_table_file`	Feature table that includes the mz, retention time, and measured intensity in each sample for each analyte. The first 2 columns should be the mz and time. The remaining columns should correspond to the samples in the class labels file with each column including the intensity profile of a sample. Full path required. Eg: C:/My Documents/test.txt The feature table should be in a tab-delimited format. An example of the input file is provided under the "example" folder.
`parentoutput_dir`	Provide full path of the folder where you want the results to be written. Eg: C:/My Documents/ProjectA/results/
`class_labels_file`	File with class labels information for each sample. Samples should be in the same order as in the feature table. Please use the same format as in the example folder. If you want to adjust for covariates in "lmreg" option, then you can add additional columns, one per covariate. Categorical variables should be strings (eg: "male", "female"). Please see "classlabels_gender.txt" file as an example.
`num_replicates`	Number of technical replicates
`feat.filt.thresh`	Percent Intensity Difference or Coefficient of variation threshold; feature filtering Use NA to skip this step.
`summarize.replicates`	Do the technical replicates per sample need to be averaged or median summarized?

summary.method
: Method for summarizing the replicates. Options: "mean" or "median"

summary.na.replacement
: How should the missing values be represented? Options: "zeros", "halffeaturemin", "halfsamplemin","halfdatamin", "none" "zeros": replaces missing values by 0 "halfsamplemin": replaces missing value by one-half of the lowest signal intensity in the corresponding sample "halfdatamin": replaces missing value by one-half of the lowest signal intensity in the complete dataset "halffeaturemin": replaces missing value by one-half of the lowest signal intensity for the current feature "none": keeps missing values as NAs

  Users are recommended to perform imputation prior to performing biomarker discovery.

missing.val
: How are the missing values represented in the input data? Options: "0" or "NA"

rep.max.missing.thresh
: What propotion of replicates are allowed to have missing values during the averaging or median summarization step of each biological sample? If the number of replicates with missing values is greater than the defined threshold, then the summarized value is represented by the "missing.val" parameter. If the number of replicates with missing values is less than or equal to the defined threshold, then the summarized value is equal to the mean or the median of the non-missing values. Default: 0.5

all.missing.thresh
: What propotion of total number of samples should have an intensity? Default: 0.5

input.intensity.scale
: Are the intensities in the input feature table at raw scale or log2 scale? eg: "raw" or "log2" Default: "raw"

group.missing.thresh
: What propotion of samples in either of the two groups should have an intensity? If at least x for further analysis. Default: 0.7

log2transform
: Data transformation: Please refer to http://www.biomedcentral.com/1471-2164/7/142 Try different combinations; such as log2transform=TRUE, znormtransfrom=FALSE or log2transform=FALSE, znormtransfrom=TRUE

medcenter
: Median centering of metabolites

znormtransform
: Auto scaling; each metabolite will have a mean of 0 and unit variance

quantile_norm
: Performs quantile normalization. Normalization options: Please set only one of the options to be TRUE

lowess_norm
: Performs lowess normalization. Normalization options: Please set only one of the options to be TRUE

madscaling
: Performs median adjusted scale normalization. Normalization options: Please set only one of the options to be TRUE

rsd.filt.list
: This parameter allows to perform feature filtering based on overall variance (across all samples) prior to performing hypothesis testing. Eg: seq(0,30,5).

pairedanalysis
: Is this a paired-study design? TRUE or FALSE If samples are paired, then the feature table and the class labels file should be organized so that the paired

samples are arranged in the same order in each group. For example, the first sample in group A and the first sample in group B should be paired.

`featselmethod`
: Options: "limma": for one-way ANOVA using LIMMA (mode=classification) "limma2way": for two-way ANOVA using LIMMA (mode=classification) "limma1wayrepeat": for one-way ANOVA repeated measures using LIMMA (mode=classification) "limma2wayrepeat": for two-way ANOVA repeated measures using LIMMA (mode=classification) "lm1wayanova": for one-way ANOVA using linear model (mode=classification) "lm2wayanova": for two-way ANOVA using linear model (mode=classification) "lm1wayanovarepeat": for one-way ANOVA repeated measures using linear model (mode=classification) "lm2wayanovarepeat": for two-way ANOVA repeated measures using linear model (mode=classification) "lmreg": variable selection based on p-values calculated using a linear regression model; allows adjustment for covariates (mode= regression or classification) "logitreg": variable selection based on p-values calculated using a logistic regression model; allows adjustment for covariates (mode= classification) "rfesvm": uses recursive feature elimination SVM algorithm for variable selection; (mode=classification) "RF": for random forest based feature selection (mode= regression or classification) "MARS": for multiple adaptive regression splines (MARS) based feature selection (mode= regression or classification) "pls": for partial least squares (PLS) based feature selection (mode= regression or classification) "spls": for sparse partial least squares (PLS) based feature selection (mode= regression or classification) "o1pls": for orthogonal partial least squares (OPLS) based feature selection (mode= regression or classification)

`fdrthresh`
: False discovery rate threshold. Eg: 0.05

`fdrmethod`
: Options: "BH", "ST", "Strimmer", "none" "BH": Benjamini-Hochberg (1995) (Default: more conservative than "ST" and "Strimmer") "ST": Storey & Tibshirani (Storey 2001, PNAS) algorithm implemented in the qvalue package "Strimmer": (Strimmer 2008, Bioinformatics) algorithm implemented in the fdrtool package "none": No FDR correction will be performed. fdrthresh will be treated as raw p-value cutoff

`cor.method`
: Correlation method. Options: "pearson" or "spearman". Default: "spearman"

`networktype`
: Options: "complete" or "GGM" "complete": performs network analysis using ordinary Pearson or Spearman correlation statistic "GGM": generates network based on partial correlation analysis using the GeneNet package

`abs.cor.thresh`
: Absolute Pearson correlation coefficient for network analysis. Default: 0.4

`cor.fdrthresh`
: False discovery rate threshold for correlation analysis. Default: 0.05

`kfold`
: Number of subsets in which the data should be divided for cross-validation. If kfold=10, then the data set will be divided into 10 subsets of size (N/10), where N is the total number of samples. 9 subsets are used for training and the remaining 1 is used for testing. This process is repeated 10 times and the CV-accuracy would be the mean of the classification accuracy from the 10 iterations. The same will be true for any other value of k. WARNING: The kfold value should be less than or equal to the total number of samples.

`pred.eval.method`
: Criteria for evaluating the performance of the model. Cross-validation or area under the curve (AUC). Eg: "CV" or "AUC". Default: "AUC"

`feat_weight`
: How much importance should be given to number of features selected in the scoring function? 0 or 1. Value of 1 gives prevents over-fitting. Default: 1

`globalcor` Do you want to perform correlation analysis after biomarker discovery? Options: "TRUE" or "FALSE"

`target.metab.file` File that includes the mz and/or retention time of the targeted metabolites. See example.

`target.mzmatch.diff` +/- ppm mass tolerance for searching the target m/z in the current feature table

`target.rtmatch.diff` +/- retention time tolerance for searching the target m/z in the current feature table

`max.cor.num` Maximum number of correlated metabolites to be included in the network figure. Default: 100

`pcacenter` Data centering for PCA. Options: "TRUE" or "FALSE". Default=TRUE

`pcascale` Data scaling for PCA. Options: "TRUE" or "FALSE". Default=TRUE

`samplermindex` Column index of any additional or irrelevant columns to be deleted. Options: "NA" or list of column numbers. eg: c(1,3,4) Default=NA

`numtrees` Number of trees to be used for random forest method. Default=500

`analysismode` "classification" for group-wise comparison (case vs control) or "regression" for continuous response variables. Default: "classification"

`net_node_colors` Colors of nodes in the correlation networks. Eg: c("pink", "skyblue"), or ("red","green")

`net_legend` Should the network be displayed for the correlation network? eg: TRUE or FALSE

`max_var` Max number of variables to be used for sPLS and Random Forest? eg:150

`svm_kernel` SVM kernel eg: "radial" or "linear"

`heatmap.col.opt` Color scheme for HCA hetmap eg: "RdBu", "topo", "heat", or "terrain"

`sample.col.opt` Color scheme for PCA and heatmap sample axis eg: "rainbow", "heat" or "topo"

`alphacol=0.3` Color scaling parameter eg:0.3

`rf_selmethod` Random forest VIP based selection method. If rankbased option is selected, variables are ranked based on the Variable Importance Measure. Only the top "max_rf_varsel" variables are selected. eg: "rawVIPsig" or "absVIPthresh" or "rankbased"

`pls_vip_thresh` Threshold for VIP score from PLS/O1PLS. eg: 1

`max_varsel` Maximum number of variables to keep if "rankbased" RF or spls is used. eg: 100

`pls_ncomp` Maximum number of components to be considered during the PLS optimal number of components selection step. eg: 2

`pca.stage2.eval` Should PCA diagnostics be performed in stage 2? eg: TRUE or FALSE

`scoreplot_legend` Should legends be included in score plots? eg: TRUE or FALSE

`pca.global.eval` Should global PCA evaluation be performed? Default:TRUE eg: TRUE or FALSE

`rocfeatlist` Vector indicating number of features to be used for ROC evaluation: eg: c(2,4,6) will generate ROC for top 2, top 4, and top 6 feautres. Default: seq(2,10,1)

`rocclassifier` Classifier to be used for ROC evaluation. Options: "svm" or "logitreg". Default: "svm"

`foldchangethresh` Secondary feature selection criteria based on fold change threshold. This is performed after statistical significance or importance evaluation.

`wgcnarsdthresh` Relative standard deviation or coefficient of variation (across all samples) based filtering threshold before performing WGCNA module preservation analysis. Default: 20

`WGCNAmodules` Perform WGCNA module preservation analysis. TRUE or FALSE Default: TRUE

`optselect` Determine optimal number of PLS components. Default: TRUE

`max_comp_sel` Number of PLS components to use for VIP or sparse loading selection (sPLS). Default=1

`saveRda` Should the results be saved in a binary R object. Default: TRUE

`legendlocation` Legend location for PLS or PCA plots

`degree_rank_method` Variable ranking method based on degree centrality. eg: "diffK" or "overall" "diffK" ranks variables based on difference in connectivity between different classes or phenotypes. "overall" ranks variables based on overall network connectivity using samples from all classes. eg: 4

`pca.cex.val` Size of points on PCA plots. eg: 4

`pca.ellipse` Should ellipse be plotted on PCA plots? eg: TRUE or FALSE

`ellipse.conf.level` Confidence interval for PCA ellipses eg: 0.95

`pls.permut.count` Number of permutations for calculating p-values for PLS or sPLS models. eg: 100

`svm.acc.tolerance` Stopping criteria for forward feature selection using "rfeSVM" method. If the difference between best accuracy and current accuracy based on the newly added feature drops below the tolerance level, the forward selection process is terminated. eg: 5

Details

This function performs data transformation, normalization, FDR analysis using LIMMA, variable selection using random forests, evaluates the predictive accuracy of the FDR significant features using k-fold cross-validation with a Support Vector Machine classifier, performs two-way hierarchical clustering analysis, and principal component analysis. An optimizaiton scheme is used to select the best set of features from different log2 fold change filtering thresholds. Finally, metabolome-wide and targeted correlation based network analysis of the FDR significant features is performed.

Value

`diffexp_metabs`
: Best set of discriminatory metabolites.

`diffexp_metabs_prescaling`
: Best set of discriminatory metabolites before auto-scaling and/or median centering.

`mw.an.fdr`
: Metabolome-wide significant correlation network of differentially expressed metabolites.

`targeted.an.fdr`
: Correlation network of differentially expressed metabolites with targeted metabolites.

Following files are generated in the parent output location: Manhattan plots: showing metabolome wide p-values; Heatmap from Two-way hierarchical clustering analysis; Pairwise score plots from Principal Component Analysis; PCA score distribution plots; ROC plots; List of differentially expressed metabolites; Boxplots of differentially expressed metabolites; Correlation network figure and matrix; Pairwise correlation matrix CIRCOS format ready to be uploaded to: http://mkweb.bcgsc.ca/tableviewer/visu
Or uploaded to Cytoscape gml format

Author(s)

Karan Uppal <kuppal2@emory.edu>

`do_wgcna` *do_wgcna*

Description

This function performs module preservation analysis using WGCNA.

Usage

```
do_wgcna(feature_table_file = NA, class_labels_file = NA, X = NA,
Y = NA, sigfeats = NA)
```

Arguments

`feature_table_file`
: Path and name of feature table that includes the mz, retention time, and measured intensity in each sample for each analyte. The first 2 columns should be the mz and time. The remaining columns should correspond to the samples in the class labels file with each column including the intensity profile of a sample. Full path required. Eg: C:/My Documents/test.txt The feature table should be in a tab-delimited format. An example of the input file is provided under the "example" folder.

`class_labels_file`
: File with class labels information for each sample. Samples should be in the same order as in the feature table. Please use the same format as in the example folder. If you want to adjust for covariates in "lmreg" option, then you can add additional columns, one per covariate. Categorical variables should be strings (eg: "male", "female"). Please see "classlabels_gender.txt" file as an example.

X	R object for feature table. If this is given, then feature table can be set to NA.
Y	R object for response/class labels matrix. If this is given, then class labels file can be set to NA.
sigfeats	List of differentially expressed features. Default: NA

Details

This function calls WGCNA to perform module preservation analysis between different classes or groups.

Value

PDF plots for module preservation from WGCNA and preservation matrix

Author(s)

Karan Uppal

References

WGCNA (Horvath 2007)

`get_boxplots` *get_boxplots*

Description

This function generates boxplots for m/z features. The input intensity matrix could be transformed or non-transformed intensities. Sample labels in the class labels file should be in the same order as the intensity matrix or feature table.

Usage

```
get_boxplots(feature_table_file, parentoutput_dir, class_labels_file, sample.col
```

Arguments

`feature_table_file`
: Feature table that includes the mz, retention time, and measured intensity in each sample for each analyte. The first 2 columns should be the mz and time. The remaining columns should correspond to the samples in the class labels file with each column including the intensity profile of a sample. Full path required. Eg: C:/My Documents/test.txt The feature table should be in a tab-delimited format. An example of the input file is provided under the "example" folder.

`parentoutput_dir`
: Provide full path of the folder where you want the results to be written. Eg: C:/My Documents/ProjectA/results/

`class_labels_file`
: File with class labels information for each sample. Samples should be in the same order as in the feature table. Please use the same format as in the example folder.

`sample.col.opt`
: Color scheme for PCA and heatmap sample axis eg: "heat" or "topo"

`alphacol=0.3` Color scaling parameter eg:0.3

Value

Creates a PDF with boxplots for each m/z feature.

Author(s)

Karan Uppal <kuppal2@emory.edu>

`get_hca` *get_hca*

Description

This function performs two-way hierarchical clustering analysis and generates a heatmap showing the clustering results. The input intensity matrix could be transformed or non-transformed intensities. Sample labels in the class labels file should be in the same order as the intensity matrix or feature table.

Usage

```
get_hca(feature_table_file, parentoutput_dir, class_labels_file, heatmap.col.opt
is.data.znorm = FALSE, analysismode = "classification", sample.col.opt = "rainbo
plots.height = 2000, plots.res = 300, alphacol = 0.3,hca_type="two-way")
```

Arguments

`feature_table_file`
: Feature table that includes the mz, retention time, and measured intensity in each sample for each analyte. The first 2 columns should be the mz and time. The remaining columns should correspond to the samples in the class labels file with each column including the intensity profile of a sample. Full path required. Eg: C:/My Documents/test.txt The feature table should be in a tab-delimited format. An example of the input file is provided under the "example" folder.

`parentoutput_dir`
: Provide full path of the folder where you want the results to be written. Eg: C:/My Documents/ProjectA/results/

`class_labels_file`
: File with class labels information for each sample. Samples should be in the same order as in the feature table. Please use the same format as in the example folder.

`heatmap.col.opt`
: Color scheme for HCA hetmap eg: "RdBu", "topo", "heat", or "terrain"

`cor.method` Correlation method. Options: "person" or "spearman". Default: "spearman"

`is.data.znorm`
: Is the data already auto-scaled or z-scaled? eg: TRUE or FALSE

`analysismode` "classification" for group-wise comparison (case vs control) or "regression" for continuous response variables. Default: "classification"

sample.col.opt	Color scheme for PCA and heatmap sample axis eg: "heat" or "topo" or "rainbow"
plots.width	Width of the tiff file. eg: 2000
plots.height	Height of the tiff file. eg: 2000
plots.res	Resolution of the tiff file. eg: 300
alphacol	Color scaling parameter eg:0.3
hcatype	Color scaling parameter eg:"two-way" or "one-way"

Value

Heatmap from Two-way hierarchical clustering analysis; Intensity matrix in the same order as the dendrograms in heatmap; Sample cluster labels

Author(s)

Karan Uppal <kuppal2@emory.edu>

---

get_manhattanplots *get_manhattanplots*

Description

Function to generate Manhattan plots.

Usage

```
get_manhattanplots(xvec, yvec, up_or_down, maintext = "", ythresh = 0.05,
ylab, xlab, colorvec = c("darkgreen", "firebrick1"),
col_seq = c("brown", "chocolate3", "orange3", "coral", "pink", "skyblue",
"blue", "darkblue", "purple", "violet"), xincrement = 150, yincrement = 1)
```

Arguments

xvec	Vector with values for the x-axis. eg: m/z or retention time values
yvec	Vector with values for the y-axis. eg: (-)Log10 of p-values, VIP, loadings, regression coefficients, etc.
up_or_down	Vector indicating directionality of change. eg: Fold change values
maintext	Text for the plot title
ythresh	Y-axis threshold for significance or differential expression. eg: 3 for p=0.001; y=(-1)*log10(0.001) or 2 for VIP from PLS
ylab	Y-axis label
xlab	X-axis label
colorvec	Vector of colors for representing up-regulation and down-regulation. eg: c("darkgreen", "firebrick1") In this case, features that are up-regulated in class A will have "darkgreen" color, and features that are up-regulated in class B will have "firebrick1" color.
col_seq	Vector of colors for plotting different segments of the x-axis

xincrement	Window size for breaking the x-axis into different segments for visualization purposes. eg: 150
yincrement	Window size for breaking the y-axis into different segments for visualization purposes. eg: 1

Details

This function can be used to generate Type 1 Manhattan plots: signficance vs m/z Type 2 Manhattan plots: significance vs retention time Type 3 Manhattan plots: significance vs intensity

Value

Manhattan plots

Note

#Example pdf("Manhattanplot.pdf") get_manhattanplots(...) #pass arguments dev.off()

Author(s)

Karan Uppal

---

get_pca *Perfors PCA analysis*

Description

This function uses the pca function implemented in the mixOmics package for PCA analysis

Usage

```
get_pca(X, samplelabels, legendlocation = "topright", filename = NA,
ncomp = 5, center = TRUE, scale = TRUE, legendcex = 0.5,
outloc = getwd(), col_vec = NA, sample.col.opt = "default",
alphacol = 0.3, class_levels = NA, pca.cex.val = 3,
pca.ellipse = TRUE, ellipse.conf.level = 0.5, samplenames = FALSE)
```

Arguments

X	Data matrix without m/z and time.
samplelabels	Vector with class label for each sample.
legendlocation	Location of the legend on PCA score plots
filename	eg: "all", "signficantfeats"
ncomp	Number of components; please use ?pca for more information
center	Should the data be centered?; please use ?pca for more information
scale	Should the data be scaled?; please use ?pca for more information
legendcex	Size of the legend text in the PCA score plots. e.g.: 0.5 or 0.7
outloc	Output folder location

`col_vec`	Provide vector of colors for each group. eg: NA or c("red","green") for cases and controls, respectively. This argument is ignored if sample.col.opt is provided
`sample.col.opt`	Select R color palette. eg: "rainbow", "terrain", "topo". "heat", "default"
`alphacol`	Semi-transparent colors eg: 0.2
`class_levels`	Vector with names of different sample groups. eg: c("case", "control") or NA
`pca.cex.val`	Size of dots in PCA score plots. eg: 0.4
`pca.ellipse`	Should the score confidence interval for each group be drawn? eg: TRUE or FALSE
`ellipse.conf.level`	Confidence interval level eg: 0.95
`samplenames`	Should the sample names be included in PCA plots? eg: TRUE or FALSE

Details

This function performs PCA analysis. The results are saved in a RDA file.

Value

The function returns PCA results as an object and generates pairwise score plots for the first three components

Author(s)

Karan Uppal

References

mixOmics

`get_pcascoredistplots` *get_pcascoredistplots*

Description

PCA score distribution (25th percentile, median, 75th percentile) plots

Usage

```
get_pcascoredistplots(X = NA, Y = NA, feature_table_file, parentoutput_dir,
class_labels_file, sample.col.opt = "rainbow", plots.width = 2000,
plots.height = 2000, plots.res = 300, alphacol = 0.3, col_vec,
pairedanalysis = FALSE, pca.cex.val = 3, legendlocation = "topright",
pca.ellipse = TRUE, ellipse.conf.level = 0.5, filename = "all")
```

Arguments

X	R object for feature table. If this is given, then feature table can be set to NA.
Y	R object for response/class labels matrix. If this is given, then class labels file can be set to NA.
feature_table_file	Path and name of feature table that includes the mz, retention time, and measured intensity in each sample for each analyte. The first 2 columns should be the mz and time. The remaining columns should correspond to the samples in the class labels file with each column including the intensity profile of a sample. Full path required. Eg: C:/My Documents/test.txt The feature table should be in a tab-delimited format. An example of the input file is provided under the "example" folder.
parentoutput_dir	Provide full path of the folder where you want the results to be written. Eg: C:/My Documents/ProjectA/results/
class_labels_file	File with class labels information for each sample. Samples should be in the same order as in the feature table. Please use the same format as in the example folder. If you want to adjust for covariates in "lmreg" option, then you can add additional columns, one per covariate. Categorical variables should be strings (eg: "male", "female"). Please see "classlabels_gender.txt" file as an example.
sample.col.opt	Color scheme for PCA and heatmap sample axis eg: "rainbow", "heat" or "topo"
alphacol	Color scaling parameter eg:0.3
col_vec	Vector of colors for each sample.
pairedanalysis	Is this a paired-study design? TRUE or FALSE If samples are paired, then the feature table and the class labels file should be organized so that the paired samples are arranged in the same order in each group. For example, the first sample in group A and the first sample in group B should be paired.
pca.cex.val	Size of points on PCA plots. eg: 4
legendlocation	Legend location on PCA plots
pca.ellipse	Should ellipse be plotted on PCA plots? eg: TRUE or FALSE
ellipse.conf.level	Confidence interval for PCA ellipses eg: 0.95
filename	Name of output PDF file

Details

This function performs PCA and generates pariwise score plots as well as score distribution plots (per group). It uses the Y vector and classlabels for color coding the samples in the pairwise score plots.

Value

The output includes: Pairwise PCA score plots, PCA score distribution plots, PCA scores and loadings text files.

Note

The plots can be sent to an external device by running the following commands: pdf("get_pcascoredistplots.pdf") get_pcascoredistplots(...) dev.off()

Author(s)

Karan Uppal

`get_roc` *get_roc*

Description

This function generates Receiver Operating Characteristic curves using SVM and Logistic Regression as classifiers.

Usage

```
get_roc(dataA, classlabels, classifier = "svm", kname = "radial",
rocfeatlist = seq(2, 10, 1), rocfeatincrement = TRUE,
testset = NA, testclasslabels = NA, mainlabel = NA)
```

Arguments

`dataA`	R object for feature table with only differentially expressed features. This is the training set.
`classlabels`	Class labels vector.
`classifier`	Classification algorithm to be used for ROC analysis. svm: Support Vector Machine logitreg: Logistic Regression eg: "svm" or "logitreg"
`kname`	Kernel for SVM. eg: "radial"
`rocfeatlist`	Vector indicating number of features to be used for ROC evaluation: eg: c(2,4,6) will generate ROC for top 2, top 4, and top 6 feautres. Default: seq(2,10,1)
`rocfeatincrement`	Turns on or off forward selection. eg: TRUE or FALSE
`testset`	R object for test feature table with only differentially expressed features. This is the test set.
`testclasslabels`	Class labels vector for samples in the test set.
`mainlabel`	Main text label for the ROC plot. eg: "Group A vs B ROC curve"

Details

Function to perform ROC curve analysis using only traning set or using both training and test set.

Value

PDF file with ROC plot

metabnet *metabnet*

Description

Function for correlation (complete or partial) based metabolome-wide network analysis. Additionally, users have the option to provide a matrix of m/z features corresponding to chemicals of interest such as (phenylalanine, choline, etc) and/or a matrix of m/z features corresponding to discriminatory metabolites.

Usage

```
metabnet(feature_table_file,target.metab.file,sig.metab.file,class_labels_file=N
num_replicates=3,cor.method="spearman",abs.cor.thresh=0.4,
cor.fdrthresh=0.05,target.mzmatch.diff=10,target.rtmatch.diff=NA,
max.cor.num=100,feat.filt.thresh=NA,summarize.replicates=TRUE,
summary.method="mean",all.missing.thresh=0.5,
group.missing.thresh=0.7,
log2transform=TRUE,medcenter=TRUE,znormtransform=FALSE,
quantile_norm=TRUE,lowess_norm=FALSE,madscaling=FALSE,
missing.val=0, networktype="complete", samplermindex=NA,
rep.max.missing.thresh=0.3,summary.na.replacement="zeros",
net_node_colors = c("pink", "skyblue"),
net_legend = FALSE)
```

Arguments

`feature_table_file`
: Feature table that includes the mz, retention time, and measured intensity in each sample for each analyte. The first 2 columns should be the mz and time. The remaining columns should correspond to the samples in the class labels file with each column including the intensity profile of a sample. Full path required. Eg: C:/My Documents/test.txt The feature table should be in a tab-delimited format. An example of the input file is provided under the "example" folder.

`target.metab.file`
: File that includes the mz and/or retention time of the targeted metabolites corresponding to pathways or chemicals of interest. See example.

`sig.metab.file`
: File that includes the mz and/or retention time of the discriminatory metabolites. See example.

`class_labels_file`
: File with class labels information for each sample. Samples should be in the same order as in the feature table. Please use the same format as in the example folder.

`parentoutput_dir`
: Provide full path of the folder where you want the results to be written. Eg: C:/My Documents/ProjectA/results/

`num_replicates`
: Number of technical replicates

`cor.method` Correlation method. Options: "pearson" or "spearman". Default: "spearman"

`abs.cor.thresh`
: Absolute Pearson correlation coefficient for network analysis. Eg: 0.5

`cor.fdrthresh`
: False discovery rate threshold for correlation analysis. Eg: 0.05

`target.mzmatch.diff`
: +/- ppm mass tolerance for searching the target m/z in the current feature table

`target.rtmatch.diff`
: +/- retention time tolerance for searching the target m/z in the current feature table

`max.cor.num`
: Maximum number of correlated metabolites to be included in the network figure. Default: 100

`feat.filt.thresh`
: Percent Intensity Difference or Coefficient of variation threshold; feature filtering Use NA to skip this step.

`summarize.replicates`
: Do the technical replicates per sample need to be averaged or median summarized?

`summary.method`
: Method for summarizing the replicates. Options: "mean" or "median"

`summary.na.replacement`
: How should the missing values be represented? Options: "zeros", "halfsamplemin","halfdatamin", "none" "zeros": replaces missing values by 0 "halfsamplemin": replaces missing value by one-half of the lowest signal intensity in the corresponding sample "halfdatamin": replaces missing value by one-half of the lowest signal intensity in the complete dataset "none": keeps missing values as NAs

  Users are recommended to perform imputation prior to performing biomarker discovery.

`all.missing.thresh`
: What propotion of total number of samples should have an intensity? Default: 0.5

`group.missing.thresh`
: What propotion of samples in either of the two groups should have an intensity? If at least x for further analysis. Default: 0.7

`log2transform`
: Data transformation: Please refer to http://www.biomedcentral.com/1471-2164/7/142 Try different combinations; such as log2transform=TRUE, znormtransfrom=FALSE or log2transform=FALSE, znormtransfrom=TRUE

`medcenter`
: Median centering of metabolites

`znormtransform`
: Auto scaling; each metabolite will have a mean of 0 and unit variance

`quantile_norm`
: Performs quantile normalization. Normalization options: Please set only one of the options to be TRUE

`lowess_norm`
: Performs lowess normalization. Normalization options: Please set only one of the options to be TRUE

`madscaling`
: Performs median adjusted scale normalization. Normalization options: Please set only one of the options to be TRUE

`missing.val`
: How are the missing values represented in the input data? Options: "0" or "NA"

`networktype` Options: "complete" or "GGM" "complete": performs network analysis using ordinary Pearson or Spearman correlation statistic "GGM": generates network based on partial correlation analysis using the GeneNet package

`samplermindex` Column index of any additional or irrelevant columns to be deleted. Options: "NA" or list of column numbers. eg: c(1,3,4) Default=NA

`rep.max.missing.thresh` What propotion of replicates are allowed to have missing values during the averaging or median summarization step of each biological sample? If the number of replicates with missing values is greater than the defined threshold, then the summarized value is represented by the "missing.val" parameter. If the number of replicates with missing values is less than or equal to the defined threshold, then the summarized value is equal to the mean or the median of the non-missing values. Default: 0.5

`net_node_colors` Colors of nodes in the correlation networks. Eg: c("pink", "skyblue"), or ("red","green")

`net_legend` Should the network be displayed for the correlation network? eg: TRUE or FALSE

Details

Function for metabolomic network analysis

Value

Correlation matrix and network of metabolites.

Author(s)

Karan Uppal <kuppal2@emory.edu>