Package 'MetaboAnalystR'

April 18, 2020

Title An R Package for Comprehensive Analysis of Metabolomics Data

Version 3.0.0

Author Jianguo Xia [aut, cre], Jasmine Chong [aut], Zhiqiang Pang [aut]

Maintainer Jasmine Chong <jasmine.chong@mail.mcgill.ca>

BugReports https://github.com/xia-lab/MetaboAnalystR/issues

Description This package contains the R functions and libraries underlying the popular MetaboAnalyst web server, including 500 functions for data processing, normalization, statistical analysis, metabolite set enrichment analysis, metabolic pathway analysis, and biomarker analysis. The package is synchronized with the web server. After installing and loading the package, users will be able to reproduce the same results from their local computers using the corresponding R command history downloaded from MetaboAnalyst, to achieve maximum flexibility and reproducibility.

Depends R (¿= 3.6.2), methods

License GPL-3

Encoding UTF-8

LazyData true

RoxygenNote 7.1.0

Imports data.table,
pls,
lattice,
Rserve,
ellipse,
scatterplot3d,
Cairo,
randomForest,
caTools,
e1071,
som,
impute,
pcaMethods,
RJSONIO,
ROCR,

globaltest,
GlobalAncova,
Rgraphviz,
preprocessCore,
genefilter,
pheatmap,
SSPA,
sva,
Rcpp,
pROC,
limma,
car,
fitdistrplus,
lars,
Hmisc,
magrittr,
xtable,
caret,
igraph,
gplots,
KEGGgraph,
reshape,
RColorBrewer,
tibble,
RSQLite,
spls,
siggenes,
ggplot2,
BiocParallel,
metap,
scales,
MSnbase,
progress,
entropy,
rsm

Suggests knitr,
rmarkdown,
devtools,
testthat,
plotly,
reshape2,

VignetteBuilder knitr

R topics documented:

`.compute.mummichog.fgsea` *Internal function for calculating GSEA, no RT*

Description

Internal function for calculating GSEA, no RT

Usage

```
.compute.mummichog.fgsea(mSetObj, permNum)
```

`.compute.mummichog.RT.fgsea` *Internal function for calculating GSEA, with RT*

Description

Internal function for calculating GSEA, with RT

Usage

```
.compute.mummichog.RT.fgsea(mSetObj, permNum)
```

.emptyscan.remove *Function for 'Empty scan' removal*

Description

Function for 'Empty scan' removal (internal use only)

Usage

```
.emptyscan.remove(raw_data, ms_list)
```

Author(s)

Zhiqiang Pang <zhiqiang.pang@mail.mcgill.ca>

.init.Permutations *Internal function to perform PSEA, no retention time*

Description

Internal function to perform PSEA, no retention time

Usage

```
.init.Permutations(mSetObj, permNum)
```

.init.RT.Permutations *Internal function to perform PSEA, with RT*

Description

Internal function to perform PSEA, with RT

Usage

```
.init.RT.Permutations(mSetObj, permNum)
```

`.read.metaboanalyst.lib` *Read RDS files from the internet*

Description

Function downloads the required file and reads it only if not already in working directory. Need to specify the file URL and the destfile.

Usage

```
.read.metaboanalyst.lib(filenm)
```

Arguments

`filenm`	Input the name of the file to download

`.readDataTable` *Read data table*

Description

Function to read in a data table. First, it will try to use fread, however, it has issues with some windows 10 files. In such case, use the slower read.table method.

Usage

```
.readDataTable(fileName)
```

Arguments

`fileName`	Input filename

Author(s)

Jeff Xia<`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`AddErrMsg` *Adds an error message*

Description

The error message will be printed in all cases. Used in higher functions.

Usage

```
AddErrMsg(msg)
```

Arguments

`msg`	Error message to print

`add_trace` *I. Internal Functions from Plotly Package THe functions are from Plotly Package and was called internally only*

Description

I. Internal Functions from Plotly Package THe functions are from Plotly Package and was called internally only

Usage

```
add_trace(p, ..., data = NULL, inherit = TRUE)
```

References

https://cran.r-project.org/package=plotly

Sievert C (2020). Interactive Web-Based Data Visualization with R, plotly, and shiny. Chapman and Hall/CRC. ISBN 9781138331457, https://plotly-r.com.

`analyze.lipids` *Lipid analysis pipeliner*

Description

Lipid analysis pipeliner

Usage

```
analyze.lipids(inFile, iso = "y")
```

Arguments

`inFile`	Input the file to read in
`iso`	Default is set to "y"

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`ANOVA.Anal` *Perform ANOVA analysis*

Description

ANOVA analysis

Usage

```
ANOVA.Anal(mSetObj=NA, nonpar=F, thresh=0.05, post.hoc="fisher")
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)
`nonpar`	Logical, use a non-parametric test (T) or not (F)
`thresh`	Numeric, from 0 to 1, indicate the p-value threshold
`post.hoc`	Input the name of the post-hoc test, "fisher" or "tukey"
`all_results`	Logical, if TRUE, it will output the ANOVA results for all compounds with no post-hoc tests performed.

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

ANOVA2.Anal *Perform Two-way ANOVA*

Description

Perform Two-way ANOVA

Usage

```
ANOVA2.Anal(mSetObj=NA, thresh=0.05, p.cor="fdr", type="time0", aov.type=1, use.interact=1)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)
thresh	Input the p-value threshold
p.cor	Select method for p-value correction, bonferroni, holm or fdr
type	Select b to perform between-subjects ANOVA, and w for within-subjects ANOVA
aov.type	Specify 1 for ANOVA type 1, or 3 for ANOVA type 3
use.interact	Numeric, whether to consider interaction in two-way repeated ANOVA (1) or not (0).

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

aof *ANOVA*

Description

Perform anova and only return p values and MSres (for Fisher's LSD)

Usage

```
aof(x, cls)
```

Arguments

x	Input the data to perform ANOVA
cls	Input class labels

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`aov.between` *Perform Two-way ANOVA*

Description

Perform Two-way ANOVA Perform between-subjects anova

Usage

```
aov.between(x)
```

Arguments

`x`	Input data to perform 2-way ANOVA

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`aov.repeated` *Perform Two-way ANOVA*

Description

Perform Two-way ANOVA Perform repeated measure one-way anova

Usage

```
aov.repeated(x, time.fac)
```

Arguments

`x`	Input the data
`time.fac`	Input the time factor

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`aov.within` *Perform Two-way ANOVA*

Description

Perform Two-way ANOVA Perform within-subjects anova

Usage

```
aov.within(x, time.fac)
```

Arguments

`x`	Input the data
`time.fac`	Input the time factor

Author(s)

Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`ASCAfun.res` *Function to perform ASCA*

Description

Perform ASCA

Usage

```
ASCAfun.res(X, Fac)
```

Arguments

`X`	Input list of compounds
`Fac`	Numeric McGill University, Canada License: GNU GPL (¿= 2)

Author(s)

Jeff Xia <`jeff.xia@mcgill.ca`>

ASCAfun1 *Function to perform ASCA*

Description

Perform ASCA

Usage

```
ASCAfun1(X, Design, Fac)
```

Arguments

X	Numeric, number of compounds
Design	Number of levels in the factor
Fac	Numeric, the factor

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

ASCAfun2 *Function to perform ASCA*

Description

Perform ASCA

Usage

```
ASCAfun2(X, Desa, Desb, Fac)
```

Arguments

X	Numeric, number of compounds
Desa	Number of levels in the factor TIME
Desb	Number of levels in the other factor
Fac	Numeric, the factor

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

calcCV *Calculatre CV method*

Description

Calculatre CV method

Usage

```
calcCV(xset)
```

Arguments

xset	XCMSnExp Object, this object is produced by 'calculateSet_doe' function.

Author(s)

Zhiqiang Pang <zhiqiang.pang@mail.mcgill.ca> Jeff Xia <jeff.xia@mcgill.ca> Mcgill University License: GNU GPL (¿= 2)

calcGaussianS *Calculatre Gaussian Peak Ratio method*

Description

Calculatre Gaussian Peak Ratio method

Usage

```
calcGaussianS(mSet, object, useNoise, BPPARAM = bpparam())
```

Arguments

mSet	MetaboAnalystR Object, this object is produced by 'calculateSet_doe' function.
object	MSnExp object, the trimmed or the original data (Generated by ImportRawMSData function with "inMemory" mode).
useNoise	Numeric, the noise level removed to evalute the gaussian peak.
BPPARAM	MulticoreParam method, used to set the parallel method. Default is bpparam().

Author(s)

Zhiqiang Pang <zhiqiang.pang@mail.mcgill.ca> Jeff Xia <jeff.xia@mcgill.ca> Mcgill University License: GNU GPL (¿= 2)

`calcPPS2` *Calculate PPS method*

Description

Calculate PPS method

Usage

```
calcPPS2(xset, isotopeIdentification = c("IPO", "CAMERA"), ...)
```

Arguments

`xset`	xcmsSet Object, this object is produced by 'calculateSet_doe' function, and transformed with as(objec,'xcmsSet') function.
`isotopeIdentification`	Character, IsotopeIdentidication method, usually includes 'IPO' and 'CAMERA'.

Author(s)

Zhiqiang Pang <zhiqiang.pang@mail.mcgill.ca> Jeff Xia <jeff.xia@mcgill.ca> Mcgill University License: GNU GPL (¿= 2)

`calcRCS_GSValues` *Calculatre RCS and GS method*

Description

Calculatre RCS and GS method

Usage

```
calcRCS_GSValues(xset)
```

Arguments

`xset`	XCMSnExp Object, this object is produced by 'calculateSet_doe' function.

Author(s)

Zhiqiang Pang <zhiqiang.pang@mail.mcgill.ca> Jeff Xia <jeff.xia@mcgill.ca> Mcgill University License: GNU GPL (¿= 2)

`calculateConcISO` *Calculate Concentration ISO*

Description

Assuming independent random distribution of FA, the most probable frequency will be the product of the each component. Note: the data is concentration, we need to get frequncies - percentage w.r.t the total nmol. the result is the saved as separate files for each lipid class data for each FA class, first col is sample name

Usage

```
calculateConcISO(dat, cls.name, cls.num, min.file, prob.file)
```

Arguments

`dat`	Input the data
`cls.name`	Input the class names
`cls.num`	Input the number of classes
`min.file`	Input the min file
`prob.file`	Input the prob file

Author(s)

Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`CalculateFeatureRanking`
Calculates feature importance

Description

Perform calculation of feature importance (AUC, p value, fold change)

Usage

```
CalculateFeatureRanking(mSetObj=NA, clust.num=5)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)
`clust.num`	Numeric, input the number of clusters for cluster-analysis

Author(s)

Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`CalculateGlobalTestScore` *Quantitative enrichment analysis with globaltest*

Description

Various enrichment analysis algorithms

Usage

```
CalculateGlobalTestScore(mSetObj = NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`calculateGPRT` *Alignment Method*

Description

Alignment Method

Usage

```
calculateGPRT(mSet, param)
```

Arguments

`mSet`	mSet object, the data produced by 'calculatePPKs' function.
`Set_parameters`	Matrix, the parameters combination produced automatically according to
`task`	Numeric, task order for XCMS paramters table to run the peak picking and alignment.

Author(s)

Zhiqiang Pang <zhiqiang.pang@mail.mcgill.ca> Jeff Xia <jeff.xia@mcgill.ca> Mcgill University License: GNU GPL (¿= 2)

CalculateHyperScore *Over-representation analysis using hypergeometric tests*

Description

Over-representation analysis using hypergeometric tests The probability is calculated from obtaining equal or higher number of hits using 1-phyper. Since phyper is a cumulative probability, to get P(X¿=hit.num) =¿ P(X¿(hit.num-1))

Usage

```
CalculateHyperScore(mSetObj = NA)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

CalculateImpVarCutoff *Calculate the Important Variable Cutoff*

Description

This function calculates the all important features based on a specfic cutoff.

Usage

```
CalculateImpVarCutoff(mSetObj, spe.thresh, lev.thresh)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)
spe.thresh	alpha threshold, less is better, default less than 5 percentile based chi-square note: spe and leverage are vectors, not a single value, but a list to store the result note: the last model is Model.res, no spe Calculate leverage cutoff based on permutation Calculate the reference distribution of leverages note: leverage.perm is a list with each member in a 3 column matrix
lev.thresh	leverage threshold, the higher better, default more than 95 percentile of permuted leverage

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

CalculateOraScore *Calculate ORA score*

Description

Calculate the over representation analysis score

Usage

```
CalculateOraScore(mSetObj=NA, nodeImp, method)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)
nodeImp	Indicate the pathway topology analysis, "rbc" for relative-betweeness centrality, and "dgr" for out-degree centrality.
method	is "fisher" or "hyperg"

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

CalculatePairwiseDiff *Calculate Pairwise Differences*

Description

Mat are log normalized, diff will be ratio. Used in higher functions.

Usage

```
CalculatePairwiseDiff(mat)
```

Arguments

mat	Input matrix of data to calculate pair-wise differences.

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`calculatePPKs` *Peak picking Method*

Description

Peak picking Method

Usage

```
calculatePPKs(object, object_mslevel, param, BPPARAM = bpparam(), msLevel = 1)
```

Arguments

`object`	MSnExp object, the trimmed or the original data.
`object_mslevel`	List, the parsed metabolomics scans produced by PeakPicking_prep.
`BPPARAM`	MulticoreParam method, used to set the parallel method. Default is bpparam().
`msLevel`	Numeric, to specifiy the msLevel, only 1 permitted for now. 2 will be supported in the near future.
`xcmsSetParameters`	Matrix, the parameters combination produced automatically according to
`task`	Numeric, task order for XCMS paramters table to run the peak picking and alignment.

Author(s)

Zhiqiang Pang <zhiqiang.pang@mail.mcgill.ca> Jeff Xia <jeff.xia@mcgill.ca> Mcgill University License: GNU GPL (¿= 2)

`CalculateQeaScore` *Calculate quantitative enrichment score*

Description

Calculate quantitative enrichment score

Usage

```
CalculateQeaScore(mSetObj=NA, nodeImp, method)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)
`nodeImp`	Indicate the pathway topology analysis, "rbc" for relative-betweeness centrality, and "dgr" for out-degree centrality.
`method`	Indicate the pathway enrichment analysis, global test is "gt" and global ancova is "ga".

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`calculateSet_doe` *Cluster of Peak Picking and Alignment*

Description

Cluster of Peak Picking and Alignment

Usage

```
calculateSet_doe(
  object,
  object_mslevel,
  Set_parameters,
  task = 1,
  BPPARAM = bpparam()
)
```

Arguments

`object`	MSnExp object, the trimmed or the original data.
`object_mslevel`	List, the parsed metabolomics scans produced by PeakPicking_prep.
`Set_parameters`	Matrix, the parameters combination produced automatically according to the primary parameters input.
`task`	Numeric, task order for XCMS paramters table to run the peak picking and alignment.
`BPPARAM`	MulticoreParam method, used to set the parallel method. Default is bpparam().

Author(s)

Zhiqiang Pang <zhiqiang.pang@mail.mcgill.ca> Jeff Xia <jeff.xia@mcgill.ca> Mcgill University License: GNU GPL (¿= 2)

`CalculateSSP` *Single sample profiling to compare with*

Description

reference concentrations stored in the library

Usage

```
CalculateSSP(mSetObj = NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`CheckMetaDataConsistency`
Check if data are ready for meta-analysis

Description

This function determines if all annotated data are ready for meta-analysis

Usage

```
CheckMetaDataConsistency(mSetObj = NA, combat = TRUE)
```

Arguments

`mSetObj`	Input name of the created mSet Object
`combat`	Adjust for batch effects, logical variable: TRUE = adjust for batch effects using an empirical Bayes framework (R package sva), FALSE = no batch effect adjustment.

Author(s)

Jeff Xia<`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

CleanData *Perform data cleaning*

Description

Cleans data and removes -Inf, Inf, NA, negative and 0s.

Usage

```
CleanData(bdata, removeNA = T, removeNeg = T, removeConst = T)
```

Arguments

bdata	Input data to clean
removeNA	Logical, T to remove NAs, F to not.
removeNeg	Logical, T to remove negative numbers, F to not.
removeConst	Logical, T to remove samples/features with 0s, F to not.

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

CleanDataMatrix *Clean the data matrix*

Description

Function used in higher functinos to clean data matrix

Usage

```
CleanDataMatrix(ndata)
```

Arguments

ndata	Input the data to be cleaned

`CleanNumber` *Replace infinite numbers*

Description

Replace -Inf, Inf to 99999 and -99999

Usage

```
CleanNumber(bdata)
```

Arguments

`bdata`	Input matrix to clean numbers

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`ClearNegatives` *Data processing: Dealing with negative values*

Description

Operates on dataSet$proc after dealing with missing values

Usage

```
ClearNegatives(mSetObj = NA, method = "abs")
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)
`method`	Input the method to clear negatives

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`ClearStrings` *Remove spaces*

Description

Remove from, within, leading and trailing spaces

Usage

```
ClearStrings(query)
```

Arguments

`query`	Input the query to clear

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`ClearUserDir` *Clear folder and memory*

Description

Clear the current folder and objects in memory

Usage

```
ClearUserDir(mSetObj = NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`Compound_function_mzlist` *Makes adducts*

Description

Makes adducts

Usage

```
Compound_function_mzlist(ms_mode, mw)
```

`ComputeAverageCurve` *Compute average ROC curve*

Description

Compute the average ROC curve

Usage

```
ComputeAverageCurve(perf, avg.method)
```

Arguments

`perf`	Input the average
`avg.method`	Input the name of the method to compute the average curve

Author(s)

Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`computeConc` *Lipid analysis*

Description

The upper limit for each combination is considered to be the minimal of the fatty acid concentration (nmol fatty acid/gram of sample) X is the lopomics data obtained above the result is the saved as separate files for each lipid class

Usage

```
computeConc(X, iso = "y")
```

Arguments

`X`	Input the data
`iso`	Default is set to "y"

Author(s)

Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`ComputeHighLow` *Compute the 95 percent interval for threshold ROC*

Description

Computes the 95 percent interval only for the y-axis. Utility function, called upon by higher functions

Usage

```
ComputeHighLow(perf)
```

Arguments

`perf`	Input the performance

Author(s)

Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`Convert2Mummichog` *Convert mSetObj to proper format for MS Peaks to Pathways module*

Description

Following t-test analysis, this functions converts the results from the mSetObj to the proper format for mummichog analysis

Usage

```
Convert2Mummichog(mSetObj = NA, rt = FALSE)
```

Arguments

`mSetObj`	Input the name of the created mSetObj.

Author(s)

Jasmine Chong, Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`CreateAnalNullMsg` *Create null message for analysis Creates a message for the Sweave report*

Description

Creates a message stating that no analyses were performed on your data.

Usage

```
CreateAnalNullMsg()
```

`CreateANOVAdoc` *Create report of analyses*

Description

Report generation using Sweave Create ANOVA document

Usage

```
CreateANOVAdoc(mSetObj = NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`CreateAOV2doc` *Create report of analyses*

Description

Report generation using Sweave ANOVA

Usage

```
CreateAOV2doc(mSetObj = NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`CreateASCAdoc` *Create report of analyses*

Description

Report generation using Sweave Random Forest ASCA

Usage

```
CreateASCAdoc(mSetObj = NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`CreateBiomarkerInputDoc`
Create biomarker analysis report: Data Input

Description

Report generation using Sweave Power analysis report, data input documentation.

Usage

```
CreateBiomarkerInputDoc(mSetObj = NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jasmine Chong McGill University, Canada License: GNU GPL (¿= 2)

`CreateBiomarkerIntr` *Create biomarker analysis report: Introduction*

Description

Report generation using Sweave Biomarker analysis report introduction

Usage

```
CreateBiomarkerIntr()
```

Author(s)

Jasmine Chong McGill University, Canada License: GNU GPL (¿= 2)

`CreateBiomarkerOverview`
Create biomarker analysis report: Overview

Description

Report generation using Sweave Power analysis report overview

Usage

```
CreateBiomarkerOverview()
```

Author(s)

Jasmine Chong McGill University, Canada License: GNU GPL (¿= 2)

`CreateBiomarkerRatioOverview`
Create biomarker analysis report: Normalization, ratio

Description

Report generation using Sweave Biomarker analysis, ratio option

Usage

```
CreateBiomarkerRatioOverview(mSetObj = NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jasmine Chong McGill University, Canada License: GNU GPL (¿= 2)

`CreateBiomarkerRnwReport` *Create report of analyses (Biomarker)*

Description

Report generation using Sweave Puts together the analysis report

Usage

```
CreateBiomarkerRnwReport(mSetObj, usrName)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)
`usrName`	Input the name of the user

Author(s)

Jasmine Chong McGill University, Canada License: GNU GPL (¿= 2)

`CreateCorrDoc` *Create report of analyses*

Description

Report generation using Sweave Create correlation document

Usage

```
CreateCorrDoc(mSetObj = NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

---

createCVset *Separate data set using k-fold cross validation (CV)*

Description

Separate data set with k-fold CV, used in higher function

Usage

```
createCVset(groupN, kfold, rseed)
```

Arguments

groupN	Input the size of the group
kfold	Input the number of cross-validations
rseed	Input the random seed

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

CreateEBAMdoc *Create report of analyses*

Description

Report generation using Sweave Create EBAM document Note: the search for delta (SAM) and a0 (EBAM) will not be plotted it is only exploration, and may cause potential inconsistentcies.

Usage

```
CreateEBAMdoc(mSetObj = NA)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`CreateEnrichAnalDoc` *Create report of analyses (Met Enrichment)*

Description

Report generation using Sweave Metabolite enrichment analysis report, analysis

Usage

```
CreateEnrichAnalDoc()
```

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`CreateEnrichInputDoc` *Create report of analyses (Met Enrichment)*

Description

Report generation using Sweave Metabolite enrichment analysis report data input

Usage

```
CreateEnrichInputDoc(mSetObj = NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`CreateEnrichIntr` *Create report of analyses (Met Enrichment)*

Description

Report generation using Sweave Metabolite enrichment analysis report introduction

Usage

```
CreateEnrichIntr()
```

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`CreateEnrichORAdoc` *Create report of analyses (Met Enrichment)*

Description

Report generation using Sweave Metabolite enrichment analysis report, over representation analysis (ORA)

Usage

```
CreateEnrichORAdoc(mSetObj = NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`CreateEnrichOverview` *Create report of analyses (Met Enrichment)*

Description

Report generation using Sweave Metabolite enrichment analysis report overview

Usage

```
CreateEnrichOverview()
```

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`CreateEnrichProcessDoc` *Create report of analyses (Met Enrichment)*

Description

Report generation using Sweave Metabolite enrichment analysis report enrichment process

Usage

```
CreateEnrichProcessDoc(mSetObj = NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`CreateEnrichQEAdoc` *Create report of analyses (Met Enrichment)*

Description

Report generation using Sweave Metabolite enrichment analysis report Quantitative enrichment analysis

Usage

```
CreateEnrichQEAdoc(mSetObj = NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`CreateEnrichRnwReport` *Create report of analyses (Met Enrichment)*

Description

Report generation using Sweave Metabolite enrichment analysis report

Usage

```
CreateEnrichRnwReport(mSetObj, usrName)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)
`usrName`	Input the name of the user

Author(s)

Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`CreateEnrichSSPdoc` *Create report of analyses (Met Enrichment)*

Description

Report generation using Sweave Metabolite enrichment analysis report Single sampling profiling

Usage

```
CreateEnrichSSPdoc(mSetObj = NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`CreateFooter` *Create report of analyses (Met Enrichment)*

Description

Report generation using Sweave Metabolite enrichment analysis report footer

Usage

```
CreateFooter()
```

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`CreateGraph` *Create igraph from the edgelist saved from graph DB and decompose into subnets*

Description

Function for the network explorer module, prepares user's data for network exploration.

Usage

```
CreateGraph(mSetObj = NA)
```

Arguments

`mSetObj`	Input name of the created mSet Object

`CreateGSEAAnalTable` *Create Mummichog report of analyses*

Description

Report generation using Sweave Function to create a summary table of mummichog analysis

Usage

```
CreateGSEAAnalTable(mSetObj = NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jasmine Chong McGill University, Canada License: GNU GPL (¿= 2)

`CreateHCdoc` *Create report of analyses*

Description

Report generation using Sweave Create hierarchical clustering document

Usage

```
CreateHCdoc(mSetObj = NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`CreateHeatmap2doc` *Create report of analyses*

Description

Report generation using Sweave 2-way heatmap

Usage

```
CreateHeatmap2doc(mSetObj = NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`CreateIntegPathwayAnalysisRnwReport`
Create report of analyses (IntegPathwayAnalysis)

Description

Report generation using Sweave Puts together the analysis report

Usage

```
CreateIntegPathwayAnalysisRnwReport(mSetObj, usrName)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)
`usrName`	Input the name of the user

Author(s)

Jasmine Chong McGill University, Canada License: GNU GPL (¿= 2)

`CreateIntegratedPathwayAnalInputDoc`
Create integrated pathway report: Data Input

Description

Report generation using Sweave integrated pathway report, data input documentation.

Usage

```
CreateIntegratedPathwayAnalInputDoc(mSetObj = NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jasmine Chong McGill University, viewingCanada License: GNU GPL (¿= 2)

`CreateIntegratedPathwayAnalIntr` *Create integrated pathway analysis report: Introduction*

Description

Report generation using Sweave Integrated pathwayr analysis report introduction

Usage

```
CreateIntegratedPathwayAnalIntr()
```

Author(s)

Jasmine Chong McGill University, Canada License: GNU GPL (¿= 2)

`CreateIntegratedPathwayDoc` *Create integrated pathway analysis report*

Description

Report generation using Sweave Biomarker analysis report, ROC Curve Based Model Creation and Evaluation

Usage

```
CreateIntegratedPathwayDoc(mSetObj = NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jasmine Chong McGill University, Canada License: GNU GPL (¿= 2)

`CreateIntegratedPathwayGeneMapTable`

Create a x-table for gene name mapping

Description

Report generation using Sweave Function to create a table for gene name mapping

Usage

```
CreateIntegratedPathwayGeneMapTable(mSetObj = NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jasmine Chong McGill University, Canada License: GNU GPL (¿= 2)

`CreateIntegratedPathwayNameMapTable`

Create a x-table for compound name mapping

Description

Report generation using Sweave Function to create a table for compound name mapping

Usage

```
CreateIntegratedPathwayNameMapTable(mSetObj = NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jasmine Chong McGill University, Canada License: GNU GPL (¿= 2)

`CreateIntegratedPathwayResultsTable` *Create a x-table for pathway results*

Description

Report generation using Sweave Function to create a table for pathway results

Usage

```
CreateIntegratedPathwayResultsTable(mSetObj = NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jasmine Chong McGill University, Canada License: GNU GPL (¿= 2)

`CreateiPCAdoc` *Create report of analyses*

Description

Report generation using Sweave For Interactive PCA

Usage

```
CreateiPCAdoc(mSetObj = NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`CreateKMdoc` *Create report of analyses*

Description

Report generation using Sweave Create Kmeans partitional clustering document

Usage

```
CreateKMdoc(mSetObj = NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`CreateLadder` *R-code for R-SVM*

Description

use leave-one-out / Nfold or bootstrape to permute data for external CV build SVM model and use mean-balanced weight to sort genes on training set and recursive elimination of least important genes

Usage

```
CreateLadder(Ntotal, Nmin = 5)
```

Arguments

`Ntotal`	Total number
`Nmin`	Minimum number, default set to 5

Author(s)

Dr. Xin Lu, Research Scientist Biostatistics Department, Harvard School of Public Health create a decreasing ladder for recursive feature elimination

CreateLibFromKEGG *Creates cpd.tree*

Description

Creates cpd.tree

Usage

```
CreateLibFromKEGG(cpd.lib, pathways, org)
```

CreateMappingResultTable *Creates the mapping result table*

Description

Creates the mapping result table

Usage

```
CreateMappingResultTable(mSetObj = NA)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)

CreateMBdoc *Create report of analyses*

Description

Report generation using Sweave Multivariate Bayes

Usage

```
CreateMBdoc(mSetObj = NA)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`CreateMetaAnalTable` *Create Mummichog report of analyses*

Description

Report generation using Sweave Function to create a summary table of mummichog analysis

Usage

```
CreateMetaAnalTable(mSetObj = NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jasmine Chong McGill University, Canada License: GNU GPL (¿= 2)

`CreateMetaAnalysisDEdoc`
Create MetaAnalysis analysis report: Data Normalization

Description

Report generation using Sweave Meta-Analysis, data normalization documentation.

Usage

```
CreateMetaAnalysisDEdoc(mSetObj = NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jasmine Chong McGill University, Canada License: GNU GPL (¿= 2)

`CreateMetaAnalysisInputDoc`

Create MetaAnalysis analysis report: Data Input

Description

Report generation using Sweave Power analysis report, data input documentation.

Usage

```
CreateMetaAnalysisInputDoc(mSetObj = NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jasmine Chong McGill University, Canada License: GNU GPL (¿= 2)

`CreateMetaAnalysisIntr`

Create MetaAnalysis analysis report: Introduction

Description

Report generation using Sweave MetaAnalysis analysis report introduction

Usage

```
CreateMetaAnalysisIntr()
```

Author(s)

Jasmine Chong McGill University, Canada License: GNU GPL (¿= 2)

`CreateMetaAnalysisNORMdoc`

Create MetaAnalysis analysis report: Data Normalization

Description

Report generation using Sweave Meta-Analysis, data normalization documentation.

Usage

```
CreateMetaAnalysisNORMdoc(mSetObj = NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jasmine Chong McGill University, Canada License: GNU GPL (¿= 2)

`CreateMetaAnalysisOutput`

Create MetaAnalysis analysis report: Data Normalization

Description

Report generation using Sweave MetaAnalysis analysis, data normalization documentation.

Usage

```
CreateMetaAnalysisOutput(mSetObj = NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jasmine Chong McGill University, Canada License: GNU GPL (¿= 2)

`CreateMetaAnalysisOverview`
Create MetaAnalysis analysis report: Overview

Description

Report generation using Sweave Power analysis report overview

Usage

```
CreateMetaAnalysisOverview()
```

Author(s)

Jasmine Chong McGill University, Canada License: GNU GPL (¿= 2)

`CreateMetaAnalysisRnwReport`
Create report of analyses (Meta-Analysis)

Description

Report generation using Sweave Puts together the analysis report

Usage

```
CreateMetaAnalysisRnwReport(mSetObj, usrName)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)
`usrName`	Input the name of the user

Author(s)

Jasmine Chong McGill University, Canada License: GNU GPL (¿= 2)

`CreateMetaTable` *Create MetaAnalysis table of results*

Description

Report generation using Sweave Function to create a table containing meta-analysis results.

Usage

```
CreateMetaTable(mSetObj = NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jasmine Chong McGill University, Canada License: GNU GPL (¿= 2)

`CreateModelBiomarkersDoc` *Create biomarker analysis report: ROC Curve Based Model Creation and Evaluation*

Description

Report generation using Sweave Biomarker analysis report, ROC Curve Based Model Creation and Evaluation

Usage

```
CreateModelBiomarkersDoc(mSetObj = NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jasmine Chong McGill University, Canada License: GNU GPL (¿= 2)

`CreateMultiBiomarkersDoc`

Create biomarker analysis report: Multivariate Biomarker Analysis

---

Description

Report generation using Sweave Biomarker analysis report, Multivariate Biomarker Analysis

Usage

```
CreateMultiBiomarkersDoc(mSetObj = NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jasmine Chong McGill University, Canada License: GNU GPL (¿= 2)

`CreateMummichogAnalTable`

Create Mummichog report of analyses

Description

Report generation using Sweave Function to create a summary table of mummichog analysis

Usage

```
CreateMummichogAnalTable(mSetObj = NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jasmine Chong McGill University, Canada License: GNU GPL (¿= 2)

`CreateMummichogAnalysisDoc`

Create mummichog analysis report

Description

Report generation using Sweave Mummichog analysis report

Usage

```
CreateMummichogAnalysisDoc(mSetObj = NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jasmine Chong McGill University, Canada License: GNU GPL (¿= 2)

`CreateMummichogInputDoc`

Create Mummichog analysis report: Data Input

Description

Report generation using Sweave Mummichog analysis report, data input documentation.

Usage

```
CreateMummichogInputDoc(mSetObj = NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jasmine Chong McGill University, Canada License: GNU GPL (¿= 2)

CreateMummichogIntro *Create mummichog analysis report: Introduction*

Description

Report generation using Sweave Mummichog analysis report introduction

Usage

```
CreateMummichogIntro()
```

Author(s)

Jasmine Chong McGill University, Canada License: GNU GPL (¿= 2)

CreateMummichogLibs *Create Mummichog Libraries from KEGG*

Description

Function to create mummichog libraries from MetaboAnalyst pathway libraries (metpa). Outputs the RDS files in the current working directory. RDS files are saved using the KEGG organism code.

Usage

```
CreateMummichogLibs("~/Desktop/MetaboAnalyst/mummichog/2020_mummichog_libs/test", kegg_compounds_202
```

Arguments

folder	Input the path of the folder containing the metpa rda files.
kegg_compounds	Input the name of the KEGG dictionary containing the KEGG compound IDs, KEGG compopund names, and molecular weight.

`CreateMummichogOverview`
Create Mummichog analysis report: Overview

Description

Report generation using Sweave Mummichog analysis report overview

Usage

```
CreateMummichogOverview()
```

Author(s)

Jasmine Chong McGill University, Canada License: GNU GPL (¿= 2)

`CreateMummichogRnwReport`
Create report of analyses (Biomarker)

Description

Report generation using Sweave Puts together the analysis report

Usage

```
CreateMummichogRnwReport(mSetObj, usrName)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)
`usrName`	Input the name of the user

Author(s)

Jasmine Chong McGill University, Canada License: GNU GPL (¿= 2)

`CreateNetworkExplorerDoc`

Create integrated pathway analysis report

Description

Report generation using Sweave Biomarker analysis report, ROC Curve Based Model Creation and Evaluation

Usage

```
CreateNetworkExplorerDoc(mSetObj = NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jasmine Chong McGill University, Canada License: GNU GPL (¿= 2)

`CreateNetworkExplorerInputDoc`

Create network explorer: Data Input

Description

Report generation using Sweave network explorer report, data input documentation.

Usage

```
CreateNetworkExplorerInputDoc(mSetObj = NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jasmine Chong McGill University, viewingCanada License: GNU GPL (¿= 2)

`CreateNetworkExplorerIntr`
Create integrated pathway analysis report: Introduction

Description

Report generation using Sweave Network explorer report introduction

Usage

```
CreateNetworkExplorerIntr()
```

Author(s)

Jasmine Chong McGill University, Canada License: GNU GPL (¿= 2)

`CreateNetworkExplorerOverview`
Create network explorer report: Overview

Description

Report generation using Sweave for the network explorer report overview

Usage

```
CreateNetworkExplorerOverview()
```

Author(s)

Jasmine Chong McGill University, Canada License: GNU GPL (¿= 2)

`CreateNetworkExplorerRnwReport`
Create report of analyses (Network Explorer)

Description

Report generation using Sweave Puts together the analysis report

Usage

```
CreateNetworkExplorerRnwReport(mSetObj, usrName)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)
usrName	Input the name of the user

Author(s)

Jasmine Chong McGill University, Canada License: GNU GPL (¿= 2)

CreateNetworkGeneMapTable

Create a x-table for gene name mapping

Description

Report generation using Sweave Function to create a table for gene name mapping

Usage

```
CreateNetworkGeneMapTable(mSetObj = NA)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jasmine Chong McGill University, Canada License: GNU GPL (¿= 2)

CreateNetworkNameMapTable

Create a x-table for compound name mapping

Description

Report generation using Sweave Function to create a table for compound name mapping

Usage

```
CreateNetworkNameMapTable(mSetObj = NA)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jasmine Chong McGill University, Canada License: GNU GPL (¿= 2)

`CreateNORMdoc` *Create report of analyses*

Description

Report generation using Sweave Create normalization document

Usage

```
CreateNORMdoc(mSetObj = NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`CreateOPLSDAdoc` *Create report of analyses*

Description

Report generation using Sweave Create OPLSDA document

Usage

```
CreateOPLSDAdoc(mSetObj = NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`CreatePathAnalDoc` *Create report of analyses (Met Pathway)*

Description

Report generation using Sweave Metabolomic pathway analysis Create pathway analysis doc

Usage

```
CreatePathAnalDoc(mSetObj = NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`CreatePathInputDoc` *Create report of analyses (Met Pathway)*

Description

Report generation using Sweave Metabolomic pathway analysis Create data input doc

Usage

```
CreatePathInputDoc()
```

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`CreatePathIntr` *Create report of analyses (Met Pathway)*

Description

Report generation using Sweave Metabolomic pathway analysis Introduction

Usage

```
CreatePathIntr()
```

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`CreatePathProcessDoc` *Create report of analyses (Met Pathway)*

Description

Report generation using Sweave Metabolomic pathway analysis Create MetPA process

Usage

```
CreatePathProcessDoc(mSetObj = NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`CreatePathResultDoc` *Create report of analyses (Met Pathway)*

Description

Report generation using Sweave Metabolomic pathway analysis Create MetPA results doc

Usage

```
CreatePathResultDoc(mSetObj = NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`CreatePathRnwReport` *Create report of analyses (Met Pathway)*

Description

Report generation using Sweave Metabolomic pathway analysis write .Rnw file template

Usage

```
CreatePathRnwReport(mSetObj, usrName)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)
`usrName`	Input the name of the user

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`CreatePCAdoc` *Create report of analyses*

Description

Report generation using Sweave Create PCA document

Usage

```
CreatePCAdoc(mSetObj = NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`CreatePLSdoc` *Create report of analyses*

Description

Report generation using Sweave Create PLS document

Usage

```
CreatePLSdoc(mSetObj = NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`CreatePowerAnalDoc` *Create power analysis report: Power Analysis*

Description

Report generation using Sweave Power analysis report, analysis

Usage

```
CreatePowerAnalDoc(mSetObj)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`CreatePowerInputDoc` *Create power analysis report: Data Input*

Description

Report generation using Sweave Power analysis report, data input documentation.

Usage

```
CreatePowerInputDoc(mSetObj = NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jasmine Chong McGill University, Canada License: GNU GPL (¿= 2)

`CreatePowerIntr` *Create power analysis report: Introduction*

Description

Report generation using Sweave Power analysis report introduction

Usage

```
CreatePowerIntr()
```

Author(s)

Jasmine Chong McGill University, Canada License: GNU GPL (¿= 2)

`CreatePowerOverview` *Create power analysis report: Overview*

Description

Report generation using Sweave Power analysis report overview

Usage

```
CreatePowerOverview()
```

Author(s)

Jasmine Chong McGill University, Canada License: GNU GPL (¿= 2)

`CreatePowerParametersDoc`
Create power analysis report: Power Parameter Selection

Description

Report generation using Sweave Power analysis report, parameter selection

Usage

```
CreatePowerParametersDoc(mSetObj = NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

CreatePowerRnwReport *Create report of analyses (Power)*

Description

Report generation using Sweave Put together the analysis report

Usage

```
CreatePowerRnwReport(mSetObj, usrName)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)
usrName	Input the name of the user

Author(s)

Jasmine Chong McGill University, Canada License: GNU GPL (¿= 2)

CreateRatioTable *Create report of analyses*

Description

Report generation using Sweave Function to create a summary table for biomarker analysis: included metabolite ratios

Usage

```
CreateRatioTable(mSetObj = NA)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jasmine Chong McGill University, Canada License: GNU GPL (¿= 2)

`CreateRFdoc` *Create report of analyses*

Description

Report generation using Sweave Create Random Forest document

Usage

```
CreateRFdoc(mSetObj = NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`CreateRHistAppendix` *Create report of analyses*

Description

Report generation using Sweave Create footer

Usage

```
CreateRHistAppendix()
```

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`CreateROCLabelsTable` *Create a x-table for newly classified samples*

Description

Report generation using Sweave Function to create a table for newly classified samples

Usage

```
CreateROCLabelsTable(mSetObj = NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jasmine Chong McGill University, Canada License: GNU GPL (¿= 2)

`CreateSAMdoc` *Create report of analyses*

Description

Report generation using Sweave Create SAM document

Usage

```
CreateSAMdoc(mSetObj = NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

CreateSemiTransColors *Create semitransparant colors*

Description

Create semitransparant colors for a given class label

Usage

```
CreateSemiTransColors(cls)
```

Arguments

cls	Input class labels

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

CreateSOMdoc *Create report of analyses*

Description

Report generation using Sweave Create SOM partitional clustering document

Usage

```
CreateSOMdoc(mSetObj = NA)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`CreateSPLSDAdoc` *Create report of analyses*

Description

Report generation using Sweave Create sPLS-DA document

Usage

```
CreateSPLSDAdoc(mSetObj = NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`CreateStatIntr` *Create report of analyses*

Description

Report generation using Sweave Create header

Usage

```
CreateStatIntr()
```

Author(s)

Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`CreateStatIOdoc` *Create report of analyses*

Description

Report generation using Sweave Read and process raw data

Usage

```
CreateStatIOdoc(mSetObj = NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`CreateStatRnwReport` *Create report for statistical analysis module*

Description

Report generation using Sweave Write .Rnw file template

Usage

```
CreateStatRnwReport(mSetObj, usrName)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)
`usrName`	Input the name of the user

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`CreateSummaryTable` *Create report of analyses*

Description

Report generation using Sweave Create a summary table for each type of uploaded data csv table has 5 col: sampleID, feature #, zero, missing #

Usage

```
CreateSummaryTable(mSetObj = NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`CreateSVMdoc` *Create report of analyses*

Description

Report generation using Sweave Create R-SVM document

Usage

```
CreateSVMdoc(mSetObj = NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

CreateTimeSeriesAnalNullMsg

Create null analysis message for time-series sweave report

Description

Creates empty time-series analysis message

Usage

```
CreateTimeSeriesAnalNullMsg()
```

CreateTimeSeriesIOdoc

Create report of analyses (Met Pathway)

Description

Report generation using Sweave Metabolomic pathway analysis, time-series Read and process the raw data

Usage

```
CreateTimeSeriesIOdoc(mSetObj = NA)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

CreateTimeSeriesRnwReport

Create report of analyses (Met Pathway)

Description

Report generation using Sweave Metabolomic pathway analysis Create timeseries .Rnw file template

Usage

```
CreateTimeSeriesRnwReport(mSetObj, usrName)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)
usrName	Input the name of the user

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

CreateUnivarBiomarkersDoc

Create power analysis report: Biomarker Univariate Analysis

Description

Report generation using Sweave Biomarker analysis report, Univariate Analysis

Usage

```
CreateUnivarBiomarkersDoc(mSetObj = NA)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jasmine Chong McGill University, Canada License: GNU GPL (¿= 2)

CreateUNIVdoc

Create report of analyses

Description

Report generation using Sweave Create univariate analyses document

Usage

```
CreateUNIVdoc(mSetObj = NA)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`CreateUnivROCTable` *Create summary table for univariate ROC analysis*

Description

Report generation using Sweave Function to create a summary table for univariate biomarker analysis

Usage

```
CreateUnivROCTable()
```

Author(s)

Jasmine Chong McGill University, Canada License: GNU GPL (¿= 2)

`CreateVennMetaTable` *Create MetaAnalysis table of results for Venn Diagram*

Description

Report generation using Sweave Function to create a table containing meta-analysis results.

Usage

```
CreateVennMetaTable(mSetObj = NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jasmine Chong McGill University, Canada License: GNU GPL (¿= 2)

`creatPeakTable` *creatPeakTable*

Description

creatPeakTable

Usage

```
creatPeakTable(xset)
```

Author(s)

Zhiqiang Pang, Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`CrossReferencing` *Various functions for mapping b/w names & database identifiers Given a list of compound names or ids, find matched name or ids from selected databases*

Description

Given a list of compound names or ids find matched name or IDs from selected databases

Usage

```
CrossReferencing(
  mSetObj = NA,
  q.type,
  hmdb = T,
  pubchem = T,
  chebi = F,
  kegg = T,
  metlin = F
)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects).
`q.type`	Input the query type, "name" for compound names, "hmdb" for HMDB IDs, "kegg" for KEGG IDs, "pubchem" for PubChem CIDs, "chebi" for ChEBI IDs, "metlin" for METLIN IDs, and "hmdb_kegg" for a both KEGG and HMDB IDs.
`hmdb`	Logical, T to cross reference to HMDB, F to not.

`pubchem`	Logical, T to cross reference to PubChem, F to not.
`chebi`	Logical, T to cross reference to CheBI, F to not.
`kegg`	Logical, T to cross reference to KEGG, F to not.
`metlin`	Logical, T to cross reference to MetLin, F to not.

Author(s)

Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`CVTest.LRmodel` *Calculate ROC performance with CV*

Description

Calculate ROC performance with CV

Usage

```
CVTest.LRmodel(data.in, fmla.in, kfold = 10, run.stepwise = FALSE)
```

Arguments

`data.in`	Input matrix of data
`fmla.in`	Input for generalized linear model
`kfold`	Numeric
`run.stepwise`	Logical

Author(s)

Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`C_imodwt_r` *Internal C fucntion - C_imodwt_r*

Description

Internal C fucntion - C_imodwt_r

Usage

```
C_imodwt_r(y, z, N, j, L, ht, gt, XX)
```

References

Percival, D. B. and A. T. Walden (2000) Wavelet Methods for Time Series Analysis, Cambridge University Press.

`C_modwt_r` *Internal C fucntion - C_modwt_r*

Description

Internal C fucntion - C_modwt_r

Usage

```
C_modwt_r(X, N, j, L, ht, gt, W, V)
```

References

Percival, D. B. and A. T. Walden (2000) Wavelet Methods for Time Series Analysis, Cambridge University Press.

`Densitygrouping_slave` *Densitygrouping_slave*

Description

Densitygrouping_slave

Usage

```
Densitygrouping_slave(
  x,
  bw,
  densFrom,
  densTo,
  densN,
  sampleGroups,
  sampleGroupTable,
  minFraction,
  minSamples,
  maxFeatures
)
```

Author(s)

Zhiqiang Pang, Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada

doCompoundMapping *Perform compound mapping*

Description

Perform compound mapping

Usage

```
doCompoundMapping(cmpd.vec, q.type)
```

Arguments

cmpd.vec	Input compound vector
q.type	Query type

doGeneIDMapping *Convert different gene IDs into entrez IDs for downstream analysis*

Description

Gene ID mapping, gene annotation, compound mapping, KEGG mapping

Usage

```
doGeneIDMapping(q.vec, org, type)
```

Arguments

q.vec	Input the query
org	Input the organism type
type	Input the type of data to annotate

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

doKEGG2NameMapping *Perform KEGG to compound name mapping*

Description

Perform KEGG to compound name mapping

Usage

```
doKEGG2NameMapping(kegg.vec)
```

Arguments

kegg.vec	Input vector of KEGG compounds

doKOFiltering *Utility function*

Description

Returns matched KO in the same order (NA if no match)

Usage

```
doKOFiltering(ko.vec, type)
```

Arguments

ko.vec	Input the vector containing KOs
type	Input the type

EBAM.Init *For EBAM analysis*

Description

deteriming a0, only applicable for z.ebam (default)

Usage

```
EBAM.Init(
  mSetObj = NA,
  isPaired,
  isVarEq,
  nonPar,
  A0 = -99,
  delta,
  imgA0,
  imgSig
)
```

Arguments

`mSetObj`	Input name of the created mSet Object
`isPaired`	Logical
`isVarEq`	Logical

Author(s)

Jeff Xia<`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`ExperimentsCluster_doe`

Experiment Functions of DoE

Description

This function is used to perform the test with Design of Experiment on the parameters dataset.

Usage

```
ExperimentsCluster_doe(
  object,
  object_mslevel,
  params,
  isotopeIdentification,
  BPPARAM = bpparam(),
  nSlaves = 4,
  ...
)
```

Arguments

object	MSnExp object, the trimmed or the original data.
object_mslevel	List, the parsed metabolomics scans produced by PeakPicking_prep.
isotopeIdentification	Character, IsotopeIdentidication method, usually includes 'IPO' and 'CAMERA'.
BPPARAM	MulticoreParam method, used to set the parallel method. Default is bpparam().
nSlave	Numeric, core number used to perform the parallel based optimization.

Author(s)

Zhiqiang Pang <zhiqiang.pang@mail.mcgill.ca> Jeff Xia <jeff.xia@mcgill.ca> Mcgill University License: GNU GPL (¿= 2)

ExtractMS2data *Extract MS2 Data*

Description

This function returns a list of spectra that matches with a user specified precursor m/z.

Usage

```
ExtractMS2data(filename, peakParams, mzmin, mzmax)
```

Arguments

filename	Name of the file (e.g. mzML, mzXML)
peakParams	Object containing parameters for peak picking.
mzmin	Minimum m/z when selecting a precursor from peak list
mzmax	Maximum m/z when selecting a precursor from peak list

Author(s)

Jasmine Chong <jasmine.chong@mail.mcgill.ca>, Mai Yamamoto <yamamoto.mai@mail.mcgill.ca>, and Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`FC.Anal.paired` *Fold change analysis, paired*

Description

Perform paired fold change analysis

Usage

```
FC.Anal.paired(
  mSetObj = NA,
  fc.thresh = 2,
  percent.thresh = 0.75,
  cmp.type = 0
)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)
`fc.thresh`	Fold-change threshold, numeric input
`percent.thresh`	Numeric input, from 0 to 1 to indicate the significant count threshold
`cmp.type`	Comparison type, 0 for group 1 minus group 2, and 1 for group

Author(s)

Jeff Xia<`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`FC.Anal.unpaired` *Fold change analysis, unpaired*

Description

Perform fold change analysis, method can be mean or median

Usage

```
FC.Anal.unpaired(mSetObj, fc.thresh=2, cmp.type = 0)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)
`fc.thresh`	Fold-change threshold, numeric input
`cmp.type`	Comparison type, 0 for group 1 minus group 2, and 1 for group 1 minus group 2

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

FeatureCorrelation *Pattern hunter*

Description

Calculate correlation of all other feature to a given feature name

Usage

```
FeatureCorrelation(mSetObj = NA, dist.name, varName)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)
dist.name	Input the name of the distance measure
varName	Input the variable name

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

---

fgsea2 *Pre-ranked gsea adapted for untargeted metabolomics*

Description

Pre-ranked gsea adapted for untargeted metabolomics

Usage

```
fgsea2(
  mSetObj,
  pathways,
  stats,
  ranks,
  nperm,
  minSize = 1,
  maxSize = Inf,
  nproc = 0,
  gseaParam = 1,
  BPPARAM = NULL
)
```

`fillpathways` *Fill in the pathways*

Description

Fill in the pathways

Usage

```
fillpathways(f)
```

`FilterVariable` *Methods for non-specific filtering of variables*

Description

This is a function that filters the dataset, dependent on the user-specified method for filtering. The function applies a filtering method, ranks the variables within the dataset, and removes variables based on its rank. The final dataset should contain no more than than 5000 variables for effective computing.

Usage

```
FilterVariable(mSetObj=NA, filter, qcFilter, rsd)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)
`filter`	Select the filter option, "rsd" which is the relative standard deviation, "nrsd" which is the non-parametric relative standard deviation, "mean" which is the mean, "sd" which is the standard deviation, "mad" which is the median absolute deviation, or "iqr" which is the interquantile range.
`qcFilter`	Filter the variables based on QC samples - True (T), or use non-QC based filtering - False (F).
`rsd`	Define the relative standard deviation cut-off. Variables with a RSD greater than this number will be removed from the dataset. It is only necessary to specify this argument if qcFilter is True (T). Otherwise, it will not be used in the function.

Author(s)

Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

FisherLSD *Fisher for ANOVA*

Description

Perform Fisher LSD for ANOVA, used in higher function

Usage

```
FisherLSD(aov.obj, thresh)
```

Arguments

aov.obj	Input the anova object
thresh	Numeric, input the alpha threshold

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

FormatPeakList *Format Peak List*

Description

This function formats the CAMERA output to a usable format for MetaboAanlyst.

Usage

```
FormatPeakList(
  annotPeaks,
  annParams,
  filtIso = TRUE,
  filtAdducts = FALSE,
  missPercent = 0.5
)
```

Arguments

annotPeaks	The object created using the PerformPeakAnnotation.
annParams	The object created using the SetAnnotationParam function, containing user's specified or default parameters for downstream raw MS data pre-processing.
filtIso	Logical, filter out all isotopes except for [M]+ for positive ion mode and [M]- for negative ion mode. By default it is set to true.

filtAdducts	Logical, filter out all adducts except [M+H]+ for positive ion more and [M-H]- for negative ion mode. By default it is set to false.
missPercent	Numeric, specify the threshold to remove features missing in X% of samples. For instance, 0.5 specifies to remove features that are missing from 50% of all samples per group. Method is only valid when there are two groups.

Author(s)

Jasmine Chong <jasmine.chong@mail.mcgill.ca>, Mai Yamamoto <yamamoto.mai@mail.mcgill.ca>, and Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

---

genLogisticRegMdl *Develop a Logistic Regression Model with all of the combined k-fold CV subsets*

Description

Develop a Logistic Regression Model with all of the combined k-fold CV subsets

Usage

```
genLogisticRegMdl(x.train, y.train, x.test, y.test)
```

Arguments

x.train	Input the X training set
y.train	Input the Y training set
x.test	Input the X test set
y.test	Input the Y test set

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`Get.asca.tss` *Function for ASCA permutation*

Description

Dummy is used only for the purpose to maintain lapply API this is used for permutation on ANOVA paritions, not on the SCA/PCA part, so the number of selected components is not applicable in this step

Usage

```
Get.asca.tss(dummy, perm = T)
```

Arguments

`dummy`	Dummy variable
`perm`	Logical, TRUE by default

Author(s)

Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`Get.bwss` *Compute within group and between group sum of squares (BSS/WSS) for each row of a matrix which may have NA*

Description

Columns have labels, x is a numeric vector, cl is consecutive integers

Usage

```
Get.bwss(x, cl)
```

Arguments

`x`	Numeric vector
`cl`	Columns

Author(s)

Jeff Xia<`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`Get.ConcRef` *Get the concentration reference*

Description

Get the concentration reference

Usage

```
Get.ConcRef(mSetObj = NA, cmpd.nm)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)
`cmpd.nm`	Input the compound name

`Get.Leverage` *Fast leverage calculation for permutation purpose*

Description

note, the leverage combines all components the importance feature is for the factor not per components

Usage

```
Get.Leverage(XKw, Fac)
```

Arguments

`XKw`	Features
`Fac`	Factor

Author(s)

Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`Get.pAUC` *Calculate partial area under ROC curve*

Description

Calculate partial area under ROC curve

Usage

```
Get.pAUC(x, y, focus, cutoff)
```

Arguments

`x`	Input X
`y`	Input Y
`focus`	Method
`cutoff`	Numeric

Author(s)

Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`Get.pred` *Get predicted class probability*

Description

Get predicted class probability, used in higher function

Usage

```
Get.pred(x.train, y.train, x.test, y.test, clsMethod = "pls")
```

Arguments

`x.train`	Training X
`y.train`	Training Y
`x.test`	Test X
`y.test`	Test Y
`clsMethod`	Method to predict class, by default it is PLS

Author(s)

Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`Get.rpart.summary` *Get the text description of a recursive partitioning (rpart) result*

Description

x must be an rpart object

Usage

```
Get.rpart.summary(x)
```

Arguments

`x`	An Rpart object

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`Get.VIP` *Calculate variable importance of projection (VIP) score for PLS object*

Description

Users give a pls object ('oscorespls'=T), function calculates VIP score usually one VIP for each component, return is the average of all VIP

Usage

```
Get.VIP(pls.obj, comp = 2)
```

Arguments

`pls.obj`	Input the PLS object
`comp`	Numeric, input the number of components, by default it is 2

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`GetAbundanceLabel` *Determine value label for plotting*

Description

Concentration or intensity data type

Usage

```
GetAbundanceLabel(data.type)
```

Arguments

`data.type`	Input concentration or intensity data

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`GetAccuracyInfo` *Export biomarker accuracy information*

Description

Export biomarker accuracy information

Usage

```
GetAccuracyInfo(mSetObj = NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)

`GetAllDataNames` *Get all meta-analysis name data*

Description

Get all meta-analysis name data

Usage

```
GetAllDataNames()
```

GetAllKMClusterMembers *K-means analysis - cluster*

Description

K-means analysis - cluster

Usage

```
GetAllKMClusterMembers(mSetObj = NA)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)

GetAllSOMClusterMembers *SOM analysis*

Description

Get members for given cluster index, return a character string

Usage

```
GetAllSOMClusterMembers(mSetObj = NA)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`GetCandidateList` *Get all candidate compound names for a given index*

Description

Returns 3 coloumns - inx, name, score

Usage

```
GetCandidateList(mSetObj = NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`GetCircleInfo` *Export information about selected circle*

Description

Export information about selected circle

Usage

```
GetCircleInfo(mSetObj = NA)
```

Arguments

`mSetObj`	Input name of the created mSet Object

`GetCIs` *Get confidence intervals*

Description

For non-parametric tests, use quantiles, use normal (1.96*std.err) if parametric

Usage

```
GetCIs(data, param = F)
```

Arguments

`data`	Input data matrix
`param`	Logical, False by default

Author(s)

Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`GetCMD` *Retrieve last command from the Rhistory.R file*

Description

Fetches the last command from the Rhistory.R file

Usage

```
GetCMD(regexp)
```

Arguments

`regexp`	Retrieve last command from Rhistory file

Author(s)

Jeff Xia<`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`GetCompoundDetails` *Function to get adduct details from a specified compound*

Description

Function to get adduct details from a specified compound. The results will be both printed in the console as well as saved as a csv file. Note that performing this function multiple times will overwrite previous queries.

Usage

```
GetCompoundDetails(mSetObj = NA, cmpd.id)
```

Arguments

`mSetObj`	Input the name of the created mSetObj object.
`cmpd.id`	Input the name of the selected compound.

`GetConvertFullPath` *Perform utilities for cropping images*

Description

Obtain the full path to convert (from imagemagik) for cropping images

Usage

```
GetConvertFullPath()
```

Author(s)

Jeff Xia<`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`getDataFromTextArea` *Transform two column text to data matrix*

Description

Transform two column input text to data matrix (single column data frame)

Usage

```
getDataFromTextArea(txtInput, sep.type = "space")
```

Arguments

`txtInput`	Input text
`sep.type`	Indicate the seperator type for input text. Default set to "space"

Author(s)

Jeff Xia<`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`GetExtendRange` *Extend axis*

Description

Extends the axis range to both ends vec is the values for that axis unit is the width to extend, 10 will increase by 1/10 of the range

Usage

```
GetExtendRange(vec, unit = 10)
```

Arguments

`vec`	Input the vector
`unit`	Numeric

Author(s)

Jeff Xia<`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

GetFC *Used by higher functions to calculate fold change*

Description

Utility method to calculate FC, used in higher function

Usage

```
GetFC(mSetObj = NA, paired = FALSE, cmpType)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)
paired	Logical, true of false
cmpType	Numeric, 0 or 1

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

GetFeatureNumbers *Numbers for subset selection*

Description

Return a series of number for subsets selection

Usage

```
GetFeatureNumbers(feat.len)
```

Arguments

feat.len	Input the feature length

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

GetFinalNameMap *Return the final (after user selection) map as dataframe*

Description

Returns three columns: original name, HMDB name and KEGG ID, for enrichment and pathway analysis, respectively

Usage

```
GetFinalNameMap(mSetObj = NA)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

GetFisherPvalue *Get fisher p-values*

Description

Get fisher p-values

Usage

```
GetFisherPvalue(numSigMembers, numSigAll, numMembers, numAllMembers)
```

Arguments

numSigMembers	Number of significant members
numSigAll	Number of all significant features
numMembers	Number of members
numAllMembers	Number of all members

GetHTMLMetSet *Given a metset inx, return hmtl highlighted metset cmpds and references*

Description

Given a metset inx, return hmtl highlighted metset cmpds and references

Usage

```
GetHTMLMetSet(mSetObj = NA, msetNm)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)
msetNm	Input the name of the metabolite set

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

GetHTMLPathSet *Given a metset inx, return hmtl highlighted pathway cmpds*

Description

Given a metset inx, return hmtl highlighted pathway cmpds

Usage

```
GetHTMLPathSet(mSetObj = NA, msetNm)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)
msetNm	Input the name of the metabolite set

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`GetImpFeatureMat` *Get important feature matrix*

Description

feat.outp is a list that contains the ranked features in each cross validation (CV) and returns a two column matrix, col 1 = median ranking and col 2 = mean importance measure

Usage

```
GetImpFeatureMat(mSetObj = NA, feat.outp, bestFeatNum)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)
`feat.outp`	Input the list that contains the ranked features in each cross validation (CV) and returns a two column matrix, col 1 = median ranking and col 2 = mean importance measure
`bestFeatNum`	Numeric

Author(s)

Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`GetKEGGNodeInfo` *Retrieves KEGG node information*

Description

Retrieves KEGG node information

Usage

```
GetKEGGNodeInfo(pathName, g, width, height, usr = par("usr"))
```

Arguments

`g`	Input data
`width`	Input the width
`height`	Input the height
`usr`	Input the user
`path.id`	Input the path ID

`GetKMClusterMembers` *K-means analysis - cluster*

Description

Get the cluster members for given index add HTML color to the names based on its group membership

Usage

```
GetKMClusterMembers(mSetObj = NA, i)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)
`i`	Input the cluster index

Author(s)

Jeff Xia<`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`GetLassoFreqs` *Compute lasso frequency*

Description

Not part of default, need to perform function to compute lasso frequency msg: There are more than 500 variables and n¡m You may wish to restart and set use.Gram=FALSE

Usage

```
GetLassoFreqs(mSetObj = NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

GetLimmaResTable *Get result table from eBayes fit object*

Description

Get result table from eBayes fit object

Usage

```
GetLimmaResTable(fit.obj)
```

Arguments

fit.obj	eBayes fit object to parse to a table

GetMapTable *Get mapping table*

Description

Return results from compound name mapping in a table

Usage

```
GetMapTable(mSetObj = NA)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)

GetMaxPCAComp *For plotting PCA, selects max top 9 components*

Description

Rotate PCA analysis

Usage

```
GetMaxPCAComp(mSetObj = NA)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

GetMeanROC *Compute data points on the ROC curve*

Description

perf is the performance object from ROCR

Usage

```
GetMeanROC(perf)
```

Arguments

perf	Performance object from ROCR

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

GetMetaResultMatrix *Single.type return logFC or p value for individual data analysis*

Description

Single.type return logFC or p value for individual data analysis

Usage

```
GetMetaResultMatrix(mSetObj = NA, single.type = "fc")
```

Arguments

mSetObj	Input name of the created mSet Object
single.type	Default is "fc"

GetMetaSigHitsTable *Export the significant hits from meta-analysis*

Description

Export the significant hits from meta-analysis

Usage

```
GetMetaSigHitsTable(mSetObj = NA)
```

Arguments

mSetObj	Input name of the created mSet Object

GetMetSetName *Given a metset inx, give its name*

Description

Given a metset inx, give its name

Usage

```
GetMetSetName(mSetObj = NA, msetInx)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)
msetInx	Input the index of the metabolite set

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`GetMsetLibCheckMsg` *Get the library check messages*

Description

Get the library check messages

Usage

```
GetMsetLibCheckMsg(mSetObj = NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)

`GetMsetLibSearchResult`
Return metset search results

Description

since String[][] is not supported, have to return as 1D vector, matrix can be directly convert to vector, note default will be column first

Usage

```
GetMsetLibSearchResult(mSetObj = NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada

`GetMsetNames` *Return the selected metset library to java for display*

Description

Return the selected metset library to java for display

Usage

```
GetMsetNames(mSetObj = NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`GetMummichogPathSetDetails`
Function to get compound details from a specified pathway

Description

Function to get compound details from a specified pathway. The results will be both printed in the console as well as saved as a csv file. Note that performing this function multiple times will overwrite previous queries. Significant compounds will be indicated with an asterisk.

Usage

```
GetMummichogPathSetDetails(mSetObj = NA, msetNm)
```

Arguments

`mSetObj`	Input the name of the created mSetObj object.
`msetNm`	Input the name of the pathway

GetNetworkGeneMappingResultTable
Exports Gene-Mapping result into a table

Description

Exports Gene-Mapping result into a table

Usage

```
GetNetworkGeneMappingResultTable(mSetObj = NA)
```

Arguments

mSetObj	Input name of the created mSet Object

GetNewSampleNames *Obtain sample names and their class labels*

Description

Obtain sample names and their class labels

Usage

```
GetNewSampleNames(mSetObj = NA)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`GetORA.pathNames` *Export pathway names from ORA analysis*

Description

Export pathway names from ORA analysis

Usage

```
GetORA.pathNames(mSetObj = NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)

`GetORA.smpdbIDs` *Only for human pathways (SMPDB)*

Description

Only for human pathways + ath, eco, mmu & sce

Usage

```
GetORA.smpdbIDs(mSetObj = NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`GetORATable` *Get ORA table*

Description

Get ORA table

Usage

```
GetORATable(mSetObj = NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)

GetQEA.keggIDs *Only for human pathways (KEGG)*

Description

Only for human pathways + ath, eco, mmu & sce

Usage

```
GetQEA.keggIDs(mSetObj = NA)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

GetQEA.pathNames *Export pathway names from QEA analysis*

Description

Export pathway names from QEA analysis

Usage

```
GetQEA.pathNames(mSetObj = NA)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)

GetQEATable *QEA table*

Description

QEA table

Usage

```
GetQEATable(mSetObj = NA)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)

`GetRCommandHistory` *Export R Command History*

Description

Export R Command History

Usage

```
GetRCommandHistory(mSetObj = NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)

`GetRFConf.Table` *Classification performance table for random forest analysis*

Description

Classification performance table for random forest analysis

Usage

```
GetRFConf.Table(mSetObj = NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)

`GetRFConfMat` *Random Forest Confusion Matrix*

Description

Return double confusion matrix

Usage

```
GetRFConfMat(mSetObj = NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jeff Xia<`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

GetRFOOB *Random Forest OOB*

Description

Get the OOB error for the last signif

Usage

```
GetRFOOB(mSetObj = NA)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

GetRFSigMat *Random Forest Significance matrix*

Description

Significance measure, double brackets

Usage

```
GetRFSigMat(mSetObj = NA)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`GetROC.coords` *Return ROC corodinates with confidence intervals*

Description

Return ROC corodinates with confidence intervals

Usage

```
GetROC.coords(mSetObj = NA, fld.nm, val, plot = TRUE, imgNm)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)
`fld.nm`	The kind of input coordinate
`val`	The coordinates to look for
`plot`	Logical, by default set to TRUE
`imgNm`	Input the image name

Author(s)

Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`GetROCLassoFreq` *Get p-values from lasso*

Description

Get p-values from lasso

Usage

```
GetROCLassoFreq(data, cls)
```

Arguments

`data`	Input data
`cls`	Input class labels

Author(s)

Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`GetROCTtestP` *Get p-values for ROC*

Description

ROC p-vaues, used in higher function

Usage

```
GetROCTtestP(data, cls)
```

Arguments

`data`	Input data
`cls`	Input class labels

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`GetSampleSizeLadder` *Retrieve sample size ladder*

Description

Return sample size ladder, used in higher functions

Usage

```
GetSampleSizeLadder(maxNum)
```

Arguments

`maxNum`	Numeric

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

GetSelectedDataNames *Retrieve data names*

Description

Retrieve data names

Usage

```
GetSelectedDataNames(mSetObj = NA)
```

Arguments

mSetObj	Input name of the created mSet Object

GetSelectedDataNumber *Retrieve selected data numbers*

Description

Retrieve selected data numbers

Usage

```
GetSelectedDataNumber(mSetObj = NA)
```

Arguments

mSetObj	Input name of the created mSet Object

GetSigTable *Create Latex table*

Description

generate Latex table

Usage

```
GetSigTable(mat, method, data.type)
```

Arguments

mat	Input matrix
method	Input method to create table
data.type	Input the data type

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

GetSigTable.Anova *Sig Table for Anova*

Description

Sig Table for Anova

Usage

```
GetSigTable.Anova(mSetObj = NA)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)

GetSigTable.Aov2 *Sig table for AOV2*

Description

Sig table for AOV2

Usage

```
GetSigTable.Aov2(mSetObj = NA)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)

`GetSigTable.ASCA` *Table of features well modelled by ASCA*

Description

Table of features well modelled by ASCA

Usage

```
GetSigTable.ASCA(mSetObj = NA, nm)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)
`nm`	Input the name of the well modelled features

Author(s)

Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`GetSigTable.Corr` *Sig table for Correlation Analysis*

Description

Sig table for Correlation Analysis

Usage

```
GetSigTable.Corr(mSetObj = NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)

GetSigTable.EBAM *Sig table for EBAM*

Description

Sig table for EBAM

Usage

```
GetSigTable.EBAM(mSetObj = NA)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)

GetSigTable.FC *Sig Table for Fold-Change Analysis*

Description

Sig Table for Fold-Change Analysis

Usage

```
GetSigTable.FC(mSetObj = NA)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)

GetSigTable.MB *Sig table for MB analysis*

Description

Sig table for MB analysis

Usage

```
GetSigTable.MB(mSetObj = NA)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)

GetSigTable.RF *Sig table for random forest analysis*

Description

Sig table for random forest analysis

Usage

```
GetSigTable.RF(mSetObj = NA)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)

GetSigTable.SAM *Sig table for SAM*

Description

Sig table for SAM

Usage

```
GetSigTable.SAM(mSetObj = NA)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)

GetSigTable.SVM *Sig table for SVM*

Description

Sig table for SVM

Usage

```
GetSigTable.SVM(mSetObj = NA)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)

GetSigTable.TT *Sig Table for T-test Analysis*

Description

Sig Table for T-test Analysis

Usage

```
GetSigTable.TT(mSetObj = NA)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)

GetSigTable.Volcano *Sig table for Volcano Analysis*

Description

Sig table for Volcano Analysis

Usage

```
GetSigTable.Volcano(mSetObj = NA)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)

GetSOMClusterMembers *SOM analysis*

Description

Get members for given cluster index, return a character string

Usage

```
GetSOMClusterMembers(mSetObj = NA, i, j)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)
i	Index of X
j	Index of Y

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`GetSSPTable` *Replace the last column of the ssp.mat with the final selection from users*

Description

Replace the last column of the ssp.mat with the final selection from users

Usage

```
GetSSPTable(mSetObj = NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`GetSuggestedSAMDelta` *For SAM analysis*

Description

obtain a default delta with reasonable number of sig features and decent FDR

Usage

```
GetSuggestedSAMDelta(mSetObj = NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`GetSVMSigMat` *Recursive Support Vector Machine (R-SVM) Significance Measure*

Description

Return significance measure, double[][]

Usage

```
GetSVMSigMat(mSetObj = NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`GetTopInx` *Volcano indices*

Description

Get indices of top n largest/smallest number

Usage

```
GetTopInx(vec, n, dec = T)
```

Arguments

`vec`	Vector containing volcano indices
`n`	Numeric
`dec`	Logical, default set to TRUE

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`GetTrainTestSplitMat` *Make random partitions*

Description

Make random partitions, returns matrices indicating whether the observation is in train/test for each run note: try to get a balanced sampling for each group (classification) or each quantile (regression). This is very useful for unbalanced data

Usage

```
GetTrainTestSplitMat(y, propTraining = 2/3, nRuns = 30)
```

Arguments

`y`	Input the data
`propTraining`	By default set to 2/3
`nRuns`	By default set to 30

Author(s)

Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`GetTtestRes` *Retrieve T-test p-values*

Description

Utility method to get p values

Usage

```
GetTtestRes(mSetObj = NA, paired = FALSE, equal.var = TRUE, nonpar = F)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)
`paired`	Default set to FALSE
`equal.var`	Default set to TRUE
`nonpar`	Use non-parametric tests, default is set to FALSE

Author(s)

Jeff Xia<`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

GetTTSigMat *T-test matrix*

Description

Return a double matrix with 2 columns - p values and lod

Usage

```
GetTTSigMat(mSetObj = NA)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

GetUnivReport *Utility method to perform the univariate analysis automatically*

Description

The approach is computationally expensive,and fails more often get around: make it lazy unless users request, otherwise the default t-test will also be affected

Usage

```
GetUnivReport(mSetObj = NA)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

GetVariableLabel *Determine variable label for plotting*

Description

Determine data type, binned spectra, nmr peak, or ms peak

Usage

```
GetVariableLabel(data.type)
```

Arguments

data.type	Input the data type

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

GetVennGeneNames *Get Venn names*

Description

Get Venn names

Usage

```
GetVennGeneNames(mSetObj = NA, areas)
```

Arguments

mSetObj	Input name of the created mSet Object
areas	Input areas to retrieve names

GetXYCluster *Determine row/column number for plotting*

Description

Determine the number of rows and columns for a given total number of plots (used by Kmeans and SOM plots)

Usage

```
GetXYCluster(total)
```

Arguments

total	Input the total

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

GroupPeakList *Group peak list*

Description

Group peaks from the peak list based on position using the XCMS grouping algorithm (align peaks wrt, rt, and mz). For NMR peaks, need to change ppm -¿ mz and add dummy rt. If the data is 2-column MS, first need to add dummy rt. If the data is 3-column MS, the data can be used directly. The default mzwid for MS is 0.25 m/z, and for NMR is 0.03 ppm. The default bw is 30 for LCMS, and 5 for GCMS.

Usage

```
GroupPeakList(mSetObj=NA, mzwid, bw, minfrac, minsamp, max)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)
mzwid,	define the width of overlapping m/z slices to use for creating peak density chromatograms and grouping peaks across samples
bw,	define the bandwidth (standard deviation or half width at half maximum) of gaussian smoothing kernel to apply to the peak density chromatogram
minfrac,	define the minimum fraction of samples necessary in at least one of the sample groups for it to be a valid group
minsamp,	define the minimum number of samples necessary in at least one of the sample groups for it to be a valid group
max,	define the maximum number of groups to identify in a single m/z slice

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`heckbert` *Heckbert algorithm*

Description

function to calculate tick mark based on Heckbert algorithm available in the "labeling" package implemented by Justin Talbot adapted from the imagemap package Heckbert's labeling algorithm Heckbert, P. S. (1990) Nice numbers for graph labels, Graphics Gems I, Academic Press Professional, Inc.

Usage

```
heckbert(dmin, dmax, m)
```

Arguments

`dmin`	Heckbert
`dmax`	Heckbert
`m`	Heckbert

Author(s)

Justin Talbot <jtalbot@stanford.edu>

`HMDBID2KEGGID` *Given a vector of HMDBIDs, return a vector of KEGG IDs*

Description

This function, when given a vector of HMDBIDs, returns a vector of KEGG ID. HMDB standing for the Human Metabolome Database.

Usage

```
HMDBID2KEGGID(ids)
```

Arguments

`ids`	Input the vector of HMDB Ids

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`HMDBID2Name` *Given a vector of HMDBIDs, return a vector of HMDB compound names*

Description

This function, when given a vector of HMDBIDs, return a vector of HMDB compound names. HMDB standing for the Human Metabolome Database.

Usage

```
HMDBID2Name(ids)
```

Arguments

`ids`	Input the vector of HMDB Ids

Author(s)

Jeff Xia<`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`ImportRawMSData` *Import raw MS data*

Description

This function handles the reading in of raw MS data (.mzML, .CDF and .mzXML). Users must set their working directory to the folder containing their raw data, divided into two subfolders named their desired group labels. The function will output two chromatograms into the user's working directory, a base peak intensity chromatogram (BPIC) and a total ion chromatogram (TIC). Further, this function sets the number of cores to be used for parallel processing. It first determines the number of cores within a user's computer and then sets it that number/2.

Usage

```
ImportRawMSData(foldername, mode = "onDisk", ncores = 4, plotSettings)
```

Arguments

`foldername`	Character, input the file path to the folder containing the raw MS spectra to be processed.
`mode`	Character, the data input mode. Default is "onDisk" to avoid memory crash. "inMemory" will absorb data into the memory.
`plotSettings`	List, plotting parameters produced by SetPlotParam Function. "plot.opts" can be added through this function for samples numbers for plotting. Defalut is "default", "all" will apply all samples for plotting and may cause memory crash, especially for large sample dataset.

Author(s)

Zhiqiang Pang <zhiqiang.pang@mail.mcgill.ca>, Jasmine Chong <jasmine.chong@mail.mcgill.ca>, Mai Yamamoto <yamamoto.mai@mail.mcgill.ca>, and Jeff Xia <jeff.xia@mcgill.ca>
McGill University, Canada License: GNU GPL (¿= 2)

`ImportRawMSDataList` *Import raw MS data*

Description

This function handles the reading in of raw MS data (.mzML, .CDF and .mzXML). Users must provide a matrix with meta information about file such that each file has the name, file path, group class and extension type. The function will output two chromatograms into the user's working directory, a base peak intensity chromatogram (BPIC) and a total ion chromatogram (TIC). Further, this function sets the number of cores to be used for parallel processing. It first determines the number of cores within a user's computer and then sets it that number/2.

Usage

```
ImportRawMSDataList(
  dataset.meta,
  format = "png",
  dpi = 72,
  width = 9,
  par.cores = TRUE,
  plot = TRUE,
  bpis_name = "BPIS_",
  tics_name = "TICS_"
)
```

Arguments

`dataset.meta`	Matrix, input the meta data for files containing the raw MS spectra to be processed.
`format`	Character, input the format of the image to create.
`dpi`	Numeric, input the dpi of the image to create.
`width`	Numeric, input the width of the image to create.
`par.cores`	Logical, if true, the function will automatically set the number of parallel cores. If false, it will not.
`plot`	Logical, if true the function will create BPIS and TICS plots.
`bpis_name`	Character, input the name of the BPIS image to create.
`tics_name`	Character, input the name of the TICS image to create.

Author(s)

Jasmine Chong <jasmine.chong@mail.mcgill.ca>, Mai Yamamoto <yamamoto.mai@mail.mcgill.ca>, and Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

ImputeVar *Data processing: Replace missing variables*

Description

Replace missing variables by min/mean/median/KNN/BPCA/PPCA/svdImpute.

Usage

```
ImputeVar(mSetObj, method)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)
method	Select the option to replace missing variables, either replacement based on the minimum ("min), the mean ("mean"), or the median ("median") value of each feature columns, or several options to impute the missing values, using k-nearest neighbour ("KNN"), probabilistic PCA ("PPCA"), Bayesian PCA ("BPCA") method, or Singular Value Decomposition ("svdImpute")

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

InitDataObjects *Constructs a dataSet object for storing data*

Description

This functions handles the construction of a mSetObj object for storing data for further processing and analysis. It is necessary to utilize this function to specify to MetaboAnalystR the type of data and the type of analysis you will perform.

Usage

```
InitDataObjects(data.type, anal.type, paired=FALSE)
```

Arguments

`data.type`	The type of data, either list (Compound lists), conc (Compound concentration data), specbin (Binned spectra data), pktable (Peak intensity table), nmrpeak (NMR peak lists), mspeak (MS peak lists), or msspec (MS spectra data)
`anal.type`	Indicate the analysis module to be performed: stat, pathora, pathqea, msetora, msetssp, msetqea, ts, cmpdmap, smpmap, or pathinteg
`paired`	Indicate if the data is paired or not. Logical, default set to FALSE

Author(s)

Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`InitPowerAnal` *Function for power analysis*

Description

Perform power analysis, requires the SSPA R package.

Usage

```
InitPowerAnal(mSetObj, clsOpts)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)
`clsOpts`	For data with ¿2 groups, specify the two classes on which to perform power analysis, otherwise for data with 2 groups, "NA" will automatically select the 2 groups.

Author(s)

Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`InitStatAnalMode` *Introduction for statistical analysis module report Initialize Statistical Analysis Report*

Description

Introduction for statistical analysis module report Initialize Statistical Analysis Report

Usage

```
InitStatAnalMode()
```

`InitTimeSeriesAnal` *Create report of analyses (Met Pathway)*

Description

Report generation using Sweave Metabolomic pathway analysis, time-series analysis

Usage

```
InitTimeSeriesAnal()
```

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`iPCA.Anal` *Perform PCA analysis, prepare file for interactive liveGraphics3D*

Description

Perform PCA analysis, prepares a JSON file for interactive liveGraphics3D, as well as interactive 3D PCA score and loading plots using the plotly R package. These plots are saved in the created mSetObj; to view these, type "mSetObj$imgSet$time$score3d" to view the interactive score plot, and "mSetObj$imgSet$time$load3d" to view the interactive loading plot.

Usage

```
iPCA.Anal(mSetObj, fileNm)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)
`fileNm`	select a file name

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`isEmptyMatrix` *Sig table matrix is empty*

Description

Test if a sig table matrix is empty

Usage

```
isEmptyMatrix(mat)
```

Arguments

`mat`	Matrix to test if empty

Author(s)

Jeff Xia<`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`IsSmallSmplSize` *Check if the sample size is small*

Description

Returns whether or not the sanity check found that there were too many groups in the dataset containing too few samples. It will return a 0 if the data passes the check, or will return a 1 if the data does not.

Usage

```
IsSmallSmplSize(mSetObj=NA)
```

Arguments

`mSetObj`	Input name of the created mSet Object

IsSpectraProcessingOK *Check if the spectra processing is ok*

Description

Check if the spectra processing is ok

Usage

```
IsSpectraProcessingOK(mSetObj = NA)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects).

KEGGID2HMDBID *Given a vector of KEGGIDs, return a vector of HMDB ID*

Description

This functionn, when given a vector of KEGGIDs, returns a vector of HMDB IDs. HMDB standing for the Human Metabolome Database.

Usage

```
KEGGID2HMDBID(ids)
```

Arguments

ids	Vector of KEGG ids

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

KEGGID2Name — *Given a vector containing KEGGIDs, returns a vector of KEGG compound names*

Description

This function, given a vector containing KEGGIDs, returns a vector of KEGG compound names.

Usage

```
KEGGID2Name(ids)
```

Arguments

ids	Vector of KEGG ids

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

KEGGPATHID2SMPDBIDs — *Given a vector containing KEGG pathway IDs, return a vector containing SMPDB IDs (only for hsa)*

Description

This function, when given a vector of KEGG pathway IDs, return a vector of SMPDB IDs (only for hsa). SMPDB standing for the Small Molecule Pathway Database, and hsa standing for human serum albumin.

Usage

```
KEGGPATHID2SMPDBIDs(ids)
```

Arguments

ids	Vector of KEGG pathway IDs

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

Kmeans.Anal *K-means analysis*

Description

Perform K-means analysis

Usage

```
Kmeans.Anal(mSetObj = NA, clust.num)
```

Arguments

mSetObj	Input name of the created mSet Object
clust.num	Numeric, input the number of clusters for K-means analysis

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

kwtest *Kruskal-Wallis*

Description

Perform Kruskal-Wallis Test

Usage

```
kwtest(x, cls)
```

Arguments

x	Input data to perform Kruskal-Wallis
cls	Input class labels

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`LoadKEGGKO_lib` *Utility function for PerformKOEnrichAnalysis_KO01100*

Description

Utility function for PerformKOEnrichAnalysis_KO01100

Usage

```
LoadKEGGKO_lib(category)
```

Arguments

`category`	Module or pathway

`LoadKEGGLib` *Load KEGG library*

Description

Load different libraries

Usage

```
LoadKEGGLib(libType, libNm)
```

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`LoadSmpLib` *Load pathway library*

Description

Load pathway library

Usage

```
LoadSmpLib(mSetObj = NA)
```

Arguments

`mSetObj`	Input name of the created mSet Object

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`LogNorm` *Column-wise Normalization*

Description

Column-wise norm methods, when x is a column Options for log, zero mean and unit variance, and several zero mean and variance/SE

Usage

```
LogNorm(x, min.val)
```

Arguments

`x`	Input data
`min.val`	Input minimum value

Author(s)

Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada

`LSD.test` *Calculate Fisher's Least Significant Difference (LSD)*

Description

Adapted from the 'agricolae' package

Usage

```
LSD.test(y, trt, alpha = 0.05)
```

Arguments

`y`	Input Y
`trt`	Input trt
`alpha`	Numeric, default is 0.05

Author(s)

Jeff Xia<`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

make_cpdlib *Gets names and exact mass of all cpds (cpd.lib)*

Description

Gets names and exact mass of all cpds (cpd.lib)

Usage

```
make_cpdlib(org)
```

make_cpdlist *Utility function to create compound lists for permutation analysis*

Description

From a vector of m/z features, this function outputs a vector of compounds.

Usage

```
make_cpdlist(mSetObj=NA, input_mzs)
```

Arguments

mSetObj	Input the name of the created mSetObj
input_mzs	The vector of randomly drawn m/z features.

Author(s)

Jasmine Chong, Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

make_ecpdlist *Utility function to create compound lists for permutation analysis*

Description

From a vector of m/z features, this function outputs a vector of compounds.

Usage

```
make_cpdlist(mSetObj=NA, input_mzs)
```

Arguments

mSetObj	Input the name of the created mSetObj
input_mzs	The vector of randomly drawn m/z features.

Author(s)

Jasmine Chong, Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

map *sPLS-DA Map*

Description

map variable for (s)plsda

Usage

```
map(Y)
```

Arguments

Y	Input data

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`MapCmpd2KEGGNodes` *Utility function for PrepareQueryJson*

Description

Utility function for PrepareQueryJson

Usage

```
MapCmpd2KEGGNodes(cmpds, net = "ko01100")
```

Arguments

`cmpds`	Input the compounds
`net`	Input the network name

`MapKO2KEGGEdges` *Utility function for PrepareQueryJson*

Description

Utility function for PrepareQueryJson

Usage

```
MapKO2KEGGEdges(kos, net = "ko01100")
```

Arguments

`kos`	Input the KOs
`net`	Input the name of the network

Match.Pattern *Match pattern for correlation analysis*

Description

Match pattern for correlation analysis

Usage

```
Match.Pattern(mSetObj = NA, dist.name = "pearson", pattern = NULL)
```

Arguments

mSetObj	Input the name of the created mSetObj
dist.name	Input the distance method, default is set to pearson
pattern	Set the pattern, default is set to NULL

MergeDatasets *Utility function for PrepareQueryJson*

Description

Utility function for PrepareQueryJson

Usage

```
MergeDatasets(dataSet1, dataSet2)
```

Arguments

dataSet1	Input the first dataset
dataSet2	Input the second dataset

`MergeDuplicates` *Merge duplicated columns or rows by their mean*

Description

dim 1 =¿ row, dim 2 =¿ column

Usage

```
MergeDuplicates(data, dim = 2)
```

Arguments

`data`	Input the data
`dim`	Numeric, input the dimensions, default is set to 2

`MetaboAnalystR` *MetaboAnalystR: A package for computating the notorious bar statistic.*

Description

The MetaboAnalystR package provides a pipeline for metabolomics processing.

MetaboAnalystR functions

The MetaboAnalystR functions ...

`MetaboliteMappingExact` *Mapping from different metabolite IDs*

Description

For compound names to other ids, can do exact or approximate matches For other IDs, except HMDB ID, all others may return multiple/non-unique hits Multiple hits or non-unique hits will allow users to manually select

Usage

```
MetaboliteMappingExact(mSetObj = NA, q.type)
```

Arguments

mSetObj	Input the name of the created mSetObj.
q.type	Inpute the query-type, "name" for compound names, "hmdb" for HMDB IDs, "kegg" for KEGG IDs, "pubchem" for PubChem CIDs, "chebi" for ChEBI IDs, "metlin" for METLIN IDs, and "hmdb_kegg" for a both KEGG and HMDB IDs.

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

---

mSet2xcmsSet *mSet2xcmsSet*

Description

mSet2xcmsSet

Usage

```
mSet2xcmsSet(mSet)
```

Author(s)

Zhiqiang Pang, Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

MSspec.fillPeaks *Function to fill in missing peaks*

Description

For each sample in the processed MS spectra data, this function will fill in missing peaks using the fillPeaks function from the XCMS package. First, the function will identify any peak groups that are missing any peaks from the samples and will then fill in those peaks by rereading the raw data and integrating signals at those regions to create a new peak.

Usage

```
MSspec.fillPeaks(mSetObj=NA)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)

`MSspec.rtCorrection` *Retention time correction for LC/GC-MS spectra*

Description

Performs retention time correction for LC/GC-MS spectra using the XCMS package. Following retention time correction, the object dataSet will be regrouped.

Usage

```
MSspec.rtCorrection(mSetObj=NA, bw=30)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)
`bw`	Numeric, define the bandwidth (standard deviation or half width at half maximum) of gaussian smoothing kernel to apply to the peak density chromatogram

---

`multi.stat` *Get multiple category statistics*

Description

Get multiple category statistics

Usage

```
multi.stat(pred, resp)
```

Arguments

`pred`	Input predictions
`resp`	Input responses

Author(s)

Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`mz.trim_random` *Data trimming Method Based on Random MS*

Description

Trim raw data scan signal randomly in the mz dimension.

Usage

```
mz.trim_random(raw_data, ms_list)
```

Arguments

`raw_data`	MSnExp object, the raw data that has been read in memory.
`ms_list`	List, the names list of all scans.

Author(s)

Zhiqiang Pang <zhiqiang.pang@mail.mcgill.ca> Jeff Xia <jeff.xia@mcgill.ca> Mcgill University License: GNU GPL (¿= 2)

`mz.trim_specific` *Data trimming Method Based on Specific MS*

Description

Trim data based on specific mz values. Positive values will be specially retained, while the negative values will be removed.

Usage

```
mz.trim_specific(raw_data, ms_list, mz, mzdiff = 100)
```

Arguments

`raw_data`	MSnExp object, the raw data that has been read in memory.
`ms_list`	List, the names list of all scans.
`mz`	Numeric, the specifric mz value that will be kept or removed.
`mzdiff`	Numeric, the deviation (ppm) for the 'mz' values. Default is 100.

Author(s)

Zhiqiang Pang <zhiqiang.pang@mail.mcgill.ca> Jeff Xia <jeff.xia@mcgill.ca> Mcgill University License: GNU GPL (¿= 2)

`Noise_evaluate` *Noise_evaluation based on Kernal density model*

Description

This functions handles the evaluation on the data noise (noise and prefilter parameters) and the identification on the molecule weights deviation evaluation.

Usage

```
Noise_evaluate(raw_data)
```

Arguments

`raw_data`	MSnExp object, the (trimmed) data in memory produced by 'Perform-DataTrimming'.

Author(s)

Zhiqiang Pang <zhiqiang.pang@mail.mcgill.ca> Jeff Xia <jeff.xia@mcgill.ca> Mcgill University License: GNU GPL (¿= 2)

References

McLean C (2020). Autotuner: Automated parameter selection for untargeted metabolomics data processing

`Normalization` *Normalization*

Description

This function performs row-wise normalization, transformation, and scaling of your metabolomic data.

Usage

```
Normalization(mSetObj, rowNorm, transNorm, scaleNorm, ref=NULL, ratio=FALSE, ratioNum=20)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)
rowNorm	Select the option for row-wise normalization, "QuantileNorm" for Quantile Normalization, "ProbNormT" for Probabilistic Quotient Normalization without using a reference sample, "ProbNormF" for Probabilistic Quotient Normalization based on a reference sample, "CompNorm" for Normalization by a reference feature, "SumNorm" for Normalization to constant sum, "MedianNorm" for Normalization to sample median, and "SpecNorm" for Normalization by a sample-specific factor.
transNorm	Select option to transform the data, "LogNorm" for Log Normalization, and "CrNorm" for Cubic Root Transformation.
scaleNorm	Select option for scaling the data, "MeanCenter" for Mean Centering, "AutoNorm" for Autoscaling, "ParetoNorm" for Pareto Scaling, amd "RangeNorm" for Range Scaling.
ref	Input the name of the reference sample or the reference feature, use " " around the name.
ratio	This option is only for biomarker analysis.
ratioNum	Relevant only for biomarker analysis.

Author(s)

Jeff Xia <jeff.xia@mcgill.ca>, Jasmine Chong McGill University, Canada

---

OPLSDA.Permut *Perform OPLS-DA permutation*

Description

Orthogonal PLS-DA (from ropls) perform permutation, using training classification accuracy as indicator, for two or multi-groups

Usage

```
OPLSDA.Permut(mSetObj = NA, num = 100)
```

Arguments

mSetObj	Input name of the created mSet Object
num	Input the number of permutations, default is set to 100.

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`OPLSR.Anal` *Perform OPLS-DA*

Description

Orthogonal PLS-DA (from ropls) Add reg (regression i.e. if class order matters)

Usage

```
OPLSR.Anal(mSetObj = NA, reg = FALSE)
```

Arguments

`mSetObj`	Input name of the created mSet Object
`reg`	Logical

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`optimize.xcms.doe` *Overall Funtion for DoE*

Description

This function is the overall function to handle the starting of the optimization process and pre-define the parameters' range according to the input of the parameters.

Usage

```
optimize.xcms.doe(raw_data, param, ncore = 8)
```

Arguments

`raw_data`	MSnExp object, The trimmed or original data input for optimization.
`param`	List, the parameters lists set by 'SetPeakParam' function. The noise, prefilter and ppm values should be defined by AutoTuner in the previous steps.
`ncore`	Numeric, core number used to perform the parallel based optimization.

Author(s)

Zhiqiang Pang <zhiqiang.pang@mail.mcgill.ca> Jeff Xia <jeff.xia@mcgill.ca> Mcgill University License: GNU GPL (¿= 2)

`optimizxcms.doe.peakpicking` *Core Optimization Function of DoE*

Description

This function is the core for parameters optimization with Design of Experiment (DoE) method.

Usage

```
optimizxcms.doe.peakpicking(
  object = NULL,
  params = params,
  BPPARAM = bpparam(),
  nSlaves = 4,
  plot = F,
  ...
)
```

Arguments

`BPPARAM`	MulticoreParam method, used to set the parallel method. Default is bpparam().
`plot`	Logical, weather to plot the Contours plots of the DoE results.
`objet`	MSnExp object, the trimmed or the original data.
`param`	List, the parameters lists set by 'SetPeakParam' function. The noise, prefilter and ppm values should be defined by AutoTuner in the previous steps.
`nSlave`	Numeric, core number used to perform the parallel based optimization.

Author(s)

Zhiqiang Pang <`zhiqiang.pang@mail.mcgill.ca`> Jeff Xia <`jeff.xia@mcgill.ca`> Mcgill University License: GNU GPL (¿= 2)

`parseFisher` *Return only the signicant comparison names*

Description

Return only the signicant comparison names, used in higher function

Usage

```
parseFisher(fisher, cut.off)
```

Arguments

fisher	Input fisher object
cut.off	Numeric, set cut-off

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

---

parseTukey *Return only the signicant comparison names*

Description

Return only the signicant comparison names, used in higher function

Usage

```
parseTukey(tukey, cut.off)
```

Arguments

tukey	Input tukey output
cut.off	Input numeric cut-off

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

PCA.Anal *Perform PCA analysis*

Description

Perform PCA analysis, obtain variance explained, store item to PCA object

Usage

```
PCA.Anal(mSetObj = NA)
```

Arguments

mSetObj	Input name of the created mSet Object McGill University, Canada License: GNU GPL (¿= 2)

Author(s)

Jeff Xia<jeff.xia@mcgill.ca>

`PCA.Flip` *Rotate PCA analysis*

Description

Rotate PCA analysis

Usage

```
PCA.Flip(mSetObj = NA, axisOpt)
```

Arguments

`mSetObj`	Input name of the created mSet Object
`axisOpt`	Input the axis option

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`PCA.GENES` *Obtain principal components into a matrix that has more variables than individuals*

Description

X is a matrix that has as columns the compounds that were considered as variables in the PCA analysis. First we center the matrix by columns (Xoff) and then we obtain the eigenvalues and the eigenvectors of the matrix Xoff use the equivalences between the loadings and scores to obtain the solution

Usage

```
PCA.GENES(X)
```

Arguments

`X`	Input matrix that has as columns the compounds that were considered as variables in the PCA analysis

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`PeakPicking_centWave_slave` *PeakPicking_centWave_slave*

Description

PeakPicking_centWave_slave

Usage

```
PeakPicking_centWave_slave(x, param)
```

Author(s)

Zhiqiang Pang, Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`PeakPicking_core` *Calculate PPS method*

Description

Peak picking method. Specifically used for parameters optimization

Usage

```
PeakPicking_core(
  object,
  object_mslevel,
  param,
  BPPARAM = bpparam(),
  msLevel = 1L
)
```

Arguments

`object_mslevel`	List, prepared by findChromPeaks_prep function.
`param`	Parameters list.
`BPPARAM`	Parallel Method.
`msLevel`	msLevel. Only 1 is supported currently.
`xset`	MSnExp object.

Author(s)

Zhiqiang Pang <zhiqiang.pang@mail.mcgill.ca> Mcgill University License: GNU GPL (¿= 2)

PeakPicking_MatchedFilter_slave *PeakPicking_MatchedFilter_slave*

Description

PeakPicking_MatchedFilter_slave

Usage

```
PeakPicking_MatchedFilter_slave(x, param)
```

Author(s)

Zhiqiang Pang, Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

PeakPicking_prep *Data Preparation for ChromPeaking Finding*

Description

Data Preparation for ChromPeaking Finding

Usage

```
PeakPicking_prep(object)
```

Arguments

object	MSnExp object.

Author(s)

Zhiqiang Pang <zhiqiang.pang@mail.mcgill.ca> Mcgill University License: GNU GPL (¿= 2)

`Perform.ASCA` *Perform ASCA*

Description

The ASCA algorithm was adapted from the ASCA-genes method (analysis of variance (ANOVA) simultaneous component analysis) by Maria Jose Nueda (mj.nueda@ua.es) and Ana Conesa (aconesa@ivia.es)

Usage

```
Perform.ASCA(mSetObj = NA, a = 1, b = 2, x = 2, res = 2)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)
`a`	specify the number of components for facA
`b`	specify the number of components for facB
`x`	specify the number of components for interaction AB
`res`	specify the number of model residuals type is string, indicating the type of analysis "abc" separately "aab" facA joins with AB "bab" facB joins with AB

Author(s)

Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`Perform.ASCA.permute` *Perform ASCA model validation by permutation*

Description

Perform ASCA model validation by permutation we use Manly's unrestricted permutation of observations which esentially permutes the data over all cells in the designed experiment, then calculates the score for each main factor or interaction components. This will get the null distribution for all effects in one go

Usage

```
Perform.ASCA.permute(mSetObj=NA, perm.num)
```

Arguments

`mSetObj`	Input name of the created mSet Object
`perm.num`	Select the number of permutations, default is 20

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`Perform.Permut` *Perform permutation tests only for ROC Tester*

Description

Perform permutation tests for the ROC Curve Based Model Creation and Evaluation module

Usage

```
Perform.Permut(mSetObj=NA, perf.measure, perm.num, propTraining = 2/3)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)
`perf.measure`	Input the performance measure to rate the performance of the model, either the area under the ROC curve ("auroc") or the predictive accuracy ("accu")
`perm.num`	Input the number of permutations to perform
`propTraining`	Numeric, input the fraction of samples to set aside for training. Default is set to 2/3.

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`Perform.permutation` *Permutation*

Description

Perform permutation, options to change number of cores used

Usage

```
Perform.permutation(perm.num, fun)
```

Arguments

`perm.num`	Numeric, input the number of permutations to perform
`fun`	Dummy function

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`Perform.UnivROC` *Perform Classical Univariate ROC*

Description

Perform Classical Univariate ROC

Usage

```
Perform.UnivROC(mSetObj=NA, feat.nm, imgName, format="png", dpi=72, isAUC, isOpt, optMethod, isPartial
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)
`feat.nm`	Input the name of the feature to perform univariate ROC analysis
`imgName`	Input a name for the plot
`format`	Select the image format, png, of pdf.
`dpi`	Input the dpi. If the image format is pdf, users need not define the dpi. For png images, the default dpi is 72. It is suggested that for high-resolution images, select a dpi of 300.
`isAUC`	Logical, select T to compute the 95 percent confidence interval band and "F" to not
`isOpt`	Logical, show the optimal cutoff, T to show it and F to not
`optMethod`	Select the optimal cutoff by using either closest.topleft for closest to top-left corner or youden for farthest to the diagonal line (Youden)
`isPartial`	Logical, input T to calculate a partial ROC curve, and F to not
`measure`	Select the parameter to limit the calculation of the partial ROC curve, se for the X-axis (maximum false-positive rate) and sp for the Y-axis, representing the minimum true positive-rate
`cutoff`	Input the threshold to limit the calculation of the partial ROC curve, the number must be between 0 and 1.

Author(s)

Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`PerformAdductMapping` *Read Adduct List*

Description

This function reads in the user's adduct list and saves it as a matrix.

Usage

```
Read.AdductData(mSetObj=NA, adductList)
```

Arguments

`mSetObj`	Input the name of the created mSetObj object
`adductList`	Input the name of the adduct list

Author(s)

Jasmine Chong, Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`PerformApproxMatch` *Perform approximate compound matches*

Description

Given a query, perform approximate compound matching

Usage

```
PerformApproxMatch(mSetObj = NA, q)
```

Arguments

`mSetObj`	Input the name of the created mSetObj.
`q`	Input the q vector.

PerformBatchCorrection

Batch Effect Correction

Description

One is a batch containing summed concentrations of each sample the other contains the features aligned across all samples

Usage

```
PerformBatchCorrection(
  mSetObj = NA,
  imgName = NULL,
  Method = NULL,
  center = NULL
)
```

Arguments

mSetObj	Input name of the created mSet Object
imgName	Input the name of the plot to create
Method	Batch effect correction method, default is "Automatically". Specific method, including "Combat", "WaveICA","EigenMS","QC_RLSC","ANCOVA","RUV_random","RUV_2 and "CCMN".
center	The center point of the batch effect correction, based on "QC" or "", which means correct to minimize the distance between batches.

Author(s)

Zhiqiang Pang, Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

PerformCurrencyMapping

Map currency metabolites to KEGG & BioCyc

Description

This function maps the user selected list of compounds to its corresponding KEGG IDs and BioCyc IDs

Usage

```
PerformCurrencyMapping(mSetObj = NA)
```

Arguments

mSetObj	Input the name of the created mSetObj object

Author(s)

Jasmine Chong, Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

PerformCV.explore *Perform Monte-Carlo Cross Validation (MCCV)*

Description

Classification MCCV, aims to find the best feature subsets using default model parameters

Usage

```
PerformCV.explore(mSetObj, cls.method, rank.method="auroc", lvNum=2, propTraining=2/3)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)
cls.method	Select the classification method, "rf" for random forest classification, "pls" for PLS-DA, and "svm" for support vector machine
rank.method	Select the ranking method, "rf" for random forest mean decrease accuracy, "fisher" for Fisher's univariate ranking based on area under the curve "auroc" for univariate ranking based on area under the curve, "tt" for T-test univariate ranking based on area under the curve, "pls" for partial least squares, and "svm" for support vector machine
lvNum	Input the number of latent variables to include in the analyis, only for PLS-DA classification
propTraining	Input the proportion of samples to use for training

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`PerformCV.test` *Perform MCCV for manually selected features*

Description

MCCV for manually selected features (no additional feature selection)

Usage

```
PerformCV.test(mSetObj, method, lvNum, propTraining=2/3, nRuns=100)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)
`method`	Select the classification method, "rf" for random forest classification, "pls" for PLS-DA, and "svm" for support vector machine
`lvNum`	Input the number of latent variables to include in the analyis, only for PLS-DA classification
`propTraining`	Input the proportion of samples to use for training, by default it is 2/3
`nRuns`	Input the number of MCCV runs, by default it is 100

Author(s)

Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`PerformDataInspect` *Data inspectation*

Description

This functions provide a path for users to visually inspect their raw data before the data trimming so as to remove the dirty or significantly uneluted peaks.

Usage

```
PerformDataInspect(datapath, rt.range, mz.range, dimension = "3D", res = 100)
```

Arguments

datapath	Character, the path of the raw MS data files (.mzXML, .CDF and .mzML) for the visual and intuitive data inspectation.
rt.range	Numerics, a congregation of two values to define the lower and upper RT range (seconds) for users to inspect. This is an optional parameter, if absent, will display the MS of the whole RT range.
mz.range	Numerics, a congregation of two values to define the lower and upper mz range for users to inspect. This is an optional parameter, if absent, will display the MS of the whole mz range.
dimension	Character, the dimension for sample to display, including '2D' or '3D'. The default is '3D'.
res	Numeric, the resolution for data inspectation. The larger the value, the higher the resolution. The default value is 100. This value is usually clearly enough and also give consideration to the speed.

Author(s)

Zhiqiang Pang <zhiqiang.pang@mail.mcgill.ca> Jeff Xia <jeff.xia@mcgill.ca> Mcgill University License: GNU GPL (¿= 2)

PerformDataTrimming *Perform raw MS data trimming*

Description

This function performs the raw data trimming. This function will output an trimmed MSnExp file to memory or hardisk according to the choice of users must provide the data path for 'datapath', and optionally provide other corresponding parameters.

Usage

```
PerformDataTrimming(
  datapath,
  mode = "ssm",
  write = F,
  mz,
  mzdiff,
  rt,
  rtdiff,
  rt.idx = 1/15,
  plot = T
)
```

Arguments

`datapath`	Character, the path of the raw MS data files' folder/path (.mzXML, .CDF and .mzML) for parameters training.
`mode`	Character, mode for data trimming to select the chraracteristic peaks. Default is 'ssm'. Users could select random trimed according to mz value (mz_random) or RT value (rt_random). Besides, specific peaks at certain mz (mz_specific) or RT (rt_specific) could also be extracted. 'none' will not trim the data.
`write`	Logical, if true, will write the trimed data to the directory 'trimed' folder in the datapath. The data in memory will be kept.
`mz`	Numeric, mz value(s) for specific selection. Positive values means including (the values indicted) and negative value means excluding/removing.
`mzdiff`	Numeric, the deviation (ppm) of mz value(s).
`rt`	Numeric, rt value for specific selection. Positive values means including and negative value means excluding.
`rtdiff`	Numeric, the deviation (seconds) of rt value(s).
`rt.idx`	Numeric, the relative rt (retention time) range, from 0 to 1. 1 means all retention time will be retained, while 0 means none. Default is 1/15. If default rt.idx produce too few peaks, please consider increasing this value.
`plot`	Logical, if ture, will plot the chromatogram of the trimed data.

Author(s)

Zhiqiang Pang <`zhiqiang.pang@mail.mcgill.ca`> Jeff Xia <`jeff.xia@mcgill.ca`> Mcgill University License: GNU GPL (¿= 2)

`PerformDetailMatch` *Perform detailed name match*

Description

Given a query, perform compound matching.

Usage

```
PerformDetailMatch(mSetObj = NA, q)
```

Arguments

`mSetObj`	Input name of the created mSet Object.
`q`	Input the query.

PerformEachDEAnal *Performs differential expression analysis on individual data*

Description

This function performs DE analysis on individual data using the common matrix, which will be used/compared in later steps of the analysis (according to the p-value). The DE for each feature may be adjusted using the p-value.

Usage

```
PerformEachDEAnal(mSetObj = NA)
```

Arguments

mSetObj	Input name of the created mSet Object

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

PerformIndNormalization *Perform normalization for individually-uploaded datasets for meta-analysis*

Description

This function performs normalization of individuall-uploaded datasets prior to meta-analysis.

Usage

```
PerformIndNormalization(mSetObj = NA, dataName, norm.opt, auto.opt)
```

Arguments

mSetObj	Input name of the created mSet Object
dataName	Input the name of the individual dataset for normalization.
norm.opt	Performs log2 normalization "log", or no normalization "none".
auto.opt	Performs auto-scaling of data (1), or no (0).

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`PerformIntegCmpdMapping`
Perform compound mapping for integrative analysis methods

Description

Perform compound mapping

Usage

```
PerformIntegCmpdMapping(mSetObj = NA, cmpdIDs, org, idType)
```

Arguments

`mSetObj`	Input name of the created mSet Object
`cmpdIDs`	Input the list of compound IDs
`org`	Input the organism code
`idType`	Input the ID type

Author(s)

Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`PerformIntegGeneMapping`
Perform integrated gene mapping

Description

Used for the pathinteg module

Usage

```
PerformIntegGeneMapping(mSetObj = NA, geneIDs, org, idType)
```

Arguments

`mSetObj`	Input name of the created mSet Object
`geneIDs`	Input the list of gene IDs
`org`	Input the organism code
`idType`	Input the ID type

PerformIntegPathwayAnalysis

Perform integrative pathway analysis

Description

used for integrative analysis as well as general pathways analysis for meta-analysis results

Usage

```
PerformIntegPathwayAnalysis(mSetObj, topo="dc", enrich="hyper", libOpt="integ")
```

Arguments

mSetObj	Input name of the created mSet Object
topo	Select the mode for topology analysis: Degree Centrality ("dc") measures the number of links that connect to a node (representing either a gene or metabolite) within a pathway; Closeness Centrality ("cc") measures the overall distance from a given node to all other nodes in a pathway; Betweenness Centrality ("bc")measures the number of shortest paths from all nodes to all the others that pass through a given node within a pathway.
enrich	Method to perform over-representation analysis (ORA) based on either hypergenometrics analysis ("hyper") or Fisher's exact method ("fisher").
libOpt	Select the different modes of pathways, either the gene-metabolite mode ("integ") which allows for joint-analysis and visualization of both significant genes and metabolites or the gene-centric ("genetic") and metabolite-centric mode ("metab") which allows users to identify enriched pathways driven by significant genes or metabolites, respectively.

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

PerformKOEnrichAnalysis_KO01100

Performs KO enrichment analysis based on the KO01100 map

Description

This function performs KO enrichment analysis based on the KO01100 map and saves the .JSON file

Usage

```
PerformKOEnrichAnalysis_KO01100(mSetObj = NA, category, file.nm)
```

Arguments

`mSetObj`	Input name of the created mSet Object
`category`	Input the option to perform enrichment analysis, "pathway"
`file.nm`	Input name of file to save

Author(s)

Othman Soufan, Jeff Xia <`jeff.xia@mcgill.ca`>, othman.soufan@mcgill.ca McGill University, Canada License: GNU GPL (¿= 2)

`PerformKOEnrichAnalysis_List` *Utility function for PerformKOEnrichAnalysis_KO01100*

Description

Please note: only return hits in map KO01100

Usage

```
PerformKOEnrichAnalysis_List(file.nm)
```

Arguments

`file.nm`	Input the file name

`PerformLimmaDE` *Perform differential expression analysis using Limma for individually-uploaded data.*

Description

This function performs DE analysis of individually-uploaded data prior to meta-analysis.

Usage

```
PerformLimmaDE(mSetObj = NA, dataName, p.lvl = 0.1, fc.lvl = 0)
```

Arguments

`mSetObj`	Input name of the created mSet Object
`dataName`	Input the name of the individual dataset for normalization.
`p.lvl`	Numeric, input the p-value (FDR) cutoff.
`fc.lvl`	Numeric, input the fold-change (FC) cutoff.

Author(s)

Jeff Xia<`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`PerformMapping` *Utility function for PrepareQueryJson geneIDs is text one string, need to make to vector*

Description

Utility function for PrepareQueryJson geneIDs is text one string, need to make to vector

Usage

```
PerformMapping(inputIDs, type)
```

Arguments

`inputIDs`	Input list of IDs
`type`	Input the type of IDs

`performMB` *Timecourse analysis*

Description

Adapted from the timecourse package by Yu Chuan Tai This method is only applicable for time-series, not for general case two/multiple factor analysis

Usage

```
performMB(mSetObj, topPerc)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)
`topPerc`	select the cut-off, default is 10

Author(s)

Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

PerformMetaMerge *Meta-Analysis Method: Direct merging of datasets*

Description

This function is one of three methods to perform meta-analysis. Direct merging of individual data into a mega-dataset results in an analysis of that mega-dataset as if the individual data were derived from the same experiment. This method thereby ignores any inherent bias and heterogeneity between the different data. Because of this, there exists several confounders such as different experimental protocols, technical platforms, and raw data processing procedures that can mask true underlying differences. It is therefore highly suggested that this approach be used only when individual data are very similar (i.e. from the same lab, same platform, without batch effects)."

Usage

```
PerformMetaMerge(mSetObj = NA, BHth = 0.05)
```

Arguments

mSetObj	Input name of the created mSet Object.
BHth	Numeric input to set the significance level. By default it is 0.05.

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

PerformMSDataOutput *Function MS Generation*

Description

Output the MS data. This function will generate .mzML MS data in the working dirctory.

Usage

```
PerformMSDataOutput(raw_data)
```

Arguments

raw_data	MS data in R environment with "MSnExp" class.

Author(s)

Zhiqiang Pang <zhiqiang.pang@mail.mcgill.ca> Jeff Xia <jeff.xia@mcgill.ca> Mcgill University License: GNU GPL (¿= 2)

`PerformMultiMatch` *Perform multiple name matches*

Description

Given a query, performs compound name matching.

Usage

```
PerformMultiMatch(mSetObj = NA, q)
```

Arguments

`mSetObj`	Input name of the created mSet Object.
`q`	Input the query.

`PerformParamsOptimization`
Perform Parameters Optimization

Description

This function is used to optimize the critical parameters of peak picking and alignment for the following data processing. It utilizes the trimed data and the internal instrument-specific parameters. Parallel computing will be performed. The number of cores user want to use could be specified.

Usage

```
PerformParamsOptimization(raw_data, param = p0, method = "DoE", ncore = 4)
```

Arguments

`raw_data`	MSnExp object, can be the (trimmed) data in memory produced by 'PerformDataTrimming' or the orignal data read by ImportRawMSData with 'inMemory" mode.
`param`	List, Parameters defined by 'SetPeakParam' function.
`method`	Character, method of parameters optimization, including "DoE' only. Default is "DoE". Other method is under development.
`ncore`	Numeric, CPU threads number used to perform the parallel based optimization. If thers is memory issue, please reduce the 'ncore' used here. For default, 2/3 CPU threads of total will be used.

Author(s)

Zhiqiang Pang <`zhiqiang.pang@mail.mcgill.ca`> Jeff Xia <`jeff.xia@mcgill.ca`> Mcgill University License: GNU GPL (¿= 2)

`PerformPeakAlignment` *PerformPeakAlignment*

Description

PerformPeakAlignment

Usage

```
PerformPeakAlignment(mSet, param)
```

Arguments

`mSet`	the mSet object generated by PerformPeakPicking function.
`param`	param list generated by updateRawSpectraParam function.

Author(s)

Zhiqiang Pang, Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`PerformPeakAnnotation` *Perform peak annotation*

Description

This function performs peak annotation on the xset object created using the PerformPeakPicking function.

Usage

```
PerformPeakAnnotation(mSet, annotaParam, ncore = 1)
```

Arguments

`xset`	The object created using the PerformPeakPicking function, containing the peak picked MS data.
`annParams`	The object created using the SetAnnotationParam function, containing user's specified or default parameters for downstream raw MS data preprocessing.

Author(s)

Zhiqiang Pang <`zhiqiang.pang@mail.mcgill.ca`>, Jasmine Chong <`jasmine.chong@mail.mcgill.ca`>, Mai Yamamoto <`yamamoto.mai@mail.mcgill.ca`>, and Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

References

Kuhl C, Tautenhahn R, Boettcher C, Larson TR, Neumann S (2012). "CAMERA: an integrated strategy for compound spectra extraction and annotation of liquid chromatography/mass spectrometry data sets." Analytical Chemistry, 84, 283-289. http://pubs.acs.org/doi/abs/10.1021/ac202

`PerformPeakFiling` *PerformPeakFiling*

Description

PerformPeakFiling

Usage

```
PerformPeakFiling(mSet, param, BPPARAM = bpparam())
```

Arguments

`mSet`	the mSet object generated by PerformPeakPicking function.
`param`	param list generated by updateRawSpectraParam function.
`BPPARAM`	parallel method used for data processing. Default is bpparam().

Author(s)

Zhiqiang Pang, Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`PerformPeakGrouping` *PerformPeakGrouping*

Description

PerformPeakGrouping

Usage

```
PerformPeakGrouping(mSet, param)
```

Author(s)

Zhiqiang Pang, Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

PerformPeakPicking *PerformPeakPicking*

Description

PerformPeakPicking

Usage

```
PerformPeakPicking(object, param, BPPARAM = bpparam())
```

Arguments

object	the raw data object read by ImportRawMSData function.
param	param list generated by updateRawSpectraParam function.
BPPARAM	parallel method used for data processing. Default is bpparam().

Author(s)

Zhiqiang Pang, Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

PerformPeakProfiling *Perform peak profiling This function performs feature extraction of user's raw MS data using the rawData object created using the ImportRawMSData function.*

Description

Perform peak profiling This function performs feature extraction of user's raw MS data using the rawData object created using the ImportRawMSData function.

Usage

```
PerformPeakProfiling(rawData, Params, plotSettings, ncore)
```

Arguments

rawData	The object created using the ImportRawMSData function, containing the raw MS data.
Params	The object created using the SetPeakParam function, containing user's specified or default parameters for downstream raw MS data pre-processing.
plotSettings	List, plotting parameters produced by SetPlotParam Function. Defaut is set to true.
ncore	Numeric, used to define the cores' number for Peak Profiling.

Author(s)

Zhiqiang Pang <zhiqiang.pang@mail.mcgill.ca>, Jasmine Chong <jasmine.chong@mail.mcgill.ca>, Mai Yamamoto <yamamoto.mai@mail.mcgill.ca>, and Jeff Xia <jeff.xia@mcgill.ca>
McGill University, Canada License: GNU GPL (¿= 2)

PerformPowerProfiling *Perform power profiling*

Description

Perform power profiling of data

Usage

```
PerformPowerProfiling(mSetObj=NA, fdr.lvl, smplSize)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)
fdr.lvl	Specify the false-discovery rate level.
smplSize	Specify the maximum sample size, the number must be between 60-1000.

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

PerformPSEA *Function to perform peak set enrichment analysis*

Description

This is the main function that performs either the mummichog algorithm, GSEA, or both for peak set enrichment analysis.

Usage

```
PerformPSEA(mSetObj=NA, lib, libVersion, permNum = 100)
```

Arguments

mSetObj	Input the name of the created mSetObj object.
lib	Input the name of the organism library, default is hsa_mfn.
libVersion	Input the version of the KEGG pathway libraries ("current" or "old").
permNum	Numeric, input the number of permutations to perform. Default is 100.

Author(s)

Jasmine Chong, Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`PerformPvalCombination` *Meta-Analysis Method: Combining p-values*

Description

This function is one of three methods to perform meta-analysis. Here, p-values are combined using either the Fisher's method or the Stouffer's method.

Usage

```
PerformPvalCombination(mSetObj = NA, method = "stouffer", BHth = 0.05)
```

Arguments

`mSetObj`	Input name of the created mSet Object.
`method`	Method of p-value combination. By default it is "stouffer", else it is "fisher".
`BHth`	Numeric input to set the significance level. By default it is 0.05.

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`PerformVoteCounting` *Meta-Analysis Method: Vote Counting*

Description

This function is one of three methods to perform meta-analysis. Here, significant features are selected based on a selected criteria (i.e. an adjusted p-value ¡0.05 and the same direction of FC) for each dataset. The votes are then calculated for each feature by counting the total of number of times a feature is significant across all included datasets. However, this method is statistically inefficient and should be considered the last resort in situations where other methods to perform meta-analysis cannot be applied.

Usage

```
PerformVoteCounting(mSetObj = NA, BHth = 0.05, minVote)
```

Arguments

mSetObj	Input name of the created mSet Object.
BHth	Numeric input to set the significance level. By default it is 0.05.
minVote	Numeric input to set the minimum vote-count.

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

---

plot.MS_3D *Function for 3D ms plotting*

Description

Function for 3D ms plotting (internal use only)

Usage

```
plot.MS_3D(object)
```

Author(s)

Zhiqiang Pang <zhiqiang.pang@mail.mcgill.ca>

Plot.Permutation *Plot results of permutation tests*

Description

Plot results of permutation tests

Usage

```
Plot.Permutation(mSetObj=NA, imgName, format="png", dpi=72)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)
imgName	Input a name for the plot
format	elect the image format, "png", of "pdf".
dpi	Input the dpi. If the image format is "pdf", users need not define the dpi. For "png" images, the default dpi is 72. It is suggested that for high-resolution images, select a dpi of 300.

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`Plot.sampletrend` *Sample Trend Scatter*

Description

Scatter sample trend comparison between all sample of different batches

Usage

```
Plot.sampletrend(
  mSetObj,
  imgName,
  format = "png",
  dpi = 300,
  width = NA,
  method
)
```

Arguments

`mSetObj`	Input name of the created mSet Object
`imgName`	Input a name for the plot
`format`	Select the image format, "png", or "pdf".
`dpi`	Input the dpi. If the image format is "pdf", users need not define the dpi. For "png" images, the default dpi is 300. It is suggested that for high-resolution images, select a dpi of 600.
`width`	Input the width, there are 2 default widths, the first, width = NULL, is 10.5. The second default is width = 0, where the width is 7.2. Otherwise users can input their own width.

Author(s)

Zhiqiang Pang <`zhiqiang.pang@mail.mcgill.ca`>, Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`PlotAccuracy` *Plot classification performance using different features for Multi-Biomarker*

Description

Plot of the accuracy of classification with an increasing number of features.

Usage

```
PlotAccuracy(mSetObj=NA, imgName, format="png", dpi=72)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)
imgName	Input a name for the plot
format	Select the image format, "png", of "pdf".
dpi	Input the dpi. If the image format is "pdf", users need not define the dpi. For "png" images, the default dpi is 72. It is suggested that for high-resolution images, select a dpi of 300.

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

PlotANOVA *Plot ANOVA*

Description

Plot ANOVA

Usage

```
PlotANOVA(mSetObj=NA, imgName, format="png", dpi=72, width=NA)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)
imgName	Input a name for the plot
format	Select the image format, "png", or "pdf".
dpi	Input the dpi. If the image format is "pdf", users need not define the dpi. For "png" images, the default dpi is 72. It is suggested that for high-resolution images, select a dpi of 300.
width	Input the width, there are 2 default widths, the first, width = NULL, is 10.5. The second default is width = 0, where the width is 7.2. Otherwise users can input their own width.

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

PlotANOVA2 *Plot Venn diagram of ANOVA results*

Description

Plot Venn diagram of ANOVA results

Usage

```
PlotANOVA2(mSetObj, imgName, format="png", dpi=72, width=NA)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)
imgName	Input a name for the plot
format	Select the image format, "png", or "pdf".
dpi	Input the dpi. If the image format is "pdf", users need not define the dpi. For "png" images, the default dpi is 72. It is suggested that for high-resolution images, select a dpi of 300.
width	Input the width, there are 2 default widths, the first, width = NULL, is 10.5. The second default is width = 0, where the width is 7.2. Otherwise users can input their own width.

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

PlotASCA.Permutation *Plot ASCA permutation*

Description

Plot plsda classification performance using different components

Usage

```
PlotASCA.Permutation(mSetObj=NA, imgName, format="png", dpi=72, width=NA)
```

Arguments

mSetObj	Input name of the created mSet Object
imgName	Input a name for the plot
format	Select the image format, "png", or "pdf".
dpi	Input the dpi. If the image format is "pdf", users need not define the dpi. For "png" images, the default dpi is 72. It is suggested that for high-resolution images, select a dpi of 300.
width	Input the width, there are 2 default widths, the first, width = NULL, is 10.5. The second default is width = 0, where the width is 7.2. Otherwise users can input their own width.

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

PlotAscaImpVar *Plot the important variables for each factor*

Description

Plot the important variables for each factor

Usage

```
PlotAscaImpVar(mSetObj=NA, imgName, format, dpi, width=NA, type)
```

Arguments

mSetObj	Input name of the created mSet Object
imgName	Input a name for the plot
format	Select the image format, "png", or "pdf".
dpi	Input the dpi. If the image format is "pdf", users need not define the dpi. For "png" images, the default dpi is 72. It is suggested that for high-resolution images, select a dpi of 300.
width	Input the width, there are 2 default widths, the first, width = NULL, is 10.5. The second default is width = 0, where the width is 7.2. Otherwise users can input their own width.
type	select model a, b, or ab

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

PlotASCAModel *Plot score plots of each ASCA model for component 1 against time*

Description

Plot score plots of each ASCA model for component 1 against time

Usage

```
PlotASCAModel(mSetObj=NA, imgName, format="png", dpi=72, width=NA, type, colorBW=FALSE)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)
imgName	Input a name for the ASCA score plot
format	Select the image format, "png", or "pdf".
dpi	Input the dpi. If the image format is "pdf", users need not define the dpi. For "png" images, the default dpi is 72. It is suggested that for high-resolution images, select a dpi of 300
width	Input the width, there are 2 default widths, the first, width = NULL, is 10.5. The second default is width = 0, where the width is 7.2. Otherwise users can input their own width.
type	select model a or b
colorBW	Logical, use black/white coloring (T) or not (F)

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

PlotBoxPlot *Plot a boxplot view of a selected compound*

Description

Plots a boxplot of the selected compound's concentration between the groups.

Usage

```
PlotBoxPlot(
  mSetObj,
  feat.nm,
  imgName,
  format = "png",
  dpi = 72,
  isOpt,
  isQuery
)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)
feat.nm	Input the name of the selected compound.
imgName	Input a name for the plot
format	Select the image format, "png", of "pdf".
dpi	Input the dpi. If the image format is "pdf", users need not define the dpi. For "png" images, the default dpi is 72. It is suggested that for high-resolution images, select a dpi of 300.

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

PlotCmpdSummary *Plot compound summary change to use dataSet$proc instead of dataSet$orig in case of too many NAs*

Description

Plot compound summary change to use dataSet$proc instead of dataSet$orig in case of too many NAs

Usage

```
PlotCmpdSummary(mSetObj = NA, cmpdNm, format = "png", dpi = 72, width = NA)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)
cmpdNm	Input the name of the compound to plot
format	Input the format of the image to create
dpi	Input the dpi of the image to create
width	Input the width of the image to create

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`PlotCmpdView` *Plot Compound View*

Description

Plots a bar-graph of selected compound over groups

Usage

```
PlotCmpdView(mSetObj=NA, cmpdNm, format="png", dpi=72, width=NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)
`cmpdNm`	Input a name for the compound
`format`	Select the image format, "png", or "pdf".
`dpi`	Input the dpi. If the image format is "pdf", users need not define the dpi. For "png" images, the default dpi is 72. It is suggested that for high-resolution images, select a dpi of 300.
`width`	Input the width, there are 2 default widths, the first, width = NULL, is 10.5. The second default is width = 0, where the width is 7.2. Otherwise users can input their own width.

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`PlotConcRange` *Plot the compound concentration data compared to the reference concentration range*

Description

Plot the compound concentration data compared to the reference concentration range

Usage

```
PlotConcRange(mSetObj, nm, format="png", dpi=72, width=NA)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)
nm	of the input compound
format	Select the image format, "png", or "pdf".
dpi	Input the dpi. If the image format is "pdf", users need not define the dpi. For "png" images, the default dpi is 72. It is suggested that for high-resolution images, select a dpi of 300.
width	Input the width, there are 2 default widths, the first, width = NULL, is 10.5. The second default is width = 0, where the width is 7.2. Otherwise users can input their own width.

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

PlotCorr *Pattern hunter, correlation plot*

Description

Plot correlation

Usage

```
PlotCorr(mSetObj = NA, imgName, format = "png", dpi = 72, width = NA)
```

Arguments

mSetObj	Input name of the created mSet Object
imgName	Input a name for the plot
format	Select the image format, "png", or "pdf".
dpi	Input the dpi. If the image format is "pdf", users need not define the dpi. For "png" images, the default dpi is 72. It is suggested that for high-resolution images, select a dpi of 300.
width	Input the width, there are 2 default widths, the first, width = NULL, is 10.5. The second default is width = 0, where the width is 7.2. Otherwise users can input their own width.

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`PlotCorrHeatMap` *Pattern hunter, corr heatmap*

Description

Plot correlation heatmap

Usage

```
PlotCorrHeatMap(
  mSetObj = NA,
  imgName,
  format = "png",
  dpi = 72,
  width = NA,
  target,
  cor.method,
  colors,
  viewOpt,
  fix.col,
  no.clst,
  top,
  topNum
)
```

Arguments

`mSetObj`	Input name of the created mSet Object.
`imgName`	Input the name of the image to create
`format`	Select the image format, "png", or "pdf".
`dpi`	Input the dpi. If the image format is "pdf", users need not define the dpi. For "png" images, the default dpi is 72. It is suggested that for high-resolution images, select a dpi of 300.
`width`	Input the width, there are 2 default widths, the first, width = NULL, is 10.5. The second default is width = 0, where the width is 7.2. Otherwise users can input their own width.
`target`	Input "row" to select features, or "col" to select samples.
`cor.method`	Indicate the correlation method, 'pearson', 'spearman', or 'kendall'.
`colors`	Indicate the colors for the heatmap, "bwm" for default, "gbr" for red/green, "heat" for heat colors, "topo" for topo colors, and "gray" for gray scale.
`viewOpt`	Indicate "overview" to get an overview of the heatmap, and "detail" to get a detailed view of the heatmap.
`fix.col`	Logical, fix colors (TRUE) or not (FALSE).

no.clst	Logical, indicate if the correlations should be clustered (TRUE) or not (FALSE).
top	View top
topNum	Numeric, view top McGill University, Canada License: GNU GPL (¿= 2)

Author(s)

Jeff Xia<jeff.xia@mcgill.ca>

PlotDetailROC *Plot detailed ROC*

Description

Plot detailed ROC

Usage

```
PlotDetailROC(mSetObj = NA, imgName, thresh, sp, se, dpi = 72, format = "png")
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)
imgName	Input a name for the plot
thresh	Input the threshold
sp	Specificity
se	Sensitivity
dpi	Input the dpi. If the image format is "pdf", users need not define the dpi. For "png" images, the default dpi is 72. It is suggested that for high-resolution images, select a dpi of 300.
format	Select the image format, "png", or "pdf".

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`PlotEBAM.Cmpd` *Plot EBAM*

Description

Plot EBAM

Usage

```
PlotEBAM.Cmpd(mSetObj=NA, imgName, format, dpi, width)
```

Arguments

`mSetObj`	Input name of the created mSet Object
`imgName`	Input a name for the plot
`format`	Select the image format, "png", or "pdf".
`dpi`	Input the dpi. If the image format is "pdf", users need not define the dpi. For "png" images, the default dpi is 72. It is suggested that for high-resolution images, select a dpi of 300.
`width`	Input the width, there are 2 default widths, the first, width = NULL, is 10.5. The second default is width = 0, where the width is 7.2. Otherwise users can input their own width.

Author(s)

Jeff Xia<`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`PlotEIC` *Plot EIC*

Description

This functionn creates an extracted ion chromatogram (EIC) for a specific m/z and retention time. This is used for quality-control of raw m/s data.

Usage

```
PlotEIC(
  raw_data,
  rt_mn,
  rt_mx,
  mz_mn,
  mz_mx,
  aggreg = "sum",
  format = "png",
  dpi = 72,
  width = 9
)
```

Arguments

raw_data	The object created using the ImportRawMSData function, containing the raw MS data.
rt_mn	Numeric, specify the minimum bound of the retention time range.
rt_mx	Numeric, specify the maximum bound of the retention time range.
mz_mn	Numeric, specify the minimum bound of the m/z range.
mz_mx	Numeric, specify the maximum bound of the m/z range.
aggreg	Character, if "sum", creates a total ion chromatogram. If "max", creates a base peak chromatogram. By default it is set to "sum".
format	Character, input the format of the image to create.
dpi	Numeric, input the dpi of the image to create.
width	Numeric, input the width of the image to create.

PlotEnrichNet.Overview

Barplot height is enrichment fold change

Description

Used in higher functions, the color is based on p values

Usage

```
PlotEnrichNet.Overview(folds, pvals, layoutOpt = layout.fruchterman.reingold)
```

Arguments

folds	Input fold-change for bar plot
pvals	Input p-values for bar plot
layoutOpt	Input the layout option, default is set to layout.fruchterman.reingold

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`PlotFC` *Plot fold change*

Description

Plot fold change analysis

Usage

```
PlotFC(mSetObj=NA, imgName, format="png", dpi=72, width=NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)
`imgName`	Input a name for the plot
`format`	Select the image format, "png", or "pdf".
`dpi`	Input the dpi. If the image format is "pdf", users need not define the dpi. For "png" images, the default dpi is 72. It is suggested that for high-resolution images, select a dpi of 300.
`width`	Input the width, there are 2 default widths, the first, width = NULL, is 10.5. The second default is width = 0, where the width is 7.2. Otherwise users can input their own width.

Author(s)

Jeff Xia<`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`PlotHCTree` *Plot Dendrogram*

Description

Dendogram

Usage

```
PlotHCTree(
  mSetObj = NA,
  imgName,
  format = "png",
  dpi = 72,
  width = NA,
  smplDist,
  clstDist
)
```

Arguments

mSetObj	Input name of the created mSet Object
imgName	Input a name for the plot
format	Select the image format, "png", or "pdf".
dpi	Input the dpi. If the image format is "pdf", users need not define the dpi. For "png" images, the default dpi is 72. It is suggested that for high-resolution images, select a dpi of 300.
width	Input the width, there are 2 default widths, the first, width = NULL, is 10.5. The second default is width = 0, where the width is 7.2. Otherwise users can input their own width.
smplDist	Method to calculate sample distance
clstDist	Method to calculate clustering distance

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

PlotHeatMap *Create Heat Map Plot*

Description

Plot a heatmap based on results from t-tests/ANOVA, VIP or randomforest

Usage

```
PlotHeatMap(
  mSetObj = NA,
  imgName,
  format = "png",
  dpi = 72,
  width = NA,
  dataOpt,
  scaleOpt,
  smplDist,
  clstDist,
  palette,
  viewOpt = "detail",
  rowV = T,
  colV = T,
  var.inx = NA,
  border = T,
  grp.ave = F
)
```

Arguments

`mSetObj`	Input name of the created mSet Object
`imgName`	Input a name for the plot
`format`	Select the image format, "png", or "pdf".
`dpi`	Input the dpi. If the image format is "pdf", users need not define the dpi. For "png" images, the default dpi is 72. It is suggested that for high-resolution images, select a dpi of 300.
`width`	Input the width, there are 2 default widths, the first, width = NULL, is 10.5. The second default is width = 0, where the width is 7.2. Otherwise users can input their own width.
`dataOpt`	Set data options
`scaleOpt`	Set the image scale
`smplDist`	Input the sample distance method
`clstDist`	Input the clustering distance method
`palette`	Input color palette choice
`viewOpt`	Set heatmap options, default is set to "detail"
`rowV`	Default is set to T
`colV`	Default is set to T
`var.inx`	Default is set to NA
`border`	Indicate whether or not to show cell-borders, default is set to T
`grp.ave`	Logical, default is set to F

Author(s)

Jeff Xia<`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`PlotHeatMap2` *Plot heatmap visualization for time-series data*

Description

Plot heatmap visualization for time-series data

Usage

```
PlotHeatMap2(mSetObj=NA, imgName, format="png", dpi=72, width=NA, smplDist="pearson", clstDist="avera
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)
imgName	Input a name for the plot
format	Select the image format, "png", or "pdf".
dpi	Input the dpi. If the image format is "pdf", users need not define the dpi. For "png" images, the default dpi is 72. It is suggested that for high-resolution images, select a dpi of 300.
width	Input the width, there are 2 default widths, the first, width = NULL, is 10.5. The second default is width = 0, where the width is 7.2. Otherwise users can input their own width.
smplDist	Select distance measure: euclidean, pearson, or minkowski
clstDist	Select clustering algorithm: ward, average, complete, single
colors	Select heatmap colors: bwm, gray
viewOpt	Select overview or detailed view: overview or detail
hiRes	Select high-resolution or not: logical, default set to F
sortInx	Sort by index
useSigFeature	Use significant features only: F or T (default false)
drawBorder	Show cell borders: F or T (default F)

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

PlotImpVar *Plot PLS important variables,*

Description

Plot PLS important variables, BHan: added bgcolor parameter for B/W color

Usage

```
PlotImpVar(mSetObj = NA, imp.vec, xlbl, feat.num = 15, color.BW = FALSE)
```

Arguments

mSetObj	Input name of the created mSet Object
imp.vec	Input the vector of important variables
xlbl	Input the x-label
feat.num	Numeric, set the feature numbers, default is set to 15
color.BW	Use black-white for plot (T) or colors (F)

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`PlotImpVars` *Plot selected compounds by their percentage frequency*

Description

Plot the important variables of single biomarker model ranked by order of importance

Usage

```
PlotImpVars(mSetObj=NA, imgName, format="png", dpi=72, mdl.inx, measure = "freq", feat.num = 15)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)
`imgName`	Input a name for the plot
`format`	elect the image format, "png", of "pdf".
`dpi`	Input the dpi. If the image format is "pdf", users need not define the dpi. For "png" images, the default dpi is 72. It is suggested that for high-resolution images, select a dpi of 300.
`mdl.inx`	Model index, -1 selects the model with the best AUC, input 1-6 to view the important features of one of the top six models
`measure`	Choose to rank features by the frequency of being selected "freq", or the mean importance measure "mean"
`feat.num`	Input the number of features to include in the plot, by default it is 15.

Author(s)

Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`PlotInmexGraph` *Plot an igraph object and return the node information (position and labels)*

Description

Plot an igraph object and return the node information (position and labels) Used in a higher function

Usage

```
PlotInmexGraph(
  mSetObj,
  pathName,
  g,
  width = NA,
  height = NA,
  bg.color = NULL,
  line.color = NULL,
  format = "png",
  dpi = NULL
)
```

Arguments

mSetObj	Input name of the created mSet Object
g	Input the graph
width	Input the width, there are 2 default widths, the first, width = NULL, is 10.5. The second default is width = 0, where the width is 7.2. Otherwise users can input their own width.
height	Input the height of the graph to create
bg.color	Set the background color, default is set to NULL
line.color	Set the line color, default is set to NULL
path.id	Input the pathway id

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

PlotInmexPath *Plot integrated methods pathway analysis*

Description

Only update the background info for matched node

Usage

```
PlotInmexPath(mSetObj=NA, path.id, width, height)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)
`width`	Input the width, there are 2 default widths, the first, width = NULL, is 10.5. The second default is width = 0, where the width is 7.2. Otherwise users can input their own width.
`height`	Input the height of the image to create.
`path.id`	Input the ID of the pathway to plot.

Author(s)

Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`PlotIntegPaths` *PlotIntegPaths*

Description

Plots both the original mummichog and the GSEA results by combining p-values using the Fisher's method (sumlog).

Usage

```
PlotIntegPaths(
  mSetObj = NA,
  imgName,
  format = "png",
  dpi = 72,
  width = 9,
  labels = "default",
  labels.x = 5,
  labels.y = 5,
  scale.axis = TRUE
)
```

Arguments

`mSetObj`	Input the name of the created mSetObj object
`imgName`	Input a name for the plot
`format`	Character, input the format of the image to create.
`dpi`	Numeric, input the dpi of the image to create.
`width`	Numeric, input the width of the image to create.
`labels.x`	Numeric, indicate the number of top-ranked pathways using the fGSEA algorithm to annotate on the plot.
`labels.y`	Numeric, indicate the number of top-ranked pathways using the original mummichog algorithm to annotate on the plot.

Labels	Character, indicate if the plot should be labeled. By default it is set to "default", and the 5 top-ranked pathways per each algorithm will be plotted. Users can adjust the number of pathways to be annotated per pathway using the "labels.x" and "labels.y" parameters. Users can set this to "none" for no annotations, or "all" to annotate all pathways.

Author(s)

Jasmine Chong, Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

PlotInteraction *Plot ASCA interaction plots*

Description

Plot ASCA interaction plots

Usage

```
PlotInteraction(mSetObj=NA, imgName, format="png", dpi=72, colorBW=FALSE, width=NA)
```

Arguments

mSetObj	Input name of the created mSet Object
imgName	Input a name for the plot
format	Select the image format, "png", or "pdf".
dpi	Input the dpi. If the image format is "pdf", users need not define the dpi. For "png" images, the default dpi is 72. It is suggested that for high-resolution images, select a dpi of 300.
colorBW	Logical, use black and white (TRUE) or colors (FALSE)
width	Input the width, there are 2 default widths, the first, width = NULL, is 10.5. The second default is width = 0, where the width is 7.2. Otherwise users can input their own width.

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`PlotKEGGPath` *Plot metabolome pathway*

Description

Orthogonal PLS-DA (from ropls)

Usage

```
PlotKEGGPath(
  mSetObj = NA,
  pathName,
  width = NA,
  height = NA,
  format = "png",
  dpi = NULL
)
```

Arguments

`mSetObj`	Input name of the created mSet Object
`pathName`	Input the name of the selected pathway
`width`	Input the width, there are 2 default widths, the first, width = NULL, is 10.5. The second default is width = 0, where the width is 7.2. Otherwise users can input their own width.
`height`	Input the height of the created plot.

Author(s)

Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`PlotKmeans` *Plot K-means analysis*

Description

Plot K-means analysis

Usage

```
PlotKmeans(mSetObj = NA, imgName, format = "png", dpi = 72, width = NA)
```

Arguments

mSetObj	Input name of the created mSet Object
imgName	Input a name for the plot
format	Select the image format, "png", or "pdf".
dpi	Input the dpi. If the image format is "pdf", users need not define the dpi. For "png" images, the default dpi is 72. It is suggested that for high-resolution images, select a dpi of 300.
width	Input the width, there are 2 default widths, the first, width = NULL, is 10.5. The second default is width = 0, where the width is 7.2. Otherwise users can input their own width.

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

PlotLoadingCmpd *Plot loading compounds*

Description

Plot loading compounds

Usage

```
PlotLoadingCmpd(mSetObj = NA, cmpdNm, format = "png", dpi = 72, width = NA)
```

Arguments

mSetObj	Input name of the created mSet Object
cmpdNm	Input the name of the selected compound
format	Select the image format, "png", or "pdf".
dpi	Input the dpi. If the image format is "pdf", users need not define the dpi. For "png" images, the default dpi is 72. It is suggested that for high-resolution images, select a dpi of 300.
width	Input the width, there are 2 default widths, the first, width = NULL, is 10.5. The second default is width = 0, where the width is 7.2. Otherwise users can input their own width.

PlotMBTimeProfile *Plot MB Time Profile*

Description

Plot MB Time Profile

Usage

```
PlotMBTimeProfile(mSetObj = NA, cmpdNm, format = "png", dpi = 72, width = NA)
```

Arguments

mSetObj	Input name of the created mSet Object
cmpdNm	Input the name of the compound
format	Select the image format, "png", or "pdf".
dpi	Input the dpi. If the image format is "pdf", users need not define the dpi. For "png" images, the default dpi is 72. It is suggested that for high-resolution images, select a dpi of 300.
width	Input the width, there are 2 default widths, the first, width = NULL, is 10.5. The second default is width = 0, where the width is 7.2. Otherwise users can input their own width.

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

PlotMetaVenn *Meta-Analysis: Plot Venn Diagram*

Description

This function plots a venn diagram of the individual studies.

Usage

```
PlotMetaVenn(mSetObj = NA, imgNM = NA)
```

Arguments

mSetObj	Input name of the created mSet Object.
imgNM	Input the name of the created Venn Diagram

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

PlotMetpaPath *Plot KEGG pathway*

Description

Plot KEGG pathway

Usage

```
PlotMetpaPath(
  mSetObj = NA,
  pathName,
  width = NA,
  height = NA,
  format = "png",
  dpi = NULL
)
```

Arguments

mSetObj	Input name of the created mSet Object
pathName	Input the name of the selected KEGG pathway
width	Input the width, there are 2 default widths, the first, width = NULL, is 10.5. The second default is width = 0, where the width is 7.2. Otherwise users can input their own width.
format	Select the image format, "png", or "pdf".
dpi	Input the dpi. If the image format is "pdf", users need not define the dpi. For "png" images, the default dpi is 72. It is suggested that for high-resolution images, select a dpi of 300.

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

PlotModelScree *Plot scree plots for each model in ASCA*

Description

Plot scree plots for each model in ASCA

Usage

```
PlotModelScree(mSetObj, imgName, format="png", dpi=72, width=NA)
```

Arguments

`mSetObj`	Input name of the created mSet Object.
`imgName`	Input a name for the plot
`format`	Select the image format, "png", or "pdf".
`dpi`	Input the dpi. If the image format is "pdf", users need not define the dpi. For "png" images, the default dpi is 72. It is suggested that for high-resolution images, select a dpi of 300.
`width`	Input the width, there are 2 default widths, the first, width = NULL, is 10.5. The second default is width = 0, where the width is 7.2. Otherwise users can input their own width.

Author(s)

Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`PlotMS.RT` *Plot rentention time corrected spectra*

Description

Plot the retention time corrected spectra

Usage

```
PlotMS.RT(mSetObj = NA, imgName, format = "png", dpi = 72, width = NA)
```

Arguments

`mSetObj`	Input name of the created mSet Object
`imgName`	Input the name for the created plot
`format`	Select the image format, "png", or "pdf".
`dpi`	Input the dpi. If the image format is "pdf", users need not define the dpi. For "png" images, the default dpi is 72. It is suggested that for high-resolution images, select a dpi of 300.
`width`	Input the width, there are 2 default widths, the first, width = NULL, is 10.5. The second default is width = 0, where the width is 7.2. Otherwise users can input their own width.

PlotMS2Spectra *Plot selected M2 spectra for an m/z feature*

Description

This function creates a plot of the user selected precursor m/z.

Usage

```
PlotMS2Spectra(ms2, spectra = 1)
```

Arguments

ms2	Spectrum2 class object containing all of the spectra for the selected m/z feature.

Author(s)

Jasmine Chong <jasmine.chong@mail.mcgill.ca>, Mai Yamamoto <yamamoto.mai@mail.mcgill.ca>, and Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

PlotMSEA.Overview *Plot MSEA overview*

Description

Barplot height is enrichment fold change color is based on p values, used in higher functions

Usage

```
PlotMSEA.Overview(folds, pvals)
```

Arguments

folds	Input the fold-change values
pvals	Input the p-values

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`PlotMSPeaksPerm` *Plot MS Peaks to Paths mummichog permutation p-values*

Description

Plots the mummichog permutation p-values

Usage

```
PlotMSPeaksPerm(
  mSetObj = NA,
  pathway,
  imgName,
  format = "png",
  dpi = 72,
  width = NA
)
```

Arguments

`mSetObj`	Input name of the created mSet Object
`pathway`	Input the name of the pathway
`imgName`	Input a name for the plot
`format`	Select the image format, "png", or "pdf".
`dpi`	Input the dpi. If the image format is "pdf", users need not define the dpi. For "png" images, the default dpi is 72. It is suggested that for high-resolution images, select a dpi of 300.
`width`	Input the width, there are 2 default widths, the first, width = NULL, is 10.5. The second default is width = 0, where the width is 7.2. Otherwise users can input their own width.

Author(s)

Jeff Xia<`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`PlotNormSummary` *Two plot summary plot: Feature View of before and after normalization*

Description

For each plot, the top is a box plot, bottom is a density plot

Usage

```
PlotNormSummary(mSetObj, imgName, format, dpi, width)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)
imgName	Input a name for the plot
format	Select the image format, "png", or "pdf".
dpi	Input the dpi. If the image format is "pdf", users need not define the dpi. For "png" images, the default dpi is 72. It is suggested that for high-resolution images, select a dpi of 300.
width	Input the width, there are 2 default widths, the first, width = NULL, is 10.5. The second default is width = 0, where the width is 7.2. Otherwise users can input their own width.

Author(s)

Jeff Xia <jeff.xia@mcgill.ca>, Jasmine Chong McGill University, Canada

---

PlotOPLS.MDL *Plot OPLS*

Description

Plot OPLS

Usage

```
PlotOPLS.MDL(mSetObj = NA, imgName, format = "png", dpi = 72, width = NA)
```

Arguments

mSetObj	Input name of the created mSet Object
imgName	Input a name for the plot
format	Select the image format, "png", or "pdf".
dpi	Input the dpi. If the image format is "pdf", users need not define the dpi. For "png" images, the default dpi is 72. It is suggested that for high-resolution images, select a dpi of 300.
width	Input the width, there are 2 default widths, the first, width = NULL, is 10.5. The second default is width = 0, where the width is 7.2. Otherwise users can input their own width.

`PlotOPLS.Permutation` *Plot OPLS-DA permutation*

Description

Orthogonal PLS-DA (from ropls) perform permutation, using training classification accuracy as indicator, for two or multi-groups

Usage

```
PlotOPLS.Permutation(
  mSetObj = NA,
  imgName,
  format = "png",
  dpi = 72,
  width = NA
)
```

Arguments

`mSetObj`	Input name of the created mSet Object
`imgName`	Input a name for the plot
`format`	Select the image format, "png", or "pdf".
`dpi`	Input the dpi. If the image format is "pdf", users need not define the dpi. For "png" images, the default dpi is 72. It is suggested that for high-resolution images, select a dpi of 300.
`width`	Input the width, there are 2 default widths, the first, width = NULL, is 10.5. The second default is width = 0, where the width is 7.2. Otherwise users can input their own width.

Author(s)

Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`PlotOPLS.Splot` *S-plot for OPLS-DA*

Description

Orthogonal PLS-DA (from ropls) S-plot for important features from OPLS-DA

Usage

```
PlotOPLS.Splot(
  mSetObj = NA,
  imgName,
  plotType = "all",
  format = "png",
  dpi = 72,
  width = NA
)
```

Arguments

mSetObj	Input name of the created mSet Object
imgName	Input a name for the plot
format	Select the image format, "png", or "pdf".
dpi	Input the dpi. If the image format is "pdf", users need not define the dpi. For "png" images, the default dpi is 72. It is suggested that for high-resolution images, select a dpi of 300.
width	Input the width, there are 2 default widths, the first, width = NULL, is 10.5. The second default is width = 0, where the width is 7.2. Otherwise users can input their own width.

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

PlotOPLS2DScore *Create OPLS-DA score plot*

Description

Orthogonal PLS-DA (from ropls) score plot

Usage

```
PlotOPLS2DScore(
  mSetObj = NA,
  imgName,
  format = "png",
  dpi = 72,
  width = NA,
  inx1,
  inx2,
  reg = 0.95,
  show = 1,
  grey.scale = 0
)
```

Arguments

`mSetObj`	Input name of the created mSet Object
`imgName`	Input a name for the plot
`format`	Select the image format, "png", or "pdf".
`dpi`	Input the dpi. If the image format is "pdf", users need not define the dpi. For "png" images, the default dpi is 72. It is suggested that for high-resolution images, select a dpi of 300.
`width`	Input the width, there are 2 default widths, the first, width = NULL, is 10.5. The second default is width = 0, where the width is 7.2. Otherwise users can input their own width.
`inx1`	Numeric, indicate the number of the principal component for the x-axis of the loading plot.
`inx2`	Numeric, indicate the number of the principal component for the y-axis of the loading plot.
`reg`	Numeric
`show`	Show variable labels, 1 or O
`grey.scale`	Numeric, indicate grey-scale, 0 for no, and 1 for yes

Author(s)

Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`PlotORA` *Plot over-representation analysis (ORA)*

Description

Plot over-representation analysis (ORA)

Usage

```
PlotORA(mSetObj=NA, imgName, imgOpt, format="png", dpi=72, width=NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)
`imgName`	Input a name for the plot
`imgOpt`	"net"
`format`	Select the image format, "png", or "pdf".
`dpi`	Input the dpi. If the image format is "pdf", users need not define the dpi. For "png" images, the default dpi is 72. It is suggested that for high-resolution images, select a dpi of 300.
`width`	Input the width, there are 2 default widths, the first, width = NULL, is 10.5. The second default is width = 0, where the width is 7.2. Otherwise users can input their own width.

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`PlotPathSummary` *Plot a scatterplot (circle) overview of the matched pathways*

Description

x axis is the pathway impact factor y axis is the p value (from ORA or globaltest) return the circle information

Usage

```
PlotPathSummary(
  mSetObj = NA,
  imgName,
  format = "png",
  dpi = 72,
  width = NA,
  x,
  y
)
```

Arguments

`mSetObj`	Input name of the created mSet Object
`imgName`	Input a name for the plot
`format`	Select the image format, "png", or "pdf".
`dpi`	Input the dpi. If the image format is "pdf", users need not define the dpi. For "png" images, the default dpi is 72. It is suggested that for high-resolution images, select a dpi of 300.
`width`	Input the width, there are 2 default widths, the first, width = NULL, is 10.5.
`x`	Input the X
`y`	Input the Y

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`PlotPathwayMZHits` *Plot m/z hits in a pathway*

Description

Function to create a boxplot of m/z features within a specific pathway. m/z features used by the original mummichog algorithm are highlighted with an asterisk.

Usage

```
PlotPathwayMZHits(mSetObj = NA, msetNM, format = "png", dpi = 300, width = 10)
```

Arguments

`mSetObj`	Input the name of the created mSetObj object.
`msetNM`	Character, input the name of the pathway.
`format`	Character, input the format of the image to create.
`dpi`	Numeric, input the dpi of the image to create. Default is set to 300.
`width`	Numeric, input the width of the image to create. Default is set to 10.

Author(s)

Jasmine Chong, Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`PlotPCA.overview` *Scatter plot colored by different batches*

Description

Scatter plot colored by different batches

Usage

```
PlotPCA.overview(
  mSetObj,
  imgName,
  format = "png",
  dpi = 300,
  width = NA,
  method
)
```

Arguments

mSetObj	Input name of the created mSet Object
imgName	Input a name for the plot
format	Select the image format, "png", or "pdf".
dpi	Input the dpi. If the image format is "pdf", users need not define the dpi. For "png" images, the default dpi is 300. It is suggested that for high-resolution images, select a dpi of 600.
width	Input the width, there are 2 default widths, the first, width = NULL, is 10.5. The second default is width = 0, where the width is 7.2. Otherwise users can input their own width.

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

PlotPCA2DScore *Create 2D PCA score plot*

Description

Rotate PCA analysis

Usage

```
PlotPCA2DScore(mSetObj=NA, imgName, format="png", dpi=72, width=NA, pcx, pcy, reg = 0.95, show=1, grey.
```

Arguments

mSetObj	Input name of the created mSet Object
imgName	Input a name for the plot
format	Select the image format, "png", or "pdf".
dpi	Input the dpi. If the image format is "pdf", users need not define the dpi. For "png" images, the default dpi is 72. It is suggested that for high-resolution images, select a dpi of 300.
width	Input the width, there are 2 default widths, the first, width = NULL, is 10.5. The second default is width = 0, where the width is 7.2. Otherwise users can input their own width.
style	Numeric, the ratio style of the figure (width/height), defalt is 1, 1:1. 2 means 4:3, while 3 means 16:9.
pcx	Specify the principal component on the x-axis
pcy	Specify the principal component on the y-axis
reg	Numeric, input a number between 0 and 1, 0.95 will display the 95 percent confidence regions, and 0 will not.
show	Display sample names, 1 = show names, 0 = do not show names.
grey.scale	Use grey-scale colors, 1 = grey-scale, 0 = not grey-scale.

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

PlotPCA3DScore *Create 3D PCA score plot*

Description

Rotate PCA analysis

Usage

```
PlotPCA3DScore(mSetObj=NA, imgName, format="json", inx1, inx2, inx3)
```

Arguments

mSetObj	Input name of the created mSet Object
imgName	Input a name for the plot
format	Select the image format, "png", or "pdf".
inx1	Numeric, indicate the number of the principal component for the x-axis of the loading plot.
inx2	Numeric, indicate the number of the principal component for the y-axis of the loading plot.
inx3	Numeric, indicate the number of the principal component for the z-axis of the loading plot.

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

PlotPCA3DScoreImg *Create 3D PCA score plot*

Description

This function creates both a static 3D PCA score plot as well as an interactive 3D PCA score plot using the plotly R package. The 3D PCA score plot is stored in the mSetObj (mSetObj$imgSet$pca.3d). To view the plot, if your mSetObj is named mSet, type "mSet$imgSet$pca.3d" inro your R console, and the 3D plot will appear.

Usage

```
PlotPCA3DScoreImg(mSetObj=NA, imgName, format="png", dpi=72, width=NA, inx1, inx2, inx3, angl)

mSet <- PlotPCA3DScore(mSetObj=NA, imgName, format="json", dpi=72, width=NA, inx1, inx2, inx3, angl)
```

Arguments

mSetObj	Input name of the created mSet Object.
imgName	Input a name for the plot.
format	Select the image format, "png", or "pdf".
dpi	Input the dpi. If the image format is "pdf", users need not define the dpi. For "png" images, the default dpi is 72. It is suggested that for high-resolution images, select a dpi of 300.
width	Input the width, there are 2 default widths, the first, width = NULL, is 10.5. The second default is width = 0, where the width is 7.2. Otherwise users can input their own width.
inx1	Numeric, indicate the number of the principal component for the x-axis of the loading plot.
inx2	Numeric, indicate the number of the principal component for the y-axis of the loading plot.
inx3	Numeric, indicate the number of the principal component for the z-axis of the loading plot.
angl	Input the angle

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

PlotPCABiplot *Create PCA Biplot, set xpd = T to plot outside margin*

Description

Rotate PCA analysis

Usage

```
PlotPCABiplot(mSetObj=NA, imgName, format="png", dpi=72, width=NA, inx1, inx2)
```

Arguments

mSetObj	Input name of the created mSet Object
imgName	Input a name for the plot
format	Select the image format, "png", or "pdf".
dpi	Input the dpi. If the image format is "pdf", users need not define the dpi. For "png" images, the default dpi is 72. It is suggested that for high-resolution images, select a dpi of 300.
width	Input the width, there are 2 default widths, the first, width = NULL, is 10.5. The second default is width = 0, where the width is 7.2. Otherwise users can input their own width.

inx1	Numeric, indicate the number of the principal component for the x-axis of the loading plot.
inx2	Numeric, indicate the number of the principal component for the y-axis of the loading plot.

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

PlotPCALoading *Plot PCA loadings and also set up the matrix for display*

Description

Rotate PCA analysis

Usage

```
PlotPCALoading(mSetObj=NA, imgName, format="png", dpi=72, width=NA, inx1, inx2, plotType, lbl.feat=1)
```

Arguments

mSetObj	Input name of the created mSet Object
imgName	Input a name for the plot
format	Select the image format, "png", or "pdf".
dpi	Input the dpi. If the image format is "pdf", users need not define the dpi. For "png" images, the default dpi is 72. It is suggested that for high-resolution images, select a dpi of 300.
width	Input the width, there are 2 default widths, the first, width = NULL, is 10.5. The second default is width = 0, where the width is 7.2. Otherwise users can input their own width.
inx1	Numeric, indicate the number of the principal component for the x-axis of the loading plot.
inx2	Numeric, indicate the number of the principal component for the y-axis of the loading plot.

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

PlotPCAPairSummary *Plot PCA pair summary, format image in png, tiff, pdf, ps, svg*

Description

Rotate PCA analysis

Usage

```
PlotPCAPairSummary(mSetObj=NA, imgName, format="png", dpi=72, width=NA, pc.num)
```

Arguments

mSetObj	Input name of the created mSet Object
imgName	Input a name for the plot
format	Select the image format, "png", or "pdf".
dpi	Input the dpi. If the image format is "pdf", users need not define the dpi. For "png" images, the default dpi is 72. It is suggested that for high-resolution images, select a dpi of 300.
width	Input the width, there are 2 default widths, the first, width = NULL, is 10.5. The second default is width = 0, where the width is 7.2. Otherwise users can input their own width.
pc.num	Numeric, input a number to indicate the number of principal components to display in the pairwise score plot.

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

PlotPCAScree *Plot PCA scree plot*

Description

Rotate PCA analysis

Usage

```
PlotPCAScree(mSetObj=NA, imgName, format="png", dpi=72, width=NA, scree.num)
```

Arguments

mSetObj	Input name of the created mSet Object
imgName	Input a name for the plot
format	Select the image format, "png", or "pdf".
dpi	Input the dpi. If the image format is "pdf", users need not define the dpi. For "png" images, the default dpi is 72. It is suggested that for high-resolution images, select a dpi of 300.
width	Input the width, there are 2 default widths, the first, width = NULL, is 10.5. The second default is width = 0, where the width is 7.2. Otherwise users can input their own width.
scree.num	Numeric, input a number to indicate the number of principal components to display in the scree plot.

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

PlotPeaks2Paths *PlotPeaks2Paths*

Description

Plots either the original mummichog or GSEA results.

Usage

```
PlotPeaks2Paths(
  mSetObj = NA,
  imgName,
  format = "png",
  dpi = 72,
  width = 9,
  labels = "default",
  num_annot = 5
)
```

Arguments

mSetObj	Input the name of the created mSetObj object
imgName	Input a name for the plot
format	Character, input the format of the image to create.
dpi	Numeric, input the dpi of the image to create.
width	Numeric, input the width of the image to create.
Labels	Character, indicate if the plot should be labeled. By default it is set to "default", and the 5 top-ranked pathways per each algorithm will be plotted. Users can adjust the number of pathways to be annotated per pathway using the "num_annot" parameter.

Author(s)

Jasmine Chong, Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`PlotPLS.Classification` *Plot PLS-DA classification performance using different components*

Description

Plot plsda classification performance using different components

Usage

```
PlotPLS.Classification(
  mSetObj = NA,
  imgName,
  format = "png",
  dpi = 72,
  width = NA
)
```

Arguments

`mSetObj`	Input name of the created mSet Object
`imgName`	Input a name for the plot
`format`	Select the image format, "png", or "pdf".
`dpi`	Input the dpi. If the image format is "pdf", users need not define the dpi. For "png" images, the default dpi is 72. It is suggested that for high-resolution images, select a dpi of 300.
`width`	Input the width, there are 2 default widths, the first, width = NULL, is 10.5. The second default is width = 0, where the width is 7.2. Otherwise users can input their own width.

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`PlotPLS.Imp` *Plot PLS important features*

Description

Plot PLS important features, BHan: added bgcolor parameter for B/W color

Usage

```
PlotPLS.Imp(
  mSetObj = NA,
  imgName,
  format = "png",
  dpi = 72,
  width = NA,
  type,
  feat.nm,
  feat.num,
  color.BW = FALSE
)
```

Arguments

`mSetObj`	Input name of the created mSet Object
`imgName`	Input a name for the plot
`format`	Select the image format, "png", or "pdf".
`dpi`	Input the dpi. If the image format is "pdf", users need not define the dpi. For "png" images, the default dpi is 72. It is suggested that for high-resolution images, select a dpi of 300.
`width`	Input the width, there are 2 default widths, the first, width = NULL, is 10.5. The second default is width = 0, where the width is 7.2. Otherwise users can input their own width.
`type`	Indicate the type variables of importance to use, "vip" to use VIp scores, or "type" for coefficients
`feat.nm`	Feature name
`feat.num`	Feature numbers
`color.BW`	Logical, true to use black and white, or false to not

Author(s)

Jeff Xia<`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

PlotPLS.Permutation *Plot PLS-DA classification performance using different components, permutation*

Description

Plot plsda classification performance using different components

Usage

```
PlotPLS.Permutation(
  mSetObj = NA,
  imgName,
  format = "png",
  dpi = 72,
  width = NA
)
```

Arguments

mSetObj	Input name of the created mSet Object
imgName	Input a name for the plot
format	Select the image format, "png", or "pdf".
dpi	Input the dpi. If the image format is "pdf", users need not define the dpi. For "png" images, the default dpi is 72. It is suggested that for high-resolution images, select a dpi of 300.
width	Input the width, there are 2 default widths, the first, width = NULL, is 10.5. The second default is width = 0, where the width is 7.2. Otherwise users can input their own width.

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

PlotPLS2DScore *Plot PLS score plot*

Description

Plot PLS score plot

Usage

```
PlotPLS2DScore(
  mSetObj = NA,
  imgName,
  format = "png",
  dpi = 72,
  width = NA,
  inx1,
  inx2,
  reg = 0.95,
  show = 1,
  grey.scale = 0,
  use.sparse = FALSE
)
```

Arguments

`mSetObj`	Input name of the created mSet Object
`imgName`	Input a name for the plot
`format`	Select the image format, "png", or "pdf".
`dpi`	Input the dpi. If the image format is "pdf", users need not define the dpi. For "png" images, the default dpi is 72. It is suggested that for high-resolution images, select a dpi of 300.
`width`	Input the width, there are 2 default widths, the first, width = NULL, is 10.5. The second default is width = 0, where the width is 7.2. Otherwise users can input their own width.
`inx1`	Numeric, indicate the number of the principal component for the x-axis of the loading plot.
`inx2`	Numeric, indicate the number of the principal component for the y-axis of the loading plot.
`reg`	Numeric, default is 0.95
`show`	Show labels, 1 or 0
`grey.scale`	Numeric, use a grey scale (0) or not (1)
`use.sparse`	Logical, use a sparse algorithm (T) or not (F)

Author(s)

Jeff Xia<`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`PlotPLS3DScore` *Plot PLS 3D score plot*

Description

Plot PLS 3D score plot

Usage

```
PlotPLS3DScore(mSetObj = NA, imgName, format = "json", inx1, inx2, inx3)
```

Arguments

`mSetObj`	Input name of the created mSet Object
`imgName`	Input a name for the plot
`format`	Select the image format, "png", or "pdf".
`inx1`	Numeric, indicate the number of the principal component for the x-axis of the loading plot.
`inx2`	Numeric, indicate the number of the principal component for the y-axis of the loading plot.
`inx3`	Numeric, indicate the number of the principal component for the z-axis of the loading plot.

Author(s)

Jeff Xia<`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`PlotPLS3DScoreImg` *Plot PLS 3D score plot*

Description

This function creates two 3D PLS-DA score plots, the first is static for Analysis Report purposes, as well as an interactive 3D plot using the plotly R package. The 3D score plot is saved in the created mSetObj (mSetObj$imgSet$plsda.3d). To view the score plot, if the name of your mSetObj is mSet, enter "mSet$imgSet$plsda.3d" to view the interactive score plot.

Usage

```
PlotPLS3DScoreImg(
  mSetObj = NA,
  imgName,
  format = "png",
  dpi = 72,
  width = NA,
  inx1,
  inx2,
  inx3,
  angl
)
```

Arguments

`mSetObj`	Input name of the created mSet Object
`imgName`	Input a name for the plot
`format`	Select the image format, "png", or "pdf".
`dpi`	Input the dpi. If the image format is "pdf", users need not define the dpi. For "png" images, the default dpi is 72. It is suggested that for high-resolution images, select a dpi of 300.
`width`	Input the width, there are 2 default widths, the first, width = NULL, is 10.5. The second default is width = 0, where the width is 7.2. Otherwise users can input their own width.
`inx1`	Numeric, indicate the number of the principal component for the x-axis of the loading plot.
`inx2`	Numeric, indicate the number of the principal component for the y-axis of the loading plot.
`inx3`	Numeric, indicate the number of the principal component for the z-axis of the loading plot.
`angl`	Input the angle

Author(s)

Jeff Xia<`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`PlotPLSLoading` *Plot PLS loading plot, also set the loading matrix for display*

Description

Plot PLS loading plot, also set the loading matrix for display

Usage

```
PlotPLSLoading(
  mSetObj = NA,
  imgName,
  format = "png",
  dpi = 72,
  width = NA,
  inx1,
  inx2
)
```

Arguments

mSetObj	Input name of the created mSet Object
imgName	Input a name for the plot
format	Select the image format, "png", or "pdf".
dpi	Input the dpi. If the image format is "pdf", users need not define the dpi. For "png" images, the default dpi is 72. It is suggested that for high-resolution images, select a dpi of 300.
width	Input the width, there are 2 default widths, the first, width = NULL, is 10.5. The second default is width = 0, where the width is 7.2. Otherwise users can input their own width.
inx1	Numeric, indicate the number of the principal component for the x-axis of the loading plot.
inx2	Numeric, indicate the number of the principal component for the y-axis of the loading plot.

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

PlotPLSPairSummary *Plot PLS pairwise summary*

Description

Plot PLS pairwise summary

Usage

```
PlotPLSPairSummary(
  mSetObj = NA,
  imgName,
  format = "png",
  dpi = 72,
```

```
  width = NA,
  pc.num
)
```

Arguments

mSetObj	Input name of the created mSet Object
imgName	Input a name for the plot
format	Select the image format, "png", or "pdf".
dpi	Input the dpi. If the image format is "pdf", users need not define the dpi. For "png" images, the default dpi is 72. It is suggested that for high-resolution images, select a dpi of 300.
width	Input the width, there are 2 default widths, the first, width = NULL, is 10.5. The second default is width = 0, where the width is 7.2. Otherwise users can input their own width.
pc.num	Numeric, indicate the number of principal components

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

PlotPowerProfile *Plot power profile*

Description

Plot power profile, specifying FDR level and sample size. It will return the image as well as the predicted power at various sample sizes.

Usage

```
PlotPowerProfile(mSetObj=NA, fdr.lvl, smplSize, imgName, format, dpi, width)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)
fdr.lvl	Specify the false-discovery rate level.
smplSize	Specify the maximum sample size, the number must be between 60-1000.
imgName	Specify the name to save the image as.
format	Specify the format of the image to save it as, either "png" or "pdf".
dpi	Specify the dots-per-inch (dpi). By default it is 72, for publications the recommended dpi is 300.
width	Specify the width of the image. NA specifies a width of 9, 0 specifies a width of 7, otherwise input a chosen width.

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`PlotPowerStat` *Plot power statistics*

Description

Create plot for power statistics

Usage

```
PlotPowerStat(mSetObj, imgName, format="png", dpi=72, width=NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)
`imgName`	Specify the name to save the image as.
`format`	Specify the format of the image to save it as, either "png" or "pdf"
`dpi`	Specify the dots-per-inch (dpi). By default it is 72, for publications the recommended dpi is 300.
`width`	Specify the width of the image. NA or 0 specifies a width of 10, otherwise input a chosen width.

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`PlotProbView` *Plot a summary view of the classification result*

Description

Plot of predicted class probabilities. On the x-axis is the proability, and the y-axis is the index of each predicted sample based on the probility. The samples are turned into separations at the x-axis. This plot can be created for multivariate ROC curve analysis using SVM, PLS, and RandomForest. Please note that sometimes, not all samples will be tested, instead they will be plotted at the 0.5 neutral line.

Usage

```
PlotProbView(mSetObj=NA, imgName, format="png", dpi=72, mdl.inx, show, showPred)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)
imgName	Input a name for the plot
format	Select the image format, "png", of "pdf".
dpi	Input the dpi. If the image format is "pdf", users need not define the dpi. For "png" images, the default dpi is 72. It is suggested that for high-resolution images, select a dpi of 300.
mdl.inx	Model index, 0 means to compare all models, -1 means to use the best model, input 1-6 to plot a ROC curve for one of the top six models
show	1 or 0, if 1, label samples classified to the wrong groups
showPred	Show predicted samples

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

---

plotProfile *Plot the variable across time points (x)*

Description

Colored by experimental conditions, used in higher function

Usage

```
plotProfile(mSetObj = NA, varName)
```

Arguments

mSetObj	Input name of the created mSet Object
varName	Input the name of the variable

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

PlotQEA.MetSet *View individual compounds related to a given metabolite set*

Description

View individual compounds related to a given metabolite set Functions for various plots for enrichment analysis

Usage

```
PlotQEA.MetSet(mSetObj=NA, setNM, format="png", dpi=72, width=NA)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)
setNM	Input the name of the metabolite set
format	Select the image format, "png", or "pdf".
dpi	Input the dpi. If the image format is "pdf", users need not define the dpi. For "png" images, the default dpi is 72. It is suggested that for high-resolution images, select a dpi of 300.
width	Input the width, there are 2 default widths, the first, width = NULL, is 10.5. The second default is width = 0, where the width is 7.2. Otherwise users can input their own width.

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

PlotQEA.Overview *Plot QEA overview*

Description

Plot QEA overview

Usage

```
PlotQEA.Overview(mSetObj=NA, imgName, imgOpt, format="png", dpi=72, width=NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)
`imgName`	Input a name for the plot
`imgOpt`	"net"
`format`	Select the image format, "png", or "pdf".
`dpi`	Input the dpi. If the image format is "pdf", users need not define the dpi. For "png" images, the default dpi is 72. It is suggested that for high-resolution images, select a dpi of 300.
`width`	Input the width, there are 2 default widths, the first, width = NULL, is 10.5. The second default is width = 0, where the width is 7.2. Otherwise users can input their own width.

Author(s)

Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`PlotRF.Classify` *Plot Random Forest*

Description

Random Forest plot

Usage

```
PlotRF.Classify(mSetObj, imgName, format, dpi, width)
```

Arguments

`mSetObj`	Input name of the created mSet Object
`imgName`	Input a name for the plot
`format`	Select the image format, "png", or "pdf".
`dpi`	Input the dpi. If the image format is "pdf", users need not define the dpi. For "png" images, the default dpi is 72. It is suggested that for high-resolution images, select a dpi of 300.
`width`	Input the width, there are 2 default widths, the first, width = NULL, is 10.5. The second default is width = 0, where the width is 7.2. Otherwise users can input their own width.

Author(s)

Jeff Xia<`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

PlotRF.Outlier *Plot Random Forest outliers*

Description

Random Forest plot of outliers

Usage

```
PlotRF.Outlier(mSetObj=NA, imgName, format="png", dpi=72, width=NA)
```

Arguments

mSetObj	Input name of the created mSet Object
imgName	Input a name for the plot
format	Select the image format, "png", or "pdf".
dpi	Input the dpi. If the image format is "pdf", users need not define the dpi. For "png" images, the default dpi is 72. It is suggested that for high-resolution images, select a dpi of 300.
width	Input the width, there are 2 default widths, the first, width = NULL, is 10.5. The second default is width = 0, where the width is 7.2. Otherwise users can input their own width.

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

PlotRF.VIP *Plot Random Forest variable importance*

Description

Random Forest plot of variable importance ranked by MeanDecreaseAccuracy

Usage

```
PlotRF.VIP(mSetObj=NA, imgName, format, dpi, width)
```

Arguments

mSetObj	Input name of the created mSet Object
imgName	Input a name for the plot
format	Select the image format, "png", or "pdf".
dpi	Input the dpi. If the image format is "pdf", users need not define the dpi. For "png" images, the default dpi is 72. It is suggested that for high-resolution images, select a dpi of 300.
width	Input the width, there are 2 default widths, the first, width = NULL, is 10.5. The second default is width = 0, where the width is 7.2. Otherwise users can input their own width.

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

PlotROC *Plot ROC*

Description

Pred and auroc are lists containing predictions and labels from different cross-validations

Usage

```
PlotROC(mSetObj=NA, imgName, format="png", dpi=72, mdl.inx, avg.method, show.conf, show.holdout, focus
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)
imgName	Input a name for the plot
format	Select the image format, "png", of "pdf".
dpi	Input the dpi. If the image format is "pdf", users need not define the dpi. For "png" images, the default dpi is 72. It is suggested that for high-resolution images, select a dpi of 300.
mdl.inx	Model index, 0 means to compare all models, input 1-6 to plot a ROC curve for one of the top six models
avg.method	Input the method to compute the average ROC curve, either "threshold", "vertical" or "horizontal"
show.conf	Logical, if 1, show confidence interval, if 0 do not show
show.holdout	Logical, if 1, show the ROC curve for hold-out validation, if 0 do not show
focus	"fpr"
cutoff	Input the threshold to limit the calculation of the ROC curve, the number must be between 0 and 1.

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`PlotROC.LRmodel` *Plot ROC for the logistic regression model*

Description

Plot ROC for the logistic regression model

Usage

```
PlotROC.LRmodel(
  mSetObj = NA,
  imgName,
  format = "png",
  dpi = 72,
  show.conf = FALSE,
  sp.bin = 0.01
)
```

Arguments

`mSetObj`	Input name of the created mSet Object
`imgName`	Input a name for the plot
`format`	Select the image format, "png", or "pdf".
`dpi`	Input the dpi. If the image format is "pdf", users need not define the dpi. For "png" images, the default dpi is 72. It is suggested that for high-resolution images, select a dpi of 300.
`show.conf`	Logical, show confidence intervals
`sp.bin`	Numeric, default is set to 0.01.

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`PlotROCTest` *Plot ROC for the ROC Curve Based Model Creation and Evaluation module*

Description

Plot the ROC curve of the biomarker model created using a user-selected subset of features. Pred and auroc are lists containing predictions and labels from different cross-validations.

Usage

```
PlotROCTest(mSetObj=NA, imgName, format="png", dpi=72, mdl.inx, avg.method, show.conf, show.holdout, f
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)
`imgName`	Input a name for the plot
`format`	Select the image format, "png", of "pdf".
`dpi`	Input the dpi. If the image format is "pdf", users need not define the dpi. For "png" images, the default dpi is 72. It is suggested that for high-resolution images, select a dpi of 300.
`mdl.inx`	Model index, 0 means to compare all models, input 1-6 to plot a ROC curve for one of the top six models
`avg.method`	Input the method to compute the average ROC curve, either "threshold", "vertical" or "horizontal"
`show.conf`	Logical, if 1, show confidence interval, if 0 do not show
`show.holdout`	Logical, if 1, show the ROC curve for hold-out validation, if 0 do not show
`focus`	"fpr"
`cutoff`	Input the threshold to limit the calculation of the ROC curve, the number must be between 0 and 1.

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

PlotRSVM.Classification

Recursive Support Vector Machine (R-SVM) plot

Description

Plot recursive SVM classification

Usage

```
PlotRSVM.Classification(mSetObj, imgName, format, dpi, width)
```

Arguments

mSetObj	Input name of the created mSet Object
imgName	Input a name for the plot
format	Select the image format, "png", or "pdf".
dpi	Input the dpi. If the image format is "pdf", users need not define the dpi. For "png" images, the default dpi is 72. It is suggested that for high-resolution images, select a dpi of 300.
width	Input the width, there are 2 default widths, the first, width = NULL, is 10.5. The second default is width = 0, where the width is 7.2. Otherwise users can input their own width.

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

PlotRSVM.Cmpd

Recursive Support Vector Machine (R-SVM) plot of important variables

Description

Plot recursive SVM variables of importance if too many, plot top 15

Usage

```
PlotRSVM.Cmpd(mSetObj=NA, imgName, format="png", dpi=72, width=NA)
```

Arguments

mSetObj	Input name of the created mSet Object
imgName	Input a name for the plot
format	Select the image format, "png", or "pdf".
dpi	Input the dpi. If the image format is "pdf", users need not define the dpi. For "png" images, the default dpi is 72. It is suggested that for high-resolution images, select a dpi of 300.
width	Input the width, there are 2 default widths, the first, width = NULL, is 10.5. The second default is width = 0, where the width is 7.2. Otherwise users can input their own width.

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

PlotSAM.Cmpd *Plot SAM*

Description

Plot SAM with positive and negative metabolite sets

Usage

```
PlotSAM.Cmpd(mSetObj = NA, imgName, format = "png", dpi = 72, width = NA)
```

Arguments

mSetObj	Input name of the created mSet Object
imgName	Input a name for the plot
format	Select the image format, "png", or "pdf".
dpi	Input the dpi. If the image format is "pdf", users need not define the dpi. For "png" images, the default dpi is 72. It is suggested that for high-resolution images, select a dpi of 300.
width	Input the width, there are 2 default widths, the first, width = NULL, is 10.5. The second default is width = 0, where the width is 7.2. Otherwise users can input their own width.

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

PlotSAM.FDR *Plot SAM Delta Plot*

Description

Plot SAM Delta Plot (FDR)

Usage

```
PlotSAM.FDR(mSetObj = NA, imgName, format = "png", dpi = 72, width = NA)
```

Arguments

mSetObj	Input name of the created mSet Object
imgName	Input a name for the plot
format	Select the image format, "png", or "pdf".
dpi	Input the dpi. If the image format is "pdf", users need not define the dpi. For "png" images, the default dpi is 72. It is suggested that for high-resolution images, select a dpi of 300.
width	Input the width, there are 2 default widths, the first, width = NULL, is 10.5. The second default is width = 0, where the width is 7.2. Otherwise users can input their own width.
delta	Input the delta

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

PlotSampleNormSummary *Two plot summary plot: Sample View of before and after normalization*

Description

For each plot, the top is a density plot and the bottom is a box plot.

Usage

```
PlotSampleNormSummary(mSetObj=NA, imgName, format="png", dpi=72, width=NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)
`imgName`	Input a name for the plot
`format`	Select the image format, "png", of "pdf".
`dpi`	Input the dpi. If the image format is "pdf", users need not define the dpi. For "png" images, the default dpi is 72. It is suggested that for high-resolution images, select a dpi of 300.
`width`	Input the width, there are 2 default widths, the first, width = NULL, is 10.5. The second default is width = 0, where the width is 7.2. Otherwise users can input their own width.

Author(s)

Jeff Xia <`jeff.xia@mcgill.ca`>, Jasmine Chong McGill University, Canada

`PlotSelectedFeature` *Create a box-plot of a feature's expression pattern across the different datasets*

Description

This function plots a box-plot of the expression pattern of a user-selected feature across the different datasets included in meta-analysis.

Usage

```
PlotSelectedFeature(mSetObj = NA, gene.id)
```

Arguments

`mSetObj`	Input name of the created mSet Object.
`gene.id`	Input the name of the selected feature.

Author(s)

Jeff Xia<`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

PlotSigVar *Supporting function for plotting important variables for each factor*

Description

Supporting function for plotting important variables for each factor note, by control xpd to plot legend outside the plotting area without using layout

Usage

```
PlotSigVar(x, y, xline, yline, title)
```

Arguments

x	Input the X variable
y	Input the Y variable
xline	Input the xline
yline	Input the yline
title	Input the title

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

PlotSOM *SOM Plot*

Description

Plot SOM map for less than 20 clusters

Usage

```
PlotSOM(mSetObj = NA, imgName, format = "png", dpi = 72, width = NA)
```

Arguments

mSetObj	Input name of the created mSet Object
imgName	Input a name for the plot
format	Select the image format, "png", or "pdf".
dpi	Input the dpi. If the image format is "pdf", users need not define the dpi. For "png" images, the default dpi is 72. It is suggested that for high-resolution images, select a dpi of 300.
width	Input the width, there are 2 default widths, the first, width = NULL, is 10.5. The second default is width = 0, where the width is 7.2. Otherwise users can input their own width.

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`PlotSPLS2DScore` *Score Plot SPLS-DA*

Description

Sparse PLS-DA (from mixOmics) score plot

Usage

```
PlotSPLS2DScore(
  mSetObj = NA,
  imgName,
  format = "png",
  dpi = 72,
  width = NA,
  inx1,
  inx2,
  reg = 0.95,
  show = 1,
  grey.scale = 0
)
```

Arguments

`mSetObj`	Input name of the created mSet Object
`imgName`	Input a name for the plot
`format`	Select the image format, "png", or "pdf".
`dpi`	Input the dpi. If the image format is "pdf", users need not define the dpi. For "png" images, the default dpi is 72. It is suggested that for high-resolution images, select a dpi of 300.
`width`	Input the width, there are 2 default widths, the first, width = NULL, is 10.5. The second default is width = 0, where the width is 7.2. Otherwise users can input their own width.
`inx1`	Numeric, indicate the number of the principal component for the x-axis of the loading plot.
`inx2`	Numeric, indicate the number of the principal component for the y-axis of the loading plot.
`reg`	Numeric, between 1 and 0
`show`	Numeric, 1 or 0
`grey.scale`	Numeric, use grey-scale, 0 for no, and 1 for yes.

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`PlotSPLS3DScore` *3D SPLS-DA score plot*

Description

Sparse PLS-DA (from mixOmics) 3D score plot

Usage

```
PlotSPLS3DScore(
  mSetObj = NA,
  imgName,
  format = "json",
  inx1 = 1,
  inx2 = 2,
  inx3 = 3
)
```

Arguments

`mSetObj`	Input name of the created mSet Object
`imgName`	Input a name for the plot
`format`	Select the image format, "png", or "pdf".
`inx1`	Numeric, indicate the number of the principal component for the x-axis of the loading plot.
`inx2`	Numeric, indicate the number of the principal component for the y-axis of the loading plot.
`inx3`	Numeric, indicate the number of the principal component for the z-axis of the loading plot.

Author(s)

Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`PlotSPLS3DScoreImg` *Plot sPLS-DA 3D score plot*

Description

This function creates two 3D sPLS-DA score plots, the first is static for Analysis Report purposes, as well as an interactive 3D plot using the plotly R package. The 3D score plot is saved in the created mSetObj (mSetObj$imgSet$splsda.3d). To view the score plot, if the name of your mSetObj is mSet, enter "mSet$imgSet$splsda.3d" to view the interactive score plot.

This function creates two 3D sPLS-DA score plots, the first is static for Analysis Report purposes, as well as an interactive 3D plot using the plotly R package. The 3D score plot is saved in the created mSetObj (mSetObj$imgSet$splsda.3d). To view the score plot, if the name of your mSetObj is mSet, enter "mSet$imgSet$splsda.3d" to view the interactive score plot.

Usage

```
PlotSPLS3DScoreImg(
  mSetObj = NA,
  imgName,
  format = "png",
  dpi = 72,
  width = NA,
  inx1,
  inx2,
  inx3,
  angl
)

PlotSPLS3DScoreImg(
  mSetObj = NA,
  imgName,
  format = "png",
  dpi = 72,
  width = NA,
  inx1,
  inx2,
  inx3,
  angl
)
```

Arguments

`mSetObj`	Input name of the created mSet Object
`imgName`	Input a name for the plot
`format`	Select the image format, "png", or "pdf".
`dpi`	Input the dpi. If the image format is "pdf", users need not define the dpi. For "png" images, the default dpi is 72. It is suggested that for high-resolution images, select a dpi of 300.

width	Input the width, there are 2 default widths, the first, width = NULL, is 10.5. The second default is width = 0, where the width is 7.2. Otherwise users can input their own width.
inx1	Numeric, indicate the number of the principal component for the x-axis of the loading plot.
inx2	Numeric, indicate the number of the principal component for the y-axis of the loading plot.
inx3	Numeric, indicate the number of the principal component for the z-axis of the loading plot.
angl	Input the angle

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`PlotSPLSDA.Classification`

Create SPLS-DA classification plot

Description

Sparse PLS-DA (from mixOmics) plot of classification performance using different components

Usage

```
PlotSPLSDA.Classification(
  mSetObj = NA,
  imgName,
  format = "png",
  dpi = 72,
  width = NA
)
```

Arguments

mSetObj	Input name of the created mSet Object
imgName	Input a name for the plot
format	Select the image format, "png", or "pdf".
dpi	Input the dpi. If the image format is "pdf", users need not define the dpi. For "png" images, the default dpi is 72. It is suggested that for high-resolution images, select a dpi of 300.
width	Input the width, there are 2 default widths, the first, width = NULL, is 10.5. The second default is width = 0, where the width is 7.2. Otherwise users can input their own width.
validOpt	"Mfold"

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`PlotSPLSLoading` *Create SPLS-DA loading plot*

Description

Sparse PLS-DA (from mixOmics) loading plot

Usage

```
PlotSPLSLoading(
  mSetObj = NA,
  imgName,
  format = "png",
  dpi = 72,
  width = NA,
  inx,
  viewOpt = "detail"
)
```

Arguments

`mSetObj`	Input name of the created mSet Object
`imgName`	Input a name for the plot
`format`	Select the image format, "png", or "pdf".
`dpi`	Input the dpi. If the image format is "pdf", users need not define the dpi. For "png" images, the default dpi is 72. It is suggested that for high-resolution images, select a dpi of 300.
`width`	Input the width, there are 2 default widths, the first, width = NULL, is 10.5. The second default is width = 0, where the width is 7.2. Otherwise users can input their own width.
`inx`	Input the model index
`viewOpt`	Detailed view "detail"

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

PlotSPLSPairSummary *Plot SPLS-DA*

Description

Sparse PLS-DA (from mixOmics) pairwise summary plot

Usage

```
PlotSPLSPairSummary(
  mSetObj = NA,
  imgName,
  format = "png",
  dpi = 72,
  width = NA,
  pc.num
)
```

Arguments

mSetObj	Input name of the created mSet Object
imgName	Input a name for the plot
format	Select the image format, "png", or "pdf".
dpi	Input the dpi. If the image format is "pdf", users need not define the dpi. For "png" images, the default dpi is 72. It is suggested that for high-resolution images, select a dpi of 300.
width	Input the width, there are 2 default widths, the first, width = NULL, is 10.5. The second default is width = 0, where the width is 7.2. Otherwise users can input their own width.
pc.num	Numeric, indicate the number of principle components

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

PlotSubHeatMap *Create Sub Heat Map Plot*

Description

Plot a sub heatmap based on results from t-tests/ANOVA, VIP or randomforest

Usage

```
PlotSubHeatMap(
  mSetObj = NA,
  imgName,
  format = "png",
  dpi = 72,
  width = NA,
  dataOpt,
  scaleOpt,
  smplDist,
  clstDist,
  palette,
  method.nm,
  top.num,
  viewOpt,
  rowV = T,
  colV = T,
  border = T,
  grp.ave = F
)
```

Arguments

`mSetObj`	Input name of the created mSet Object
`imgName`	Input a name for the plot
`format`	Select the image format, "png", or "pdf".
`dpi`	Input the dpi. If the image format is "pdf", users need not define the dpi. For "png" images, the default dpi is 72. It is suggested that for high-resolution images, select a dpi of 300.
`width`	Input the width, there are 2 default widths, the first, width = NULL, is 10.5. The second default is width = 0, where the width is 7.2. Otherwise users can input their own width.
`dataOpt`	Set data options
`scaleOpt`	Set the image scale
`smplDist`	Input the sample distance method
`clstDist`	Input the clustering distance method
`palette`	Input color palette choice
`method.nm`	Input the method for sub-heat map
`top.num`	Input the top number
`viewOpt`	Set heatmap options, default is set to "detail"
`rowV`	Default is set to T
`colV`	Default is set to T
`border`	Indicate whether or not to show cell-borders, default is set to T
`grp.ave`	Logical, default is set to F

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`PlotTestAccuracy` *Plot classification performance using different features for Biomarker Tester*

Description

Plot of the accuracy of classification with an increasing number of features.

Usage

```
PlotTestAccuracy(mSetObj=NA, imgName, format="png", dpi=72)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)
`imgName`	Input a name for the plot
`format`	Select the image format, "png", of "pdf".
`dpi`	Input the dpi. If the image format is "pdf", users need not define the dpi. For "png" images, the default dpi is 72. It is suggested that for high-resolution images, select a dpi of 300.

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`PlotTT` *Plot t-test*

Description

Plot t-test

Usage

```
PlotTT(mSetObj=NA, imgName, format="png", dpi=72, width=NA)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)
imgName	Input a name for the plot
format	Select the image format, "png", or "pdf".
dpi	Input the dpi. If the image format is "pdf", users need not define the dpi. For "png" images, the default dpi is 72. It is suggested that for high-resolution images, select a dpi of 300.
width	Input the width, there are 2 default widths, the first, width = NULL, is 10.5. The second default is width = 0, where the width is 7.2. Otherwise users can input their own width.

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

PlotVolcano *Create volcano plot*

Description

For labelling interesting points, it is defined by the following rules: need to be signficant (sig.inx) and or 2. top 5 p, or 2. top 5 left, or 3. top 5 right.

Usage

```
PlotVolcano(mSetObj=NA, imgName, plotLbl, format="png", dpi=72, width=NA)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)
imgName	Input a name for the plot
plotLbl	Logical, plot labels, 1 for yes and 0 for no.
format	Select the image format, "png", or "pdf".
dpi	Input the dpi. If the image format is "pdf", users need not define the dpi. For "png" images, the default dpi is 72. It is suggested that for high-resolution images, select a dpi of 300.
width	Input the width, there are 2 default widths, the first, width = NULL, is 10.5. The second default is width = 0, where the width is 7.2. Otherwise users can input their own width.

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

PLSDA.CV *PLS-DA classification and feature selection*

Description

PLS-DA classification and feature selection

Usage

```
PLSDA.CV(
  mSetObj = NA,
  methodName = "T",
  compNum = GetDefaultPLSCVComp(mSetObj),
  choice = "Q2"
)
```

Arguments

mSetObj	Input name of the created mSet Object
methodName	Logical, by default set to TRUE
compNum	GetDefaultPLSCVComp()
choice	Input the choice, by default it is Q2

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

PLSDA.Permut *Perform PLS-DA permutation*

Description

Perform PLS-DA permutation using training classification accuracy as indicator, for two or multi-groups

Usage

```
PLSDA.Permut(mSetObj = NA, num = 100, type = "accu")
```

Arguments

mSetObj	Input name of the created mSet Object
num	Numeric, input the number of permutations
type	Type of accuracy, if "accu" indicate prediction accuracy, else "sep" is separation distance

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`PLSR.Anal` *PLS analysis using oscorespls (Orthogonal scores algorithm) so that VIP can be calculated note: the VIP is calculated only after PLSDA-CV is performed to determine the best # of comp. used for VIP*

Description

PLS analysis using oscorespls

Usage

```
PLSR.Anal(mSetObj = NA, reg = FALSE)
```

Arguments

`mSetObj`	Input name of the created mSet Object
`reg`	Logical

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`Predict.class` *Get predicted class probability*

Description

Get predicted class probability

Usage

```
Predict.class(x.train, y.train, x.test, clsMethod = "pls", lvNum, imp.out = F)
```

Arguments

`x.train`	Input the x training samples
`y.train`	Input the y training samples
`x.test`	Input the x testing samples
`clsMethod`	Se the classification method, default is set to pls
`lvNum`	Input the number of levels
`imp.out`	Logical, set to F by default

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

PrepareIntegData *Prepare integrated data*

Description

Used for the pathinteg module.

Usage

```
PrepareIntegData(mSetObj = NA)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)

PrepareNetworkData *Prepare data for network exploration*

Description

Function for the network explorer module, prepares user's data for network exploration.

Usage

```
PrepareNetworkData(mSetObj = NA)
```

Arguments

mSetObj	Input name of the created mSet Object

`PreparePDFReport` *Create report of analyses*

Description

Report generation using Sweave Note: most analyses were already performed, only need to embed the results to the right place without rerunning the whole analysis through Sweave. Only some auxilliary info (i.e. time, version etc need to run in R through Sweave

Usage

```
PreparePDFReport(mSetObj = NA, usrName)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)
`usrName`	Input the name of the user

Author(s)

Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`PreparePermResult` *Prepare report for permutation tests*

Description

Function to prepare a report for permutation tests, used in higher functions

Usage

```
PreparePermResult(perm.vec)
```

Arguments

`perm.vec`	Input permutation vector

Author(s)

Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

PreparePrenormData *Prepare data for normalization*

Description

Function should always be initialized (new or overwrite previous prenorm object).

Usage

```
PreparePrenormData(mSetObj = NA)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)

PrepareQueryJson *Prepare user's query for mapping KEGG Global Metabolic Network*

Description

This function prepares the user's data for the KEGG Global Metabolic Network

Usage

```
PrepareQueryJson(mSetObj = NA)
```

Arguments

mSetObj	Input name of the created mSet Object

Author(s)

Othman Soufan, Jeff Xia <jeff.xia@mcgill.ca>, othman.soufan@mcgill.ca McGill University, Canada License: GNU GPL (¿= 2)

PrepareROCData *Prepare data for ROC analysis*

Description

Prepare data for ROC analysis

Usage

```
PrepareROCData(mSetObj = NA)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

PrepareROCDetails *ROC with CI for AUC*

Description

ROC with CI for AUC

Usage

```
PrepareROCDetails(mSetObj = NA, feat.nm)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)
feat.nm	Input the feature name

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

PrepareVennData *Prepare data for Venn diagram*

Description

Prepare data for Venn diagram

Usage

```
PrepareVennData(mSetObj = NA)
```

Arguments

mSetObj	Input name of the created mSet Object

RankFeatures *Rank features based on different importance measures*

Description

Ranks features based on various importance measures, return imp.vec which contains the importance measures of unordered features

Usage

```
RankFeatures(x.in, y.in, method, lvNum)
```

Arguments

x.in	Input the X features
y.in	Input the Y features
method	Input the method
lvNum	Input the number of levels

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

rda2list *Utility function Make list of KEGG rda files*

Description

Utility function Make list of KEGG rda files

Usage

```
rda2list(file)
```

Read.BatchCSVdata *Data I/O for batch effect checking*

Description

Read multiple user uploaded CSV data one by one format: row, col

Usage

```
Read.BatchCSVdata(mSetObj = NA, filePath, format)
```

Arguments

mSetObj	Input name of the created mSet Object
filePath	Input the path to the batch files
format	Input the format of the batch files
label	Input the label-type of the files

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

Read.MSspec *Read LC/GC-MS spectra (.netCDF, .mzXML, mzData)*

Description

This function handles reading in LC/GC-MS spectra files and fills in the dataSet object. It uses functions from the XCMS package to perform peak detection and alignment (grouping).

Usage

```
Read.MSspec(mSetObj, folderName, profmethod, fwhm, bw)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)
folderName	the name of the folder containing the MS spectra
profmethod	specify the method to use for profile generation, supports "bin", "binlin", "binlinbase" and "intlin"
fwhm	specify the full width at half maximum of the matched filtration gaussian model peak
bw	define the bandwidth (standard deviation of the smoothing kernel) to be used

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

Read.PeakList *Read peak list files*

Description

This function reads peak list files and fills the data into a dataSet object. For NMR peak lists, the input should be formatted as two-columns containing numeric values (ppm, int). Further, this function will change ppm to mz, and add a dummy 'rt'. For MS peak data, the lists can be formatted as two-columns (mz, int), in which case the function will add a dummy 'rt', or the lists can be formatted as three-columns (mz, rt, int).

Usage

```
Read.PeakList(mSetObj=NA, foldername)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects).
foldername	Name of the folder containing the NMR or MS peak list files to read.

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`Read.PeakListData` *Constructor to read uploaded user files into the mummichog object*

Description

This function handles reading in CSV or TXT files and filling in the mSet object for mummichog analysis. It makes sure that all necessary columns are present.

Usage

```
Read.PeakListData(mSetObj=NA, filename = NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj.
`filename`	Input the path name for the CSV/TXT files to read.

Author(s)

Jasmine Chong, Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`Read.TextData` *Constructor to read uploaded CSV or TXT files into the dataSet object*

Description

This function handles reading in CSV or TXT files and filling in the dataSet object created using "InitDataObjects".

Usage

```
Read.TextData(mSetObj=NA, filePath, format, lbl.type)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects).
`filePath`	Input the path name for the CSV/TXT files to read.
`format`	Specify if samples are paired and in rows (rowp), unpaired and in rows (rowu), in columns and paired (colp), or in columns and unpaired (colu).
`lbl.type`	Specify the data label type, either discrete (disc) or continuous (cont).

Author(s)

Jeff Xia <jeff.xia@mcgill.ca>, Jasmine Chong McGill University, Canada License: GNU GPL (¿= 2)

ReadIndData *Read in individual data*

Description

This function determines reads in user's individual data for meta-analysis.

Usage

```
ReadIndData(mSetObj = NA, dataName, format = "colu")
```

Arguments

mSetObj	Input name of the created mSet Object
dataName	Name of inputted dataset.
format	Specify if samples are paired and in rows (rowp), unpaired and in rows (rowu), in columns and paired (colp), or in columns and unpaired (colu).

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

ReadPairFile *Read paired peak or spectra files*

Description

This function reads paired peak lists or spectra files. The pair information is stored in a file where each line is a pair and names are separated by ":".

Usage

```
ReadPairFile(filePath = "pairs.txt")
```

Arguments

filePath	Set file path

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`RecordRCommand` *Record R Commands*

Description

Record R Commands

Usage

```
RecordRCommand(mSetObj = NA, cmd)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)
`cmd`	Commands

`RegisterData` *Register data in R*

Description

When there are multiple datasets, record their name and save the inputted data as a .RDS file to save memory. Note, the memory will only contain one mSetObj$dataSet object. By default the last one will be the most recent/current dataSet object. Users can switch which data to load into memory.

Usage

```
RegisterData(mSetObj = NA, dataSet)
```

Arguments

`mSetObj`	Input name of the created mSet Object
`dataSet`	Input dataset to be registered in R.

Author(s)

Jeff Xia<`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`RemoveCmpd` *Remove selected compounds*

Description

Remove compounds

Usage

```
RemoveCmpd(mSetObj = NA, inx)
```

Arguments

`mSetObj`	Input name of the created mSet Object
`inx`	Input the index of compound to remove

`RemoveData` *Remove data object, the current dataSet will be the last one by default*

Description

Remove data object, the current dataSet will be the last one by default

Usage

```
RemoveData(dataName)
```

Arguments

`dataName`	Input name of data to remove

`RemoveDuplicates` *Given a data with duplicates, remove duplicates*

Description

Dups is the one with duplicates

Usage

```
RemoveDuplicates(data, lvlOpt = "mean", quiet = T)
```

Arguments

`data`	Input data to remove duplicates
`lvlOpt`	Set options, default is mean
`quiet`	Set to quiet, logical, default is T

Author(s)

Jeff Xia<`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`RemoveFile` *Remove file*

Description

Remove file

Usage

```
RemoveFile(fileName)
```

Arguments

`fileName`	Input name of file to remove

Author(s)

Jeff Xia<`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

RemoveFolder *Remove folder*

Description

Remove folder

Usage

```
RemoveFolder(folderName)
```

Arguments

folderName	Input name of folder to remove

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

RemoveGene *Remove selected genes*

Description

Remove selected genes based on an index

Usage

```
RemoveGene(mSetObj = NA, inx)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)
inx	Input compound index

`RemoveMissingPercent` *Data processing: remove variables with missing values*

Description

Remove variables based upon a user-defined percentage cut-off of missing values. If a user specifies a threshold of 20 in at least 20

Usage

```
RemoveMissingPercent(mSetObj, percent)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)
`percent`	Input the percentage cut-off you wish to use. For instance, 50 percent is represented by percent=0.5.

Author(s)

Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`ReplaceMin` *Replace missing or zero values*

Description

This function will replace zero/missing values by half of the smallest positive value in the original dataset. This method will be called after all missing value imputations are conducted. Also, it directly modifies the mSet$dataSet$proc if executed after normalization, or the mSet$dataSet$norm if before normalization.

Usage

```
ReplaceMin(mSetObj=NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

RerenderMetPAGraph *Redraw current graph for zooming or clipping then return a value*

Description

Redraw current graph for zooming or clipping then return a value

Usage

```
RerenderMetPAGraph(mSetObj = NA, imgName, width, height, zoom.factor = NA)
```

Arguments

mSetObj	Input name of the created mSet Object
imgName	Input the name of the plot
width	Input the width, there are 2 default widths, the first, width = NULL, is 10.5. The second default is width = 0, where the width is 7.2. Otherwise users can input their own width.
height	Input the height of the created plot.

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

resultIncreased_doe *Identify whether results improved or not*

Description

Identify whether results improved or not

Usage

```
resultIncreased_doe(history)
```

Arguments

history	List, an interal media objects used to save the optimization results of peaks.

Author(s)

Zhiqiang Pang <zhiqiang.pang@mail.mcgill.ca> Jeff Xia <jeff.xia@mcgill.ca> Mcgill University License: GNU GPL (¿= 2)

`RF.Anal` *Perform Random Forest Analysis*

Description

Perform Random Forest

Usage

```
RF.Anal(mSetObj = NA, treeNum = 500, tryNum = 7, randomOn = 1)
```

Arguments

`mSetObj`	Input name of the created mSet Object
`treeNum`	Input the number of trees to create, default is set to 500
`tryNum`	Set number of tries, default is 7
`randomOn`	Set random, default is 1

Author(s)

Jeff Xia<`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`ROCPredSamplesTable` *Create a table of newly classified samples*

Description

Function to create the table of newly classified samples

Usage

```
ROCPredSamplesTable(mSetObj = NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects) Function to create the table of newly classified samples

RSVM *R-SVM core code*

Description

Core code to perform R-SVM

Usage

```
RSVM(x, y, ladder, CVtype, CVnum = 0)
```

Arguments

x	Row matrix of data
y	Class label: 1 / -1 for 2 classes
ladder	Input the ladder
CVtype	Integer (N fold CV), "LOO" leave-one-out CV, "bootstrape" bootstrape CV
CVnum	Number of CVs, LOO: defined as sample size, Nfold and bootstrape: user defined, default as sample size outputs a named list Error: a vector of CV error on each level SelFreq: a matrix for the frequency of each gene being selected in each level with each column corresponds to a level of selection and each row for a gene The top important gene in each level are those high-freqent ones

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

RSVM.Anal *Recursive Support Vector Machine (R-SVM)*

Description

recursive SVM for feature selection and classification

Usage

```
RSVM.Anal(mSetObj = NA, cvType)
```

Arguments

mSetObj	Input name of the created mSet Object
cvType	Cross-validation type

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`RT.Adjust_Slave` *RT.Adjust_Slave*

Description

RT.Adjust_Slave

Usage

```
RT.Adjust_Slave(
  peaks,
  peakIndex,
  rtime,
  minFraction = 0.9,
  extraPeaks = 1,
  smooth = c("loess", "linear"),
  span = 0.2,
  family = c("gaussian", "symmetric"),
  peakGroupsMatrix = matrix(ncol = 0, nrow = 0),
  subsetAdjust = c("average", "previous")
)
```

Author(s)

Zhiqiang Pang, Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`rt.trim_random` *Data trimming Method Based on Random RT*

Description

Trim raw data scan signal randomly in the RT dimension.

Usage

```
rt.trim_random(raw_data, ms_list)
```

Arguments

`raw_data`	MSnExp object, the raw data that has been read in memory.
`ms_list`	List, the names list of all scans.

Author(s)

Zhiqiang Pang <zhiqiang.pang@mail.mcgill.ca> Jeff Xia <jeff.xia@mcgill.ca> Mcgill University License: GNU GPL (¿= 2)

---

rt.trim_specific *Data trimming Method Based on Specific RT*

Description

Trim data based on specific RT values. Positive values will be specially retained, while the negative values will be removed.

Usage

```
rt.trim_specific(raw_data, ms_list, rt, rtdiff = 10)
```

Arguments

raw_data	MSnExp object, the raw data that has been read in memory.
ms_list	List, the names list of all scans.
mz	Numeric, the specifric RT value that will be kept or removed.
mzdiff	Numeric, the deviation (ppm) for the 'rt' values. Default is 100.

Author(s)

Zhiqiang Pang <zhiqiang.pang@mail.mcgill.ca> Jeff Xia <jeff.xia@mcgill.ca> Mcgill University License: GNU GPL (¿= 2)

SAM.Anal *Perform Signifiance Analysis of Microarrays (SAM) analysis*

Description

Perform SAM

Usage

```
SAM.Anal(
  mSetObj = NA,
  method = "d.stat",
  paired = FALSE,
  varequal = TRUE,
  delta = 0,
  imgName
)
```

Arguments

`mSetObj`	Input name of the created mSet Object
`method`	Method for SAM analysis, default is set to "d.stat", alternative is "wilc.stat"
`paired`	Logical, indicates if samples are paired or not. Default is set to FALSE
`varequal`	Logical, indicates if variance is equal. Default is set to TRUE

Author(s)

Jeff Xia<`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`SanityCheckData` *Sanity Check Data*

Description

SanityCheckData is used for data processing, and performs a basic sanity check of the uploaded content, ensuring that the data is suitable for further analysis. The function will return a message if the data has successfully passed the check and is deemed suitable for further analysis. If it fails, the function will return a 0. The function will perform the check directly onto the mSet$dataSet object, and must be performed immediately after reading in data. The sanity check function evaluates the accuracy of sample and class labels, data structure, deals with non-numeric values, removes columns that are constant across all samples (variance = 0), and by default replaces missing values with half of the original minimal positive value in your dataset.

Usage

```
SanityCheckData(mSetObj=NA)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)

Author(s)

Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

SanityCheckIndData *Sanity check of individual datasets for meta-analysis*

Description

Performs a sanity check on each-uploaded dataset for meta-analysis. Briefly, this function will exclude empty rows, check class labels, ensure only 2 groups are being compared within the dataset, ensure sample names are unique, remove low quality samples/features, and replace missing values.

Usage

```
SanityCheckIndData(mSetObj = NA, dataName)
```

Arguments

mSetObj	Input name of the created mSet Object
dataName	Input name of the dataset to perform the sanity check.

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

SanityCheckMummichogData
Sanity Check Data

Description

SanityCheckData is used for data processing, and performs a basic sanity check of the uploaded data, ensuring that the data is suitable for further analysis. The function ensure that all parameters are properly set based on updated parameters.

Usage

```
SanityCheckMummichogData(mSetObj=NA)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects).

Author(s)

Jasmine Chong, Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

SaveTransformedData *Save the processed data with class names*

Description

This function saves the processed data with class names as CSV files. Several files may be generated, the original data, processed data, peak normalized, and/or normalized data.

Usage

```
SaveTransformedData(mSetObj = NA)
```

Arguments

mSetObj	Input name of the created mSet Object

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

SearchByCompound *Search for compound from all member compounds of metabolite set*

Description

Search for compound from all member compounds of metabolite set

Usage

```
SearchByCompound(mSetObj = NA, query)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)
query	Input the query to search

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

SearchByName *Given a metabolite set name, search its index*

Description

Given a metabolite set name, search its index

Usage

```
SearchByName(mSetObj = NA, query)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)
query	Input the query to search

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

SearchMsetLibraries *Search metabolite set libraries*

Description

Search metabolite set libraries

Usage

```
SearchMsetLibraries(mSetObj = NA, query, type)
```

Arguments

mSetObj	Input name of the created mSet Object
query	Input the query to search
type	Input the data type (name or compound)

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

SearchNetDB — *Perform mapping of user's data to interaction network*

Description

This function performs mapping of user's data to the internal network to create a network from the seed nodes

Usage

```
SearchNetDB(
  mSetObj = NA,
  db.type,
  table.nm,
  require.exp = TRUE,
  min.score = 900
)
```

Arguments

mSetObj	Input name of the created mSet Object
db.type	Input the database type
table.nm	Input the organism code for the sqlite table (ppi). For chemical type, the table.nm is drugbank of ctd
require.exp	Logical, only used for the STRING database
min.score	Input the minimal score, only used for the STRING database

Author(s)

Othman Soufan, Jeff Xia <jeff.xia@mcgill.ca>, othman.soufan@mcgill.ca McGill University, Canada License: GNU GPL (¿= 2)

SelectMultiData — *Select one or more datasets for meta-analysis*

Description

This function selects one or more datasets to be used for meta-analysis. 1 is used to indicate that a dataset is selected and by default, all datasets will be selected for meta-analysis.

Usage

```
SelectMultiData(mSetObj = NA)
```

Arguments

mSetObj	Input name of the created mSet Object

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

SetAnalysisMode *Set biomarker analysis mode*

Description

ROC utilities

Usage

```
SetAnalysisMode(mSetObj, mode)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)s
mode	Input the selected mode for biomarker analysis, "univ" for univariate ROC curve analysis, "explore" for multivariate ROC curve analysis, and "test" for ROC curve based model creation and evaluation. McGill University, Canada License: GNU GPL (¿= 2)

Author(s)

Jeff Xia <jeff.xia@mcgill.ca>

SetAnnotationParam *Set annotation parameters*

Description

This function sets the parameters for peak annotation.

Usage

```
SetAnnotationParam(
  polarity = "positive",
  perc_fwhm = 0.6,
  mz_abs_iso = 0.005,
  max_charge = 2,
  max_iso = 2,
  corr_eic_th = 0.85,
  mz_abs_add = 0.001
)
```

Arguments

polarity	Character, specify the polarity of the MS instrument. Either "negative" or "positive".
perc_fwhm	Numeric, set the percentage of the width of the FWHM for peak grouping. Default is set to 0.6.
mz_abs_iso	Numeric, set the allowed variance for the search (for isotope annotation). The default is set to 0.005.
max_charge	Numeric, set the maximum number of the isotope charge. For example, the default is 2, therefore the max isotope charge is 2+/-.
max_iso	Numeric, set the maximum number of isotope peaks. For example, the default is 2, therefore the max number of isotopes per peaks is 2.
corr_eic_th	Numeric, set the threshold for intensity correlations across samples. Default is set to 0.85.
mz_abs_add	Numeric, set the allowed variance for the search (for adduct annotation). The default is set to 0.001.

Author(s)

Jasmine Chong <jasmine.chong@mail.mcgill.ca>, Mai Yamamoto <yamamoto.mai@mail.mcgill.ca>, and Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

SetCachexiaSetUsed *Set the cachexia set used*

Description

Set cachexia set used

Usage

```
SetCachexiaSetUsed(mSetObj = NA, used)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)
used	Set data to be used

SetCandidate *Set matched name based on user selection from all potential hits*

Description

Note: to change object in the enclosing enviroment, use "¡¡-"

Usage

```
SetCandidate(mSetObj = NA, query_nm, can_nm)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects).
query_nm	Input the query name.
can_nm	Input the candidate name.

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

SetClass *Set class information for MS data*

Description

This function sets the class information for preprocessing MS data.

Usage

```
SetClass(class)
```

Author(s)

Jasmine Chong <jasmine.chong@mail.mcgill.ca>, Mai Yamamoto <yamamoto.mai@mail.mcgill.ca>, and Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`SetCurrentGroups` *To choose from two groups*

Description

Choose two groups (when more than two groups uploaded)

Usage

```
SetCurrentGroups(mSetObj = NA, grps)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)
`grps`	Input the groups

Author(s)

Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`SetCurrentMsetLib` *Set current user selected metset library for search*

Description

if enrichment analysis, also prepare lib by creating a list of metabolite sets

Usage

```
SetCurrentMsetLib(mSetObj=NA, lib.type, excludeNum)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)
`excludeNum`	Users input the mimimum number compounds within selected metabolite sets (metabolitesets ¡ excludeNum)
`lib.type`	Input user selected name of library, "self", "kegg_pathway", "smpdb_pathway", "blood", "urine", "csf", "snp", "predicted", "location", and "drug".

Author(s)

Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

SetCustomData *Set custom data*

Description

The "selected.cmpds" should be for extraction

Usage

```
SetCustomData(mSetObj = NA, selected.cmpds, selected.smpls)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)
selected.cmpds	Input the vector containing the compounds
selected.smpls	Input the vector containing the samples

SetDesignType *For two factor time series only*

Description

For two factor time series only

Usage

```
SetDesignType(mSetObj = NA, design)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)
design	Input the design type

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

SetKEGG.PathLib *Set KEGG pathway library*

Description

note, this process can be long, need to return a value to force Java to wait

Usage

```
SetKEGG.PathLib(mSetObj = NA, libNm, lib.version)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)
lib.version	Input the KEGG pathway version. "current" for the latest KEGG pathway library or "v2018" for the KEGG pathway library version prior to November 2019.
kegg.rda	Input the name of the KEGG library

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

SetMetabolomeFilter *Set metabolome filter*

Description

Set metabolome filter

Usage

```
SetMetabolomeFilter(mSetObj = NA, TorF)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)
TorF	Input metabolome filter

SetMummichogPval *Set the cutoff for mummichog analysis*

Description

Set the p-value cutoff for mummichog analysis.

Usage

```
SetMummichogPval(mSetObj = NA, cutoff)
```

Arguments

mSetObj	Input the name of the created mSetObj.

Author(s)

Jasmine Chong, Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

SetMummichogPvalFromPercent
Set the cutoff for mummichog analysis

Description

Set the p-value cutoff for mummichog analysis.

Usage

```
SetMummichogPvalFromPercent(mSetObj = NA, fraction)
```

Arguments

mSetObj	Input the name of the created mSetObj.

Author(s)

Jasmine Chong, Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`SetOrganism` *Set organism for further analysis*

Description

Set organism for further analysis

Usage

```
SetOrganism(mSetObj = NA, org)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)
`org`	Set organism ID

`SetPeakEnrichMethod` *Set the peak enrichment method for the MS Peaks to Paths module*

Description

This function sets the peak enrichment method.

Usage

```
SetPeakEnrichMethod(mSetObj = NA, algOpt, version = "v2")
```

Arguments

`mSetObj`	Input the name of the created mSetObj.

Author(s)

Jasmine Chong, Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

SetPeakFormat *Set the peak format for the mummichog analysis*

Description

Set the peak format for mummichog analysis.

Usage

```
SetPeakFormat(type)
```

Arguments

mSetObj	Input the name of the created mSetObj.

Author(s)

Jasmine Chong, Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

SetPeakList.GroupValues *Set peak list group values*

Description

Set peak list group values

Usage

```
SetPeakList.GroupValues(mSetObj = NA)
```

Arguments

mSetObj	Input name of mSetObj, the data used is the nmr.xcmsSet object

SetPeakParam *Set parameters for peak profiling and parameters optimization*

Description

This function sets all the parameters used for downstream pre-processing of user's raw MS data based on specific LC-MS platform or parameters optimization. The database will be under an real-time update based on the progress in this field.

Usage

```
SetPeakParam(
  platform = "general",
  Peak_method = "centWave",
  RT_method = "loess",
  mzdiff,
  snthresh,
  bw,
  ppm,
  min_peakwidth,
  max_peakwidth,
  noise,
  prefilter,
  value_of_prefilter,
  fwhm,
  steps,
  sigma,
  profStep,
  minFraction,
  minSamples,
  maxFeatures,
  max,
  extra,
  span,
  smooth,
  family,
  fitgauss,
  verbose.columns,
  mzCenterFun,
  integrate,
  ...
)
```

Arguments

platform	Character, specify the LC-MS platform used in pratice, including "UPLC-Q/E","UPLC-Q/TOF","UPLC-T/TOF","UPLC-Ion_Trap","UPLC-Orbitrap","UPLC-G2S","HPLC-Q/TOF","HPLC-Ion_Trap","HPLC-Orbitrap","HPLC-S/Q".

	Default is "general", which is a more common option for all platform. If the platform is not listed above, please use this one.
Peak_method	Character, specify the algorithm to perform peak detection. "centwave" to use the CentWave algorithm, and "matchedFilter" to use the MatchedFilter algorithm.
RT_method	Character, specify the algorithm to perform tetention time alignment, including "loess" and "obiwarp". Default is "loess".
mzdiff	Numeric, specify the minimum m/z difference for signals to be considered as different features when retention times are overlapping.
snthresh	Numeric, specify the signal to noise threshold.
bw	Numeric, specify the band width (sd or half width at half maximum) of gaussian smoothing kernel to be applied during peak grouping.
ppm	Numeric, specify the mass error in ppm.
min_peakwidth	Numeric, specify the minimum peak width in seconds.Only work for 'centWave'.
max_peakwidth	Numeric, specify the maximum peak width in seconds.Only work for 'centWave'.
noise	Numeric, specify the noise level for peaking picking.Only work for 'centWave'.
prefilter	Numeric, specify the scan number threshold for prefilter.Only work for 'centWave'.
value_of_prefilter	Numeric, specify the scan abundance threshold for prefilter. Only work for 'centWave'.
fwhm	numeric specifying the full width at half maximum of matched filtration gaussian model peak. Only work for 'matchedFilter'.
steps	numeric defining the number of bins to be merged before filtration. Only work for 'matchedFilter'.
sigma	numeric specifying the standard deviation (width) of the matched filtration model peak. Only work for 'matchedFilter'.
profStep	numeric defining the bin size (in mz dimension) to be used for the profile matrix generation. Only work for 'obiwarp'.
minFraction	Numeric, specify fraction of samples in each group that contain the feature for it to be grouped.
minSamples	Numeric, specify minimum number of sample(s) in each group that contain the feature for it to be included.
maxFeatures	Numeric, specify the maximum number of features to be identified.
...	Other parameters, including max,extra,span,smooth,family,fitgauss, verbose.columns,mzCenterFun,integrate. Usually don't need to change.

Author(s)

Zhiqiang Pang <zhiqiang.pang@mail.mcgill.ca>, Jasmine Chong <jasmine.chong@mail.mcgill.ca>, Mai Yamamoto <yamamoto.mai@mail.mcgill.ca>, and Jeff Xia <jeff.xia@mcgill.ca>
McGill University, Canada License: GNU GPL (¿= 2)

`SetPlotParam` *Set generic Plotting Parameters*

Description

This function sets the generic Plotting Parameters for different functions

Usage

```
SetPlotParam(Plot = F, labels = TRUE, format = "png", dpi = 72, width = 9, ...)
```

Arguments

`Plot`	Logical, if true, the function will plot internal figures for different functions.
`labels`	Logical, if true, the labels in the plot will be added.
`format`	Numeric, input the format of the image to create.
`dpi`	Numeric, input the dpi of the image to create.
`width`	Numeric, input the width of the image to create.
`...`	Other specific parameters for specific function. Please set them according to the corresponding function.

Author(s)

Zhiqiang Pang <zhiqiang.pang@mail.mcgill.ca>, and Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`SetSMPDB.PathLib` *Set SMPDB pathway library*

Description

note, this process can be long, need to return a value to force Java to wait

Usage

```
SetSMPDB.PathLib(mSetObj = NA, libNm)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)
`smpdb.rda`	Input the name of the SMPDB library (e.g. hsa or mmu)

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

Setup.AdductData *Save adduct names for mapping*

Description

Save adduct names for mapping

Usage

```
Setup.AdductData(mSetObj = NA, qvec)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)
qvec	Input the vector to query

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

Setup.BiofluidType *Save biofluid type for SSP*

Description

Save biofluid type for SSP

Usage

```
Setup.BiofluidType(mSetObj = NA, type)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)
type	Input the biofluid type

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`Setup.ConcData` *Save concentration data*

Description

Save concentration data

Usage

```
Setup.ConcData(mSetObj = NA, conc)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)
`conc`	Input the concentration data

Author(s)

Jeff Xia<`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`Setup.HMDBReferenceMetabolome` *Read user uploaded metabolome as a list of HMDB compound names*

Description

Read user uploaded metabolome as a list of HMDB compound names

Usage

```
Setup.HMDBReferenceMetabolome(mSetObj = NA, filePath)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)
`filePath`	Input the path to the user's list of HMDB compound names

Author(s)

Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`Setup.KEGGReferenceMetabolome` *Read user uploaded metabolome as a list of KEGG pathway ids*

Description

Read user uploaded metabolome as a list of KEGG pathway ids

Usage

```
Setup.KEGGReferenceMetabolome(mSetObj = NA, filePath)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)
`filePath`	Input the path to the user's list of KEGG pathway ids

Author(s)

Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`Setup.MapData` *Save compound name for mapping*

Description

Save compound name for mapping

Usage

```
Setup.MapData(mSetObj = NA, qvec)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)
`qvec`	Input the vector to query

Author(s)

Jeff Xia<`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`Setup.UserMsetLibData` *Read user upload metabolite set library file*

Description

Return two col csv file, first name, second cmpd list

Usage

```
Setup.UserMsetLibData(mSetObj = NA, filePath)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)
`filePath`	Input the path to the user's uploaded metabolite set library

Author(s)

Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`SetupKEGGLinks` *Only works for human (hsa.rda) data*

Description

Only works for human (hsa.rda) data 2018 - works for ath, eco, mmu, sce

Usage

```
SetupKEGGLinks(smpdb.ids)
```

Arguments

`kegg.ids`	Input the list of KEGG ids to add SMPDB links

Author(s)

Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`SetupMSdataMatrix` *Create a MS spectra data matrix of peak values for each group*

Description

This function sets up a MS spectra data matrix using the 'groupval' function from XCMS. This generates a usable matrix of peak values for analysis where columns represent peak groups and rows represent samples. Collisions where more than one peak from a single sample are in the same group get resolved utilizing "medret", which uses the peak closest to the median retention time.

Usage

```
SetupMSdataMatrix(mSetObj, intvalue)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)
`intvalue`	name of peak column to enter into the returned matrix, if intvalue = 'into', use integrated area of original (raw) peak intensities, if intvalue = 'intf', use integrated area of filtered peak intensities, if intvalue = 'intb', use baseline corrected integrated peak intensities, if intvalue = 'maxo', use the maximum intensity of original (raw) peaks, or if intvalue = 'maxf' use the maximum intensity of filtered peaks

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`SetupSMPDBLinks` *Only works for human (hsa.rda) data*

Description

Only works for human (hsa.rda) data 2018 - works for ath, eco, mmu, sce

Usage

```
SetupSMPDBLinks(kegg.ids)
```

Arguments

`kegg.ids`	Input the list of KEGG ids to add SMPDB links

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

SlaveCluster_doe *Core Peak Picking Slave Cluster*

Description

Core Peak Picking Slave Cluster

Usage

```
SlaveCluster_doe(
  task,
  Set_parameters,
  object,
  object_mslevel,
  isotopeIdentification,
  BPPARAM = bpparam(),
  ...
)
```

Arguments

task	Numeric, task order for XCMS paramters table to run the peak picking and alignment.
object_mslevel	List, the parsed metabolomics scans produced by PeakPicking_prep.
isotopeIdentification	Character, IsotopeIdentidication method, usually includes 'IPO' and 'CAMERA'.
BPPARAM	MulticoreParam method, used to set the parallel method. Default is bpparam().
xcmsSet_parameters	Matrix, the parameters combination produced automatically according to the primary parameters input.
MSnExp	object, the trimmed or the original data.

Author(s)

Zhiqiang Pang <zhiqiang.pang@mail.mcgill.ca> Jeff Xia <jeff.xia@mcgill.ca> Mcgill University License: GNU GPL (¿= 2)

`SOM.Anal` *SOM analysis*

Description

SOM analysis

Usage

```
SOM.Anal(mSetObj = NA, x.dim, y.dim, initMethod, neigb = "gaussian")
```

Arguments

`mSetObj`	Input name of the created mSet Object
`x.dim`	Input X dimension for SOM analysis
`y.dim`	Input Y dimension for SOM analysis
`initMethod`	Input the method
`neigb`	Default is set to 'gaussian'

Author(s)

Jeff Xia<`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`sparse.mint.block_iteration`
Perform Sparse Generalized Canonical Correlation (sgccak)

Description

Runs sgccak() modified from RGCCA

Usage

```
sparse.mint.block_iteration(
  A,
  design,
  study = NULL,
  keepA.constraint = NULL,
  keepA = NULL,
  scheme = "horst",
  init = "svd",
  max.iter = 100,
  tol = 1e-06,
  verbose = TRUE,
  bias = FALSE,
  penalty = NULL
)
```

Arguments

A	Data
design	Set design
study	Default set to NULL
keepA.constraint	Default set to NULL
keepA	Default set to NULL
scheme	Scheme, default set to "horst"
init	Init mode, default set to "svd"
max.iter	Max number of iterations, numeric, default set to 100
tol	Tolerance, numeric, default set to 1e-06
verbose	Default set to TRUE
bias	Default set to FALSE
penalty	Default set to NULL

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

---

splsda *Perform sPLS-DA*

Description

Sparse PLS functions (adapted from mixOmics package for web-based usage) this function is a particular setting of internal_mint.block the formatting of the input is checked in internal_wrapper.mint

Usage

```
splsda(
  X,
  Y,
  ncomp = 2,
  mode = c("regression", "canonical", "invariant", "classic"),
  keepX,
  keepX.constraint = NULL,
  scale = TRUE,
  tol = 1e-06,
  max.iter = 100,
  near.zero.var = FALSE,
  logratio = "none",
  multilevel = NULL
)
```

Arguments

X	numeric matrix of predictors
Y	a factor or a class vector for the discrete outcome
ncomp	the number of components to include in the model. Default to 2.
mode	Default set to c("regression", "canonical", "invariant", "classic")
keepX	Number of X variables kept in the model on the last components (once all keepX.constraint[[i]] are used).
keepX.constraint	A list containing which variables of X are to be kept on each of the first PLS-components.
scale	Boleean. If scale = TRUE, each block is standardized to zero means and unit variances (default: TRUE).
tol	Convergence stopping value.
max.iter	integer, the maximum number of iterations.
near.zero.var	boolean, see the internal nearZeroVar function (should be set to TRUE in particular for data with many zero values). Setting this argument to FALSE (when appropriate) will speed up the computations
logratio	"None" by default, or "CLR"
multilevel	Designate multilevel design, "NULL" by default

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

SPLSR.Anal *Perform SPLS-DA*

Description

Sparse PLS-DA (from mixOmics)

Usage

```
SPLSR.Anal(mSetObj = NA, comp.num, var.num, compVarOpt, validOpt = "Mfold")
```

Arguments

mSetObj	Input name of the created mSet Object
comp.num	Input the number of computations to run
var.num	Input the number of variables
compVarOpt	Input the option to perform SPLS-DA

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

ssm_trim *Standards Simulation Method*

Description

Whole mass spectra will be divided as 4 bins according to the mz range. Trimming the raw with slide window method in every bins and retained the windows with highest scan intensity and remove other scan signal in mz dimension. Then the data will be trimed again in the RT dimension with slide window method. The window with highest intensity scans will be kept. After the timming alongside mz and RT dimension, the peaks not only the high intensity peaks, but also the relatively low intensity peaks will also be retained as the 'simulated standards' data for parameters optimization.

Usage

```
ssm_trim(raw_data, ms_list, rt.idx)
```

Arguments

raw_data	MSnExp object, the raw data that has been read in memory.
ms_list	List, the names list of all scans.
rt.idx	Numeric, the retention time percentage, from 0 to 1. Default is 1/15.

Author(s)

Zhiqiang Pang <zhiqiang.pang@mail.mcgill.ca> Jeff Xia <jeff.xia@mcgill.ca> Mcgill University License: GNU GPL (¿= 2)

Statistic_doe *Analyze DoE Result*

Description

Analyze DoE Result

Usage

```
Statistic_doe(
  object,
  object_mslevel,
  isotopeIdentification,
  BPPARAM = bpparam(),
  mSet_OPT,
  subdir = NULL,
  plot = F,
```

```
  iterator,
  index.set,
  useNoise
)
```

Arguments

object	MSnExp object, the trimmed or the original data.
object_mslevel	List, the parsed metabolomics scans produced by PeakPicking_prep.
isotopeIdentification	Character, IsotopeIdentidication method, usually includes 'IPO' and 'CAMERA'.
BPPARAM	MulticoreParam method, used to set the parallel method. Default is bpparam().
mSet_OPT	List, the result produced by 'ExperimentsCluster'.
subdir	Logical, weather to creat a sub-directory (if true) or not (if false).
plot	Logical, weather to plot the Contours plots of the DoE results.
iterator	Numeric, the round number of the DoE.
index.set	List, the indexes set (including PPS, CV, RCS, GS and Gaussian Index) produced by ExperiemntCluster.
useNoise	Numeric, the noise level removed to evalute the gaussian peak.

Author(s)

Zhiqiang Pang <zhiqiang.pang@mail.mcgill.ca> Jeff Xia <jeff.xia@mcgill.ca> Mcgill University License: GNU GPL (¿= 2)

SumNorm *Row-wise Normalization*

Description

Row-wise norm methods, when x is a row. Normalize by a sum of each sample, assume constant sum (1000). Options for normalize by sum median, reference sample, reference reference (compound), or quantile normalization

Usage

```
SumNorm(x)
```

Arguments

x	Input data to normalize

Author(s)

Jeff Xia <jeff.xia@mcgill.ca> McGill University, Canada

template.match *Pattern hunter*

Description

Run template on all the high region effect genes

Usage

```
template.match(x, template, dist.name)
```

Arguments

x	Input data
template	Input template
dist.name	Input distance method

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

Ttests.Anal *Perform t-test analysis*

Description

This function is used to perform t-test analysis.

Usage

```
Ttests.Anal(
  mSetObj = NA,
  nonpar = F,
  threshp = 0.05,
  paired = FALSE,
  equal.var = TRUE,
  all_results = FALSE
)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)
nonpar	Logical, use a non-parametric test, T or F. False is default.
threshp	Numeric, enter the adjusted p-value (FDR) cutoff
paired	Logical, is data paired (T) or not (F).
equal.var	Logical, evaluates if the group variance is equal (T) or not (F).
all_results	Logical, if TRUE, returns T-Test analysis results for all compounds.

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

UnzipUploadedFile *Unzip .zip files*

Description

Unzips uploaded .zip files, removes the uploaded file, checks for success

Usage

```
UnzipUploadedFile(inPath, outPath, rmFile = T)
```

Arguments

inPath	Input the path of the zipped files
outPath	Input the path to directory where the unzipped files will be deposited
rmFile	Logical, input whether or not to remove files. Default set to T

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

UpdateData *Update data for filtering*

Description

Function to update the mSetObj after removing features or samples.

Usage

```
UpdateData(mSetObj = NA)
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)

`UpdateEC_Rules` *Update the mSetObj with user-selected parameters for MS Peaks to Pathways.*

Description

This functions handles updating the mSet object for mummichog analysis.

Usage

```
UpdateEC_Rules(mSetObj = NA, force_primary_ion, rt_tol)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects).
`force_primary_ion`	Character, if "yes", only mz features that match compounds with a primary ion are kept.
`rt_tol`	Numeric. Input the retention time tolerance used for determining ECs (in seconds).

Author(s)

Jasmine Chong, Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`UpdateGraphSettings` *Update graph settings*

Description

Function to update the graph settings.

Usage

```
UpdateGraphSettings(mSetObj = NA, colVec, shapeVec)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects)

`UpdateInstrumentParameters`

Update the mSetObj with user-selected parameters for MS Peaks to Pathways.

Description

This functions handles updating the mSet object for mummichog analysis. It is necessary to utilize this function to specify to the organism's pathways to use (libOpt), the mass-spec mode (msModeOpt) and mass-spec instrument (instrumentOpt).

Usage

```
UpdateInstrumentParameters(mSetObj=NA, instrumentOpt, msModeOpt, custom=FALSE)
```

Arguments

`mSetObj`	Input the name of the created mSetObj (see InitDataObjects).
`instrumentOpt`	Numeric. Define the mass-spec instrument used to perform untargeted metabolomics.
`msModeOpt`	Character. Define the mass-spec mode of the instrument used to perform untargeted metabolomics.
`force_primary_ion`	Character, if "yes", only mz features that match compounds with a primary ion are kept.
`custom`	Logical, select adducts for mummichog to consider.

Author(s)

Jasmine Chong, Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`UpdateIntegPathwayAnalysis`

Update integrative pathway analysis for new input list

Description

used for integrative analysis as well as general pathways analysis for meta-analysis results

Usage

```
UpdateIntegPathwayAnalysis(mSetObj=NA, qids, file.nm, topo="dc", enrich="hyper", libOpt="integ")
```

Arguments

`mSetObj`	Input name of the created mSet Object
`qids`	Input the query IDs
`file.nm`	Input the name of the file
`topo`	Select the mode for topology analysis: Degree Centrality ("dc") measures the number of links that connect to a node (representing either a gene or metabolite) within a pathway; Closeness Centrality ("cc") measures the overall distance from a given node to all other nodes in a pathway; Betweenness Centrality ("bc")measures the number of shortest paths from all nodes to all the others that pass through a given node within a pathway.
`enrich`	Method to perform over-representation analysis (ORA) based on either hypergenometrics analysis ("hyper") or Fisher's exact method ("fisher").
`libOpt`	Select the different modes of pathways, either the gene-metabolite mode ("integ") which allows for joint-analysis and visualization of both significant genes and metabolites or the gene-centric ("genetic") and metabolite-centric mode ("metab") which allows users to identify enriched pathways driven by significant genes or metabolites, respectively.

Author(s)

Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

`UpdateOPLS.Splot` *Update OPLS loadings*

Description

Update the OPLS loadings

Usage

```
UpdateOPLS.Splot(mSetObj = NA, plotType)
```

Arguments

`mSetObj`	Input name of the created mSet Object
`plotType`	Set annotation type, "all" to label all variables and "none" to label no variables.

Author(s)

Jeff Xia<`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

UpdatePCA.Loading *Update PCA loadings*

Description

Update the PCA loadings

Usage

```
UpdatePCA.Loading(mSetObj = NA, plotType)
```

Arguments

mSetObj	Input name of the created mSet Object
plotType	Set annotation type, "all" to label all variables and "none" to label no variables.

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

UpdatePLS.Loading *Update PLS loadings*

Description

Update the PLS loadings

Usage

```
UpdatePLS.Loading(mSetObj = NA, plotType)
```

Arguments

mSetObj	Input name of the created mSet Object
plotType	Set annotation type, "all" to label all variables and "none" to label no variables.

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

`updateRawSpectraParam` *updateRawSpectraParam*

Description

updateRawSpectraParam

Usage

```
updateRawSpectraParam(Params)
```

Arguments

`Params`	object generated by SetPeakParams function.

Author(s)

Zhiqiang Pang, Jeff Xia <`jeff.xia@mcgill.ca`> McGill University, Canada

`usr2png` *Perform utilities for MetPa*

Description

Convert user coords (as used in current plot) to pixels in a png adapted from the imagemap package

Usage

```
usr2png(xy, im)
```

Arguments

`xy`	Input coordinates
`im`	Input coordinates

Author(s)

Jeff Xia<`jeff.xia@mcgill.ca`> McGill University, Canada License: GNU GPL (¿= 2)

Volcano.Anal *Perform Volcano Analysis*

Description

Perform volcano analysis

Usage

```
Volcano.Anal(mSetObj=NA, paired=FALSE, fcthresh, cmpType, percent.thresh, nonpar=F, threshp, equal.var
```

Arguments

mSetObj	Input the name of the created mSetObj (see InitDataObjects)
paired	Logical, T if data is paired, F if data is not.
fcthresh	Numeric, input the fold change threshold
cmpType	Comparison type, 1 indicates group 1 vs group 2, and 2 indicates group 2 vs group 1
percent.thresh	Only for paired data, numeric, indicate the significant count threshold
nonpar	Logical, indicate if a non-parametric test should be used (T or F)
threshp	Numeric, indicate the p-value threshold
equal.var	Logical, indicates if the group variance is equal (T) or unequal (F)
pval.type	To indicate raw p-values, use "raw". To indicate FDR-adjusted p-values, use "fdr".

Author(s)

Jeff Xia<jeff.xia@mcgill.ca> McGill University, Canada License: GNU GPL (¿= 2)

XSet2MSet *Converts xset object from XCMS to mSet object for MetaboAnalyst*

Description

This function converts processed raw LC/MS data from XCMS to a usable data object (mSet) for MetaboAnalyst. The immediate next step following using this function is to perform a SanityCheck, and then further data processing and analysis can continue.

Usage

```
XSet2MSet(xset, dataType, analType, paired = F, format, lbl.type)
```

Arguments

`xset`	The name of the xcmsSet object created.
`dataType`	The type of data, either list (Compound lists), conc (Compound concentration data), specbin (Binned spectra data), pktable (Peak intensity table), nmrpeak (NMR peak lists), mspeak (MS peak lists), or msspec (MS spectra data).
`analType`	Indicate the analysis module to be performed: stat, pathora, pathqea, msetora, msetssp, msetqea, ts, cmpdmap, smpmap, or pathinteg.
`paired`	Logical, is data paired (T) or not (F).
`format`	Specify if samples are paired and in rows (rowp), unpaired and in rows (rowu), in columns and paired (colp), or in columns and unpaired (colu).
`lbl.type`	Specify the data label type, either discrete (disc) or continuous (cont).

Index