Package 'mbQTL'

July 10, 2025

Type Package

Title mbQTL: A package for SNP-Taxa mGWAS analysis

Version 1.8.0

Description mbQTL is a statistical R package for simultaneous 16srRNA,16srDNA (microbial) and variant, SNP, SNV (host) relationship, correlation, regression studies.
We apply linear, logistic and correlation based statistics to identify the relationships of taxa, genus, species and variant, SNP, SNV in the infected host. We produce various statistical significance measures such as P values, FDR, BC and probability estimation to show significance of these relationships. Further we provide various visualization function for ease and clarification of the results of these analysis. The package is compatible with dataframe, MRexperiment and text formats.

License MIT + file LICENSE

Encoding UTF-8

Depends R (>= 4.3.0)

DeploySubPath mbQTL

biocViews SNP, Microbiome, WholeGenome, Metagenomics, StatisticalMethod, Regression

Roxygen list(markdown = TRUE)

RoxygenNote 7.2.1

Imports MatrixEQTL, dplyr, ggplot2, readxl, stringr, tidyr, metagenomeSeq, pheatmap, broom, graphics, stats, methods

Suggests knitr, rmarkdown, BiocStyle

VignetteBuilder knitr

URL ``https://github.com/Mercedeh66/mbQTL''

BugReport ``https://github.com/Mercedeh66/mbQTL/issues"

git_url https://git.bioconductor.org/packages/mbQTL

git_branch RELEASE_3_21

git_last_commit 226f80d

git_last_commit_date 2025-04-15

Repository Bioconductor 3.21

Date/Publication 2025-07-09

Author Mercedeh Movassagh [aut, cre] (ORCID:
<https://orcid.org/0000-0001-7690-0230>),
Steven Schiff [aut],
Joseph N Paulson [aut]

Maintainer Mercedeh Movassagh <mercedeh.movassagh@yale.edu>

Contents

`mbQTL-package` *title* `mbQTL` *is a package for microbial QTL/GWAS Analysis*

Description

This package provides statistical methods for identifying significant relationships between microbial/taxa and genetic SNP signatures. We use three models 1) linear regression between all taxa-snp. Main function is `linearTaxaSnp()`. 2) Correlation analysis between taxa-snp across all taxa and snps. Main function is `taxaSnpCor()` and 3) Logistic regression analysis between each taxa and each snp simultaneously or for a specific cases. Main function is `logRegSnpsTaxa()`.

Author(s)

Maintainer: Mercedeh Movassagh <mercedeh.movassagh@yale.edu> (ORCID)

Authors:

- Steven Schiff
- Joseph N Paulson

See Also

The package vignette can be accessed with `vignette("mbQTL")`.

`allToAllProduct` `allToAllProduct` *creates a dataframe of snp and taxa correlations*

Description

This internal function takes the original snp dataframe and returns a long parsed snp dataframe

Usage

```
allToAllProduct(SnpFile, microbeAbund, rsID = NULL)
```

Arguments

`SnpFile`	the snp file (rownames is sample number and colnames is the snps)
`microbeAbund`	he taxa abundance dataframe (rownames sample names and colnames taxa Genus/species/family)
`rsID`	Default is NULL and will run across the who dataset unless specific rsID/SNP/chr_region is specified

Value

A dataframe of correlations between snps and taxa

Examples

```
data(microbeAbund)
data(SnpFile)
x <- allToAllProduct(SnpFile, microbeAbund, "chr1.171282963_T")
```

`binarizeMicrobe` `binarizeMicrobe` *binarizes microbe abundace file based on user's cutoff*

Description

This function creates a dataframe output produces a formatted dataframe prepared.

Usage

```
binarizeMicrobe(microbeAbund, cutoff = NULL, selectmicrobe = NULL)
```

Arguments

microbeAbund	the taxa abundance dataframe (rownames sample names and colnames taxa Genus/species/family)
cutoff	cutoff at which the user chose to call taxa positive or negative across samples (should be a numeric value for normalized or count values).
selectmicrobe	default is and all taxa are considered at the same time, if the user is interested in a specific pathogen use name of the pathogen for example "Haemophilus".

Value

A data frame of microbial abundance.

---

coringTaxa — coringTaxa *creates correlation dataframe for taxa*

Description

This function creates an output correlation data frame for all microbial taxa (or other organisms such as viral or parasitic taxa)

Usage

```
coringTaxa(microbeAbund)
```

Arguments

microbeAbund	the taxa abundance dataframe (rownames sample names and colnames taxa Genus/species/family)

Value

A data frame of correlations between taxa

Examples

```
data(microbeAbund)
x <- coringTaxa(microbeAbund)
```

CovFile — mbQTL *"CovFile"*

Description

The "CovFile" is the covariate file for linear regression option of taxa and snp association. The covariance file is generated randomly by assigning sex and site of collection to the samples.) rownames are covariate and colnames samples.

`histPvalueLm` *histPvalueLm a histogram of Taxa and SNP linear regression analysis. This function creates a histogram object of all SNPs with all taxa Linear regression analysis p values.*

Description

histPvalueLm a histogram of Taxa and SNP linear regression analysis. This function creates a histogram object of all SNPs with all taxa Linear regression analysis p values.

Usage

```
histPvalueLm(LinearAnalysisTaxaSNP)
```

Arguments

`LinearAnalysisTaxaSNP`
: the data frame result created from the `linearTaxaSnp()` function.

Value

A histogram object of p values observed from taxa and SNP Linear Regression analysis.

Examples

```
data(microbeAbund)
data(microbeAbund)
data(SnpFile)
data(CovFile)
LinearAnalysisTaxaSNPFile <- linearTaxaSnp(microbeAbund, SnpFile, Covariate = CovFile)
x <- histPvalueLm(LinearAnalysisTaxaSNPFile)
```

`linearTaxaSnp` *linearTaxaSnp Performs linear regression analysis between taxa and SNPs and returns concordance statistics*

Description

This function creates a dataframe output from the results all snps with all taxa linear regression analysis of all snps in the dataset. The result is a dataframe with P values and FDRs of all regressions. MatrixeQTL core functions are utilized to achieve this. Note the main functions used are Matrix_eQTL_engine() assuming linear regression with or without a covariate file.

Usage

```
linearTaxaSnp(microbeAbund, SnpFile, Covariate = NULL)
```

Arguments

microbeAbund	the taxa abundance dataframe (rownames sample names and colnames taxa Genus/species/family)
SnpFile	the snp dataframe (values 0,1,2 indicating zygosity), rownames sample names and colnames snp names.
Covariate	default is NULL, hence assumed non-existent. If covariates are available they need to be formatted in the CovFile format, that is colnames are sample numbers matching samples in the microbe abundance and snp file and row names are the co-variates names (such as sex, disease etc).

Value

A data frame which is a result of Linear Regression of all snp, taxa relationships, with P values and P value corrected values.

Examples

```
data(microbeAbund)
data(SnpFile)
data(CovFile)
x <- linearTaxaSnp(microbeAbund, SnpFile, Covariate = CovFile)
```

---

logitPlotSnpTaxa *logitPlotSnpTaxa produces bar plots for counts of ref vs alt vs het allells for particular rsID taxa combinations*

Description

This function creates a dataframe output produces a formatted dataframe prepared.

Usage

```
logitPlotSnpTaxa(
  microbeAbund,
  SnpFile,
  selectmicrobe = NULL,
  rsID,
  ref = NULL,
  alt = NULL,
  het = NULL,
  color = NULL,
  cutoff = NULL
)
```

Arguments

microbeAbund	original microbe abundance file (colnames microbe, rownames= sample IDs)
SnpFile	original snp file with (0,1,2 values for ref, het, alt genotypes), colnames SNP names, rownames, sample IDs.
selectmicrobe	name of the microbe of interest (for example individual significant microbes associate with a snp).
rsID	name of the snp of interest (for example individual significant snps associated with a microbe)
ref	the name of reference genotype for example "GG"
alt	the name of snp (variant) genotype for example "AA"
het	the name of hetrozygote genotype for example "GA"
color	the default is NULL and the color is set to c("#ffaa1e", "#87365b").
cutoff	cutoff at which we call microbe present or absent

Value

A bar ggplot comparing the counts of ref vs alt vs het genotype

Examples

```
data(microbeAbund)
data(SnpFile)
x <- logitPlotSnpTaxa(microbeAbund, SnpFile,
  selectmicrobe = "Neisseria", rsID = "chr2.241072116_A",
  ref = NULL, alt = NULL, het = NULL, color = NULL, cutoff = NULL
)
```

---

logRegSnpsTaxa — logRegSnpsTaxa *Performs logistic regression analysis between taxa and SNPs and returns concordance statistics*

Description

This function creates a dataframe output from the results of either a unique taxa and all snps or all taxa and all snps in the dataset. The result is a dataframe with P values and FDRs of all regressions.

Usage

```
logRegSnpsTaxa(microbeAbund, SnpFile, cutoff = NULL, selectmicrobe = NULL)
```

Arguments

microbeAbund	the taxa abundance dataframe (rownames sample names and colnames taxa Genus/species/family)
SnpFile	the snp dataframe (values 0,1,2 indicating zygosity), rownames sample names and colnames snp names.
cutoff	default is NULL, hence anything above cutoff is considered positive, the cut-off at which the specific or all taxa are considered positive for the pathogen (1 indicates positive and 0 negative).
selectmicrobe	default is and all taxa are considered at the same time, if the user is interested in a specific pathogen use name of the pathogen for example "Haemophilus".

Value

A data frame which is a result of Logistic regression products of individual snp, taxa relationships, with P values and P value corrected values (FDR, Bonferroni).

Examples

```
data(microbeAbund)
data(SnpFile)
x <- logRegSnpsTaxa(microbeAbund, SnpFile, selectmicrobe = c("Haemophilus"))
```

---

mbQtlCorHeatmap mbQtlCorHeatmap *for making heatmap for snp, taxa rho values*

Description

This function produces a log heatmap +1 of the correlation rho values across snp, taxa datasets

Usage

```
mbQtlCorHeatmap(final_var_long, labels_col = NULL, ...)
```

Arguments

final_var_long	the long data frame of rho values created by the taxaSnpCor() function.
labels_col	set to NULL ass default if TRUE, labels will appear on the heatmap.
...	all other parameters for pheatmap.

Value

A data frame of correlations between taxa

Examples

```
data(microbeAbund)
data(SnpFile)
for_all_rsids <- allToAllProduct(SnpFile, microbeAbund)
correlationMicrobes <- coringTaxa(microbeAbund)
taxaSnpCor(for_all_rsids, correlationMicrobes)
final_var_long <- taxaSnpCor(for_all_rsids, correlationMicrobes, probs = c(0.0001, 0.9999))
x <- mbQtlCorHeatmap(final_var_long)
```

`metagenomeSeqObj` *mbQTL "metagenomeSeqObj" "MetagenomSeqObj" is an* `MRexperiment` *object format of the "microbeAbund" file.*

Description

`mbQTL` "metagenomeSeqObj"

"MetagenomSeqObj" is an `MRexperiment` object format of the "microbeAbund" file.

`metagenomeSeqToMbqtl` *%*

Written by Mercedeh Movassagh Rhrefmailto:mercedeh@ds.dfci.harvard.edumercedeh@ds.dfci.harvard.edu, January 2023

metagenomeSeqToMbqtl Converts metagenomeSeq obj to compatible taxa dataframe

---

Description

This function takes and `MRexperiement` class object transforms it and makes the result dataframe compatible with `mbQTL` taxa input file

Usage

```
metagenomeSeqToMbqtl(meta_glom, norm, log, aggregate_taxa = NULL)
```

Arguments

`meta_glom`	`MRexperiement` class obj from `metagenomeSeq` package.
`norm`	A logical indicating whether or not to return normalized counts.
`log`	TRUE/FALSE whether or not to log2 transform scale.
`aggregate_taxa`	it is recommended that the normalization occurs at taxa level (default NULL) however, if the user chooses to aggregate on the phyla/family/Genus or Species level before normalization they have the option.

Value

A data frame of normalized/not normalized counts compatible with mbQTL.

Examples

```
data(metagenomeSeqObj)
x <- metagenomeSeqToMbqtl(metagenomeSeqObj, norm = TRUE, log = TRUE, aggregate_taxa = NULL)
```

---

microbeAbund mbQTL *"microbiomeAbund" File*

Description

This is the microbial Abudnance file generated from 16s it is either this file or the "metaGenomeSeqObj".The "microbiomeAbund" file is a randomly generated file format for total microbe presence (number of reads)(parasite/viral transcripts) for specific species.This could be generated from select taxa results from biom() or MRexperiment objects as long as the samples are colnames and taxa are rownames.

---

prepareCorData prepareCorData *prpares and joins snp-taxa and taxa-taxa correlation file.*

Description

This function creates a dataframe output produces a formatted dataframe prepared.

Usage

```
prepareCorData(microbeAbund, SnpFile, cutoff = NULL, selectmicrobe = NULL)
```

Arguments

microbeAbund	the taxa abundance dataframe (rownames sample names and colnames taxa Genus/species/family)
SnpFile	the snp dataframe (values 0,1,2 indicating zygosity), rownames sample names and colnames snp names.
cutoff	default is NULL, hence anything above cutoff is considered positive, the cutoff at which the specific or all taxa are considered positive for the pathogen (1 indicates positive and 0 negative).
selectmicrobe	default is and all taxa are considered at the same time, if the user is interested in a specific pathogen use name of the pathogen for example "Haemophilus".

Value

A data frame which is a result of Logistic regression products of individual snp, taxa relationships, with P values and P value corrected values.

qqPlotLm *qqPlotLm creates QQ-Plot of all SNPs with all taxa Linear regression analysis This function creates QQ-Plot object of all SNPs with all taxa Linear regression analysis of expected versus observed P values*

Description

qqPlotLm creates QQ-Plot of all SNPs with all taxa Linear regression analysis This function creates QQ-Plot object of all SNPs with all taxa Linear regression analysis of expected versus observed P values

Usage

```
qqPlotLm(microbeAbund, SnpFile, Covariate = NULL)
```

Arguments

microbeAbund	the taxa abundance dataframe (rownames sample names and colnames taxa Genus/species/family)
SnpFile	the snp dataframe (values 0,1,2 indicating zygosity), rownames sample names and colnames snp names.
Covariate	default is NULL, hence assumed non-existent. If covariates are available they need to be formatted in the CovFile format, that is colnames are sample numbers matching samples in the microbe abundance and snp file and row names are the covariates names (such as sex, disease etc).

Value

A QQplot object of expected versus obsesrved taxa and SNP Linear Regression analysis

Examples

```
data(microbeAbund)
data(SnpFile)
data(CovFile)
x <- qqPlotLm(microbeAbund, SnpFile, Covariate = CovFile)
```

RegSnp RegSnp *creates a dataframe of parsed long snp files*

Description

This internal function takes the orginal snp dataframe and retruns a long parsed snp dataframe

Usage

```
RegSnp(SnpFile, microbeAbund)
```

Arguments

SnpFile	the snp file (rownames is sample number and colnames is the snps)
microbeAbund	the microbial abundance file (rownames is sample number and colnames is the microbe)

Value

A long parsed datframe of snps

SnpFile — mbQTL *"SnpFile"*

Description

The "SnpFile" is randomly generated from GATK (Van der Auwera GA & O'Connor BD. (2020). Genomics in the Cloud: Using Docker, GATK, and WDL in Terra (1st Edition). O'Reilly Media) snp calls followed by plink (Purcell S, et al. (2007) PLINK: a toolset for whole-genome association and population-based linkage analysis. American Journal of Human Genetics) processing and it needs to be in (0,1,2) format representing the zygosity of the snps.

---

taxaSnpCor — taxaSnpCor *for estimation of the rho value between snp, taxa correlations across datasets*

Description

This function produces a log heatmap +1 of the correlation rho values across snp, taxa dataframe.

Usage

```
taxaSnpCor(for_all_rsids, correlationMicrobes, probs = NULL)
```

Arguments

for_all_rsids	A dataframe result of correlation analysis between the snps and taxa dataframe, an output of allToAllProduct() function.
correlationMicrobes	A dataframe of correlation between coringTaxa() function.
probs	Default is NULL if other that all rho values are wanted the value can be subseted using c(x,y).

Value

A data frame of correlations between taxa

Examples

```
data(microbeAbund)
data(SnpFile)

for_all_rsids <- allToAllProduct(SnpFile, microbeAbund)
correlationMicrobes <- coringTaxa(microbeAbund)
x <- taxaSnpCor(for_all_rsids, correlationMicrobes)
```

Index