Package ‘getmstatistic’

May 9, 2021

**Title**  Quantifying Systematic Heterogeneity in Meta-Analysis

**Version**  0.2.2

**Description**  Quantifying systematic heterogeneity in meta-analysis using R. The M statistic aggregates heterogeneity information across multiple variants to, identify systematic heterogeneity patterns and their direction of effect in meta-analysis. It’s primary use is to identify outlier studies, which either show “null” effects or consistently show stronger or weaker genetic effects than average across, the panel of variants examined in a GWAS meta-analysis. In contrast to conventional heterogeneity metrics (Q-statistic, I-squared and tau-squared) which measure random heterogeneity at individual variants, M measures systematic (non-random) heterogeneity across multiple independently associated variants. Systematic heterogeneity can arise in a meta-analysis due to differences in the study characteristics of participating studies. Some of the differences may include: ancestry, allele frequencies, phenotype definition, age-of-disease onset, family-history, gender, linkage disequilibrium and quality control thresholds. See <https://magosil86.github.io/getmstatistic/> for statistical statistical theory, documentation and examples.

**Depends**  R (>= 3.1.0)

**License**  MIT + file LICENSE

**URL**  https://magosil86.github.io/getmstatistic/

**BugReports**  https://github.com/magosil86/getmstatistic/issues

**LazyData**  true

**Imports**  ggplot2 (>= 1.0.1), gridExtra (>= 0.9.1), gtable (>= 0.1.2), metafor (>= 1.9-6), psych (>= 1.5.1), stargazer (>= 5.1)

**Suggests**  foreign (>= 0.8-62), knitr (>= 1.10.5), testthat, covr, rmarkdown

**RoxygenNote**  7.1.1

**VignetteBuilder**  knitr

**NeedsCompilation**  no
draw_table

Author  Lerato E Magosi [aut],
        Jemma C Hopewell [aut],
        Martin Farrall [aut],
        Lerato E Magosi [cre]

Maintainer  Lerato E Magosi <magosil86@gmail.com>

Repository  CRAN

Date/Publication  2021-05-09 05:10:15 UTC

R topics documented:

<table>
<thead>
<tr>
<th>Function</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>draw_table</td>
<td>2</td>
</tr>
<tr>
<td>getmstatistic</td>
<td>3</td>
</tr>
<tr>
<td>heartgenes214</td>
<td>7</td>
</tr>
</tbody>
</table>

Index 9

---

draw_table  Helper function to draw table grobs.

Description

draw_table() Pre and post version: 2.0.0 gridExtra packages handle drawing tables differently. draw_table() determines the installed version of gridExtra and applies the appropriate syntax. If gridExtra version < 2.0.0 then it uses old gridExtra syntax to build table Grob(graphical object) else uses new syntax. draw_table()

Usage

draw_table(body, heading, ...)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>body</td>
<td>A dataframe. Table body.</td>
</tr>
<tr>
<td>heading</td>
<td>A string. Table title.</td>
</tr>
<tr>
<td>...</td>
<td>Further arguments to control the gtable.</td>
</tr>
</tbody>
</table>

Details

prints tables without rownames.

Acknowledgements

Thanks to Ryan Welch, https://github.com/welchr/LocusZoom/issues/16
getmstatistic

Examples

library(gridExtra)

## Not run:

# Table of iris values
iris_dframe <- head(iris)
title_iris_dframe <- paste("Table: Length and width measurements (cm) of sepals and petals,"
  "for 50 flowers from 3 species of iris (setosa, versicolor,"
  "and virginica).\n", sep = " ")
# Wrap title text at column 60
title_iris_dframe <- sapply(strwrap(title_iris_dframe, width = 60, simplify = FALSE),
paste, collapse = "\n")
# Draw table
table_influential_studies <- draw_table(body = iris_dframe, heading = title_iris_dframe)

# Table of mtcars values
mtcars_dframe <- head(mtcars)
  "for fuel consumption, automobile design and performance.\n", sep = " ")
# Wrap title text at column 60
title_mtcars_dframe <- sapply(strwrap(title_mtcars_dframe, width = 60, simplify = FALSE),
paste, collapse = "\n")
# Draw table
table_influential_studies <- draw_table(body = mtcars_dframe, heading = title_mtcars_dframe)

## End(Not run)

getmstatistic  Quantifying Systematic Heterogeneity in Meta-Analysis.

Description

gemstatistic computes M statistics to assess the contribution of each participating study in a meta-analysis. The M statistic aggregates heterogeneity information across multiple variants to identify systematic heterogeneity patterns and their direction of effect in meta-analysis. It’s primary use is to identify outlier studies, which either show "null" effects or consistently show stronger or weaker genetic effects than average, across the panel of variants examined in a GWAS meta-analysis.

Usage

gemstatistic(betain, lamdase_in, study_names_in, variant_names_in, ...)

## Default S3 method:
gemstatistic(
  beta_in,
lambda_se_in,  
study_names_in,  
variant_names_in,  
save_dir = getwd(),  
tau2_method = "DL",  
x_axis_increment_in = 0.02,  
x_axis_round_in = 2,  
produce_plots = TRUE,  
verbose_output = FALSE,  
...  
)

Arguments

beta_in  
A numeric vector of study effect-sizes e.g. log odds-ratios.

lambda_se_in  
A numeric vector of standard errors, genomically corrected at study-level.

study_names_in  
A character vector of study names.

variant_names_in  
A character vector of variant names e.g. rsIDs.

...  
Further arguments.

save_dir  
A character scalar specifying a path to the directory where plots should be stored (optional). Required if produce_plots = TRUE.

tau2_method  
A character scalar, method to estimate heterogeneity: either "DL" or "REML" (Optional). Note: The REML method uses the iterative Fisher scoring algorithm (step length = 0.5, maximum iterations = 10000) to estimate tau2.

x_axis_increment_in  
A numeric scalar, value by which x-axis of M scatterplot will be incremented (Optional).

x_axis_round_in  
A numeric scalar, value to which x-axis labels of M scatterplot will be rounded (Optional).

produce_plots  
A boolean to generate plots (optional).

verbose_output  
An optional boolean to display intermediate output.

Details

In contrast to conventional heterogeneity metrics (Q-statistic, I-squared and tau-squared) which measure random heterogeneity at individual variants, $M$ measures systematic (non-random) heterogeneity across multiple independently associated variants.

Systematic heterogeneity can arise in a meta-analysis due to differences in the study characteristics of participating studies. Some of the differences may include: ancestry, allele frequencies, phenotype definition, age-of-disease onset, family-history, gender, linkage disequilibrium and quality control thresholds. See the getmstatistic website for statistical theory, documentation and examples.

getmstatistic uses summary data i.e. study effect-sizes and their corresponding standard errors to calculate $M$ statistics (One $M$ for each study in the meta-analysis).
In particular, getMstatistic employs the inverse-variance weighted random effects regression model provided in the metafor R package to extract SPREs (standardized predicted random effects) which are then aggregated to formulate $M$ statistics.

Value

Returns a list containing:

- Mstatistic_expected_mean, A numeric scalar for the expected mean for $M$
- Mstatistic_expected_sd, A numeric scalar for the expected standard deviation for $M$
- number_studies, A numeric scalar for the number of studies
- number_variants, A numeric scalar for the number of variants
- Mstatistic_crit_alpha_0_05, A numeric scalar of the critical $M$ value at the 5 percent significance level.
- M_dataset (dataframe) A dataset of the computed $M$ statistics, which includes the following fields:
  - M, Mstatistic
  - M_sd, standard deviation of $M$
  - M_se, standard error of $M$
  - lowerbound, lowerbound of $M$ 95
  - upperbound, upperbound of $M$ 95
  - bonfpvalue, 2-sided bonferroni pvalues of $M$
  - qvalue, false discovery rate adjusted pvalues of $M$
  - tau2, tau_squared, DL estimates of between-study heterogeneity
  - I2, I_squared, proportion of total variation due to between study variance
  - Q, Cochran’s Q
  - xb, fitted values excluding random effects
  - usta, standardized predicted random effect (SPRE)
  - xbu, fitted values including random effects
  - stdxbu, standard error of prediction (fitted values) including random effects
  - hat, diagonal elements of the projection hat matrix
  - study, study numbers
  - snp, variant numbers
  - beta_mean, average variant effect size
  - oddsratio, average variant effect size as oddsratio
  - beta_n, number of variants in each study
- influential_studies_0_05 (dataframe) A dataset of influential studies significant at the 5 percent level.
- weaker_studies_0_05 (dataframe) A dataset of under-performing studies significant at the 5 percent level.

Methods (by class)

- default: Computes $M$ statistics
See Also

* rma.uni function in metafor for random effects model, and https://magosil86.github.io/getmstatistic/ for getmstatistic website.

Examples

```r
library(getmstatistic)
library(gridExtra)

# Basic M analysis using the heartgenes214 dataset.
# heartgenes214 is a multi-ethnic GWAS meta-analysis dataset for coronary artery disease.
# To learn more about the heartgenes214 dataset ?heartgenes214

# Running an M analysis on 20 GWAS significant variants (p < 5e-08) in the first 10 studies
heartgenes44_10studies <- subset(heartgenes214, studies <= 10 & fdr214_gwas46 == 2)
heartgenes20_10studies <- subset(heartgenes44_10studies, variants %in% unique(heartgenes44_10studies$variants)[1:20])

# Set directory to store plots, this can be a temporary directory
# or a path to a directory of choice e.g. plots_dir <- "~/Downloads"
plots_dir <- tempdir()
getmstatistic_results <- getmstatistic(heartgenes20_10studies$beta_flipped, heartgenes20_10studies$gcse, heartgenes20_10studies$variants, heartgenes20_10studies$studies, save_dir = plots_dir)

getmstatistic_results

# Explore results generated by getmstatistic function

# Retrieve dataset of M statistics
dframe <- getmstatistic_results$M_dataset

str(dframe)

# Retrieve dataset of stronger than average studies (significant at 5% level)
getmstatistic_results$influential_studies_0.05

# Retrieve dataset of weaker than average studies (significant at 5% level)
getmstatistic_results$weaker_studies_0.05

# Retrieve number of studies and variants
getmstatistic_results$number_studies
getmstatistic_results$number_variants

# Retrieve expected mean, sd and critical M value at 5% significance level
```
getmstatistic_results$M_expected_mean
getmstatistic_results$M_expected_sd
getmstatistic_results$M_crit_alpha_0.05

# To view plots stored in a temporary directory, call `tempdir()` to view the directory path

# Additional examples: These take a little bit longer to run

## Not run:

# Set directory to store plots, this can be a temporary directory
# or a path to a directory of choice e.g. plots_dir <- "~/Downloads"
plots_dir <- tempdir()

# Run M analysis on all 214 lead variants
# heartgenes214 is a multi-ethnic GWAS meta-analysis dataset for coronary artery disease.
getmstatistic_results <- getmstatistic(heartgenes214$beta_flipped,
                                       heartgenes214$gcse,
                                       heartgenes214$variants,
                                       heartgenes214$studies,
                                       save_dir = plots_dir)

getmstatistic_results

# Subset the GWAS significant variants (p < 5e-08) in heartgenes214
heartgenes44 <- subset(heartgenes214, heartgenes214$fdr214_gwas46 == 2)

# Exploring getmstatistic options:
# Estimate heterogeneity using "REML", default is "DL"
# Modify x-axis of M scatterplot
# Run M analysis verbosely
getmstatistic_results <- getmstatistic(heartgenes44$beta_flipped,
                                       heartgenes44$gcse,
                                       heartgenes44$variants,
                                       heartgenes44$studies,
                                       save_dir = plots_dir,
                                       tau2_method = "REML",
                                       x_axis_increment_in = 0.03,
                                       x_axis_round_in = 3,
                                       produce_plots = TRUE,
                                       verbose_output = TRUE)

getmstatistic_results

## End(Not run)
Description

heartgenes214 is a multi-ethnic GWAS meta-analysis dataset for coronary artery disease.

Usage

heartgenes214

Format

A data frame with seven variables:

- **beta_flipped**: Effect-sizes expressed as log odds ratios. Numeric
- **gcse**: Standard errors
- **studies**: Names of participating studies
- **variants**: Names of genetic variants/SNPs
- **cases**: Number of cases in each participating study
- **controls**: Number of controls in each participating study
- **fdr214_gwas46**: Flag indicating GWAS significant variants, 1: Not GWAS-significant, 2: GWAS-significant

Details

It comprises summary data (effect-sizes and their corresponding standard errors) for 48 studies (68,801 cases and 123,504 controls), at 214 lead variants independently associated with coronary artery disease (P < 0.00005, FDR < 5%). Of the 214 lead variants, 44 are genome-wide significant (p < 5e-08). The meta-analysis dataset is based on individuals of: African American, Hispanic American, East Asian, South Asian, Middle Eastern and European ancestry.

The study effect-sizes have been flipped to ensure alignment of the effect alleles.

Standard errors were genomically corrected at the study-level.

Source


https://magosil86.github.io/getmstatistic/
Index

* datasets
  heartgenes214, 8

draw_table, 2

getm (getmstatistic), 3
getmstat (getmstatistic), 3
getmstatistic, 3

heartgenes214, 7

Rgetmstatistic (getmstatistic), 3
rma.uni, 6