Package 'MetaHD'

July 21, 2025

Type Package

Title A Multivariate Meta-Analysis Model for High-Dimensional Metabolomics Data

Version 0.1.3

Maintainer Jayamini Liyanage <j.liyanage@latrobe.edu.au>

Author Jayamini Liyanage [aut, cre],
Alysha De Livera [aut]

Description Performs multivariate meta-analysis for high-dimensional metabolomics data for integrating and collectively analysing individual-level data generated from multiple studies as well as for combining summary estimates. This approach accounts for correlation between outcomes, considers variability within and between studies, handles missing values and uses shrinkage estimation to allow for high dimensionality. A detailed vignette with example datasets and code to prepare data and analyses are available on <https://bookdown.org/a2delivera/MetaHD/>.

License GPL-3

Encoding UTF-8

Depends R (>= 3.5.0)

LazyData true

RoxygenNote 7.3.2

Imports dplyr, tidyr, metafor, corpcor, nloptr, Matrix, matrixcalc,
Rcpp (>= 1.0.0)

LinkingTo Rcpp, RcppArmadillo

NeedsCompilation yes

Repository CRAN

Date/Publication 2024-09-24 04:20:03 UTC

Contents

MetaHD *A Multivariate Meta-Analysis Model for High-Dimensional Metabolomics Data*

Description

The MetaHD function performs a multivariate meta-analysis for combining summary estimates obtained from multiple metabolomic studies by using restricted maximum likelihood estimation. Assuming a meta-analysis is based on N outcomes and K studies:

Usage

```
MetaHD(
  Y, Slist,
  Psi = NULL,
  method = c("reml", "fixed"),
  bscov = c("unstructured", "diag"),
  optim.algorithm = c("BOBYQA","hybrid","L-BFGS-B"),
  initPsi = NULL,
  optim.maxiter = 2000,
  rigls.iter = 1,
  est.wscor = FALSE,
  shrinkCor = TRUE,
  impute.na = FALSE,
  impute.var = 10^4
)
```

Arguments

`Y`	: treatment effect sizes of the outcomes. This should be in the form of a K x N matrix
`Slist`	: K-dimensional list of N x N matrices representing within-study variances and covariances of the treatment effects. If within-study correlations are not available, input associated variances of treatment effects in the form of a K x N matrix and set est.wscor = TRUE.
`Psi`	: N x N matrix representing between-study variances and covariances of the treatment effects. (optional, if not specified this will be estimated internally by "MetaHD" using "estimateBSvar" and "estimateCorMat" functions in "MetaHD" package).
`method`	: estimation method: "fixed" for fixed-effects models,"reml" for random-effects models fitted through restricted maximum likelihood

`bscov` : a character vector defining the structure of the random-effects covariance matrix. Among available covariance structures, the user can select "unstructured" to obtain between-study covariance matrix with diagonal elements (variances) estimated using restricted maximul likelihood and off-diagonal elements (co-variances) reflecting the correlations estimated via shrinkage and "diag" (diagonal) for between-study variances as diagonal elements and zero co-variances

`optim.algorithm` : specifies the algorithm used to maximize the restricted log-likelihood function for estimating between-study variances. The default algorithm is "BOBYQA", which offers derivative-free, bound-constrained optimization by iteratively constructing a quadratic approximation of the objective function. The "hybrid" option performs up to rigls.iter iterations of the RIGLS algorithm, followed by quasi-Newton (BFGS algorithm) iterations until convergence. If rigls.iter is set to zero, only the quasi-Newton method (BFGS algorithm) is used for estimation. The "L-BFGS-B" algorithm is a limited-memory version of the BFGS quasi-Newton method, which supports box constraints, allowing each variable to have specified lower and/or upper bounds.

`initPsi` : N x N diagonal matrix representing the starting values of the between-study variances to be used in the optimization procedures. If not specified, the starting values in Psi default to a diagonal matrix with variances set to 1.

`optim.maxiter` : maximum number of iterations in methods involving optimization procedures.

`rigls.iter` : number of iterations of the restricted iterative generalized least square algorithm (RIGLS) when used in the initial phase of hybrid optimization procedure. Default is set to 1

`est.wscor` : a logical value indicating whether the within-study correlation matrix needs to be estimated or not. Default is FALSE

`shrinkCor` : a logical value indicating whether a shrinkage estimator should be used to estimate within- or between-study correlation matrix. Default is TRUE

`impute.na` : a logical value indicating whether missing values need to be imputed or not. Default is FALSE

`impute.var` : multiplier for replacing the missing variances in Slist.(a large value, default is 10^4)

Value

A list of objects containing estimate : a N-dimensional vector of the combined estimates, std.err : a N-dimensional vector of the associated standard errors, pVal : a N-dimensional vector of the p-values, I2.stat : I2 statistic

`MetaHDInput` *Creating Input Data for MetaHD When Individual-Level Data are Available*

Description

The MetaHDInput function creates input data Y (treatment effects) and Slist (within-study covariance matrices) for MetaHD when individual-level data are available. Assuming that the individual-level data are in the following format, with 'study' in column 1, 'group' in column 2 and outcomes in rest of the columns, with samples in rows.

Usage

```
MetaHDInput(data)
```

Arguments

`data`	a dataframe consisting of individual-level data in the format, where 'study' in column 1, 'group' in column 2 and outcomes in rest of the columns and samples in rows.

Value

A list of objects containing :

`Y`	treatment effect sizes of the outcomes in the form of a K x N matrix, where K is the number of studies and N is the number of outcomes.
`Slist`	K-dimensional list of N x N matrices representing within-study variances and covariances of the treatment effects

Examples

```
input_data <- MetaHDInput(realdata)

Y <- input_data$Y
Slist <- input_data$Slist

## MULTIVARIATE RANDOM-EFFECTS META-ANALYSIS, ESTIMATED WITH REML
model <- MetaHD(Y, Slist, method = "reml", bscov = "unstructured")
model$estimate
model$pVal
```

realdata *An Individual-Level Metabolomics Dataset*

Description

This is a subset of data, publicly available on MetaboAnalyst example datasets.

Usage

```
realdata
```

Format

A data frame with 172 observations on 14 metabolites.

Examples

```
head(realdata)
```

simdata.1 *Simulated Dataset 1 : With Complete Data*

Description

This dataset consists of a list of two data frames containing treatment effect-sizes and within-study covariance matrices

Usage

```
simdata.1
```

Format

A list of data frames as follows:

- `Y` treatment effect sizes of the metabolites in the form of a 12 x 30 matrix, where 12 is the number of studies and 30 is the number of metabolites.
- `Slist` 12-dimensional list of 30 x 30 matrices representing within-study variances and covariances of the treatment effects

Examples

```
Y <- simdata.1$Y
Slist <- simdata.1$Slist

head(Y)
head(Slist[[1]])
head(Slist[[12]])
```

simdata.2 *Simulated Dataset 2 : With Data Missing-At-Random*

Description

This dataset consists of a list of two data frames containing treatment effect-sizes and within-study covariance matrices with missing values

Usage

```
simdata.2
```

Format

A list of data frames as follows:

- `Y` treatment effect sizes of the metabolites in the form of a 12 x 30 matrix, where 12 is the number of studies and 30 is the number of metabolites.
- `Slist` 12-dimensional list of 30 x 30 matrices representing within-study variances and covariances of the treatment effects

Examples

```
Y <- simdata.2$Y
Slist <- simdata.2$Slist

head(Y)
head(Slist[[1]])
head(Slist[[12]])
```

Index