Package: mr.raps (via r-universe)

March 6, 2025

Title Two Sample Mendelian Randomization using Robust Adjusted Profile Score

Version 0.4.2

Description Mendelian randomization is a method of identifying and estimating a confounded causal effect using genetic instrumental variables. This packages implements methods for two-sample Mendelian randomization with summary statistics by using Robust Adjusted Profile Score (RAPS). References: Qingyuan Zhao, Jingshu Wang, Gibran Hemani, Jack Bowden, Dylan S. Small. Statistical inference in two-sample summary-data Mendelian randomization using robust adjusted profile score. <arXiv:1801.09652>; Qingyuan Zhao, Yang Chen, Jingshu Wang, Dylan S. Small. Powerful genome-wide design and robust statistical inference in two-sample summary-data Mendelian randomization. <arXiv:1804.07371>.

License GPL-3

Depends R (>= 2.10)

Imports ggplot2, ggrepel, graphics, gridExtra, nortest, rootSolve, rsnps, splines, stats

Suggests bumphunter, TxDb.Hsapiens.UCSC.hg38.knownGene

Remotes ropensci/rsnps

Encoding UTF-8

LazyData true

RoxygenNote 7.3.2

Config/pak/sysreqs libicu-dev libssl-dev

Repository https://mrcieu.r-universe.dev

RemoteUrl https://github.com/qingyuanzhao/mr.raps

RemoteRef HEAD

RemoteSha 27b96f31e26ae97a395422bd757f514a7f96cc85

Contents

`mr.raps-package` *Two Sample Mendelian Randomization using Robust Adjusted Profile Score*

Description

Mendelian randomization is a method of identifying and estimating a confounded causal effect using genetic instrumental variables. This packages implements methods for two sasmple Mendelian randomization with summary statistics by using Robust Adjusted Profile Score (RAPS).

Author(s)

Maintainer: Qingyuan Zhao <qingyzhao@gmail.com>

Authors:

- Jingshu Wang <wangjingshususan@gmail.com>

bmi.ais *Effect of Body Mass Index (BMI) on Acute Ischemic Stroke (AIS)*

Description

This dataset is created from three genome-wide association studies:

1. A 2017 GWAS of BMI by Akiyama et al.
2. The UK BioBank GWAS of BMI (2nd round release by the Neale lab).
3. A 2018 GWAS of AIS by Malik et al.

To obtain this dataset, the Akiyama study is used for SNP selection (column `pval.selection`). The UK BioBank dataset estimates the SNPs' effect on BMI and the Malik dataset estimates the SNPs' effect on AIS.

Usage

```
data(bmi.ais)
```

Format

A `data.frame` with 1880 rows and 29 variables.

bmi.bmi *"Effect" of Body Mass Index (BMI) on Body Mass Index (BMI)*

Description

Summary data obtained by combining three genome-wide association studies:

1. BMI-GIANT: BMI in the Genetic Investigation of ANthropometric Traits (GIANT) consortium (sample size: 339224).
2. BMI-UKBB-1: BMI in a half of the United Kingdom BioBank (UKBB) data (sample size: 234070)
3. SBP-UKBB-2: BMI in the other half of the UKBB data (sample size: 234070)

Usage

```
data(bmi.bmi)
```

Format

A `data.frame` with 812 rows and 28 variables.

Details

The BMI-GIANT dataset is used for SNP selection (column `pval.selection`). The BMI-UKBB-1 dataset estimates the SNPs' effects on BMI (columns `beta.exposure` and `se.exposure`) and the BMI-UKBB-2 dataset provides independent estimates of the same effects (columns `beta.outcome` and `se.outcome`).

---

bmi.cad *Effect of Body Mass Index (BMI) on Coronary Artery Disease (CAD)*

Description

This dataset is created from three genome-wide association studies:

1. A 2017 GWAS of BMI by Akiyama et al.
2. The UK BioBank GWAS of BMI (2nd round release by the Neale lab).
3. The CARDIoGRAMplusC4D with 1000 Genome Project imputation GWAS of CAD.

To obtain this dataset, the Akiyama study is used for SNP selection (column `pval.selection`). The UK BioBank dataset estimates the SNPs' effect on BMI and the CARDIoGRAMplusC4D estimates the SNPs' effect on CAD.

Usage

```
data(bmi.cad)
```

Format

A `data.frame` with 1119 rows and 42 variables.

---

bmi.sbp *Effect of Body Mass Index (BMI) on Systolic Blood Pressure (SBP)*

Description

Summary data obtained by combining three genome-wide association studies:

1. BMI-FEM: BMI in females by the Genetic Investigation of ANthropometric Traits (GIANT) consortium (sample size: 171977).
2. BMI-MAL: BMI in males in the same study by the GIANT consortium (sam- ple size: 152893)
3. SBP-UKBB: SBP using the United Kingdom BioBank (UKBB) data (sample size: 317754)

Usage

```
data(bmi.sbp)
```

Format

A `data.frame` with 160 rows and 29 variables.

Details

The BMI-FEM dataset is used for SNP selection (column `pval.selection`). The BMI-MAL dataset estimates the SNPs' effect on BMI and the SBP-UKBB dataset estimates the SNPs' on SBP.

`cad.bmi` *Effect of Coronary Artery Disease (CAD) on Body Mass Index (BMI) (To study reverse causality)*

Description

This dataset is created from three genome-wide association studies:

1. The C4D GWAS of CAD.
2. The CARDIoGRAM GWAS of CAD.
3. The UK BioBank GWAS of BMI (2nd round release by the Neale lab).

To obtain this dataset, the C4D study is used for SNP selection (column `pval.selection`). The CARDIoGRAM dataset estimates the SNPs' effect on CAD and the UK BioBank dataset estimates the SNPs' on BMI.

Usage

```
data(cad.bmi)
```

Format

A `data.frame` with 1625 rows and 34 variables.

`cad.cad` *Effect of Coronary Artery Disease (BMI) on Coronary Artery Disease (CAD)*

Description

This dataset is created from three genome-wide association studies:

1. The UK BioBank GWAS of heart attack (2nd round release by the Neale lab).
2. The C4D GWAS of CAD.
3. The CARDIoGRAM GWAS of CAD.

This dataset serves the purpose of a validation study. Since both the "exposure" and the "outcome" are CAD, the joint normal model of the GWAS summary statistics is expected to hold with no pleiotropy and "causal effect" equal to 1.

Usage

```
data(cad.cad)
```

Format

A `data.frame` with 1650 rows and 39 variables.

---

crp.cad *Effect of C-Reactive Protein on Coronary Artery Disease (CAD)*

Description

This dataset is created from three genome-wide association studies:

1. Prins et al.\ (2017) GWAS of CRP.
2. Dehghan et al.\ (2011) GWAS of CRP
3. The CARDIoGRAMplusC4D GWAS of CAD.

To obtain this dataset, the Prins study is used for SNP selection (column `pval.selection`). The Dehghan dataset estimates the SNPs' effect on CRP and the CARDIoGRAMplusC4D dataset estimates the SNPs' on CAD.

Usage

```
data(crp.cad)
```

Format

A `data.frame` with 1575 rows and 30 variables.

---

fit.mixture.model *Fit a Gaussian mixture deconvolution model*

Description

Fit a Gaussian mixture deconvolution model

Usage

```
fit.mixture.model(
  z,
  n = 2,
  ntry = 20,
  force.mu.zero = TRUE,
  diagnostics = FALSE
)
```

Arguments

z	A vector of z-scores.
n	Number of mixture components.
ntry	Number of random initializations.
force.mu.zero	Should the means be forced to zero?
diagnostics	Logical indicator for showing diagnostic plots.

Details

This function assumes that z is distributed as $N(\gamma, 1)$ and γ follows a Gaussian mixture model. It fits this deconvolution model by maximum likelihood and outputs the estimated mixture distribution.

Value

A list of p (mixture proportion), mu (mean), sigma (standard deviation).

Examples

```
z <- c(sqrt(2) * rnorm(900), sqrt(17) * rnorm(100))
## So the correct sigma = (1, 4) and p = (0.9, 0.1)
fit.mixture.model(z)
```

---

lipid.cad *Effect of blood lipids (LDL cholesterol, HDL cholesterol, Triglycerides) on Cardiovascular disease risk*

Description

This dataset is created from four genome-wide association studies:

1. A 2010 GWAS of blood lipids (Teslovich et al.\, 2010), named "teslovich_2010" in the dataset.
2. The MetaboChip (MC) data in a 2013 GWAS of blood lipids (Willer et al.\, 2013), named "mc_2013" in the dataset.
3. The CARDIoGRAMplusC4D meta-analysis of coronary artery disease (CARDIoGRAMplusC4D Consortium, 20135, named "cardiogramplusc4d_1000genome" in the dataset.
4. The UK BioBank GWAS of self reported heart attack (interim release by the Neale lab), named "ukbb_6150_round2" in the dataset.

Usage

```
data(lipid.cad)
```

Format

A data.frame with 12026 rows and 24 variables.

Details

lipid.cad contains in total 24 sub-datasets, each is suitable for a Mendelian randomization study. To obtain a sub-dataset, you must decide on

lipid Which lipid trait to consider? Either ldl, hdl, or tg.

gwas.selection Which GWAS is used for selection? Either teslovich_2010 or mc_2013.

gwas.exposure Which GWAS is used for exposure? Either teslovich_2010 or mc_2013 and must be different from gwas.selection.

gwas.outcome Which GWAS is used for outcome? Either cardiogramplusc4d or ukbb_self_report_heart.

restrict Should we use SNPs that are not associated with the other lipids? For example, if we are studying the effect of HDL cholesterol (so lipid is "hdl") and restrict is TRUE, then the SNPs are not associated with LDL cholesterol and triglycerides (p-value > 0.01 in the gwas.selection data).

---

modal.plot *Modal plot to detect heterogeneity*

Description

Modal plot to detect heterogeneity

Usage

```
modal.plot(
  b_exp = NULL,
  b_out = NULL,
  se_exp = NULL,
  se_out = NULL,
  data = NULL,
  k = 1.5,
  weight.option = c("MLE", "shrinkage"),
  beta.range = NULL
)
```

Arguments

b_exp	A vector of SNP effects on the exposure variable, usually obtained from a GWAS.
b_out	A vector of SNP effects on the outcome variable, usually obtained from a GWAS.
se_exp	A vector of standard errors of b_exp.
se_out	A vector of standard errors of b_out.
data	Alternatively, dataset can be passed by the argument data, which must be a data frame with columns beta.exposure, beta.outcome, se.exposure, se.outcome.
k	Locality of the robust likelihood (smaller k has more sensitivity for mode detection)
weight.option	Character. Choice of "MLE" or "shrinkage".
beta.range	range of beta in the plot

Examples

```
data(lipid.cad)
data <- subset(lipid.cad, lipid == "hdl" & restrict &
gwas.selection == "teslovich_2010" &
gwas.outcome == "cardiogramplusc4d_1000genome" &
pval.selection < 1e-5)

modal.plot(data$beta.exposure, data$beta.outcome, data$se.exposure, data$se.outcome, k = 1)

data <- subset(lipid.cad, lipid == "ldl" & restrict &
gwas.selection == "teslovich_2010" &
gwas.outcome == "cardiogramplusc4d_1000genome" &
pval.selection < 1e-5)

modal.plot(data$beta.exposure, data$beta.outcome, data$se.exposure, data$se.outcome)
```

`mr.raps` *Recommended* `mr.raps` *procedure*

Description

Recommended `mr.raps` procedure

Usage

```
mr.raps(
  data,
  diagnostics = TRUE,
  over.dispersion = TRUE,
  loss.function = "huber",
  ...
)
```

Arguments

`data`	A data frame (see Details)
`diagnostics`	Logical indicator for showing diagnostic plots.
`over.dispersion`	Should the model consider overdispersion (systematic pleiotropy)? Default is FALSE.
`loss.function`	Either the squared error loss (`l2`) or robust loss functions/scores (`huber` or `tukey`).
`...`	Additional parameters to be passed to `mr.raps.shrinkage` (default is `shrinkage=FALSE`).

Details

This function calls `mr.raps.shrinkage` with `overdispersion = TRUE`, `loss.function = "huber"`. The input data frame should contain the following variables:

1. beta.exposure
2. beta.outcome
3. se.exposure
4. se.outcome

See Also

mr.raps.shrinkage

Examples

```
# Example 1
mr.raps(bmi.sbp)

# Example 2
data(lipid.cad)
data <- subset(lipid.cad, lipid == "hdl" & restrict &
gwas.selection == "teslovich_2010" &
gwas.outcome == "cardiogramplusc4d_1000genome")
mr.raps(data)
```

`mr.raps.mle` *Main function for RAPS (MLE weights)*

Description

Main function for RAPS (MLE weights)

`mr.raps.all`: Quick analysis with all six MLE methods

`mr.raps.simple`: No overdispersion, l2 loss

`mr.raps.overdispersed`: Overdispersion, l2 loss

`mr.raps.simple.robust`: No overdispersion, robust loss

`mr.raps.overdispersed.robust`: Overdispersed, robust loss

Usage

```
mr.raps.mle(
  b_exp,
  b_out,
  se_exp,
  se_out,
  over.dispersion = FALSE,
  loss.function = c("l2", "huber", "tukey"),
  diagnostics = FALSE,
  pruning = TRUE,
  se.method = c("sandwich", "bootstrap"),
  k = switch(loss.function[1], l2 = NULL, huber = 1.345, tukey = 4.685),
  B = 1000,
  suppress.warning = FALSE
)

mr.raps.mle.all(b_exp, b_out, se_exp, se_out)

mr.raps.simple(b_exp, b_out, se_exp, se_out, diagnostics = FALSE)

mr.raps.overdispersed(
  b_exp,
  b_out,
  se_exp,
  se_out,
  initialization = c("simple", "mode"),
  suppress.warning = FALSE,
  diagnostics = FALSE,
  pruning = TRUE,
  niter = 20,
  tol = .Machine$double.eps^0.5
)

mr.raps.simple.robust(
  b_exp,
  b_out,
  se_exp,
  se_out,
  loss.function = c("huber", "tukey"),
  k = switch(loss.function[1], huber = 1.345, tukey = 4.685),
  diagnostics = FALSE
)

mr.raps.overdispersed.robust(
  b_exp,
  b_out,
  se_exp,
  se_out,
```

```
  loss.function = c("huber", "tukey"),
  k = switch(loss.function[1], huber = 1.345, tukey = 4.685),
  initialization = c("l2", "mode"),
  suppress.warning = FALSE,
  diagnostics = FALSE,
  pruning = TRUE,
  niter = 20,
  tol = .Machine$double.eps^0.5
)
```

Arguments

b_exp	A vector of SNP effects on the exposure variable, usually obtained from a GWAS.
b_out	A vector of SNP effects on the outcome variable, usually obtained from a GWAS.
se_exp	A vector of standard errors of b_exp.
se_out	A vector of standard errors of b_out.
over.dispersion	Should the model consider overdispersion (systematic pleiotropy)? Default is FALSE.
loss.function	Either the squared error loss (l2) or robust loss functions/scores (huber or tukey).
diagnostics	Should the function returns diagnostic plots and results? Default is FALSE
pruning	Should the function remove unusually large se_exp?
se.method	How should the standard error be estimated? Either by sandwich variance formula (default and recommended) or the bootstrap.
k	Threshold parameter in the Huber and Tukey loss functions.
B	Number of bootstrap resamples
suppress.warning	Should warning messages be suppressed?
initialization	Method to initialize the robust estimator. "Mode" is not supported currently.
niter	Maximum number of interations to solve the estimating equations.
tol	Numerical precision.

Details

mr.raps.mle is the main function for RAPS. It is replaced by the more general and robust function mr.raps.shrinkage.

Value

A list

beta.hat Estimated causal effect

beta.se Standard error of beta.hat

beta.p.value Two-sided p-value of `beta.hat`

tau2.hat Overdispersion parameter if `over.dispersion = TRUE`

tau2.se Standard error of `tau2.hat`

std.resid Standardized residuals of each SNP, returned if `diagnostics = TRUE`

beta.hat.loo Leave-one-out estimates of `beta.hat`, returned if `diagnostics = TRUE`

beta.hat.bootstrap Median of the bootstrap estimates, returned if `se.method = "bootstrap"`

beta.se.bootstrap Median absolute deviation of the bootstrap estimates, returned if `se.method = "bootstrap"`

Functions

- `mr.raps.mle.all()`:
- `mr.raps.simple()`:
- `mr.raps.overdispersed()`:
- `mr.raps.simple.robust()`:
- `mr.raps.overdispersed.robust()`:

References

Qingyuan Zhao, Jingshu Wang, Gibran Hemani, Jack Bowden, Dylan S. Small. Statistical inference in two-sample summary-data Mendelian randomization using robust adjusted profile score. https://arxiv.org/abs/1801.09652.

Examples

```
data(bmi.sbp)
attach(bmi.sbp)

## All estimators
mr.raps.mle.all(beta.exposure, beta.outcome, se.exposure, se.outcome)

## Diagnostic plots
res <- mr.raps.mle(beta.exposure, beta.outcome, se.exposure, se.outcome,
diagnostics = TRUE)
res <- mr.raps.mle(beta.exposure, beta.outcome, se.exposure, se.outcome,
TRUE, diagnostics = TRUE)
res <- mr.raps.mle(beta.exposure, beta.outcome, se.exposure, se.outcome,
TRUE, "tukey", diagnostics = TRUE)

detach(bmi.sbp)

data(bmi.bmi)
attach(bmi.bmi)

## Because both the exposure and the outcome are BMI, the true "causal" effect should be 1.

## All estimators
mr.raps.mle.all(beta.exposure, beta.outcome, se.exposure, se.outcome)
```

```
detach(bmi.bmi)
```

`mr.raps.scatterplot` *Scatter plot with annotation*

Description

Scatter plot with annotation

Usage

```
mr.raps.scatterplot(
  data,
  annotate = TRUE,
  annotate.genes = NULL,
  rank.method = c("pval.both", "pval.selection", "pval.exposure"),
  num.snps = 10,
  fit = mr.raps(data, FALSE),
  alpha = 0.8
)
```

Arguments

`data`	A data frame (see `mr.raps`).
`annotate`	Annotating the points? (rsid, chromosome, position)
`annotate.genes`	Further annotation of closest genes? See example.
`rank.method`	How to select strongest SNPs for plot?
`num.snps`	How many SNPs are shown?
`fit`	A `mr.raps` fit.
`alpha`	Alpha transparency value passed to `geom_errorbarh`. Default 0.8.

Examples

```
# data(bmi.sbp)
# mr.raps.scatterplot(bmi.sbp)


# require(bumphunter)
# require(TxDb.Hsapiens.UCSC.hg38.knownGene)
# genes <- annotateTranscripts(TxDb.Hsapiens.UCSC.hg38.knownGene)
# mr.raps.scatterplot(bmi.sbp, annotate.genes = genes)
```

`mr.raps.shrinkage` *Main function for RAPS (shrinkage weights)*

Description

Main function for RAPS (shrinkage weights)

Usage

```
mr.raps.shrinkage(
  b_exp,
  b_out,
  se_exp,
  se_out,
  over.dispersion = FALSE,
  loss.function = c("l2", "huber", "tukey"),
  k = switch(loss.function[1], l2 = 2, huber = 1.345, tukey = 4.685),
  shrinkage = FALSE,
  prior.param = NULL,
  diagnostics = FALSE,
  se.method = c("sandwich", "bootstrap"),
  num.init = 10,
  multiple.root.warning = 1
)

## S3 method for class 'mr.raps'
print(x, ...)

## S3 method for class 'mr.raps'
plot(x, ...)
```

Arguments

`b_exp`	A vector of SNP effects on the exposure variable, usually obtained from a GWAS.
`b_out`	A vector of SNP effects on the outcome variable, usually obtained from a GWAS.
`se_exp`	A vector of standard errors of `b_exp`.
`se_out`	A vector of standard errors of `b_out`.
`over.dispersion`	Should the model consider overdispersion (systematic pleiotropy)? Default is FALSE.
`loss.function`	Either the squared error loss (`l2`) or robust loss functions/scores (`huber` or `tukey`).
`k`	Threshold parameter in the Huber and Tukey loss functions.

shrinkage	If shrinkage (empirical partially Bayes) should be used. Shrinkage does not affect the unbiasedness of the estimating equations and generally will increase the estimation accuracy. If TRUE, prior.param must be provided.
prior.param	Parameters of the Gaussian spike-and-slab prior
diagnostics	Should the function returns diagnostic plots and results? Default is FALSE
se.method	How should the standard error be estimated? Either by sandwich variance formula (default and recommended) or the bootstrap.
num.init	Number of initializations.
multiple.root.warning	How to handle multiple roots of the estimating equations? When this happens, the results of mr.raps.shrinkage are less reliable. This parameter can take three values: 0—nothing will be done; 1—a warning is given; 2—an error is given. Default is 1.
x	a mr.raps object
...	further arguments (not supported)

Details

mr.raps.shrinkage is the main function for RAPS in conjunction with empirical partially Bayes. It is more general than the first generation mr.raps.mle function and should be preferred in practice. With the option shrinkage = TRUE, it essentially reduces to mr.raps.mle. In that case, the main difference is that the standard errors in mr.raps.shrinkage are computed based on observed information (and also an empirical estimate of the variance of the score function). This is preferred over using the plugged-in Fisher information in mr.raps.mle. See Efron and Hinkley (1978) referenced below.

Because the estimating equations are highly non-linear, it is possible that there are multiple roots. To overcome this issue, we use multiple initializations (controlled by num.init) around the mr.raps.mle point estimate. A warning is given if there seems to be another finite root, and no solution is returned if there are two roots close to the initialization. When the program does not find a finite solution, consider increasing the value of num.init.

Functions

- print(mr.raps): Print
- plot(mr.raps): Diagnostic plots

References

Qingyuan Zhao, Q., Chen, Y., Wang, J., and Small, D. S. (2018). A genome-wide design and an empirical partially Bayes approach to increase the power of Mendelian randomization, with application to the effect of blood lipids on cardiovascular disease. <arXiv:1804.07371>. Efron, B. and Hinkley, D. V. (1978). Assessing the accuracy of the maximum likelihood estimator: Observed versus expected Fisher information. Biometrika, 65(3), 457–483.

Examples

```
require(mr.raps)
data(lipid.cad)
data <- subset(lipid.cad, lipid == "hdl" & restrict &
gwas.selection == "teslovich_2010" &
gwas.outcome == "cardiogramplusc4d_1000genome")
z <- data$beta.exposure / data$se.exposure
prior.param <- fit.mixture.model(z)

## Results
mr.raps.shrinkage(data$beta.exposure, data$beta.outcome,
data$se.exposure, data$se.outcome, TRUE, "huber", shrinkage = FALSE)

mr.raps.shrinkage(data$beta.exposure, data$beta.outcome,
data$se.exposure, data$se.outcome, TRUE, "huber", shrinkage = TRUE,
prior.param = prior.param)
```

`posterior.mean` *Compute the posterior mean under spike-and-slab Gaussian prior*

Description

Compute the posterior mean under spike-and-slab Gaussian prior

Usage

```
posterior.mean(z, sigma, p, mu, sigma.prior, deriv = 0)
```

Arguments

`z`	a vector of z-scores
`sigma`	a vector of standard deviations of `z` (if `sigma` is a single number, it is expanded to a vector)
`p`	prior: mixture proportion
`mu`	prior: mean
`sigma.prior`	prior: standard deviation
`deriv`	compute the posterior mean (`deriv = 0`) or its derivative (`deriv = 1`)

Details

Similar to `fit.mixture.model`, this function assumes that z is distributed as $N(\gamma, 1)$ and γ follows a Gaussian mixture model. The function computes the posterior mean $E[\gamma|z]$.

Value

a vector

Examples

```
require(mr.raps)
data(lipid.cad)
data <- subset(lipid.cad, lipid == "hdl" & restrict &
gwas.selection == "teslovich_2010" &
gwas.outcome == "cardiogramplusc4d_1000genome")
z <- data$beta.exposure / data$se.exposure
prior.param <- fit.mixture.model(z)

z.seq <- seq(-5, 5, 0.1)
gamma.hat <- posterior.mean(z.seq, 1, prior.param$p, prior.param$mu, prior.param$sigma)
gamma.hat.deriv <- posterior.mean(z.seq, 1, prior.param$p,
prior.param$mu, prior.param$sigma, deriv = 1)
par(mfrow = c(1, 2))
plot(z.seq, gamma.hat, type = "l")
plot(z.seq, gamma.hat.deriv, type = "l")
```

Index