Package 'CopSens'

October 12, 2022

Type Package

Title Copula-Based Sensitivity Analysis for Observational Causal Inference

Version 0.1.0

Description Implements the copula-based sensitivity analysis method, as discussed in Copula-based Sensitivity Analysis for Multi-Treatment Causal Inference with Unobserved Confounding <arXiv:2102.09412>, with Gaussian copula adopted in particular.

URL https://github.com/JiajingZ/CopSens

License GPL-3

Encoding UTF-8

LazyData true

RoxygenNote 7.1.1

biocViews

Depends R (>= 3.6.0)

Imports pcaMethods (>= 1.78.0), CVXR (>= 1.0), MASS (>= 7.3-53), ggplot2 (>= 3.3.3), stats (>= 4.0.3), dplyr (>= 1.0.3), tidyr (>= 1.1.2), tidyverse (>= 1.3.0)

NeedsCompilation no

Author Jiajing Zheng [aut, cre],
Alexander Franks [aut],
Alex D'Amour [ctb]

Maintainer Jiajing Zheng <carolinez6850@gmail.com>

Repository CRAN

Date/Publication 2022-05-12 08:30:02 UTC

R topics documented:

---

bcalibrate   *Calibration for Binary Outcomes*

Description

Calibrates the naive estimates to account for unobserved confounding when outcome variables are
binary. The calibration can be done with user-specific sensitivity parameter or with our pre-provided
calibration methods, the worst-case calibration for a single contrast or multivariate calibration for
multiple contrasts.

Usage

```
bcalibrate(
  y,
  tr,
  t,
  gamma,
  R2 = NULL,
  mu_y_t = NULL,
  mu_u_tr = NULL,
  mu_u_t = NULL,
  cov_u_t = NULL,
  nU = NULL,
  nsim = 4000,
  ...
)
```

Arguments

y	`data.frame`, `matrix` or `vector`. Binary outcome variable.
tr	`data.frame`. Treatment variables with rows corresponding to observations and columns to variables.
t	`data.frame`. Treatment arms of interest. May contain a single or multiple treatments in rows.
gamma	a vector specifying the direction of sensitivity parameters.

R2	an optional scalar or vector specifying the proportion of residual variance in outcome given the treatment that can be explained by confounders, which determines the magnitude of sensitivity parameters.
mu_y_t	an optional scalar or vector that contains naive estimates of treatment effects ignoring confounding.
mu_u_tr	an optional matrix of conditional confounder means for all observed treatments with latent variables in columns.
mu_u_t	an optional matrix of conditional confounder means for treatments of interest with latent variables in columns.
cov_u_t	an optional covariance matrix of confounders conditional on treatments.
nU	Number of latent confounders to consider.
nsim	an optional scalar specifying the number of sample draws.
...	further arguments passed to kEstimate or pca.

Value

A data.frame with naive and calibrated estimates of population average outcome receiving treatment t.

Examples

```
# load the example data #
y <- GaussianT_BinaryY$y
tr <- subset(GaussianT_BinaryY, select = -c(y))
t1 <- tr[1:5,]
t2 <- rep(0, times = ncol(tr))
# calibration #
est_b <- bcalibrate(y = y, tr = tr, t = rbind(t1, t2),
                    nU = 3, gamma = c(1.27, -0.28, 0),
                    R2 = c(0.2, 0.7))
est_b_rr <- list(est_df = est_b$est_df[1:5,] / as.numeric(est_b$est_df[6,]),
                R2 = c(0.2, 0.7))
plot_estimates(est_b_rr)
```

---

cali_mean_ybinary_algm
Calibrate Estimate of Intervention Mean for Binary Outcome

Description

Calibrate Estimate of Intervention Mean for Binary Outcome

Usage

```
cali_mean_ybinary_algm(i, gamma, mu_u_tr, mu_u_t, mu_y_t, nsim = 4000)
```

Arguments

i	Observation index.
gamma	Scalar or vector specifying the sensitivity parameters.
mu_u_tr	Matrix of conditional confounder means for all observed treatments with latent variables in columns.
mu_u_t	Matrix of conditional confounder means for treatments of interest with latent variables in columns.
mu_y_t	Scalar or vector that contains naive estimates of treatment effects ignoring confounding.
nsim	Number of simulation sample draws.

Value

Scalar of calibrated intervention mean.

---

cal_rv	*Calculate Robustness Value When Executing Worstcase Calibration*

Description

Calculate Robustness Value When Executing Worstcase Calibration

Usage

```
cal_rv(
  y,
  tr,
  t1,
  t2,
  mu_y_dt = NULL,
  sigma_y_t = NULL,
  mu_u_dt = NULL,
  cov_u_t = NULL,
  nU = NULL,
  ...
)
```

Arguments

y	data.frame, matrix or vector. Gaussian outcome variable.
tr	data.frame. Treatment variables with rows corresponding to observations and columns to variables.
t1	data.frame. First treatment arms of interest. May contain a single or multiple treatments in rows.

t2	data.frame. Second treatment arms of interest, which has same number of row as t1.
mu_y_dt	an optional scalar or vector that contains naive estimates of treatment effects ignoring confounding.
sigma_y_t	an optional scalar of the standard deviation of outcome conditional on treatments.
mu_u_dt	an optional matrix of difference in conditional confounder means, $E(U \mid t1) - E(U \mid t2)$, with latent variables in columns.
cov_u_t	an optional covariance matrix of confounders conditional on treatments.
nU	Number of latent confounders to consider.
...	further arguments passed to kEstimate, pca

Value

A numeric vector with elements being the robustness value or NA if the ignorance region doesn't contains 0 for each contrast of interest.

Examples

```
# load the example data #
y <- GaussianT_GaussianY$y
tr <- subset(GaussianT_GaussianY, select = -c(y))
# calculate robustness value #
cal_rv(y = y, tr = tr, t1 = tr[1:2,], t2 = tr[3:4,])
```

GaussianT_BinaryY   *Dataset with Gaussian Treatments and Binary Outcomes*

Description

A dataset containing Gaussian treatments and binary outcomes of 10,000 observations.

Usage

```
GaussianT_BinaryY
```

Format

A data frame with eleven variables: one binary outcome, y, and ten Gaussian treatments, t1, t2, ..., t10.

Source

For data generating process, see data-raw/Data_Generation.R.

GaussianT_GaussianY   *Dataset with Gaussian Treatments and Outcomes*

Description

A dataset containing Gaussian treatments and outcomes of 10,000 observations.

Usage

```
GaussianT_GaussianY
```

Format

A data frame with eleven variables: one Gaussian outcome, y, and ten Gaussian treatments, t1, t2, ..., t10.

Source

For data generating process, see data-raw/Data_Generation.R.

---

gcalibrate   *Calibration for Gaussian Outcomes*

Description

Calibrates the naive estimates to account for unobserved confounding when outcome variables are Gaussian. The calibration can be done with user-specific sensitivity parameters or with our pre-provided calibration methods, the worst-case calibration for a single contrast or multivariate calibration for multiple contrasts.

Usage

```
gcalibrate(
  y,
  tr,
  t1,
  t2,
  calitype = c("worstcase", "multicali", "null"),
  mu_y_dt = NULL,
  sigma_y_t = NULL,
  mu_u_dt = NULL,
  cov_u_t = NULL,
  nU = NULL,
  R2 = 1,
  gamma = NULL,
  R2_constr = 1,
```

```
    nc_index = NULL,
    ...
)
```

Arguments

y	data.frame, matrix or vector. Gaussian outcome variable.
tr	data.frame. Treatment variables with rows corresponding to observations and columns to variables.
t1	data.frame. First treatment arms of interest. May contain a single or multiple treatments in rows.
t2	data.frame. Second treatment arms of interest, which has same number of row as t1.
calitype	character. The calibration method to be applied. Can be one of: "worstcase" - apply worst-case calibration when considering a single contrast. "multicali" - apply mutlivariate calibration when considering multiple contrasts. "null" - apply calibration with user-specified sensitivity parameter, γ.
mu_y_dt	an optional scalar or vector that contains naive estimates of treatment effects ignoring confounding.
sigma_y_t	an optional scalar of the standard deviation of outcome conditional on treatments.
mu_u_dt	an optional matrix of difference in conditional confounder means, $E(U \mid t1) - E(U \mid t2)$, with latent variables in columns.
cov_u_t	an optional covariance matrix of confounders conditional on treatments.
nU	Number of latent confounders to consider.
R2	an optional scalar or vector specifying the proportion of residual variance in outcome given the treatment that can be explained by confounders.
gamma	sensitivity parameter vector. Must be given when calitype = "null".
R2_constr	an optional scalar or vector specifying the upper limit constraint on R^2. By default, R2_constr = 1.
nc_index	an optional vector containing indexes of negative control treatments. If not NULL, worstcase calibration will be executed with constraints imposed by negative control treatments.
...	further arguments passed to [kEstimate], [pca] or [get_opt_gamma].

Value

gcalibrate returns a list containing the following components:

est_df a data.frame with naive and calibrated estimates of average treatment effects.

R2 a vector of R^2 with elements corresponding to columns of est_df.

gamma a matrix returned when calitype = "multicali" or "worstcase". If calitype = "multicali", optimized gamma are in columns, respectively resulting in estimates in columns of est_df. If calitype = "worstcase", gamma are in rows, which respectively lead to the worstcase ignorance region with $R^2 = 1$ for each contrast of interest.

rv a numeric vector returned when calitype = "worstcase", with elements being the robust-
ness value or NA if the ignorance region doesn't contains 0 for each contrast of interest.

Examples

```
# load the example data #
y <- GaussianT_GaussianY$y
tr <- subset(GaussianT_GaussianY, select = -c(y))

# worst-case calibration #
t1 <- data.frame(diag(ncol(tr)))
t2 <- data.frame(matrix(0, nrow = ncol(tr), ncol = ncol(tr)))
colnames(t1) = colnames(t2) <- colnames(tr)
est_g1 <- gcalibrate(y = y, tr = tr, t1 = t1, t2 = t2, nU = 3,
                     calitype = "worstcase", R2 = c(0.3, 1))
plot_estimates(est_g1)
# with negative conotrls #
est_g1_nc <- gcalibrate(y = y, tr = tr, t1 = t1, t2 = t2, nU = 3,
                        calitype = "worstcase", R2 = c(0.3, 1), nc_index = c(3, 6))
plot_estimates(est_g1_nc)


# multivariate calibration #
est_g2 <- gcalibrate(y = y, tr = tr, t1 = tr[1:10,], t2 = tr[11:20,], nU = 3,
                     calitype = "multicali", R2_constr = c(1, 0.15))
plot_estimates(est_g2)

# user-specified calibration #
est_g3 <- gcalibrate(y = y, tr = tr, t1 = tr[1:2,], t2 = tr[3:4,],
                     nU = 3, calitype = "null",
                     gamma = c(0.96, -0.29, 0), R2 = c(0.2, 0.6, 1))
plot_estimates(est_g3)
# apply gamma that maximizes the bias for the first contrast considered in est_g1 #
est_g4 <- gcalibrate(y = y, tr = tr, t1 = tr[1:2,], t2 = tr[3:4,],
                     nU = 3, calitype = "null",
                     gamma = est_g1$gamma[1,], R2 = c(0.2, 0.6, 1))
plot_estimates(est_g4)
```

---

get_opt_gamma   *Obtain Optimized Sensitivity Parameters Using Multivariate Calibra-*
                             tion Criterion

Description

Obtain Optimized Sensitivity Parameters Using Multivariate Calibration Criterion

Usage

```
get_opt_gamma(
  mu_y_dt,
  mu_u_dt,
  cov_u_t,
  sigma_y_t,
  R2_constr = 1,
  normtype = "L2",
  ...
)
```

Arguments

mu_y_dt	Scalar or vector that contains naive estimates of treatment effects ignoring confounding.
mu_u_dt	Matrix of difference in conditional confounder means, $E(U \mid t1) - E(U \mid t2)$, with latent variables in columns.
cov_u_t	Covariance matrix of confounders conditional on treatments.
sigma_y_t	Scalar of the standard deviation of outcome conditional on treatments.
R2_constr	an optional scalar or vector specifying the upper limit constraint on R^2 . By default, R2_constr = 1.
normtype	character. Optional function m for the multivariate calibration criterion. By default, the L2 norm will be applied. "L1" - apply the L1 norm, sum(abs(x)). "L2" - apply the L2 norm, sqrt(sum(x^2)). "Inf" - apply the infinity norm, max(abs(x)).
...	further arguments passed to solve

Value

Optimized sensitivity parameters.

---

micedata	*Body weight and gene expressions of 287 mice*

Description

A dataset are collected from 287 mice, including the body weight, 37 gene expressions, and 5 single nucleotide polymorphisms.

Usage

```
micedata
```

Format

A data frame with forty-three variables: the mice body weight, y, 5 single nucleotide polymorphisms, rs3663003, rs4136518, rs3694833, rs4231406, rs3661189, and the rest are thirty-seven genes.

Source

<https://arxiv.org/abs/2011.04504>

---

mice_est_nulltr	*Estimates of genes' effects on mice body weight using null treatments approach from Miao et al. (2020)*

Description

The dataset consists of estimates of treatment effects of 17 genes, which are likely to affect mouse weight, by using the null treatments approach from Miao et al. (2020), assuming that at least half of the confounded treatments have no causal effect on the outcome.

Usage

mice_est_nulltr

Format

A data frame with 17 rows and 6 variables:

esti mean estimates of genes' treatment effects on mouse body weight

X2.5. 2.5% percentile of the estimates of genes' treatment effects on mouse body weight

X97.5. 97.5% percentile of the estimates of genes' treatment effects on mouse body weight

X5. 5% percentile of the estimates of genes' treatment effects on mouse body weight

X95. 95% percentile of the estimates of genes' treatment effects on mouse body weight

signif significance

Source

<https://arxiv.org/abs/2011.04504>

---

plot_estimates   *Visualize Estimates of Treatment Effects*

Description

Visualize Estimates of Treatment Effects

Usage

```
plot_estimates(est, show_rv = TRUE, order = "naive", labels = NULL, ...)
```

Arguments

est   an return object from [gcalibrate](#) or [bcalibrate](#), or data.frame containing
              estimates of treatment effects with estimates' type in columns and contrasts of
              interest in rows.

show_rv   logical. Whether robustness values should be printed in the plot or not? Avail-
              able only for the "worstcase" calibration.

order   character. The type of order used to plot treatment effects from left to right. Can
              be one of the following:
              "naive" - order by the naive estimate from smallest to largest. "worstcase" - place
              all treatments with negative robust effects on the left, with positive robust effects
              on the right, and all sensitive ones in the middle. Within the negative robust
              group, order treatments by the upper bound of the worst-case ignorance region
              from smallest to largest; within the positive robust group, order treatments by
              the lower bound of the worst-case ignorance region from smallest to largest; and
              within the sensitive group, order by the naive estimate from smallest to largest.

labels   character. Labels of treatments.

...   further arguments passed to [theme](#)

Value

A graph plotting ignorance regions of the causal estimands of interest.

Note

For examples, please refer to [bcalibrate](#) or [gcalibrate](#)

Index