Package 'sdafilter'

May 22, 2021

Title Symmetrized Data Aggregation

Version 1.0.0

Author Lilun Du [aut, cre],
Xu Guo [ctb],
Wenguang Sun [ctb],
Changliang Zou [ctb]

Maintainer Lilun Du <dulilun@ust.hk>

Description We develop a new class of distribution free multiple testing rules for false discovery rate (FDR) control under general dependence. A key element in our proposal is a symmetrized data aggregation (SDA) approach to incorporating the dependence structure via sample splitting, data screening and information pooling. The proposed SDA filter first constructs a sequence of ranking statistics that fulfill global symmetry properties, and then chooses a data driven threshold along the ranking to control the FDR. For more information, see the website below and the accompanying paper: Du et al. (2020), ``False Discovery Rate Control Under General Dependence By Symmetrized Data Aggregation'', <arXiv:2002.11992>.

License GPL (>= 2)

Encoding UTF-8

LazyData true

RoxygenNote 7.1.1

Imports glmnet,
glasso,
huge,
POET,
stats,
selectiveInference,
pfa,
MASS

Suggests testthat (>= 2.1.0)

Repository CRAN

R topics documented:

SDA_2S   *Symmetrized Data Aggregation for two-sample t-test*

Description

Symmetrized Data Aggregation for two-sample t-test

Usage

```
SDA_2S(dat_I, dat_II, alpha, Sigma_I, Sigma_II, stable = TRUE)
```

Arguments

dat_I	a n_1 by p data matrix, first sample
dat_II	a n_2 by p data matrix, second sample
alpha	the FDR level
Sigma_I	the covariance matrix of sample 1; if missing, it will be estimated by the glasso package
Sigma_II	the covariance matrix of sample 2; if missing, it will be estimated by the glasso package
stable	if TRUE, the sample will be randomly splitted B=10 times for stability performance; otherwise, only single sample splitting is used.

Value

the indices of the hypotheses rejected

Examples

```
p = 100
n = 30
dat_I = matrix(rnorm(n*p),nrow = n)
mu = rep(0, p)
mu[1:10] = 1.5
dat_I = dat_I = rep(1, n)%*%t(mu)

dat_II = matrix(rnorm(n*p), nrow = n)
Sigma_I = diag(p)
Sigma_II = diag(p)
out = SDA_2S(dat_I, dat_II, alpha=0.05, Sigma_I, Sigma_II)
print(out)
```

SDA_M   *Symmetrized Data Aggregation for one-sample t-test*

Description

This is the core function for the paper posted in arXiv preprint arXiv:2002.11992

Usage

```
SDA_M(dat, alpha, Omega, nonsparse = FALSE, stable = TRUE)
```

Arguments

dat   a n by p data matrix

alpha   the FDR level

Omega   the inverse covariance matrix; if missing, it will be estimated by the glasso package

nonsparse   if TRUE, the covariance matrix will be estimated by the POET package

stable   if TRUE, the sample will be randomly splitted B=10 times for stability performance; otherwise, only single sample splitting is used.

Value

the indices of the hypotheses rejected

Examples

```
n = 50
p = 100
dat = matrix(rnorm(n*p), nrow=n)
mu = rep(0, p)
mu[1:as.integer(0.1*p)]=0.3
dat = dat+rep(1, n)%*%t(mu)
alpha = 0.2
out = SDA_M(dat, alpha, diag(p))
print(out)
```

SDA_robust   *Symmetrized Data Aggregation for one sample t-test; other options for calculating the test statistics from the first sample*

Description

Other commomly used test statistics for the first sample are allowed in this function.

Usage

```
SDA_robust(
  dat,
  alpha,
  Omega,
  nonsparse = FALSE,
  stable = TRUE,
  kwd = c("lasso", "de_lasso", "innovate", "pfa"),
  scale = TRUE
)
```

Arguments

dat	a n by p data matrix
alpha	the FDR level
Omega	the inverse covariance matrix; if it is missing, it will be estimated by the glasso package.
nonsparse	if it is TRUE, the covariance matrix will be estimated by the POET package; otherwise it will be fitted by glasso by default.
stable	if it is TRUE, the sample will be randomly splitted $B = 10$ times for stability performance; otherwise, only single sample splitting is used.
kwd	various methods for calculating the test statistics from the first sample
scale	if it is TRUE, the test statistic from the first sample will be standardized.

Details

We provide other commonly used test statistics for the first sample. These include the debiased lasso, innovated transformation, and the factor-adjusted test statistics.

Value

the indices of the hypotheses rejected

Examples

```
n = 50
p = 100
rho = 0.8
Sig = matrix(rho, p, p)
diag(Sig) = 1
dat <- MASS::mvrnorm(n, rep(0, p), Sig)
mu = rep(0, p)
mu[1:as.integer(0.1*p)]=0.5
dat = dat+rep(1, n)%*%t(mu)
alpha = 0.2
out = SDA_robust(dat, alpha, solve(Sig), kwd=innovate)
print(out)
```

Index