Package ‘drcarlate’

July 22, 2025

Type Package

Title Improving Estimation Efficiency in CAR with Imperfect Compliance

Version 1.2.0

Description We provide a list of functions for replicating the results of the Monte Carlo simulations and empirical application of Jiang et al. (2022).
In particular, we provide corresponding functions for generating the three types of random data described in this paper, as well as all the estimation strategies.
Detailed information about the data generation process and estimation strategy can be found in Jiang et al. (2022) <doi:10.48550/arXiv.2201.13004>.

License MIT + file LICENSE

Encoding UTF-8

LazyData true

Imports pracma, MASS, stringr, splus2R, glmnet, stats, purrr

RoxygenNote 7.2.3

Suggests knitr, rmarkdown

VignetteBuilder knitr

Depends R (>= 2.10)

NeedsCompilation no

Author Liang Jiang [aut, cph],
Oliver B. Linton [aut, cph],
Haihan Tang [aut, cph],
Yichong Zhang [aut, cph],
Mingxin Zhang [cre]

Maintainer Mingxin Zhang <21110680035@m.fudan.edu.cn>

Repository CRAN

Date/Publication 2023-06-12 09:40:02 UTC

Contents

`ATEDGP` *Simulates the data for ATE estimators*

Description

ATEDGP is the version of FuncDGP under full compliance.

Usage

```
ATEDGP(dgptype, rndflag, n, g, pi)
```

Arguments

`dgptype`	A scalar. 1, 2, 3 (Almost the same as 1-3 in the paper except that it does not have the DGP for D(1) or D(0)).
`rndflag`	A scalar. method of covariate-adaptive randomization. 1-SRS; 2-WEI; 3-BCD; 4-SBR.
`n`	Sample size.
`g`	Number of strata. The authors set g = 4 in the Jiang et al. (2022).
`pi`	A g x 1 vector. Targeted assignment probabilities across strata.

Value

ATEDGP returns a list containing 7 nx1 vectors named Y, X, S, A, Y1, Y0 and D. These seven vectors are the same as defined in Jiang et al. (2022). Note that vector X does not contain the constant term.

References

Jiang L, Linton O B, Tang H, Zhang Y. Improving estimation efficiency via regression-adjustment in covariate-adaptive randomizations with imperfect compliance [J]. 2022.

Examples

```
ATEDGP(dgptype = 1, rndflag = 1, n = 200, g = 4, pi = c(0.5, 0.5, 0.5, 0.5))
```

ATEJLTZ *ATEJLTZ runs the code for ATE estimator*

Description

ATEJLTZ is the version of JLTZ under full compliance.

Usage

```
ATEJLTZ(iMonte, dgptype, n, g, pi, iPert, iq = 0.05, iridge = 0.001, seed = 1)
```

Arguments

`iMonte`	A scalar. Monte Carlo sizes.
`dgptype`	A scalar. The value can be string 1, 2, or 3, respectively corresponding to the three DGP schemes in the paper (See Jiang et al. (2022) for DGP details).
`n`	Sample size.
`g`	Number of strata. The authors set g=4 in Jiang et al. (2022).
`pi`	Targeted assignment probability across strata.
`iPert`	A scalar. iPert = 0 means size. Otherwise means power: iPert is the perturbation of false null.
`iq`	A scalar. Size of hypothesis testing. The authors set iq = 0.05 in Jiang et al. (2022).
`iridge`	A scalar. The penalization parameter in ridge regression.
`seed`	A scalar. The random seed, the authors set seed = 1 in Jiang et al. (2022).

Value

A table summarizing the estimated results, mProd.

References

Jiang L, Linton O B, Tang H, Zhang Y. Improving estimation efficiency via regression-adjustment in covariate-adaptive randomizations with imperfect compliance [J]. 2022.

Examples

```
# size, iPert = 0
ATEJLTZ(iMonte = 10, dgptype = 1, n = 200, g = 4,
    pi = c(0.5, 0.5, 0.5, 0.5), iPert = 0, iq = 0.05, iridge = 0.001)

# power, iPert = 1
ATEJLTZ(iMonte = 10, dgptype = 1, n = 200, g = 4,
    pi = c(0.5, 0.5, 0.5, 0.5), iPert = 1, iq = 0.05, iridge = 0.001)
```

`ATEOutput` *Computes linear, nonparametric and regularized ATE estimator*

Description

ATEOutput is the version of Output under full compliance.

Usage

```
ATEOutput(ii, tau, dgptype, rndflag, n, g, pi, iPert, iq, iridge)
```

Arguments

`ii`	Monte Carlo index.
`tau`	A scalar. The simulated true LATE effect.
`dgptype`	A Scalar. 1, 2, 3 (See Jiang et al. (2022) for DGP details).
`rndflag`	Method of CAR (covariate-adaptive randomizations). Its value can be 1, 2, 3 or4. 1-SRS; 2-WEI; 3-BCD; 4-SBR. See Jiang et al. (2022) for more details about CAR.
`n`	Sample size.
`g`	Number of strata. The authors set g=4 in Jiang et al. (2022).
`pi`	Targeted assignment probability across strata.
`iPert`	A scalar. iPert =0 means size. Otherwise means power: iPert is the perturbation of false null.
`iq`	Size of hypothesis testing. We set iq = 0.05.
`iridge`	A scalar. The penalization parameter in ridge regression.

Value

A list containing four matrices named vtauhat, vsighat, vstat and vdeci respectively. vtauhat is a 1x4 vector: (1) L (2) NL (3) R(dgp = 1 or 2) (4) R(dgp = 3). vsighat is a 1x4 vector: unscaled standard errors for vtauhat. vstat is a 1x4 vector: test statistic. vdeci is a 1x4 logical vector: if applicable, 1 means rejecting the null. 0 means not rejecting the null.

Examples

```
ATEOutput(ii = 1, tau = 0.9122762, dgptype = 1,
        rndflag = 4, n = 2000, g = 4, pi = c(0.5,0.5,0.5,0.5),
        iPert = 1, iq = 0.05, iridge = 0.001)
```

`ATETrueValue` *Calculates the true ATE effect.*

Description

ATETrueValue is the version of TrueValue under full compliance.

Usage

```
ATETrueValue(dgptype, vIdx, n, g, pi)
```

Arguments

`dgptype`	A scalar. The value can be string 1, 2, or 3, respectively corresponding to the three DGP schemes in the paper (See Jiang et al. (2022) for DGP details).
`vIdx`	A 1xR vector. The authors set vIdx=[1 2 3 4] in Jiang et al. (2022). Every number declares the method of covariate-adaptive randomization. 1-SRS; 2-WEI; 3-BCD; 4-SBR.
`n`	Sample size.
`g`	Number of strata. The authors set g=4 in Jiang et al. (2022).
`pi`	Targeted assignment probability across strata.

Value

A 1xR vector. Simulated true ATE effect.

References

Jiang L, Linton O B, Tang H, Zhang Y. Improving estimation efficiency via regression-adjustment in covariate-adaptive randomizations with imperfect compliance [J]. 2022.

Examples

```
ATETrueValue(dgptype = 1, vIdx = c(1,2,3,4), n = 100, g = 4, pi = c(0.5,0.5,0.5,0.5))
ATETrueValue(dgptype = 2, vIdx = c(1,2,3,4), n = 100, g = 4, pi = c(0.5,0.5,0.5,0.5))
ATETrueValue(dgptype = 3, vIdx = c(1,2,3,4), n = 100, g = 4, pi = c(0.5,0.5,0.5,0.5))
```

CovAdptRnd	*Generate treatment assignment under various CARs*

Description

Generate treatment assignment under various CARs.

Usage

```
CovAdptRnd(rndflag, S, pi)
```

Arguments

rndflag	Index of the assignment rule. 1 for SRS; 2 for WEI; 3 for BCD; 4 for SBR
S	A nx1 vector.
pi	Targeted assignment probability across strata. It should be a vector with the length of max(S), It should be noted that the treatment assignment process is independent of pi when rndflag == 2 or 3.

Value

A nx1 treatment assignment vector generated according to the specified method.

References

Jiang L, Linton O B, Tang H, Zhang Y. Improving estimation efficiency via regression-adjustment in covariate-adaptive randomizations with imperfect compliance [J]. 2022.

Examples

```
CovAdptRnd(rndflag = 1, S = matrix(sample(1:4,100,TRUE)), pi = c(0.5, 0.5, 0.5, 0.5))
CovAdptRnd(rndflag = 2, S = matrix(sample(1:4,100,TRUE)), pi = c(0.5, 0.5, 0.5, 0.5))
CovAdptRnd(rndflag = 3, S = matrix(sample(1:4,100,TRUE)), pi = c(0.5, 0.5, 0.5, 0.5))
CovAdptRnd(rndflag = 4, S = matrix(sample(1:4,100,TRUE)), pi = c(0.5, 0.5, 0.5, 0.5))
```

data_table	*Data used to reproduce Table 5 results in Jiang et. al. (2022)*

Description

Data used to reproduce Table 5 results in Jiang et. al. (2022).

Usage

```
data("data_table")
```

Format

A data frame with 2159 observations on the following 69 variables.

X1 a numeric vector

X2 a numeric vector

X3 a numeric vector

X4 a numeric vector

X5 a numeric vector

X6 a numeric vector

X7 a numeric vector

X8 a numeric vector

X9 a numeric vector

X10 a numeric vector

X11 a numeric vector

X12 a numeric vector

X13 a numeric vector

X14 a numeric vector

X15 a numeric vector

X16 a numeric vector

X17 a numeric vector

X18 a numeric vector

X19 a numeric vector

X20 a numeric vector

X21 a numeric vector

X22 a numeric vector

X23 a numeric vector

X24 a numeric vector

X25 a numeric vector

X26 a numeric vector

X27 a numeric vector

X28 a numeric vector

X29 a numeric vector

X30 a numeric vector

X31 a numeric vector

X32 a numeric vector

X33 a numeric vector

X34 a numeric vector

X35 a numeric vector

X36 a numeric vector
X37 a numeric vector
X38 a numeric vector
X39 a numeric vector
X40 a numeric vector
X41 a numeric vector
X42 a numeric vector
X43 a numeric vector
X44 a numeric vector
X45 a numeric vector
X46 a numeric vector
X47 a numeric vector
X48 a numeric vector
X49 a numeric vector
X50 a numeric vector
X51 a numeric vector
X52 a numeric vector
X53 a numeric vector
X54 a numeric vector
X55 a numeric vector
X56 a numeric vector
X57 a numeric vector
X58 a numeric vector
X59 a numeric vector
X60 a numeric vector
X61 a numeric vector
X62 a numeric vector
X63 a numeric vector
X64 a numeric vector
X65 a numeric vector
X66 a numeric vector
X67 a numeric vector
X68 a numeric vector
X69 a numeric vector

References

Jiang L, Linton O B, Tang H, Zhang Y. Improving estimation efficiency via regression-adjustment in covariate-adaptive randomizations with imperfect compliance [J]. 2022.

feasiblePostLassoMatTool

Feasible Post Lasso Mat Tool

Description

Under the condition of high dimensional data, the function first selects covariables through lasso regression, then performs logit regression or linear regression according to the caller's requirements, and finally returns the adjusted Lasso regression coefficient vector. This function has been slightly adapted for this package.

Usage

```
feasiblePostLassoMatTool(
  x,
  y,
  MaxIter = 30,
  UpsTol = 1e-06,
  beta0 = c(),
  clusterVar = c(),
  Dist = "normal",
  link = "identity",
  glmTol = 1e-08,
  initScale = 0.5
)
```

Arguments

`x`	A nxk Matrix.
`y`	A nx1 vector.
`MaxIter`	Maximum iteration. The default value is 30.
`UpsTol`	Upper limit of tolerance. The default value is 1e-6.
`beta0`	NULL.
`clusterVar`	NULL.
`Dist`	The default value is normal.
`link`	Link can be identity or logit. This determines the method used for regression with the selected write variable after lasso. See Jiang et al. (2022) for more details.
`glmTol`	Maximum tolerance in GLM. The default value is 1e-8.
`initScale`	Initial scale, the default value is 0.5.

Value

A kx1 cector, the coefficients b.

References

Belloni, A., Chernozhukov, V., Fernández-Val, I. and Hansen, C. (2017), Program Evaluation and Causal Inference With High-Dimensional Data. Econometrica, 85: 233-298. https://doi.org/10.3982/ECTA12723

Examples

```
set.seed(1)
# Notice that when we set dgptype = 3, FuncDGP will generate a high dimensional data for us.
DGP <- FuncDGP(dgptype = 3, rndflag = 1, n = 10000, g = 4, pi = c(0.5, 0.5, 0.5, 0.5))
X <- DGP$X
Y <- DGP$Y
A <- DGP$A
S <- DGP$S
D <- DGP$D
feasiblePostLassoMatTool(x = X[S==1 & A==0,], y = Y[S==1 & A==0,])
feasiblePostLassoMatTool(x = X[S==1 & A==0,], y = D[S==1 & A==0,], link = "logit")
```

FuncDGP *Generate Data for LATE*

Description

Generate data according to one of the three DGPs in Jiang et al. (2022).

Usage

```
FuncDGP(dgptype, rndflag, n, g, pi)
```

Arguments

dgptype	A Scalar. 1, 2, 3 (See Jiang et al. (2022) for DGP details)
rndflag	A Scalar. Declare the method of covariate-adaptive randomization. 1-SRS; 2-WEI; 3-BCD; 4-SBR.
n	Sample size
g	Number of strata. The authors set g=4 in the Jiang et al. (2022).
pi	Targeted assignment probability across strata.

Value

FuncDGP returns a list containing 9 nx1 vectors named Y, X, S, A, Y1, Y0, D1, D0 and D. These nine vectors are the same as defined in Jiang et al. (2022). Note that vector X does not contain the constant term.

References

Jiang L, Linton O B, Tang H, Zhang Y. Improving estimation efficiency via regression-adjustment in covariate-adaptive randomizations with imperfect compliance [J]. 2022.

Examples

```
FuncDGP(dgptype = 1, rndflag = 1, n = 200, g = 4, pi = c(0.5,0.5,0.5,0.5))
FuncDGP(dgptype = 1, rndflag = 2, n = 200, g = 4, pi = c(0.5,0.5,0.5,0.5))
FuncDGP(dgptype = 1, rndflag = 3, n = 200, g = 4, pi = c(0.5,0.5,0.5,0.5))
FuncDGP(dgptype = 1, rndflag = 4, n = 200, g = 4, pi = c(0.5,0.5,0.5,0.5))

FuncDGP(dgptype = 2, rndflag = 1, n = 200, g = 4, pi = c(0.5,0.5,0.5,0.5))
FuncDGP(dgptype = 2, rndflag = 2, n = 200, g = 4, pi = c(0.5,0.5,0.5,0.5))
FuncDGP(dgptype = 2, rndflag = 3, n = 200, g = 4, pi = c(0.5,0.5,0.5,0.5))
FuncDGP(dgptype = 2, rndflag = 4, n = 200, g = 4, pi = c(0.5,0.5,0.5,0.5))

FuncDGP(dgptype = 3, rndflag = 1, n = 200, g = 4, pi = c(0.5,0.5,0.5,0.5))
FuncDGP(dgptype = 3, rndflag = 2, n = 200, g = 4, pi = c(0.5,0.5,0.5,0.5))
FuncDGP(dgptype = 3, rndflag = 3, n = 200, g = 4, pi = c(0.5,0.5,0.5,0.5))
FuncDGP(dgptype = 3, rndflag = 4, n = 200, g = 4, pi = c(0.5,0.5,0.5,0.5))
```

JLTZ — *Reproduce the results of the Jiang et al. (2022)*

Description

Helps the user reproduce the results of the data simulation section of Jiang et al. (2022).

Usage

```
JLTZ(iMonte, dgptype, n, g, pi, iPert, iq = 0.05, iridge = 0.001, seed = 1)
```

Arguments

`iMonte`	A scalar. Monte Carlo sizes.
`dgptype`	A scalar. The value can be string 1, 2, or 3, respectively corresponding to the three random data generation methods in the paper (See Jiang et al. (2022) for DGP details).
`n`	Sample size.
`g`	Number of strata. We set g=4 in Jiang et al. (2022).
`pi`	Targeted assignment probability across strata.
`iPert`	A scalar. iPert = 0 means size. Otherwise means power: iPert is the perturbation of false null.
`iq`	A scalar. Size of hypothesis testing. The authors set iq = 0.05.
`iridge`	A scalar. The penalization parameter in ridge regression.
`seed`	A scalar. The random seed, the authors set seed = 1 in Jiang et al. (2022).

Value

A table summarizing the estimated results, mProd.

References

Jiang L, Linton O B, Tang H, Zhang Y. Improving estimation efficiency via regression-adjustment in covariate-adaptive randomizations with imperfect compliance [J]. 2022.

Examples

```
# size, iPert = 0
JLTZ(iMonte = 10, dgptype = 1, n = 200, g = 4,
    pi = c(0.5, 0.5, 0.5, 0.5), iPert = 0, iq = 0.05, iridge = 0.001, seed = 1)

# power, iPert = 1
JLTZ(iMonte = 10, dgptype = 1, n = 200, g = 4,
    pi = c(0.5, 0.5, 0.5, 0.5), iPert = 1, iq = 0.05, iridge = 0.001, seed = 1)
```

LinearLogit *Linear Regression or Logit Regression*

Description

LinearLogit generates estimated pseudo true values for parametric models. Different estimation strategies are adopted according to different values of modelflag. See Jiang et al. (2022) for more details about different strategies.

Usage

```
LinearLogit(Y, D, A, X, S, s, modelflag, iridge)
```

Arguments

Y	The outcome vector. A nx1 vector.
D	A nx1 vector.
A	The treatment assignment. A nx1 vector.
X	Extra covariate matrix, A nxK matrix without constant.
S	The strata variable.
s	A particular stratum.
modelflag	Its value ranges from characters 1, 2, and 3, respectively declaring different estimation strategies. 1-L; 2-NL; 3-R.
iridge	A scalar. The penalization parameter in ridge regression.

Value

theta_0s, theta_1s, beta_0s, beta_1s are estimated coefficients vectors. The dimension is Kx1 if modelflag = 1; (K+1)x1 if modelflag = 2 or 3.

References

Jiang L, Linton O B, Tang H, Zhang Y. Improving estimation efficiency via regression-adjustment in covariate-adaptive randomizations with imperfect compliance [J]. 2022.

Examples

```
#' set.seed(1)
DGP <- FuncDGP(dgptype = 3, rndflag = 1, n = 10000, g = 4, pi = c(0.5, 0.5, 0.5, 0.5))
X <- DGP$X
Y <- DGP$Y
A <- DGP$A
S <- DGP$S
D <- DGP$D
LinearLogit(Y = Y, D = D, A = A, X = X, S = S, s = 1, modelflag = 1, iridge = 0.001)
LinearLogit(Y = Y, D = D, A = A, X = X, S = S, s = 2, modelflag = 2, iridge = 0.001)
LinearLogit(Y = Y, D = D, A = A, X = X, S = S, s = 3, modelflag = 3, iridge = 0.001)
LinearLogit(Y = Y, D = D, A = A, X = X, S = S, s = 4, modelflag = 3, iridge = 0.001)
```

LogisticReg *Logistic Regression Function*

Description

Logestic CDF(cumulative distribution function).

Usage

```
LogisticReg(x)
```

Arguments

x	A nx1 matrix.

Value

y A nx1 matrix. y equals to exp(x)/(1+exp(x)) if y is not NA and 0 else.

Examples

```
x <- pracma::rand(5,1)
y <- LogisticReg(x = x)
```

`norminv` *Inverse of the normal cumulative distribution function (cdf)*

Description

Returns the inverse cdf for the normal distribution with mean MU and standard deviation SIGMA at P value Reference: https://rdrr.io/github/maxto/qapi/src/R/stats.R

Usage

```
norminv(p, mu = 0, sigma = 1)
```

Arguments

`p`	probability value in range 0-1
`mu`	mean value
`sigma`	standard deviation

Value

numeric

Examples

```
xx <- c(0.003,0.026,0.015,-0.009,-0.014,-0.024,0.015,0.066,-0.014,0.039)
norminv(0.01,mean(xx),sd(xx))
```

`Output` *Computes All the Estimators*

Description

Output is an integrated function that computes all the estimates (including NA, TSLS, L, NL, F, NP, R) used in Jiang et al. (2022). See the paper for more details.

Usage

```
Output(ii, tau, dgptype, rndflag, n, g, pi, iPert, iq, iridge)
```

Arguments

- `ii` Monte Carlo index.
- `tau` A scalar. The simulated true LATE effect.
- `dgptype` A Scalar. 1, 2, 3 (See Jiang et al. (2022) for DGP details).
- `rndflag` Method of CAR (covariate-adaptive randomizations). Its value can be 1, 2, 3 or4. 1-SRS; 2-WEI; 3-BCD; 4-SBR. See Jiang et al. (2022) for more details about CAR.
- `n` Sample size.
- `g` Number of strata. The authors set g=4 in Jiang et al. (2022).
- `pi` Targeted assignment probability across strata.
- `iPert` A scalar. iPert =0 means size. Otherwise means power: iPert is the perturbation of false null.
- `iq` Size of hypothesis testing. The authors set iq = 0.05 in Jiang et al. (2022).
- `iridge` A scalar. The penalization parameter in ridge regression.

Value

A list containing four matrices named vtauhat, vsighat, vstat and vdeci respectively. vtauhat is a 1x8 vector: (1) NA (2) LP (3) LG (4) F (5) NP (6) R (when dgp = 3) (7) 2SLS (8) R (when dgp = 1 or 2). vsighat is a 1x8 vector: unscaled standard errors for vtauhat. vstat is a 1x8 vector: test statistic. vdeci is a 1x8 logical vector: if applicable, 1 means rejecting the null. 0 means not rejecting the null.

References

Jiang L, Linton O B, Tang H, Zhang Y. Improving estimation efficiency via regression-adjustment in covariate-adaptive randomizations with imperfect compliance [J]. 2022.

Examples

```
Output(ii = 1, tau = 0.9122762, dgptype = 1,
       rndflag = 4, n = 2000, g = 4, pi = c(0.5,0.5,0.5,0.5),
       iPert = 1, iq = 0.05, iridge = 0.001)
```

---

pihat *Compute Estimated Treatment Assignment Probabilities*

Description

Pihat computes the targeted treatment assignment probabilities across all strata in Jiang et al. (2022) and stacks them in an nx1 vector.

Usage

```
pihat(A, S, stratnum = NULL)
```

Arguments

A	A nx1 vector.
S	A nx1 vector.
stratnum	A nx1 vector about the unique strara numbers, the default value is NULL.

Value

A nx1 cector, each element corresponds to the targeted treatment assignment probabilities across all strata in Jiang et al. (2022).

References

Jiang L, Linton O B, Tang H, Zhang Y. Improving estimation efficiency via regression-adjustment in covariate-adaptive randomizations with imperfect compliance [J]. 2022.

Examples

```
DGP <-FuncDGP(dgptype = 1,rndflag = 2,n = 100,g = 4,pi = c(0.5, 0.5, 0.5, 0.5))
A <- DGP[["A"]]
S <- DGP[["S"]]
pihat(A = A, S = S)
```

---

splinebasis — *For each column of an input matrix, elements which are less than the median of that column are set to 0, leaving the rest of the elements unchanged*

Description

For each column of an input matrix, elements which are less than the median of that column are set to 0, leaving the rest of the elements unchanged.

Usage

```
splinebasis(X)
```

Arguments

X	The extra covariates, a n x K matrix. No constant included.

Value

H A n x K matrix. All elements of the X that are less than the median of their corresponding columns are set to 0, leaving the rest unchanged.

Examples

```
library(pracma)
X <- rand(4,4)
H <- splinebasis(X = X)
```

---

stanE *Compute the Estimated Standard Error of the Input Estimator*

Description

stanE Computes the estimated standard error of the input estimator.

Usage

```
stanE(muY1, muY0, muD1, muD0, A, S, Y, D, tauhat, stratnum = NULL)
```

Arguments

muY1	A nx1 vector of hat{mu}^Y(A=1)s.
muY0	A nx1 vector of hat{mu}^Y(A=0)s.
muD1	A nx1 vector of hat{mu}^D(A=1)s.
muD0	A nx1 vector of hat{mu}^D(A=0)s.
A	A nx1 vector. Each of its elements is the treatment assignment of the corresponding observation.
S	A nx1 vector. Each of its elements is the stratum of corresponding observation.
Y	A nx1 vector. Each of its elements is the observed outcome of interest of corresponding observation.
D	A nx1 vector. Each of its elements is is a binary random variable indicating whether the individual i received treatment (Di = 1) or not (Di = 0) in the actual study.
tauhat	A scalar. LATE estimate.
stratnum	A scalar. Number of stratum.

Value

A scalar. The estimated standard deviation in Jiang et al. (2022).

References

Jiang L, Linton O B, Tang H, Zhang Y. Improving estimation efficiency via regression-adjustment in covariate-adaptive randomizations with imperfect compliance [J]. 2022.

Examples

```
DGP <- FuncDGP(dgptype = 1, rndflag = 1, n = 200, g = 4, pi = c(0.5,0.5,0.5,0.5))
muY1 <- DGP[["Y1"]]
muY0 <- DGP[["Y0"]]
muD1 <- DGP[["D1"]]
muD0 <- DGP[["D0"]]
A <- DGP[["A"]]
S <- DGP[["S"]]
Y <- DGP[["Y"]]
D <- DGP[["D"]]
tauhat <- tau(muY1, muY0, muD1, muD0, A, S, Y, D)
stanE(muY1, muY0, muD1, muD0, A, S, Y, D, tauhat)
```

---

tau — *Compute Estimated LATE*

Description

Computes the estimated LATE in Jiang et al. (2022).

Usage

```
tau(muY1, muY0, muD1, muD0, A, S, Y, D, stratnum = NULL)
```

Arguments

muY1	A nx1 vector of hat{mu}^Y(A=1)s.
muY0	A nx1 vector of hat{mu}^Y(A=0)s.
muD1	A nx1 vector of hat{mu}^D(A=1)s.
muD0	A nx1 vector of hat{mu}^D(A=0)s.
A	A nx1 vector. Each of its elements is the treatment assignment of the corresponding observation.
S	A nx1 vector. Each of its elements is the stratum of corresponding observation.
Y	A nx1 vector. Each of its elements is the observed outcome of interest of corresponding observation.
D	A nx1 vector. Each of its elements is is a binary random variable indicating whether the individual i received treatment (Di = 1) or not (Di = 0) in the actual study.
stratnum	A nx1 vector about the unique strata numbers, the default value is NULL.

Value

A scalar. LATE estimate.

References

Jiang L, Linton O B, Tang H, Zhang Y. Improving estimation efficiency via regression-adjustment in covariate-adaptive randomizations with imperfect compliance [J]. 2022.

Examples

```
DGP <- FuncDGP(dgptype = 1, rndflag = 1, n = 200, g = 4, pi = c(0.5, 0.5, 0.5, 0.5))
muY1 <- DGP[["Y1"]]
muY0 <- DGP[["Y0"]]
muD1 <- DGP[["D1"]]
muD0 <- DGP[["D0"]]
A <- DGP[["A"]]
S <- DGP[["S"]]
Y <- DGP[["Y"]]
D <- DGP[["D"]]
tau(muY1, muY0, muD1, muD0, A, S, Y, D)
```

TrueValue *Calculate the True LATE tau.*

Description

Calculate the true LATE tau in Jiang et al. (2022).

Usage

```
TrueValue(dgptype, vIdx, n, g, pi)
```

Arguments

dgptype	A scalar. The value can be string 1, 2, or 3, respectively corresponding to the three random data generation methods in the paper (See Jiang et al. (2022)for DGP details)
vIdx	A 1xR vector. The authors set vIdx=[1 2 3 4]. Every number declares the method of covariate-adaptive randomization which simulates the LATE across different CAR schemes: 1-SRS; 2-WEI; 3-BCD; 4-SBR.
n	Sample size.
g	Number of strata. The authors set g=4 in Jiang et al. (2022).
pi	Targeted assignment probability across strata.

Value

A list containing two vectors named tau and mPort. tau is a 1xR vector which Simulated true LATE effect, mPort is a 3xR vector. The 1st row of mPort: the LATE of never takers across varies CAR schemes, the 2nd row of mPort: the LATE of compilers across varies CAR schemes, the 3rd row of mPort: the LATE of always takers across varies CAR schemes.

References

Jiang L, Linton O B, Tang H, Zhang Y. Improving estimation efficiency via regression-adjustment in covariate-adaptive randomizations with imperfect compliance [J]. 2022.

Examples

```
TrueValue(dgptype = 1, vIdx = c(1,2,3,4), n=100, g = 4, pi = c(0.5,0.5,0.5,0.5))
TrueValue(dgptype = 2, vIdx = c(1,2,3,4), n=100, g = 4, pi = c(0.5,0.5,0.5,0.5))
TrueValue(dgptype = 3, vIdx = c(1,2,3,4), n=100, g = 4, pi = c(0.5,0.5,0.5,0.5))
```

Index