Package ‘causalweight’

June 15, 2020

Type Package

Title Causal Inference Based on Inverse Probability Weighting, Doubly Robust Estimation, and Double Machine Learning

Version 0.2.1

Author Hugo Bodory and Martin Huber

Maintainer Hugo Bodory <hugo.bodory@unifr.ch>

Description Various estimators of causal effects based on inverse probability weighting, doubly robust estimation, and double machine learning. Specifically, the package includes methods for estimating average treatment effects, direct and indirect effects in causal mediation analysis, and dynamic treatment effects. The models refer to studies of Froelich (2007) <doi:10.1016/j.jeconom.2006.06.004>, Huber (2012) <doi:10.3102/1076998611411917>, Huber (2014) <doi:10.1080/07474938.2013.806197>, Huber (2014) <doi:10.1002/jae.2341>, Froelich and Huber (2017) <doi:10.1111/rssb.12232>, Hsu, Huber, Lee, and Lettry (2020) <doi:10.1002/jae.2765>, and others.

License MIT + file LICENSE

Encoding UTF-8

LazyData true

RoxygenNote 7.1.0

Depends R (>= 3.5.0)

Imports mvtnorm, np, LARF, hdm, SuperLearner, glmnet, ranger, xgboost, e1071

Suggests knitr, rmarkdown

VignetteBuilder knitr

NeedsCompilation no

Repository CRAN

Date/Publication 2020-06-15 12:10:11 UTC

R topics documented:

`attrlateweight` *Local average treatment effect estimation in multiple follow-up periods with outcome attrition based on inverse probability weighting*

Description

Instrumental variable-based evaluation of local average treatment effects using weighting by the inverse of the instrument propensity score.

Usage

```
attrlateweight(
  y1,
  y2,
  s1,
  s2,
  d,
  z,
  x0,
  x1,
  weightmax = 0.1,
  boot = 1999,
  cluster = NULL
)
```

Arguments

`y1`	Outcome variable in the first outcome period.
`y2`	Outcome variable in the second outcome period.

s1	Selection indicator for first outcome period. Must be one if y1 is observed (non-missing) and zero if y1 is not observed (missing).
s2	Selection indicator for second outcome period. Must be one if y1 is observed (non-missing) and zero if y1 is not observed (missing).
d	Treatment, must be binary (either 1 or 0), must not contain missings.
z	Instrument for the endogenous treatment, must be binary (either 1 or 0), must not contain missings.
x0	Baseline (pre-instrument) confounders of the instrument and outcome, must not contain missings.
x1	Confounders in outcome period 1 (may include outcomes of period 1 y1)
weightmax	Trimming rule based on the maximum relative weight a single observation may obtain in estimation - observations with higher weights are discarded. Default is 0.1 (no observation can be assigned more than 10 percent of weights)
boot	Number of bootstrap replications for estimating standard errors. Default is 1999.
cluster	A cluster ID for block or cluster bootstrapping when units are clustered rather than iid. Must be numerical. Default is NULL (standard bootstrap without clustering).

Details

Estimation of local average treatment effects of a binary endogenous treatment on outcomes in two follow up periods that are prone to attrition. Treatment endogeneity is tackled by a binary instrument that is assumed to be conditionally valid given observed baseline confounders x0. Outcome attrition is tackled by either assuming that it is missing at random (MAR), i.e. selection w.r.t. observed variables d, z, x0, x1 (in the case of y2), and s1 (in the case of y2); or by assuming latent ignorability (LI), i.e. selection w.r.t. the treatment compliance type as well as z, x0, x1 (in the case of y2), and s1 (in the case of y2). Units are weighted by the inverse of their conditional instrument and selection propensities, which are estimated by probit regression. Standard errors are obtained by bootstrapping the effect.

Value

An attrlateweight object contains one component results:

results: a 4X4 matrix containing the effect estimates in the first row ("effects"), standard errors in the second row ("se"), p-values in the third row ("p-value"), and the number of trimmed observations due to too large weights in the fourth row ("trimmed obs"). The first column provides the local average treatment effect (LATE) on y1 among compliers under missingness at random (MAR). The second column provides the local average treatment effect (LATE) on y2 under missingness at random (MAR). The third column provides the local average treatment effect (LATE) on y1 under latent ignorability (LI). The forth column provides the local average treatment effect (LATE) on y2 under latent ignorability (LI).

References

Frölich, M., Huber, M. (2014): "Treatment Evaluation With Multiple Outcome Periods Under Endogeneity and Attrition", Journal of the American Statistical Association, 109, 1697-1711.

Examples

```
# A little example with simulated data (4000 observations)
## Not run:
n=4000
e=(rmvnorm(n,rep(0,3), matrix(c(1,0.3,0.3,  0.3,1,0.3,  0.3,0.3,1),3,3) ))
x0=runif(n,0,1)
z=(0.25*x0+rnorm(n)>0)*1
d=(1.2*z-0.25*x0+e[,1]>0.5)*1
y1_star=0.5*x0+0.5*d+e[,2]
s1=(0.25*x0+0.25*d+rnorm(n)>-0.5)*1
y1=s1*y1_star
x1=(0.5*x0+0.5*rnorm(n))
y2_star=0.5*x0+x1+d+e[,3]
s2=s1*((0.25*x0+0.25*x1+0.25*d+rnorm(n)>-0.5)*1)
y2=s2*y2_star
# The true LATEs on y1 and y2 are equal to 0.5 and 1, respectively.
output=attrlateweight(y1=y1,y2=y2,s1=s1,s2=s2,d=d,z=z,x0=x0,x1=x1,boot=19)
round(output$results,3)
## End(Not run)
```

---

didweight — *Difference-in-differences based on inverse probability weighting*

Description

Difference-in-differences-based estimation of the average treatment effect on the treated in the post-treatment period, given a binary treatment with one pre- and one post-treatment period. Permits controlling for differences in observed covariates across treatment groups and/or time periods based on inverse probability weighting.

Usage

```
didweight(y, d, t, x = NULL, boot = 1999, trim = 0.05, cluster = NULL)
```

Arguments

`y`	Dependent variable, must not contain missings.
`d`	Treatment, must be binary (either 1 or 0), must not contain missings.
`t`	Time period, must be binary, 0 for pre-treatment and 1 for post-treatment, must not contain missings.
`x`	Covariates to be controlled for by inverse probability weighting. Default is `NULL`.
`boot`	Number of bootstrap replications for estimating standard errors. Default is 1999.
`trim`	Trimming rule for discarding observations with extreme propensity scores in the 3 reweighting steps, which reweight (1) treated in the pre-treatment period, (2) non-treated in the post-treatment period, and (3) non-treated in the pre-treatment period according to the covariate distribution of the treated in the post-treatment period. Default is 0.05, implying that observations with a probability lower than 5 percent of not being treated in some weighting step are discarded.

`cluster` A cluster ID for block or cluster bootstrapping when units are clustered rather than iid. Must be numerical. Default is `NULL` (standard bootstrap without clustering).

Details

Estimation of the average treatment effect on the treated in the post-treatment period based Difference-in-differences. Inverse probability weighting is used to control for differences in covariates across treatment groups and/or over time. That is, (1) treated observations in the pre-treatment period, (2) non-treated observations in the post-treatment period, and (3) non-treated observations in the pre-treatment period are reweighted according to the covariate distribution of the treated observations in the post-treatment period. The respective propensity scores are obtained by probit regressions.

Value

A didweight object contains 4 components, `eff`, `se`, `pvalue`, and `ntrimmed`.

`eff`: estimate of the average treatment effect on the treated in the post-treatment period.

`se`: standard error obtained by bootstrapping the effect.

`pvalue`: p-value based on the t-statistic.

`ntrimmed`: total number of discarded (trimmed) observations in any of the 3 reweighting steps due to extreme propensity score values.

References

Abadie, A. (2005): "Semiparametric Difference-in-Differences Estimators", The Review of Economic Studies, 72, 1-19.

Lechner, M. (2011): "The Estimation of Causal Effects by Difference-in-Difference Methods", Foundations and Trends in Econometrics, 4, 165-224.

Examples

```
# A little example with simulated data (4000 observations)
## Not run:
n=4000                            # sample size
t=1*(rnorm(n)>0)                  # time period
u=rnorm(n)                        # time constant unobservable
x=0.5*t+rnorm(n)                  # time varying covariate
d=1*(x+u+rnorm(n)>0)              # treatment
y=d*t+d+t+x+u                     # outcome
# The true effect equals 1
didweight(y=y,d=d,t=t,x=x, boot=199)
## End(Not run)
```

dyntreatDML *Dynamic treatment effect evaluation with double machine learning*

Description

Dynamic treatment effect estimation for assessing the average effects of sequences of treatments (consisting of two sequential treatments). Combines estimation based on (doubly robust) efficient score functions with double machine learning to control for confounders in a data-driven way.

Usage

```
dyntreatDML(
  y2,
  d1,
  d2,
  x0,
  x1,
  s = NULL,
  d1treat = 1,
  d2treat = 1,
  d1control = 0,
  d2control = 0,
  trim = 0.01,
  MLmethod = "lasso",
  fewsplits = FALSE
)
```

Arguments

`y2`	Dependent variable in the second period (=outcome period), must not contain missings.
`d1`	Treatment in the first period, must be discrete, must not contain missings.
`d2`	Treatment in the second period, must be discrete, must not contain missings.
`x0`	Covariates in the baseline period (prior to the treatment in the first period), must not contain missings.
`x1`	Covariates in the first period (prior to the treatment in the second period), must not contain missings.
`s`	Indicator function for defining a subpopulation for whom the treatment effect is estimated as a function of the subpopulation's distribution of `x0`. Default is `NULL` (estimation of the treatment effect in the total population).
`d1treat`	Value of the first treatment in the treatment sequence. Default is 1.
`d2treat`	Value of the second treatment in the treatment sequence. Default is 1.
`d1control`	Value of the first treatment in the control sequence. Default is 0.
`d2control`	Value of the second treatment in the control sequence. Default is 0.

trim — Trimming rule for discarding observations with products of treatment propensity scores in the first and second period that are smaller than trim (to avoid too small denominators in weighting by the inverse of the propensity scores). Default is 0.01.

MLmethod — Machine learning method for estimating the nuisance parameters based on the SuperLearner package. Must be either "lasso" (default) for lasso estimation, "randomforest" for random forests, "xgboost" for xg boosting, "svm" for support vector machines, "ensemble" for using an ensemble algorithm based on all previously mentioned machine learners, or "parametric" for linear or logit regression.

fewsplits — If set to TRUE, the same training data are used for estimating a nested model of conditional mean outcomes, namely E[E[y2|d1,d2,x0,x1]|d1,x0]. If fewsplits is FALSE, the training data are split for the sequential estimation of the nested model. Default of fewsplits is FALSE.

Details

Estimation of the causal effects of sequences of two treatments under sequential conditional independence, assuming that all confounders of the treatment in either period and the outcome of interest are observed. Estimation is based on the (doubly robust) efficient score functions for potential outcomes, see e.g. Bodory, Huber, and Laffers (2020), in combination with double machine learning with cross-fitting, see Chernozhukov et al (2018). To this end, one part of the data is used for estimating the model parameters of the treatment and outcome equations based machine learning. The other part of the data is used for predicting the efficient score functions. The roles of the data parts are swapped (using 3-fold cross-fitting) and the average dynamic treatment effect is estimated based on averaging the predicted efficient score functions in the total sample. Standard errors are based on asymptotic approximations using the estimated variance of the (estimated) efficient score functions.

Value

A dyntreatDML object contains ten components, effect, se, pval, ntrimmed, meantreat, meancontrol, psd1treat, psd2treat, psd1control, and psd2control :

effect: estimate of the average effect of the treatment sequence.

se: standard error of the effect estimate.

pval: p-value of the effect estimate.

ntrimmed: number of discarded (trimmed) observations due to low products of propensity scores.

meantreat: Estimate of the mean potential outcome under the treatment sequence.

meancontrol: Estimate of the mean potential outcome under the control sequence.

psd1treat: P-score estimates for first treatment in treatment sequence.

psd2treat: P-score estimates for second treatment in treatment sequence.

psd1control: P-score estimates for first treatment in control sequence.

psd2control: P-score estimates for second treatment in control sequence.

References

Bodory, H., Huber, M., Laffers, L. (2020): "Double machine learning for (weighted) dynamic treatment effects", working paper, University of Fribourg.

Chernozhukov, V., Chetverikov, D., Demirer, M., Duflo, E., Hansen, C., Newey, W., Robins, J. (2018): "Double/debiased machine learning for treatment and structural parameters", The Econometrics Journal, 21, C1-C68.

van der Laan, M., Polley, E., Hubbard, A. (2007): "Super Learner", Statistical Applications in Genetics and Molecular Biology, 6.

Examples

```
# A little example with simulated data (2000 observations)
## Not run:
n=2000
# sample size
p0=10
# number of covariates at baseline
s0=5
# number of covariates that are confounders at baseline
p1=10
# number of additional covariates in period 1
s1=5
# number of additional covariates that are confounders in period 1
x0=matrix(rnorm(n*p0),ncol=p0)
# covariate matrix at baseline
beta0=c(rep(0.25,s0), rep(0,p0-s0))
# coefficients determining degree of confounding for baseline covariates
d1=(x0%*%beta0+rnorm(n)>0)*1
# equation of first treatment in period 1
x1=matrix(rnorm(n*p1),ncol=p1)
# covariate matrix for covariates of period 1
beta1=c(rep(0.25,s1), rep(0,p1-s1))
# coefficients determining degree of confounding for additonal covariates of period 1
d2=(x0%*%beta0+x1%*%beta1+0.5*d1+rnorm(n)>0)*1
# equation of second treatment in period 2
y2=x0%*%beta0+x1%*%beta1+1*d1+0.5*d2+rnorm(n)
# outcome equation in period 2
output=dyntreatDML(y2=y2,d1=d1,d2=d2,x0=x0,x1=x1,
       d1treat=1,d2treat=1,d1control=0,d2control=0)
cat("dynamic ATE: ",round(c(output$effect),3),", standard error: ",
    round(c(output$se),3), ", p-value: ",round(c(output$pval),3))
output$ntrimmed
# The true effect of the treatment sequence is 1.5
## End(Not run)
```

---

ivnr — *Instrument-based treatment evaluation under endogeneity and non-response bias*

Description

Non- and semiparaemtric treatment effect estimation under treatment endogeneity and selective non-response in the outcome based on a binary instrument for the treatment and a continous instrument for response.

Usage

```
ivnr(
  y,
  d,
  r,
  z1,
  z2,
  x = NULL,
  xpar = NULL,
  ruleofthumb = 1,
  wgtfct = 2,
  rtype = "ll",
  numresprob = 20,
  boot = 499,
  estlate = TRUE,
  trim = 0.01
)
```

Arguments

`y`	Dependent variable.
`d`	Treatment, must be binary and must not contain missings.
`r`	Response, must be a binary indicator for whether the outcome is observed.
`z1`	Binary instrument for the treatment, must not contain missings.
`z2`	Continuous instrument for response, must not contain missings.
`x`	A data frame of covariates to be included in the nonparametric estimation, must not contain missings. Factors and ordered varaibles must be appropriately defined as such by `factor()` and `ordered()`. Default is `NULL` (no covariates included). Covariates are only considered if both `x` and `xpar` are not `NULL`.
`xpar`	Covariates to be included in the semiparametric estimation, must not contain missings. Default is `NULL` (no covariates included). Covariates are only considered if both `x` and `xpar` are not `NULL`.
`ruleofthumb`	If 1, bandwidth selection in any kernel function is based on the Silverman (1986) rule of thumb. Otherwise, least squares cross-validation is used. Default is 1.
`wgtfct`	Weighting function to be used in effect estimation. If set to 1, equation (18) in Fricke et al (2020) is used as weight. If set to 2, equation (19) in Fricke et al (2020) is used as weight. If set to 3, the median of LATEs across values of response probabilities `numresprob` is used. Default is 2.
`rtype`	Regression type used for continuous outcomes in the kernel regressions. Either `"ll"` for local linear or `"lc"` for local constant regression. Default is `"ll"`.

`numresprob`	number of response probabilities at which the effects are evaluated. An equidistant grid is constructed based on the number provided. Default is 20.
`boot`	Number of bootstrap replications for estimating standard errors of the effects. Default is 499.
`estlate`	If set to `TRUE` the local average treatment effect on compliers (LATE) is estimated, otherwise the average treatment effect (ATE) is estimated. Default is `TRUE`.
`trim`	Trimming rule for too extreme denominators in the weighting functions or inverses of products of conditional treatment probabilities. Values below `trim` are set to `trim` to avoid values that are too close to zero in any denominator. Default is 0.01.

Details

Non- and semiparametric treatment effect estimation under treatment endogeneity and selective non-response in the outcome based on a binary instrument for the treatment and a continuous instrument for response. The effects are estimated both semi-parametrically (using probit and OLS for the estimation of plug-in parameters like conditional probabilities and outcomes) and fully non-parametrically (based on kernel regression for any conditional probability/mean). Besides the instrument-based estimates, results are also presented under a missing-at-random assumption (MAR) when not using the instrument `z2` for response (but only `z1` for the treatment). See Fricke et al. (2020) for further details.

Value

A `ivnr` object contains one output component:

`output`: The first row provides the effect estimates under non- and semi-parametric estimation using both instruments, see `"nonpara (L)ATE IV"` and `"semipara (L)ATE IV"` as well as under a missing-at-random assumption for response when using only the first instrument for the treatment, see `"nonpara (L)ATE MAR"` and `"semipara (L)ATE MAR"`. The second row provides the standard errors based on bootstrapping the effects. The third row provides the p-values based on the t-statistics.

References

Fricke, H., Frölich, M., Huber, M., Lechner, M. (2020): "Endogeneity and non-response bias in treatment evaluation - nonparametric identification of causal effects by instruments", Journal of Applied Econometrics, forthcoming.

Examples

```
# A little example with simulated data (1000 observations)
## Not run:
n=1000          # sample size
e<-(rmvnorm(n,rep(0,3), matrix(c(1,0.5,0.5,  0.5,1,0.5,  0.5,0.5,1),3,3)))
# correlated error term of treatment, response, and outcome equation
x=runif(n,-0.5,0.5)           # observed confounder
z1<-(-0.25*x+rnorm(n)>0)*1    # binary instrument for treatment
z2<- -0.25*x+rnorm(n)         # continuous instrument for selection
```

```
d<-(z1-0.25*x+e[,1]>0)*1      # treatment equation
 y_star<- -0.25*x+d+e[,2]     # latent outcome
 r<-(-0.25*x+z2+d+e[,3]>0)*1  # response equation
 y=y_star                     # observed outcome
 y[r==0]=0                    # nonobserved outcomes are set to zero
 # The true treatment effect is 1
 ivnr(y=y,d=d,r=r,z1=z1,z2=z2,x=x,xpar=x,numresprob=4,boot=39)
## End(Not run)
```

`lateweight` *Local average treatment effect estimation based on inverse probability weighting*

Description

Instrumental variable-based evaluation of local average treatment effects using weighting by the inverse of the instrument propensity score.

Usage

```
lateweight(
  y,
  d,
  z,
  x,
  LATT = FALSE,
  trim = 0.05,
  logit = FALSE,
  boot = 1999,
  cluster = NULL
)
```

Arguments

y	Dependent variable, must not contain missings.
d	Treatment, must be binary (either 1 or 0), must not contain missings.
z	Instrument for the endogenous treatment, must be binary (either 1 or 0), must not contain missings.
x	Confounders of the instrument and outcome, must not contain missings.
LATT	If FALSE, the local average treatment effect (LATE) among compliers (whose treatment reacts to the instrument) is estimated. If TRUE, the local average treatment effect on the treated compliers (LATT) is estimated. Default is FALSE.
trim	Trimming rule for discarding observations with extreme propensity scores. If `LATT=FALSE`, observations with Pr(Z=1|X)<`trim` or Pr(Z=1|X)>(1-`trim`) are dropped. If `LATT=TRUE`, observations with Pr(Z=1|X)>(1-`trim`) are dropped. Default is 0.05.

`logit`	If FALSE, probit regression is used for propensity score estimation. If TRUE, logit regression is used. Default is FALSE.
`boot`	Number of bootstrap replications for estimating standard errors. Default is 1999.
`cluster`	A cluster ID for block or cluster bootstrapping when units are clustered rather than iid. Must be numerical. Default is NULL (standard bootstrap without clustering).

Details

Estimation of local average treatment effects of a binary endogenous treatment based on a binary instrument that is conditionally valid, implying that all confounders of the instrument and the outcome are observed. Units are weighted by the inverse of their conditional instrument propensities given the observed confounders, which are estimated by probit or logit regression. Standard errors are obtained by bootstrapping the effect.

Value

A lateweight object contains 10 components, `effect`, `se.effect`, `pval.effect`, `first`, `se.first`, `pval.first`, `ITT`, `se.ITT`, `pval.ITT`, and `ntrimmed`:

`effect`: local average treatment effect (LATE) among compliers if `LATT=FALSE` or the local average treatment effect on treated compliers (LATT) if `LATT=TRUE`.

`se.effect`: bootstrap-based standard error of the effect.

`pval.effect`: p-value of the effect.

`first`: first stage estimate of the complier share if `LATT=FALSE` or the first stage estimate among treated if `LATT=TRUE`.

`se.first`: bootstrap-based standard error of the first stage effect.

`pval.first`: p-value of the first stage effect.

`ITT`: intention to treat effect (ITT) of `z` on `y` if `LATT=FALSE` or the ITT among treated if `LATT=TRUE`.

`se.ITT`: bootstrap-based standard error of the ITT.

`pval.ITT`: p-value of the ITT.

`ntrimmed`: number of discarded (trimmed) observations due to extreme propensity score values.

References

Frölich, M. (2007): "Nonparametric IV estimation of local average treatment effects with covariates", Journal of Econometrics, 139, 35-75.

Examples

```
# A little example with simulated data (10000 observations)
## Not run:
n=10000
u=rnorm(n)
x=rnorm(n)
z=(0.25*x+rnorm(n)>0)*1
d=(z+0.25*x+0.25*u+rnorm(n)>0.5)*1
```

```
y=0.5*d+0.25*x+u
# The true LATE is equal to 0.5
output=lateweight(y=y,d=d,z=z,x=x,trim=0.05,LATT=FALSE,logit=TRUE,boot=19)
cat("LATE: ",round(c(output$effect),3),", standard error: ",
            round(c(output$se.effect),3), ", p-value: ",
            round(c(output$pval.effect),3))
output$ntrimmed
## End(Not run)
```

`medDML` *Causal mediation analysis with double machine learning*

Description

Causal mediation analysis (evaluation of natural direct and indirect effects) for a binary treatment and one or several mediators using double machine learning to control for confounders based on (doubly robust) efficient score functions for potential outcomes.

Usage

```
medDML(
  y,
  d,
  m,
  x,
  k = 3,
  trim = 0.05,
  order = 1,
  multmed = TRUE,
  fewsplits = FALSE
)
```

Arguments

`y`	Dependent variable, must not contain missings.
`d`	Treatment, must be binary (either 1 or 0), must not contain missings.
`m`	Mediator, must not contain missings. May be a scalar or a vector of binary, categorical, or continuous variables if `multmed` is `TRUE`. Must be a binary scalar if `multmed` is `FALSE`.
`x`	(Potential) pre-treatment confounders of the treatment, mediator, and/or outcome, must not contain missings.
`k`	Number of folds in k-fold cross-fitting if `multmed` is `FALSE`. `k`-1 folds are used for estimating the model parameters of the treatment, mediator, and outcome equations and one fold is used for predicting the efficient score functions. The roles of the folds are swapped. Default for `k` is 3. If `multmed` is `TRUE`, then 3-fold cross-valdiation is used, irrespective of the number provided in `k` (i.e. `k` is ignored if `multmed` is `TRUE`).

trim
: Trimming rule for discarding observations with extreme conditional treatment or mediator probabilities (or products thereof). Observations with (products of) conditional probabilities that are smaller than `trim` in any denominator of the potential outcomes are dropped. Default is 0.05.

order
: If set to an integer larger than 1, then polynomials of that order and interactions (using the power series) rather than the original control variables are used in the estimation of any conditional probability or conditional mean outcome. Polynomials/interactions are created using the `Generate.Powers` command of the `LARF` package.

multmed
: If set to `TRUE`, a representation of direct and indirect effects that avoids conditional mediator densities/probabilities is used, see Farbmacher, Huber, Langen, and Spindler (2019). This method can incorporate multiple and/or continuous mediators. If `multmed` is `FALSE`, the representation of Tchetgen Tchetgen and Shpitser (2012) is used, which involves mediator densities. In this case, the mediator must be a binary scalar. Default of `multimed` is `TRUE`.

fewsplits
: If set to `TRUE`, the same training data are used for estimating nested models of nuisance parameters, i.e. `E[Y|D=d,M,X]` and `E[E[Y|D=d,M,X]|D=1-d,X]`. If `fewsplits` is `FALSE`, the training data are split for the sequential estimation of nested models `E[Y|D=d,M,X]` and `E[E[Y|D=d,M,X]|D=1-d,X]`. This parameter is only relevant if `multmed` is `TRUE`. Default of `fewsplits` is `FALSE`.

Details

Estimation of causal mechanisms (natural direct and indirect effects) of a treatment under selection on observables, assuming that all confounders of the binary treatment and the mediator, the treatment and the outcome, or the mediator and the outcome are observed and not affected by the treatment. Estimation is based on the (doubly robust) efficient score functions for potential outcomes, see Tchetgen Tchetgen and Shpitser (2012) and Farbmacher, Huber, Langen, and Spindler (2019), as well as on double machine learning with cross-fitting, see Chernozhukov et al (2018). To this end, one part of the data is used for estimating the model parameters of the treatment, mediator, and outcome equations based on post-lasso regression, using the `rlasso` and `rlassologit` functions (for conditional means and probabilities, respectively) of the `hdm` package with default settings. The other part of the data is used for predicting the efficient score functions. The roles of the data parts are swapped and the direct and indirect effects are estimated based on averaging the predicted efficient score functions in the total sample. Standard errors are based on asymptotic approximations using the estimated variance of the (estimated) efficient score functions.

Value

A medDML object contains two components, `results` and `ntrimmed`:

`results`: a 3X6 matrix containing the effect estimates in the first row ("effects"), standard errors in the second row ("se"), and p-values in the third row ("p-value"). The first column provides the total effect, namely the average treatment effect (ATE). The second and third columns provide the direct effects under treatment and control, respectively ("dir.treat", "dir.control"). The fourth and fifth columns provide the indirect effects under treatment and control, respectively ("indir.treat", "indir.control"). The sixth column provides the estimated mean under non-treatment ("Y(0,M(0))").

`ntrimmed`: number of discarded (trimmed) observations due to extreme conditional probabilities.

References

Chernozhukov, V., Chetverikov, D., Demirer, M., Duflo, E., Hansen, C., Newey, W., Robins, J. (2018): "Double/debiased machine learning for treatment and structural parameters", The Econometrics Journal, 21, C1-C68.

Farbmacher, H., Huber, M., Langen, H., and Spindler, M. (2019): "Causal mediation analysis with double machine learning", working paper, University of Fribourg.

Tchetgen Tchetgen, E. J., and Shpitser, I. (2012): "Semiparametric theory for causal mediation analysis: efficiency bounds, multiple robustness, and sensitivity analysis", The Annals of Statistics, 40, 1816-1845.

Tibshirani, R. (1996): "Regression shrinkage and selection via the lasso", Journal of the Royal Statistical Society: Series B, 58, 267-288.

Examples

```
# A little example with simulated data (10000 observations)
## Not run:
n=10000                          # sample size
p=100                            # number of covariates
s=2                              # number of covariates that are confounders
x=matrix(rnorm(n*p),ncol=p)      # covariate matrix
beta=c(rep(0.25,s), rep(0,p-s))  # coefficients determining degree of confounding
d=(x%*%beta+rnorm(n)>0)*1        # treatment equation
m=(x%*%beta+0.5*d+rnorm(n)>0)*1  # mediator equation
y=x%*%beta+0.5*d+m+rnorm(n)      # outcome equation
# The true direct effects are equal to 0.5, the indirect effects equal to 0.19
output=medDML(y=y,d=d,m=m,x=x)
round(output$results,3)
output$ntrimmed
## End(Not run)
```

---

medlateweight — *Causal mediation analysis with instruments for treatment and mediator based on weighting*

Description

Causal mediation analysis (evaluation of natural direct and indirect effects) with instruments for a binary treatment and a continuous mediator based on weighting as suggested in Frölich and Huber (2017), Theorem 1.

Usage

```
medlateweight(
  y,
  d,
  m,
  zd,
```

```
  zm,
  x,
  trim = 0.1,
  csquared = FALSE,
  boot = 1999,
  cminobs = 40,
  bwreg = NULL,
  bwm = NULL,
  logit = FALSE,
  cluster = NULL
)
```

Arguments

`y`	Dependent variable, must not contain missings.
`d`	Treatment, must be binary (either 1 or 0), must not contain missings.
`m`	Mediator(s),must be a continuous scalar, must not contain missings.
`zd`	Instrument for the treatment, must be binary (either 1 or 0), must not contain missings.
`zm`	Instrument for the mediator, must contain at least one continuous element, may be a scalar or a vector, must not contain missings. If no user-specified bandwidth is provided for the regressors when estimating the conditional cumulative distribution function F(M|Z2,X), i.e. if `bwreg=NULL`, then `zm` must be exclusively numeric.
`x`	Pre-treatment confounders, may be a scalar or a vector, must not contain missings. If no user-specified bandwidth is provided for the regressors when estimating the conditional cumulative distribution function F(M|Z2,X), i.e. if `bwreg=NULL`, then `x` must be exclusively numeric.
`trim`	Trimming rule for discarding observations with extreme weights. Discards observations whose relative weight would exceed the value in `trim` in the estimation of any of the potential outcomes. Default is 0.1 (i.e. a maximum weight of 10 percent per observation).
`csquared`	If TRUE, then not only the control function C, but also its square is used as regressor in any estimated function that conditions on C. Default is FALSE.
`boot`	Number of bootstrap replications for estimating standard errors. Default is 1999.
`cminobs`	Minimum number of observations to compute the control function C, see the numerator of equation (7) in Frölich and Huber (2017). A larger value increases boundary bias when estimating the control function for lower values of M, but reduces the variance. Default is 40, but should be adapted to sample size and the number of variables in Z2 and X.
`bwreg`	Bandwidths for `zm` and `x` in the estimation of the conditional cumulative distribution function F(M|Z2,X) based on the np package by Hayfield and Racine (2008). The length of the numeric vector must correspond to the joint number of elements in `zm` and `x` and will be used both in the original sample for effect estimation and in bootstrap samples to compute standard errors. If set to `NULL`, then the rule of thumb is used for bandwidth calculation, see the np package

for details. In the latter case, all elements in the regressors must be numeric. Default is `NULL`.

`bwm` Bandwidth for `m` in the estimation of the conditional cumulative distribution function F(M|Z2,X) based on the np package by Hayfield and Racine (2008). Must be scalar and will be used both in the original sample for effect estimation and in bootstrap samples to compute standard errors. If set to `NULL`, then the rule of thumb is used for bandwidth calculation, see the np package for details. Default is `NULL`.

`logit` If FALSE, probit regression is used for any propensity score estimation. If TRUE, logit regression is used. Default is FALSE.

`cluster` A cluster ID for block or cluster bootstrapping when units are clustered rather than iid. Must be numerical. Default is NULL (standard bootstrap without clustering).

Details

Estimation of causal mechanisms (natural direct and indirect effects) of a binary treatment among treatment compliers based on distinct instruments for the treatment and the mediator. The treatment and its instrument are assumed to be binary, while the mediator and its instrument are assumed to be continuous, see Theorem 1 in Frölich and Huber (2017). The instruments are assumed to be conditionally valid given a set of observed confounders. A control function is used to tackle mediator endogeneity. Standard errors are obtained by bootstrapping the effects.

Value

A medlateweight object contains two components, `results` and `ntrimmed`:

`results`: a 3x7 matrix containing the effect estimates in the first row ("effects"), standard errors in the second row ("se"), and p-values in the third row ("p-value"). The first column provides the total effect, namely the local average treatment effect (LATE) on the compliers. The second and third columns provide the direct effects under treatment and control, respectively ("dir.treat", "dir.control"). The fourth and fifth columns provide the indirect effects under treatment and control, respectively ("indir.treat", "indir.control"). The sixth and seventh columns provide the parametric direct and indirect effect estimates ("dir.para", "indir.para") without intercation terms, respectively. For the parametric estimates, probit or logit specifications are used for the treatment model and OLS specifications for the mediator and outcome models.

`ntrimmed`: number of discarded (trimmed) observations due to large weights.

References

Frölich, M. and Huber, M. (2017): "Direct and indirect treatment effects: Causal chains and mediation analysis with instrumental variables", Journal of the Royal Statistical Society Series B, 79, 1645–1666.

Examples

```
# A little example with simulated data (3000 observations)
## Not run:
n=3000; sigma=matrix(c(1,0.5,0.5,0.5,1,0.5,0.5,0.5,1),3,3)
```

```
e=(rmvnorm(n,rep(0,3),sigma))
x=rnorm(n)
zd=(0.5*x+rnorm(n)>0)*1
d=(-1+0.5*x+2*zd+e[,3]>0)
zm=0.5*x+rnorm(n)
m=(0.5*x+2*zm+0.5*d+e[,2])
y=0.5*x+d+m+e[,1]
# The true direct and indirect effects on compliers are equal to 1 and 0.5, respectively
medlateweight(y,d,m,zd,zm,x,trim=0.1,csquared=FALSE,boot=19,cminobs=40,
bwreg=NULL,bwm=NULL,logit=FALSE)
## End(Not run)
```

`medweight` *Causal mediation analysis based on inverse probability weighting with optional sample selection correction.*

Description

Causal mediation analysis (evaluation of natural direct and indirect effects) based on weighting by the inverse of treatment propensity scores as suggested in Huber (2014) and Huber and Solovyeva (2018).

Usage

```
medweight(
  y,
  d,
  m,
  x,
  w = NULL,
  s = NULL,
  z = NULL,
  selpop = FALSE,
  ATET = FALSE,
  trim = 0.05,
  logit = FALSE,
  boot = 1999,
  cluster = NULL
)
```

Arguments

`y`	Dependent variable, must not contain missings.
`d`	Treatment, must be binary (either 1 or 0), must not contain missings.
`m`	Mediator(s), may be a scalar or a vector, must not contain missings.
`x`	Pre-treatment confounders of the treatment, mediator, and/or outcome, must not contain missings.

`w` Post-treatment confounders of the mediator and the outcome. Default is `NULL`. Must not contain missings.

`s` Optional selection indicator. Must be one if `y` is observed (non-missing) and zero if `y` is not observed (missing). Default is `NULL`, implying that `y` does not contain any missings. Is ignored if `w` is not `NULL`.

`z` Optional instrumental variable(s) for selection `s`. If `NULL`, outcome selection based on observables (`x`,`d`,`m`) - known as "missing at random" - is assumed.

`selpop` Only to be used if both `s` and `z` are defined. If `TRUE`, the effects are estimated for the selected subpopulation with `s`=1 only. If `FALSE`, the effects are estimated for the total population.

`ATET` If FALSE, the average treatment effect (ATE) and the corresponding direct and indirect effects are estimated. If TRUE, the average treatment effect on the treated (ATET) and the corresponding direct and indirect effects are estimated. Default is FALSE.

`trim` Trimming rule for discarding observations with extreme propensity scores. In the absence of post-treatment confounders (w=NULL), observations with Pr(D=1|M,X)<`trim` or Pr(D=1|M,X)>(1-`trim`) are dropped. In the presence of post-treatment confounders (`w` is defined), observations with Pr(D=1|M,W,X)<`trim` or Pr(D=1|M,W,X)>(1-`trim`) are dropped. Default is 0.05. If `s` is defined (only considered if `w` is `NULL!`) and `z` is `NULL`, observations with low selection propensity scores, Pr(S=1|D,M,X)<`trim`, are discarded, too. If `s` and `z` are defined, the treatment propensity scores to be trimmed change to Pr(D=1|M,X,Pr(S=1|D,X,Z)).

`logit` If FALSE, probit regression is used for propensity score estimation. If TRUE, logit regression is used. Default is FALSE.

`boot` Number of bootstrap replications for estimating standard errors. Default is 1999.

`cluster` A cluster ID for block or cluster bootstrapping when units are clustered rather than iid. Must be numerical. Default is NULL (standard bootstrap without clustering).

Details

Estimation of causal mechanisms (natural direct and indirect effects) of a binary treatment under a selection on observables assumption assuming that all confounders of the treatment and the mediator, the treatment and the outcome, or the mediator and the outcome are observed. Units are weighted by the inverse of their conditional treatment propensities given the mediator and/or observed confounders, which are estimated by probit or logit regression. The form of weighting depends on whether the observed confounders are exclusively pre-treatment (`x`), or also contain post-treatment confounders of the mediator and the outcome (`w`). In the latter case, only partial indirect effects (from `d` to `m` to `y`) can be estimated that exclude any causal paths from `d` to `w` to `m` to `y`, see the discussion in Huber (2014). Standard errors are obtained by bootstrapping the effects. In the absence of post-treatment confounders (such that `w` is `NULL`), defining `s` allows correcting for sample selection due to missing outcomes based on the inverse of the conditional selection probability. The latter might either be related to observables, which implies a missing at random assumption, or in addition also to unobservables, if an instrument for sample selection is available. Effects are then estimated for the total population, see Huber and Solovyeva (2018) for further details.

Value

A medweight object contains two components, `results` and `ntrimmed`:

`results`: a 3X5 matrix containing the effect estimates in the first row ("effects"), standard errors in the second row ("se"), and p-values in the third row ("p-value"). The first column provides the total effect, namely the average treatment effect (ATE) if `ATET=FALSE` or the average treatment effect on the treated (ATET) if `ATET=TRUE`. The second and third columns provide the direct effects under treatment and control, respectively ("dir.treat", "dir.control"). See equation (6) if `w=NULL` (no post-treatment confounders) and equation (13) if `w` is defined, respectively, in Huber (2014). If `w=NULL`, the fourth and fifth columns provide the indirect effects under treatment and control, respectively ("indir.treat", "indir.control"), see equation (7) in Huber (2014). If `w` is defined, the fourth and fifth columns provide the partial indirect effects under treatment and control, respectively ("par.in.treat", "par.in.control"), see equation (14) in Huber (2014).

`ntrimmed`: number of discarded (trimmed) observations due to extreme propensity score values.

References

Huber, M. (2014): "Identifying causal mechanisms (primarily) based on inverse probability weighting", Journal of Applied Econometrics, 29, 920-943.

Huber, M. and Solovyeva, A. (2018): "Direct and indirect effects under sample selection and outcome attrition ", SES working paper 496, University of Fribourg.

Examples

```
# A little example with simulated data (10000 observations)
## Not run:
n=10000
x=rnorm(n)
d=(0.25*x+rnorm(n)>0)*1
w=0.2*d+0.25*x+rnorm(n)
m=0.5*w+0.5*d+0.25*x+rnorm(n)
y=0.5*d+m+w+0.25*x+rnorm(n)
# The true direct and partial indirect effects are all equal to 0.5
output=medweight(y=y,d=d,m=m,x=x,w=w,trim=0.05,ATET=FALSE,logit=TRUE,boot=19)
round(output$results,3)
output$ntrimmed
## End(Not run)
```

`medweightcont` *Causal mediation analysis with a continuous treatment based on weighting by the inverse of generalized propensity scores*

Description

Causal mediation analysis (evaluation of natural direct and indirect effects) of a continuous treatment based on weighting by the inverse of generalized propensity scores as suggested in Hsu, Huber, Lee, and Pipoz (2018).

Usage

```
medweightcont(
  y,
  d,
  m,
  x,
  d0,
  d1,
  ATET = FALSE,
  trim = 0.1,
  lognorm = FALSE,
  bw = NULL,
  boot = 1999,
  cluster = NULL
)
```

Arguments

y	Dependent variable, must not contain missings.
d	Continuous treatment, must not contain missings.
m	Mediator(s), may be a scalar or a vector, must not contain missings.
x	Pre-treatment confounders of the treatment, mediator, and/or outcome, must not contain missings.
d0	Value of d under non-treatment. Effects are based on pairwise comparisons, i.e. differences in potential outcomes evaluated at d1 and d0.
d1	Value of d under treatment. Effects are based on pairwise comparisons, i.e. differences in potential outcomes evaluated at d1 and d0.
ATET	If FALSE, the average treatment effect (ATE) and the corresponding direct and indirect effects are estimated. If TRUE, the average treatment effect on the treated (ATET) and the corresponding direct and indirect effects are estimated. Default is FALSE.
trim	Trimming rule for discarding observations with too large weights in the estimation of any mean potential outcome. That is, observations with a weight>trim are dropped from the sample. Default is a maximum weight of 0.1 (or 10 percent) per observation.
lognorm	If FALSE, a linear model with normally distributed errors is assumed for generalized propensity score estimation. If TRUE, a lognormal model is assumed. Default is FALSE.
bw	Bandwith for the second order Epanechnikov kernel functions of the treatment. If set to NULL, bandwidth computation is based on the rule of thumb for Epanechnikov kernels, determining the bandwidth as the standard deviation of the treatment times 2.34/(n^0.25), where n is the sample size. Default is NULL.
boot	Number of bootstrap replications for estimating standard errors. Default is 1999.
cluster	A cluster ID for block or cluster bootstrapping when units are clustered rather than iid. Must be numerical. Default is NULL (standard bootstrap without clustering).

Details

Estimation of causal mechanisms (natural direct and indirect effects) of a continuous treatment under a selection on observables assumption assuming that all confounders of the treatment and the mediator, the treatment and the outcome, or the mediator and the outcome are observed. Units are weighted by the inverse of their conditional treatment densities (known as generalized propensity scores) given the mediator and/or observed confounders, which are estimated by linear or loglinear regression. Standard errors are obtained by bootstrapping the effects.

Value

A medweightcont object contains two components, `results` and `ntrimmed`:

`results`: a 3X5 matrix containing the effect estimates in the first row ("effects"), standard errors in the second row ("se"), and p-values in the third row ("p-value"). The first column provides the total effect, namely the average treatment effect (ATE) if `ATET=FALSE` or the average treatment effect on the treated (ATET), i.e. those with D=`d1`, if `ATET=TRUE`. The second and third columns provide the direct effects under treatment and control, respectively ("dir.treat", "dir.control"). The fourth and fifth columns provide the indirect effects under treatment and control, respectively ("indir.treat", "indir.control").

`ntrimmed`: number of discarded (trimmed) observations due to extreme propensity score values.

References

Hsu, Y.-C., Huber, M., Lee, Y.-Y., Lettry, L. (2020): "Direct and indirect effects of continuous treatments based on generalized propensity score weighting", Journal of Applied Econometrics, forthcoming.

Examples

```
# A little example with simulated data (10000 observations)
## Not run:
n=10000
x=runif(n=n,min=-1,max=1)
d=0.25*x+runif(n=n,min=-2,max=2)
d=d-min(d)
m=0.5*d+0.25*x+runif(n=n,min=-2,max=2)
y=0.5*d+m+0.25*x+runif(n=n,min=-2,max=2)
# The true direct and indirect effects are all equal to 0.5
output=medweightcont(y,d,m,x,d0=2,d1=3,ATET=FALSE,trim=0.1,
       lognorm=FALSE,bw=NULL,boot=19)
round(output$results,3)
output$ntrimmed
## End(Not run)
```

`swissexper` *Correspondence test in Swiss apprenticeship market*

Description

A dataset related to a field experiment (correspondence test) in the Swiss apprenticeship market 2018/2019. The experiment investigated the effects of applicant gender and parental occupation in applications to apprenticeships on callback rates (invitations to interviews, assessment centers, or trial apprenticeships)

Usage

```
swissexper
```

Format

A data frame with 2928 rows and 18 variables:

city agglomeration of apprenticeship: 1=Bern,2=Zurich,3=Basel,6=Lausanne

foundatdate date when job add was found

employees (estimated) number of employees: 1=1-20; 2=21-50; 3=51-100; 4=101-250; 5=251-500; 6=501-1000; 7=1001+

sector 1=public sector; 2=trade/wholesale; 3=manufacturing/goods; 4=services

uniqueID ID of application

sendatdate date when application was sent

job_father treatment: father's occupation: 1=professor; 2=unskilled worker; 3=intermediate commercial; 4=intermediate technical

job_mother treatment: mother's occupation: 1= primary school teacher; 2=homemaker

tier skill tier of apprenticeship: 1=lower; 2=intermediate; 3=upper

hasmoved applicant moved from different city: 1=yes; 0=no

contgender gender of contact person in company: 0=unknown; 1=female; 2=male

letterback 1: letters sent from company to applicant were returned; 0: no issues with returned letters

outcome_invite outcome: invitation to interview, assessment center, or trial apprenticeship: 1=yes; 0=no

female_appl treatment: 1=female applicant; 0=male applicant

antidiscrpolicy 1=explicit antidiscrimination policy on company's website; 0=no explicit antidiscrimination policy

outcome_interest outcome: either invitation, or asking further questions, or keeping application for further consideration

gender_neutrality 0=gender neutral job type; 1=female dominated job type; 2=male dominated type

company_activity scope of company's activity: 0=local; 1=national; 2=international

References

Fernandes, A., Huber, M., and Plaza, C. (2019): "The Effects of Gender and Parental Occupation in the Apprenticeship Market: An Experimental Evaluation", SES working paper 506, University of Fribourg. View(swissexper)

`treatDML` *Binary or multiple discrete treatment effect evaluation with double machine learning*

Description

Treatment effect estimation for assessing the average effects of discrete (multiple or binary) treatments. Combines estimation based on (doubly robust) efficient score functions with double machine learning to control for confounders in a data-driven way.

Usage

```
treatDML(
  y,
  d,
  x,
  s = NULL,
  dtreat = 1,
  dcontrol = 0,
  trim = 0.01,
  MLmethod = "lasso",
  k = 3
)
```

Arguments

`y`	Dependent variable, must not contain missings.
`d`	Treatment variable, must be discrete, must not contain missings.
`x`	Covariates, must not contain missings.
`s`	Indicator function for defining a subpopulation for whom the treatment effect is estimated as a function of the subpopulation's distribution of `x`. Default is `NULL` (estimation of the average treatment effect in the total population).
`dtreat`	Value of the treatment in the treatment group. Default is 1.
`dcontrol`	Value of the treatment in the control group. Default is 0.
`trim`	Trimming rule for discarding observations with treatment propensity scores that are smaller than `trim` or larger than `1-trim` (to avoid too small denominators in weighting by the inverse of the propensity scores). Default is 0.01.

`MLmethod` Machine learning method for estimating the nuisance parameters based on the `SuperLearner` package. Must be either `"lasso"` (default) for lasso estimation, `"randomforest"` for random forests, `"xgboost"` for xg boosting, `"svm"` for support vector machines, `"ensemble"` for using an ensemble algorithm based on all previously mentioned machine learners, or `"parametric"` for linear or logit regression.

`k` Number of folds in k-fold cross-fitting. Default is 3.

Details

Estimation of the causal effects of binary or mutliple discrete treatments under conditional independence, assuming that confounders jointly affecting the treatment and the outcome can be controlled for by observed covariates. Estimation is based on the (doubly robust) efficient score functions for potential outcomes in combination with double machine learning with cross-fitting, see Chernozhukov et al (2018). To this end, one part of the data is used for estimating the model parameters of the treatment and outcome equations based machine learning. The other part of the data is used for predicting the efficient score functions. The roles of the data parts are swapped (using k-fold cross-fitting) and the average treatment effect is estimated based on averaging the predicted efficient score functions in the total sample. Standard errors are based on asymptotic approximations using the estimated variance of the (estimated) efficient score functions.

Value

A `treatDML` object contains eight components, `effect`, `se`, `pval`, `ntrimmed`, `meantreat`, `meancontrol`, `pstreat`, and `pscontrol`:

`effect`: estimate of the average treatment effect.

`se`: standard error of the effect.

`pval`: p-value of the effect estimate.

`ntrimmed`: number of discarded (trimmed) observations due to extreme propensity scores.

`meantreat`: Estimate of the mean potential outcome under treatment.

`meancontrol`: Estimate of the mean potential outcome under control.

`pstreat`: P-score estimates for treatment in treatment group.

`pscontrol`: P-score estimates for treatment in control group.

References

Chernozhukov, V., Chetverikov, D., Demirer, M., Duflo, E., Hansen, C., Newey, W., Robins, J. (2018): "Double/debiased machine learning for treatment and structural parameters", The Econometrics Journal, 21, C1-C68.

van der Laan, M., Polley, E., Hubbard, A. (2007): "Super Learner", Statistical Applications in Genetics and Molecular Biology, 6.

Examples

```
# A little example with simulated data (2000 observations)
## Not run:
```

```
n=2000                          # sample size
p=100                           # number of covariates
s=2                             # number of covariates that are confounders
x=matrix(rnorm(n*p),ncol=p)     # covariate matrix
beta=c(rep(0.25,s), rep(0,p-s))  # coefficients determining degree of confounding
d=(x%*%beta+rnorm(n)>0)*1       # treatment equation
y=x%*%beta+0.5*d+rnorm(n)     # outcome equation
# The true ATE is equal to 0.5
output=treatDML(y,d,x)
cat("ATE: ",round(c(output$effect),3),", standard error: ",
    round(c(output$se),3), ", p-value: ",round(c(output$pval),3))
output$ntrimmed
## End(Not run)
```

`treatweight` *Treatment evaluation based on inverse probability weighting with optional sample selection correction.*

Description

Treatment evaluation based on inverse probability weighting with optional sample selection correction.

Usage

```
treatweight(
  y,
  d,
  x,
  s = NULL,
  z = NULL,
  selpop = FALSE,
  ATET = FALSE,
  trim = 0.05,
  logit = FALSE,
  boot = 1999,
  cluster = NULL
)
```

Arguments

y	Dependent variable.
d	Treatment, must be binary (either 1 or 0), must not contain missings.
x	Confounders of the treatment and outcome, must not contain missings.
s	Selection indicator. Must be one if `y` is observed (non-missing) and zero if `y` is not observed (missing). Default is `NULL`, implying that `y` does not contain any missings.

z	Optional instrumental variable(s) for selection s. If NULL, outcome selection based on observables (x,d) - known as "missing at random" - is assumed. If z is defined, outcome selection based on unobservables - known as "non-ignorable missingness" - is assumed. Default is NULL. If s is NULL, z is ignored.
selpop	Only to be used if both s and z are defined. If TRUE, the effect is estimated for the selected subpopulation with s=1 only. If FALSE, the effect is estimated for the total population. (note that this relies on somewhat stronger statistical assumptions). Default is FALSE. If s or z is NULL, selpop is ignored.
ATET	If FALSE, the average treatment effect (ATE) is estimated. If TRUE, the average treatment effect on the treated (ATET) is estimated. Default is FALSE.
trim	Trimming rule for discarding observations with extreme propensity scores. If ATET=FALSE, observations with Pr(D=1|X)<trim or Pr(D=1|X)>(1-trim) are dropped. If ATET=TRUE, observations with Pr(D=1|X)>(1-trim) are dropped. If s is defined and z is NULL, observations with extremely low selection propensity scores, Pr(S=1|D,X)<trim, are discarded, too. If s and z are defined, the treatment propensity scores to be trimmed change to Pr(D=1|X,Pr(S=1|D,X,Z)). If in addition selpop is FALSE, observation with Pr(S=1|D,X,Z)<trim are discarded, too. Default for trim is 0.05.
logit	If FALSE, probit regression is used for propensity score estimation. If TRUE, logit regression is used. Default is FALSE.
boot	Number of bootstrap replications for estimating standard errors. Default is 1999.
cluster	A cluster ID for block or cluster bootstrapping when units are clustered rather than iid. Must be numerical. Default is NULL (standard bootstrap without clustering).

Details

Estimation of treatment effects of a binary treatment under a selection on observables assumption assuming that all confounders of the treatment and the outcome are observed. Units are weighted by the inverse of their conditional treatment propensities given the observed confounders, which are estimated by probit or logit regression. Standard errors are obtained by bootstrapping the effect. If s is defined, the procedure allows correcting for sample selectiondue to missing outcomes based on the inverse of the conditional selection probability. The latter might either be related to observables, which implies a missing at random assumption, or in addition also to unobservables, if an instrument for sample selection is available. See Huber (2012, 2014) for further details.

Value

A treatweight object contains six components: effect, se, pval, y1, y0, and ntrimmed.

effect: average treatment effect (ATE) if ATET=FALSE or the average treatment effect on the treated (ATET) if ATET=TRUE.

se: bootstrap-based standard error of the effect.

pval: p-value of the effect.

y1: mean potential outcome under treatment.

y0: mean potential outcome under control.

ntrimmed: number of discarded (trimmed) observations due to extreme propensity score values.

References

Horvitz, D. G., and Thompson, D. J. (1952): "A generalization of sampling without replacement from a finite universe", Journal of the American Statistical Association, 47, 663–685.

Huber, M. (2012): "Identification of average treatment effects in social experiments under alternative forms of attrition", Journal of Educational and Behavioral Statistics, 37, 443-474.

Huber, M. (2014): "Treatment evaluation in the presence of sample selection", Econometric Reviews, 33, 869-905.

Examples

```
# A little example with simulated data (10000 observations)
## Not run:
n=10000
x=rnorm(n); d=(0.25*x+rnorm(n)>0)*1
y=0.5*d+0.25*x+rnorm(n)
# The true ATE is equal to 0.5
output=treatweight(y=y,d=d,x=x,trim=0.05,ATET=FALSE,logit=TRUE,boot=19)
cat("ATE: ",round(c(output$effect),3),", standard error: ",
    round(c(output$se),3), ", p-value: ",round(c(output$pval),3))
output$ntrimmed
## End(Not run)
# An example with non-random outcome selection and an instrument for selection
## Not run:
n=10000
sigma=matrix(c(1,0.6,0.6,1),2,2)
e=(2*rmvnorm(n,rep(0,2),sigma))
x=rnorm(n)
d=(0.5*x+rnorm(n)>0)*1
z=rnorm(n)
s=(0.25*x+0.25*d+0.5*z+e[,1]>0)*1
y=d+x+e[,2]; y[s==0]=0
# The true ATE is equal to 1
output=treatweight(y=y,d=d,x=x,s=s,z=z,selpop=FALSE,trim=0.05,ATET=FALSE,
       logit=TRUE,boot=19)
cat("ATE: ",round(c(output$effect),3),", standard error: ",
    round(c(output$se),3), ", p-value: ",round(c(output$pval),3))
output$ntrimmed
## End(Not run)
```

---

wexpect *Wage expectations of students in Switzerland*

Description

A dataset containing information on wage expectations of 804 students at the University of Fribourg and the University of Applied Sciences in Bern in the year 2017.

Usage

```
wexpect
```

Format

A data frame with 804 rows and 39 variables:

wexpect1 wage expectations after finishing studies: 0=less than 3500 CHF gross per month; 1=3500-4000 CHF; 2=4000-4500 CHF;...; 15=10500-11000 CHF; 16=more than 11000 CHF

wexpect2 wage expectations 3 years after studying: 0=less than 3500 CHF gross per month; 1=3500-4000 CHF; 2=4000-4500 CHF;...; 15=10500-11000 CHF; 16=more than 11000 CHF

wexpect1othersex expected wage of other sex after finishing studies in percent of own expected wage

wexpect2othersex expected wage of other sex 3 years after studying in percent of own expected wage

male 1=male; 0=female

business 1=BA in business

econ 1=BA in economics

communi 1=BA in communication

businform 1=BA in business informatics

plansfull 1=plans working fulltime after studies

planseduc 1=plans obtaining further education (e.g. MA) after studies

sectorcons 1=planned sector: construction

sectortradesales 1=planned sector: trade and sales

sectortransware 1=planned sector: transport and warehousing

sectorhosprest 1=planned sector: hospitality and restaurant

sectorinfocom 1=planned sector: information and communication

sectorfininsur 1=planned sector: finance and insurance

sectorconsult 1=planned sector: consulting

sectoreduscience 1=planned sector: education and science

sectorhealthsocial 1=planned sector: health and social services

typegenstratman 1=planned job type: general or strategic management

typemarketing 1=planned job type: marketing

typecontrol 1=planned job type: controlling

typefinance 1=planned job type: finance

typesales 1=planned job type: sales

typetechengin 1=planned job type: technical/engineering

typehumanres 1=planned job type: human resources

posmanager 1=planned position: manager

age age in years

swiss 1=Swiss nationality

hassiblings 1=has one or more siblings

motherhighedu 1=mother has higher education

fatherhighedu 1=father has higher education

motherworkedfull 1=mother worked fulltime at respondent's age 4-6

motherworkedpart 1=mother worked parttime at respondent's age 4-6

matwellbeing self-assessed material wellbeing compared to average Swiss: 1=much worse; 2=worse; 3=as average Swiss; 4=better; 5=much better

homeowner 1=home ownership

treatmentinformation 1=if information on median wages in Switzerland was provided (randomized treatment)

treatmentorder 1=if order of questions on professional plans and personal information in survey has been reversed (randomized treatment), meaning that personal questions are asked first and professional ones later

References

Fernandes, A., Huber, M., and Vaccaro, G. (2020): "Gender Differences in Wage Expectations", arXiv preprint arXiv:2003.11496.

Examples

```
data(wexpect)
attach(wexpect)
# effect of randomized wage information (treatment) on wage expectations 3 years after
# studying (outcome)
treatweight(y=wexpect2,d=treatmentinformation,x=cbind(male,business,econ,communi,
businform,age,swiss,motherhighedu,fatherhighedu),boot=199)
# direct effect of gender (treatment) and indirect effect through choice of field of
# studies (mediator) on wage expectations (outcome)
medweight(y=wexpect2,d=male,m=cbind(business,econ,communi,businform),
x=cbind(treatmentinformation,age,swiss,motherhighedu,fatherhighedu),boot=199)
```

Index