Package 'PSweight'

March 29, 2024

Type Package

Title Propensity Score Weighting for Causal Inference with Observational Studies and Randomized Trials

Version 1.2.0

Date 2024-03-27

Description Supports propensity score weighting analysis of observational studies and randomized trials. Enables the estimation and inference of average causal effects with binary and multiple treatments using overlap weights (ATO), inverse probability of treatment weights (ATE), average treatment effect among the treated weights (ATT), matching weights (ATM) and entropy weights (ATEN), with and without propensity score trimming. These weights are members of the family of balancing weights introduced in Li, Morgan and Zaslavsky (2018) <doi:10.1080/01621459.2016.1260466> and Li and Li (2019) <doi:10.1214/19-AOAS1282>.

Depends R (>= 3.5.0)

License GPL (>= 2)

URL https://github.com/thuizhou/PSweight

Encoding UTF-8

LazyData true

RoxygenNote 7.1.1

Suggests knitr, rmarkdown

Imports lme4, nnet, MASS, ggplot2, numDeriv, gbm, SuperLearner

VignetteBuilder knitr

NeedsCompilation no

Author Tianhui Zhou [aut],
Guangyu Tong [aut],
Fan Li [aut],
Laine Thomas [aut],
Fan Li [aut],
Yukang Zeng [cre]

Maintainer Yukang Zeng <yukang.zeng@yale.edu>

Repository CRAN

Date/Publication 2024-03-29 22:40:02 UTC

R topics documented:

`coef.PSweight` *Point estimates of PSweight*

Description

The `coef` method for class "PSweight"

Usage

```
## S3 method for class 'PSweight'
coef(object, ...)
```

Arguments

`object`	an object for which the extraction of model coefficients is meaningful.
`...`	other arguments.

Value

The output from `coef`

NCDS *Illustrative dataset for PSweight*

Description

This is a real observational study with binary and multiple treatment groups to illustrate the utility of PSweight functions.

Usage

```
data(NCDS)
```

Format

A data frame with 3642 rows and 16 columns.

Details

An dataset from the National Child Development Survey (NCDS) of the United Kingdom (UK). This dataset is obtained through the CC0 waiver from the work by Battistin and Sianesi (2011). For illustration, missing entries are imputed once with Multiple Imputation by Chained Equations (MICE).

This data frame contains the following columns:

- white: self-identified as white.
- wage: gross hourly wage in pound in log scale.
- Dany: whether received any education before.
- Dmult: three levels of educational attainment.
- maemp: employment status of mother.
- scht: school type.
- qmab: math score at age 7.
- qmab2: math score at age 11.
- qvab: reading score at age 7.
- qvab2: reading score at age 11.
- paed_u: father's years of education.
- maed_u: mother's years of education.
- age_pa: age of father.
- age_ma: age of mother.
- sub_u: number of siblings.
- wagebin: dichotomized wage obtained with a cutoff of average hourly wage.

References

https://cls.ucl.ac.uk/cls-studies/1958-national-child-development-study/

Battistin E, Sianesi B. (2011). Misclassified Treatment Status and Treatment Effects: an Application to Returns to Education in the United Kindom. Review of Economics and Statistics, 93(2), 495-509.

Battistin E, Sianesi B. (2012). Replication data for: Misclassified Treatment Status and Treatment Effects: an Application to Returns to Education in the United Kindom. URL https://doi.org.10.7910/DVN/EPCYUL.

Examples

```
data("NCDS")
```

`OUTmethod` *Fitting potential outcome regression with different methods*

Description

The function `OUTmethod` is an internal function to estimate the potential outcomes given a specified model through formula. It is built into function `PSweight`, and is used for constructing the augmented estimators.

Usage

```
OUTmethod(
  out.formula = out.formula,
  y = y,
  out.method = "glm",
  family = "gaussian",
  datain = datain,
  dataout = dataout,
  out.control = list()
)
```

Arguments

`out.formula`	an object of class `formula` (or one that can be coerced to that class): a symbolic description of the outcome model to be fitted.
`y`	a vector of the observed outcome in the training data (`datain`).
`out.method`	a character to specify the method for estimating the outcome regression model. `"glm"` is default, and `"gbm"` and `"SuperLearner"` are also allowed.
`family`	a description of the error distribution and link function to be used in the outcome model. Supported distributional families include `"gaussian"` (`link = identity`), `"binomial"` (`link = logit`) and `"poisson"` (`link = log`). Default is `"gaussian"`.
`datain`	The training data for the outcome model. In the context of `PSweight`, it refers to the data observed for each treatment group.

dataout	The prediction data for the outcome model. In the context of PSweight, it refers to the full data.
out.control	a list to specify additional options when out.method is set to "gbm" or "SuperLearner".

Details

A typical form for out.formula is y ~ terms where y is the outcome variable and terms is a series of terms which specifies linear predictors (on the link function scale). out.formula by default specifies generalized linear model given the gaussian error through the default arguments method = "glm" and family = "gaussian". It fits the logistic regression when family = "binomal",and poisson regression when family = "poisson". The argument out.method allows user to choose model other than glm to fit the outcome regression models for constructing the augmented estimator. We have included gbm and SuperLearner as alternative machine learning estimators. Additional argument in them can be supplied through the ... argument. Please refer to the user manual of the gbm and SuperLearner packages for all the allowed arguments.

Value

m.est a vector of predicted outcome on the dataout.

gamma.h estimated coefficient of the outcome model when method = "glm".

Examples

```
#' the outcome model
out.formula <- Y~cov1+cov2+cov3+cov4+cov5+cov6
y <- psdata$Y
#train on model with treatment group 1
datain <- psdata[psdata$trt==1, ]
outfit <- OUTmethod(out.formula = out.formula, y=y, datain = datain, dataout = psdata)
```

---

plot.SumStat *Plot the distribution of propensity scores and balance statistics*

Description

Summarize the SumStat object, generate histogram or density of estimated propensity scores and plot the balance statistics under weighting versus no weighting.

Usage

```
## S3 method for class 'SumStat'
plot(
  x,
  type = "balance",
  weighted.var = TRUE,
  threshold = 0.1,
```

```
  metric = "ASD",
  breaks = 50,
  ...
)
```

Arguments

x	a SumStat object obtained with SumStat function.
type	a character indicating the type of plot to produce, including histogram of estimated propensity scores ("hist"), density of estimated propensity scores ("density"), and plot of balance statistics ("balance").
weighted.var	logical. Indicating whether weighted variance should be used in calculating the balance statistics. Default is TRUE.
threshold	an optional numeric value indicating the balance threshold for the balance plot. Default is 0.1. Only valid when type = "balance".
metric	a character indicating the type of metric used in balance plot. Only "ASD" or "PSD" is allowed. If not specified, the default is "ASD". See summary.SumStat for additional details on balance metrics.
breaks	a single number giving the number of cells for the histogram. Default is 50.
...	further arguments passed to or from other methods.

Details

For the balance plot, a vertical line at threshold is used to define balance on covariates. The default value is threshold = 0.1 following Austin and Stuart (2015). If more than 2 treatments are considered, only density of the estimated generalized propensity scores will be produced, regardless of whether type = "density" or type = "hist".

Value

Plot of the indicated type.

References

Austin, P.C. and Stuart, E.A. (2015). Moving towards best practice when using inverse probability of treatment weighting (IPTW) using the propensity score to estimate causal treatment effects in observational studies. Statistics in Medicine, 34(28), 3661-3679.

Examples

```
data("psdata")
ps.formula<-trt~cov1+cov2+cov3+cov4+cov5+cov6
msstat <- SumStat(ps.formula, trtgrp="2", data=subset(psdata,trt>1),
   weight=c("IPW","overlap","treated","entropy","matching"))

plot(msstat, type="hist")
plot(msstat, type="balance", weighted.var=TRUE, threshold=0.1, metric="ASD")
```

print.PStrim *Print the results of PStrim*

Description

The print method for class "PStrim"

Usage

```
## S3 method for class 'PStrim'
print(x, ...)
```

Arguments

x	an object used to select a method.
...	further arguments passed to or from other methods.

Value

The output from print

print.PSweight *Print the results of PSweight*

Description

The print method for class "PSweight"

Usage

```
## S3 method for class 'PSweight'
print(x, ...)
```

Arguments

x	an object used to select a method.
...	further arguments passed to or from other methods.

Value

The output from print

print.PSweightsum *Print the results of Summary.PSweight*

Description

The print method for class "PSweightsum"

Usage

```
## S3 method for class 'PSweightsum'
print(x, ...)
```

Arguments

x	an object used to select a method.
...	further arguments passed to or from other methods.

Value

The output from print

print.SumStat *Print the results of SumStat*

Description

The print method for class "SumStat"

Usage

```
## S3 method for class 'SumStat'
print(x, ...)
```

Arguments

x	an object used to select a method.
...	further arguments passed to or from other methods.

Value

The output from print

print.SumSumStat *Print the results of Summary.SumStat*

Description

The print method for class "SumSumStat"

Usage

```
## S3 method for class 'SumSumStat'
print(x, ...)
```

Arguments

x	an object used to select a method.
...	further arguments passed to or from other methods.

Value

The output from print

psdata *Simulated dataset for PSweight*

Description

This is a simulated observational study with three treatment groups to illustated the ulity of PSweight.

Usage

```
data(psdata)
```

Format

A data frame with 1500 rows and 8 columns.

Details

The simulated dataset includes 1500 rows, with each row represents information recorded from each individual. There are 8 variables (columns). The treatment is the variable trt, which has three treatment arms. The outcome of interest is variable Y. cov1-cov6 are pre-treatment covariates among which cov1-cov5 are continous, and cov6 is binary.

Examples

```
data("psdata")
```

psdata_cl *Simulated dataset for PSweight*

Description

This is a simulated observational study with three treatment groups to illustated the ulity of PSweight.

Usage

```
data(psdata_cl)
```

Format

A data frame with 1500 rows and 9 columns.

Details

The simulated dataset includes 1500 rows, with each row represents information recorded from each individual. There are 9 variables (columns). The treatment is the variable trt, which has two treatment arms. The clt is the cluster level. The outcome of interest is variable Y. cov1-cov6 are pre-treatment covariates among which cov1-cov5 are continous, and cov6 is binary.

Examples

```
# data("psdata_cl")
```

PSmethod *Fitting propensity score models with different methods*

Description

The function PSmethod is an internal function to estimate the propensity scores given a specified model through formula. It is built into function Sumstat, PStrim and PSweight.

Usage

```
PSmethod(
  ps.formula = ps.formula,
  method = "glm",
  data = data,
  ncate = ncate,
  ps.control = list()
)
```

Arguments

ps.formula	an object of class formula (or one that can be coerced to that class): a symbolic description of the propensity score model to be fitted. Additional details of model specification are given under "Details". This argument is optional if ps.estimate is not NULL.
method	a character to specify the method for estimating propensity scores. "glm" is default, and "gbm" and "SuperLearner" are also allowed.
data	an optional data frame containing the variables in the propensity score model.
ncate	a numeric to specify the number of treatment groups present in the given data.
ps.control	a list to specify additional options when method is set to "gbm" or "SuperLearner".

Details

A typical form for ps.formula is treatment ~ terms where treatment is the treatment variable and terms is a series of terms which specifies a linear predictor. ps.formula by default specifies generalized linear models given the default argument method = "glm". It fits the logistic regression when ncate = 2,and multinomial logistic regression when ncate > 2. The argument method allows user to choose model other than glm to fit the propensity score models. We have included gbm and SuperLearner as two alternative machine learning methods. Additional arguments of the machine learning estimators can be supplied through the ... argument. Note that SuperLearner does not handle multiple groups and the current version of multinomial logistic regression is not supported by gbm. We suggest user to use them with extra caution. Please refer to the user manual of the gbm and SuperLearner packages for all the allowed arguments.

Value

e.h a data frame of estimated propensity scores.

ps.fitObjects the fitted propensity model details

beta.h estimated coefficient of the propensity model when method = "glm".

Examples

```
# the propensity model
ps.formula <- trt~cov1+cov2+cov3+cov4+cov5+cov6
psfit <- PSmethod(ps.formula = ps.formula,data = psdata,ncate=3)
```

PStrim *Trim the input data and propensity estimate*

Description

Trim the original data and propensity estimate according to symmetric propensity score trimming rules.

Usage

```
PStrim(
  data,
  ps.formula = NULL,
  zname = NULL,
  ps.estimate = NULL,
  delta = 0,
  optimal = FALSE,
  out.estimate = NULL,
  method = "glm",
  ps.control = list()
)
```

Arguments

data	an optional data frame containing the variables required by ps.formula.
ps.formula	an object of class formula (or one that can be coerced to that class): a symbolic description of the propensity score model to be fitted. Additional details of model specification are given under "Details". This argument is optional if ps.estimate is not NULL.
zname	an optional character specifying the name of the treatment variable in data. Unless ps.formula is specified, zname is required.
ps.estimate	an optional matrix or data frame containing estimated (generalized) propensity scores for each observation. Typically, this is an N by J matrix, where N is the number of observations and J is the total number of treatment levels. Preferably, the column name of this matrix should match the name of treatment level, if column name is missing or there is a mismatch, the column names would be assigned according to alphabatic order of the treatment levels. A vector of propensity score estimates is also allowed in ps.estimate, in which case a binary treatment is implied and the input is regarded as the propensity to receive the last category of treatment by alphabatic order, unless otherwise stated by trtgrp.
delta	trimming threshold for estimated (generalized) propensity scores. Should be no larger than 1 / number of treatment groups. Default is 0, corresponding to no trimming.
optimal	an logical argument indicating if optimal trimming should be used. Default is FALSE.
out.estimate	an optional matrix or data frame containing estimated potential outcomes for each observation. Typically, this is an N by J matrix, where N is the number of observations and J is the total number of treatment levels. Preferably, the column name of this matrix should match the name of treatment level, if column name is missing or there is a mismatch, the column names would be assigned according to alphabatic order of the treatment levels, with a similar mechanism as in ps.estimate.
method	a character to specify the method for estimating propensity scores. "glm" is default, and "gbm" and "SuperLearner" are also allowed.
ps.control	a list to specify additional options when method is set to "gbm" or "SuperLearner".

Details

A typical form for ps.formula is treatment ~ terms where treatment is the treatment variable (identical to the variable name used to specify zname) and terms is a series of terms which specifies a linear predictor for treatment. ps.formula specifies a model for estimating the propensity scores, when ps.estimate is NULL. "glm" is the default method for propensity score estimation. Logistic regression will be used for binary outcomes, and multinomial logistic regression will be used for outcomes with more than two categories. The alternative method option of "gbm" serves as an API to call the gbm() function from the gbm package. Additional argument in the gbm() function can be supplied through the ps.control=list() argument in SumStat(). Please refer to the user manual of the "gbm" package for all the allowed arguments. Currently, models for binary or multinomial treatment will be automatically chosen based on the number of treatment categories. "SuperLearner" is also allowed in the method argument to call the SuperLearner() function in SuperLearner package. Currently, the SuperLearner method only support binary treatment with the default method set to "SL.glm". The estimation approach is default to "method.NNLS". Prediction algorithm and other tuning parameters can also be passed through ps.control=list(). Please refer to the user manual of the SuperLearner package for all the allowed specifications.

When comparing two treatments, ps.estimate can either be a vector or a two-column matrix of estimated propensity scores. If a vector is supplied, it is assumed to be the propensity scores to receive the treatment, and the treatment group corresponds to the last group in the alphebatic order, unless otherwise specified by trtgrp. When comparing multiple (J>=3) treatments, ps.estimate needs to be specified as an N by J matrix, where N indicates the number of observations, and J indicates the total number of treatments. This matrix specifies the estimated generalized propensity scores to receive each of the J treatments. The same mechanism applies to out.estimate, except that the input for out.estimate must be an N by J matrix, where each row corresponds to the estimated potential outcomes (corresponding to each treatment) for each observation.

With binary treatments, delta defines the symmetric propensity score trimming rule following Crump et al. (2009). With multiple treatments, delta defines the symmetric multinomial trimming rule introduced in Yoshida et al. (2019). With binary treatments and when optimal equals TRUE, the trimming function implements the optimal symmetric trimming rule in Crump et al. (2009). The optimal trimming threshold delta is then returned. With multiple treatments and optimal equals TRUE, the trimming function implements the optimal trimming rule in Yang et al. (2016). The optimal cutoff lambda, which defines the acceptable upper bound for the sum of inverse generalized propensity scores, is returned. See Yang et al. (2016) and Li and Li (2019) for details.

The argument zname is required when ps.estimate is not NULL.

Value

PStrim returns a list of the following values:

data a data frame of trimmed data.

trim_sum a table summrizing the number of cases by treatment groups before and after trimming.

ps.estimate a data frame of propensity estimate after trimming.

delta an optional output of trimming threshold for symmetric trimming.

lambda an optional output trimming threshold for optimal trimming with multiple treatment groups.

out.estimate a data frame of estimated potential outcomes after trimming.

References

Crump, R. K., Hotz, V. J., Imbens, G. W., Mitnik, O. A. (2009). Dealing with limited overlap in estimation of average treatment effects. Biometrika, 96(1), 187-199.

Yoshida, K., Solomon, D.H., Haneuse, S., Kim, S.C., Patorno, E., Tedeschi, S.K., Lyu, H., Franklin, J.M., Stürmer, T., Hernández-Díaz, S. and Glynn, R.J. (2019). Multinomial extension of propensity score trimming methods: A simulation study. American Journal of Epidemiology, 188(3), 609-616.

Yang, S., Imbens, G. W., Cui, Z., Faries, D. E., Kadziola, Z. (2016). Propensity score matching and subclassification in observational studies with multi-level treatments. Biometrics, 72(4), 1055-1065.

Li, F., Li, F. (2019). Propensity score weighting for causal inference with multiple treatments. The Annals of Applied Statistics, 13(4), 2389-2415.

Examples

```
data("psdata")

# the propensity model
ps.formula<-trt~cov1+cov2+cov3+cov4+cov5+cov6

# trim the original data by setting the threshold of propensity as 0.05
PStrim(data=psdata, ps.formula=ps.formula, delta=0.05)
PStrim(data=psdata, ps.formula=ps.formula, optimal=TRUE)
```

`PSweight` *Estimate average causal effects by propensity score weighting*

Description

The function `PSweight` is used to estimate the average potential outcomes corresponding to each treatment group among the target population. The function currently implements the following types of weights: the inverse probability of treatment weights (IPW: target population is the combined population), average treatment effect among the treated weights (treated: target population is the population receiving a specified treatment), overlap weights (overlap: target population is the overlap population at clinical equipoise), matching weights (matching: target population is population obtained under 1:1 matching), entropy weights (entropy: target population is the population weighted by the entropy function). Augmented propensity score weighting estimators are also allowed, with propensity scores and outcome model estimates either estimated within the function, or supplied by external routines.

Usage

```
PSweight(
  ps.formula = NULL,
  ps.estimate = NULL,
  trtgrp = NULL,
```

```
  zname = NULL,
  yname,
  data,
  weight = "overlap",
  delta = 0,
  augmentation = FALSE,
  bootstrap = FALSE,
  R = 50,
  out.formula = NULL,
  out.estimate = NULL,
  family = "gaussian",
  ps.method = "glm",
  ps.control = list(),
  out.method = "glm",
  out.control = list()
)
```

Arguments

ps.formula	an object of class formula (or one that can be coerced to that class): a symbolic description of the propensity score model to be fitted. Additional details of model specification are given under "Details". This argument is optional if ps.estimate is not NULL.
ps.estimate	an optional matrix or data frame containing estimated (generalized) propensity scores for each observation. Typically, this is an N by J matrix, where N is the number of observations and J is the total number of treatment levels. Preferably, the column name of this matrix should match the name of treatment level, if column name is missing or there is a mismatch, the column names would be assigned according to alphabatic order of the treatment levels. A vector of propensity score estimates is also allowed in ps.estimate, in which case a binary treatment is implied and the input is regarded as the propensity to receive the last category of treatment by alphabatic order, unless otherwise stated by trtgrp.
trtgrp	an optional character defining the "treated" population for estimating the average treatment effect among the treated (ATT). Only necessary if weight = "treated". This option can also be used to specify the treatment (in a two-treatment setting) when a vector argument is supplied for ps.estimate. Default value is the last group in the alphebatic order.
zname	an optional character specifying the name of the treatment variable in data.
yname	an optional character specifying the name of the outcome variable in data.
data	an optional data frame containing the variables in the propensity score model and outcome model (if augmented estimator is used). If not found in data, the variables are taken from environment(formula).
weight	a character or vector of characters including the types of weights to be used. "IPW" specifies the inverse probability of treatment weights for estimating the average treatment effect among the combined population. "treated" specifies the weights for estimating the average treatment effect among the treated.

	"overlap" specifies the (generalized) overlap weights for estimating the average treatment effect among the overlap population, or population at clinical equipoise. "matching" specifies the matching weights for estimating the average treatment effect among the matched population (ATM). "entropy" specifies the entropy weights for the average treatment effect of entropy weighted population (ATEN). Default is "overlap".
delta	trimming threshold for estimated (generalized) propensity scores. Should be no larger than 1 / number of treatment groups. Default is 0, corresponding to no trimming.
augmentation	logical. Indicate whether augmented weighting estimators should be used. Default is FALSE.
bootstrap	logical. Indaicate whether bootstrap is used to estimate the standard error of the point estimates. Default is FALSE.
R	an optional integer indicating number of bootstrap replicates. Default is R = 50.
out.formula	an object of class formula (or one that can be coerced to that class): a symbolic description of the outcome model to be fitted. Additional details of model specification are given under "Details". This argument is optional if out.estimate is not NULL.
out.estimate	an optional matrix or data frame containing estimated potential outcomes for each observation. Typically, this is an N by J matrix, where N is the number of observations and J is the total number of treatment levels. Preferably, the column name of this matrix should match the name of treatment level, if column name is missing or there is a mismatch, the column names would be assigned according to alphabatic order of the treatment levels, with a similar mechanism as in ps.estimate.
family	a description of the error distribution and link function to be used in the outcome model. Only required if out.formula is provided. Supported distributional families include "gaussian" (link = identity), "binomial" (link = logit) and "poisson" (link = log). See family in glm for more details. Default is "gaussian".
ps.method	a character to specify the method for estimating propensity scores. "glm" is default, and "gbm" and "SuperLearner" are also allowed.
ps.control	a list to specify additional options when method is set to "gbm" or "SuperLearner".
out.method	a character to specify the method for estimating the outcome regression model. "glm" is default, and "gbm" and "SuperLearner" are also allowed.
out.control	a list to specify additional options when out.method is set to "gbm" or "SuperLearner".

Details

A typical form for ps.formula is treatment ~ terms where treatment is the treatment variable (identical to the variable name used to specify zname) and terms is a series of terms which specifies a linear predictor for treatment. Similarly, a typical form for out.formula is outcome ~ terms where outcome is the outcome variable (identical to the variable name used to specify yname) and terms is a series of terms which specifies a linear predictor for outcome. Both ps.formula and out.formula by default specify generalized linear models when ps.estimate

and/or out.estimate is NULL. The argument ps.method and out.method allow user to choose model other than glm to fit the propensity score and outcome regression models for augmentation. Additional argument in the gbm() function can be supplied through the ps.control and out.control argument. Please refer to the user manual of the gbm package for all the allowed arguments. "SuperLearner" is also allowed in the ps.method and out.method arguments. Currently, the SuperLearner method only supports binary treatment with the default method set to "SL.glm". The estimation approach is default to "method.NNLS" for both propensity and outcome regression models. Prediction algorithm and other tuning parameters can also be passed throughps.control and out.control. Please refer to the user manual of the SuperLearner package for all the allowed specifications.

When comparing two treatments, ps.estimate can either be a vector or a two-column matrix of estimated propensity scores. If a vector is supplied, it is assumed to be the propensity scores to receive the treatment, and the treatment group corresponds to the last group in the alphebatic order, unless otherwise specified by trtgrp. When comparing multiple (J>=3) treatments, ps.estimate needs to be specified as an N by J matrix, where N indicates the number of observations, and J indicates the total number of treatments. This matrix specifies the estimated generalized propensity scores to receive each of the J treatments. In general, ps.estimate should have column names that indicate the level of the treatment variable, which should match the levels given in Z. If column names are empty or there is a mismatch, the column names will be created following the alphebatic order of values in Z, and the rightmost coulmn of ps.estimate is assumed to be the treatment group, when estimating ATT. trtgrp can also be used to specify the treatment group for estimating ATT. The same mechanism applies to out.estimate, except that the input for out.estimate must be an N by J matrix, where each row corresponds to the estimated potential outcomes (corresponding to each treatment) for each observation.

The argument zname and/or yname is required when ps.estimate and/or out.estimate is not NULL.

Current version of PSweight allows for five types of propensity score weights used to estimate ATE (IPW), ATT (treated) and ATO (overlap), ATM (matching) and ATEN (entropy). These weights are members of larger class of balancing weights defined in Li, Morgan, and Zaslavsky (2018). Specific definitions of these weights are provided in Li, Morgan, and Zaslavsky (2018), Li and Greene (2013), Zhou, Matsouaka and Thomas (2020). When there is a practical violation of the positivity assumption, delta defines the symmetric propensity score trimming rule following Crump et al. (2009). With multiple treatments, delta defines the multinomial trimming rule introduced in Yoshida et al. (2019). The overlap weights can also be considered as a data-driven continuous trimming strategy without specifying trimming rules, see Li, Thomas and Li (2019). Additional details on balancing weights and generalized overlap weights for multiple treatment groups are provided in Li and Li (2019).

If augmentation = TRUE, an augmented weighting estimator will be implemented. For binary treatments, the augmented weighting estimator is presented in Mao, Li and Greene (2018). For multiple treatments, the augmented weighting estimator is mentioned in Li and Li (2019), and additional details will appear in our ongoing work (Zhou et al. 2020+). When weight = "IPW", the augmented estimator is also referred to as a doubly-robust (DR) estimator.

When bootstrap = TRUE, the variance will be calculated by nonparametric bootstrap, with R bootstrap replications. The default of R is 50. Otherwise, the variance will be calculated using the sandwich variance formula obtained in the M-estimation framework.

Value

PSweight returns a `PSweight` object containing a list of the following values: estimated propensity scores, average potential outcomes corresponding to each treatment, variance-covariance matrix of the point estimates, the label for each treatment group, and estimates in each bootstrap replicate if `bootstrap = TRUE`. A summary of PSweight can be obtained with `summary.PSweight`.

`trtgrp` a character indicating the treatment group.

`propensity` a data frame of estimated propensity scores.

`muhat` average potential outcomes by treatment groups, with reference to specific target populations.

`covmu` variance-covariance matrix of `muhat`.

`muboot` an optional list of point estimates in each bootstrap replicate `bootstrap = TRUE`.

`group` a table of treatment group labels corresponding to the output point estimates `muhat`.

References

Crump, R. K., Hotz, V. J., Imbens, G. W., Mitnik, O. A. (2009). Dealing with limited overlap in estimation of average treatment effects. Biometrika, 96(1), 187-199.

Li, L., Greene, T. (2013). A weighting analogue to pair matching in propensity score analysis. The International Journal of Biostatistics, 9(2), 215-234.

Li, F., Morgan, K. L., Zaslavsky, A. M. (2018). Balancing covariates via propensity score weighting. Journal of the American Statistical Association, 113(521), 390-400.

Mao, H., Li, L., Greene, T. (2019). Propensity score weighting analysis and treatment effect discovery. Statistical Methods in Medical Research, 28(8), 2439-2454.

Li, F., Thomas, L. E., Li, F. (2019). Addressing extreme propensity scores via the overlap weights. American Journal of Epidemiology, 188(1), 250-257.

Yoshida, K., Solomon, D.H., Haneuse, S., Kim, S.C., Patorno, E., Tedeschi, S.K., Lyu, H., Franklin, J.M., Stürmer, T., Hernández-Díaz, S. and Glynn, R.J. (2019). Multinomial extension of propensity score trimming methods: A simulation study. American Journal of Epidemiology, 188(3), 609-616.

Li, F., Li, F. (2019). Propensity score weighting for causal inference with multiple treatments. The Annals of Applied Statistics, 13(4), 2389-2415.

Zhou, Y., Matsouaka, R. A., Thomas, L. (2020). Propensity score weighting under limited overlap and model misspecification. Statistical Methods in Medical Research, 29(12), 3721-3756.

Examples

```
data("psdata")
# the propensity and outcome models
ps.formula<-trt~cov1+cov2+cov3+cov4+cov5+cov6
out.formula<-Y~cov1+cov2+cov3+cov4+cov5+cov6

# without augmentation
ato1<-PSweight(ps.formula = ps.formula,yname = 'Y',data = psdata,weight = 'overlap')
summary(ato1)

# augmented weighting estimator, takes longer time to calculate sandwich variance
```

```
# ato2<-PSweight(ps.formula = ps.formula,yname = 'Y',data = psdata,
#         augmentation = TRUE,out.formula = out.formula,family = 'gaussian',weight = 'overlap')
# summary(ato2)
```

PSweight_cl *Estimate average causal effects by propensity score weighting for a binary treatment with clustering.*

Description

The function PSweight_cl is used to estimate the average potential outcomes corresponding to each treatment group among the target population with two-level data. The function currently implements the following types of weights: the inverse probability of treatment weights (IPW: target population is the combined population), average treatment effect among the treated weights (treated: target population is the population receiving a specified treatment), overlap weights (overlap: target population is the overlap population at clinical equipoise), matching weights (matching: target population is population obtained under 1:1 matching), entropy weights (entropy: target population is the population weighted by the entropy function). Augmented propensity score weighting estimators are also allowed, with propensity scores and outcome model estimated within the function through mixed effect model.

Usage

```
PSweight_cl(
  ps.formula = NULL,
  trtgrp = NULL,
  yname,
  data,
  weight = "overlap",
  delta = 0,
  augmentation = FALSE,
  bootstrap = FALSE,
  bs_level = NULL,
  R = 50,
  out.formula = NULL,
  family = "gaussian",
  nAGQ = 1L
)
```

Arguments

ps.formula	an object of class formula (or one that can be coerced to that class): a symbolic description of the propensity score model to be fitted. Additional details of model specification are given under "Details".

trtgrp	an optional character defining the "treated" population for estimating the average treatment effect among the treated (ATT). Only necessary if weight = "treated". This option can also be used to specify the treatment (in a two-treatment setting) when a vector argument is supplied for ps.estimate. Default value is the last group in the alphebatic order.
yname	an optional character specifying the name of the outcome variable in data.
data	an optional data frame containing the variables in the propensity score model and outcome model (if augmented estimator is used). If not found in data, the variables are taken from environment(formula).
weight	a character or vector of characters including the types of weights to be used. "IPW" specifies the inverse probability of treatment weights for estimating the average treatment effect among the combined population. "treated" specifies the weights for estimating the average treatment effect among the treated. "overlap" specifies the (generalized) overlap weights for estimating the average treatment effect among the overlap population, or population at clinical equipoise. "matching" specifies the matching weights for estimating the average treatment effect among the matched population (ATM). "entropy" specifies the entropy weights for the average treatment effect of entropy weighted population (ATEN). Default is "overlap".
delta	trimming threshold for estimated (generalized) propensity scores. Should be no larger than 1 / number of treatment groups. Default is 0, corresponding to no trimming.
augmentation	logical. Indicate whether augmented weighting estimators should be used. Default is FALSE.
bootstrap	logical. Indaicate whether bootstrap is used to estimate the standard error of the point estimates. Default is FALSE.
bs_level	an optional character defining the cluster level (name of the variable) for each bootstrap resampling. Default is NULL.
R	an optional integer indicating number of bootstrap replicates. Default is R = 50.
out.formula	an object of class formula (or one that can be coerced to that class): a symbolic description of the outcome model to be fitted. Additional details of model specification are given under "Details". Different from the out.formula in PSweight, this formula should include the treatment label with cooresponding cluster forms.
family	a description of the error distribution and link function to be used in the outcome model. Only required if out.formula is provided. Supported distributional families include "gaussian" (link = identity), "binomial" (link = logit) and "poisson" (link = log). See family in glm for more details. Default is "gaussian".
nAGQ	integer scalar - the number of points per axis for evaluating the adaptive Gauss-Hermite approximation to the log-likelihood. Defaults to 1, corresponding to the Laplace approximation. Please refer to lme4 package for more details.

Details

A typical form for ps.formula is treatment ~ terms+1|clusters where treatment is the treatment variable and terms is a series of terms which specifies a linear predictor for treatment and

cluster level effects. Similarly, a typical form for `out.formula` is `outcome ~ treatment+terms+1|cluster` where `outcome` is the outcome variable (identical to the variable name used to specify `yname`); `terms` is a series of terms which specifies a linear predictor for `outcome`; `clusters` is the random effects term for clusters. Both `ps.formula` and `out.formula` by default specify generalized linear mixed effect models.

Current version of `PSweight_cl` allows for five types of propensity score weights used to estimate ATE (IPW), ATT (treated) and ATO (overlap), ATM (matching) and ATEN (entropy). These weights are members of larger class of balancing weights defined in Li, Morgan, and Zaslavsky (2018). Specific definitions of these weights are provided in Li, Morgan, and Zaslavsky (2018), Li and Greene (2013), Zhou, Matsouaka and Thomas (2020). When there is a practical violation of the positivity assumption, `delta` defines the symmetric propensity score trimming rule following Crump et al. (2009). The overlap weights can also be considered as a data-driven continuous trimming strategy without specifying trimming rules, see Li, Thomas and Li (2019). Additional details on balancing weights and generalized overlap weights for multiple treatment groups are provided in Li and Li (2019).

If `augmentation = TRUE`, an augmented weighting estimator will be implemented. For binary treatments, the augmented weighting estimator is presented in Mao, Li and Greene (2018). When `weight = "IPW"`, the augmented estimator is also referred to as a doubly-robust (DR) estimator.

When `bootstrap = TRUE`, the variance will be calculated by nonparametric bootstrap, with `R` bootstrap replications. `bs_level` needs to be specified as the variable name for the cluster in order to conduct cluster level resampling and maintaining the cluster level coorelation. The default value `NULL` treat each observation independently. The default of `R` is 50. Otherwise, the variance will be calculated using the sandwich variance formula obtained in the M-estimation framework.

Value

PSweight_cl returns a `PSweight` object containing a list of the following values: estimated propensity scores, average potential outcomes corresponding to each treatment, variance-covariance matrix of the point estimates, the label for each treatment group, and estimates in each bootstrap replicate if `bootstrap = TRUE`. A summary of PSweight_cl can be obtained with `summary.PSweight`.

`trtgrp` a character indicating the treatment group.

`propensity` a data frame of estimated propensity scores.

`muhat` average potential outcomes by treatment groups, with reference to specific target populations.

`covmu` variance-covariance matrix of `muhat`.

`muboot` an optional list of point estimates in each bootstrap replicate `bootstrap = TRUE`.

`group` a table of treatment group labels corresponding to the output point estimates `muhat`.

References

Li, F., Zaslavsky, A. M., Landrum, M. B. (2013). Propensity score weighting with multilevel data. Statistics in Medicine, 32(19), 3373-3387.

Fuentes, A., Lüdtke, O., Robitzsch, A. (2021). Causal inference with multilevel data: A comparison of different propensit score weighting appropaches. Multivariate Behavioral Research, 1-24.

Crump, R. K., Hotz, V. J., Imbens, G. W., Mitnik, O. A. (2009). Dealing with limited overlap in estimation of average treatment effects. Biometrika, 96(1), 187-199.

Li, L., Greene, T. (2013). A weighting analogue to pair matching in propensity score analysis. The International Journal of Biostatistics, 9(2), 215-234.

Li, F., Morgan, K. L., Zaslavsky, A. M. (2018). Balancing covariates via propensity score weighting. Journal of the American Statistical Association, 113(521), 390-400.

Mao, H., Li, L., Greene, T. (2019). Propensity score weighting analysis and treatment effect discovery. Statistical Methods in Medical Research, 28(8), 2439-2454.

Li, F., Thomas, L. E., Li, F. (2019). Addressing extreme propensity scores via the overlap weights. American Journal of Epidemiology, 188(1), 250-257.

Yoshida, K., Solomon, D.H., Haneuse, S., Kim, S.C., Patorno, E., Tedeschi, S.K., Lyu, H., Franklin, J.M., Stürmer, T., Hernández-Díaz, S. and Glynn, R.J. (2019). Multinomial extension of propensity score trimming methods: A simulation study. American Journal of Epidemiology, 188(3), 609-616.

Li, F., Li, F. (2019). Propensity score weighting for causal inference with multiple treatments. The Annals of Applied Statistics, 13(4), 2389-2415.

Zhou, Y., Matsouaka, R. A., Thomas, L. (2020). Propensity score weighting under limited overlap and model misspecification. Statistical Methods in Medical Research 29(12), 3721-3756.

Examples

```
#data("psdata_cl")
#ps.formula<-trt~cov1+cov2+cov3+cov4+cov5+cov6+(1|clt)
#ato_cl<-PSweight(ps.formula = ps.formula,yname = 'Y',data = psdata_cl)
#summary(ato_cl)
```

`summary.PSweight` *Summarize a PSweight object*

Description

`summary.PSweight` is used to summarize the results from `PSweight`. The output contains the average causal effects defined by specific contrasts, as well as their standard error estimates.

Usage

```
## S3 method for class 'PSweight'
summary(object, contrast = NULL, type = "DIF", CI = TRUE, ...)
```

Arguments

object	a PSweight object obtained from the PSweight function.
contrast	a vector or matrix specifying the causal contrast of interest. The average causal effects will be defined by such contrats. For multiple treatments, the contrast parameters are explained in Li and Li (2019) for estimating general causal effects. Default is all pairwise contrasts between any two treatment groups.
type	a character specifying the target estimand. The most commonly seen additive estimand is specified by type = "DIF", abbreviated for weighted difference-in-means. This is the usual pairwise average treatment effects as those defined in Li, Morgan, and Zaslavsky (2018) and Li and Li (2019). For binary (or count outcomes), we also allow two ratio estimands: causal relative risk (type = "RR") and causal odds ratio (type = "OR"). Estimates for these two ratio estimands will be reported on the log scale (log relative risk and log odds ratio) to improve the approximate for asymptotic normality. With binary outcomes, "DIF" is the same as the average causal risk difference. Default is "DIF" if left empty.
CI	a logical argument indicates whether confidence interval should be calculated. Default is CI = TRUE.
...	further arguments passed to or from other methods.

Details

For the contrast argument, one specifies the contrast of interest and thus defines the target estimand for comparing treatments. For example, if there are three treatment levels: A, B, and C, the contrast A-C (i.e., E[Y(A)] - E[Y(C)]) can be specified by c(1,0,-1). The contrasts of A-C and B-C can be jointly specified by rbind(c(1,0,-1), c(0,1,-1)).

For estimating the causal relative risk (type = "RR"), the contrast is specified at the log scale. For example, the contrast A-C (specified by c(1,0,-1)) implies the estimation of log{E[Y(A)]} - log{E[Y(C)]}. For estimating the causal odds ratio, the contrast is specified at the log odds scale. For example, the contrast A-C (specified by c(1,0,-1)) implies the estimation of log{E[Y(A)]/E[1-Y(A)]} - log{E[Y(C)]/E[1-Y(C)]}.

The variance of the contrasts will be estimated by the delta method (if sandwich variance is used, or bootstrap = FALSE), or nonparametric bootstrap (if bootstrap = TRUE). Details will be given in Zhou et al. (2020+).

The argument type takes one of three options: "DIF", "RR", or "RR", with "DIF" as the default option. Typically, "RR" is relavent for binary or count outcomes, and "OR" is relavent only for binary outcomes. "DIF" applies to all types of outcomes.

Value

A list of following values:

estimates	a matrix of point estimates, standard errors, test statistics, 95 for contrasts of interest.
bootestimates	a list of data frames containing estimated contrasts in each bootstrap replicate, if bootstrap is used to estimate standard errors.
contrast	a table listing the specified contrasts of interest.

`group` a table of treatment group labels corresponding to the output point estimates, provided in results obtained from `PSweight`.

`trtgrp` a character indicating the treatment group, or target population under ATT weights.

`type` a character specifying the target estimand.

`CI` a logical indaicator of whether confidence interval should be reported.

References

Li, F., Morgan, K. L., Zaslavsky, A. M. (2018). Balancing covariates via propensity score weighting. Journal of the American Statistical Association, 113(521), 390-400.

Li, F., Li, F. (2019). Propensity score weighting for causal inference with multiple treatments. The Annals of Applied Statistics, 13(4), 2389-2415.

Examples

```
## For examples, run: example(PSweight).
```

---

summary.SumStat *Summarize a SumStat object.*

Description

`summary.SumStat` is used to summarize results obtained from function `SumStat`. The output includes effective sample sizes and tables for balance statistics.

Usage

```
## S3 method for class 'SumStat'
summary(object, weighted.var = TRUE, metric = "ASD", ...)
```

Arguments

`object`	a `SumStat` object obtained with the `SumStat` function.
`weighted.var`	logical. Indicate whether the propensity score weighted variance should be used in calculating the balance metrics. Default is `TRUE`.
`metric`	a chatacter indicating the type of balance metrics. `"ASD"` refers to the pairwise absolute standardized difference and `"PSD"` refers to the population standardized difference. Default is `"ASD"`.
`...`	further arguments passed to or from other methods.

Details

For `metric`, the two options `"ASD"` and `"PSD"` are defined in Li and Li (2019) for the general family of balancing weights. Similar definitions are also given in McCaffrey et al. (2013) for inverse probability weighting. `weighted.var` specifies whether weighted or unweighted variance should be used in calculating ASD or PSD. An example of weighted variance with two treatment groups is given in Austin and Stuart (2015). For more than two treatment groups, the maximum of ASD (across all pairs of treatments) and maximum of PSD (across all treatments) are calcualted, as explained in Li and Li (2019).

Value

A list of tables containing effective sample sizes and balance statistics on covariates for specified propensity score weighting schemes.

`effective.sample.size` a table of effective sample sizes. This serves as a conservative measure to characterize the variance inflation or precision loss due to weighting, see Li and Li (2019).

`unweighted` A table summarizing mean, variance by treatment groups, and standardized mean difference.

`IPW` If `"IPW"` is specified, this is a data table summarizing mean, variance by treatment groups, and standardized mean difference under inverse probability of treatment weighting.

`treated` If `"treated"` is specified, this is a data table summarizing mean, variance by treatment groups, and standardized mean difference under the ATT weights.

`overlap` If `"overlap"` is specified, this is a data table summarizing mean, variance by treatment groups, and standardized mean difference under the (generalized) overlap weights.

`matching` If `"matching"` is specified, this is a data table summarizing mean, variance by treatment groups, and standardized mean difference under the (generalized) matching weights.

`entropy` If `"entropy"` is specified, this is a data table summarizing mean, variance by treatment groups, and standardized mean difference under the (generalized) entropy weights.

References

Crump, R. K., Hotz, V. J., Imbens, G. W., Mitnik, O. A. (2009). Dealing with limited overlap in estimation of average treatment effects. Biometrika, 96(1), 187-199.

Li, L., Greene, T. (2013). A weighting analogue to pair matching in propensity score analysis. The International Journal of Biostatistics, 9(2), 215-234.

Austin, P.C. and Stuart, E.A. (2015). Moving towards best practice when using inverse probability of treatment weighting (IPTW) using the propensity score to estimate causal treatment effects in observational studies. Statistics in Medicine, 34(28), 3661-3679.

Li, F., Li, F. (2019). Propensity score weighting for causal inference with multiple treatments. The Annals of Applied Statistics, 13(4), 2389-2415.

Zhou, Y., Matsouaka, R. A., Thomas, L. (2020). Propensity score weighting under limited overlap and model misspecification. Statistical Methods in Medical Research (Online)

Examples

```
## For examples, run: example(SumStat).
```

SumStat *Calculate summary statistics for propensity score weighting*

Description

SumStat is used to generate distributional plots of the estimated propensity scores and balance diagnostics after propensity score weighting.

Usage

```
SumStat(
  ps.formula = NULL,
  ps.estimate = NULL,
  trtgrp = NULL,
  Z = NULL,
  covM = NULL,
  zname = NULL,
  xname = NULL,
  data = NULL,
  weight = "overlap",
  delta = 0,
  method = "glm",
  ps.control = list()
)
```

Arguments

ps.formula	an object of class formula (or one that can be coerced to that class): a symbolic description of the propensity score model to be fitted. Additional details of model specification are given under "Details". This argument is optional if ps.estimate is not NULL.
ps.estimate	an optional matrix or data frame containing estimated (generalized) propensity scores for each observation. Typically, this is an N by J matrix, where N is the number of observations and J is the total number of treatment levels. Preferably, the column names of this matrix should match the names of treatment level, if column names are missing or there is a mismatch, the column names would be assigned according to the alphabatic order of treatment levels. A vector of propensity score estimates is also allowed in ps.estimate, in which case a binary treatment is implied and the input is regarded as the propensity to receive the last category of treatment by alphabatic order, unless otherwise stated by trtgrp.
trtgrp	an optional character defining the "treated" population for estimating the average treatment effect among the treated (ATT). Only necessary if weight = "treated". This option can also be used to specify the treatment (in a two-treatment setting) when a vector argument is supplied for ps.estimate. Default value is the last group in the alphebatic order.

Z	an optional vector specifying the values of treatment, only necessary when the covariate matrix covM is provided instead of data.
covM	an optional covariate matrix or data frame including covariates, their interactions and higher-order terms. When the covariate matrix covM is provided, the balance statistics are generated according to each column of this matrix.
zname	an optional character specifying the name of the treatment variable in data.
xname	an optional vector of characters including the names of covariates in data.
data	an optional data frame containing the variables in the propensity score model. If not found in data, the variables are taken from environment(formula).
weight	a character or vector of characters including the types of weights to be used. "IPW" specifies the inverse probability weights for estimating the average treatment effect among the combined population (ATE). "treated" specifies the weights for estimating the average treatment effect among the treated (ATT). "overlap" specifies the (generalized) overlap weights for estimating the average treatment effect among the overlap population (ATO), or population at clinical equipoise. "matching" specifies the matching weights for estimating the average treatment effect among the matched population (ATM). "entropy" specifies the entropy weights for the average treatment effect of entropy weighted population (ATEN). Default is "overlap".
delta	trimming threshold for estimated (generalized) propensity scores. Should be no larger than 1 / number of treatment groups. Default is 0, corresponding to no trimming.
method	a character to specify the method for estimating propensity scores. "glm" is default, and "gbm" and "SuperLearner" are also allowed.
ps.control	a list to specify additional options when method is set to "gbm" or "SuperLearner".

Details

A typical form for ps.formula is treatment ~ terms where treatment is the treatment variable (identical to the variable name used to specify zname) and terms is a series of terms which specifies a linear predictor for treatment. ps.formula specifies logistic or multinomial logistic models for estimating the propensity scores, when ps.estimate is NULL.

When comparing two treatments, ps.estimate can either be a vector or a two-column matrix of estimated propensity scores. If a vector is supplied, it is assumed to be the propensity scores to receive the treatment, and the treatment group corresponds to the last group in the alphebatic order, unless otherwise specified by trtgrp. When comparing multiple (J>=3) treatments, ps.estimate needs to be specified as an N by J matrix, where N indicates the number of observations, and J indicates the total number of treatments. This matrix specifies the estimated generalized propensity scores to receive each of the J treatments. In general, ps.estimate should have column names that indicate the level of the treatment variable, which should match the levels given in Z. If column names are empty or there is a mismatch, the column names will be created following the alphebatic order of treatmentlevels. The rightmost coulmn of ps.estimate is then assumed to be the treatment group when estimating ATT ("treated"). trtgrp can also be used to specify the treatment group for estimating ATT.

To generate balance statistics, one can directly specify Z and covM to indicate the treatment levels and covariate matrix. Alternatively, one can supply data, zname, and xname to indicate the same

information. When both are specified, the function will prioritize inputs from Z and covM. When ps.estimate is not NULL, argument zname.

Current version of PSweight allows for five types of propensity score weights used to estimate ATE ("IPW"), ATT ("treated"), and ATO("overlap"), ATM ("matching") and ATEN ("entropy"). These weights are members of a larger class of balancing weights defined in Li, Morgan, and Zaslavsky (2018). When there is a practical violation of the positivity assumption, delta defines the symmetric propensity score trimming rule following Crump et al. (2009). With multiple treatments, delta defines the multinomial trimming rule introduced in Yoshida et al. (2019). The overlap weights can also be considered as a data-driven continuous trimming strategy without specifying trimming rules, see Li, Thomas and Li (2019). Additional details on balancing weights and generalized overlap weights for multiple treatment groups are provided in Li and Li (2019). For details about matching weights and entropy weights, please refer to Li and Greene (2013) and Zhou, Matsouaka and Thomas (2020).

"glm" is the default method for propensity score estimation. Logistic regression will be used for binary outcomes, and multinomial logistic regression will be used for outcomes with more than two categories. The alternative method option of "gbm" serves as an API to call the gbm() function from the gbm package. Additional argument in the gbm() function can be supplied through the ps.control=list() argument in SumStat(). Please refer to the user manual of the gbm package for all the allowed arguments. Currently, models for binary or multinomial treatment will be automatically chosen based on the number of treatment categories. "SuperLearner" is also allowed in the method argument to pass the propensity score estimation to the SuperLearner() function in SuperLearner package. Currently, the SuperLearner method only supports binary treatment with the default method set to "SL.glm". The estimation approach is default to "method.NNLS" in the SumStat() function. Prediction algorithm and other tuning parameters can also be passed through ps.control=list() to SumStat(). Please refer to the user manual of the SuperLearner package for all the allowed specifications.

Value

SumStat returns a SumStat object including a list of the following value: treatment group, propensity scores, fitted propensity model, propensity score weights, effective sample sizes, and balance statistics. A summary of SumStat can be obtained with summary.SumStat.

trtgrp a character indicating the treatment group.

propensity a data frame of estimated propensity scores.

ps.fitObjects the fitted propensity model details

ps.weights a data frame of propensity score weights.

ess a table of effective sample sizes. This serves as a conservative measure to characterize the variance inflation or precision loss due to weighting, see Li and Li (2019).

unweighted.sumstat A list of tables including covariate means and variances by treatment group and standardized mean differences.

ATE.sumstat If "IPW" is included in weight, this is a list of summary statistics using inverse probability weighting.

ATT.sumstat If "treated" is included in weight, this is a list of summary statistics using the ATT weights.

`ATO.sumstat` If `"overlap"` is included in `weight`, this is a list of summary statistics using the overlap weights.

`ATM.sumstat` If `"matching"` is included in `weight`, this is a list of summary statistics using the matching weights.

`ATEN.sumstat` If `"entropy"` is included in `weight`, this is a list of summary statistics using the entropy weights.

`trim` If `delta > 0`, this is a table summarizing the number of observations before and after trimming.

References

Crump, R. K., Hotz, V. J., Imbens, G. W., Mitnik, O. A. (2009). Dealing with limited overlap in estimation of average treatment effects. Biometrika, 96(1), 187-199.

Greenwell B., Boehmke B.,Cunningham J, GBM Developers (2020) gbm: Generalized Boosted Regression Models. Cran: https://cran.r-project.org/web/packages/gbm/index.html

Li, L., Greene, T. (2013). A weighting analogue to pair matching in propensity score analysis. The International Journal of Biostatistics, 9(2), 215-234.

Li, F., Morgan, K. L., Zaslavsky, A. M. (2018). Balancing covariates via propensity score weighting. Journal of the American Statistical Association, 113(521), 390-400.

Li, F., Thomas, L. E., Li, F. (2019). Addressing extreme propensity scores via the overlap weights. American Journal of Epidemiology, 188(1), 250-257.

Polley E., LeDell E., Kennedy C., Lendle S., van der Laan M. (2019) SuperLearner: Super Learner Prediction. Cran: https://cran.r-project.org/web/packages/SuperLearner/index.html

Yoshida, K., Solomon, D.H., Haneuse, S., Kim, S.C., Patorno, E., Tedeschi, S.K., Lyu, H., Franklin, J.M., Stürmer, T., Hernández-Díaz, S. and Glynn, R.J. (2019). Multinomial extension of propensity score trimming methods: A simulation study. American Journal of Epidemiology, 188(3), 609-616.

Li, F., Li, F. (2019). Propensity score weighting for causal inference with multiple treatments. The Annals of Applied Statistics, 13(4), 2389-2415.

Zhou, Y., Matsouaka, R. A., Thomas, L. (2020). Propensity score weighting under limited overlap and model misspecification. Statistical Methods in Medical Research (Online)

Examples

```
data("psdata")
# the propensity model
ps.formula<-trt~cov1+cov2+cov3+cov4+cov5+cov6

# using SumStat to estimate propensity scores
msstat <- SumStat(ps.formula, trtgrp="2", data=psdata,
   weight=c("IPW","overlap","treated","entropy","matching"))
#summary(msstat)

# importing user-supplied propensity scores "e.h"
# fit <- nnet::multinom(formula=ps.formula, data=psdata, maxit=500, trace=FALSE)
# e.h <- fit$fitted.values
# varname <- c("cov1","cov2","cov3","cov4","cov5","cov6")
```

```
# msstat0 <- SumStat(zname="trt", xname=varname, data=psdata, ps.estimate=e.h,
#  trtgrp="2",  weight=c("IPW","overlap","treated","entropy","matching"))
# summary(msstat0)
```

SumStat_cl *Calculate summary statistics for propensity score weighting with clustering (for binary treatment only)*

Description

SumStat_cl is used to generate distributional plots of the estimated propensity scores and balance diagnostics after propensity score weighting with two-level data.

Usage

```
SumStat_cl(
  ps.formula = NULL,
  trtgrp = NULL,
  data = NULL,
  weight = "overlap",
  delta = 0,
  nAGQ = 1L
)
```

Arguments

ps.formula	an object of class formula (or one that can be coerced to that class): a symbolic description of the propensity score model to be fitted. Additional details of model specification are given under "Details".
trtgrp	an optional character defining the "treated" population for estimating the average treatment effect among the treated (ATT). Only necessary if weight = "treated". This option can also be used to specify the treatment (in a two-treatment setting). Default value is the last group in the alphebatic order.
data	an data frame containing the variables in the propensity score model. If not found in data, the variables are taken from environment(formula).
weight	a character or vector of characters including the types of weights to be used. "IPW" specifies the inverse probability weights for estimating the average treatment effect among the combined population (ATE). "treated" specifies the weights for estimating the average treatment effect among the treated (ATT). "overlap" specifies the (generalized) overlap weights for estimating the average treatment effect among the overlap population (ATO), or population at clinical equipoise. "matching" specifies the matching weights for estimating the average treatment effect among the matched population (ATM). "entropy" specifies the entropy weights for the average treatment effect of entropy weighted population (ATEN). Default is "overlap".

delta	trimming threshold for estimated (generalized) propensity scores. Should be no larger than 1 / number of treatment groups. Default is 0, corresponding to no trimming.
nAGQ	integer scalar - the number of points per axis for evaluating the adaptive Gauss-Hermite approximation to the log-likelihood. Defaults to 1, corresponding to the Laplace approximation. Please refer to lme4 package for more details.

Details

A typical form for ps.formula is treatment ~ terms+1|clusters where treatment is the treatment variable, terms is a series of terms which specifies a linear predictor for treatment, and clusters is the cluster indicator. The current version supports two-level models and the random-effects term is required to be the last piece in the formula. ps.formula specifies a mixed-effects logistic regression model for estimating propensity scores. The treatment group corresponds to the last group in the alphebatic order, unless otherwise specified by trtgrp.

Current version of PSweight allows for five types of propensity score weights used to estimate ATE ("IPW"), ATT ("treated"), and ATO("overlap"), ATM ("matching") and ATEN ("entropy"). These weights are members of a larger class of balancing weights defined in Li, Morgan, and Zaslavsky (2018). When there is a practical violation of the positivity assumption, delta defines the symmetric propensity score trimming rule following Crump et al. (2009). With multiple treatments, delta defines the multinomial trimming rule introduced in Yoshida et al. (2019). The overlap weights can also be considered as a data-driven continuous trimming strategy without specifying trimming rules, see Li, Thomas and Li (2019). Additional details on balancing weights and generalized overlap weights for multiple treatment groups are provided in Li and Li (2019). For details about matching weights and entropy weights, please refer to Li and Greene (2013) and Zhou, Matsouaka and Thomas (2020).

Value

SumStat_cl returns a SumStat object including a list of the following value: treatment group, propensity scores, fitted propensity model, propensity score weights, effective sample sizes, and balance statistics. A summary of SumStat can be obtained with summary.SumStat.

trtgrp a character indicating the treatment group.

propensity a data frame of estimated propensity scores.

ps.fitObjects the fitted propensity model details

ps.weights a data frame of propensity score weights.

ess a table of effective sample sizes. This serves as a conservative measure to characterize the variance inflation or precision loss due to weighting, see Li and Li (2019).

unweighted.sumstat A list of tables including covariate means and variances by treatment group and standardized mean differences.

ATE.sumstat If "IPW" is included in weight, this is a list of summary statistics using inverse probability weighting.

ATT.sumstat If "treated" is included in weight, this is a list of summary statistics using the ATT weights.

ATO.sumstat If "overlap" is included in weight, this is a list of summary statistics using the overlap weights.

`ATM.sumstat` If `"matching"` is included in `weight`, this is a list of summary statistics using the matching weights.

`ATEN.sumstat` If `"entropy"` is included in `weight`, this is a list of summary statistics using the entropy weights.

`trim` If `delta > 0`, this is a table summarizing the number of observations before and after trimming.

References

Crump, R. K., Hotz, V. J., Imbens, G. W., Mitnik, O. A. (2009). Dealing with limited overlap in estimation of average treatment effects. Biometrika, 96(1), 187-199.

Li, L., Greene, T. (2013). A weighting analogue to pair matching in propensity score analysis. The International Journal of Biostatistics, 9(2), 215-234.

Li, F., Morgan, K. L., Zaslavsky, A. M. (2018). Balancing covariates via propensity score weighting. Journal of the American Statistical Association, 113(521), 390-400.

Li, F., Thomas, L. E., Li, F. (2019). Addressing extreme propensity scores via the overlap weights. American Journal of Epidemiology, 188(1), 250-257.

Li, F., Li, F. (2019). Propensity score weighting for causal inference with multiple treatments. The Annals of Applied Statistics, 13(4), 2389-2415.

Zhou, Y., Matsouaka, R. A., Thomas, L. (2020). Propensity score weighting under limited overlap and model misspecification. Statistical Methods in Medical Research 29(12), 3721-3756.

Li, F., Zaslavsky, A. M., & Landrum, M. B. (2013). Propensity score weighting with multilevel data. Statistics in Medicine, 32(19), 3373-3387.

Examples

```
data("psdata_cl")
# the propensity model
# ps.formula<-trt~cov1+cov2+cov3+cov4+cov5+cov6+(1|clt)

# using SumStat to estimate propensity scores
# msstat <- SumStat_cl(ps.formula, trtgrp="1", data=psdata_cl,
#   weight=c("IPW","overlap","treated","entropy","matching"))
#summary(msstat)
```

`vcov.PSweight` *Variance of PSweight*

Description

The vcov method for class "PSweight"

Usage

```
## S3 method for class 'PSweight'
vcov(object, ...)
```

Arguments

object	an object used to select a method.
...	further arguments passed to or from other methods.

Value

The output from vcov

Index