Package 'AIPW'

July 21, 2025

Title Augmented Inverse Probability Weighting

Version 0.6.9.2

Maintainer Yongqi Zhong <`yq.zhong7@gmail.com`>

Description The 'AIPW' package implements the augmented inverse probability weighting, a doubly robust estimator, for average causal effect estimation with user-defined stacked machine learning algorithms. To cite the 'AIPW' package, please use: ``Yongqi Zhong, Edward H. Kennedy, Lisa M. Bodnar, Ashley I. Naimi (2021). AIPW: An R Package for Augmented Inverse Probability Weighted Estimation of Average Causal Effects. American Journal of Epidemiology. <doi:10.1093/aje/kwab207>". Visit: <`https://yqzhong7.github.io/AIPW/`> for more information.

License GPL-3

Encoding UTF-8

Language es

LazyData true

Suggests testthat (>= 2.1.0), knitr, rmarkdown, covr, tmle

RoxygenNote 7.2.2

Imports stats, utils, R6, SuperLearner, ggplot2, future.apply, progressr, Rsolnp

URL `https://github.com/yqzhong7/AIPW`

BugReports `https://github.com/yqzhong7/AIPW/issues`

VignetteBuilder knitr

Depends R (>= 2.10)

NeedsCompilation no

Author Yongqi Zhong [aut, cre] (ORCID:
<`https://orcid.org/0000-0002-4042-7450`>),
Ashley Naimi [aut] (ORCID: <`https://orcid.org/0000-0002-1510-8175`>),
Gabriel Conzuelo [ctb],
Edward Kennedy [ctb]

Repository CRAN

Date/Publication 2025-04-05 17:10:02 UTC

Contents

`AIPW` *Augmented Inverse Probability Weighting (AIPW)*

Description

An R6Class of AIPW for estimating the average causal effects with users' inputs of exposure, outcome, covariates and related libraries for estimating the efficient influence function.

Details

An AIPW object is constructed by `new()` with users' inputs of data and causal structures, then it `fit()` the data using the libraries in `Q.SL.library` and `g.SL.library` with `k_split` cross-fitting, and provides results via the `summary()` method. After using `fit()` and/or `summary()` methods, propensity scores and inverse probability weights by exposure status can be examined with `plot.p_score()` and `plot.ip_weights()`, respectively.

If outcome is missing, analysis assumes missing at random (MAR) by estimating propensity scores of I(A=a, observed=1) with all covariates `W`. (`W.Q` and `W.g` are disabled.) Missing exposure is not supported.

See examples for illustration.

Value

`AIPW` object

Constructor

`AIPW$new(Y = NULL, A = NULL, W = NULL, W.Q = NULL, W.g = NULL, Q.SL.library = NULL, g.SL.library = NULL, k_split = 10, verbose = TRUE, save.sl.fit = FALSE)`

Constructor Arguments:

Argument	Type	Details
`Y`	Integer	A vector of outcome (binary (0, 1) or continuous)
`A`	Integer	A vector of binary exposure (0 or 1)
`W`	Data	Covariates for **both** exposure and outcome models.
`W.Q`	Data	Covariates for the **outcome** model (Q).
`W.g`	Data	Covariates for the **exposure** model (g).
`Q.SL.library`	SL.library	Algorithms used for the **outcome** model (Q).
`g.SL.library`	SL.library	Algorithms used for the **exposure** model (g).
`k_split`	Integer	Number of folds for splitting (Default = 10).
`verbose`	Logical	Whether to print the result (Default = TRUE)
`save.sl.fit`	Logical	Whether to save Q.fit and g.fit (Default = FALSE)

Constructor Argument Details:

`W`, `W.Q` & `W.g` It can be a vector, matrix or data.frame. If and only if `W == NULL`, `W` would be replaced by `W.Q` and `W.g`.

`Q.SL.library` & `g.SL.library` Machine learning algorithms from **SuperLearner** libraries or `sl3` learner object (Lrnr_base)

`k_split` It ranges from 1 to number of observation-1. If k_split=1, no cross-fitting; if k_split>=2, cross-fitting is used (e.g., `k_split=10`, use 9/10 of the data to estimate and the remaining 1/10 leftover to predict). **NOTE: it's recommended to use cross-fitting.**

`save.sl.fit` This option allows users to save the fitted sl object (libs$Q.fit & libs$g.fit) for debug use. **Warning: Saving the SuperLearner fitted object may cause a substantive storage/memory use.**

Public Methods

Methods	Details	Link
`fit()`	Fit the data to the AIPW object	fit.AIPW
`stratified_fit()`	Fit the data to the AIPW object stratified by `A`	stratified_fit.AIPW
`summary()`	Summary of the average treatment effects from AIPW	summary.AIPW_base
`plot.p_score()`	Plot the propensity scores by exposure status	plot.p_score
`plot.ip_weights()`	Plot the inverse probability weights using truncated propensity scores	plot.ip_weights

Public Variables

Variable	Generated by	Return

`n`	Constructor	Number of observations
`stratified_fitted`	`stratified_fit()`	Fit the outcome model stratified by exposure status
`obs_est`	`fit()` & `summary()`	Components calculating average causal effects
`estimates`	`summary()`	A list of Risk difference, risk ratio, odds ratio
`result`	`summary()`	A matrix contains RD, ATT, ATC, RR and OR with their SE and 95%CI
`g.plot`	`plot.p_score()`	A density plot of propensity scores by exposure status
`ip_weights.plot`	`plot.ip_weights()`	A box plot of inverse probability weights
`libs`	`fit()`	**SuperLearner** or sl3 libraries and their fitted objects
`sl.fit`	Constructor	A wrapper function for fitting **SuperLearner** or sl3
`sl.predict`	Constructor	A wrapper function using `sl.fit` to predict

Public Variable Details:

`stratified_fit` An indicator for whether the outcome model is fitted stratified by exposure status in the `fit()` method. Only when using `stratified_fit()` to turn on `stratified_fit = TRUE`, `summary` outputs average treatment effects among the treated and the controls.

`obs_est` After using `fit()` and `summary()` methods, this list contains the propensity scores (`p_score`), counterfactual predictions (`mu`, `mu1` & `mu0`) and efficient influence functions (`aipw_eif1` & `aipw_eif0`) for later average treatment effect calculations.

`g.plot` This plot is generated by `ggplot2::geom_density`

`ip_weights.plot` This plot uses truncated propensity scores stratified by exposure status (`ggplot2::geom_boxplot`)

References

Zhong Y, Kennedy EH, Bodnar LM, Naimi AI (2021). AIPW: An R Package for Augmented Inverse Probability Weighted Estimation of Average Causal Effects. *American Journal of Epidemiology*.

Robins JM, Rotnitzky A (1995). Semiparametric efficiency in multivariate regression models with missing data. *Journal of the American Statistical Association*.

Chernozhukov V, Chetverikov V, Demirer M, et al (2018). Double/debiased machine learning for treatment and structural parameters. *The Econometrics Journal*.

Kennedy EH, Sjolander A, Small DS (2015). Semiparametric causal inference in matched cohort studies. *Biometrika*.

Examples

```
library(SuperLearner)
library(ggplot2)

#create an object
aipw_sl <- AIPW$new(Y=rbinom(100,1,0.5), A=rbinom(100,1,0.5),
                    W.Q=rbinom(100,1,0.5), W.g=rbinom(100,1,0.5),
                    Q.SL.library="SL.mean",g.SL.library="SL.mean",
                    k_split=1,verbose=FALSE)

#fit the object
aipw_sl$fit()
# or use `aipw_sl$stratified_fit()` to estimate ATE and ATT/ATC
```

```
#calculate the results
aipw_sl$summary(g.bound = 0.025)

#check the propensity scores by exposure status after truncation
aipw_sl$plot.p_score()
```

`AIPW_base` *Augmented Inverse Probability Weighting Base Class (AIPW_base)*

Description

A base class for AIPW that implements the common methods, such as `summary()` and `plot.p_score()`, inheritted by AIPW and AIPW_tmle class

Format

R6 object.

Value

`AIPW` base object

See Also

AIPW and AIPW_tmle

`AIPW_nuis` *Augmented Inverse Probability Weighting (AIPW) uses tmle or tmle3 as inputs*

Description

`AIPW_nuis` class for users to manually input nuisance functions (estimates from the exposure and the outcome models)

Details

Create an AIPW_nuis object that uses users' input nuisance functions from the exposure model $P(A|W)$, and the outcome models $P(Y|do(A=0),W)$ and $P(Y|do(A=1),W.Q)$:

$$\psi(a) = E[I(A=a)/P(A=a|W)] * [Y - P(Y=1|A,W)] + P(Y=1|do(A=a),W)$$

Note: If outcome is missing, replace (A=a) with (A=a, observed=1) when estimating the propensity scores.

Value

`AIPW_nuis` object

Constructor

`AIPW$new(Y = NULL, A = NULL, tmle_fit = NULL, verbose = TRUE)`

Constructor Arguments:

Argument	Type	Details
`Y`	Integer	A vector of outcome (binary (0, 1) or continuous)
`A`	Integer	A vector of binary exposure (0 or 1)
`mu0`	Numeric	User input of $P(Y = 1 \| do(A = 0), W_Q)$
`mu1`	Numeric	User input of $P(Y = 1 \| do(A = 1), W_Q)$
`raw_p_score`	Numeric	User input of $P(A = a \| W_g)$
`verbose`	Logical	Whether to print the result (Default = TRUE)
`stratified_fitted`	Logical	Whether mu0 & mu1 was estimated only using `A=0` & `A=1` (Default = FALSE)

Public Methods

Methods	Details	Link
`summary()`	Summary of the average treatment effects from AIPW	summary.AIPW_base
`plot.p_score()`	Plot the propensity scores by exposure status	plot.p_score
`plot.ip_weights()`	Plot the inverse probability weights using truncated propensity scores	plot.ip_weights

Public Variables

Variable	Generated by	Return
`n`	Constructor	Number of observations
`obs_est`	Constructor	Components calculating average causal effects
`estimates`	`summary()`	A list of Risk difference, risk ratio, odds ratio
`result`	`summary()`	A matrix contains RD, ATT, ATC, RR and OR with their SE and 95%CI
`g.plot`	`plot.p_score()`	A density plot of propensity scores by exposure status
`ip_weights.plot`	`plot.ip_weights()`	A box plot of inverse probability weights

Public Variable Details:

`stratified_fit` An indicator for whether the outcome model is fitted stratified by exposure status in the`fit()` method. Only when using `stratified_fit()` to turn on `stratified_fit = TRUE`, `summary` outputs average treatment effects among the treated and the controls.

`obs_est` This list includes propensity scores (`p_score`), counterfactual predictions (`mu`, `mu1` & `mu0`) and efficient influence functions (`aipw_eif1` & `aipw_eif0`)

`g.plot` This plot is generated by `ggplot2::geom_density`

`ip_weights.plot` This plot uses truncated propensity scores stratified by exposure status (`ggplot2::geom_boxplot`)

`AIPW_tmle` *Augmented Inverse Probability Weighting (AIPW) uses tmle or tmle3 as inputs*

Description

`AIPW_tmle` class uses a fitted `tmle` or `tmle3` object as input

Details

Create an AIPW_tmle object that uses the estimated efficient influence function from a fitted `tmle` or `tmle3` object

Value

`AIPW_tmle` object

Constructor

`AIPW$new(Y = NULL, A = NULL, tmle_fit = NULL, verbose = TRUE)`

Constructor Arguments:

Argument	**Type**	**Details**
`Y`	Integer	A vector of outcome (binary (0, 1) or continuous)
`A`	Integer	A vector of binary exposure (0 or 1)
`tmle_fit`	Object	A fitted `tmle` or `tmle3` object
`verbose`	Logical	Whether to print the result (Default = TRUE)

Public Methods

Methods	**Details**	**Link**
`summary()`	Summary of the average treatment effects from AIPW	summary.AIPW_base
`plot.p_score()`	Plot the propensity scores by exposure status	plot.p_score
`plot.ip_weights()`	Plot the inverse probability weights using truncated propensity scores	plot.ip_weights

Public Variables

Variable	**Generated by**	**Return**
`n`	Constructor	Number of observations
`obs_est`	Constructor	Components calculating average causal effects
`estimates`	`summary()`	A list of Risk difference, risk ratio, odds ratio
`result`	`summary()`	A matrix contains RD, ATT, ATC, RR and OR with their SE and 95%CI

`g.plot`	`plot.p_score()`	A density plot of propensity scores by exposure status
`ip_weights.plot`	`plot.ip_weights()`	A box plot of inverse probability weights

Public Variable Details:

`obs_est` This list extracts from the fitted `tmle` or `tmle3` object. It includes propensity scores (`p_score`), counterfactual predictions (`mu`, `mu1` & `mu0`) and efficient influence functions (`aipw_eif1` & `aipw_eif0`)

`g.plot` This plot is generated by `ggplot2::geom_density`

`ip_weights.plot` This plot uses truncated propensity scores stratified by exposure status (`ggplot2::geom_boxplot`)

Examples

```
## Not run:
vec <- function() sample(0:1,100,replace = TRUE)
df <- data.frame(replicate(4,vec()))
names(df) <- c("A","Y","W1","W2")

## From tmle
library(tmle)
library(SuperLearner)
tmle_fit <- tmle(Y=df$Y,A=df$A,W=subset(df,select=c("W1","W2")),
                 Q.SL.library="SL.glm",
                 g.SL.library="SL.glm",
                 family="binomial")
AIPW_tmle$new(A=df$A,Y=df$Y,tmle_fit = tmle_fit,verbose = TRUE)$summary()


## From tmle3
# tmle3 simple implementation
library(tmle3)
library(sl3)
node_list <- list(A = "A",Y = "Y",W = c("W1","W2"))
or_spec <- tmle_OR(baseline_level = "0",contrast_level = "1")
tmle_task <- or_spec$make_tmle_task(df,node_list)
lrnr_glm <- make_learner(Lrnr_glm)
sl <- Lrnr_sl$new(learners = list(lrnr_glm))
learner_list <- list(A = sl, Y = sl)
tmle3_fit <- tmle3(or_spec, data=df, node_list, learner_list)

# parse tmle3_fit into AIPW_tmle class
AIPW_tmle$new(A=df$A,Y=df$Y,tmle_fit = tmle3_fit,verbose = TRUE)$summary()

## End(Not run)
```

`aipw_wrapper` *AIPW wrapper function*

Description

A wrapper function for `AIPW$new()$fit()$summary()`

Usage

```
aipw_wrapper(
  Y,
  A,
  verbose = TRUE,
  W = NULL,
  W.Q = NULL,
  W.g = NULL,
  Q.SL.library,
  g.SL.library,
  k_split = 10,
  g.bound = 0.025,
  stratified_fit = FALSE
)
```

Arguments

`Y`	Outcome (binary integer: 0 or 1)
`A`	Exposure (binary integer: 0 or 1)
`verbose`	Whether to print the result (logical; Default = FALSE)
`W`	covariates for both exposure and outcome models (vector, matrix or data.frame). If null, this function will seek for inputs from `W.Q` and `W.g`.
`W.Q`	Only valid when `W` is null, otherwise it would be replaced by `W`. Covariates for outcome model (vector, matrix or data.frame).
`W.g`	Only valid when `W` is null, otherwise it would be replaced by `W`. Covariates for exposure model (vector, matrix or data.frame)
`Q.SL.library`	SuperLearner libraries or sl3 learner object (Lrnr_base) for outcome model
`g.SL.library`	SuperLearner libraries or sl3 learner object (Lrnr_base) for exposure model
`k_split`	Number of splitting (integer; range: from 1 to number of observation-1): if k_split=1, no cross-fitting; if k_split>=2, cross-fitting is used (e.g., `k_split=10`, use 9/10 of the data to estimate and the remaining 1/10 leftover to predict). NOTE: it's recommended to use cross-fitting.
`g.bound`	Value between [0,1] at which the propensity score should be truncated. Defaults to 0.025.
`stratified_fit`	An indicator for whether the outcome model is fitted stratified by exposure status in the`fit()` method. Only when using `stratified_fit()` to turn on `stratified_fit = TRUE`, `summary` outputs average treatment effects among the treated and the controls.

Value

A fitted `AIPW` object with summarised results

See Also

AIPW

Examples

```
library(SuperLearner)
aipw_sl <- aipw_wrapper(Y=rbinom(100,1,0.5), A=rbinom(100,1,0.5),
                    W.Q=rbinom(100,1,0.5), W.g=rbinom(100,1,0.5),
                    Q.SL.library="SL.mean",g.SL.library="SL.mean",
                    k_split=1,verbose=FALSE)
```

`eager_sim_obs` *Simulated Observational Study*

Description

Datasets were simulated using baseline covariates (sampling with replacement) from the Effects of Aspirin in Gestation and Reproduction (EAGeR) study. Data generating mechanisms were described in our manuscript (Zhong et al. (inpreparation), Am. J. Epidemiol.). True marginal causal effects on risk difference, log risk ratio and log odds ratio scales were attached to the dataset attributes (true_rd, true_logrr,true_logor).

Usage

```
data(eager_sim_obs)
```

Format

An object of class data.frame with 200 rows and 8 columns:

sim_Y binary, simulated outcome which is condition on all other covariates in the dataset

sim_A binary, simulated exposure which is conditon on all other covarites expect sim_Y.

eligibility binary, indicator of the eligibility stratum

loss_num count, number of prior pregnancy losses

age continuous, age in years

time_try_pregnant count, months of conception attempts prior to randomization

BMI continuous, body mass index

meanAP continuous, mean arterial blood pressure

References

Schisterman, E.F., Silver, R.M., Lesher, L.L., Faraggi, D., Wactawski-Wende, J., Townsend, J.M., Lynch, A.M., Perkins, N.J., Mumford, S.L. and Galai, N., 2014. Preconception low-dose aspirin and pregnancy outcomes: results from the EAGeR randomised trial. The Lancet, 384(9937), pp.29-36.

Zhong, Y., Naimi, A.I., Kennedy, E.H., (In preparation). AIPW: An R package for Augmented Inverse Probability Weighted Estimation of Average Causal Effects. American Journal of Epidemiology

See Also

eager_sim_rct

`eager_sim_rct` *Simulated Randomized Trial*

Description

Datasets were simulated using baseline covariates (sampling with replacement) from the Effects of Aspirin in Gestation and Reproduction (EAGeR) study.

Usage

```
data(eager_sim_rct)
```

Format

An object of class data.frame with 1228 rows and 8 columns:

sim_Y binary, simulated outcome which is condition on all other covariates in the dataset

sim_T binary, simulated treatment which is condition on eligibility only.

eligibility binary, indicator of the eligibility stratum

loss_num count, number of prior pregnancy losses

age continuous, age in years

time_try_pregnant count, months of conception attempts prior to randomization

BMI continuous, body mass index

meanAP continuous, mean arterial blood pressure

References

Schisterman, E.F., Silver, R.M., Lesher, L.L., Faraggi, D., Wactawski-Wende, J., Townsend, J.M., Lynch, A.M., Perkins, N.J., Mumford, S.L. and Galai, N., 2014. Preconception low-dose aspirin and pregnancy outcomes: results from the EAGeR randomised trial. The Lancet, 384(9937), pp.29-36.

Zhong, Y., Naimi, A.I., Kennedy, E.H., (In preparation). AIPW: An R package for Augmented Inverse Probability Weighted Estimation of Average Causal Effects. American Journal of Epidemiology

See Also

eager_sim_obs

`fit` *Fit the data to the AIPW object*

Description

Fitting the data into the AIPW object with/without cross-fitting to estimate the efficient influence functions

Value

A fitted AIPW object with `obs_est` and `libs` (public variables)

R6 Usage

`$fit()`

See Also

AIPW

`plot.ip_weights` *Plot the inverse probability weights using truncated propensity scores by exposure status*

Description

Plot and check the balance of propensity scores by exposure status

Value

`ip_weights.plot` (public variable): A box plot of inverse probability weights using truncated propensity scores by exposure status (`ggplot2::geom_boxplot`)

R6 Usage

`$plot.ip_weights()`

See Also

AIPW and AIPW_tmle

`plot.p_score` *Plot the propensity scores by exposure status*

Description

Plot and check the balance of propensity scores by exposure status

Value

`g.plot` (public variable): A density plot of propensity scores by exposure status (`ggplot2::geom_density`)

R6 Usage

`$plot.p_plot()`

See Also

AIPW and AIPW_tmle

`Repeated` *Repeated Crossfitting Procedure for AIPW*

Description

An R6Class that allows repeated crossfitting procedure for an AIPW object

Details

See examples for illustration.

Value

`AIPW` object

Constructor

`Repeated$new(aipw_obj = NULL)`

Constructor Arguments:

Argument	Type	Details
`aipw_obj`	AIPW object	an AIPW object

Public Methods

Methods	**Details**	**Link**
`repfit()`	Fit the data to the AIPW object `num_reps` times	repfit.Repeated
`summary_median()`	Summary (median) of estimates from the `repfit()`	summary_median.Repeated

Public Variables

Variable	**Generated by**	**Return**
`repeated_estimates`	`repfit()`	A data.frame of estiamtes form `num_reps` cross-fitting
`repeated_results`	`summary_median()`	A list of sumarised estimates
`result`	`summary_median()`	A data.frame of sumarised estimates

Public Variable Details:

`repeated_estimates` Estimates from `num_reps` cross-fitting.

`result` Summarised estimates from “repeated_estimates‘ using median methods.

References

Zhong Y, Kennedy EH, Bodnar LM, Naimi AI (2021). AIPW: An R Package for Augmented Inverse Probability Weighted Estimation of Average Causal Effects. *American Journal of Epidemiology*.

Robins JM, Rotnitzky A (1995). Semiparametric efficiency in multivariate regression models with missing data. *Journal of the American Statistical Association*.

Chernozhukov V, Chetverikov V, Demirer M, et al (2018). Double/debiased machine learning for treatment and structural parameters. *The Econometrics Journal*.

Kennedy EH, Sjolander A, Small DS (2015). Semiparametric causal inference in matched cohort studies. *Biometrika*.

Examples

```
library(SuperLearner)
library(ggplot2)

#create an object
aipw_sl <- AIPW$new(Y=rbinom(100,1,0.5), A=rbinom(100,1,0.5),
                    W.Q=rbinom(100,1,0.5), W.g=rbinom(100,1,0.5),
                    Q.SL.library="SL.mean",g.SL.library="SL.mean",
                    k_split=2,verbose=FALSE)

#create a repeated crossfitting object from the previous step
repeated_aipw_sl <- Repeated$new(aipw_sl)

#fit repetitively (stratified = TRUE will use stratified_fit() method in AIPW class)
```

```
repeated_aipw_sl$repfit(num_reps = 3, stratified = FALSE)

#summarise the results
repeated_aipw_sl$summary_median()
```

`repfit` *Fit the data to the AIPW object repeatedly*

Description

Fitting the data into the AIPW object with cross-fitting repeatedly to obtain multiple estimates from repetitions to avoid randomness due to splits in cross-fitting

Arguments

`num_reps`	Integer. Number of repetition of cross-fitting procedures (`fit()` or `stratified_fit()` see blow).
`stratified`	Boolean. `stratified` = `TRUE` will use `stratified_fit()` in the AIPW object to cross-fitting.

Value

A Repeated object with `repeated_estimates` (`estimates` from num_reps times repetition)

R6 Usage

`$repfit(num_reps = 20, stratified = FALSE)`

References

Chernozhukov V, Chetverikov V, Demirer M, et al (2018). Double/debiased machine learning for treatment and structural parameters. *The Econometrics Journal*.

See Also

Repeated and AIPW

`stratified_fit` *Fit the data to the AIPW object stratified by* `A` *for the outcome model*

Description

Fitting the data into the AIPW object with/without cross-fitting to estimate the efficient influence functions. Outcome model is fitted, stratified by exposure status `A`

Value

A fitted AIPW object with `obs_est` and `libs` (public variables)

R6 Usage

`$stratified_fit.AIPW()`

See Also

AIPW

`summary` *Summary of the average treatment effects from AIPW*

Description

Calculate average causal effects in RD, RR and OR in the fitted AIPW or AIPW_tmle object using the estimated efficient influence functions

Arguments

`g.bound`	Value between [0,1] at which the propensity score should be truncated. Propensity score will be truncated to $[g.bound, 1-g.bound]$ when one g.bound value is provided, or to $[min(g.bound), max(g.bound)]$ when two values are provided. **Defaults to 0.025**.

Value

`estimates` and `result` (public variables): Risks, Average treatment effect in RD, RR and OR.

R6 Usage

```
$summary(g.bound = 0.025)
$summary(g.bound = c(0.025,0.975))
```

See Also

AIPW and AIPW_tmle

`summary_median` *Summary of the* `repeated_estimates` *from* `repfit()` *in the Repeated object using median methods.*

Description

From `repeated_estimates`, calculate the median estimate (`median(Estimates)`), median SE (`median(SE)`), SE adjusting for variations across `num_reps` times, and 95% CI using SE adjusting for SE adjusted for variability.

Value

`repeated_results` and `result` (public variables).

R6 Usage

```
$summary_median.Repeated()
```

References

Chernozhukov V, Chetverikov V, Demirer M, et al (2018). Double/debiased machine learning for treatment and structural parameters. *The Econometrics Journal*.

See Also

Repeated and AIPW

Index