Package 'NMMIPW'

January 20, 2025

Type Package

Title Inverse Probability Weighting under Non-Monotone Missing

Version 0.1.0

Date 2022-01-01

Maintainer Andrew Ying <aying9339@gmail.com>

Description We fit inverse probability weighting estimator and the augmented inverse probability weighting for non-monotone missing at random data.

License GPL (>= 2)

Depends R (>= 4.0), lava, nloptr, numDeriv

RoxygenNote 7.1.2

Encoding UTF-8

NeedsCompilation no

Author Andrew Ying [aut, cre],
Baoluo Sun [ctb]

Repository CRAN

Date/Publication 2021-12-20 15:42:10 UTC

Contents

nmm_fit *Fitting IPW or AIPW Estimators under Nonmonotone Missing at Random Data*

Description

nmm_fit is the main function used to fit IPW or AIPW estimators under nonmonotone missing at random data

Usage

```
nmm_fit(
  data,
  O,
  AIPW = FALSE,
  formula = NULL,
  func = NULL,
  weights = NULL,
  ...
)
```

Arguments

data	a data.frame to fit
O	missing indicator
AIPW	indicator if fitting augmented IPW
formula	optional formula specified to fit
func	optional fitting function, currently support 'lm' and 'glm'
weights	optional weights used in the estimation
...	further arguments passed to func, e.g. family = 'quasibinomial' for glm

Value

NMMIPW returns an object of class "NMMIPW". An object of class "NMMIPW" is a list containing the following components:

coefficients	the fitted values, only reported when formula and func are given
coef_sd	the standard deviations of coefficients, only reported when formula and func are given
coef_IF	the influnece function of coefficients, only reported when formula and func are given
gamma_para	the first step fitted valus
AIPW	an indicator of whether AIPW is fitted
second_step	an indicator of whether the second step is fitted

second_fit	if second step fitted, we report the fit object
by_prod	a list of by products that might be useful for users, including first step IF, jacobian matrices

Examples

```
n = 100
X = rnorm(n, 0, 1)
Y = rnorm(n, 1 * X, 1)
O1 = rbinom(n, 1, 1/(1 + exp(- 1 - 0.5 * X)))
O2 = rbinom(n, 1, 1/(1 + exp(+ 0.5 + 1 * Y)))
O = cbind(O1, O2)
df <- data.frame(Y = Y, X = X)
fit <- nmm_fit(data = df, O = O, formula = Y ~ X, func = lm)
```

---

summary.NMMIPW *Summarizing IPW or AIPW Estimators under Nonmonotone Missing at Random Data*

Description

summary method for class "NMMIPW".

Usage

```
## S3 method for class 'NMMIPW'
summary(object, ...)

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

Arguments

object	an object of class "NMMIPW", usually, a result of a call to NMMIPW.
...	further arguments passed to or from other methods.
x	an object of class "summary.NMMIPW", usually, a result of a call to summary.NMMIPW.

Details

print.summary.NMMIPW tries to be smart about formatting coefficients, an estimated variance covariance matrix of the coefficients, Z-values and the corresponding P-values.

Value

The function summary.NMMIPW computes and returns a list of summary statistics of the fitted model given in object.

Examples

```
n = 100
X = rnorm(n, 0, 1)
Y = rnorm(n, 1 * X, 1)
O1 = rbinom(n, 1, 1/(1 + exp(-1 - 0.5 * X)))
O2 = rbinom(n, 1, 1/(1 + exp(+0.5 + 1 * Y)))
O = cbind(O1, O2)
df <- data.frame(Y = Y, X = X)
fit <- nmm_fit(data = df, O = O, formula = Y ~ X, funct = lm)
summary(fit)
```

Index