Package: POPInf (via r-universe)

August 19, 2024

Type Package

Title Assumption-Lean and Data-Adaptive Post-Prediction Inference

Version 1.0.0

Maintainer Jiacheng Miao <jiacheng.miao@wisc.edu>

Description Implementation of assumption-lean and data-adaptive post-prediction inference (POPInf), for valid and efficient statistical inference based on data predicted by machine learning. See Miao, Miao, Wu, Zhao, and Lu (2023) <arXiv:2311.14220>.

URL https://arxiv.org/abs/2311.14220, https://github.com/qlu-lab/POPInf

Depends R (>= 3.5.0),

Imports randomForest, MASS

License GPL-3

Encoding UTF-8

RoxygenNote 7.2.3

Repository https://qlu-lab.r-universe.dev

RemoteUrl https://github.com/qlu-lab/popinf

RemoteRef HEAD

RemoteSha 6177791fefd82ecf7c55ec9d89aba3419547e99f

Contents

`A` *Calculation of the matrix A based on single dataset*

Description

`A` function for the calculation of the matrix A based on single dataset

Usage

```
A(X, Y, quant = NA, theta, method)
```

Arguments

`X`	Array or DataFrame containing covariates
`Y`	Array or DataFrame of outcomes
`quant`	quantile for quantile estimation
`theta`	parameter theta
`method`	indicates the method to be used for M-estimation. Options include "mean", "quantile", "ols", "logistic", and "poisson".

Value

matrix A based on single dataset

`est_ini` *Initial estimation*

Description

`est_ini` function for initial estimation

Usage

```
est_ini(X, Y, quant = NA, method)
```

Arguments

X	Array or DataFrame containing covariates
Y	Array or DataFrame of outcomes
quant	quantile for quantile estimation
method	indicates the method to be used for M-estimation. Options include "mean", "quantile", "ols", "logistic", and "poisson".

Value

initial estimatior

---

link_grad *gradient of the link function*

Description

`link_grad` function for gradient of the link function

Usage

```
link_grad(t, method)
```

Arguments

t	t
method	indicates the method to be used for M-estimation. Options include "mean", "quantile", "ols", "logistic", and "poisson".

Value

gradient of the link function

---

link_Hessian *Hessians of the link function*

Description

`link_Hessian` function for Hessians of the link function

Usage

```
link_Hessian(t, method)
```

Arguments

`t`	t
`method`	indicates the method to be used for M-estimation. Options include "mean", "quantile", "ols", "logistic", and "poisson".

Value

Hessians of the link function

`mean_psi` *Sample expectation of psi*

Description

`mean_psi` function for sample expectation of psi

Usage

```
mean_psi(X, Y, theta, quant = NA, method)
```

Arguments

`X`	Array or DataFrame containing covariates
`Y`	Array or DataFrame of outcomes
`theta`	parameter theta
`quant`	quantile for quantile estimation
`method`	indicates the method to be used for M-estimation. Options include "mean", "quantile", "ols", "logistic", and "poisson".

Value

sample expectation of psi

mean_psi_pop *Sample expectation of POP-Inf psi*

Description

mean_psi_pop function for sample expectation of POP-Inf psi

Usage

```
mean_psi_pop(
  X_lab,
  X_unlab,
  Y_lab,
  Yhat_lab,
  Yhat_unlab,
  w,
  theta,
  quant = NA,
  method
)
```

Arguments

X_lab	Array or DataFrame containing observed covariates in labeled data.
X_unlab	Array or DataFrame containing observed or predicted covariates in unlabeled data.
Y_lab	Array or DataFrame of observed outcomes in labeled data.
Yhat_lab	Array or DataFrame of predicted outcomes in labeled data.
Yhat_unlab	Array or DataFrame of predicted outcomes in unlabeled data.
w	weights vector POP-Inf linear regression (d-dimensional, where d equals the number of covariates).
theta	parameter theta
quant	quantile for quantile estimation
method	indicates the method to be used for M-estimation. Options include "mean", "quantile", "ols", "logistic", and "poisson".

Value

sample expectation of POP-Inf psi

optim_est *Gradient descent for obtaining estimator*

Description

`optim_est` function for gradient descent for obtaining estimator

Usage

```
optim_est(
  X_lab,
  X_unlab,
  Y_lab,
  Yhat_lab,
  Yhat_unlab,
  w,
  theta,
  quant = NA,
  method,
  step_size = 0.1,
  max_iterations = 500,
  convergence_threshold = 1e-06
)
```

Arguments

`X_lab`	Array or DataFrame containing observed covariates in labeled data.
`X_unlab`	Array or DataFrame containing observed or predicted covariates in unlabeled data.
`Y_lab`	Array or DataFrame of observed outcomes in labeled data.
`Yhat_lab`	Array or DataFrame of predicted outcomes in labeled data.
`Yhat_unlab`	Array or DataFrame of predicted outcomes in unlabeled data.
`w`	weights vector POP-Inf linear regression (d-dimensional, where d equals the number of covariates).
`theta`	parameter theta
`quant`	quantile for quantile estimation
`method`	indicates the method to be used for M-estimation. Options include "mean", "quantile", "ols", "logistic", and "poisson".
`step_size`	step size for gradient descent
`max_iterations`	maximum of iterations for gradient descent
`convergence_threshold`	convergence threshold for gradient descent

Value

estimator

optim_weights — *Gradient descent for obtaining the weight vector*

Description

`optim_weights` function for gradient descent for obtaining estimator

Usage

```
optim_weights(
  j,
  X_lab,
  X_unlab,
  Y_lab,
  Yhat_lab,
  Yhat_unlab,
  w,
  theta,
  quant = NA,
  method
)
```

Arguments

`j`	j-th coordinate of weights vector
`X_lab`	Array or DataFrame containing observed covariates in labeled data.
`X_unlab`	Array or DataFrame containing observed or predicted covariates in unlabeled data.
`Y_lab`	Array or DataFrame of observed outcomes in labeled data.
`Yhat_lab`	Array or DataFrame of predicted outcomes in labeled data.
`Yhat_unlab`	Array or DataFrame of predicted outcomes in unlabeled data.
`w`	weights vector POP-Inf linear regression (d-dimensional, where d equals the number of covariates).
`theta`	parameter theta
`quant`	quantile for quantile estimation
`method`	indicates the method to be used for M-estimation. Options include "mean", "quantile", "ols", "logistic", and "poisson".

Value

weights

pop_M *POP-Inf M-Estimation*

Description

pop_M function conducts post-prediction M-Estimation.

Usage

```
pop_M(
  X_lab = NA,
  X_unlab = NA,
  Y_lab,
  Yhat_lab,
  Yhat_unlab,
  alpha = 0.05,
  weights = NA,
  max_iterations = 100,
  convergence_threshold = 0.05,
  quant = NA,
  intercept = FALSE,
  focal_index = NA,
  method
)
```

Arguments

X_lab	Array or DataFrame containing observed covariates in labeled data.
X_unlab	Array or DataFrame containing observed or predicted covariates in unlabeled data.
Y_lab	Array or DataFrame of observed outcomes in labeled data.
Yhat_lab	Array or DataFrame of predicted outcomes in labeled data.
Yhat_unlab	Array or DataFrame of predicted outcomes in unlabeled data.
alpha	Specifies the confidence level as 1 - alpha for confidence intervals.
weights	weights vector POP-Inf linear regression (d-dimensional, where d equals the number of covariates).
max_iterations	Sets the maximum number of iterations for the optimization process to derive weights.
convergence_threshold	Sets the convergence threshold for the optimization process to derive weights.
quant	quantile for quantile estimation
intercept	Boolean indicating if the input covariates' data contains the intercept (TRUE if the input data contains)
focal_index	Identifies the focal index for variance reduction.
method	indicates the method to be used for M-estimation. Options include "mean", "quantile", "ols", "logistic", and "poisson".

Value

A summary table presenting point estimates, standard error, confidence intervals (1 - alpha), P-values, and weights.

Examples

```
data <- sim_data()
X_lab <- data$X_lab
X_unlab <- data$X_unlab
Y_lab <- data$Y_lab
Yhat_lab <- data$Yhat_lab
Yhat_unlab <- data$Yhat_unlab
pop_M(Y_lab = Y_lab, Yhat_lab = Yhat_lab, Yhat_unlab = Yhat_unlab,
      alpha = 0.05, method = "mean")
pop_M(Y_lab = Y_lab, Yhat_lab = Yhat_lab, Yhat_unlab = Yhat_unlab,
      alpha = 0.05, quant = 0.75, method = "quantile")
pop_M(X_lab = X_lab, X_unlab = X_unlab,
      Y_lab = Y_lab, Yhat_lab = Yhat_lab, Yhat_unlab = Yhat_unlab,
      alpha = 0.05, method = "ols")
```

---

psi *Esimating equation*

Description

`psi` function for esimating equation

Usage

```
psi(X, Y, theta, quant = NA, method)
```

Arguments

`X`	Array or DataFrame containing covariates
`Y`	Array or DataFrame of outcomes
`theta`	parameter theta
`quant`	quantile for quantile estimation
`method`	indicates the method to be used for M-estimation. Options include "mean", "quantile", "ols", "logistic", and "poisson".

Value

esimating equation

Sigma_cal *Variance-covariance matrix of the estimation equation*

Description

Sigma_cal function for variance-covariance matrix of the estimation equation

Usage

```
Sigma_cal(
  X_lab,
  X_unlab,
  Y_lab,
  Yhat_lab,
  Yhat_unlab,
  w,
  theta,
  quant = NA,
  A_lab_inv,
  A_unlab_inv,
  method
)
```

Arguments

X_lab	Array or DataFrame containing observed covariates in labeled data.
X_unlab	Array or DataFrame containing observed or predicted covariates in unlabeled data.
Y_lab	Array or DataFrame of observed outcomes in labeled data.
Yhat_lab	Array or DataFrame of predicted outcomes in labeled data.
Yhat_unlab	Array or DataFrame of predicted outcomes in unlabeled data.
w	weights vector POP-Inf linear regression (d-dimensional, where d equals the number of covariates).
theta	parameter theta
quant	quantile for quantile estimation
A_lab_inv	Inverse of matrix A using labeled data
A_unlab_inv	Inverse of matrix A using unlabeled data
method	indicates the method to be used for M-estimation. Options include "mean", "quantile", "ols", "logistic", and "poisson".

Value

variance-covariance matrix of the estimation equation `sim_data` *Simulate the data for testing the functions*

Description

`sim_data` function for the calculation of the matrix A

Usage

```
sim_data(r = 0.9, binary = FALSE)
```

Arguments

`r`	imputation correlation
`binary`	simulate binary outcome or not

Value

simulated data

Index