Package 'ProSGPV'

January 6, 2021

Title Penalized Regression with Second-Generation P-Values

Version 0.1.0

Maintainer Yi Zuo <yi.zuo@vanderbilt.edu>

Description Implementation of penalized regression with second-generation p-values for variable selection. The one-stage algorithm is extremely fast and the two-stage algorithm has lower parameter estimation bias when data are highly correlated. S3 methods print(), summary(), coef(), and predict() are available for both algorithms, and S3 method plot() is available for the two-stage algorithm. Technical details of the algorithms can be found at <arXiv:2012.07941>.

Depends R (>= 3.5.0), glmnet

Imports MASS

License GPL-3

Encoding UTF-8

URL https://github.com/zuoyi93/ProSGPV, https://arxiv.org/abs/2012.07941

BugReports https://github.com/zuoyi93/ProSGPV/issues

LazyData true

RoxygenNote 7.1.1

NeedsCompilation no

Author Yi Zuo [aut, cre] (<https://orcid.org/0000-0002-6643-8326>), Thomas Stewart [aut], Jeffrey Blume [aut]

Repository CRAN

Date/Publication 2021-01-06 09:50:05 UTC

R topics documented:

`coef.sgpv` *Extract coefficients from the model fit*

Description

S3 method `coef` for an S3 object of class `sgpv`

Usage

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

Arguments

`object`	An `sgpv` object
`...`	Other `coef` arguments

Value

Coefficients in the OLS model

Examples

```
# load the package
library(ProSGPV)

# prepare the data
x <- t.housing[, -ncol(t.housing)]
y <- t.housing$V9

# run one-stage algorithm
out.sgpv.1 <- pro.sgpv(x = x, y = y, stage = 1)

# get coefficients
coef(out.sgpv.1)
```

gen.data *Generate simulation data*

Description

This function can be used to generate autoregressive simulation data

Usage

```
gen.data(n = 100, p = 50, s = 10, beta.min = 1, beta.max = 5, rho = 0, nu = 2)
```

Arguments

n	The number of observations
p	The number of explanatory variables
s	The number of true signals
beta.min	The smallest effect size in absolute value
beta.max	The largest effect size in absolute value
rho	The autocorrelation level
nu	The signal to noise ratio

Value

A list of following components:

X The generated explanatory variable matrix

Y A vector of outcome

index The indices of true signals

Examples

```
# generate data
data <- gen.data()

# extract x
x <- data[[1]]

# extract y
y <- data[[2]]

# extract the indices of true signals
index <- data[[3]]
```

get.coef *Get coefficients at each* lambda

Description

Get the coefficients and confidence intervals from regression at each lambda as well as the null bound in SGPVs

Usage

```
get.coef(xs, ys, lambda, lasso)
```

Arguments

xs	Standardized design matrix
ys	Standardized outcome
lambda	lambda in the lasso
lasso	An glmnet object

Value

A vector that contains the point estimates, confidence intervals and the null bound

get.var *Get indices*

Description

Get the indices of the variables selected by the algorithm

Usage

```
get.var(candidate.index, xs, ys)
```

Arguments

candidate.index	Indices of the candidate set
xs	Standardized independent variables
ys	Standardized dependent variable

Value

Indices of variables selected

plot.sgpv *Plot variable selection results*

Description

S3 method plot for an object of class sgpv. This function plots the fully relaxed lasso solution path on the standardized scale and the final variable selection results

Usage

```
## S3 method for class 'sgpv'
plot(x, lpv = 3, lambda.max = NULL, ...)
```

Arguments

x	An sgpv object
lpv	Lines per variable. It can take the value of 1 meaning that only the bound that is closest to the null will be plotted, or the value of 3 meaning that point estimates as well as 95% confidence interval will be plotted. Default is 3.
lambda.max	The maximum lambda on the plot. Default is NULL.
...	Other plot arguments

Examples

```
# load the package
library(ProSGPV)

# prepare the data
x <- t.housing[, -ncol(t.housing)]
y <- t.housing$V9

# two-stage algorithm
out.sgpv.2 <- pro.sgpv(x = x, y = y, stage = 2)

# plot the fully relaxed lasso solution path and final solution
plot(out.sgpv.2)

# zoom in a little bit
plot(out.sgpv.2, lambda.max = 0.01)

# only plot one confidence bound
plot(out.sgpv.2, lpv = 1, lambda.max = 0.01)
```

predict.sgpv *Prediction using the fitted model*

Description

S3 method `predict` for an object of class `sgpv`

Usage

```
## S3 method for class 'sgpv'
predict(object, newx, ...)
```

Arguments

`object`	An `sgpv` objectect
`newx`	Prediction data set
`...`	Other `predict` arguments

Value

Predicted values

Examples

```
# load the package
library(ProSGPV)

# prepare the data
x <- t.housing[, -ncol(t.housing)]
y <- t.housing$V9

# run one-stage algorithm
out.sgpv.1 <- pro.sgpv(x = x, y = y, stage = 1)

predict(out.sgpv.1)
```

print.sgpv *Print variable selection results*

Description

S3 method `print` for an S3 object of class `sgpv`

Usage

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

Arguments

x	An sgpv object
...	Other print arguments

Value

Variable selection results

Examples

```
# load the package
library(ProSGPV)

# prepare the data
x <- t.housing[, -ncol(t.housing)]
y <- t.housing$V9

# run one-stage algorithm
out.sgpv.1 <- pro.sgpv(x = x, y = y, stage = 1)

out.sgpv.1
```

---

pro.sgpv *pro.sgpv function*

Description

This function outputs the variable selection results from either one-stage algorithm or two-stage algorithm.

Usage

```
pro.sgpv(x, y, stage = c(1, 2))
```

Arguments

x	Independent variables, can be a matrix or a data.frame
y	Dependent variable, can be a vector or a column from a data.frame
stage	Algorithm indicator. 1 denotes the one-stage algorithm and 2 denotes the two-stage algorithm. Default is 1.

Value

A list of following components:

var.index A vector of indices of selected variables

var.label A vector of labels of selected variables

lambda Cross-validated lambda in the two-stage algorithm. `NULL` for the one-stage algorithm

x Input data `x`

y Input data `y`

See Also

- `print.sgpv()` prints the variable selection results
- `coef.sgpv()` extracts coefficient estimates
- `summary.sgpv()` summarizes the OLS outputs
- `predict.sgpv()` predicts the outcome
- `plot.sgpv()` plots variable selection results

Examples

```
# load the package
library(ProSGPV)

# prepare the data
x <- t.housing[, -ncol(t.housing)]
y <- t.housing$V9

# run one-stage algorithm
out.sgpv.1 <- pro.sgpv(x = x, y = y, stage = 1)

# More examples at https://github.com/zuoyi93/ProSGPV
```

---

summary.sgpv *Summary of the OLS model on selected variables*

Description

S3 method `summary` for an S3 object of class `sgpv`

Usage

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

Arguments

object	An sgpv object
...	Other arguments

Value

Summary of an OLS model

Examples

```
# load the package
library(ProSGPV)

# prepare the data
x <- t.housing[, -ncol(t.housing)]
y <- t.housing$V9

# run one-stage algorithm
out.sgpv.1 <- pro.sgpv(x = x, y = y, stage = 1)

# get regression summary
summary(out.sgpv.1)
```

---

t.housing *Tehran housing data*

Description

A dataset containing Tehran housing data. The data set has 372 observations. There are 26 explanatory variables at baseline, including 7 project physical and financial features (V2-V8) and 19 economic variables and indices (V11-V29). The outcome (V9) is the sales price of a real estate single-family residential apartment.

Usage

```
t.housing
```

Format

V9 Actual sales price

V2 Total floor area of the building

V3 Lot area

V4 Total Preliminary estimated construction cost based on the prices at the beginning of the project

V5 Preliminary estimated construction cost based on the prices at the beginning of the project

V6 Equivalent preliminary estimated construction cost based on the prices at the beginning of the project in a selected base year

V7 Duration of construction

V8 Price of the unit at the beginning of the project per square meter

V11 The number of building permits issued

V12 Building services index for preselected base year

V13 Wholesale price index of building materials for the base year

V14 Total floor areas of building permits issued by the city/municipality

V15 Cumulative liquidity

V16 Private sector investment in new buildings

V17 Land price index for the base year

V18 The number of loans extended by banks in a time resolution

V19 The amount of loans extended by banks in a time resolution

V20 The interest rate for loan in a time resolution

V21 The average construction cost by private sector at the completion of construction

V22 The average cost of buildings by private sector at the beginning of construction

V23 Official exchange rate with respect to dollars

V24 Nonofficial (street market) exchange rate with respect to dollars

V25 Consumer price index (CPI) in the base year

V26 CPI of housing, water, fuel & power in the base year

V27 Stock market index

V28 Population of the city

V29 Gold price per ounce

Source

http://archive.ics.uci.edu/ml/datasets/Residential+Building+Data+Set

Index