Package 'expectEnv'

August 21, 2019

Type Package

Title Computing Envelope Expectile Regression (EER) Estimators

Version 2.0.0

Author Tuo Chen

Maintainer Tuo Chen <chentuo@ufl.edu>

Depends R (>= 2.15.0), neldermead, pls, SALES

Imports stats

Description This package estimates the parameters in the EER model by using the generalized method of moments (GMM). It also provides envelope subspace dimension selection results and bootstrap standard errors for the EER estimators.

License GPL-2

Encoding UTF-8

LazyData true

RoxygenNote 6.1.1

R topics documented:

boot_eenv *Bootstrap for eenv_gmm*

Description

Compute bootstrap standard error for the EER estimator.

Usage

```
boot_eenv(formula, u, tau = 0.5, data = NULL, B)
```

Arguments

formula	An object of class "formula". It could takes form "response ~ terms" where "response" is the response vector and "terms" is a series of predictors connected by "+". It could also takes form "response ~ predictors matrix".
u	The dimension of the envelope subspace.
tau	Expectile level. The default value is 0.5.
data	An optional data frame which containes the variables in the model.
B	The number of bootstrap samples. A positive integer.

Details

This function computes the bootstrap standard errors for the regression slope coefficients in the EER model by paired bootstrap.

Value

A list consisting of:

bootbeta	The booststrap estimators for beta. An B by p matrix.
bootse	The standard error for elements in beta computed by bootstrap.

Examples

```
## Not run:
set.seed(0401)
y=state.x77[,5]
x=state.x77[,c(1,2,3,7)]
xs= scale(x, center = FALSE, scale = apply(x, 2, sd, na.rm = TRUE))
boot_eenv(y~xs,u=1,tau=0.9,B=200)

## End(Not run)
```

cv_eenv *Dimension selection for eenv_gmm*

Description

For each possible envelope subspace dimension, compute the prediction error for the EER estimator using cross validation. Pick the most appropriate envelope subspace dimension based on 'one-standard error' rule.

Usage

```
cv_eenv(formula, dims = NULL, tau = 0.5, data = NULL, fold = 5,
  numcv = 1)
```

Arguments

formula	An object of class "formula". It could takes form "response ~ terms" where "response" is the response vector and "terms" is a series of predictors connected by "+". It could also takes form "response ~ predictors matrix".
dims	Candidate envelope subspace dimension. The default value is integers from 1 to p.
tau	Expectile level. The default value is 0.5.
data	An optional data frame which containes the variables in the model.
fold	The number of the cross validation folders. The default value is 5.
numcv	The number of corss validations. The default value is 1.

Value

The output is a vector of length 4.

u_select	The selected dimensions based on different criterions. Its first component is derived from cross validation without 'one-standard error' rule based on mean squared prediction error. Its second component is derived from cross validation with 'one-standard error' rule based on mean squared prediction error. Its third component is derived from cross validation without 'one-standard error' rule based on expectile prediction error. Its forth component is derived from cross validation with 'one-standard error' rule based on expectile prediction error. We use its forth component as the dimension selection result for eenv_gmm.

Examples

```
## Not run:
set.seed(2018)
y=state.x77[,5]
x=state.x77[,c(1,2,3,7)]
xs= scale(x, center = FALSE, scale = apply(x, 2, sd, na.rm = TRUE))
cv_eenv(y~xs,tau=0.9)

## End(Not run)
```

eenv_gmm *Fit the EER model*

Description

Fit the envelope model in linear expectile regression with dimension u.

Usage

```
eenv_gmm(formula, u, tau = 0.5, data = NULL, maxiter = 100,
  ftol = 0.01)
```

Arguments

formula	An object of class "formula". It could takes form "response ~ terms" where "response" is the response vector and "terms" is a series of predictors connected by "+". It could also takes form "response ~ predictors matrix".
u	The dimension of the envelope subspace.
tau	Expectile level. The default value is 0.5.
data	An optional data frame which containes the variables in the model.
maxiter	The uplimit of the number of iterations in the optimization algorithm. The default value is 100.
ftol	The convergence tolerance in the optimization algorithm. The default value is 0.01.

Details

The optimization algorithm is Nelder–Mead method.

Value

A list consisting of:

alpha	The EER estimator for the intercept.
beta	The EER estimator for the slope coefficients beta. A vector of length p.
eta_ast	The EER estimator for eta. A vector of length u.
A	The EER estimator for A. A (p-u) by u matrix.
G1	The EER estimator for Gamma. A p by u matirx.
G2	The EER estimator for Gamma0. A p by (p-u) matrix.
Omega	The EER estimator for Omega. An u by u matrix.
Omega0	The EER estimator for Omega0. A (p-u) by (p-u) matrix.
muX	The EER estimator for the mean of the predictors. A vector of length p.
SigmaX	The EER estimator for the corvariance matrix of the predictors. A p by p matrix.
paramNum	The number of parameters in the EER model.
DeltaM_inv	The EER estimator for the scale matrix.
GEidx	A vector of length p. The first u elements are the indices of the pivoting rows in the initial estimator of an orthonormal basis matrix of the envelope subspace.
fit_time	The fit time of the function eenv_gmm.

Examples

```
## Not run:
y=state.x77[,5]
x=state.x77[,c(1,2,3,7)]
xs= scale(x, center = FALSE, scale = apply(x, 2, sd, na.rm = TRUE))
eenv_gmm(y~xs,u=1,tau=0.9)

## End(Not run)
```

GE *Gaussian elimination*

Description

Gaussian elimination with partial pivoting.

Usage

```
GE(A)
```

Arguments

A	An n by p matrix. n must be greater than or equal to p.

Details

This function performs Gaussian elimination to the input matrix and returns the locations of pivoting elements.

Value

The output is a vector of length n. The first p elements are the indices of the pivoting elements, ordered accoridng to columns, and the rest n-p elements are the remaining indices from 1 to n.

---

grams *Gram-Schmidt orthogonalization*

Description

Return the Gram-Schmidt orthogonalization of the columns of a matrix.

Usage

```
grams(A)
```

Arguments

A	An m by n matrix, m must be greater than or equal to n. The columns of A are assumed to be linearly independent.

Value

An m by n matrix whose columns are an orthonormal basis for the column space of A.

inv.vech *Stack a vector*

Description

Stack a vector into a symmetric matrix. The elements in the vector are used as the elements in the lower triangle of the symmetric matrix.

Usage

```
inv.vech(y)
```

Arguments

y	A vector of length n, n must satisfy n=R(R+1)/2 for some positive integer R.

Value

An R by R symmetric matrix.

nulbasis *Basis for nullspace*

Description

Return a basis for the nullspace of a matrix.

Usage

```
nulbasis(A)
```

Arguments

A	An m by n matrix, m and n can be any integers.

Value

A basis for the nullspace of A in the columns.

Examples

```
## Not run:
A=matrix(c(1,2,0,3,0,0,1,4),2,4,T)
nulbasis(A)

## End(Not run)
```

theoretcial.exp *Expectiles for the exponential distribution*

Description

Return the corresponding expectile at the given level for the exponential distribution Exp(1).

Usage

```
theoretcial.exp(w)
```

Arguments

w	An expectile level between 0 and 1.

Value

The corresponding expectile.

theoretcial.mix *Expectiles for the mixed normal distribution*

Description

Return the corresponding expectile at the given level for the mixed normal distribution 0.9N(0,1)+0.1N(1,5).

Usage

```
theoretcial.mix(w)
```

Arguments

w	An expectile level between 0 and 1.

Value

The corresponding expectile.

theoretcial.norm *Expectiles for the standard normal distribution*

Description

Return the corresponding expectile at the given level for the standard normal distribution.

Usage

```
theoretcial.norm(w)
```

Arguments

w	An expectile level between 0 and 1.

Value

The corresponding expectile.

theoretcial.t *Expectiles for the student's t distribution*

Description

Return the corresponding expectile at the given level for the student's t distribution t(4).

Usage

```
theoretcial.t(w)
```

Arguments

w	An expectile level between 0 and 1.

Value

The corresponding expectile.

vec *Vectorize a matrix*

Description

Vectorize a matrix into a vector.

Usage

```
vec(x)
```

Arguments

x	A m by n matrix, m and n can be any positive integers.

Value

A vector of length mn.

vech *Vectorize a symmetric matrix*

Description

Vectorize the lower triangle of a symmetric matrix into a vector.

Usage

```
vech(x)
```

Arguments

x	An n by n symmetric matrix, n can be any positive integer.

Value

A vector of length n(n+1)/2.

Index