Package ‘ILSE’

October 12, 2022

Type Package
Title Linear Regression Based on 'ILSE' for Missing Data
Version 1.1.7
License GPL-3
Author Wei Liu [aut, cre],
         Huazhen Lin [aut],
         Wei Lan [aut]
Maintainer Wei Liu <weiliu@smail.swufe.edu.cn>
Description Linear regression when covariates include missing values by embedding the
correlation information between covariates. Especially for block missing data,
it works well. 'ILSE' conducts imputation and regression simultaneously and iteratively.
More details can be referred to
URL https://github.com/feiyoung/ILSE
BugReports https://github.com/feiyoung/ILSE/issues
Encoding UTF-8
LazyData true
Date 2022-01-31
Depends R (>= 3.0.1)
Imports stats, Rcpp, pbapply
Suggests knitr, rmarkdown
LinkingTo Rcpp, RcppArmadillo
VignetteBuilder knitr
NeedsCompilation yes
RoxygenNote 7.1.1
Repository CRAN
Date/Publication 2022-01-31 03:10:05 UTC
R topics documented:

- Coef .................................................. 2
- cor.mat .............................................. 3
- cov.mat .............................................. 4
- fimlreg .............................................. 5
- ilse ................................................... 6
- kern ................................................... 8
- nhanes .............................................. 9
- print .................................................. 10
- summary ........................................... 11

Index .......................... 13

---

| Coef | Extracts Regression Coefficients |

Description

extracts model coefficients from object of class "ilse".

Usage

Coef(object)

Arguments

object an object of class "ilse".

Value

Coefficients extracted from object.

See Also

coeff, coefficient

Examples

# example one
data(nhanes)
NAlm2 <- ilse(age~., data=nhanes)
Coef(NAlm2)
Generate Two Type of Correlation Matrix

Description
Generate two type of correlation matrix

Usage
```
cor.mat(p, rho, type='toeplitz')
```

Arguments
- `p`: a positive integer, the dimension of correlation matrix.
- `rho`: a value between 0 and 1, a baseline value of correlation coefficient.
- `type`: a character, specify the type of correlation matrix and only include 'toeplitz' and 'identity' in current version.

Details
The argument `rho` specify the size of correlation coefficient. As for argument `type`, if `type='toeplitz'`, `sigma_ij=rho^|i-j|`; if `type='identity'`, `sigma_ij=rho` when `i!=j` and `sigma_ij=1` when `i=j`.

Value
return a correlation matrix with a type of specified structure.

Note
nothing

Author(s)
Liu Wei

References
nothing.

See Also
cov2cor

Examples
```
cor.mat(5, 0.5)
cor.mat(5, 0.5, type='identity')
```
Description
Generate two type of covariance matrix

Usage
`cov.mat(sdvec, rho, type='toeplitz')`

Arguments
- `sdvec`: a positive vector, standard deviation of each random variable.
- `rho`: a value between 0 and 1, a baseline value of correlation coefficient.
- `type`: a character, specify the type of correlation matrix and only include 'toeplitz' and 'identity' in current version.

Details
The argument `rho` specify the size of correlation coefficient. As for argument type, if `type='toeplitz'`, `sigma_ij= rho^|i-j|`; if `type='identity'`, `sigma_ij=rho` when `i!=j` and `sigma_ij=1` when `i=j`.

Value
return a covariance matrix with a type of specified structure.

Note
nothing

Author(s)
Liu Wei

References
nothing.

See Also
cov2cor

Examples
`cov.mat(rep(5, 5), 0.5)`
`cov.mat(c(2, 4, 3), 0.5, type='identity')`
**fimlreg**  

*Full Information Maximum Likelihood Linear Regression*

**Description**

Estimate regression coefficients based on Full Information Maximum Likelihood Estimation, which can couple missing data, including response missing or covariates missing.

**Usage**

```r
fimlreg(...)  
## S3 method for class 'formula'
fimlreg(formula, data=NULL, ...)
## S3 method for class 'numeric'
fimlreg(Y, X, ...)
```

**Arguments**

- `formula` an object of class "formula" (or one that can be coerced to that class): a symbolic description of the model to be fitted. The details of model specification are given under 'Details'.
- `Y` a numeric vector, the response variable.
- `X` a numeric matrix that may include NAs, the covariate matrix.
- `data` an optional data frame, list or environment (or object coercible by as.data.frame to a data frame) containing the variables in the model. If not found in data, the variables are taken from environment(formula), typically the environment from which else is called.
- `...` Optional arguments.

**Details**

Note that arguments ... of stats::nlm are the parameters of algorithm, see the details in help file of "nlm". "fimlreg" can couple with any type of missing data.

**Value**

Return a list including following components:

- `beta` A named vector of coefficients
- `formula` The formula used
- `data` The raw data

**Author(s)**

Liu Wei
See Also

ilse

Examples

data(nhanes)
## example one: include missing value
fiml1 <- fimlreg(age~., data=nhanes)
print(fiml1)
# example two: No missing value
## example two: No missing value
n <- 100
group <- rnorm(n, sd=4)
weight <- 3.2*group + 1.5 + rnorm(n, sd=0.1)
fimllm <- fimlreg(weight~group, data=data.frame(weight=weight, group=group))
print(fimllm)

ilse

Linear Regression by Iterative Least Square Estimation

Description

Linear regression when covariates include missing values embedding the correlation information between covariates by Iterative Least Square Estimation.

Usage

ilse(...)
## S3 method for class 'formula'
ilse(formula, data=NULL, bw=NULL, k.type=NULL, method="Par.cond", ...)
## S3 method for class 'numeric'
ilse(Y, X,bw=NULL, k.type=NULL, method="Par.cond", max.iter=20,
    peps=1e-5, feps = 1e-7, arma=TRUE, verbose=FALSE, ...)

Arguments

... Arguments passed to other methods.
formula an object of class "formula" (or one that can be coerced to that class): a symbolic description of the model to be fitted. The details of model specification are given under 'Details'.
Y a numeric vector, the response variable.
X a numeric matrix that may include NAs, the covariate matrix.
data an optional data frame, list or environment (or object coercible by as.data.frame to a data frame) containing the variables in the model. If not found in data, the variables are taken from environment(formula), typically the environment from which ilse is called.
ilse

bw  a positive value, specify the bandwidth in estimating missing values, default as NULL. When bw=NULL, it is automatically selected by empirical method.

k.type  an optional character string, specify the type of kernel used in iterative estimating algorithm and support 'epk', 'biweight', 'triangle', 'gaussian', 'triweight', 'tricube', 'cosine', 'uniform' in current version, default as 'gaussian'.

method  an optional character string, specify the iterative algorithm, support 'Par.cond' and 'Full.cond' in current version.

max.iter  an optional positive integer, the maximum iterative times, default as '20'.

peps  an optional positive value, tolerance value of relative variation rate of estimated parametric vector, default as '1e-7'.

feps  an optional positive value, tolerance value of relative variation rate of objective function value, default as '1e-7'.

arma  an optional logical value, whether use armadillo and Rcpp to speed computation, default as TRUE.

verbose  an optional logical value, indicate whether output the iterative information, default as 'TRUE'.

Details

Models for ilse are specified symbolically. A typical model has the form response ~ terms where response is the (numeric) response vector and terms is a series of terms which specifies a linear predictor for response. A terms specification of the form first + second indicates all the terms in first together with all the terms in second with duplicates removed. A specification of the form first:second indicates the set of terms obtained by taking the interactions of all terms in first with all terms in second. The specification first*second indicates the cross of first and second. This is the same as first + second + first:second.

Value

ilse returns an object of class "ilse".

The functions summary and anova are used to obtain and print a summary and analysis of variance table of the results. The generic accessor functions coefficients, effects, fitted.values and residuals extract various useful features of the value returned by lm.

An object of class "ilse" is a list containing at least the following components:

beta  a named vector of coefficients
hX  a imputed design matrix
d.fn  a nonnegative value, value of relative variation rate of objective function value
d.par  a nonnegative value, relative variation rate of estimated parametric vector when algorithm stopped.
iterations  a positive integer, iterative times in total.
residuals  the residuals, that is response minus fitted values.
fitted.values  the fitted mean values.
inargs  a list including all input arguments.
Note

nothing

Author(s)

Wei Liu

References


See Also

lm

Examples

## example one: include missing value
data(nhanes)
NAlm1 <- ilse(age~., data=nhanes,bw=1,
    method = 'Par.cond', k.type='gaussian', verbose = TRUE)
print(NAlm1)
NAlm2 <- ilse(age~., data=nhanes, method = 'Full.cond')
print(NAlm2)
## example two: No missing value
n <- 100
group <- rnorm(n, sd=4)
weight <- 3.2*group + 1.5 + rnorm(n, sd=0.1)
NAlm3 <- ilse(weight~group, data=data.frame(weight=weight, group=group),
    intercept = FALSE)
print(NAlm3)

kern

<table>
<thead>
<tr>
<th>Kernel Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>epk</td>
<td>Different type of kernel functions.</td>
</tr>
</tbody>
</table>

Usage

kern(u, type='epk')
Arguments

u  a numeric vector, evaluated points in kernel function.
type  a optional character string, specify the type of used kernel function and support 'epk', 'biweight', 'triangle', 'guassian', 'triweight', 'tricube', 'cosine', 'uniform' in current version, default as 'epk'.

Details

Note that \( K(u_i) = K(X_i - x_0) \) where \( u = (X_1 - x_0, \ldots, X_n - x_0) \) and \( K_h(u_i) = \frac{1}{h} K((X_i - x_0)/h) \) where \( h \) is bandwidth.

Value

Return a numeric vector with length equal to 'u'.

Author(s)

Liu Wei

See Also

KernSmooth package

Examples

library(graphics)
u <- seq(-1,1,by=0.01)
(Ku <- kern(u))
plot(u, Ku, type='l')
# gaussian kernel
plot(u, kern(u, type='gaussian'), type='l')
# cosine kernel
plot(u, Ku <- kern(u, type='cosine'), type='l')

Description

A small data set with missing values.

Format

A data frame with 25 observations on the following 4 variables. age: Age group (1=20-39, 2=40-59, 3=60+).
bmi: Body mass index (kg/m**2).
hyp: Hypertensive (1=no, 2=yes).
chl: Total serum cholesterol (mg/dL).
Details

A small data set with all numerical variables. The data set nhanes2 is the same data set, but with age and hyp treated as factors.

Source


Examples

# example one
data(nhanes)
bw <- 1
ilse(age~., data=nhanes,bw=bw)

print

Print the Information of FIML or ILSE methods

Description

print method for class "ilse" or class "fiml".

Usage

print(object)
## S3 method for class 'ilse'
print(object)

## S3 method for class 'fiml'
print(object)

Arguments

object an object of class "ilse" or "fiml".

Value

For "ilse", print the basic information of ilse estimation and algorithm and return a list including

beta a named vector of coefficients
Bmat a named matrix that summary the estimated beta in every iteration.
residuals the residuals, that is response minus fitted values.
fitted.values the fitted mean values.
d.fn a nonnegative value, vlaue of relative variation rate of objective function value
**Summary**

- **d.par**: a nonnegative value, relative variation rate of estimated parametric vector when algorithm stopped.
- **K**: a positive integer, iterative times in total.

For "fiml", print the basic information of fiml estimation and return a list including

- **beta**: A named vector of coefficients
- **iterations**: A positive integer, iterative times in total.
- **stop.code**: The stop code returned by nlm.

**See Also**

print.lm

**Examples**

```r
data(nhanes)
NAlm1 <- ilse(age~., data=nhanes)
a <- print(NAlm1)
a

fimllm <- fimlreg(age~., data=nhanes, iterlim=40)
b <- print(fimllm)
b
```

---

**summary**

_Summarizing the inference information for ILSE or FIML methods_

**Description**

summary method for class "ilse" or "fiml".

**Usage**

```r
summary(object, Nbt=20)
```

```r
## S3 method for class 'ilse'
summary(object, Nbt=20)
```

```r
## S3 method for class 'fiml'
summary(object, Nbt=20)
```

```r
##
Fitted.values(object)
##
Residuals(object)
```
Arguments

- **object**: an object of class "ilse".
- **Nbt**: an positive integer, the repeated times of bootstrap to estimate covariance matrix of regression coefficient.

Value

The function `summary.ilse` computes and returns a named matrix of summary statistics of the fitted linear model given in object by ILSE or FIML methods. The function `Fitted.values` return a vector, fitted response values. The function `Residuals` return a vector, residuals.

See Also

`summary.lm`, `fitted.values`, `residuals`

Examples

```r
# example one
data(nhanes)
NAlm <- ilse(age~., data=nhanes)
summary(NAlm, Nbt=5)

fimllm <- fimlreg(age~., data=nhanes)
summary(fimllm, Nbt = 5)
```
Index

* Full Information Maximum Likelihood
  fimlreg, 5

* Linear Model
  fimlreg, 5
  ilse, 6

* Missing Value
  ilse, 6

Coef, 2
cor.mat, 3
cov.mat, 4

fimlreg, 5
Fitted.values (summary), 11

ilse, 6, 6
kern, 8
lm, 8
nhanes, 9
print, 10
Residuals (summary), 11
summary, 11