

Package ‘fANCOVA’

July 22, 2025

Title Nonparametric Analysis of Covariance

Version 0.6-1

Description A collection of R functions to perform nonparametric analysis of covariance for regression curves or surfaces. Testing the equality or parallelism of nonparametric curves or surfaces is equivalent to analysis of variance (ANOVA) or analysis of covariance (ANCOVA) for one-sample functional data. Three different testing methods are available in the package, including one based on L-2 distance, one based on an ANOVA statistic, and one based on variance estimators.

License GPL-3

Encoding UTF-8

LazyData true

RoxygenNote 7.1.1

NeedsCompilation no

Author Xiaofeng Wang [aut, cre],
Xinge Ji [ctb]

Maintainer Xiaofeng Wang <wangx6@ccf.org>

Repository CRAN

Date/Publication 2020-11-13 16:40:02 UTC

Contents

loess.ancova	2
loess.as	4
plot.fANCOVA	6
T.aov	7
T.L2	10
T.var	12
USpopu	15
wild.boot	16

Index	18
--------------	-----------

loess.ancova	<i>Fit a semiparametric ANCOVA model with a local polynomial smoother</i>
--------------	---

Description

Fit a semiparametric ANCOVA model with a local polynomial smoother. The specific model considered here is

$$y_{ij} = g_i + m(x_{ij}) + e_{ij},$$

where the parametric part of the model, g_i , is a factor variable; the nonparametric part of the model, $m(\cdot)$, is a nonparametric smooth function; e_{ij} are independent identically distributed errors. The errors e_{ij} do not have to be independent $N(0, \sigma^2)$ errors. The errors can be heteroscedastic, i.e., $e_{ij} = \sigma_i(x_{ij}) * u_{ij}$, where u_{ij} are independent identically distributed errors with mean 0 and variance 1. The model is fitted by the direct estimation method (Speckman, 1988), or by the backfitting method (Buja, Hastie and Tibshirani, 1989; Hastie and Tibshirani, 1990).

Usage

```
loess.ancova(x, y, group, degree = 2, criterion = c("aicc", "gcv"),
family = c("gaussian", "symmetric"), method=c("Speckman", "Backfitting"),
iter = 10, tol = 0.01, user.span = NULL, plot = FALSE,
data.points = FALSE, legend.position = "topright", ...)
```

Arguments

x	a vector or two-column matrix of covariate values.
y	a vector of response values.
group	a vector of group indicators that has the same length as y.
degree	the degree of the local polynomials to be used. It can be 0, 1 or 2.
criterion	the criterion for automatic smoothing parameter selection: "aicc" denotes bias-corrected AIC criterion, "gcv" denotes generalized cross-validation.
family	if "gaussian" fitting is by least-squares, and if "symmetric" a re-descending M estimator is used with Tukey's biweight function.
method	if "Speckman" the direct estimation method by Speckman (1988) will be used, and if "Backfitting" The model is fitted by the backfitting method (Buja, Hastie and Tibshirani, 1989; Hastie and Tibshirani, 1990).
iter	the number of iterations.
tol	the number of tolerance in the iterations.
user.span	the user-defined parameter which controls the degree of smoothing. If it is not specified, the smoothing parameter will be selected by "aicc" or "gcv" criterion.
plot	if TRUE (when x is one-dimensional), the fitted curves for all groups will be generated; if TRUE (when x is two-dimensional), only the smooth component in the model will be plotted.

`data.points` if TRUE, the data points will be displayed in the plot.
`legend.position` the position of legend in the plot: “topright”, “topleft”, “bottomright”, “bottom-left”, etc.
`...` control parameters.

Details

Fit a local polynomial regression with automatic smoothing parameter selection. The predictor x can either one-dimensional or two-dimensional.

Value

a list of a vector of the parametric estimates and an object of class “loess”.

Author(s)

X.F. Wang <wangx6@ccf.org>

References

Speckman, P. (1988). Kernel Smoothing in Partial Linear Models. *Journal of the Royal Statistical Society. Series B (Methodological)*, 50, 413–436.

Buja, A., Hastie, T. J. and Tibshirani, R. J. (1989). Linear smoothers and additive models (with discussion). *Annals of Statistics*, 17, 453–555.

Hastie, T. J. and Tibshirani, R. J. (1990). *Generalized Additive Models*. Vol. 43 of Monographs on Statistics and Applied Probability, Chapman and Hall, London.

See Also

[loess](#).

Examples

```
## Fit semiparametric ANCOVA model
set.seed(555)
n1 <- 80
x1 <- runif(n1,min=0, max=3)
sd1 <- 0.3
e1 <- rnorm(n1,sd=sd1)
y1 <- 3*cos(pi*x1/2) + 6 + e1

n2 <- 75
x2 <- runif(n2, min=0, max=3)
sd2 <- 0.2
e2 <- rnorm(n2, sd=sd2)
y2 <- 3*cos(pi*x2/2) + 3 + e2

n3 <- 90
x3 <- runif(n3, min=0, max=3)
```

```

sd3 <- 0.3
e3 <- rnorm(n3, sd=sd3)
y3 <- 3*cos(pi*x3/2) + e3

data.bind <- rbind(cbind(x1,y1,1), cbind(x2,y2,2),cbind(x3,y3,3))
data.bind <- data.frame(data.bind)
colnames(data.bind)=c('x','y','group')

x <- data.bind[,1]
y <- data.bind[,2]
group <- data.bind[,3]

loess.ancova(x,y,group, plot=TRUE, data.points=TRUE)

## Fit semiparametric ANCOVA model when the predictor is two-dimensional
n1 <- 100
x11 <- runif(n1,min=0, max=3)
x12 <- runif(n1,min=0, max=3)
sd1 <- 0.2
e1 <- rnorm(n1,sd=sd1)
y1 <- sin(2*x11) + sin(2*x12) + e1

n2 <- 100
x21 <- runif(n2, min=0, max=3)
x22 <- runif(n2, min=0, max=3)
sd2 <- 0.25
e2 <- rnorm(n2, sd=sd2)
y2 <- sin(2*x21) + sin(2*x22) + 1 + e2

n3 <- 120
x31 <- runif(n3, min=0, max=3)
x32 <- runif(n3, min=0, max=3)
sd3 <- 0.25
e3 <- rnorm(n3, sd=sd3)
y3 <- sin(2*x31) + sin(2*x32) + 3 + e3

data.bind <- rbind(cbind(x11, x12 ,y1,1), cbind(x21, x22, y2,2),cbind(x31, x32,y3,3))
data.bind <- data.frame(data.bind)
colnames(data.bind)=c('x1','x2', 'y','group')

loess.ancova(data.bind[,c(1,2)], data.bind$y, data.bind$group, plot=TRUE)

```

loess.as

Fit a local polynomial regression with automatic smoothing parameter selection

Description

Fit a local polynomial regression with automatic smoothing parameter selection. Two methods are available for the selection of the smoothing parameter: bias-corrected Akaike information criterion (aicc); and generalized cross-validation (gcv).

Usage

```
loess.as(x, y, degree = 1, criterion = c("aicc", "gcv"),
family = c("gaussian", "symmetric"), user.span = NULL,
plot = FALSE, ...)
```

Arguments

x	a vector or two-column matrix of covariate values.
y	a vector of response values.
degree	the degree of the local polynomials to be used. It can be 0, 1 or 2.
criterion	the criterion for automatic smoothing parameter selection: "aicc" denotes bias-corrected AIC criterion, "gcv" denotes generalized cross-validation.
family	if "gaussian" fitting is by least-squares, and if "symmetric" a re-descending M estimator is used with Tukey's biweight function.
user.span	the user-defined parameter which controls the degree of smoothing.
plot	if TRUE, the fitted curve or surface will be generated.
...	control parameters.

Details

Fit a local polynomial regression with automatic smoothing parameter selection. The predictor x can either be one-dimensional or two-dimensional.

Value

An object of class "loess".

Author(s)

X.F. Wang <wangx6@ccf.org>

References

Cleveland, W. S. (1979) Robust locally weighted regression and smoothing scatterplots. *Journal of the American Statistical Association*. 74, 829–836.

Hurvich, C.M., Simonoff, J.S., and Tsai, C.L. (1998), Smoothing Parameter Selection in Nonparametric Regression Using an Improved Akaike Information Criterion. *Journal of the Royal Statistical Society B*. 60, 271–293.

Golub, G., Heath, M. and Wahba, G. (1979). Generalized cross validation as a method for choosing a good ridge parameter. *Technometrics*. 21, 215–224.

See Also

[loess](#), [loess.ancova](#), [T.L2](#), [T.aov](#), [T.var](#).

Examples

```
## Fit Local Polynomial Regression with Automatic Smoothing Parameter Selection
n1 <- 100
x1 <- runif(n1,min=0, max=3)
sd1 <- 0.2
e1 <- rnorm(n1,sd=sd1)
y1 <- sin(2*x1) + e1

(y1.fit <- loess.as(x1, y1, plot=TRUE))

n2 <- 100
x21 <- runif(n2, min=0, max=3)
x22 <- runif(n2, min=0, max=3)
sd2 <- 0.25
e2 <- rnorm(n2, sd=sd2)
y2 <- sin(2*x21) + sin(2*x22) + 1 + e2

(y2.fit <- loess.as(cbind(x21, x22), y2, plot=TRUE))
```

plot.fANCOVA

Plot a fANCOVA Object

Description

To plot a fANCOVA object

Usage

```
## S3 method for class 'fANCOVA'
plot(x, test.statistic=TRUE, main="", n=256, legend.position="topright", ...)
```

Arguments

x	a fANCOVA object
test.statistic	if TRUE, plot the density of the test statistic under null hypothesis; if FALSE, plot the estimated curves.
main	the title of the plot
n	the number of points that are used to draw the curves or surfaces in the plot.
legend.position	the position of legend in the plot: "topright", "topleft", "bottomright", "bottomleft", etc.
...	control parameters of the plot function

Details

This function is to plot a fANCOVA object. The plot will be generated only if the predictor x is one-dimensional. If “test.statistic=TRUE”, a density plot of the test statistic under null hypothesis will be generated; if “test.statistic=FALSE”, the estimated curves for all groups are drawn.

Author(s)

X.F. Wang <wangx6@ccf.org>

See Also

[T.L2](#), [T.aov](#), [T.var](#).

T.aov	<i>Test the equality of nonparametric curves or surfaces based on an ANOVA-type statistic</i>
-------	---

Description

Test the equality of nonparametric curves or surfaces based on an ANOVA-type statistic. The specific model considered here is

$$y_{ij} = m_i(x_{ij}) + e_{ij},$$

where $m_i(\cdot)$, are nonparametric smooth functions; e_{ij} are independent identically distributed errors. The errors e_{ij} do not have to be independent $N(0, \sigma^2)$ errors. The errors can be heteroscedastic, i.e., $e_{ij} = \sigma_i(x_{ij}) * u_{ij}$, where u_{ij} are independent identically distributed errors with mean 0 and variance 1.

We are interested in the problem of testing the equality of the regression curves (when x is one-dimensional) or surfaces (when x is two-dimensional),

$H_0: m_1(\cdot) = m_2(\cdot) = \dots$ v.s. $H_1: \text{otherwise}$

The problem can also be viewed as the test of the equality in the one-sample problem for functional data.

Usage

```
T.aov(x, ...)
## Default S3 method:
T.aov(x, y, group, B = 200, degree = 1, criterion = c("aicc", "gcv"),
family = c("gaussian", "symmetric"), tstat = c("DN", "YB"),
user.span = NULL, ...)
```

Arguments

x	a vector or two-column matrix of covariate values.
y	a vector of response values.
group	a vector of group indicators that has the same length as y.
B	the number of bootstrap replicates. Usually this will be a single positive integer.
degree	the degree of the local polynomials to be used. It can be 0, 1 or 2.
criterion	the criterion for automatic smoothing parameter selection: “aicc” denotes bias-corrected AIC criterion, “gcv” denotes generalized cross-validation.
family	if “gaussian” fitting is by least-squares, and if “symmetric” a re-descending M estimator is used with Tukey’s biweight function.
tstat	the test statistic used here: if “DN” Dette, H., Neumeyer, N. (2001)’s statistic is used; if “YB” Young, S.G. and Bowman, A.W. (1995)’s statistic is used.
user.span	The user-defined parameter which controls the degree of smoothing.
...	some control parameters can also be supplied directly

Details

A wild bootstrap algorithm is applied to test the equality of nonparametric curves or surfaces based on an ANOVA-type statistic.

Value

An object of class “fANCOVA”

Author(s)

X.F. Wang <wangx6@ccf.org>

References

- Dette, H., Neumeyer, N. (2001). Nonparametric analysis of covariance. *Annals of Statistics*. 29, 1361–1400.
- Young, S.G. and Bowman, A.W. (1995). Nonparametric analysis of covariance. *Biometrics*. 51, 920–931.
- Wang, X.F. and Ye, D. (2010). On nonparametric comparison of images and regression surfaces. *Journal of Statistical Planning and Inference*. 140, 2875–2884.

See Also

[T.L2](#), [T.var](#), [loess.as](#), [loess.ancova](#).

Examples

```

## Nonparametric test the equality of multiple regression curves
## Simulate data sets

n1 <- 100
x1 <- runif(n1,min=0, max=3)
sd1 <- 0.2
e1 <- rnorm(n1,sd=sd1)
y1 <- sin(2*x1) + e1

n2 <- 100
x2 <- runif(n2, min=0, max=3)
sd2 <- 0.25
e2 <- rnorm(n2, sd=sd2)
y2 <- sin(2*x2) + 1 + e2

n3 <- 120
x3 <- runif(n3, min=0, max=3)
sd3 <- 0.25
e3 <- rnorm(n3, sd=sd3)
y3 <- sin(2*x3) + e3

data.bind <- rbind(cbind(x1,y1,1), cbind(x2,y2,2),cbind(x3,y3,3))
data.bind <- data.frame(data.bind)
colnames(data.bind)=c('x','y','group')

t1 <- T.aov(data.bind$x, data.bind$y, data.bind$group)
t1
plot(t1)
plot(t1, test.statistic=FALSE)

#####
## Nonparametric test the equality for regression surfaces
## Simulate data sets

n1 <- 100
x11 <- runif(n1,min=0, max=3)
x12 <- runif(n1,min=0, max=3)
sd1 <- 0.2
e1 <- rnorm(n1,sd=sd1)
y1 <- sin(2*x11) + sin(2*x12) + e1

n2 <- 100
x21 <- runif(n2, min=0, max=3)
x22 <- runif(n2, min=0, max=3)
sd2 <- 0.25
e2 <- rnorm(n2, sd=sd2)
y2 <- sin(2*x21) + sin(2*x22) + 1 + e2

n3 <- 120
x31 <- runif(n3, min=0, max=3)

```

```

x32 <- runif(n3, min=0, max=3)
sd3 <- 0.25
e3 <- rnorm(n3, sd=sd3)
y3 <- sin(2*x31) + sin(2*x32) + e3

data.bind <- rbind(cbind(x11, x12 ,y1,1), cbind(x21, x22, y2,2),cbind(x31, x32,y3,3))
data.bind <- data.frame(data.bind)
colnames(data.bind)=c('x1','x2', 'y','group')

T.aov(data.bind[,c(1,2)], data.bind$y, data.bind$group)

```

T.L2 *Test the equality of nonparametric curves or surfaces based on L2 distance*

Description

Test the equality of nonparametric curves or surfaces based on L2 distance. The specific model considered here is

$$y_{ij} = m_i(x_{ij}) + e_{ij},$$

where $m_i(\cdot)$, are nonparametric smooth functions; e_{ij} are independent identically distributed errors. The errors e_{ij} do not have to be independent $N(0, \sigma^2)$ errors. The errors can be heteroscedastic, i.e., $e_{ij} = \sigma_i(x_{ij}) * u_{ij}$, where u_{ij} are independent identically distributed errors with mean 0 and variance 1.

We are interested in the problem of testing the equality of the regression curves (when x is one-dimensional) or surfaces (when x is two-dimensional),

$H_0: m_1(\cdot) = m_2(\cdot) = \dots$ v.s. $H_1: \text{otherwise}$

The problem can also be viewed as the test of the equality in the one-sample problem for functional data.

Usage

```

T.L2(x, ...)
## Default S3 method:
T.L2(x, y, group, B = 200, degree = 1, criterion = c("aicc", "gcv"),
family = c("gaussian", "symmetric"), m = 225, user.span = NULL, ...)

```

Arguments

x a vector or two-column matrix of covariate values.

y a vector of response values.

group a vector of group indicators that has the same length as **y**.

B the number of bootstrap replicates. Usually this will be a single positive integer.

degree the degree of the local polynomials to be used. It can be 0, 1 or 2.

criterion	the criterion for automatic smoothing parameter selection: “aicc” denotes bias-corrected AIC criterion, “gcv” denotes generalized cross-validation.
family	if “gaussian” fitting is by least-squares, and if “symmetric” a re-descending M estimator is used with Tukey’s biweight function.
m	the number of the sampling points for the Monte-Carlo integration.
user.span	the user-defined parameter which controls the degree of smoothing.
...	some control parameters can also be supplied directly.

Details

A wild bootstrap algorithm is applied to test the equality of nonparametric curves or surfaces based on L2 distance.

Value

An object of class “fANCOVA”.

Author(s)

X.F. Wang <wangx6@ccf.org>

References

- Dette, H., Neumeyer, N. (2001). Nonparametric analysis of covariance. *Annals of Statistics*. 29, 1361–1400.
- Wang, X.F. and Ye, D. (2010). On nonparametric comparison of images and regression surfaces. *Journal of Statistical Planning and Inference*. 140, 2875–2884.

See Also

[T.aov](#), [T.var](#), [loess.as](#), [loess.ancova](#).

Examples

```
## Nonparametric test the equality of multiple regression curves
## Simulate data sets

n1 <- 100
x1 <- runif(n1,min=0, max=3)
sd1 <- 0.2
e1 <- rnorm(n1,sd=sd1)
y1 <- sin(2*x1) + e1

n2 <- 100
x2 <- runif(n2, min=0, max=3)
sd2 <- 0.25
e2 <- rnorm(n2, sd=sd2)
y2 <- sin(2*x2) + 1 + e2

n3 <- 120
```

```

x3 <- runif(n3, min=0, max=3)
sd3 <- 0.25
e3 <- rnorm(n3, sd=sd3)
y3 <- sin(2*x3) + e3

data.bind <- rbind(cbind(x1,y1,1), cbind(x2,y2,2),cbind(x3,y3,3))
data.bind <- data.frame(data.bind)
colnames(data.bind)=c('x','y','group')

t1 <- T.L2(data.bind$x, data.bind$y, data.bind$group, degree=2)
t1
plot(t1)
plot(t1, test.statistic=FALSE)

#####
## Nonparametric test the equality for regression surfaces
## Simulate data sets

n1 <- 100
x11 <- runif(n1,min=0, max=3)
x12 <- runif(n1,min=0, max=3)
sd1 <- 0.2
e1 <- rnorm(n1,sd=sd1)
y1 <- sin(2*x11) + sin(2*x12) + e1

n2 <- 100
x21 <- runif(n2, min=0, max=3)
x22 <- runif(n2, min=0, max=3)
sd2 <- 0.25
e2 <- rnorm(n2, sd=sd2)
y2 <- sin(2*x21) + sin(2*x22) + 1 + e2

n3 <- 120
x31 <- runif(n3, min=0, max=3)
x32 <- runif(n3, min=0, max=3)
sd3 <- 0.25
e3 <- rnorm(n3, sd=sd3)
y3 <- sin(2*x31) + sin(2*x32) + e3

data.bind <- rbind(cbind(x11, x12 ,y1,1), cbind(x21, x22, y2,2),cbind(x31, x32,y3,3))
data.bind <- data.frame(data.bind)
colnames(data.bind)=c('x1','x2', 'y','group')

T.L2(data.bind[,c(1,2)], data.bind$y, data.bind$group, degree=2)

```

T.var

Test the equality of nonparametric curves or surfaces based on variance estimators

Description

Test the equality of nonparametric curves or surfaces based on variance estimators. The specific model considered here is

$$y_{ij} = m_i(x_{ij}) + e_{ij},$$

where $m_i(\cdot)$, are nonparametric smooth functions; e_{ij} are independent identically distributed errors. The errors e_{ij} do not have to be independent $N(0, \sigma^2)$ errors. The errors can be heteroscedastic, i.e., $e_{ij} = \sigma_i(x_{ij}) * u_{ij}$, where u_{ij} are independent identically distributed errors with mean 0 and variance 1.

We are interested in the problem of testing the equality of the regression curves (when x is one-dimensional) or surfaces (when x is two-dimensional),

$H_0: m_1(\cdot) = m_2(\cdot) = \dots$ v.s. $H_1: \text{otherwise}$

The problem can also be viewed as the test of the equality in the one-sample problem for functional data.

Usage

```
T.var(x, ...)
## Default S3 method:
T.var(x, y, group, B = 200, degree = 1, criterion = c("aicc", "gcv"),
family = c("gaussian", "symmetric"), user.span = NULL, ...)
```

Arguments

<code>x</code>	a vector or two-column matrix of covariate values.
<code>y</code>	a vector of response values.
<code>group</code>	a vector of group indicators that has the same length as <code>y</code> .
<code>B</code>	the number of bootstrap replicates. Usually this will be a single positive integer.
<code>degree</code>	the degree of the local polynomials to be used. It can be 0, 1 or 2.
<code>criterion</code>	the criterion for automatic smoothing parameter selection: "aicc" denotes bias-corrected AIC criterion, "gcv" denotes generalized cross-validation.
<code>family</code>	if "gaussian" fitting is by least-squares, and if "symmetric" a re-descending M estimator is used with Tukey's biweight function.
<code>user.span</code>	the user-defined parameter which controls the degree of smoothing.
<code>...</code>	some control parameters can also be supplied directly

Details

A wild bootstrap algorithm is applied to test the equality of nonparametric curves or surfaces based on variance estimators.

Value

An object of class "fANCOVA".

Author(s)

X.F. Wang <wangx6@ccf.org>

References

- Dette, H., Neumeyer, N. (2001). Nonparametric analysis of covariance. *Annals of Statistics*. 29, 1361–1400.
- Wang, X.F. and Ye, D. (2010). On nonparametric comparison of images and regression surfaces. *Journal of Statistical Planning and Inference*. 140, 2875–2884.

See Also

[T.L2](#), [T.aov](#), [loess.as](#), [loess.ancova](#).

Examples

```
## Nonparametric test the equality of multiple regression curves
## Simulate data sets

n1 <- 100
x1 <- runif(n1,min=0, max=3)
sd1 <- 0.2
e1 <- rnorm(n1,sd=sd1)
y1 <- sin(2*x1) + e1

n2 <- 100
x2 <- runif(n2, min=0, max=3)
sd2 <- 0.25
e2 <- rnorm(n2, sd=sd2)
y2 <- sin(2*x2) + 1 + e2

n3 <- 120
x3 <- runif(n3, min=0, max=3)
sd3 <- 0.25
e3 <- rnorm(n3, sd=sd3)
y3 <- sin(2*x3) + e3

data.bind <- rbind(cbind(x1,y1,1), cbind(x2,y2,2),cbind(x3,y3,3))
data.bind <- data.frame(data.bind)
colnames(data.bind)=c('x','y','group')

t1 <- T.var(data.bind$x, data.bind$y, data.bind$group, degree=2, criterion="gcv")
t1
plot(t1)
plot(t1, test.statistic=FALSE)

#####
## Nonparametric test the equality for regression surfaces
## Simulate data sets
```

```

n1 <- 100
x11 <- runif(n1,min=0, max=3)
x12 <- runif(n1,min=0, max=3)
sd1 <- 0.2
e1 <- rnorm(n1,sd=sd1)
y1 <- sin(2*x11) + sin(2*x12) + e1

n2 <- 100
x21 <- runif(n2, min=0, max=3)
x22 <- runif(n2, min=0, max=3)
sd2 <- 0.25
e2 <- rnorm(n2, sd=sd2)
y2 <- sin(2*x21) + sin(2*x22) + 1 + e2

n3 <- 120
x31 <- runif(n3, min=0, max=3)
x32 <- runif(n3, min=0, max=3)
sd3 <- 0.25
e3 <- rnorm(n3, sd=sd3)
y3 <- sin(2*x31) + sin(2*x32) + e3

data.bind <- rbind(cbind(x11, x12 ,y1,1), cbind(x21, x22, y2,2),cbind(x31, x32,y3,3))
data.bind <- data.frame(data.bind)
colnames(data.bind)=c('x1', 'x2', 'y', 'group')

T.var(data.bind[,c(1,2)], data.bind$y, data.bind$group)

```

USpopu

US national population

Description

US national population by four groups from 1900 to 1979. The four groups are: Age 0; Age 20; Age 40; Age 60.

Usage

```
data(USpopu)
```

Format

A data frame with 320 observations on 3 variables.

age	numeric	the group variable of age
year	numeric	a numeric vector, giving year
population	numeric	a numeric vector, giving population in millions

References

<https://www.census.gov/data/tables/time-series/demo/popest/pre-1980-national.html>,
U.S. Census Bureau, National Intercensal Tables: 1900-1990. Last Revised: November 30, 2016

See Also

[T.L2](#), [T.aov](#), [T.var](#).

Examples

```
data(USpopu)
t1 <- T.L2(USpopu$year, USpopu$population, USpopu$age, degree=2)
t1
plot(t1)
plot(t1, test.statistic=FALSE, legend.position="topleft")
```

wild.boot

Generate one or multiple bootstrap samples of regression residuals using the wild bootstrap method

Description

Generate bootstrap samples using the wild bootstrap method introduced by Wu (1986). One of the advantages for the wild bootstrap method is that it allows for a heterogeneous variance in the residuals in regression analysis.

Usage

```
wild.boot(x, nboot = 1)
```

Arguments

`x` a vector of regression residuals.
`nboot` the number of bootstrap replicates. Usually this will be a single positive integer.

Details

This function is to generate bootstrap residuals using the wild bootstrap method.

Value

a vector or a matrix.

Author(s)

X.F. Wang <wangx6@ccf.org>

References

- Wu, C. (1986) Jackknife, bootstrap and other resampling methods in regression analysis (with discussion). *Annals of Statistics*. 14, 1261–1350.
- Mammen, E. (1991). Bootstrap, wild bootstrap, and asymptotic normality. *Probability Theory and Related Fields*. 93, 439–455.

See Also

[T.L2](#), [T.aov](#), [T.var](#).

Examples

```
n <- 1000
x <- runif(n, min=0, max=1)
## generate heteroscedastic error variances
sig.x <- sqrt(exp(x)/2.5-0.4)
err <- sapply(sig.x, function(x) rnorm(1, sd=x))
x2 <- x^2
y <- 10+3*x+2*x2 +err
plot(x,y)
fit <- lm(y ~ x + x2)
## obtain 12 samples of the wild bootstrap residuals
res.boot <- wild.boot(fit$res, nboot=12)
## obtain 12 samples of the wild bootstrap responses
y.boot <- matrix(rep(fit$fit,time=12), ncol=12) + res.boot
## plot the 12 wild bootstrap samples
## The wild bootstrap method keeps the patterns of variance heterogeneity
## in the original sample.
par(mfrow=c(4,3))
for (i in 1:12) plot(x, y.boot[,i])
```

Index

- * **bootstrap**
 - wild.boot, 16
 - * **datasets**
 - USpopu, 15
 - * **nonparametric**
 - loess.ancova, 2
 - loess.as, 4
 - T.aov, 7
 - T.L2, 10
 - T.var, 12
 - * **plot**
 - plot.fANCOVA, 6
 - * **regression**
 - wild.boot, 16
 - * **semiparametric**
 - loess.ancova, 2
 - * **smoothing**
 - loess.ancova, 2
 - * **smooth**
 - loess.as, 4
 - T.aov, 7
 - T.L2, 10
 - T.var, 12
 - * **test the equality**
 - T.aov, 7
 - T.L2, 10
 - T.var, 12
 - * **wild bootstrap**
 - wild.boot, 16
- loess, 3, 5
loess.ancova, 2, 5, 8, 11, 14
loess.as, 4, 8, 11, 14
- plot (plot.fANCOVA), 6
plot.fANCOVA, 6
- T.aov, 5, 7, 7, 11, 14, 16, 17
T.L2, 5, 7, 8, 10, 14, 16, 17
T.var, 5, 7, 8, 11, 12, 16, 17
- USpopu, 15
wild.boot, 16