Package ‘AID’

Type Package
Title Box-Cox Power Transformation
Version 2.9
Date 2023-09-13
Depends R (>= 3.2.0)
Imports MASS, tseries, nortest, ggplot2, graphics, psych, stats, meta,
        stringr
Suggests onewaytests
Author Osman Dag [aut, cre], Muhammed Ali Yilmaz [aut], Ozgur Asar [ctb], Ozlem Ilk [ctb]
Maintainer Osman Dag <osman.dag@outlook.com>
Description Performs Box-Cox power transformation for different purposes, graphical approaches, assesses the success of the transformation via tests and plots, computes mean and confidence interval for back transformed data.
License GPL (>= 2)
NeedsCompilation no
Repository CRAN
Date/Publication 2023-09-13 11:42:34 UTC

R topics documented:

  AID-package .................................................. 2
  AADT .......................................................... 2
  boxcoxfr ...................................................... 3
  boxcoxlm ...................................................... 6
  boxcoxmeta ................................................... 8
  boxcoxnc ...................................................... 9
  confInt.boxcoxfr ............................................ 11
  confInt.boxcoxmeta .......................................... 13
  confInt.boxcoxnc ............................................ 14
  grades ......................................................... 15
  textile ......................................................... 15

Index 17
AID-package

Box-Cox Power Transformation

Description

Performs Box-Cox power transformation for different purposes, graphical approaches, assesses the success of the transformation via tests and plots, computes mean and confidence interval for back transformed data.

Details

Package: AID
Type: Package
License: GPL (>=2)

AADT

Average Annual Daily Traffic Data

Description

Average annual daily traffic data collected from the Minnesota Department of Transportation database.

Usage

data(AADT)

Format

A data frame with 121 observations on the following 8 variables.

aadt  average annual daily traffic for a section of road
ctypop population of county
lanes  number of lanes in the section of road
width  width of the section of road (in feet)
control  a factor with levels: access control; no access control
class  a factor with levels: rural interstate; rural noninterstate; urban interstate; urban noninterstate truck  availability situation of road section to trucks
locale  a factor with levels: rural; urban, population <= 50,000; urban, population > 50,000
References


Examples

library(AID)

data(AADT)
attach(AADT)

hist(aadt)

out <- boxcoxfr(aadt, class)

confInt(out)

boxcoxfr

Box-Cox Transformation for One-Way ANOVA

Description

boxcoxfr performs Box-Cox transformation for one-way ANOVA. It is useful to use if the normality or/and the homogeneity of variance is/are not satisfied while comparing two or more groups.

Usage

boxcoxfr(y, x, option = "both", lambda = seq(-3, 3, 0.01), lambda2 = NULL, tau = 0.05, alpha = 0.05, verbose = TRUE)

Arguments

y a numeric vector of data values.

x a vector or factor object which gives the group for the corresponding elements of y.

option a character string to select the desired option for the objective of transformation. "nor" and "var" are the options which search for a transformation to satisfy the normality of groups and the homogeneity of variances, respectively. "both" is the option which searches for a transformation to satisfy both the normality of groups and the homogeneity of variances. Default is set to "both".

lambda a vector which includes the sequence of feasible lambda values. Default is set to (-3, 3) with increment 0.01.

lambda2 a numeric for an additional shifting parameter. Default is set to lambda2 = 0.

tau the feasible region parameter for the construction of feasible region. Default is set to 0.05. If tau = 0, it returns the MLE of transformation parameter.
alpha the level of significance to check the normality and variance homogeneity after transformation. Default is set to alpha = 0.05.

verbose a logical for printing output to R console.

Details

Denote $y$ the variable at the original scale and $y'$ the transformed variable. The Box-Cox power transformation is defined by:

$$
y' = \begin{cases} 
    \frac{y^{\lambda - 1}}{\lambda} , & \text{if } \lambda \neq 0 \\
    \log(y) , & \text{if } \lambda = 0 
\end{cases}
$$

If the data include any nonpositive observations, a shifting parameter $\lambda_2$ can be included in the transformation given by:

$$
y' = \begin{cases} 
    \frac{(y+\lambda_2)^{\lambda - 1}}{\lambda} , & \text{if } \lambda \neq 0 \\
    \log(y + \lambda_2) , & \text{if } \lambda = 0 
\end{cases}
$$

Maximum likelihood estimation in feasible region (MLEFR) is used while estimating transformation parameter. MLEFR maximizes the likelihood function in feasible region constructed by Shapiro-Wilk test and Bartlett’s test. After transformation, normality of the data in each group and homogeneity of variance are assessed by Shapiro-Wilk test and Bartlett’s test, respectively.

Value

A list with class "boxcoxfr" containing the following elements:

- method method applied in the algorithm
- lambda.hat the estimated lambda
- lambda2 additional shifting parameter
- shapiro a data frame which gives the test results for the normality of groups via Shapiro-Wilk test
- bartlett a matrix which returns the test result for the homogeneity of variance via Bartlett’s test
- alpha the level of significance to assess the assumptions.
- tf.data transformed data set
- x a factor object which gives the group for the corresponding elements of y
- y.name variable name of y
- x.name variable name of x

Author(s)

Osman Dag, Ozlem Ilk
References


Examples

```
####
# Communication between AID and onewaytests packages
library(AID)
library(onewaytests)

# Average Annual Daily Traffic Data (AID)
data(AADT)

# to obtain descriptive statistics by groups (onewaytests)
describe(aadt ~ class, data = AADT)

# to check normality of data in each group (onewaytests)
nor.test(aadt ~ class, data = AADT)

# to check variance homogeneity (onewaytests)
homog.test(aadt ~ class, data = AADT, method = "Bartlett")

# to apply Box-Cox transformation (AID)
out <- boxcoxfr(AADT$aadt, AADT$class)

# to obtain transformed data
AADT$tf.aadt <- out$tf.data

# to conduct one-way ANOVA with transformed data (onewaytests)
result <- aov.test(tf.aadt ~ class, data = AADT)

# to make pairwise comparison (onewaytests)
paircomp(result)

# to convert the statistics into the original scale (AID)
confInt(out, level = 0.95)
####
```

```
library(AID)
data <- rnorm(120, 10, 1)
factor <- rep(c("X", "Y", "Z"), each = 40)
out <- boxcoxfr(data, factor, lambda = seq(-5, 5, 0.01), tau = 0.01, alpha = 0.01)
confInt(out, level = 0.95)
```
boxcoxlm performs Box-Cox transformation for linear models and provides graphical analysis of residuals after transformation.

Usage

```r
boxcoxlm(x, y, method = "lse", lambda = seq(-3, 3, 0.01), lambda2 = NULL, plot = TRUE, alpha = 0.05, verbose = TRUE)
```

Arguments

- **x**: a n*p* matrix, *n* is the number of observations and *p* is the number of variables.
- **y**: a vector of response variable.
- **method**: a character string to select the desired method to be used to estimate Box-Cox transformation parameter. To use Shapiro-Wilk test method should be set to "sw". For method = "ad", boxcoxnc function uses Anderson-Darling test to estimate Box-Cox transformation parameter. Similarly, method should be set to "cvm", "pt", "sf", "lt", "jb", "mle", "lse" to use Cramer-von Mises, Pearson Chi-square, Shapiro-Francia, Lilliefors and Jarque-Bera tests, maximum likelihood estimation and least square estimation, respectively. Default is set to method = "lse".
- **lambda**: a vector which includes the sequence of candidate lambda values. Default is set to (-3,3) with increment 0.01.
- **lambda2**: a numeric for an additional shifting parameter. Default is set to lambda2 = 0.
- **plot**: a logical to plot histogram with its density line and qqplot of residuals before and after transformation. Defaults plot = TRUE.
- **alpha**: the level of significance to assess the normality of residuals after transformation. Default is set to alpha = 0.05.
- **verbose**: a logical for printing output to R console.

Details

Denote *y* the variable at the original scale and *y'* the transformed variable. The Box-Cox power transformation is defined by:

\[
y' = \begin{cases} 
  \frac{y^{\lambda} - 1}{\lambda} = \beta_0 + \beta_1 x_1 + ... + \epsilon, & \text{if } \lambda \neq 0 \\
  \log(y) = \beta_0 + \beta_1 x_1 + ... + \epsilon, & \text{if } \lambda = 0 
\end{cases}
\]
If the data include any nonpositive observations, a shifting parameter $\lambda_2$ can be included in the transformation given by:

$$y' = \begin{cases} 
\frac{(y+\lambda_2)^{\lambda-1}}{\lambda} = \beta_0 + \beta_1x_1 + \ldots + \epsilon, & \text{if } \lambda \neq 0 \\
\log(y + \lambda_2) = \beta_0 + \beta_1x_1 + \ldots + \epsilon, & \text{if } \lambda = 0 
\end{cases}$$

Maximum likelihood estimation and least square estimation are equivalent while estimating Box-Cox power transformation parameter (Kutner et al., 2005). Therefore, these two methods return the same result.

**Value**

A list with class "boxcoxlm" containing the following elements:

- `method` method preferred to estimate Box-Cox transformation parameter
- `lambda.hat` estimate of Box-Cox Power transformation parameter based on corresponding method
- `lambda2` additional shifting parameter
- `statistic` statistic of normality test for residuals after transformation based on specified normality test in method. For mle and lse, statistic is obtained by Shapiro-Wilk test for residuals after transformation
- `p.value` p.value of normality test for residuals after transformation based on specified normality test in method. For mle and lse, p.value is obtained by Shapiro-Wilk test for residuals after transformation
- `alpha` the level of significance to assess normality of residuals
- `tf.y` transformed response variable
- `tf.residuals` residuals after transformation
- `y.name` response name
- `x.name` x matrix name

**Author(s)**

Osman Dag, Ozlem Ilk

**References**


Examples

library(AID)
trees=as.matrix(trees)
boxcoxlm(x = trees[,1:2], y = trees[,3])

Description

boxcoxmeta performs ensemble based Box-Cox transformation via meta analysis for normality of a variable and provides graphical analysis.

Usage

boxcoxmeta(data, lambda = seq(-3,3,0.01), nboot = 100, lambda2 = NULL, plot = TRUE, alpha = 0.05, verbose = TRUE)

Arguments

data a numeric vector of data values.
lambda a vector which includes the sequence of candidate lambda values. Default is set to (-3,3) with increment 0.01.
nboot a number of Bootstrap samples to estimate standard errors of lambda estimates.
lambda2 a numeric for an additional shifting parameter. Default is set to lambda2 = 0.
plot a logical to plot histogram with its density line and qqplot of raw and transformed data. Defaults plot = TRUE.
alpha the level of significance to check the normality after transformation. Default is set to alpha = 0.05.
verbose a logical for printing output to R console.

Details

Denote \( y \) the variable at the original scale and \( y' \) the transformed variable. The Box-Cox power transformation is defined by:

\[
y' = \begin{cases} 
  \frac{y^{\lambda}-1}{\lambda}, & \text{if } \lambda \neq 0 \\
  \log(y), & \text{if } \lambda = 0 
\end{cases}
\]

If the data include any nonpositive observations, a shifting parameter \( \lambda_2 \) can be included in the transformation given by:

\[
y' = \begin{cases} 
  \frac{(y+\lambda_2)^{\lambda}-1}{\lambda}, & \text{if } \lambda \neq 0 \\
  \log(y + \lambda_2), & \text{if } \lambda = 0 
\end{cases}
\]
Value

A list with class "boxcoxmeta" containing the following elements:

- **method**
- **lambda.hat**
- **lambda2**
- **result**
- **alpha**
- **tf.data**
- **var.name**

Author(s)

Muhammed Ali Yilmaz, Osman Dag

References


Examples

```r
library(AID)
data(textile)

out <- boxcoxmeta(textile[,1])
out$lambda.hat # the estimate of Box-Cox parameter
out$tf.data # transformed data set
```

boxcoxnc performs Box-Cox transformation for normality of a variable and provides graphical analysis.

Usage

```r
boxcoxnc(data, method = "sw", lambda = seq(-3,3,0.01), lambda2 = NULL, plot = TRUE,
alpha = 0.05, verbose = TRUE)
```
**Arguments**

- **data**: a numeric vector of data values.
- **method**: a character string to select the desired method to be used to estimate Box-Cox transformation parameter. To use Shapiro-Wilk test method should be set to "sw". For method = "ad", boxcoxnc function uses Anderson-Darling test to estimate Box-Cox transformation parameter. Similarly, method should be set to "cvm", "pt", "sf", "lt", "jb", "ac", "mle" to use Cramer-von Mises, Pearson Chi-square, Shapiro-Francia, Lilliefors, Jarque-Bera tests, artificial covariate method and maximum likelihood estimation, respectively. Default is set to method = "sw".
- **lambda**: a vector which includes the sequence of candidate lambda values. Default is set to (-3,3) with increment 0.01.
- **lambda2**: a numeric for an additional shifting parameter. Default is set to lambda2 = 0.
- **plot**: a logical to plot histogram with its density line and qqplot of raw and transformed data. Defaults plot = TRUE.
- **alpha**: the level of significance to check the normality after transformation. Default is set to alpha = 0.05.
- **verbose**: a logical for printing output to R console.

**Details**

Denote $y$ the variable at the original scale and $y'$ the transformed variable. The Box-Cox power transformation is defined by:

$$
y' = \begin{cases} 
  y^{\lambda-1}, & \text{if } \lambda \neq 0 \\
  \log(y), & \text{if } \lambda = 0
\end{cases}
$$

If the data include any nonpositive observations, a shifting parameter $\lambda_2$ can be included in the transformation given by:

$$
y' = \begin{cases} 
  (y+\lambda_2)^{\lambda-1}, & \text{if } \lambda \neq 0 \\
  \log(y + \lambda_2), & \text{if } \lambda = 0
\end{cases}
$$

**Value**

A list with class "boxcoxnc" containing the following elements:

- **method**: method preferred to estimate Box-Cox transformation parameter
- **lambda.hat**: estimate of Box-Cox Power transformation parameter based on corresponding method
- **lambda2**: additional shifting parameter
- **statistic**: statistic of normality test for transformed data based on specified normality test in method. For artificial covariate method, statistic is obtained by Shapiro-Wilk test for transformed data
**confInt.boxcoxfr**

<table>
<thead>
<tr>
<th>p.value</th>
<th>p.value of normality test for transformed data based on specified normality test in method. For artificial covariate method, p.value is obtained by Shapiro-Wilk test for transformed data</th>
</tr>
</thead>
<tbody>
<tr>
<td>alpha</td>
<td>the level of significance to assess normality.</td>
</tr>
<tr>
<td>tf.data</td>
<td>transformed data set</td>
</tr>
<tr>
<td>var.name</td>
<td>variable name</td>
</tr>
</tbody>
</table>

**Author(s)**

Osman Dag, Ozgur Asar, Ozlem Ilk

**References**


**Examples**

```r
library(AID)
data(textile)

out <- boxcoxnc(textile[,1], method = "sw")
out$lambda.hat # the estimate of Box-Cox parameter based on Shapiro-Wilk test statistic
out$p.value # p.value of Shapiro-Wilk test for transformed data
out$tf.data # transformed data set
confInt(out) # mean and confidence interval for back transformed data

out2 <- boxcoxnc(textile[,1], method = "sf")
out2$lambda.hat # the estimate of Box-Cox parameter based on Shapiro-Francia test statistic
out2$p.value # p.value of Shapiro-Francia test for transformed data
out2$tf.data
confInt(out2)
```

**confInt.boxcoxfr**

*Mean and Asymmetric Confidence Interval for Back Transformed Data*

**Description**

`confInt.boxcoxfr` calculates mean and asymmetric confidence interval for back transformed data in each group and plots their error bars with confidence intervals.
Usage

```r
## S3 method for class 'boxcoxfr'
confInt(x, level = 0.95, plot = TRUE, xlab = NULL, ylab = NULL, title = NULL,
       width = NULL, verbose = TRUE, ...)
```

Arguments

- `x`: a boxcoxfr object.
- `level`: the confidence level.
- `plot`: a logical to plot error bars with confidence intervals.
- `xlab`: a label for the x axis, defaults to a description of `x`.
- `ylab`: a label for the y axis, defaults to a description of `y`.
- `title`: a main title for the plot.
- `width`: a numeric giving the width of the little lines at the tops and bottoms of the error bars (defaults to 0.15).
- `verbose`: a logical for printing output to R console.
- `...`: additional argument(s) for methods.

Details

Confidence interval in each group is constructed separately.

Value

A matrix with columns giving mean, lower and upper confidence limits for back transformed data. These will be labelled as (1 - `level`)/2 and 1 - (1 - `level`)/2 in % (by default 2.5% and 97.5%).

Author(s)

Osman Dag

Examples

```r
library(AID)
data(AADT)
attach(AADT)
out <- boxcoxfr(aadt, class)
confInt(out, level = 0.95)
```
Mean and Asymmetric Confidence Interval for Back Transformed Data

Description

confInt.boxcoxmeta calculates mean and asymmetric confidence interval for back transformed data.

Usage

## S3 method for class 'boxcoxmeta'
confInt(x, level = 0.95, verbose = TRUE, ...)

Arguments

x
a boxcoxmeta object.

level
the confidence level.

verbose
a logical for printing output to R console.

... additional argument(s) for methods.

Value

A matrix with columns giving mean, lower and upper confidence limits for back transformed data. These will be labelled as (1 - level)/2 and 1 - (1 - level)/2 in % (by default 2.5% and 97.5%).

Author(s)

Osman Dag, Muhammed Ali Yilmaz

Examples

library(AID)
data(textile)

out <- boxcoxmeta(textile[,1])
confInt(out) # mean and confidence interval for back transformed data
Description

confInt is a generic function to calculate mean and asymmetric confidence interval for back transformed data.

Usage

```r
## S3 method for class 'boxcoxnc'
confInt(x, level = 0.95, verbose = TRUE, ...)
```

Arguments

- `x`: a boxcoxnc object.
- `level`: the confidence level.
- `verbose`: a logical for printing output to R console.
- `...`: additional argument(s) for methods.

Value

A matrix with columns giving mean, lower and upper confidence limits for back transformed data. These will be labelled as $(1 - \text{level})/2$ and $1 - (1 - \text{level})/2$ in % (by default 2.5% and 97.5%).

Author(s)

Osman Dag

Examples

```r
library(AID)
data(textile)
out <- boxcoxnc(textile[,1])
confInt(out) # mean and confidence interval for back transformed data
```
### grades

*Student Grades Data*

**Description**

Overall student grades for a class taught by Dr. Ozlem Ilk

**Usage**

```r
data(grades)
```

**Format**

A data frame with 42 observations on the following variable.

- grades: a numeric vector for the student grades

**Examples**

```r
library(AID)

data(grades)
hist(grades[,1])
out <- boxcoxnc(grades[,1])
confInt(out, level = 0.95)
```

### textile

*Textile Data*

**Description**

Number of Cycles to Failure of Worsted Yarn

**Usage**

```r
data(textile)
```

**Format**

A data frame with 27 observations on the following variable.

- textile: a numeric vector for the number of cycles
References


Examples

```r
library(AID)
data(textile)
hist(textile[,1])
out <- boxcoxnc(textile[,1])
confInt(out)
```
Index

* datasets
  AADT, 2
  grades, 15
  textile, 15

* functions
  boxcoxfr, 3
  boxcoxlm, 6
  boxcoxmeta, 8
  boxcoxnc, 9
  confInt.boxcoxfr, 11
  confInt.boxcoxmeta, 13
  confInt.boxcoxnc, 14

AADT, 2
AID-package, 2

boxcoxfr, 3
boxcoxlm, 6
boxcoxmeta, 8
boxcoxnc, 9

confInt(confInt.boxcoxnc), 14
confInt.boxcoxfr, 11
confInt.boxcoxmeta, 13
confInt.boxcoxnc, 14

grades, 15

textile, 15