package: MASS

# boxcox

From EnvStats v2.3.1
by Alexander Kowarik

99.9
Perce

## Boxcox Power Transformation

`boxcox` is a generic function used to compute the value(s) of an objective for one or more Box-Cox transformations, or to compute an optimal power transformation based on a specified objective. The function invokes particular `methods` which depend on the `class` of the first argument.

Currently, there is a default method and a method for objects of class `"lm"` .

**Keywords** models, univar

## Usage

```
boxcox(x, ...)

# S3 method for default
boxcox(x,
    lambda = {if (optimize) c(-2, 2) else seq(-2, 2, by = 0.5)},
    optimize = FALSE, objective.name = "PPCC",
    eps = .Machine$double.eps, include.x = TRUE, ...)

# S3 method for lm
boxcox(x,
    lambda = {if (optimize) c(-2, 2) else seq(-2, 2, by = 0.5)},
    optimize = FALSE, objective.name = "PPCC",
    eps = .Machine$double.eps, include.x = TRUE, ...)
```

## Arguments

**x** an object of class `"lm"` for which the response variable is all positive numbers, or els numeric vector of positive numbers. When `x` is an object of class `"lm"` , the object have been created with a call to the function `lm` that includes the `data` argument. `x` is a numeric vector of positive observations, missing ( `NA` ), undefined ( `NaN` ), anc

**lambda** numeric vector of finite values indicating what powers to use for the Box-Cox transformation. When `optimize=FALSE` , the default value is `lambda=seq(-2, 2, by=0` When `optimize=TRUE` , `lambda` must be a vector with two values indicating the rang which the optimization will occur and the range of these two values must include 1. In case, the default value is `lambda=c(-2, 2)` .

**optimize** logical scalar indicating whether to simply evalute the objective function at the given v of `lambda` ( `optimize=FALSE` ; the default), or to compute the optimal power transformation within the bounds specified by `lambda` ( `optimize=TRUE` ).

**objective.name** character string indicating what objective to use. The possible values are `"PPCC"` (probability plot correlation coefficient; the default), `"Shapiro-Wilk"` (the Shapiro-Wil goodness-of-fit statistic), and `"Log-Likelihood"` (the log-likelihood function).

**eps** finite, positive numeric scalar. When the absolute value of `lambda` is less than `eps` , lambda is assumed to be 0 for the Box-Cox transformation. The default value is `eps=.Machine$double.eps` .

**include.x** logical scalar indicating whether to include the finite, non-missing values of the argum `x` with the returned object. The default value is `include.x=TRUE` .

**...** optional arguments for possible future methods. Currently not used.

## Details

Two common assumptions for several standard parametric hypothesis tests are:

1. The observations all come from a normal distribution.
2. The observations all come from distributions with the same variance.

For example, the standard one-sample t-test assumes all the observations come from the same norr distribution, and the standard two-sample t-test assumes that all the observations come from a norr distribution with the same variance, although the mean may differ between the two groups.

When the original data do not satisfy the above assumptions, data transformations are often used tc attempt to satisfy these assumptions. The rest of this section is divided into two parts: one that discu Box-Cox transformations in the context of the original observations, and one that discusses Box-Cox transformations in the context of linear models.

**Box-Cox Transformations Based on the Original Observations** Box and Cox (1964) presented a

| | | | |
|---|---|---|---|
| $Y$ | = | $\frac{X^\lambda - 1}{\lambda}$ | $\lambda \neq 0$ |

where $Y$ is assumed to come from a normal distribution. This transformation is continuous in $\lambda$. Not this transformation also preserves ordering. See the help file for `boxcoxTransform` for more informa on data transformations.

Let $\underline{x} = x_1, x_2, \ldots, x_n$ denote a random sample of $n$ observations from some distribution and ass that there exists some value of $\lambda$ such that the transformed observations

| | | | |
|---|---|---|---|
| $y_i$ | = | $\frac{x_i^\lambda - 1}{\lambda}$ | $\lambda \neq 0$ |

($i = 1, 2, \ldots, n$) form a random sample from a normal distribution.

Box and Cox (1964) proposed choosing the appropriate value of $\lambda$ based on maximizing the likelihoc function. Alternatively, an appropriate value of $\lambda$ can be chosen based on another objective, such as maximizing the probability plot correlation coefficient or the Shapiro-Wilk goodness-of-fit statistic.

In the case when `optimize=TRUE` , the function `boxcox` calls the R function `nlminb` to minimize th negative value of the objective (i.e., maximize the objective) over the range of possible values of $\lambda$ spe in the argument `lambda` . The starting value for the optimization is always $\lambda = 1$ (i.e., no transforma

The rest of this sub-section explains how the objective is computed for the various options for `objective.name` .

*Objective Based on Probability Plot Correlation Coefficient* ( `objective.name="PPCC"` ) When `objective.name="PPCC"` , the objective is computed as the value of the normal probability plot correl coefficient based on the transformed data (see the description of the Probability Plot Correlation Coefficient (PPCC) goodness-of-fit test in the help file for `gofTest` ). That is, the objective is the corre coefficient for the normal quantile-quantile plot for the transformed data. Large values of the PPCC t indicate a good fit to a normal distribution.

*Objective Based on Shapiro-Wilk Goodness-of-Fit Statistic* ( `objective.name="Shapiro-Wilk"` ) When `objective.name="Shapiro-Wilk"` , the objective is computed as the value of the Shapiro-Wilk goodne fit statistic based on the transformed data (see the description of the Shapiro-Wilk test in the help file `gofTest` ). Large values of the Shapiro-Wilk statistic tend to indicate a good fit to a normal distributic

Assuming the transformed observations in Equation (2) above come from a normal distribution with $\mu$ and standard deviation $\sigma$, we can use the change of variable formula to write the log-likelihood fur as:

$$log[L(\lambda, \mu, \sigma)] = \frac{-n}{2} log(2\pi) - \frac{n}{2} log(\sigma^2) - \frac{1}{2\sigma^2} \sum_{i=1}^{n} (y_i - \mu)^2 + (\lambda - 1) \sum_{i=1}^{n} log(x_i)$$

where $y_i$ is defined in Equation (2) above (Box and Cox, 1964). For a fixed value of $\lambda$, the log-likelihoc function is maximized by replacing $\mu$ and $\sigma$ with their maximum likelihood estimators:

$$\hat{\mu} = \frac{1}{n} \sum_{i=1}^{n} y_i \qquad (4)$$

$$\hat{\sigma} = [\frac{1}{n} \sum_{i=1}^{n} (y_i - \bar{y})^2]^{1/2} \qquad (5)$$

Thus, when `optimize=TRUE` , Equation (3) is maximized by iteratively solving for $\lambda$ using the values fo and $\sigma$ given in Equations (4) and (5). When `optimize=FALSE` , the value of the objective is computed I using Equation (3), using the values of $\lambda$ specified in the argument `lambda` , and using the values for $\sigma$ given in Equations (4) and (5).

**Box-Cox Transformation for Linear Models** In the case of a standard linear regression model wit observations and $p$ predictors:

$$Y_i = \beta_0 + \beta_1 X_{i1} + \ldots + \beta_p X_{ip} + \epsilon_i, \ i = 1, 2, \ldots, n \qquad (6)$$

the standard assumptions are:

1. The error terms $\epsilon_i$ come from a normal distribution with mean 0.
2. The variance is the same for all of the error terms and does not depend on the predictor varial

Assuming $Y$ is a random variable from some distribution that may depend on the predictor variables $Y$ takes on only positive values, the Box-Cox family of power transformations is defined as:

| | | | |
|---|---|---|---|
| $Y^*$ | = | $\frac{Y^\lambda - 1}{\lambda}$ | $\lambda \neq 0$ |

where $Y^*$ becomes the new response variable and the errors are now assumed to come from a nor distribution with a mean of 0 and a constant variance.

## Value

When `x` is an object of class `"lm"` , `boxcox` returns a list of class `"boxcoxLm"` containing the res See the help file for `boxcoxLm.object` for details.

When `x` is simply a numeric vector of positive numbers, `boxcox` returns a list of class `"boxcox"` containing the results. See the help file for `boxcox.object` for details.

## Note

Data transformations are often used to induce normality, homoscedasticity, and/or linearity, commo assumptions of parametric statistical tests and estimation procedures. Transformations are not "tric used by the data analyst to hide what is going on, but rather useful tools for understanding and deal with data (Berthouex and Brown, 2002, p.61). Hoaglin (1988) discusses "hidden" transformations tha used everyday, such as the pH scale for measuring acidity. Johnson and Wichern (2007, p.192) note t "Transformations are nothing more than a reexpression of the data in different units."

In the case of a linear model, there are at least two approaches to improving a model fit: transform t and/or $X$ variable(s), and/or use more predictor variables. Often in environmental data analysis, we assume the observations come from a lognormal distribution and automatically take logarithms of t data. For a simple linear regression (i.e., one predictor variable), if regression diagnostic plots indicat a straight line fit is not adequate, but that the variance of the errors appears to be fairly constant, yo only need to transform the predictor variable $X$ or perhaps use a quadratic or cubic model in $X$. Or other hand, if the diagnostic plots indicate that the constant variance and/or normality assumptions suspect, you probably need to consider transforming the response variable $Y$. Data transformations linear regression models are discussed in Draper and Smith (1998, Chapter 13) and Helsel and Hirsc (1992, pp. 228-229).

One problem with data transformations is that translating results on the transformed scale back to t original scale is not always straightforward. Estimating quantities such as means, variances, and confi limits in the transformed scale and then transforming them back to the original scale usually leads tc biased and inconsistent estimates (Gilbert, 1987, p.149; van Belle et al., 2004, p.400). For example, exponentiating the confidence limits for a mean based on log-transformed data does not yield a confidence interval for the mean on the original scale. Instead, this yields a confidence interval for th median (see the help file for `elnormAlt` ). It should be noted, however, that quantiles (percentiles) a rank-based procedures are invariant to monotonic transformations (Helsel and Hirsch, 1992, p.12).

Finally, there is no guarantee that a Box-Cox tranformation based on the "optimal" value of $\lambda$ will pro an adequate transformation to allow the assumption of approximate normality and constant varianc set of transformed data should be inspected relative to the assumptions you want to make about it (Johnson and Wichern, 2007, p.194).

## References

Berthouex, P.M., and L.C. Brown. (2002). *Statistics for Environmental Engineers, Second Edition*. Lewis Publishers, Boca Raton, FL.

Box, G.E.P., and D.R. Cox. (1964). An Analysis of Transformations (with Discussion). *Journal of the Roya Statistical Society, Series B* **26**(2), 211--252.

Draper, N., and H. Smith. (1998). *Applied Regression Analysis*. Third Edition. John Wiley and Sons, New Y pp.47-53.

Gilbert, R.O. (1987). *Statistical Methods for Environmental Pollution Monitoring*. Van Nostrand Reinhold,

Helsel, D.R., and R.M. Hirsch. (1992). *Statistical Methods in Water Resources Research*. Elsevier, New York

Hinkley, D.V., and G. Runger. (1984). The Analysis of Transformed Data (with Discussion). *Journal of the American Statistical Association* **79**, 302--320.

Hoaglin, D.C., F.M. Mosteller, and J.W. Tukey, eds. (1983). *Understanding Robust and Exploratory Data A* John Wiley and Sons, New York, Chapter 4.

Hoaglin, D.C. (1988). Transformations in Everyday Experience. *Chance* **1**, 40--45.

Johnson, N. L., S. Kotz, and A.W. Kemp. (1992). *Univariate Discrete Distributions, Second Edition*. John Wi and Sons, New York, p.163.

Johnson, R.A., and D.W. Wichern. (2007). *Applied Multivariate Statistical Analysis, Sixth Edition*. Pearson Prentice Hall, Upper Saddle River, NJ, pp.192--195.

Shumway, R.H., A.S. Azari, and P. Johnson. (1989). Estimating Mean Concentrations Under Transform. for Environmental Data With Detection Limits. *Technometrics* **31**(3), 347--356.

Stoline, M.R. (1991). An Examination of the Lognormal and Box and Cox Family of Transformations in Environmental Data. *Environmetrics* **2**(1), 85--106.

van Belle, G., L.D. Fisher, Heagerty, P.J., and Lumley, T. (2004). *Biostatistics: A Methodology for the Health Sciences, 2nd Edition*. John Wiley & Sons, New York.

Zar, J.H. (2010). *Biostatistical Analysis*. Fifth Edition. Prentice-Hall, Upper Saddle River, NJ, Chapter 13.

## See Also

`boxcox.object` , `plot.boxcox` , `print.boxcox` , `boxcoxLm.object` , `plot.boxcoxLm` , `print.boxcoxL` `boxcoxTransform` , Data Transformations, Goodness-of-Fit Tests.

```
# NOT RUN {
  # Generate 30 observations from a lognormal distribution with
  # mean=10 and cv=2.  Look at some values of various objectives
  # for various transformations.  Note that for both the PPCC and
  # the Log-Likelihood objective, the optimal value of lambda is
  # about 0, indicating that a log transformation is appropriate.
  # (Note: the call to set.seed simply allows you to reproduce this example.)

  set.seed(250)
  x <- rlnormAlt(30, mean = 10, cv = 2)

  dev.new()
  hist(x, col = "cyan")

  # Using the PPCC objective:
  #--------------------------

  boxcox(x)
  #Results of Box-Cox Transformation
  #---------------------------------
  #
  #Objective Name:                  PPCC
  #
  #Data:                            x
  #
  #Sample Size:                     30
  #
  # lambda      PPCC
  #   -2.0 0.5423739
  #   -1.5 0.6402782
  #   -1.0 0.7818160
  #   -0.5 0.9272219
  #    0.0 0.9921702
  #    0.5 0.9581178
  #    1.0 0.8749611
  #    1.5 0.7827009
  #    2.0 0.7004547

  boxcox(x, optimize = TRUE)
  #Results of Box-Cox Transformation
  #---------------------------------
  #
  #Objective Name:                  PPCC
  #
  #Data:                            x
  #
  #Sample Size:                     30
  #
```

```
#                                 upper =  2
#
#Optimal Value:                   lambda = 0.04530789
#
#Value of Objective:              PPCC = 0.9925919


# Using the Log-Likelihodd objective
#------------------------------------

boxcox(x, objective.name = "Log-Likelihood")
#Results of Box-Cox Transformation
#----------------------------------
#
#Objective Name:                  Log-Likelihood
#
#Data:                            x
#
#Sample Size:                     30
#
# lambda Log-Likelihood
#   -2.0     -154.94255
#   -1.5     -128.59988
#   -1.0     -106.23882
#   -0.5      -90.84800
#    0.0      -85.10204
#    0.5      -88.69825
#    1.0      -99.42630
#    1.5     -115.23701
#    2.0     -134.54125

boxcox(x, objective.name = "Log-Likelihood", optimize = TRUE)
#Results of Box-Cox Transformation
#----------------------------------
#
#Objective Name:                  Log-Likelihood
#
#Data:                            x
#
#Sample Size:                     30
#
#Bounds for Optimization:         lower = -2
#                                 upper =  2
#
#Optimal Value:                   lambda = 0.0405156
#
#Value of Objective:              Log-Likelihood = -85.07123

#----------
```

```
# Plot the results based on the PPCC objective
#---------------------------------------------
boxcox.list <- boxcox(x)
dev.new()
plot(boxcox.list)

#Look at QQ-Plots for the candidate values of lambda
#---------------------------------------------------
plot(boxcox.list, plot.type = "Q-Q Plots", same.window = FALSE)

#==========

# The data frame Environmental.df contains daily measurements of
# ozone concentration, wind speed, temperature, and solar radiation
# in New York City for 153 consecutive days between May 1 and
# September 30, 1973.  In this example, we'll plot ozone vs.
# temperature and look at the Q-Q plot of the residuals.  Then
# we'll look at possible Box-Cox transformations.  The "optimal" one
# based on the PPCC looks close to a log-transformation
# (i.e., lambda=0).  The power that produces the largest PPCC is
# about 0.2, so a cube root (lambda=1/3) transformation might work too.

head(Environmental.df)
#           ozone radiation temperature wind
#05/01/1973    41       190          67  7.4
#05/02/1973    36       118          72  8.0
#05/03/1973    12       149          74 12.6
#05/04/1973    18       313          62 11.5
#05/05/1973    NA        NA          56 14.3
#05/06/1973    28        NA          66 14.9

tail(Environmental.df)
#           ozone radiation temperature wind
#09/25/1973    14        20          63 16.6
#09/26/1973    30       193          70  6.9
#09/27/1973    NA       145          77 13.2
#09/28/1973    14       191          75 14.3
#09/29/1973    18       131          76  8.0
#09/30/1973    20       223          68 11.5


# Fit the model with the raw Ozone data
#------------------------------------
ozone.fit <- lm(ozone ~ temperature, data = Environmental.df)

# Plot Ozone vs. Temperature, with fitted line
#---------------------------------------------
dev.new()
with(Environmental.df,
```

```
    ylab = "Ozone (ppb)", main = "Ozone vs. Temperature"))
abline(ozone.fit)

# Look at the Q-Q Plot for the residuals
#---------------------------------------
dev.new()
qqPlot(ozone.fit$residuals, add.line = TRUE)

# Look at Box-Cox transformations of Ozone
#-----------------------------------------
boxcox.list <- boxcox(ozone.fit)
boxcox.list
#Results of Box-Cox Transformation
#---------------------------------
#
#Objective Name:                  PPCC
#
#Linear Model:                    ozone.fit
#
#Sample Size:                     116
#
# lambda      PPCC
#   -2.0 0.4286781
#   -1.5 0.4673544
#   -1.0 0.5896132
#   -0.5 0.8301458
#    0.0 0.9871519
#    0.5 0.9819825
#    1.0 0.9408694
#    1.5 0.8840770
#    2.0 0.8213675

# Plot PPCC vs. lambda based on Q-Q plots of residuals
#-----------------------------------------------------
dev.new()
plot(boxcox.list)

# Look at Q-Q plots of residuals for the various transformation
#--------------------------------------------------------------
plot(boxcox.list, plot.type = "Q-Q Plots", same.window = FALSE)

# Compute the "optimal" transformation
#-------------------------------------
boxcox(ozone.fit, optimize = TRUE)
#Results of Box-Cox Transformation
#---------------------------------
#
#Objective Name:                  PPCC
#
```

```
  #
  #Sample Size:                     116
  #
  #Bounds for Optimization:         lower = -2
  #                                 upper =  2
  #
  #Optimal Value:                   lambda = 0.2004305
  #
  #Value of Objective:              PPCC = 0.9940222

  #==========

  # Clean up
  #---------
  rm(x, boxcox.list, ozone.fit)
  graphics.off()
# }
```

*Documentation reproduced from package EnvStats, version 2.3.1, License: GPL (>= 3)*

## Community examples

Looks like there are no examples yet.

## Post a new example:

B I H | “ ≡ ≡ | 🔗 🖼 ▦ — | 👁 ◫ ✕ | ? | </>

## New example
Use markdown to format your example

R code blocks are runnable and interactive:
```r
a <- 2
print(a)
```

You can also display normal code blocks
```
```