Package ‘extremevalues’

January 20, 2016

Description  Detect outliers in one-dimensional data.
Version  2.3.2
Date  2016-01-05
Title  Univariate Outlier Detection
Author  Mark van der Loo <mark.vanderloo@gmail.com>
Maintainer  Mark van der Loo <mark.vanderloo@gmail.com>
Depends  R (>= 2.8.0)
Imports  gWidgets, gWidgets.tcltk, utils, stats, graphics
License  GPL-2

URL  https://www.github.com/markvanderloo/extremevalues
Repository  CRAN
NeedsCompilation  no
Date/Publication  2016-01-20 10:29:56

R topics documented:

  evGui                          .................................................. 2
  extremevalues                  ................................................. 3
  getOutliers                    ................................................. 3
  invErf                         ................................................... 5
  outlierPlot                    ................................................. 6
  pareto                         ................................................... 7

Index  9
evGui

GUI to explore options and results of the "extremevalues" package

Description

Opens a Graphical User Interface and plots results. Options of the extremevalue package functions can be set and results are updated instantly. Includes a code generator button.

Usage

evGui(y)

Arguments

y A vector of type numeric

Note

The GUI is programmed in a very quick and pretty dirty way, but it works fine. It will be replaced by a gtk-version in the future.

Author(s)

Mark van der Loo

References

www.markvanderloo.eu

See Also

gGetOutliers

Examples

## Not run:
y <- rnorm(100)
evGui(y)

## End(Not run)
**Description**

This package offers outlier detection and plot functions for univariate data.

The package is the implementation of the outlier detection methods introduced in the reference below. Briefly, the methods work as follows. Using a subset of the data, the parameters for a model distribution are estimated using regression of the sorted data on their QQ-plot positions.

A value in the data is an outlier when it is unlikely to be drawn from the estimated distribution. There are two methods to determine the "unlikelyness". The first, called "Method I", determines the value above which less than \( \rho \) observations are expected, given the total number of observations in the data. Here \( \rho \) is a parameter which should have a value of 1 or less. The second notion of unlikelyness uses the fit residuals. Extremely large or small values are outliers when their residuals are above or below a confidence limit \( \alpha \), to be determined by the user.

**References**


**See Also**

getOutliers, outlierPlot

**Usage**

getOutliers(y, method="I", ...)  
getOutliersI(y, rho=c(1,1), FLim=c(0.1,0.9), distribution="normal")  
getOutliersII(y, alpha=c(0.05, 0.05), FLim=c(0.1, 0.9),  
                   distribution="normal", returnResiduals=TRUE)
getOutliers

Arguments

\textit{y} \hspace{10mm} \text{Vector of one-dimensional nonnegative data}

\textit{method} \hspace{5mm} "I" or "II"

... \hspace{10mm} \text{Optional arguments to be passed to getOutliersI or getOutliersII}

\textit{distribution} \hspace{5mm} \text{Model distribution used to estimate the limit. Choose from "lognormal", "exponential", "pareto", "weibull" or "normal" (default).}

\textit{FLim} \hspace{5mm} c(Fmin,Fmax) quantile limits indicating which data should be used to fit the model distribution. Must obey 0 < Fmin < Fmax < 1.

\textit{rho} \hspace{5mm} (Method I) A value \(y_i\) is an outlier if it is below (above) the limit where less then \(\rho[2]\) (\(\rho[1]\)) observations are expected. Must be >0.

\textit{alpha} \hspace{5mm} (Method II) A value \(y_i\) is an outlier if it has a residual below (above) the alpha[1] (alpha[2]) confidence limit for the residues. Must be between 0 and 1.

\textit{returnResiduals} \hspace{5mm} (Method II) Whether or not to return a vector of residuals from the fit

Details

Both methods use the subset of \(y\)-values between the Fmin and Fmax quantiles to fit a model cumulative density distribution. \textbf{Method I} detects outliers by checking which are below (above) the limit where according to the model distribution less then \(\rho[1]\) (\(\rho[2]\)) observations are expected (given length(y) observations). \textbf{Method II} detects outliers by finding the observations (not used in the fit) who’s fit residuals are below (above) the estimated confidence limit alpha[1] (alpha[2]) while all lower (higher) observations are outliers too.

Value

\textit{nOut} \hspace{10mm} \text{Number of left and right outliers.}

\textit{iLeft} \hspace{10mm} \text{Index vector indicating left outliers in } y

\textit{iRight} \hspace{10mm} \text{Index vector indicating right outliers in } y

\textit{limit} \hspace{5mm} \text{For Method I: } y\text{-values below (above) limit[1] (limit[2]) are outliers. For Method II: elements with residuals below (above) limit[1] (limit[2]) are outliers if all smaller (larger) elements are outliers as well.}

\textit{method} \hspace{5mm} \text{The used method: "method I" or "method II"}

\textit{distribution} \hspace{5mm} \text{The used model distribution}

\textit{Fmin} \hspace{5mm} FLim[1]

\textit{Fmax} \hspace{5mm} FLim[2]

\textit{yMin} \hspace{5mm} \text{Smallest y-value used in fit}

\textit{yMax} \hspace{5mm} \text{Largest y-value used in fit}

\textit{Nfit} \hspace{5mm} \text{Number of values used in the fit}

\textit{rho} \hspace{5mm} \text{Method I, the input rho-values for left and right outliers}

\textit{alphaConf} \hspace{5mm} \text{Method II, the input confidence levels for left and right outliers}
R2  R-squared value for the fit. Note that this is the ordinary least squares value, defined by $R^2 = 1 - SS_{err}/SS_y$. Where $SS_{err}$ is the squared sum of residuals. For the lognormal, Pareto and Weibull models, the $y$-variable is transformed before fitting. Since predicted values are transformed back before calculating $SS_{err}$, this $R^2$ can be negative.

lambda (exponential distribution) Estimated location (and spread) parameter for $f(y) = \lambda \exp(-\lambda y)$

mu (lognormal distribution) Estimated $E(\ln(y))$ for lognormal distribution

sigma (lognormal distribution) Estimated $Var(\ln(y))$ for lognormal distribution

ym (pareto distribution) Estimated location parameter (mode) for pareto distribution

alpha (pareto distribution) Estimated spread parameter for pareto distribution

k (weibull distribution) estimated shape parameter $k$ for weibull distribution

lambda (weibull distribution) estimated scale parameter $\lambda$ for weibull distribution

mu (normal distribution) Estimated $E(y)$ for normal distribution

sigma (normal distribution) Estimated $Var(y)$ for normal distribution

Author(s)
Mark van der Loo, see www.markvanderloo.eu

References

The file <your R directory>/R-<version>/library/extremevalues/extremevalues.pdf contains a worked example. It can also be downloaded from my website.

Examples

```r
y <- rlnorm(100)
y <- c(0.1*min(y),y,10*max(y))
K <- getOutliers(y,method="I",distribution="lognormal")
L <- getOutliers(y,method="II",distribution="lognormal")
par(mfrow=c(1,2))
outlierPlot(y,K,mode="qq")
outlierPlot(y,L,mode="residual")
```

invErf  Inverse error function

Description
Inverse error function
Usage

\texttt{invErf(x)}

Arguments

\textbf{x} \hspace{1cm} \text{(Vector of) real value(s) in the range (-1,1)}

Value

\text{(vector of) value(s) of the inverse error function}

Author(s)

Mark van der Loo, www.markvanderloo.eu

Examples

\begin{verbatim}
x <- seq(-0.99, 0.99, 0.01); plot(x, invErf(x), 'l');
\end{verbatim}

\section*{Description}

This is a wrapper for two plot functions which can be used to analyse the results of outlier detection with the extremevalues package.

Usage

\texttt{outlierPlot(y, L, mode="qq", \ldots)}
\texttt{qqFitPlot(y, L, title=NA, xlab=NA, ylab=NA, fat=FALSE)}
\texttt{plotMethodII(y, L, title=NA, xlab=NA, ylab=NA, fat=FALSE)}

Arguments

\begin{itemize}
  \item \texttt{y} \hspace{1cm} \text{A vector of values}
  \item \texttt{L} \hspace{1cm} \text{The result of \texttt{L <- getOutliers(y,...)}}
  \item \texttt{mode} \hspace{1cm} \text{Plot type. "qq" for Quantile-quantile plot with indicated outliers, "residual" for plot of fit residuals with indicated outliers (Method II only)}
  \item \texttt{\ldots} \hspace{1cm} \text{Optional arguments, to be transferred to \texttt{qqFitPlot} or \texttt{plotMethodII} (see below)}
  \item \texttt{title} \hspace{1cm} \text{A custom title (must be a string)}
  \item \texttt{xlab} \hspace{1cm} \text{A custom label for the x-axis (must be a string)}
  \item \texttt{ylab} \hspace{1cm} \text{A custom label for the y-axis (must be a string)}
  \item \texttt{fat} \hspace{1cm} \text{If TRUE, axis, fonts, labels, points and lines are thicker for export and publication}
\end{itemize}
Pareto distribution

Description

Pareto density distribution, quantile function and random generator.

Usage

dpareto(x, xm=1, alpha=1)
qpareto(p, xm=1, alpha=1)
rrpareto(n, xm=1, alpha=1)

Arguments

xm
alpha
x
p
n
location parameter (mode of distribution)
spread parameter
Vector of realizations
Vector of probabilities
number of samples to draw
Value

- dpareto: Probability density
- qpareto: Quantile at probability p (inverse cdf)
- rpareto: Random value

Author(s)

Mark van der Loo www.markvanderloo.eu

Examples

```r
q <- qpareto(0.5);
```
Index

*Topic outliers
   evGui, 2

dpareto (pareto), 7

evGui, 2
extremevalues, 3
extremevalues-package (extremevalues), 3

getOutliers, 2, 3, 3
getOutliersI (getOutliers), 3
getOutliersII (getOutliers), 3

invErf, 5

outlierPlot, 3, 6

pareto, 7
plotMethodII (outlierPlot), 6

qpareto (pareto), 7
qqFitPlot (outlierPlot), 6

rpareto (pareto), 7