extremevalues
A package for outlier detection

Version 2.1

M.P.J. van der Loo
mark.vanderloo@gmail.com, www.markvanderloo.eu

June 4, 2010

Contents

1 Introduction

This package provides the implementation of the outlier detection method as described in van der Loo (2010). Briefly, the method works as follows: given a univariate dataset $\mathbf{y}$, with values $y_1 \leq y_2 \leq \ldots \leq y_N$ which is assumed to be drawn from a model distribution with cdf $F(Y|\boldsymbol{\theta})$. The package determines which (if any) of the extreme observations $y_1, y_2, \ldots$ and/or $y_N, y_{N-1}, \ldots$ are outliers in two steps:

1. Estimate the distribution's parameters $\hat{\boldsymbol{\theta}}$ by fitting a model cdf F to the QQ plot positions of the bulk of the observed data. The bulk is all data between a minimum and maximum quantile (default: 0.1 and 0.9) which can be set by the user.

2. Determine whether the extreme observations are unlikely to be drawn from $F(Y|\hat{\boldsymbol{\theta}})$. There are two ways of doing this:

Method I: Calculate the limit $\ell^{\pm}_{\rho}$ above (+) or below (−) less than ρ observations are expected, given $F(Y|\hat{\boldsymbol{\theta}})$ and the number of observations N. The observations $y_1, y_2, \ldots < \ell^{-}_{\rho}$ are left outliers and the observations $y_N, y_{N-1} > \ell^{+}_{\rho}$ are right outliers. The value of ρ (default: 1.0) can be set by the user for left and right outliers separately.

Method II: Estimate the $1-2\alpha$ confidence interval for the residuals resulting from the fit which determines the model distribution $F(Y|\hat{\boldsymbol{\theta}})$. Label the upper (+) and lower (−) limits of this interval with $\ell^{\pm}_{\alpha}$. Extreme values which were not used in the fit are left (right) outliers when their residuals are larger negative (larger positive) than ℓ^{-}_{α} (ℓ^{+}_{α}). The values $\ell^{\pm}_{\alpha}$ are estimated assuming normally distributed residuals and the value of α (default: 0.05) may be set by the user for left and right outliers separately.

The main purpose of the extremevalues package is to provide functions which can detect outliers using the methods described above. Additionally, plotfunctions are provided for graphical analysis of the result. The package currently supports five model distributions:

- Lognormal distribution
- Exponential distribution
- Pareto distribution
- Weibull distribution
- Normal distribution

In this document we work through an example to familiarize the reader with the use of the package. I refer to the R help files for a complete description of the functions and to the reference mentioned above for a better explanation of the methodology.

2 A quick example

Generate some lognormally distributed data:

```
> y <- rlnorm(200)
```

(You can probably reconstruct the exact results using `set.seed(123456789)` first, as I did. There might still be differences since the output of a pseudorandom generator can depend on the architecture of the machine its running on.) Let's add a left and a right outlier:

```
> y <- c(y,0.1*min(y),10*max(y))
```

And try to detect them using (the default) Method I.

```
> L <- getOutliers(y, distribution="lognormal")
```

The number of left and right detected outliers are

```
> L$nOut
 Left Right
    1     1
```

The position of left and right outliers are stored as indices:

```
> L$iLeft
[1] 201

> L$iRight
[1] 202
```

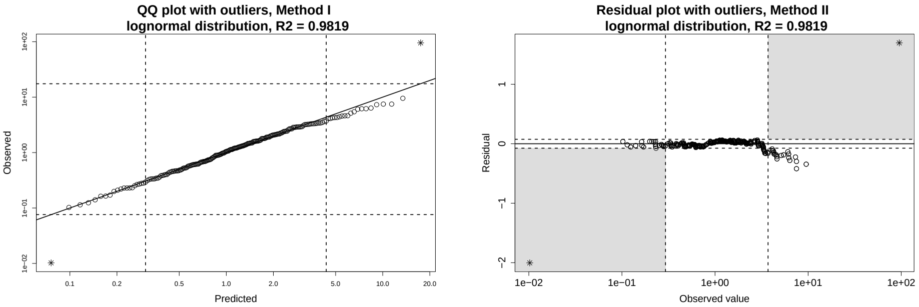
Now, let's use Method II:

```
M <- getOutliers(y,distribution="lognormal", method="II")
```

To see what we have done, we can plot the results with:

```
> par(mfrow=c(1,2))
> outlierPlot(y,L)
> outlierPlot(y,M,mode="residual")
```

The resulting picture looks something like this:



The left panel shows a QQ plot for y based on the lognormal distribution. The values between the vertical dotted lines are used in to fit the distribution. Horizontal dotted lines indicate values $\ell^{\pm}_{\rho=1.0}$ above or below which observations are identified as outliers. The outliers are indicated with a $\ast$.

The right panel shows a residual plot, based on the fit of the lognormal cdf to the QQ plot positions. Points between vertical lines correspond to data used in the fit. The horizontal lines indicate the outlier limits $\ell^{\pm}_{\alpha=0.05}$. Outliers can only occur in the grey areas and are indicated with a $\ast$.

Note. The plots resulting on your screen after copying the code above will have different font sizes and line widths from the figure above. You can get the larger fonts and linewidhts by using the `fat=TRUE` option in the plot commands. (but don't use `par(mfrow=c(2,1))` in that case).

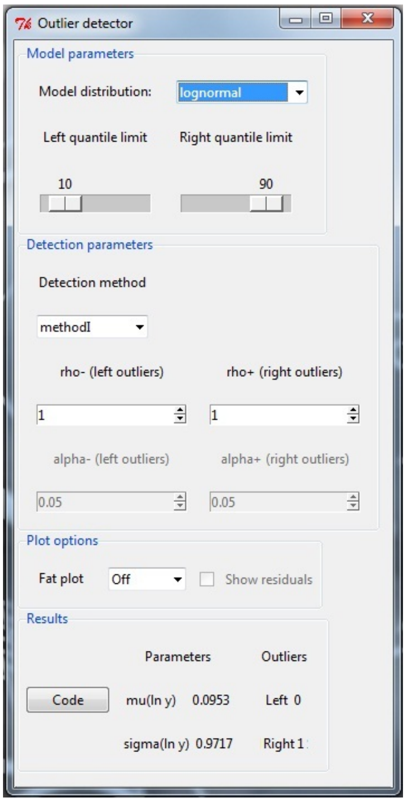
3 evGui: browse outliers

Version 2.1 comes with a GUI function which allows you to explore the detection and plottting options interactively. The GUI is written in gWidgets and works with the tcl/tk support that comes standard with R. If evGui() does not work on your system, you probably have to start the R installer again and make sure you check the tcl/tk option. Since the GUI is written with gWidgets functionality only, you are free to choose another toolkit, like Gtk2.

Starting the GUI is easy, just call

```
> y <- rlnorm(200)
> evGui(y)
```

to start the interface. An outlier-plot will appear, along with a control panel. The panel contains widgets to control every setting of the outlier detection function getOutliers, the plot function, and a code generator. In the top frame one can choose the type of distribution and the left and right quantile limits, which determine the observations used in the fit. The second frame contains widgets to set detection method and sensitivity. Parameters $\rho_{\pm}$ and $\alpha_{\pm}$ can only be set when their corresponding method is chosen. The third frame contains plot options, and the bottom frame contains a code generator button and shows some results.



Model distribution: lognormal

Left quantile limit 10

Right quantile limit 90

Detection method methodI

rho- (left outliers) 1

rho+ (right outliers) 1

alpha- (left outliers) 0.05

alpha+ (right outliers) 0.05

Fat plot Off

□ Show residuals

Parameters		Outliers
mu(ln y)	0.0953	Left 0
sigma(ln y)	0.9717	Right 1

The plot and detection model are updated instantly when new parameter settings are chosen.

4 Version history and list of changes

Version history	
Version	Uploaded
version 1.0	03.12.2009
version 2.0	10.03.2010
version 2.1	04.06.2010

4.1 Important changes in version 2.1

- Killed a bug in lognormal distribution/method II
- Added function evGui
- Completed citation information

4.2 Important changes in version 2.0

- Added automatic left outlier detection.
- Solved a bug in fit of exponential distribution
- Changed standard value of ρ (Method I)
- Added Method II
- Renaming main functions
- Revised and extended plotting capabilities
- Minor revisions of helpfiles
- Changed equation for calculation of plot positions (after reading paper of Makkonen (2008)).

5 Function listing

User functions	
evGui	Interactive outlier detection
getOutliers	Detect outliers, wrapper.
getOutliersI	idem, using Method I
getOutliersII	idem, using Method II
outlierPlot	Plot detection results, wrapper
qqFitPlot	plot results, Method I
plotMethodII	plot results, Method II
rpareto	Draw from pareto distribution
dpareto	Pareto density function
qpareto	Pareto quantile function
invErf	Inverse error function
Internal functions	
fitPareto	Fit data to cumulative pareto distribution
fitLognormal	Fit data to cumulative lognormal distribution
fitExponential	Fit data to cumulative exponential distribution
fitWeibull	Fit data to cumulative weibull distribution
fitNormal	Fit data to cumulative normal distribution
(get/qq)ParetoLimit	Determine outlier limit assuming pareto distribution
(get/qq)LognormalLimit	Determine outlier limit assuming lognormal distribution
(get/qq)ExponentialLimit	Determine outlier limit assuming exponential distribution
(get/qq)WeibullLimit	Determine outlier limit assuming weibull distribution
(get/qq)NormalLimit	Determine outlier limit assuming normal distribution

References

L. Makkonen. Bringing closure to the plotting position controversy. *Communications in Statistics*, 37:460–467, 2008.

M. P. J. van der Loo. Distribution based outlier detection for univariate data. Technical Report 10003, Statistics Netherlands, The Hague, 2010.