Package: washeR (via r-universe)

November 3, 2024

Type Package

Title Time Series Outlier Detection

Version 0.1.3

Author Andrea Venturini

Maintainer Andrea Venturini <andrea.venturini@bancaditalia.it>

Description Time series outlier detection with non parametric test.
This is a new outlier detection methodology (washer): efficient
for time saving elaboration and implementation procedures,
adaptable for general assumptions and for needing very short
time series, reliable and effective as involving robust non
parametric test. You can find two approaches: single time
series (a vector) and grouped time series (a data frame). For
other informations: Andrea Venturini (2011) Statistica -
Universita di Bologna, Vol.71, pp.329-344. For an informal
explanation look at R-bloggers on web.

License GPL (>= 2)

Encoding UTF-8

LazyData true

Imports gplots,grDevices,graphics,stats,utils

RoxygenNote 7.2.1

NeedsCompilation no

Repository CRAN

Date/Publication 2022-09-20 07:40:02 UTC

Contents

`dati` *Data frame of meteorological data*

Description

This sample data would be invented meteorological information detected by weather stations.

Usage

```
dati
```

Format

A data frame with 800 rows and 4 variables:

phen Temperature, Rain

time ordered numbers for time (a number in the format YYYYMMDD [Year Month Day] is possible too)

zone label classification for the group, for example the identification code of a wheather station.

value values

`ts` *Time series*

Description

This is an example of a single time series with increasing trend and some variability.

Usage

```
ts
```

Format

A data frame with 35 rows and 1 variable:

dati pseudo random numers

wash.out *Outlier detection for single or grouped time series*

Description

This function provides anomaly signals (even a graphical visualization) when there is a 'jump' in a single time series, or the 'jump' is too much different respect those ones of grouped similar time series.

Usage

```
wash.out(
  dati,
  graph = FALSE,
  linear_analysis = FALSE,
  val_test_limit = 5,
  save_out = FALSE,
  out_out = "out.csv",
  pdf_out = "out.pdf",
  r_out = 3,
  c_out = 2,
  first_line = 1,
  pace_line = 6
)
```

Arguments

`dati`	data frame (grouped time series: phenomenon+date+group+values) or vector (single time series)
`graph`	logical value for graphical analysis (default=FALSE)
`linear_analysis`	logical value for linear analysis (default=FALSE)
`val_test_limit`	value for outlier detection sensitiveness (default=5 ; max=10)
`save_out`	logical value for saving detected outliers (default=FALSE)
`out_out`	a character file name for saving outliers in csv form, delimited with ";" and using ',' as decimal separator (default out.csv)
`pdf_out`	a character file name for saving graphic analysis in pdf file (default=out.pdf)
`r_out`	rows number of graphs (default=3)
`c_out`	cols number of graphs (default=2)
`first_line`	value for first dotted line in graphic analysis (default=1)
`pace_line`	value for pace in dotted line in graphic analysis (default=6)

Value

Data frame of possible outliers in a triad. Output record: rows/time.2/series/y1/y2/y3/test(AV)/AV/n/median(AV)/mad(AV)/madindex(AV). Where time.2 is the center of the triad y1, y2, y3; test(AV) is the number to compare with 5 to detect outlier; n is the number of observations of the group ....

Examples

```
## we can start with data without outliers but structured with co-movement between groups
data("dati")
## first column for phenomenon
## 2° col for time written in ordered numbers or strings
## 3° col for group classification variable
## 4° col for values
str(dati)
########################################
## a data frame without any outlier
########################################
out=wash.out(dati)
out    ## empity data frame
length(out[,1])  ## no row
## we can add two outliers
####  time=3 temperature value=0
dati[99,4]=  0
## ... and then for "rain" phenomenon!
####  time=3 rain value=37
dati[118,4]=  37
########################################
##   data.frame with 2 fresh outliers
########################################
out=wash.out(dati)
##  all "three terms" time series
## let's take a look at anomalous time series
out
## ... the same but we save results in a file....
## If we don't specify a name, out.csv  is the default
out=wash.out(dati,save_out=TRUE,out_out="tabel_out.csv")
out
## we put the parameter from 5 to 10, using this upper one  to capture
##       only  particularly anomalous outliers
out=wash.out(dati, val_test_limit = 10)
out
## save plots and outliers in a pdf file "out.pdf" as a default
out=wash.out(dati, val_test_limit = 10, graph=TRUE)
out
## we can make the usual analysis for groups but we can also use that one
## reserved for every single time series
## (linear_analysis): two files for saved outliers (out.csv and linout.csv)
##  and for graph display in two pdf files (out.pdf and linout.pdf)
out=wash.out(dati,val_test_limit=5,save_out=TRUE,linear_analysis=TRUE,graph=TRUE)
out
## out return only the linear analysis...
## ... in this case we lose the co-movement information an we run the risk
##     of finding too much variance in a single time series
##     and detecting not too much likely outliers
###########################################################
##  single time series analysis
###########################################################
data(ts)
str(ts)
sts= ts$dati
plot(sts,type="b",pch=20,col="red")
## a time series with a variability and an increasing trend
## sts is a vector and linear analysis is the default one
out=wash.out(sts)
out
## we find no outlier
out=wash.out(sts,val_test_limit=5,linear_analysis=TRUE,graph=TRUE)
out
## no outlier
## We can add an outlier with limited amount
sts[5]=sts[5]*2
plot(sts,type="b",pch=20,col="red")
out=wash.out(sts,val_test_limit=5)
out
## test is over 5 for a bit
out=wash.out(sts,val_test_limit=5,save_out=TRUE,graph=TRUE)
out
data(ts)
sts= ts$dati
sts[5]=sts[5]*3
## we can try a greater value to put an outlier of a certain importance
plot(sts,type="b",pch=20,col="blue")
out=wash.out(sts,val_test_limit=5,save_out=TRUE,graph=TRUE)
out
## washer procedure identify three triads of outliers values
system("rm *.csv *.pdf")
```

Index