Package ‘OutlierDM’

February 19, 2015

Title   Outlier Detection for Multi-replicated High-throughput Data
Date    2014-12-22
Version 1.1.1
Description Detecting outlying values such as genes, peptides or samples for multi-replicated high-throughput high-dimensional data
Depends R (>= 3.1.0)
Imports quantreg, MatrixModels, outliers, pcaPP, methods, graphics
Maintainer Soo-Heang Eo <eo.sooheang@gmail.com>
License GPL-3
Author Soo-Heang Eo [aut, cre],
        HyungJun Cho [aut]
NeedsCompilation no
Repository CRAN
Date/Publication 2014-12-24 01:28:29

R topics documented:

OutlierDM-package .................................................. 2
lcms3 ................................................................. 3
odm ................................................................. 4
odm.control ......................................................... 8
oneplot ............................................................. 9
OutlierDM-class ................................................... 10
plot ................................................................. 11
rgrubbs.test .................................................... 12
toy ................................................................. 12

Index 14
OutlierDM-package

Functions for detecting outlying parameters (peptides) or observations (samples) in multi-replicated high-throughput data such as mass spectrometry experiments

Description
This package provides several outlier detection algorithms for multi-replicated high-throughput data ranged from classical approaches to boxplot approaches based on a MA plot.

Details

| Package: | OutlierDM |
| Type: | Package |
| Version: | 1.1.1 |
| Date: | 2014-12-23 |
| License: | GPL version 3 |
| LazyLoad: | no |

Author(s)
Soo-Heang Eo <eo.sooheang@gmail.com> and HyungJun Cho <hj4cho@korea.ac.kr>
Maintainer: Soo-Heang Eo <eo.sooheang@gmail.com>

References


See Also

odm, odm.control, quantreg

Description

This data set consists of three-replicated LC/MS/MS data obtained from the laboratory of gaseous ion chemistry in department of chemistry, Korea University.

Usage

data(lcms3)

Format

a matrix for LCMS data, rows = peptides, columns = samples

Details

LC/MS/MS experiments were performed on a home-built ultrahigh-pressure dual on-line solid phase extraction/capillary reverse-phase liquid chromatography (DO-SPE/cRPLC) (Min et al., 2007) that was coupled to a Fourier transform ion cyclotron resonance (FT-ICR, LTQ-FT, Thermo, San Jose) mass spectrometer by means of a home built nanoESI interface.

Source


Examples

data(lcms3)
str(lcms3)
pairs(log2(lcms3), pch = 20, cex = .7)
**Description**

This function provides some routines for detecting outlying observations (peptides) for multi-replicated high-throughput data, especially in LC/MS experiments.

**Usage**

```r
odm(x, k = 3,
    quantreg = c("linear", "nonlin", "constant", "nonpar"),
    method = c("proj", "diff", "pair", "grubbs", "dixon",
               "iqr", "siqr", "Zscore"), ...
)
```

**Arguments**

- `x` data vectors or matrices. These can be given as named arguments. If the number of predictors is 2, `x1` describes one n-by-1 vector for data and `x2` describes the other n-by-1 vector for data (n = number of peptides, proteins, or genes).
- `k` non-negative tuning parameter for the outlier detection algorithm. For IQR-based algorithms such as `iqr`, `siqr`, `proj`, `diff`, and `pair`, it works in the formula of \(Q1-k*IQR\) and \(Q3+k*IQR\), where IQR = \(Q3-Q1\). For `Zscore`, it works for the 'k' in |Z| > k. A default value is 3.
- `quantreg` type of quantile regression models used for the outlier detection method. You can use one of the 'constant', 'linear', 'nonlin', and 'nonpar' which mean the constant, linear, non-linear, and non-parametric quantile regression in order. For more details, see the quantreg package.
- `method` type of outlier detection methods. You can select one of the 'Zscore', 'iqr', 'dixon', 'grubbs', 'pair', 'diff', and 'proj' algorithms as follows.
  - **Zscore**: Z-score based criterion (Cho and Eo, 2015)
  - **iqr**: Interquartile range (IQR) criterion (Cho and Eo, 2015)
  - **siqr**: Semi-interquartile range (IQR) criterion (Cho and Eo, 2015)
  - **dixon**: Dixon's test (Dixon, 1950; 1951)
  - **grubbs**: Grubbs test (Grubbs, 1950; 1969)
  - **pair**: Pariwise OutlierD algorithm (Cho et al., 2008; Eo et al., 2012)
  - **proj**: Projection-based OutlierD algorithm (Eo et al., 2012)
  - **diff**: Difference-based OutlierD algorithm (Eo and Cho, 2015)

**Value**

- `call` evaluated function call
- `raw.data` raw dataset used in the model fitting
res: result matrix of the model fitting. It consists of used data set with some transformation and outlying statistic.

x.pair: Object of class "list"

k: threshold parameter for constructing outlier detection methods

outlier: matrix including the status of each outlying peptide and sample

n.outliers: the number of outlying parameters (peptides) to be detected by the model fitting.

quantreg: type of quantile regression used for the model fitting

method: type of outlier detection method used for the model fitting

contrl.param: a list of minor parameters

References


See Also

OutlierDM-package to provide the general information about the OutlierDC package
OutlierDM-class to provide the information about the "OutlierDM" class

Examples

## Not run:

### Not run:

# Outlier Detection for Mass Spectrometry Data
# Section 3. Illustration
# by Hyungjun Cho and Soo-Heang Eo,
# Dept of Statistics, Korea University, Seoul, Korea
#
# ***************************************************************************
#
#
###
#
# Load a package OutlierDM
# If an OutlierDM package is not installed on your system, type
#install.package('OutlierDM', dependency = TRUE)
library(OutlierDM)

###
# Sec 3.1 When the number of replicates is large enough
## Load toy dataset
data(toy)
head(toy)
pairs(log2(toy), pch = 20, cex = .7)

###
# Fit 1. Z-score based criterion
fit1 = odm(x = toy, method = "Zscore", k = 3)
fit1
summary(fit1)
head(input(fit1))
head(output(fit1))
print(outliers(fit1), digits = 3)
plot(fit1)
rect(1, -4, 10, 4, col = heat.colors(20, alpha = 0.3), border = heat.colors(20, alpha = 0.5))
oneplot(object = fit1, i = 4)
title("Outlier Detection by the Z-score criterion")

# Add a peptide name on a dot-plot
#oneplot(fit1, 191,1)
#title("Outlier Detection by the Z-score criterion")

###
# Fit 2. Grubbs test criteria
fit2 = odm(x = toy, method = "grubbs", alpha = 0.01)
fit2
summary(fit2)
head(output(fit2))
print(outliers(fit2), digits = 3)
oneplot(object = fit2, i = 1)
title("Outlier Detection by the Grubbs criterion")

# Add text
#oneplot(fit2, 191,1)
#title("Outlier Detection by the Grubbs criterion")

###
# Fit 3. IQR criteria
fit3 = odm(x = toy, method = "iqr", k = 3)
fit3
summary(fit3)
print(outliers(fit3), digits = 3)
plot(fit3)
rect(1, -4, 10, 40, col = heat.colors(20, alpha = 0.3), border = heat.colors(20, alpha = 0.5))
oneplot(fit3, 1)
title("Outlier Detection by the IQR criterion")
# Add a peptide name on a dot-plot
oneplot(fit3, 1, 1)
#title("Outlier Detection by the IQR criterion")

#######
# Fit 4. SIQR criteria
fit4 = odm(x = toy, method = "siqr", k = 3)
summary(fit4)
print(outliers(fit4), digits = 3)
plot(fit4)
rect(1, -4, 10, 4, col = heat.colors(20, alpha = 0.3), border = heat.colors(20, alpha = 0.5))
oneplot(fit4, 1)
title("Outlier Detection by the SIQR criterion")

########################################################
## Real data example
########################################################
data(lc3)
head(lc3)
pairs(log2(lc3), pch = 20, cex = .7)

#######
# Fit 5. OutlierD
fit5 = odm(lc3[,1:2], method = "pair", k = 3)
summary(fit5)
head(output(fit5))
print(outliers(fit5), digits = 3)
plot(fit5)
title("Outlier Detection by the OutlierD algorithm")

#######
# Fit 6. OutlierDM
fit6 = odm(lc3, method = "proj", k = 3, center = TRUE)
summary(fit6)
print(outliers(fit6), digits = 3)
plot(fit6)
title("Outlier Detection by the OutlierDM algorithm")
oneplot(fit6, 18)
#oneplot(fit6, 18, 1)
title("The dotplot for the 18th samples of the lc3 data")
#### End of the illustration

#######
## Other OutlierDM algorithms
data(lc3)

## Load
## Fit projection approaches
odm.control

Control tuning parameters for "OutlierDM" object

Description

various parameters that control aspects of the "OutlierDM" object

Usage

odm.control(pair.cre = 1, dist.mthd = "median",
             Lower = .25, Upper = .75, trans = "log2",
             centering = TRUE, projection.type = "PCA", lbda = 1,
             nonlin.method = "L-BFGS-B", nonlin.SS = "AsymOff",
             nonlin.Frank = c(2, -8, 0, 1), ncl = 2, alpha = 0.05)

Arguments

pair.cre a scalar parameter to specify the minimum number of pairs, used in "type = pair".
dist.mthd a distance parameter used in "type = diff". You can choose one of "median", "mean" or and so on.
Lower a criterion for lower quantile value used to construct boxplot
Upper a criterion for upper quantile value used to construct boxplot
trans a parameter for a logarithm and exponential transformation. If a log 2 transformation is needed, set "trans = log2". If no transformation is needed, set "trans = FALSE".
centering a logical parameter for the status of centering. If "centering = TRUE", data are centered by its column means.
projection.type a parameter to determine a type of projection methods. Choose one of "naive", "pca", and "robust".
lbda a criterion about lambda used for nonlinear quantile regression.
oneplot

nonlin.method  a parameter to determine a type of methods used for nonlinear quantile regression. choose one of "L-BFGS-B" and "BFGS". Default is "L-BFGS-B".

nonlin.ss  a parameter to determine a type of structure used for nonlinear quantile regression. choose one of "Frank", "Self", "Asym" and "AsymOff". Default is "AsymOff", Asymptotic Regression Model with an Offset.

nonlin.Frank  a structure parameter used for Frank copula model. Gain c(df, delta, mu, sigma) in the Frank copula formula

nc1  A parameter to determine the number of cores used in parallel computing. A default value is 2.

alpha  A significance level of the Grubbs test. A default value is 0.05.

See Also

odm

---

oneplot  

*Draw a dot-plot for a selected observation (peptide)*

Description

This function draws a dot plot for a selected peptide based on the OutlierDM object.

Usage

```r
oneplot(object, i, ...)
```

```r
## S4 method for signature 'OutlierDM'
oneplot(object, i = 1, pick = 0)
```

Arguments

- **object**: a fitted object
- **i**: a row number in order or drawing a dot-plot
- **pick**: the number of locators to denote the names of the peptide
- **...**: do not use at this term

See Also

odm
OutlierDM-class

Examples

data(lcms3)
fit = odm(lcms3, method = "grubbs")
oneplot(fit, i = 100)

## Not run:
# Add row name
oneplot(fit, i = 100, pick = 1)

## End(Not run)

---

### OutlierDM-class

#### Description

A S4 class for the OutlierDM package

#### Objects from the Class

Objects can be created by calls of the form `new("OutlierDM", ...)`. See following information about slots.

#### Slots

- `call`: evaluated function call
- `raw.data`: data to be used in the fitted model
- `res`: a data.frame including the information about the fitted model. It consists of several columns including outlier, M, A, Q3, Q1, UB and LB.
- `x.pair`: a list including the information of the pairwise outlierD algorithm
- `k`: a scalar parameter for constructing boxplot used in the fitted models
- `outlier`: a boolean matrix for outlier information
- `n.outliers`: a scalar value that denotes the number of outliers to be detected by the fitted model.
- `quantreg`: type of quantile regression used for the model fitting
- `method`: type of outlier detection method used for the modeling fitting
- `contrl.par`: a list including information about tuning parameters

#### Methods

- `show` signature(object = "OutlierDM"): Same as the show method without the optional arguments
- `summary` signature(object = "OutlierDM"): Print summarized information for the fitted algorithm
- `plot` signature(x = "OutlierDM", y = "missing"): Plot an object.
\textbf{plot}\newline
\texttt{oneplot signature(x = "OutlierDM", y = "numeric"): Draw a dot-plot for a selected observation (peptide)}
\texttt{input signature(object = "OutlierDM"): Show an input data set}
\texttt{output signature(object = "OutlierDM"): Show the result}
\texttt{outliers signature(object = "OutlierDM"): Show the candidate outliers}

\textbf{See Also}
\texttt{odm}

\textbf{Examples}
\texttt{showClass("OutlierDM")}

\underline{plot} \hspace{1cm} a plot-method for a "OutlierDM" object

\textbf{Description}
This function provides a MA scatter plot with quantile regression based boxplot.

\textbf{Usage}
\begin{verbatim}
## S4 method for signature 'OutlierDM'
plot(x, y = NA, pch = 20, cex = 0.5,
     legend.use = TRUE, main, ...)
\end{verbatim}

\textbf{Arguments}
\begin{itemize}
\item \texttt{x} \hspace{1cm} fitted model object of class \texttt{odm}.
\item \texttt{y} \hspace{1cm} the "y" argument is not used in the plot-method for "OutlierDM" object.
\item \texttt{pch} \hspace{1cm} a vector of plotting characters or symbols: see \texttt{plot.default}.
\item \texttt{cex} \hspace{1cm} See \texttt{plot.default}.
\item \texttt{legend.use} \hspace{1cm} logical option for using legend box
\item \texttt{main} \hspace{1cm} main title for the plot
\item \texttt{...} \hspace{1cm} \texttt{plot.default} arguments
\end{itemize}

\textbf{Details}
This function is a method for the generic function \texttt{plot} for the S4 class \texttt{OutlierDM}. It can be invoked by calling \texttt{print} for an object of the appropriate class, or directly by calling \texttt{plot.OutlierDM} regardless of the class of the object.

\textbf{See Also}
\texttt{odm}
rgrubbs.test  
*Recursive Grubbs test for an outlier detection*

**Description**

This function works to detect outlying observation given one peptide by using the Grubbs test recursively.

**Usage**

`
  rgrubbs.test(x, alpha = 0.05)
`

**Arguments**

- `x`: a peptide
- `alpha`: a significance level alpha for a p-value

**Details**

It is a recursive version of the Grubbs test to detect outlying observations assuming that a peptide is given.

**References**


**See Also**

- `odm`

**Examples**

```r
  data(lcms3)
  rgrubbs.test(log2(lcms3[100,]))
```

---

**toy**  
*an artificial dataset for a LC/MS/MS experiment*

**Description**

An artificial dataset from the simulation study of Eo et al. (2012).

**Usage**

```r
  data("toy")
```
**Format**

A data frame with 200 peptides (rows) and 15 samples (columns).

**Source**


**Examples**

```r
data(toy)
str(toy)
pairs(log2(toy), pch = 20, cex = .7)
```
Index

*Topic **datasets**
  * lcms3, 3
  * toy, 12

*Topic **odm**
  * odm, 4
  * odm.control, 3–5
  * OutlierDM-class, 10

*Topic **package**
  * OutlierDM-package, 2

*Topic **subfunction**
  * oneplot, 9
  * plot, 11
  * rgrubbs.test, 12

input (OutlierDM-class), 10
input, OutlierDM-method
  * (OutlierDM-class), 10

lcms3, 3

odm, 3, 4, 9, 11, 12
odm.control, 3–5, 8
oneplot, 9
oneplot, OutlierDM, numeric-method
  * (oneplot), 9
oneplot, OutlierDM-method (oneplot), 9
OutlierDM (OutlierDM-package), 2
OutlierDM-class, 10
OutlierDM-package, 2
outliers (OutlierDM-class), 10
outliers, OutlierDM-method
  * (OutlierDM-class), 10
output (OutlierDM-class), 10
output, OutlierDM-method
  * (OutlierDM-class), 10

plot, 11
plot, OutlierDM, missing-method (plot), 11
plot, OutlierDM-method (plot), 11
plot.default, 11