Package: odetector (via r-universe)

October 10, 2024

Type Package

Title Outlier Detection Using Partitioning Clustering Algorithms

Version 1.0.0

Date 2022-10-01

Author Zeynel Cebeci [aut, cre]
(<https://orcid.org/0000-0002-7641-7094>), Cagatay Cebeci [ctb]
(<https://orcid.org/0000-0003-2644-1261>), Yalcin Tahtali [ctb]
(<https://orcid.org/0000-0003-0012-0611>)

Maintainer Zeynel Cebeci <zcebeci@cukurova.edu.tr>

Description An object is called ``outlier'' if it remarkably deviates from the other objects in a data set. Outlier detection is the process to find outliers by using the methods that are based on distance measures, clustering and spatial methods (Ben-Gal, 2005 <ISBN 0-387-24435-2>). It is one of the intensively studied research topics for identification of novelties, frauds, anomalies, deviations or exceptions in addition to its use for outlier removing in data processing. This package provides the implementations of some novel approaches to detect the outliers based on typicality degrees that are obtained with the soft partitioning clustering algorithms such as Fuzzy C-means and its variants.

Depends R (>= 3.0.0)

Encoding UTF-8

License GPL (>= 2)

URL https://github.com/zcebeci/odetector

BugReports https://github.com/zcebeci/odetector/issues

LazyData true

Imports ppclust, utils, graphics, grDevices

Suggests knitr, rmarkdown

VignetteBuilder knitr

Repository https://zcebeci.r-universe.dev

RemoteUrl https://github.com/zcebeci/odetector

RemoteRef HEAD

RemoteSha 745aec179081e933b6506b9cfc02cb04f499338d

Contents

`odetector-package` *Outlier Detection Using Fuzzy and Possibilistic Clustering Algorithms*

Description

An object is an "outlier" if it remarkably deviates from the other objects in a data set. Outlier detection is a process to identify outliers with the methods based on distance measures, clustering and spatial methods (Ben-Gal, 2005). This package introduces the functions for some novel approaches to detect the outliers based on the typicality degrees, obtained using the fuzzy and possibilistic clustering algorithms, i.e, the Unsupervised Possibilistic Fuzzy C-Means clustering algorithm (Wu et al, 2010).

Details

Although it is mainly called as outlier detection or anomaly detection, there are many synonym terms of outlier detection in the different application domains, i.e., fraud detection, discordants detection, exception mining, aberration detection, surprise detection, peculiarity detection or contaminant detection etc.

Outlier detection methods/algorithms can be classified with different taxonomies. In a common taxonomy, they are categorized as clustering-based methods, distance based methods and density based methods. Clustering-based methods divides data objects into clusters and seeks the objects which are not typical members of any clusters. The novel approaches applied in this package use typicality degrees from a possibilistic and fuzzy clustering algorithms. These approaches are basically decide the atypicality of data points. For example, an object is decided to be atypical if its average possibilistic membership degree to all clusters is less than a pre-defined threshold typicality degree. The objects are labeled as the outliers if they satify the above rule.

Author(s)

Zeynel Cebeci, Cagatay Cebeci, Yalcin Tahtali

References

Ben-Gal, I. (2005). Outlier detection, in *Maimon, O. & Rockach, L. (Eds.) Data Mining and Knowledge Discovery Handbook: A Complete Guide for Practitioners and Researchers*, Kluwer Academic Publishers, <ISBN 0-387-24435-2>.

Wu, X., Wu, B., Sun, J. & Fu, H. (2010). Unsupervised possibilistic fuzzy clustering. *J. of Information & Computational Sci.*, 7 (5): 1075-1080.

See Also

upfc, detect.outliers, plot.outliers, pairs.outliers, print.outliers, remove.outliers, summary.outliers

---

detect.outliers *Detect outliers using typicality degrees*

Description

The detect.outliers function finds the outliers by using four different approaches based on the typicality degrees of the data objects in a data set.

Usage

```
detect.outliers (x, k, alpha=0.05, alpha2=0.2, tsc="m1")
```

Arguments

x	an object of class 'ppclust' containing the clustering results from a possibilistic and fuzzy clustering algorithm in the package **ppclust**. Alternatively, a numeric data frame or matrix containing data set can be input to generate the object of class 'ppclust' internally.
k	an integer specifying the number of cluster. If the argument x specified as the data frame or matrix k should be also specified. Its default value is 2.
alpha	a number to specify the threshold typicality value to be used to detect the outliers. If the typicality value of an object is less than this value the object is determined as an outlier. The default value of alpha is 0.05. Although the higher value alpha leads to find more outliers it should not be increased more than 0.1.
alpha2	a number specifying the threshold typicality value to be used with the Approach 2 in order to detect the outliers. The objects which the rows sums of their typicality degrees are less than this value are evaluated as the outliers. The default value of alpha2 is 0.2. For more outliers the value of this argument should be increased.

`tsc` a string specifying the method to determine the size of small clusters for finding collective outliers. The default value is '`m1`' and the alternative is '`m2`'. See the *Details* for the details.

Details

The function `detect.outliers` computes the outliers by using four different approaches. The first approach (Approach 1) assumes that a data object is an outlier if its average typicality is less than the `alpha`, a user-defined threshold typicality degree. If the sum of typicality degrees of an object to all clusters is less than the `alpha2`, a user-defined threshold value for typicalities row sums. In the third approach (Approach 3) an object is labeled as an outlier, if its typicality to all clusters is less than the `alpha`. The last approach (Approach 4) is that all members of a small cluster are the collective outliers and can be labeled as the outliers.

With Approach 4, the members of a small clusters are considered as the collective outliers. In the function `detect.outliers`, two different methods are available to compute the threshold small cluster size (`tsc`). In the following equations, the first one has been proposed by Santos-Pereira & Pires(2002) and works good for the small data sets. The second is a novel method is proposed by the authors of this document and works better than the previous one for the larger data sets.

$$tsc_1 = 2p + 2$$

$$tsc_2 = \frac{log_2 n}{k}\, log_2 p$$

where: `p` is the number of features, `k` is the number of clusters, `n` is the number of objects.

Value

an object of class 'outliers' containing the following items:

`X` a numeric data matrix containing the processed data set.

`outliers1` a numeric vector containing the labels (row indexes) of outliers found by the Approach 1.

`outliers2` a numeric vector containing the labels (row indexes) of outliers found by the Approach 2.

`outliers3` a numeric vector containing the labels (row indexes) of outliers found by the Approach 3.

`outliers4` a numeric vector containing the labels (row indexes) objects in the small clusters to be treated as outliers.

Author(s)

Zeynel Cebeci

References

Santos-Pereira, C.M. & Pires, A.M. (2002), Detection of outliers in multivariate data: A method based on clustering and robust estimators. In *Haerdle W., Roenz B. (eds) Compstat. Physica*, Heidelberg. pp. 291-296.

Wu, X., Wu, B., Sun, J. & Fu, H. (2010). Unsupervised possibilistic fuzzy clustering. *J. of Information & Computational Sci.*, 7 (5): 1075-1080.

See Also

plot.outliers, pairs.outliers, print.outliers, remove.outliers, summary.outliers, upfc

Examples

```
# Load the dataset x3p4c and extract the first three columns
data(x3p4c)
x <- x3p4c[,1:3]

# For 4 clusters, run Unsupervised Possibilistic
# Fuzzy C-Means (UPFC) algorithm of the package ppclust
res.upfc <- ppclust::upfc(x, centers=4)

# Detect the outliers with a ppclust object
out <- detect.outliers(res.upfc)

# Summarize and plot the outliers
summary(out)
plot(out)

# Detect the outliers with a higher possibility
out <- detect.outliers(res.upfc, alpha=0.1)

# Summarize and plot the outliers
summary(out)
plot(out)

# Detect the outliers with an original data frame or matrix
x <- x3p4c[,1:3]
head(x)
out <- detect.outliers(x=x, k=4, alpha=0.1)

# Summarize and plot the outliers
summary(out)
plot(out)

# Summarize and plot the outliers
summary(out)
plot(out)
```

---

pairs.outliers *Scatter plots for diagnosing outliers*

Description

Plots the scatter plots showing the outliers found in a data set.

Usage

```
## S3 method for class 'outliers'
pairs(x, ...)
```

Arguments

x	an object of outliers class containing the outliers to be plotted.
...	additional arguments for S3 method pairs.

Value

scatter plots showing the outliers by the variable pairs.

Author(s)

Zeynel Cebeci, Cagatay Cebeci, Yalcin Tahtali

See Also

detect.outliers, plot.outliers, print.outliers, remove.outliers, summary.outliers

Examples

```
# Load the dataset x3p4c and extract the first three columns to x
data(x3p4c)
x <- x3p4c[,1:3]

# For 4 clusters, run Unsupervised Possibilistic Fuzzy C-Means (UPFC) algorithm
# of the package ppclust
res.upfc <- ppclust::upfc(x, centers=4)

# Detect the outliers
out <- detect.outliers(res.upfc)

# Plot the outliers by the variable pairs
pairs(out)
```

---

plot.outliers *Plot outliers*

Description

Plots the outliers found in a data set.

Usage

```
## S3 method for class 'outliers'
plot(x, ot=1, ...)
```

Arguments

x	an object of outliers class containing the outliers to be plotted.
ot	an integer ranges [1,4] representing the outlier detection approach.
...	additional arguments for S3 method plot.

Value

plots of the object of outliers class.

Author(s)

Zeynel Cebeci, Cagatay Cebeci, Yalcin Tahtali

See Also

detect.outliers, pairs.outliers, print.outliers, remove.outliers, summary.outliers

Examples

```
# Load the dataset x3p4c and extract the first three columns to x
data(x3p4c)
x <- x3p4c[,1:3]

# For 4 clusters, run Unsupervised Possibilistic Fuzzy C-Means (UPFC) algorithm
# of the package ppclust
res.upfc <- ppclust::upfc(x, centers=4)

# Detect the outliers
outs <- detect.outliers(res.upfc)

# Plot the outliers
plot(outs, ot=1)
```

---

print.outliers *Print outliers*

Description

Prints the outliers found in a data set.

Usage

```
## S3 method for class 'outliers'
print(x, ...)
```

Arguments

x	an object of outliers class containing the outliers to be printed. See detect.outliers for details.
...	additional arguments for S3 method print.

Value

Print out of the object of outliers class.

Author(s)

Zeynel Cebeci, Cagatay Cebeci, Yalcin Tahtali

See Also

detect.outliers, pairs.outliers, plot.outliers, remove.outliers, summary.outliers, upfc

Examples

```
# Load the dataset x3p4c and use the first three columns
data(x3p4c)
x <- x3p4c[,1:3]

# For 4 clusters, run Unsupervised Possibilistic Fuzzy C-Means (UPFC) algorithm
# of the package ppclust
res.upfc <- ppclust::upfc(x, centers=4)

# Detect the outliers
out <- detect.outliers(res.upfc)

# Print the outliers
print(out)
```

---

remove.outliers *Remove outliers*

Description

Removes the detected outliers from a data set.

Usage

```
remove.outliers(x, ot=1, sc=FALSE)
```

Arguments

x	an object of outliers class containing the outliers to be removed.
ot	an integer specifying the outlier detection approach. The default is 1 for the Approach 1. For the other methods use 2 or 3. See detect.outliers for the details.
sc	a logical value for including the objects in the small clusters into removal process. The default is 'FALSE'. Use 'TRUE' for removing the objects in the small clusters.

Value

Xr	a numeric matrix containing the outliers-removed data set.

Author(s)

Zeynel Cebeci, Cagatay Cebeci, Yalcin Tahtali

See Also

detect.outliers, pairs.outliers, plot.outliers, print.outliers, summary.outliers

Examples

```
# Load the dataset x3p4c and extract the first three columns to x
data(x3p4c)
x <- x3p4c[,1:3]

# For 4 clusters, run Unsupervised Possibilistic Fuzzy C-Means (UPFC) algorithm
# of the package ppclust
res.upfc <- ppclust::upfc(x, centers=4)

# Detect the outliers
out <- detect.outliers(res.upfc)

# Remove the outliers
Xr1 <- remove.outliers(out, ot=1)
print(Xr1)

# Remove the outliers including the collective outliers
Xr2 <- remove.outliers(out, ot=1, sc=TRUE)
print(Xr2)
```

---

summary.outliers *Summary of outliers*

Description

Summarizes the detected outliers for a data set.

Usage

```
## S3 method for class 'outliers'
summary(object, ...)
```

Arguments

object	an object of outliers class containing the outliers to be summarized. See detect.outliers for the details.
...	additional arguments for S3 method summary.

Value

Print out of the descriptive statistics for the outliers in an object of outliers class.

Author(s)

Zeynel Cebeci, Yalcin Tahtali, Cagatay Cebeci

See Also

detect.outliers, pairs.outliers, plot.outliers, print.outliers, remove.outliers

Examples

```
# Load the dataset x3p4c and extract the first three columns to x
data(x3p4c)
x <- x3p4c[,1:3]

# For 4 clusters, run Unsupervised Possibilistic Fuzzy C-Means (UPFC) algorithm
# of the package ppclust
res.upfc <- ppclust::upfc(x, centers=4)

# Detect the outliers
out <- detect.outliers(res.upfc)

# Summarize the outliers
summary(out)
```

---

x3p4c *Synthetic data set consists of three variables with four clusters*

Description

A synthetic data set which was created by using the R package 'MixSim' (Melnykov et al, 2013). It consists of three continous variables forming four clusters. The last ten rows between Line 121 and 130 of the data set contains the outliers which are labeled as the class "0".

Usage

```
data(x3p4c)
```

Format

A data frame with 130 rows and 3 numeric variables:

p1 a numeric continous variable

p2 a numeric continous variable

p3 a numeric continous variable

cl an integer variable containing the class labels. While the label '0' represents the generated outliers, the labels '1-4' stand for the classes of the clusters.

Note

The data set `x3p4c` is recommended to learn the outlier detection algorithms.

References

Melnykov, V., Chen,W-C. & Maitra, R. (2013). MixSim: An R package for simulating data to study performance of clustering algorithms. *Journal of Statistical Software*, 51(12):1-25.

Examples

```
data(x3p4c)
# Descriptive statistics of the data set
summary(x3p4c)
# Plot the data set
pairs(x3p4c[,-4], col=x3p4c[,4], pch=19, cex=2)
```

Index