

Package ‘lookout’

August 26, 2022

Type Package

Title Leave One Out Kernel Density Estimates for Outlier Detection

Version 0.1.2

Maintainer Sevvandi Kandanaarachchi <sevvandik@gmail.com>

Description Outlier detection using leave-one-out kernel density estimates and extreme value theory. The bandwidth for kernel density estimates is computed using persistent homology, a technique in topological data analysis. Using peak-over-threshold method, a generalized Pareto distribution is fitted to the log of leave-one-out kde values to identify outliers.

License GPL-3

Encoding UTF-8

RoxygenNote 7.2.1

Imports TDAstats, evd, RANN, ggplot2, tidyr

Suggests knitr, rmarkdown

URL <https://sevvandi.github.io/lookout/>

NeedsCompilation no

Author Sevvandi Kandanaarachchi [aut, cre]
(<<https://orcid.org/0000-0002-0337-0395>>),
Rob Hyndman [aut] (<<https://orcid.org/0000-0002-2140-5352>>)

Repository CRAN

Date/Publication 2022-08-26 08:00:05 UTC

R topics documented:

autoplot.lookoutliers	2
autoplot.persistingoutliers	2
lookout	3
lookout_ts	4
persisting_outliers	5
Index	7

autoplot.lookoutliers *Plots outliers identified by lookout algorithm.*

Description

Scatterplot of two columns from the data set with outliers highlighted.

Usage

```
## S3 method for class 'lookoutliers'
autoplot(object, columns = 1:2, ...)
```

Arguments

object	The output of the function 'lookout'.
columns	Which columns of the original data to plot (specified as either numbers or strings)
...	Other arguments currently ignored.

Value

A ggplot object.

Examples

```
X <- rbind(
  data.frame(x = rnorm(500),
            y = rnorm(500)),
  data.frame(x = rnorm(5, mean = 10, sd = 0.2),
            y = rnorm(5, mean = 10, sd = 0.2))
)
lo <- lookout(X)
autoplot(lo)
```

autoplot.persistingoutliers
Plots outlier persistence for a range of significance levels.

Description

This function plots outlier persistence for a range of significance levels using the algorithm lookout, an outlier detection method that uses leave-one-out kernel density estimates and generalized Pareto distributions to find outliers.

Usage

```
## S3 method for class 'persistingoutliers'
autoplot(object, alpha = object$alpha, ...)
```

Arguments

object	The output of the function ‘persisting_outliers‘.
alpha	The significance levels to plot.
...	Other arguments currently ignored.

Value

A ggplot object.

Examples

```
X <- rbind(
  data.frame(
    x = rnorm(500),
    y = rnorm(500)
  ),
  data.frame(
    x = rnorm(5, mean = 10, sd = 0.2),
    y = rnorm(5, mean = 10, sd = 0.2)
  )
)
plot(X, pch = 19)
outliers <- persisting_outliers(X, unitize = FALSE)
autoplot(outliers)
```

lookout	<i>Identifies outliers using the algorithm lookout.</i>
---------	---

Description

This function identifies outliers using the algorithm lookout, an outlier detection method that uses leave-one-out kernel density estimates and generalized Pareto distributions to find outliers.

Usage

```
lookout(X, alpha = 0.05, unitize = TRUE, bw = NULL, gpd = NULL)
```

Arguments

X	The input data in a dataframe, matrix or tibble format.
alpha	The level of significance. Default is 0.05.
unitize	An option to normalize the data. Default is TRUE, which normalizes each column to $[0, 1]$.
bw	Bandwidth parameter. Default is NULL as the bandwidth is found using Persistent Homology.
gpd	Generalized Pareto distribution parameters. If ‘NULL’ (the default), these are estimated from the data.

Value

A list with the following components:

outliers	The set of outliers.
outlier_probability	The GPD probability of the data.
outlier_scores	The outlier scores of the data.
bandwidth	The bandwidth selected using persistent homology.
kde	The kernel density estimate values.
lookde	The leave-one-out kde values.
gpd	The fitted GPD parameters.

Examples

```
X <- rbind(
  data.frame(x = rnorm(500),
            y = rnorm(500)),
  data.frame(x = rnorm(5, mean = 10, sd = 0.2),
            y = rnorm(5, mean = 10, sd = 0.2))
)
lo <- lookout(X)
lo
autoplot(lo)
```

lookout_ts	<i>Identifies outliers in univariate time series using the algorithm lookout.</i>
------------	---

Description

This is the time series implementation of lookout.

Usage

```
lookout_ts(x, alpha = 0.05)
```

Arguments

x	The input univariate time series.
alpha	The level of significance. Default is 0.05.

Value

A lookout object.

See Also[lookout](#)**Examples**

```
set.seed(1)
x <- arima.sim(list(order = c(1,1,0), ar = 0.8), n = 200)
x[50] <- x[50] + 10
plot(x)
lo <- lookout_ts(x)
lo
```

`persisting_outliers` *Computes outlier persistence for a range of significance values.*

Description

This function computes outlier persistence for a range of significance values, using the algorithm `lookout`, an outlier detection method that uses leave-one-out kernel density estimates and generalized Pareto distributions to find outliers.

Usage

```
persisting_outliers(
  X,
  alpha = seq(0.01, 0.1, by = 0.01),
  st_qq = 0.9,
  unitize = TRUE,
  num_steps = 20
)
```

Arguments

<code>X</code>	The input data in a matrix, data.frame, or tibble format. All columns should be numeric.
<code>alpha</code>	Grid of significance levels.
<code>st_qq</code>	The starting quantile for death radii sequence. This will be used to compute the starting bandwidth value.
<code>unitize</code>	An option to normalize the data. Default is TRUE, which normalizes each column to $[0, 1]$.
<code>num_steps</code>	The length of the bandwidth sequence.

Value

A list with the following components:

<code>out</code>	A 3D array of $N \times \text{num_steps} \times \text{num_alpha}$ where N denotes the number of observations, <code>num_steps</code> denote the length of the bandwidth sequence and <code>num_alpha</code> denotes the number of significance levels. This is a binary array and the entries are set to 1 if that observation is an outlier for that particular bandwidth and significance level.
<code>bw</code>	The set of bandwidth values.
<code>gpdparas</code>	The GPD parameters used.
<code>lookoutbw</code>	The bandwidth chosen by the algorithm <code>lookout</code> using persistent homology.

Examples

```
X <- rbind(
  data.frame(x = rnorm(500),
            y = rnorm(500)),
  data.frame(x = rnorm(5, mean = 10, sd = 0.2),
            y = rnorm(5, mean = 10, sd = 0.2))
)
plot(X, pch = 19)
outliers <- persisting_outliers(X, unitize = FALSE)
outliers
autoplot(outliers)
```

Index

`autoplot.lookoutliers`, 2
`autoplot.persistingoutliers`, 2

`lookout`, 3, 5
`lookout_ts`, 4

`persisting_outliers`, 5