Package 'akmedoids'

March 2, 2019

Type Package

Title Akmedoids: Anchored kmedoids for Longitudinal Data

Version 0.1.0

Date 2019-02-06

Author Monsuru Adepeju [cre, aut], Samuel Langton [aut], Jon Bannister [aut]

Maintainer Monsuru Adepeju <monsuurg2010@gmail.com>

Description Advances a longitudinal clustering technique, 'akmeans' which identify groups of trajectories of similar
long-term linear trendline and use the Calinski and Harabatz criterion to determine the optimal separation between the trendlines.
The package also include a list of other useful functions for exploring and manipulating longitudinal data/repeated dataset.

Depends R (>= 3.5.0)

License GPL-2

Encoding UTF-8

LazyData true

Imports kml, devtools, Hmisc, ggplot2, rgdal, base, utils, reshape2, later, Rdpack, longitudinalData

RoxygenNote 6.1.1

Suggests knitr,
rmarkdown

VignetteBuilder knitr

R topics documented:

---

akmedoids.clust   *Clustering of longitudinal data*

Description

This function group trajectories based on a functional definition of their long-term trends.

Usage

```
akmedoids.clust(traj, id_field = FALSE, method = "linear", k = c(3,6)) ##' #details Given a list of t
```

Arguments

traj	A matrix or data.frame with each row representing the trajectory of observations of a unique location. The columns show the observations at consecutive time steps.
id_field	Whether the first column is a unique (id) field. Default: FALSE
method	Initialisation strategy. Available method: linear
k	either an exact integer number of clusters, or a vector of length two specifying the minimum and maximum numbers of clusters to be examined. The default is c(3,15). When k is a range, the actual number of clusters is determined by Calinski-Harabatz criterion. number of clusters to generate. Default: 3: (minimum value)

Value

A list containing cluster solutions for all value of k, including the solution at the optimal value of k, based on the Calinski-Harabatz criterion (Calinski T, Harabasz J, 1974)

References

Calinski T, Harabasz J (1974) A dendrite method for cluster analysis. Commun Stat 3:1-27

Examples

```
traj <- gm.crime.sample1
print(traj)
traj <- missingVal(traj, id_field = TRUE, method = 2, replace_with = 1,
fill_zeros = FALSE) #filling the missing values
traj <- props(traj, id_field = TRUE)
print(traj)
output <- akmedoids.clust(traj, id_field = TRUE, method = "linear", k = c(3,6))
print(output)
```

---

alphaLabel   *Numerics ids to alphabetical ids*

Description

Function to transform a list of numeric ids to alphabetic ids

Usage

```
alphaLabel(x)
```

Arguments

x   A vector of numeric ids

---

gm.crime.sample1   *Sample crime dataset*

Description

Simulated crime dataset with missing values.

Usage

```
gm.crime.sample1
```

Format

A matrix

---

missingVal   *Data imputing for longitudinal data*

Description

This function fills up any missing entries (NA, Inf, 0) in a matrix or dataframe using a value derived using a chosen method.

Usage

```
missingVal(traj, id_field = FALSE, method = 2, replace_with = 1, fill_zeros = FALSE)
```

Arguments

`traj`	A matrix or data.frame with each row representing the trajectory of a unique location. The columns show the observations at consecutive time steps.
`id_field`	Whether the first column is a unique (id) field. Default: FALSE
`method`	Method for calculating the missing values. Available options: 1: arithmetic, 2: regression. default: 1
`replace_with`	How to calculate the missing value. For `arithmetic` method: `replace_with` options are: 1: Mean value of column, 2: Minimum value of column, 3: Maximum value of column, 4: Mean value of row, 5: Minimum value of row, or 6: Maximum value of row. For `regression` method: the only available option for `replace_with` is: 1: linear. That is, use a linear regression to interpolate or extrapolate the missing data values. Note: only the missing data points derive their new values from the regression line while the rest of the data points retain their original values. Trajectories with only one observation will be removed.
`fill_zeros`	Whether to consider zeros (0) as missing values. Default: FALSE. Only available for 2: `regression` method.

Details

Given a matrix or data.frame with some missing values represented by (`NA`, `Inf`, `0`), the function `missingVal` determines the missing values using either the `arithmetic` or `regression` method.

Value

A data.frame with missing values (`NA`, `Inf`, `0`) filled up

Examples

```
traj <- gm.crime.sample1
print(traj)
missingVal(traj, id_field = TRUE, method = 2, replace_with = 1, fill_zeros = FALSE)
```

---

outlierDetect	*Outlier detection and replacements*

Description

This function identifies outlier observations in the trajectories and determine whether to replace them or remove the trajectories accordingly.

Usage

```
outlierDetect(traj, id_field = FALSE, method = 1, threshold = 0.95, hortz_tolerance = 1, replace_wit
```

Arguments

`traj`	A matrix or dataframe with each row representing the trajectory of observations of a unique location. The columns show the observations at consecutive time steps.
`id_field`	Whether the first column is a unique (id) field. Default: `FALSE`
`method`	Specify the method for identifying the outlier. Available methods: 1: quantile method, and 2: manual method".
`threshold`	The cut off value for an observation to be flagged as outlier. For quantile method, `threshold` options are: a numeric vector of probability between `[0,1]`. For manual method, `threshold` options are: a `user-specified value`
`hortz_tolerance`	
	[Default: 1]. Specifying the count of outlier observations that must exist in a trajectory in order for the trajectory to be considered an `outlier`.
`replace_with`	How to replace the outlier observations. [Default: 1] - mean value of all the observations in the column in which an outlier observation is located. Other options are: 2: Mean value of row in which the observation is found, 3: To remove the outlier trajectory.

Details

Given a matrix or data.frame with some suspected outlier observations, this function identified those observations based on the method chosen and replace all the observations accordingly.

Value

A dataframe with outlier observations replaced or trajectories containing outlier observation removed.

Examples

```
traj <- gm.crime.sample1
traj <- missingVal(traj, id_field=TRUE, replace_with=1)
outlierDetect(traj, id_field = FALSE, method = 1, threshold = 0.95,
hortz_tolerance = 1, replace_with = 1)
```

---

plotr	*Plotting the cluster solution*

Description

Function to plot the clusters generated by the akmedoids.clust function

Usage

```
plotr(clustr, traj, id_field = TRUE, y.scaling = "fixed")
```

Arguments

clustr	A vector of cluster labels (e.g. from an output of [akmedoids.clust](#))
traj	The matrix or data.frame of trajectories from which the cluster solution clustr is generated.
id_field	Whether the first column of traj is a unique (id) field. Default: TRUE.
y.scaling	To set the vertical scales of the cluster panels. Options are: ″fixed″: uses uniform scale for all panels, ″free″: uses variable scales for panels.

Details

Given a matrix or data.frame with some suspected outlier observations, this function identified those observations based on the method chosen and replace all the observations accordingly.

Value

A plot showing clusters of trajectories, including the mean trajectory of each group.

References

Wickham H. (2016). Elegant graphics for Data Analysis. Spring-Verlag New York (2016)

Examples

```
traj <- gm.crime.sample1
print(traj)
traj <- missingVal(traj, id_field = TRUE, method = 2, replace_with = 1,
fill_zeros = FALSE) #filling the missing values
print(traj)
traj <- props(traj, id_field = TRUE)
clustr <- akmedoids.clust(traj, id_field = TRUE, method = "linear", k = c(3,6))
print(plotr(clustr, traj, id_field=TRUE, y.scaling="fixed"))
```

---

props	*Convert counts or rates to proportion*

Description

This function converts counts or rates data to proportion.

Usage

```
props(traj, id_field = FALSE)
```

Arguments

traj	A matrix or data.frame with each row representing the trajectory of observations of a unique location. The columns show the observation at consecutive time steps.
id_field	Whether the first column is a unique (id) field. Default: FALSE

Details

Given a matrix of observations (counts or rates), this function convert each observation to a proportion measure. A matrix or data.frame count or rate is converted by dividing each cell entry by the sum of the corresponding column, i.e. prop = [a cell value] / sum[column]

Value

A matrix of proportion measures

Examples

```
traj <- gm.crime.sample1
head(traj) #
traj <- missingVal(traj, id_field = TRUE, method = 2, replace_with = 1,
fill_zeros = FALSE) #filling the missing values
traj <- props(traj, id_field = TRUE)
print(traj)
```

---

wSpaces	*Removing whitespaces*

Description

This function removes all leading and trailing whitespaces in longitudinal data

Usage

```
wSpaces(traj)
```

Arguments

traj   A matrix or data.frame with each row representing the trajectory of observations of a unique location. The columns show the observation at consecutive time steps.

Value

A cleaned data.frame with whitespaces (if any) removed.

References

https://en.wikipedia.org/wiki/Whitespace_character

Examples

```
traj <- gm.crime.sample1
wSpaces(traj)
```

Index