# Package 'akmeans'

February 20, 2019

**Type** Package

**Title** akmeans: 'Anchored' kmeans 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 an akmeans clustering technque and a stability-based quality criterion for longitudinal data. Also, contains functions for useful for the analysis of longitudinal data.

**License** GPL-2

**Encoding** UTF-8

**LazyData** TRUE

**Imports** kml, devtools, Hmisc, ggplot2, rgdal, base, utils, reshape2, later

**Suggests** knitr, rmarkdown

**RoxygenNote** 6.1.1

**VignetteBuilder** knitr

## R topics documented:

## akmeans_clust *akmeans_clust*

### Description

This function group trajectories based on a given list of initial centroids

### Usage

```
akmeans_clust(traj, id_field = FALSE, init_method = "lpm", n_clusters = 3)
```

### 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 |
| init_method | initialisation method. Specifying a method to determine the initial centroids for clustering. Default: "lpm" - linear partitioning medoids @seealso lpm_centroids] |
| n_clusters | number of clusters to generate. Default: 3: (minimum value) |

### Details

Given a list of trajectories represented in a matrix or data.frame, and a method for choosing initial cluster centroids (e.g. lpm_centroids), a list of clusters is generated after a limited number of iterations. traj <- assault_data print(traj) result <- akmeans_clust(traj, id_field = TRUE, init_method = "lpm", n_clusters = 3) plot_clust(result)

### Value

The original (traj) data with cluster label appended

## 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 |

## assault_data *Sample crime (assault) dataset*

### Description

Simulated crime dataset with missing values.

### Usage

```
assault_data
```

### Format

A matrix

## gm_crime_data *Sample crime dataset*

### Description

Crime dataset of greater Manchester crime data aggregated at the LSOA geographical level data (Source: data.police.uk)

### Usage

```
gm_crime_data
```

### Format

A matrix

## lpm_centroids *Linear Partition Medoids (LPM) Centroids*

### Description

This function to create the initial centroids based on linear partitioning medoids (lpm) initialisation (Adepeju et al. 2019, submitted)

### Usage

```
lpm_centroids(dat, id_field2 = FALSE, n_centroids = 3)
```

### Arguments

| | |
|---|---|
| dat | 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_field2 | Whether the first column is a unique (id) field. default: FALSE |
| n_centroids | Number of initial (linear) centroids to generate based on lpm technique |

## Value

l_centroids

## References

Adepeju M, Langton S, Bannister J. (2019). akmeans: Anchored k-means: A longitudinal clustering technique for measuring long-term inequality in the exposure to crime at the micro-area levels (submitted).

---

`missingV_filler` *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

```
missingV_filler(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 `missingV_filler` 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 <- assault_data
print(traj)
missingV_filler(traj, id_field = TRUE, method = 2, replace_with = 1, fill_zeros = FALSE)
```

---

`outlierDetect` *Outlier detection in longitudinal or repeated observations*

---

## Description

Detect outlier in a longitudinal or repeated data. This function identify the outlier observations according to a specified method. A matrix, 'outlier_mat', is created with entries 'TRUE' or 'FALSE' indicating whether or not an observation is an outlier. The final list of outlier trajectories is determined by the 'hortz_tolerance' parameter i.e. how many observation in a trajectory exceed the 'threshold' value.

## Usage

```
outlierDetect(dat, id_field = FALSE, method = "quantile",
  threshold = 0.95, hortz_tolerance = 1, replace_with = "Mean_row")
```

## Arguments

| | |
|---|---|
| `dat` | 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] |
| `method` | Specify the method for identifying the outlier. Available methods: (1) "quantile" (2) "manual" - a user-defined value |
| `threshold` | Value in which an observation must exceed in order to be flagged as outlier. Depending on the method specified: (1) for "quantile" method, enter a numeric vector of probabilities with values in [0,1], (2) for "Manual" method: a user-specified value. |
| `hortz_tolerance` | Specifying the number of observations of a trajectory that have to exceed the cut-off 'threshold' value in order for the trajectory to be flagged as outlier. [default: 1] |
| `replace_with` | Value to replace the outlier observation with. Values to replace with [Values: "Mean_col" or "Mean_row"]. The default is "Mean_row", meaning to imput the average values of the field in which the observation is located. |

## Value

dat_

## plot_clust *To plot the clusters*

### Description

To plot the clusters

### Usage

```
plot_clust(data_clusters_list, id_field = TRUE)
```

### Arguments

| | |
|---|---|
| data_clusters_list | A data.frame of clusters from akmeans_clust, in which the last column represents alphabetical cluster ids (labels) |
| id_field | Whether the first column is a unique (id) field. [default: TRUE] |

### Value

data_clusters_list

## props *Function to convert counts or rates to proportion*

### Description

Function to convert counts or rates to proportion

### Usage

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

### Arguments

| | |
|---|---|
| rates | 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] |

### Value

props

## qpm_centroids *Quadratic Partition Medoids (QPM) Centroids*

### Description

Quadratic Partition Medoids (QPM) Centroids

### Usage

```
qpm_centroids(dat, n_centroids = 3, id_field = FALSE)
```

### Arguments

| | |
|---|---|
| dat | 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. |
| n_centroids | Number of initial (quadratic) centroids to generate based on the qpm method (See attached Vignette) |
| id_field | Whether the first column is a unique (id) field. [default: FALSE] |

### Value

q_centroids

## whiteSpaces *Function to remove whitespaces in data entries*

### Description

Function to remove whitespaces in data entries

### Usage

```
whiteSpaces(dat, head = TRUE)
```

### Arguments

| | |
|---|---|
| dat | A matrix or data.frame |
| head | If column names exist |

### Value

dat_Cleaned

# Index