

# Package ‘MajKMeans’

January 20, 2025

**Type** Package

**Title** k-Means Algorithm with a Majorization-Minimization Method

**Version** 0.1.0

**Maintainer** Sheikhi Ayyub <sheikhy.a@uk.ac.ir>

**Description** A hybrid of the K-means algorithm and a Majorization-Minimization method to introduce a robust clustering. The reference paper is: Julien Mairal, (2015) <[doi:10.1137/140957639](https://doi.org/10.1137/140957639)>. The two most important functions in package 'MajKMeans' are cluster\_km() and cluster\_MajKm(). cluster\_km() clusters data without Majorization-Minimization and cluster\_MajKm() clusters data with Majorization-Minimization method. Both of these functions calculate the sum of squares (SS) of clustering.

**Imports** MASS

**License** GPL-3

**Encoding** UTF-8

**LazyData** true

**RoxygenNote** 7.0.2

**NeedsCompilation** no

**Author** Sheikhi Ayyub [aut, cre],  
Yaghoubi Mohammad Ali [aut]

**Repository** CRAN

**Date/Publication** 2020-03-28 14:40:02 UTC

## Contents

clusters_km . . . . .	2
clusters_MajKm . . . . .	2
Euclid . . . . .	3
kmeans . . . . .	4

<b>Index</b>	<b>5</b>
--------------	----------

---

clusters\_km                      *clustering results of the k-mean algorithm*

---

### Description

clusters data into two clusters. This function uses the kmeans function to cluster the data and exports the clustering results as well as the sum of square (SS) of clustering using the Euclidian distance.

### Usage

```
clusters_km(x, k = 2)
```

### Arguments

x                      matrix of data (dim 1: samples (must be equal to dim 1 of X), dim 2: attributes (must be equal to dim 2 of X))

k                      number of clusters ( this version considers 2 clusters )

### Value

sum of square (SS) of clustering

### Examples

```
{
X=rbind(matrix(rnorm(1000*2) ,4,.1),1000,2),matrix(rnorm(1000*2, 3, 0.2),1000,2))
M <- X[sample(nrow(X), 2),]
clusters_km(X,2)
}
```

---

clusters\_MajKm                      *clustering results of the majorized k-mean algorithm*

---

### Description

clusters data into two clusters with a majorization k-means This function uses a hybrid of the k-means and the majorization-minimization method to cluster the data and exports the clustering results as well as the sum of square (SS) of clustering

### Arguments

x                      matrix of data (dim 1: samples (must be equal to dim 1 of X), dim 2: attributes (must be equal to dim 2 of X))

k                      number of clusters ( this version considers 2 clusters )

La                      the tunning parameter

**Value**

sum of square (SS) of clustering and the 'delta' (difference of two successive majorization function).

**Examples**

```
{
X=rbind(matrix(rnorm(1000*2 ,4,.1),1000,2),matrix(rnorm(1000*2, 3, 0.2),1000,2))
M <- X[sample(nrow(X), 2),]
clusters_MajKm(X,2, 0.5)
}
```

---

Euclid

*Euclidian distance*

---

**Description**

Calculates the Euclidian distance between points. This function can use in kmeans function to do the clustering procedure using the Euclidian distance.

**Usage**

```
Euclid(x, mu)
```

**Arguments**

x                    matrix of data (dim 1: samples (must be equal to dim 1 of X), dim 2: attributes (must be equal to dim 2 of X))

mu                    initial selected centroids (randomly or another method).

**Value**

Euclidian distance between two points.

**Examples**

```
{
X=rbind(matrix(rnorm(1000*2 ,4,.1),1000,2),matrix(rnorm(1000*2, 3, 0.2),1000,2))
M <- X[sample(nrow(X), 2),]
Euclid(X,M)
}
```

---

kmeans	<i>k-means function</i>
--------	-------------------------

---

**Description**

k-means algorithm in clustering. This function export the clustered results based on one replication of the k-means method

**Arguments**

x	matrix of data (dim 1: samples (must be equal to dim 1 of X), dim 2: attributes (must be equal to dim 2 of X))
centers	initial seleted centroids (randomly or another method)
distFun	function (in this package the distance is Euclidian)
nItter	Number of itteration function

**Value**

clustered results based on k-means methods.

**Examples**

```
{  
X=rbind(matrix(rnorm(1000*2 ,4,.1),1000,2),matrix(rnorm(1000*2, 3, 0.2),1000,2))  
M <- X[sample(nrow(X), 2),]  
kmeans(X,M, Euclid, 4)  
}
```

# Index

clusters\_km, [2](#)  
clusters\_MajKm, [2](#)  
Euclid, [3](#)  
kmeans, [4](#)