

Package ‘MKMeans’

September 8, 2025

Type Package

Title A Modern K-Means (MKMeans) Clustering Algorithm

Version 3.4.4

Date 2025-09-08

Depends methods, MASS, gtools

Description It's a Modern K-Means clustering algorithm which works for data of any number of dimensions, has no limit with the number of clusters expected, offers both methods with and without initial cluster centers, and can start with any initial cluster centers for the method with initial cluster centers.

Collate AllClasses.R MKMeans.R C.f.R Dist.R

License GPL-2

NeedsCompilation no

Author Yarong Yang [aut, cre],
Nader Ebrahimi [ctb],
Yoram Rubin [ctb],
Jacob Zhang [ctb]

Maintainer Yarong Yang <Yi.YA_yaya@hotmail.com>

Repository CRAN

Date/Publication 2025-09-08 12:00:14 UTC

Contents

MKMeans-package	2
C.f	3
Dist	4
MKMean	5
MKMeans	6

Index	9
--------------	----------

MKMeans-package*Modern K-Means (MKMeans) Clustering.*

Description

It's a Modern K-Means clustering algorithm which works for data of any number of dimensions, has no limit with the number of clusters expected, offers both methods with and without initial cluster centers, and can start with any initial cluster centers for the method with initial cluster centers.

Details

Package: MKMeans
Type: Package
Version: 3.4.4
Date: 2025-09-08
License: GPL-2

Author(s)

Yarong Yang, Nader Ebrahimi, Yoram Rubin, and Jacob Zhang

References

Yarong Yang, Nader Ebrahimi, Yoram Rubin, and Jacob Zhang.(2025) MKMeans: A Modern K-Means Clustering Algorithm. technical report in preparation

Examples

```
# Generate 10 bivariate normal samples
mu1 <- c(0, 0)
sigma1 <- matrix(c(1, 0.5, 0.5, 1), nrow=2)
SP1 <- mvrnorm(n=10, mu=mu1, Sigma=sigma1)

# Generate another 10 bivariate normal samples
mu2<-c(1,1)
sigma2<-matrix(c(1,0,0,1),nrow=2)
SP2<-mvrnorm(n=10,mu=mu2,Sigma=sigma2)

# Generate 10 more new bivariate normal samples
mu3<-c(2,2)
sigma3<-matrix(c(1,0.5,0.5,1),nrow=2)
SP3<-mvrnorm(n=10,mu=mu3,Sigma=sigma3)

# Combine the three groups of bivariate normal samples
data<-rbind(SP1,SP2,SP3)
```

```

# Conduct MKMeans analysis with K=3 by the "YY" method
Res<-MKMeans(data,3,method="YY",iteration=1000,type=1)
names(Res@Classes[[1]])<-rep("red",length(Res@Classes[[1]]))
names(Res@Classes[[2]])<-rep("blue",length(Res@Classes[[2]]))
names(Res@Classes[[3]])<-rep("green",length(Res@Classes[[3]]))
Cols<-names(sort(c(Res@Classes[[1]],Res@Classes[[2]],Res@Classes[[3]])))
plot(data[,1],data[,2],type="p",pch=19,col=Cols,lwd=2,xlab=paste("Total SSE = ",
  round(Res@SSE[length(Res@SSE)],2),sep=""),ylab="",
  main="MKMeans Clustering Results by the 'YY' method")
points(Res@Centers,pch=5,col=c("red","blue","green"))

# Compare the clustering results with the original samples
par(mfrow=c(1,2))
plot(data[,1],data[,2],type="p",pch=19,col=rep(c("sky blue","orange","purple"),rep(10,3)),
  lwd=2,xlab="",ylab="",main="Original Data")
plot(data[,1],data[,2],type="p",pch=19,col=Cols,lwd=2,xlab=paste("Total SSE = ",
  round(Res@SSE[length(Res@SSE)],2),sep=""),ylab="",
  main="MKMeans Clustering Results by the 'YY' method")
points(Res@Centers,pch=5,col=c("red","blue","green"))

# Conduct MKMeans analysis with K=3 and randomly picking 3 samples as initial cluster centers
Res2<-MKMeans(data,3,method="initial",initial=data[sample(1:nrow(data),3),],iteration=1000,type=1)
names(Res2@Classes[[1]])<-rep("red",length(Res2@Classes[[1]]))
names(Res2@Classes[[2]])<-rep("blue",length(Res2@Classes[[2]]))
names(Res2@Classes[[3]])<-rep("green",length(Res2@Classes[[3]]))
Cols2<-names(sort(c(Res2@Classes[[1]],Res2@Classes[[2]],Res2@Classes[[3]])))
plot(data[,1],data[,2],type="p",pch=19,col=Cols2,lwd=2,xlab=paste("Total SSE = ",
  round(Res2@SSE[length(Res2@SSE)],2),sep=""),ylab="",
  main="MKMeans Clustering Results by the 'initial' method")
points(Res2@Centers,pch=5,col=c("red","blue","green"))

# Compare the clustering results by the "YY" method and by the "initial" method
par(mfrow=c(1,2))
plot(data[,1],data[,2],type="p",pch=19,col=Cols,lwd=2,xlab=paste("Total SSE = ",
  round(Res@SSE[length(Res@SSE)],2),sep=""),ylab="",
  main="MKMeans Clustering Results by the 'YY' method")
points(Res@Centers,pch=5,col=c("red","blue","green"))
plot(data[,1],data[,2],type="p",pch=19,col=Cols2,lwd=2,xlab=paste("Total SSE = ",
  round(Res2@SSE[length(Res2@SSE)],2),sep=""),ylab="",
  main="MKMeans Clustering Results by the 'initial' method")
points(Res2@Centers,pch=5,col=c("red","blue","green"))

```

C.f

*Finding the center of a cluster.***Description**

It's a function of finding the center of a cluster.

Usage

```
C.f(dat, type)
```

Arguments

<code>dat</code>	Numeric. A cluster matrix with each row being an observaion.
<code>type</code>	Integer. The type of distance between observations. 1 for Euclidean distance. 2 for Manhattan distance. 3 for maximum deviation along dimensions.

Value

A vector.

Author(s)

Yarong Yang

Examples

```
x<-rnorm(5,0,1)
y<-rnorm(5,1,1)
data<-cbind(x,y)
Res<-C.f(dat=data,type=1)
```

Dist

Finding the distance between two observations.

Description

It's a function of finding the distance between two observations.

Usage

```
Dist(x,y,type)
```

Arguments

<code>x</code>	Numeric. A vector denoting an observation.
<code>y</code>	Numeric. A vector denoting an observation.
<code>type</code>	Integer. The type of distance between observations. 1 for Euclidean distance. 2 for Manhattan distance. 3 for maximum deviation among dimensions.

Value

A numeric number.

Examples

```
x<-rnorm(10,0,1)
y<-rnorm(10,1,1)
z<-rnorm(10,2,1)
data<-cbind(x,y,z)
Res<-Dist(data[1,],data[2,],type=1)
```

MKMean

*Class to contain the results from function MKMeans.***Description**

The function MKMeans return object of class MKMean that contains the number of clusters, the center of each cluster, and the observations in each cluster.

Objects from the Class

```
new("MKMean",K=new("numeric"),Centers=new("matrix"),Classes=new("list"),Clusters=new("list"),SSE=new("numeric"))
```

Slots

K: An integer being the number of clusters.

Centers: A numeric matrix with each row being center of a cluster.

Classes: An integer list showing the original indexes of the observations in each cluster.

Clusters: A numeric list showing the observations in each cluster.

SSE: A numeric vector composed of SSE of each cluster and the total SSE of all the clusters.

Author(s)

Yarong Yang

References

Yarong Yang, Nader Ebrahimi, Yoram Rubin, and Jacob Zhang.(2025) MKMeans: A Modern K-Means Clustering Algorithm. technical report in preparation

Examples

```
showClass("MKMean")
```

MKMeans

*Modern K-Means clustering.***Description**

It's a Modern K-Means clustering algorithm which works for data of any number of dimensions, has no limit with the number of clusters expected, offers both methods with and without initial cluster centers, and can start with any initial cluster centers for the method with initial cluster centers.

Usage

```
MKMeans(data, K, method, initial, iteration, type)
```

Arguments

data	Numeric. An observation matrix with each row being an observation.
K	Integer. The number of clusters expected.
method	Character. "YY" or "initial". No initial cluster centers are required for "YY" method. "initial" method can work for any initial cluster centers.
initial	Numeric. Either the selected initial center matrix with each row being an observation, or 1 for the first K rows of the data matrix being the initial center.
iteration	Integer. The number of the most iterations wanted for the clustering process.
type	Integer. The type of distance between observations. 1 for Euclidean distance. 2 for Manhattan distance. 3 for maximum deviation among dimensions.

Value

An object of class MKMean.

Author(s)

Yarong Yang

References

Yarong Yang, Nader Ebrahimi, Yoram Rubin, and Jacob Zhang.(2025) MKMeans: A Modern K-Means Clustering Algorithm. technical report in preparation

Examples

```
# Generate 10 bivariate normal samples
mu1 <- c(0, 0)
sigma1 <- matrix(c(1, 0.5, 0.5, 1), nrow=2)
SP1 <- mvrnorm(n=10, mu=mu1, Sigma=sigma1)

# Generate another 10 bivariate normal samples
mu2<-c(1,1)
```

```

sigma2<-matrix(c(1,0,0,1),nrow=2)
SP2<-mvrnorm(n=10,mu=mu2,Sigma=sigma2)

# Generate 10 more new bivariate normal samples
mu3<-c(2,2)
sigma3<-matrix(c(1,0.5,0.5,1),nrow=2)
SP3<-mvrnorm(n=10,mu=mu3,Sigma=sigma3)

# Combine the three groups of bivariate normal samples
data<-rbind(SP1,SP2,SP3)

# Conduct MKMeans analysis with K=3 by the "YY" method
Res<-MKMeans(data,3,method="YY",iteration=1000,type=1)
names(Res@Classes[[1]])<-rep("red",length(Res@Classes[[1]]))
names(Res@Classes[[2]])<-rep("blue",length(Res@Classes[[2]]))
names(Res@Classes[[3]])<-rep("green",length(Res@Classes[[3]]))
Cols<-names(sort(c(Res@Classes[[1]],Res@Classes[[2]],Res@Classes[[3]])))
plot(data[,1],data[,2],type="p",pch=19,col=Cols,lwd=2,xlab=paste("Total SSE = ",
  round(Res@SSE[length(Res@SSE)],2),sep=""),ylab="",
  main="MKMeans Clustering Results by the 'YY' method")
points(Res@Centers,pch=5,col=c("red","blue","green"))

# Compare the clustering results with the original samples
par(mfrow=c(1,2))
plot(data[,1],data[,2],type="p",pch=19,col=rep(c("sky blue","orange","purple"),rep(10,3)),
  lwd=2,xlab="",ylab="",main="Original Data")
plot(data[,1],data[,2],type="p",pch=19,col=Cols,lwd=2,xlab=paste("Total SSE = ",
  round(Res@SSE[length(Res@SSE)],2),sep=""),ylab="",
  main="MKMeans Clustering Results by the 'YY' method")
points(Res@Centers,pch=5,col=c("red","blue","green"))

# Conduct MKMeans analysis with K=3 and randomly picking 3 samples as initial cluster centers
Res2<-MKMeans(data,3,method="initial",initial=data[sample(1:nrow(data),3),],iteration=1000,type=1)
names(Res2@Classes[[1]])<-rep("red",length(Res2@Classes[[1]]))
names(Res2@Classes[[2]])<-rep("blue",length(Res2@Classes[[2]]))
names(Res2@Classes[[3]])<-rep("green",length(Res2@Classes[[3]]))
Cols2<-names(sort(c(Res2@Classes[[1]],Res2@Classes[[2]],Res2@Classes[[3]])))
plot(data[,1],data[,2],type="p",pch=19,col=Cols2,lwd=2,xlab=paste("Total SSE = ",
  round(Res2@SSE[length(Res2@SSE)],2),sep=""),ylab="",
  main="MKMeans Clustering Results by the 'initial' method")
points(Res2@Centers,pch=5,col=c("red","blue","green"))

# Compare the clustering results by the "YY" method and by the "initial" method
par(mfrow=c(1,2))
plot(data[,1],data[,2],type="p",pch=19,col=Cols,lwd=2,xlab=paste("Total SSE = ",
  round(Res@SSE[length(Res@SSE)],2),sep=""),ylab="",
  main="MKMeans Clustering Results by the 'YY' method")
points(Res@Centers,pch=5,col=c("red","blue","green"))
plot(data[,1],data[,2],type="p",pch=19,col=Cols2,lwd=2,xlab=paste("Total SSE = ",
  round(Res2@SSE[length(Res2@SSE)],2),sep=""),ylab="",
  main="MKMeans Clustering Results by the 'initial' method")
points(Res2@Centers,pch=5,col=c("red","blue","green"))

```


Index

- * **classes**

- MKMean, [5](#)

- * **package**

- MKMeans-package, [2](#)

C.f, [3](#)

Dist, [4](#)

MKMean, [5](#)

MKMean-class (MKMean), [5](#)

MKMeans, [6](#)

MKMeans-package, [2](#)