Package ‘FPDclustering’

April 12, 2019

Type Package
Title PD-Clustering and Factor PD-Clustering
Version 1.3
Date 2019-04-22
Author Cristina Tortora [aut, cre, cph], Noe Vidales [aut], and Paul D. McNicholas [fnd]
Maintainer Cristina Tortora <grikris1@gmail.com>
Description Probabilistic distance clustering (PD-clustering) is an iterative, distribution free, probabilistic clustering method. PD-clustering assigns units to a cluster according to their probability of membership, under the constraint that the product of the probability and the distance of each point to any cluster centre is a constant. PD-clustering is a flexible method that can be used with non-spherical clusters, outliers, or noisy data. PDQ is an extension of the algorithm for clusters of different size. Factor PD-clustering (FPDC) is a recently proposed factor clustering method that involves a linear transformation of variables and a cluster optimizing the PD-clustering criterion. It works on high dimensional datasets.
Depends ThreeWay , mvtnorm , R (>= 3.5)
Imports ExPosition, cluster
License GPL (>= 2)
NeedsCompilation no
Repository CRAN
Date/Publication 2019-04-12 10:03:30 UTC

R topics documented:

asymmetric20 ................................................................. 2
asymmetric3 ................................................................. 2
FPDC ................................................................. 3
outliers ................................................................. 5
PDclust ................................................................. 5
PDQ ................................................................. 6
Silh ................................................................. 8
TuckerFactors ................................................................. 9

Index 11
asymmetric20  Asymmetric data set shape=20

Description
Each cluster has been generated according to a multivariate asymmetric Gaussian distribution, with shape=20, covariance matrix equal to the identity matrix and randomly generated centres.

Usage
data(asymmetric20)

Format
A data frame with 800 observations on the following 101 variables. The first variable is the membership.

Source
Generated with R using the package sn (The skew-normal and skew-t distributions), function rsn

Examples
data(asymmetric20)
plot(asymmetric20[,2:3])

asymmetric3  Asymmetric data set shape=3

Description
Each cluster has been generated according to a multivariate asymmetric Gaussian distribution, with shape=3, covariance matrix equal to the identity matrix and randomly generated centres.

Usage
data(asymmetric3)

Format
A data frame with 800 observations on 101 variables. The first variable is the membership labels.

Source
Generated with R using the package sn (The skew-normal and skew-t distributions), function rsn
**Examples**

```r
data(asymmetric3)
plot(asymmetric3[,2:3])
```

---

**FPDC**

*Factor probabilistic distance clustering*

---

**Description**

An implementation of FPDC, a probabilistic factor clustering algorithm that involves a linear transformation of variables and a cluster optimizing the PD-clustering criterion.

**Usage**

```r
FPDC(data = NULL, k = 2, nf = 2, nu = 2)
```

**Arguments**

- `data`: A matrix or data frame such that rows correspond to observations and columns correspond to variables.
- `k`: A numerical parameter giving the number of clusters.
- `nf`: A numerical parameter giving the number of factors for variables.
- `nu`: A numerical parameter giving the number of factors for units.

**Value**

A list with components:

- `label`: A vector of integers indicating the cluster membership for each unit.
- `centers`: A matrix of cluster centers.
- `probability`: A matrix of probability of each point belonging to each cluster.
- `JDF`: The value of the Joint distance function.
- `iter`: The number of iterations.
- `explained`: The explained variability.

**Author(s)**

Cristina Tortora and Paul D. McNicholas.
References


See Also

PDclust

Examples

```r
## Not run:
# Asymmetric data set clustering example (with shape=3).
data('asymmetric3')
x<-asymmetric3[,1]
fpdas3=FPDC(x,4,3,3)
table(asymmetric3[,1],fpdas3$label)
Silh(fpdas3$probability)

## End(Not run)

## Not run:
# Asymmetric data set clustering example (with shape=20).
data('asymmetric20')
x<-asymmetric20[,1]
fpdas20=FPDC(x,4,3,3)
table(asymmetric20[,1],fpdas20$label)
Silh(fpdas20$probability)

## End(Not run)

## Not run:
# Clustering example with outliers.
data('outliers')
x<-outliers[,1]
fpdout=FPDC(x,4,5,4)
table(outliers[,1],fpdout$label)
Silh(fpdout$probability)

## End(Not run)
```
outliers

Data set with outliers

Description

Each cluster has been generated according to a multivariate Gaussian distribution, with centers \( c \) randomly generated. For each cluster, 20% of uniform distributed outliers have been generated at a distance included in \( \max(x-c) \) and \( \max(x-c)+5 \) form the center.

Usage

data(outliers)

Format

A data frame with 960 observations on the following 101 variables. The first variable corresponds to the membership.

Source

generated with R

Examples

data(outliers)
plot(outliers[,2:3])

PDclust

Probabilistic Distance Clustering

Description

Probabilistic distance clustering (PD-clustering) is an iterative, distribution free, probabilistic clustering method. PD clustering assigns units to a cluster according to their probability of membership, under the constraint that the product of the probability and the distance of each point to any cluster centre is a constant.

Usage

PDclust(data = NULL, k = 2)

Arguments

data A matrix or data frame such that rows correspond to observations and columns correspond to variables.

k A numerical parameter giving the number of clusters
Value

A list with components

label A vector of integers indicating the cluster membership for each unit
centers A matrix of cluster centers
probability A matrix of probability of each point belonging to each cluster
JDF The value of the Joint distance function
iter The number of iterations

Author(s)

Cristina Tortora and Paul D. McNicholas

References


Examples

```r
# Normally generated clusters
c1 = c(+2,+2,2,2)
c2 = c(-2,-2,-2,-2)
c3 = c(-3,3,-3,3)
n=200
x1 = cbind(rnorm(n, c1[1]), rnorm(n, c1[2]), rnorm(n, c1[3]), rnorm(n, c1[4]) )
x2 = cbind(rnorm(n, c2[1]), rnorm(n, c2[2]), rnorm(n, c2[3]), rnorm(n, c2[4]) )
x3 = cbind(rnorm(n, c3[1]), rnorm(n, c3[2]), rnorm(n, c3[3]), rnorm(n, c3[4]) )
x = rbind(x1,x2,x3)
pdn=PDclust(x,3)
plot(x[,1:2],col=pdn$label)
plot(x[,3:4],col=pdn$label)
```

PDQ

Probabilistic Clustering Adjusted for Cluster Size

Description

An implementation of PDQ, a probabilistic distance clustering algorithm that involves optimizing the PD-clustering criterion with the option of Euclidean and Chi as dissimilarity measurements.

Usage

```r
PDQ(data=NULL,K=2,method="random", distance="euc", cent=NULL)
```
Arguments

data A matrix or data frame such that rows correspond to observations and columns correspond to variables.
K A numerical parameter giving the number of clusters
method A parameter that selects center starts. Options available are random, kmedoid, and center(user inputs center starts)
distance A parameter that selects the distance measure used. Options available are Euclidean euc and chi square chi
cent User inputed centers if method selected is "random"

Value

A list with components

label A vector of integers indicating the cluster membership for each unit
centers A matrix of cluster centers
probability A matrix of probability of each point belonging to each cluster
JDF The value of the Joint distance function
iter The number of iterations
jdfvector collection of all jdf calculations at each iteration

Author(s)

Cristina Tortora and Noe Vidales

References


See Also

PDclust

Examples

# Gaussian Generated Data no overlap
x<-rmvnorm(100, mean=c(1,5,10), sigma=diag(1,3))
y<-rmvnorm(100, mean=c(4,8,13), sigma=diag(1,3))
data<-rbind(x,y)
pdq1=PDQ(data,2,method="random",distance="euc")
table(rep(c(2,1),each=100),pdq1$label)
Silh(pdq1$probability)
# Gaussian Generated Data with overlap
x2<-rmvnorm(100, mean=c(1,5,10), sigma=diag(1,3))
y2<-rmvnorm(100, mean=c(2,6,11), sigma=diag(1,3))
data2<-rbind(x2,y2)

pdq2<-PDQ(data2,2,method="random",distance="euc")
table(rep(c(1,2),each=100),pdq2$label)
Silh(pdq2$probability)

---

**Silh**

**Probabilistic silhouette plot**

**Description**

Graphical tool to see how well each point belongs to the cluster.

**Usage**

Silh(p)

**Arguments**

p

A matrix of probabilities such that rows correspond to observations and columns correspond to clusters.

**Details**

The probabilistic silhouettes are an adaptation of the ones proposed by Menardi(2011) according to the following formula:

$$\text{dbs}_k = \frac{(\log(p_{ikm_k} / p_{im_1}))}{\max_i[\log(p_{ikm_k} / p_{im_1})]}$$

where $m_k$ is such that $x_i$ belongs to cluster $k$ and $m_1$ is such that $p_{im_1}$ is maximum for $m$ different from $m_k$.

**Value**

Probabilistic silhouette plot

**Author(s)**

Cristina Tortora

**References**

TuckerFactors

Examples

```r
## Not run:
# Asymmetric data set silhouette example (with shape=3).
data('asymmetric3')
x<-asymmetric3[,1]
fpdas3=FPDC(x,4,3,3)
Silh(fpdas3$probability)

## End(Not run)

## Not run:
# Asymmetric data set shiluette example (with shape=20).
data('asymmetric20')
x<-asymmetric20[,1]
fpdas20=FPDC(x,4,3,3)
Silh(fpdas20$probability)

## End(Not run)

## Not run:
# Shiluette example with outliers.
data('outliers')
x<-outliers[,1]
fpdout=FPDC(x,4,3,3)
Silh(fpdout$probability)

## End(Not run)
```

TuckerFactors   Choice of the number of Tucker 3 factors

Description

An empirical way of choosing the number of factors. The algorithm returns a graph and a table representing the explained variability varying the number of factors.

Usage

TuckerFactors(data = NULL, nc = 2)

Arguments

data    A matrix or data frame such that rows correspond to observations and columns correspond to variables.

nc      A numerical parameter giving the number of clusters

Value

A table containing the explained variability varying the number of factors for units (column) and for variables (row) and a plot
Author(s)

Cristina Tortora

References


See Also

T3

Examples

```r
## Not run:
# Asymmetric data set example (with shape=3).
data('asymmetric3')
xp=TuckerFactors(asymmetric3[,-1], nc = 4)

## End(Not run)

## Not run:
# Asymmetric data set example (with shape=20).
data('asymmetric20')
xp=TuckerFactors(asymmetric20[,-1], nc = 4)

## End(Not run)
```
Index

asymmetric20, 2
asymmetric3, 2

FPDC, 3

outliers, 5

PDclust, 4, 5, 7
PDQ, 6

Silh, 8

T3, 10
TuckerFactors, 9