Package 'advclust'

September 4, 2016

Type Package

Title Object Oriented Advanced Clustering

Version 0.4

Date 2016-09-03

Author Achmad Fauzi Bagus F, Setia Pramana

Maintainer Achmad Fauzi Bagus F <fauzi.pandya@gmail.com>

Description

S4 Object Oriented for Advanced Fuzzy Clustering and Fuzzy COnsensus Clustering. Techniques that provided by this package are Fuzzy C-Means, Gustafson Kessel (Babuska Version), Gath-Geva, Sum Voting Consensus, Product Voting Consensus, and Borda Voting Consensus. This package also provide visualization via Biplot and Radar Plot.

License GPL-2

Depends R (>= 3.2.5)

Imports knitr, methods, MASS, clue, ggplot2, reshape2

LazyData TRUE

RoxygenNote 5.0.1

Collate 'as_Membership.R' 'biplot.R' 'c_fuzzycluster.R' 'class_membership.R' 'class_co_fuzzycluster.R' 'class_fuzzycluster.R' 'class_fuzzycluster_list.R' 'class_validation.R' 'co.vote.R' 'cpair_fuzzy.R' 'fuzzy.CM.R' 'fuzzy.GG.R' 'fuzzy.GK.R' 'is_Membership.R' 'membership.R' 'method_co_fuzzycluster.R' 'method_fuzzycluster.R' 'method_fuzzycluster_list.R' 'method_membership.R' 'method_validation.R' 'minWeightBipartiteMatching.R' 'print_Membership.R' 'print_co_fuzzycluster.R' 'print_fuzzycluster.R' 'print_validation.R' 'radarplot.R' 'validation.R' 'zzz.R'

Suggests rmarkdown

VignetteBuilder knitr

NeedsCompilation no

Repository CRAN

Date/Publication 2016-09-04 13:48:57

R topics documented:

`as.membership` *As Membership/Membership Matrix*

Description

Convert matrix to membership matrix

Usage

```
as.membership(member)
```

Arguments

`member`	membership matrix

Value

Membership Membership object

Slots

`member` membership matrix

`hard.label` hard label

`biploting` *Biploting Fuzzy Cluster Result*

Description

Biploting Fuzzy Cluster Result

Usage

```
biploting(object, data.X, scale)
```

Arguments

`object`	a cluster object
`data.X`	a data matrix that used for clustering
`scale`	scaling option (T/F)

Details

Make Visualization Biplot from fuzzy cluster / consensus fuzzy cluster analysis result

Examples

```
fuzzy.CM(iris[,1:4],K=3,m=2,max.iteration=100,threshold=1e-5,RandomNumber=1234)->cl1
biploting(cl1,iris[,1:4])
```

co.vote *Consensus / Ensemble Fuzzy Clustering: Voting*

Description

Provide consensus / ensemble fuzzy clustering with voting method. Several option for voting step provided

Usage

```
co.vote(object, method)
```

Arguments

object	a fuzzycluster_list object
method	voting step that used to combine the partition ("sum","borda","product")

Details

Consensus clustering is method for combine several result of clustering into one robust result. This method used to overcome unstability of cluster result.

This function perform consensus clustering with voting approach. Voting approach look the domination of membership with several algorithm like sum rule, product rule and borda rule.

The differences of that method are how to combine several membership. Sum rule use Sum operation. Product rule use Product operation, and Borda use Borda count algorithm.

Value

Fuzzy Consensus Object

Slots

`member` membership matrix

`hard.label` hard.label

`method.consensus` method of consensus

References

Sevillano, X., Alias, F., & Socoro, J. C. (2013). Posisional and Confidence voting-based Consensus Function For Fuzzy Cluster Ensemble.

Examples

```
fuzzy.CM(iris[,1:4],K=2,m=2,max.iteration=20,threshold=1e-3,RandomNumber=1234)->cl1
fuzzy.GK(iris[,1:4],K=2,m=2,max.iteration=20,threshold=1e-3,RandomNumber=1234)->cl2
fuzzy.CM(iris[,1:4],K=2,m=2,max.iteration=20,threshold=1e-3,RandomNumber=1234)->cl3
c_fuzzycluster(cl1,cl2,cl3)->CL
co.vote(CL,"borda")
```

co_fuzzycluster-class *Consensus Fuzzy Cluster Result*

Description

Consensus Fuzzy Cluster Result

Slots

`member` membership matrix

`hard.label` vector of hard partition

`method.consensus` method of fuzzy clustering used

c_fuzzycluster *Combine fuzzy cluster result*

Description

combining fuzzy cluster result before ensembling

Usage

```
c_fuzzycluster(x, ...)
```

Arguments

`x`	a fuzzy cluster object
`...`	a fuzzy cluster object

Value

Fuzzy Clustering List

Slots

`pair` pair list

Examples

```
fuzzy.CM(iris[,1:4],K=2,m=2,max.iteration=20,threshold=1e-3,RandomNumber=1234)->cl1
fuzzy.GK(iris[,1:4],K=2,m=2,max.iteration=20,threshold=1e-3,RandomNumber=1234)->cl2
c_fuzzycluster(cl1,cl2)
```

`fuzzy.CM` *Fuzzy C-Means*

Description

Fuzzy C-Means clustering Algorithm (Bezdek, 1984)

Usage

```
fuzzy.CM(X, K, m, max.iteration, threshold, member.init, RandomNumber = 0,
  print.result = 0)
```

Arguments

`X`	dataset (matrix/data frame)
`K`	number of cluster
`m`	fuzzyfier
`max.iteration`	maximum iteration for convergence
`threshold`	convergence criteria
`member.init`	membership object or matrix that will be used for initialized
`RandomNumber`	random number for start initializing
`print.result`	print result (9/1)

Details

This function perform Fuzzy C-Means algorithm by Bezdek (1984). Fuzzy C-Means is one of fuzzy clustering methods to clustering dataset become K cluster. Number of cluster (K) must be greater than 1. To control the overlaping or fuzziness of clustering, parameter m must be specified. Maximum iteration and threshold is specific number for convergencing the cluster. Random Number is number that will be used for seeding to firstly generate fuzzy membership matrix.

Clustering will produce fuzzy membership matrix (U) and fuzzy cluster centroid (V). The greatest value of membership on data point will determine cluster label. Centroid or cluster center can be use to interpret the cluster. Both membership and centroid produced by calculating mathematical distance. Fuzzy C-Means calculate distance with Euclideans norm.

Value

Fuzzy Clustering object

Slots

`centroid` centroid matrix

`distance` distance matrix

`func.obj` function objective

`call.func` called function

`fuzzyfier` fuzzyness parameter

`method.fuzzy` method of fuzzy clustering used

`member` membership matrix

`hard.label` hard.label

References

Balasko, B., Abonyi, J., & Feil, B. (2002). Fuzzy Clustering and Data Analysis Toolbox: For Use with Matlab. Veszprem, Hungary.

Bezdek, J. C., Ehrlich, R., & Full, W. (1984). FCM: The Fuzzy C-Means Clustering Algorithm. Computers and Geosciences Vol 10, 191-203

Examples

```
fuzzy.CM(iris[,1:4],K=3,m=2,max.iteration=100,threshold=1e-5,RandomNumber=1234)
```

`fuzzy.cPair` *Pairing fuzzy cluster*

Description

Pairing fuzzy cluster

Usage

```
fuzzy.cPair(x, y)
```

Arguments

`x`	an fuzzycluster object
`y`	an fuzzycluster object

`fuzzy.GG` *Gath Geva Clustering*

Description

Gath Geva for Fuzzy Clustering

Usage

```
fuzzy.GG(X, K, m, max.iteration, threshold, member.init, RandomNumber = 0,
  print.result = 0)
```

Arguments

`X`	dataset (matrix/data frame)
`K`	number of cluster
`m`	fuzzyfier
`max.iteration`	maximum iteration for convergence
`threshold`	convergence criteria
`member.init`	membership object or matrix that will be used for initialized
`RandomNumber`	random number for start initializing
`print.result`	print result (0/1)

Details

This function perform Gath Geva algorithm by Gath-Geva (1989). Gath Geva is one of fuzzy clustering methods to clustering dataset become K cluster. Number of cluster (K) must be greater than 1. To control the overlaping or fuzziness of clustering, parameter m must be specified. Maximum iteration and threshold is specific number for convergencing the cluster. Random Number is number that will be used for seeding to firstly generate fuzzy membership matrix.

Clustering will produce fuzzy membership matrix (U) and fuzzy cluster centroid (V). The greatest value of membership on data point will determine cluster label. Centroid or cluster center can be use to interpret the cluster. Both membership and centroid produced by calculating mathematical distance. Gath Geva distance with Covariance Cluster and norm distribution assumption

Value

Fuzzy Clustering object

Slots

`centroid`	centroid matrix
`distance`	distance matrix
`func.obj`	function objective

`call.func` called function

`fuzzyfier` fuzzyness parameter

`method.fuzzy` method of fuzzy clustering used

`member` membership matrix

`hard.label` hard.label

References

Gath and A.B. Geva,(1989) Unsupervised Optimal Fuzzy Clustering Balasko, B., Abonyi, J., & Feil, B. (2002). Fuzzy Clustering and Data Analysis Toolbox: For Use with Matlab. Veszprem, Hungary.

Examples

```
fuzzy.GG(iris[,1:4],K=2,m=2,max.iteration=20,threshold=1e-3,RandomNumber=1234)
```

---

fuzzy.GK *Gustafson Kessel Clustering with Babuska Improvisation*

Description

Gustafson Kessel clustering Algorithm that improved by Babuska for estimating covariance cluster (Babuska, 2002)

Usage

```
fuzzy.GK(X, K, m, gamma, rho, max.iteration, threshold, member.init,
  RandomNumber = 0, print.result = 0)
```

Arguments

`X`	dataset (matrix/data frame)
`K`	number of cluster
`m`	fuzzyfier
`gamma`	tuning parameter
`rho`	volume cluster parameter
`max.iteration`	maximum iteration for convergence
`threshold`	convergence criteria
`member.init`	membership object or matrix that will be used for initialized
`RandomNumber`	random number for start initializing
`print.result`	print result (0/1)

Details

This function perform Gustafson Kessel algorithm by Gustafson Kessel (1968) that improved by Babuska et al (2002). Gustafson Kessel (GK) is one of fuzzy clustering methods to clustering dataset become K cluster. Number of cluster (K) must be greater than 1. To control the overlaping or fuzziness of clustering, parameter m must be specified. Maximum iteration and threshold is specific number for convergencing the cluster. Random Number is number that will be used for seeding to firstly generate fuzzy membership matrix.

Clustering will produce fuzzy membership matrix (U) and fuzzy cluster centroid (V). The greatest value of membership on data point will determine cluster label. Centroid or cluster center can be use to interpret the cluster. Both membership and centroid produced by calculating mathematical distance. Gustafson Kessel calculate distance with Covariance Cluster norm distance. So it can be said that cluster will have both sperichal and elipsodial shape of geometry.

Babuska improve the covariance estimation via tuning covariance cluster with covariance of data. Tuning parameter determine proportion of covariance data and covariance cluster that will be used to estimate new covariance cluster. Beside improving via tuning, Basbuka improve the algorithm with decomposition of covariance so it will become non singular matrix.

Value

Fuzzy Clustering object

Slots

`centroid` centroid matrix

`distance` distance matrix

`func.obj` function objective

`call.func` called function

`fuzzyfier` fuzzyness parameter

`method.fuzzy` method of fuzzy clustering used

`member` membership matrix

`hard.label` hard.label

References

Babuska, R., Veen, P. v., & Kaymak, U. (2002). Improved Covarians Estimation for Gustafson Kessel Clustering. IEEE, 1081-1084.

Balasko, B., Abonyi, J., & Feil, B. (2002). Fuzzy Clustering and Data Analysis Toolbox: For Use with Matlab. Veszprem, Hungary.

Gustafson, D. E., & Kessel, W. C. (1978). Fuzzy Clustering With A Fuzzy Covariance Matrix. 761-766.

Examples

```
fuzzy.GK(iris[,1:4],K=2,m=2,max.iteration=20,threshold=1e-3,RandomNumber=1234)
```

fuzzycluster-class *Fuzzy Result*

Description

Fuzzy Result

Slots

`centroid` centroid matrix

`distance` distance matrix

`func.obj` function objective

`call.func` called function

`fuzzyfier` fuzzyness parameter

`method.fuzzy` method of fuzzy clustering used

`member` membership matrix

`hard.label` hard.label

fuzzycluster_list-class *Fuzzy Result List*

Description

Fuzzy Result List

Slots

`pair` list of fuzzy cluster object

is.membership *Check membership Matrix/Object*

Description

checking object is membership object or not

Usage

```
is.membership(object)
```

Arguments

object	an object that used for membership checking

Value

T/F

is.na,co_fuzzycluster-method *Method for co_fuzzycluster classes*

Description

Method for co_fuzzycluster classes

Usage

```
## S4 method for signature 'co_fuzzycluster'
is.na(x)

## S4 method for signature 'co_fuzzycluster'
show(object)

method.consensus(x)

## S4 method for signature 'co_fuzzycluster'
method.consensus(x)
```

Arguments

x	an co_fuzzycluster object
object	an co_fuzzycluster object

`is.na,fuzzycluster-method` *Method for fuzzycluster classes*

Description

Method for fuzzycluster classes

Usage

```
## S4 method for signature 'fuzzycluster'
is.na(x)

## S4 method for signature 'fuzzycluster'
show(object)

centroid(x)

## S4 method for signature 'fuzzycluster'
centroid(x)

distance(x)

## S4 method for signature 'fuzzycluster'
distance(x)

func.obj(x)

## S4 method for signature 'fuzzycluster'
func.obj(x)

call.func(x)

## S4 method for signature 'fuzzycluster'
call.func(x)

fuzzyfier(x)

## S4 method for signature 'fuzzycluster'
fuzzyfier(x)

method.fuzzy(x)

## S4 method for signature 'fuzzycluster'
method.fuzzy(x)

.cPair(x, y)
```

```
## S4 method for signature 'fuzzycluster,fuzzycluster'
.cPair(x, y)

## S4 method for signature 'fuzzycluster,ANY'
.cPair(x, y)

## S4 method for signature 'ANY,fuzzycluster'
.cPair(x, y)

## S4 method for signature 'ANY,ANY'
.cPair(x, y)
```

Arguments

x	an fuzzycluster object
object	an fuzzycluster object
y	an fuzzycluster object

`is.na,membership-method` *Method for membership classes*

Description

Method for membership classes

Usage

```
## S4 method for signature 'membership'
is.na(x)

## S4 method for signature 'membership'
show(object)

member(x)

## S4 method for signature 'membership'
member(x)

hard.label(x)

## S4 method for signature 'membership'
hard.label(x)
```

Arguments

x	an object
object	an object

`membership` *Make an Partition Matrix*

Description

Make an Partition Matrix

Usage

```
membership(member, K, n, RandomNumber = 0)
```

Arguments

member	membership matrix
K	number of cluster
n	number of observation
RandomNumber	random number/seed used

Value

membership object

Slots

`member` membership object

`hard.label` hard label

Examples

```
#make an matrix membership
membership(K=3,n=20,RandomNumber=1234)
```

`membership-class` *Membership*

Description

Membership

Slots

`member` membership matrix

`hard.label` vector of hard hard.labeling

minWeightBipartiteMatching *Matching label*

Description

Matching label of 2 cluster via Hungary Algorithm

Usage

```
minWeightBipartiteMatching(x, y)
```

Arguments

x	an fuzzycluster object
y	an fuzzy cluster object

pair *Method for fuzzycluster_list classes*

Description

Method for fuzzycluster_list classes

Usage

```
pair(object)

## S4 method for signature 'fuzzycluster_list'
pair(object)
```

Arguments

object	an fuzzycluster_list object

print.co_fuzzycluster *Print Consensus Fuzzy Clustering Result*

Description

Print Consensus Fuzzy Clustering

Usage

```
## S3 method for class 'co_fuzzycluster'
print(x, ...)
```

Arguments

x	consensus fuzzy clustering object
...	another paramater

print.fuzzycluster *Print Fuzzy Clustering Result*

Description

Print Fuzzy Clustering

Usage

```
## S3 method for class 'fuzzycluster'
print(x, ...)
```

Arguments

x	fuzzy clustering object
...	another paramater

print.membership *Print membership*

Description

Print membership object

Usage

```
## S3 method for class 'membership'
print(x, ...)
```

Arguments

x	membership object
...	another parameter

print.validation *Print Validation Index*

Description

Print Validation Index for fuzzy clustering

Usage

```
## S3 method for class 'validation'
print(x, ...)
```

Arguments

x	validation object
...	another parameter

radar.plotting *Radar Ploting Fuzzy Cluster Result*

Description

Radar Ploting Fuzzy Cluster Result

Usage

```
radar.plotting(object, data.X)
```

Arguments

object	a fuzzycluster object
data.X	a matrix data

Details

Make Visualization Radar Ploting

Examples

```
fuzzy.CM(iris[,1:4],K=3,m=2,max.iteration=100,threshold=1e-5,RandomNumber=1234)->cl
radar.plotting(cl,iris[,1:4])
```

show,validation-method *Method for validation classes*

Description

Method for validation classes

Usage

```
## S4 method for signature 'validation'
show(object)

PC(x)

## S4 method for signature 'validation'
PC(x)

MPC(x)

## S4 method for signature 'validation'
```

```
MPC(x)

CE(x)

## S4 method for signature 'validation'
CE(x)

XB(x)

## S4 method for signature 'validation'
XB(x)

S(x)

## S4 method for signature 'validation'
S(x)

Tang(x)

## S4 method for signature 'validation'
Tang(x)

Kwon(x)

## S4 method for signature 'validation'
Kwon(x)
```

Arguments

object	an object
x	an object

`validation-class` *Validation Class*

Description

Validation Class

Slots

`PC` Partition Coefficient index

`MPC` Modified Partition Coefficient index

`CE` Classification Entropy index

`S` Separation index

`XB` Xie Beni index

`Kwon` Kwon index

`Tang` Tang index

`validation.index` *Validation Index*

Description

Validation index for validating fuzzy clustering result

Usage

```
validation.index(object)
```

Arguments

`object`	fuzzy clustering object

Details

This function provide several validation indexs that calculated from fuzzy clustering result. Validation index can be used for choose best optimum parameter.

There are PC, MPC, CE, S, Xie Beni, Kwon, and Tang index. PC (Partition Coefficient), MPC (Modified Partition Coefficient), and CE (Classification Entropy) are calculated from membership matrix. S (Separation Index), Xie Beni, Kwon, and Tang use both distance and membership matrix.

The best cluster result can be decided with minimum value of index, except MPC and PC use maximum value.

Value

validation index object.

Slots

`XB` Xie Beni Index

`PC` Partition Coef.

`MPC` Modifief Partition Coef.

`Kwon` Kwon Index

`Tang` Tang Index

`S` Separation Index

`CE` Classification Entropy

Author(s)

Achmad Fauzi Bagus F

References

Wang, W., & Zhang, Y. (2007). On Fuzzy Cluster Validity Indices. Fuzzy Sets and System, 2095-2117.

Examples

```
fuzzy.CM(iris[,1:4],K=3,m=2,max.iteration=100,threshold=1e-5,RandomNumber=1234)->cl
validation.index(cl)->valid
#example for Xie Beni index
XB(valid)
```

Index