

Package ‘oasw’

March 19, 2020

Type Package

Title Optimum average silhouette width clustering methods

Version 1.0

Date 2020-03-19

Author Fatima Batool

Maintainer Fatima Batool <ucakfba@ucl.ac.uk>

Description The package implements the hierarchical and partitional clustering methods based on the optimization of the average silhouette width index.

License GPL (>= 2)

Encoding UTF-8

Imports Rcpp (>= 1.0.3), ggplot2, MASS, sn, cluster, nnet, GGally, mclust

LinkingTo Rcpp

RoxygenNote 7.0.2

R topics documented:

| | |
|----------------------------|----|
| oasw-package | 2 |
| C14D2 | 3 |
| C7D10 | 4 |
| CCo4D2 | 5 |
| Co5d5 | 6 |
| ElongUniGaussian | 7 |
| FarUniGauss | 8 |
| fossil | 9 |
| fossilFix | 10 |
| Gaussian3 | 11 |
| Gaussian4 | 12 |
| hosil | 13 |
| init | 14 |
| LongUniGauss | 16 |
| NoisyGaussian | 17 |
| osil | 18 |
| osilFix | 19 |
| pairplots | 20 |
| pamsil | 21 |

| | |
|--------------------------------|----|
| pamsilFix | 22 |
| plot2d | 23 |
| ShortUniGauss | 24 |
| TChiGaussianF | 25 |
| TenNest | 26 |
| ThreeMicroarray | 27 |
| ThreeMicroarrayMulti | 28 |
| TNestGaussian | 29 |
| TwoGaussian | 30 |
| TwoGaussianT | 31 |
| UniGauss | 32 |
| UnitGaussian3D | 33 |

Index 34

| | |
|--------------|--|
| oasw-package | <i>Optimum average silhouette width clustering methods</i> |
|--------------|--|

Description

The package implements the hierarchical and partitional clustering methods based on the optimization of the average silhouette width index. The following are the major functions of the package.

Clustering algorithms

pamsil Computes PAMSIL clustering as introduced in Van der Laan (2003).

osil Computes osil clustering as introduced in Batool (2019).

fossil Computes fossil clustering as introduced in Batool (2019).

hosil Computes hosil clustering as introduced in Batool (2019).

Author(s)

Fatima Batool

Maintainer: Fatima Batool <fatima.batool.14@ucl.ac.uk>

References

- Batool F. (2019). Optimum average silhouette width clustering. *PhD Thesis*, University College London.
- Batool, F., (2019). A new hierarchical clustering algorithm based on optimization of ASW linkage criterion. *Journal of classification*. <https://arxiv.org/abs/1909.12356>.
- Batool, F., and Hennig, C. (2019). Characterization and Development of Average Silhouette Width Clustering. *Advanced in data analysis and classification*. <https://arxiv.org/abs/1910.11339>.
- Batool, F., and Hennig, C. (2019). Initializations and related challenges for clustering by optimizing the average silhouette width. <https://arxiv.org/abs/1910.08644>.
- Van der Laan, M., Pollard, K., & Bryan, J. (2003). A new partitioning around medoids algorithm. *Journal of Statistical Computation and Simulation*, 73(8), 575-584.

See Also

[kmeans](#) [pam](#) [hclust](#) [Mclust](#)

C14D2

*A data generating model with 14-clusters***Description**

Generates a data set consists of 14-clusters, one each from the specified Gaussian distributions.

Usage

```
C14D2(n)
```

Arguments

n number of observations in the dataset. These are equally divided between clusters.

Details

The data set has two iid dimensions.

Value

Returns a list having two components described as follows;

n number of observations.

K number of clusters.

cluster size number of observations in each cluster.

data data set generated from the model.

truelab clustering label vector corresponding to the known data generating model.

Author(s)

Fatima Batool <ucakfba@ucl.ac.uk>

References

Batool F. (2019). Optimum average silhouette width clustering. *PhD Thesis*, University College London.

Examples

```
dmat <- C14D2(350)
plot2d(dmat$data, dmat$truelab)
```

C7D10*A data generating model with Seven clusters*

Description

Seven clusters in multiple dimensions having unequal within cluster variations. All the clusters are generated from Gaussain distributions.

Usage

C7D10(n)

Arguments

n number of observations in the dataset. These are equally divided between clusters.

Details

The data set has ten dimensions.

Value

Returns a list having two components described as follows;

n number of observations.

K number of clusters.

cluster size number of observations in each cluster.

data data set generated from the model.

truelab clustering label vector correspounding to the known data generating model.

Author(s)

Fatima Batool <ucakfba@ucl.ac.uk>

References

Batool F. (2019). Optimum average silhouette width clustering. *PhD Thesis*, University College London.

Examples

```
dmat <- C7D10(700)
plot2d(dmat$data, dmat$truelab)
pairplots(dmat$data, dmat$truelab)
```

CCo4D2*A data generating model with four clusters*

Description

Generates a data set consists of 4-clusters, one each from the specified Gaussian distributions.

Usage

```
CCo4D2(n)
```

Arguments

n number of observations in the dataset. These are equally divided between clusters.

Details

The data set has two iid dimensions and K=4 clusters.

Value

Returns a list having two components described as follows;

n number of observations.

K number of clusters.

cluster size number of observations in each cluster.

data data set generated from the model.

truelab clustering label vector corresponding to the known data generating model.

Author(s)

Fatima Batool <ucakfba@ucl.ac.uk>

References

Batool F. (2019). Optimum average silhouette width clustering. *PhD Thesis*, University College London.

Examples

```
dmat <- CCo4D2(1000)
plot2d(dmat$data, dmat$truelab)
```

Co5d5

A data generating model with two clusters

Description

Generates data set consists of five correlated clusters, one each from Gaussian distributions.

Usage

Co5d5(n)

Arguments

n number of observations in the dataset. These are equally divided between clusters.

Details

The data set has five correlated dimensions.

Value

Returns a list having two components described as follows;

n number of observations.

K number of clusters.

cluster size number of observations in each cluster.

data data set generated from the model.

truelab clustering label vector corresponding to the known data generating model.

Author(s)

Fatima Batool <ucakfba@ucl.ac.uk>

References

Batool F. (2019). Optimum average silhouette width clustering. *PhD Thesis*, University College London.

Examples

```
dmat <- Co5d5(1000)
plot2d(dmat$data, dmat$truelab)
pairplots(dmat$data, dmat$truelab)
```

ElongUniGaussian*A data generating model with 9-clusters*

Description

Generates data set consists of 9-clusters, from Gaussian and Uniform distributions. See details for complete model definition.

Usage

```
ElongUniGaussian(n)
```

Arguments

n number of observations in the dataset. These are equally divided between clusters.

Details

The data set has two dimensions.

Value

Returns a list having two components described as follows;

n number of observations.

K number of clusters.

cluster size number of observations in each cluster.

data data set generated from the model.

truelab clustering label vector corresponding to the known data generating model.

Author(s)

Fatima Batool <ucakfba@ucl.ac.uk>

References

Batool F. (2019). Optimum average silhouette width clustering. *PhD Thesis*, University College London.

Examples

```
dmat <- ElongUniGaussian(900)
plot2d(dmat$data, dmat$truelab)
```

FarUniGauss*A data generating model with two clusters*

Description

Generates a data set consists of 2-clusters, one each from the Gaussian and Uniform distributions.

Usage

```
FarUniGauss(n)
```

Arguments

n number of observations in the dataset. These are equally divided between clusters.

Details

The data set has two iid dimensions.

Value

Returns a list having two components described as follows;

n number of observations.

K number of clusters.

cluster size number of observations in each cluster.

data data set generated from the model.

truelab clustering label vector corresponding to the known data generating model.

Author(s)

Fatima Batool <ucakfba@ucl.ac.uk>

References

Batool F. (2019). Optimum average silhouette width clustering. *PhD Thesis*, University College London.

Examples

```
dmat <- FarUniGauss(1000)
plot2d(dmat$data, dmat$truelab)
plot(dmat$data, col=dmat$truelab, xlim=c(-20, 20), ylim=c(-20, 20))
```


fossil

*Fast osil-estimation of number of clusters***Description**

This is the fast version of the OSil algorithm. OSil is an optimum average silhouette width (OASW) clustering method that donot make use of any kind of cluster centriods. Only data is needed as input. The algorithm can estimate number of clusters.

Usage

```
fossil(dmat, distmethod="euclidean", kmin=2, kmax=12)
```

Arguments

| | |
|-------------------|---|
| dmat | Either a numeric matrix or data frame of observed values. The row represent observations to cluster and columnn represents the variables. |
| distmethod | the distance method to be used. Current available methods are "euclidean", "maximum", "manhattan", "canberra", "binary" or "minkowski". See dist for more details on these methods. |
| kmin | minimum number of clusters |
| kmax | maximum value to be used for the estimation of number of clusters. |

Details

osil has an initialization phase. Based on extensive simulaitons the best initialization methods from among a wide range of existing clustering methods, for the algorithm has been identified namely average, Ward's, pam, kmeans, and model-based clustering. In case distances are provided for clustering kmeans and model-based clustering are excluded from the initialization methods.

Value

Returns a list having following components:

- n** total number of data points.
- K** estimated number of clusters.
- clus_lab** clustering label vector.
- clus_size** cluster sizes.
- silh** OASW value of each cluster.
- avg_clus_silh** OASW for each cluster.
- avg_silh** OASW value for the clustering.
- \$avg_silh_kmin_kmax** ASW value for all the clusterings.

Author(s)

Fatima Batool <ucakfba@ucl.ac.uk>

References

Batool F. (2019). Optimum average silhouette width clustering. *PhD Thesis*, University College London.

See Also

[osil](#) for not so fast version.

Examples

```
require(mlbench)
dmat <- mlbench.shapes(100)$x
oasw_clus <- fossil(dmat)
oasw_clus <- fossil(dmat, distmethod="euclidean", kmin=2, kmax=5)
plot(dmat, col = oasw_clus$clus_lab, pch = 16, cex = 1.5)
```

fossilFix

Fast osil-fixed number of clusters

Description

This is the fast version of the OSil algorithm for a single number of cluster. OSil is an optimum average silhouette width (OASW) clustering method that donot use any kind of cluster centriods. Only data is needed as input. The algorithm can estimate number of clusters.

Usage

```
fossilFix(dmat, distmethod="euclidean", k)
```

Arguments

| | |
|------------|---|
| dmat | Either a numeric matrix or data frame of observed values. The row represent observations to cluster and columnn represents the variables. |
| distmethod | the distance method to be used. Current available methods are "euclidean", "maximum", "manhattan", "canberra", "binary" or "minkowski". See dist for more details on these methods. |
| k | number of clusters |

Details

osil has an initialization phase. Based on extensive simulaitons the best initialization methods from among a wide range of existing clustering methods, for the algorithm has been identified namely average, Ward's, pam, kmeans, and model-based clustering. In case distances are provided for clustering kmeans and model-based clustering are excluded from the initialization methods.

Value

Returns a list having following components:

n total number of data points.

K estimated number of clusters.

clus_lab clustering label vector.

clus_size cluster sizes.

silh OASW value of each cluster.

avg_clus_silh OASW for each cluster.

avg_silh OASW value for the clustering.

\$avg_silh_kmin_kmax ASW value for all the clusterings.

Author(s)

Fatima Batool <ucakfba@ucl.ac.uk>

References

Batool F. (2019). Optimum average silhouette width clustering. *PhD Thesis*, University College London.

See Also

[osil](#) for not so fast version.

Examples

```
require(mlbench)
dmat <- mlbench.shapes(100)$x
oasw_clus <- fosil(dmat)
k <- 4
oasw_clus <- fosilFix(dmat, distmethod="euclidean", k)
plot(dmat, col = oasw_clus$clus_lab, pch = 16, cex = 1.5)
```

Gaussian3

A data generating model with three clusters

Description

Generates data set consists of three clusters, one each from Gaussian distributions. The clusters are of unequal within cluster variations.

Usage

```
Gaussian3(n)
```

Arguments

n number of observations in the dataset. These are equally divided between clusters.

Details

The data set has two iid dimensions.

Value

Returns a list having two components described as follows;

n number of observations.

K number of clusters.

cluster size number of observations in each cluster.

data data set generated from the model.

truelab clustering label vector corresponding to the known data generating model.

Author(s)

Fatima Batool <ucakfba@ucl.ac.uk>

References

Batool F. (2019). Optimum average silhouette width clustering. *PhD Thesis*, University College London.

Examples

```
dmat <- Gaussian3(1000)
plot2d(dmat$data, dmat$truelab)
```

Gaussian4

A data generating model with four clusters

Description

Generates data set consists of three clusters, one each from Gaussian distributions.

Usage

```
Gaussian4(n)
```

Arguments

n number of observations in the dataset. These are equally divided between clusters.

Details

The data set has two dimensions. The dimensions are independently drawn from each other.

Value

Returns a list having two components described as follows;

n number of observations.

K number of clusters.

cluster size number of observations in each cluster.

data data set generated from the model.

truelab clustering label vector corresponding to the known data generating model.

Author(s)

Fatima Batool <ucakfba@ucl.ac.uk>

References

Batool F. (2019). Optimum average silhouette width clustering. *PhD Thesis*, University College London.

Examples

```
dmat <- Gaussian4(1000)
plot2d(dmat$data, dmat$truelab)
```

hosil

Hierarchical optimum average silhouette width clustering

Description

hosil is a clustering algorithmic hierarchical clustering algorithm. The cluster merge are defined at each hierarchy using a new linkage method defined by the optimization of the ASW index. The method can also estimate the number of clusters based on OASW linkage criterion. If number of clusters is to be fixed (see fixK) the minimum allowed is 3 and can be at most (n-1).

Usage

```
hosil(dys, distmethod = "euclidean", fixK = "NA")
```

Arguments

| | |
|------------|--|
| dys | A vector of pairwise distances between observations. Usually an object of class "dist" or a data matrix or data frame. In latter case also needs distance method. The default is set at euclidean. |
| distmethod | the distance method to be used. Current available methods are euclidean, maximum, manhattan, canberra, binary, or minkowski. See dist for more details on these methods. |
| fixK | user defined number of clusters against which clustering is required. |

Value

Returns a list with the following components

est_K estimated number of clusters.

OASW_est_K Value of the objective function against the estimated number of clusters.

clus_vect_est clustering label vector for the estimated number of clusters.

all_OASW Values of the objective function (OASW linkage) against $n-1$ to 2 number of clusters. Thus the first value `all_OASW[1]` gives the OASW linkage value against $(n-1)$ number of clusters, the next value contains $(n-2)$ then $(n-3), \dots, 3, 2$.

fix_K user specified number of clusters

clus_vect_fix clustering label vector for the user specified number of clusters

OASW_fix_K Value of the objective function against the user specified number of clusters

Author(s)

Fatima Batool < ucakfba@ucl.ac.uk >

References

Batool F. (2019). Optimum average silhouette width clustering. *PhD Thesis*, University College London.

Kaufman, L. and P. J. Rousseeuw (1990). Finding groups in data: an introduction to cluster analysis, Volume 344. John Wiley & Sons.

Examples

```
dmat <- MultiDist(350)
dys <- dist(dmat$data)
oasw_clustering <- hosil(dys)
oasw_clustering <- hosil(dys, fixK=6)
plot(dmat, col = oasw_clus$clus_vect_fix, pch = 16)
```

init

initialization function for OASW clustering algorithms

Description

This function was originally wrote to provide the initialization for osil and fosil but can have standalone usage. The function takes a fixed number of clusters.

Usage

```
init(dmat, K, distmethod = "euclidean", ...)
```

Arguments

| | |
|------------|--|
| dmat | either a numeric matrix or data frame of observed values or pairwise distances between observations. |
| K | number of cluster |
| distmethod | distance method to be used for the calculation of pairwise distances between observations. |
| ... | additional parameters |

Details

The function is originally written as an initialization function for `osil` and `fossil` clusterings. The `init` functions returns the best clustering out of the six methods initialization methods based on the ASW values. Several clustering methods were considered in a systematic simulation set-up to find out the best for the OASW clustering initialization. Among those considered six showed good performance for single or multiple data generating structures considered in the simulations (see Batool, 2019 for results and discussion). The six best initialization methods are average (Sokal and Michener 1958), Ward's (Ward 1963), `pam` (Kaufman and Rousseeuw 1990), `kmeans` (Hartigan and Wong 1979), and model-based (Fraley and Raftery 1998) clustering. Works with both distances and data matrix. Currently, in case distances are provided `kmeans` and `Mclust` are excluded from initialization methods.

Value

Returns a list having following components

n number of observations.

K number of clusters.

lab_best clustering labels corresponding to the best ASW clustering among the initialization methods.

asw_best best ASW value among the initialization methods.

best_init_method name of the best initialization methods based on ASW value.

Author(s)

Fatima Batool <ucakfba@ucl.ac.uk>

References

- Batool F. (2019). Optimum average silhouette width clustering. *PhD Thesis*, University College London.
- C. Fraley and A. E. Raftery (1998). How many clusters? which clustering method? answers via model-based cluster analysis. *The Computer Journal*, 41(8):578 588.
- Hartigan, J. A. and Wong, M. A. (1979). Algorithm AS 136: A K-means clustering algorithm. *Applied Statistics*, 28, 100 108.
- J. H. Ward Jr. (1963). Hierarchical grouping to optimize an objective function. *Journal of the American Statistical Association*, 58(301):236 244.
- Kaufman, L. and P. J. Rousseeuw (1990). Finding groups in data: an introduction to cluster analysis, Volume 344. John Wiley & Sons.
- McQuitty, L. L. (1957). Elementary linkage analysis for isolating orthogonal and oblique types and typal relevancies. *Educational and Psychological Measurement* 17(2), 207 229.
- R. Sokal and C. D. Michener (1958). A statistical method for evaluating systematic relationships. *Univesity Kansas Science Bulletin*, 38(22):1409 1438.

See Also

`hclust`, `pam`, `kmeans`, `Mclust` functions to pass additional arguments to `init`.

Examples

```
dmat <- TwoGaussian(100)
dys <- dist(dmat$data)
plot(dmat$data, col = dmat$truelab)
init_res_1 <- init(dmat$data, KK=2, distmethod = "euclidean")
init_res_2 <- init(dys, KK=2, distmethod = "euclidean")
print(init_res_1)
```

LongUniGauss

A data generating model with two clusters

Description

Generates a data set consists of 2-clusters, one each from the Gaussian and Uniform distributions.

Usage

```
LongUniGauss(n)
```

Arguments

n number of observations in the dataset. These are equally divided between clusters.

Details

The data set has two iid dimensions.

Value

Returns a list having two components described as follows;

n number of observations.

K number of clusters.

cluster size number of observations in each cluster.

data data set generated from the model.

truelab clustering label vector corresponding to the known data generating model.

Author(s)

Fatima Batool <ucakfba@ucl.ac.uk>

References

Batool F. (2019). Optimum average silhouette width clustering. *PhD Thesis*, University College London.

Examples

```
dmat <- LongUniGauss(1000)
plot2d(dmat$data, dmat$truelab)
plot(dmat$data, col=dmat$truelab, xlim=c(-20, 20), ylim=c(-20, 20))
```

| | |
|---------------|---|
| NoisyGaussian | <i>A data generating model with two clusters and added Uniform noise points</i> |
|---------------|---|

Description

Generates data set consists of two closely located Gaussian clusters with 500 Uniform noise points.

Usage

```
NoisyGaussian(n,noise.points)
```

Arguments

| | |
|---------------------|--|
| n | number of observations in the dataset. These are equally divided between clusters. |
| noise.points | number of noise points required to be added in data set |

Details

The data set has two iid dimensions.

Value

Returns a list having two components described as follows;

n number of observations.

K number of clusters.

cluster size number of observations in each cluster.

data data set generated from the model.

truelab clustering label vector corresponding to the known data generating model.

Author(s)

Fatima Batool <ucakfba@ucl.ac.uk>

References

Batool F. (2019). Optimum average silhouette width clustering. *PhD Thesis*, University College London.

Examples

```
dmat <- NoisyGaussian(200, 50)
plot(dmat$data, col=c("blue","green", "red")[dmat$truelab])
```

osil

*Optimum Average Silhouette Width clustering***Description**

An OASW clustering method that does not use cluster centriods with estimation of number of clusters.

Usage

```
osil(dmat, distmethod = "euclidean", kmin=2, kmax=12)
```

Arguments

| | |
|-------------------|---|
| dmat | either a numeric matrix or data frame of observed values or pairwise distances between observations. In first case the row represent observations to cluster and columnn represents the variables. If data matrix is provided then the distance method can be specfied as well. In second case usually an object of class dist. Missing values are not allowed in both cases. |
| distmethod | the distance mehtod to be used. Current available methods are "euclidean", "maximum", "manhattan", "canberra", "binary" or "minkowski". See dist for more details on these methods. |
| kmin | minimum value to be used for the estimation of number of clusters. |
| kmax | maximum value to be used for the estimation of number of clusters. |

Details

The data given in matrix form is clustered by the newly proposed OSil algorithm (see Batool 2019) based on the optimization of ASW index. osil has an initialization phase. Based on extensive simulaitons the best initialization methods from among a wide range of existing clustering methods, for the algorithm has been identified namely average (Sokal and Michener 1958), Ward's (Ward 1963), pam (Kaufman and Rousseeuw 1990), kmeans (Hartigan and Wong 1979), and model-based (Fraley and Raftery 1998) clustering. In case distances are provided for clustering kmeans and model-based clustering are excluded from the initialization methods.

kmin and kmax define the range for the estimation of number of clusters. If a single valued clustering solution say K is required, specify kmin=K and kmax=K or use [osilFix](#).

Value

Returns a list having following components;

n total number of data points.

K estimated number of clusters.

clus_lab clustering labels.

clus_size number of observations in clusters.

silh ASW value of each cluster.

avg_clus_silh ASW for each cluster.

avg_silh ASW value for the clustering for K clusters.

iter number of iteration taken by the algorithm to converge.

avg_silh_k ASW value for the clustering for kmin to kmax clusterings.

Author(s)

Fatima Batool <ucakfba@ucl.ac.uk>

References

- Batool F. (2019). Optimum average silhouette width clustering. *PhD Thesis*, University College London.
- C. Fraley and A. E. Raftery (1998). How many clusters? which clustering method? answers via model-based cluster analysis. *The Computer Journal*, 41(8):578 588.
- Hartigan, J. A. and Wong, M. A. (1979). Algorithm AS 136: A K-means clustering algorithm. *Applied Statistics*, 28, 100 108.
- J. H. Ward Jr. (1963). Hierarchical grouping to optimize an objective function. *Journal of the American Statistical Association*, 58(301):236 244.
- Kaufman, L. and P. J. Rousseeuw (1990). Finding groups in data: an introduction to cluster analysis, Volume 344. John Wiley & Sons.
- McQuitty, L. L. (1957). Elementary linkage analysis for isolating orthogonal and oblique types and typal relevancies. *Educational and Psychological Measurement* 17(2), 207 229.
- R. Sokal and C. D. Michener (1958). A statistical method for evaluating systematic relationships. *Univesity Kansas Science Bulletin*, 38(22):1409 1438.

Examples

```
dmat <- TwoGaussian(100)$data
oasw_clustering <- osil(dmat)
dys <- dist(dmat)
oasw_clustering <- osil(dys)
plot(dmat, col = oasw_clus$clus_lab, pch = 16, cex = 1.5)
```

osilFix

osil clustering-fixed number of clusters

Description

Produces a clustering solution for a fixed number of clusters

Usage

```
osilFix(dys, n, K, clus_lab)
```

Arguments

| | |
|----------|--|
| dys | pairwise distances between observations. |
| n | number of observations. |
| K | number of clusters. |
| clus_lab | clustering labels. |

Details

This is a wrapper function for C++ functions. This function is called from within osil function. Can be called standalone for clustering for fixed number of clusters with an initialization clustering label set. This will return an OASW clustering based on labels.

Value

Returns a list having following components.

n total number of data points.

K estimated number of clusters.

clus_lab clustering labels.

clus_size cluster sizes.

silh ASW value of each cluster.

avg_clus_silh ASW for each cluster.

avg_silh ASW value for the clustering.

iter number of iteration taken by the algorithm to converge.

Author(s)

Fatima Batool <ucakfba@ucl.ac.uk>

pairplots

pairplots

Description

pair plots for more than two dimensional data

Usage

```
pairplots(data, labels)
```

Arguments

data data set to plot.

labels labels against which plotting is needed.

Details

none

Value

Returns a plot.

Author(s)

Fatima Batool <ucakfba@ucl.ac.uk>

Examples

```
dmat <- C7D10(70)
pairplots(dmat$data, dmat$truelab)
```

pamsil

*pamsil clustering***Description**

An OASW clustering method based on medoids.

Usage

```
pamsil(dmat, distmethod = "euclidean", kmin=2, kmax=12)
```

Arguments

| | |
|------------|---|
| dmat | Either a matrix or data frame of observed values or a vector of pairwise distances between observations. In first case the row represent observations to cluster and columnn represents the variables. If data matrix is provided needs to specify the distance method as well. In second case usually an object of class "dist". Missing values are not allowed in both cases. |
| distmethod | distance method to be used. |
| kmin | minimum number of clusters for estimation of number of clusters. |
| kmax | maximum number of clusters for estimation of number of clusters. |

Details

This function is based on the standalone C functions written by Van et al. (2003). pamsil() scepts both data matrix and distances.

Value

Returns a list having following components:

- est_K** number of clusters estimated by pamsil algorithms. All other results are based on this estimated number.
- clus_lab** pamsil clustering labels against estimated K.
- silh** optimum ASW value for each data point in the clustering.
- clus_size** number of observations in clusters.
- avg_silh** ASW value for pamsil clustering.
- avg_clus_silh** ASW for each cluster.
- iter** number of iteration taken by the algorithm to converge.
- avg_silh_kmin_kmax** ASW values for the clusterings corresponding to the number of clusters in the range kmin to kmax number of clusters.

Author(s)

Fatima Batool <ucakfba@ucl.ac.uk>

References

Van der Laan, M., Pollard, K., & Bryan, J. (2003). A new partitioning around medoids algorithm. *Journal of Statistical Computation and Simulation*, 73(8), 575-584.

Examples

```
dmat <- iris[,1:4]
dys <- dist(dmat)
oasw_clustering <- pamsil(dys, 2, 4)
oasw_clustering <- pamsil(dmat, distmethod = "manhattan", 2, 4)
```

pamsilFix

A new partitioning around medoids algorithm

Description

An OASW clustering method based on medoids. A PAM like clustering algorithm for the optimization of ASW based on medoids proposed in Van et al. (2003).he

Usage

```
pamsilFix(dys, K, distmethod = "euclidean")
```

Arguments

| | |
|------------|---|
| dys | Either a matrix or data frame of observed values or a vector of pairwise distances between observations. In first case the row represent observations to cluster and columnn represents the variables. If data matrix is provided needs to specify the distance method as well. In second case usually an object of class "dist". Missing values are not allowed in both cases. |
| K | number of clusters |
| distmethod | distance method to be used |

Details

This function is based on the standalone C functions written by Van et al. (2003). Use of this function is recommended if number of clusters are known. pamsilFix accepts both data matrix and distances.

Value

Returns a list having following components:

- clus_lab** pamsil clustering labels.
- silh** ASW value for each data point in the clustering.
- clus_size** number of observations in clusters.
- avg_clus_silh** ASW for each cluster.
- avg_silh** ASW value for the clustering.
- iter** number of iteration taken by the algorithm to converge.

Author(s)

Fatima Batool <ucakfba@ucl.ac.uk>

References

Van der Laan, M., Pollard, K., & Bryan, J. (2003). A new partitioning around medoids algorithm. *Journal of Statistical Computation and Simulation*, 73(8), 575-584.

Examples

```
dmat <- iris[,1:4]
dys <- dist(dmat)
oasw_clustering <- pamsilFix(dys, 3)
oasw_clustering <- pamsilFix(dmat, 3, distmethod = "manhattan")
```

plot2d

Enhanced 2d plotting

Description

plots for two dimensional data

Usage

```
plot2d(data, labels)
```

Arguments

| | |
|--------|---|
| data | data to plot |
| labels | labels against which plotting is needed |

Details

none

Value

Returns a plot

Author(s)

Fatima Batool <ucakfba@ucl.ac.uk>

Examples

```
dmat <- C7D10(1000)
plot2d(dmat$data, dmat$truelab)
```

`ShortUniGauss`*A data generating model with two clusters*

Description

Generates a data set consists of 2-clusters, one each from the Gaussian and Uniform distributions.

Usage

```
ShortUniGauss(n)
```

Arguments

n number of observations in the dataset. These are equally divided between clusters.

Details

The data set has two iid dimensions.

Value

Returns a list having two components described as follows;

n number of observations.

K number of clusters.

cluster size number of observations in each cluster.

data data set generated from the model.

truelab clustering label vector corresponding to the known data generating model.

Author(s)

Fatima Batool <ucakfba@ucl.ac.uk>

References

Batool F. (2019). Optimum average silhouette width clustering. *PhD Thesis*, University College London.

Examples

```
dmat <- ShortUniGauss(1000)
plot2d(dmat$data, dmat$truelab)
plot(dmat$data, col=dmat$truelab, xlim=c(-20, 20), ylim=c(-20, 20))
```

TChiGaussianF*A data generating model with five clusters*

Description

Generates data set consists of five clusters, one each from Gaussian, Student's t, Chi-squared, skew Gaussian, and F distributions.

Usage

```
TChiGaussianF(n)
```

Arguments

n number of observations in the dataset. These are equally divided between clusters.

Details

The data set has two iid dimensions.

Value

Returns a list having two components described as follows;

n number of observations.

K number of clusters.

cluster size number of observations in each cluster.

data data set generated from the model.

truelab clustering label vector corresponding to the known data generating model.

Author(s)

Fatima Batool <ucakfba@ucl.ac.uk>

References

Batool F. (2019). Optimum average silhouette width clustering. *PhD Thesis*, University College London.

Examples

```
dmat <- TChiGaussianF(500)
plot2d(dmat$data, dmat$truelab)
```

TenNest

A data generating model with ten nested clusters

Description

Generates data set consists of ten clusters, one each from Gaussian distributions. See details for full model definition.

Usage

```
TenNest(n)
```

Arguments

n number of observations in the dataset. These are equally divided between clusters.

Details

The data set has five thousands iid dimensions.

Value

Returns a list having two components described as follows;

n number of observations.

K number of clusters.

cluster size number of observations in each cluster.

data data set generated from the model.

truelab clustering label vector corresponding to the known data generating model.

Author(s)

Fatima Batool <ucakfba@ucl.ac.uk>

References

Batool F. (2019). Optimum average silhouette width clustering. *PhD Thesis*, University College London.

Examples

```
dmat <- TenNest(1000)
plot2d(dmat$data, dmat$truelab)
```

| | |
|-----------------|--|
| ThreeMicroarray | <i>A data generating model with three microarray like clusters</i> |
|-----------------|--|

Description

Generates data set consists of three clusters.

Usage

```
ThreeMicroarray(n)
```

Arguments

| | |
|----------|--|
| n | number of observations in the dataset. These are equally divided between clusters. |
|----------|--|

Details

The data set has 1000-dimensions. The dimensions are iid.

Value

Returns a list having two components described as follows;

n number of observations.

K number of clusters.

cluster size number of observations in each cluster.

data data set generated from the model.

truelab clustering label vector corresponding to the known data generating model.

Author(s)

Fatima Batool <ucakfba@ucl.ac.uk>

References

Batool F. (2019). Optimum average silhouette width clustering. *PhD Thesis*, University College London.

Examples

```
dmat <- ThreeMicroarray(120)
plot2d(dmat$data, dmat$truelab)
```

ThreeMicroarrayMulti *A data generating model to simulate microarray-like settings*

Description

Generates data set consists of three clusters as defined in Van der Laan (2003).

Usage

```
ThreeMicroarrayMulti(n)
```

Arguments

n number of observations in the dataset. These are equally divided between clusters.

Details

The data set has 1000-dimensions and K=7 number of clusters. The dimensions are iid.

Value

Returns a list having two components described as follows;

n number of observations.

K number of clusters.

cluster size number of observations in each cluster.

data data set generated from the model.

truelab clustering label vector corresponding to the known data generating model.

Author(s)

Fatima Batool <ucakfba@ucl.ac.uk>

References

Van der Laan, M., K. Pollard, and J. Bryan (2003). A new partitioning around medoids algorithm. *Journal of Statistical Computation and Simulation*, 73(8), 575-584.

Examples

```
dmat <- ThreeMicroarrayMulti(60)
```

TNestedGaussian

A data generating model with three nested clusters

Description

Generates data set consists of three clusters. Two closely located Gaussain clusters are nested inside Student's t cluster.

Usage

```
TNestedGaussian(n)
```

Arguments

n number of observations in the dataset. These are equally divided between clusters.

Details

The data set has two iid dimensions.

Value

Returns a list having two components described as follows;

n number of observations.

K number of clusters.

cluster size number of observations in each cluster.

data data set generated from the model.

truelab clustering label vector correspondung to the known data generating model.

Author(s)

Fatima Batool <ucakfba@ucl.ac.uk>

References

Batool F. (2019). Optimum average silhouette width clustering. *PhD Thesis*, University College London.

Batool F. (2019). Optimum average silhouette width clustering. *PhD Thesis*, University College London.

Examples

```
dmat <- TNestedGaussian(300)
plot(dmat$data, col=c("blue", "red", "green")[dmat$truelab], xlab = " ", ylab = " ")
```

TwoGaussian*A data generating model with two clusters*

Description

Generates data set consists of two clusters, one each from Gaussian distributions.

Usage

```
TwoGaussian(n)
```

Arguments

n number of observations in the dataset. These are equally divided between clusters.

Details

The data set has two iid dimensions.

Value

Returns a list having two components described as follows;

n number of observations.

K number of clusters.

cluster size number of observations in each cluster.

data data set generated from the model.

truelab clustering label vector corresponding to the known data generating model.

Author(s)

Fatima Batool <ucakfba@ucl.ac.uk>

References

Batool F. (2019). Optimum average silhouette width clustering. *PhD Thesis*, University College London.

Examples

```
dmat <- TwoGaussian(1000)
plot2d(dmat$data, dmat$truelab)
```

TwoGaussianT

*A data generating model with three clusters***Description**

Generates data set consists of three clusters, two clusters from Gaussian distributions which are compact and closely located to each other but far from the third cluster generated from the Student's t distribution with wider spread.

Usage

```
TwoGaussianT(n)
```

Arguments

n number of observations in the dataset. These are equally divided between clusters.

Details

The data set has two iid dimensions.

Value

Returns a list having two components described as follows;

n number of observations.

K number of clusters.

cluster size number of observations in each cluster.

data data set generated from the model.

truelab clustering label vector corresponding to the known data generating model.

Author(s)

Fatima Batool <ucakfba@ucl.ac.uk>

References

Batool F. (2019). Optimum average silhouette width clustering. *PhD Thesis*, University College London.

Examples

```
dmat <- TwoGaussianT(1000)
plot2d(dmat$data, dmat$truelab)
```

UniGauss*A data generating model with two clusters*

Description

Generates a data set consists of 2-clusters, one each from the Gaussian and Uniform distributions.

Usage

```
UniGauss(n)
```

Arguments

n number of observations in the dataset. These are equally divided between clusters.

Details

The data set has two iid dimensions.

Value

Returns a list having two components described as follows;

n number of observations.

K number of clusters.

cluster size number of observations in each cluster.

data data set generated from the model.

truelab clustering label vector corresponding to the known data generating model.

Author(s)

Fatima Batool <ucakfba@ucl.ac.uk>

References

Batool F. (2019). Optimum average silhouette width clustering. *PhD Thesis*, University College London.

Examples

```
dmat <- UniGauss(1000)
plot2d(dmat$data, dmat$truelab)
plot(dmat$data, col=dmat$truelab, xlim=c(-20, 20), ylim=c(-20, 20))
```


UnitGaussian3D

*A data generating model with nine clusters in three dimensions***Description**

Generates data set consists of nine clusters each drawn from a Gaussian distribution. See details for complete discription of the model's structure.

Usage

```
UnitGaussian3D(n)
```

Arguments

n number of observations in the dataset. These are equally divided between clusters.

Details

The data set has three iid dimensions.

Value

Returns a list having two components described as follows:

Returns a list having two components described as follows;

n number of observations.

K number of clusters.

cluster size number of observations in each cluster.

data data set generated from the model.

truelab clustering label vector corresponding to the known data generating model.

Author(s)

Fatima Batool <ucakfba@ucl.ac.uk>

References

Batool F. (2019). Optimum average silhouette width clustering. *PhD Thesis*, University College London.

Examples

```
dmat <- UnitGaussian3D(891)
plot2d(dmat$data, dmat$truelab)
scatter3D(data[,1], data[,2], data[,3], colkey = FALSE, colvar = truelab,
  ticktype = "detailed", pch = 16, bty = "b")
```

Index

*Topic **Rcpp**
oasw-package, [2](#)

C14D2, [3](#)
C7D10, [4](#)
CCo4D2, [5](#)
Co5d5, [6](#)

dist, [9](#), [10](#), [13](#), [18](#)

ElongUniGaussian, [7](#)

FarUniGauss, [8](#)
fossil, [9](#)
fossilFix, [10](#)

Gaussian3, [11](#)
Gaussian4, [12](#)

hclust, [2](#), [15](#)
hosil, [13](#)

init, [14](#), [15](#)

kmeans, [2](#), [15](#)

LongUniGauss, [16](#)

Mclust, [2](#), [15](#)

NoisyGaussian, [17](#)

oasw-package, [2](#)
osil, [10](#), [11](#), [18](#)
osilFix, [18](#), [19](#)

pairplots, [20](#)
pam, [2](#), [15](#)
pamsil, [21](#)
pamsilFix, [22](#)
plot2d, [23](#)

ShortUniGauss, [24](#)

TChiGaussianF, [25](#)
TenNest, [26](#)
ThreeMicroarray, [27](#)

ThreeMicroarrayMulti, [28](#)
TNestedGaussian, [29](#)
TwoGaussian, [30](#)
TwoGaussianT, [31](#)

UniGauss, [32](#)
UnitGaussian3D, [33](#)