

Package: DBCVindex (via r-universe)

February 20, 2025

Encoding UTF-8

Title Calculates the Density-Based Clustering Validation (DBCV) Index

Version 1.4

Description A metric called 'Density-Based Clustering Validation index' (DBCV) index to evaluate clustering results, following the
<<https://github.com/pajaskowiak/clusterConfusion/blob/main/R/dbcv.R>>
'R' implementation by Pablo Andretta Jaskowiak. Original 'DBCV' index article: Moulavi, D., Jaskowiak, P. A., Campello, R. J., Zimek, A., and Sander, J. (April 2014), ``Density-based clustering validation'', Proceedings of SDM 2014 -- the 2014 SIAM International Conference on Data Mining (pp. 839-847), <[doi:10.1137/1.9781611973440.96](https://doi.org/10.1137/1.9781611973440.96)>.

Depends R (>= 4.0.0)

License GPL-3

URL <https://github.com/davidechicco/DBCVindex>

BugReports <https://github.com/davidechicco/DBCVindex/issues>

Imports qpdf

Suggests knitr, rmarkdown

VignetteBuilder knitr

RoxygenNote 7.3.2

NeedsCompilation no

Author Davide Chicco [cre] (<<https://orcid.org/0000-0001-9655-7142>>),
Pablo Andretta Jaskowiak [aut]
(<<https://orcid.org/0000-0002-6377-3372>>)

Maintainer Davide Chicco <davidechicco@davidechicco.it>

Repository CRAN

Date/Publication 2025-02-19 23:40:15 UTC

Config/pak/sysreqs libjpeg-dev libssl-dev

Contents

dbcv_index	2
matrix_mutual_reachability_distance	3
MST_Edges	4

Index	6
--------------	----------

dbcv_index	<i>Function that calculates the Density-Based Clustering Validation index (DBCV) of clustering results</i>
------------	--

Description

Function that calculates the Density-Based Clustering Validation index (DBCV) of clustering results

Usage

```
dbcv_index(data, partition, noiseLabel = -1)
```

Arguments

data	input clustering results
partition	labels of the clustering
noiseLabel	the code of the noise cluster points, -1 by default

Value

a real value containing the DBCV coefficient in the [-1;+1] interval

Examples

```
n = 300; noise = 0.05;
seed = 1782;
theta <- seq(0, pi, length.out = n / 2)
x1 <- cos(theta) + rnorm(n / 2, sd = noise)
y1 <- sin(theta) + rnorm(n / 2, sd = noise)
x2 <- cos(theta + pi) + rnorm(n / 2, sd = noise)
y2 <- sin(theta + pi) + rnorm(n / 2, sd = noise)
X <- rbind(cbind(x1, y1), cbind(x2, y2))
y <- c(rep(0, n / 2), rep(1, n / 2))

cat("dbcv_index(X, y) = ", dbcv_index(X, y), "\n", sep="")
```

`matrix_mutual_reachability_distance`*Function that calculates the mutual reachability distance within a matrix*

Description

Function that calculates the mutual reachability distance within a matrix

Usage

```
matrix_mutual_reachability_distance(MinPts, G_edges_weights, d)
```

Arguments

MinPts	number of minimal points
G_edges_weights	matrix of edges weights
d	number of features

Value

a list of two elements: `d_ucore` and `G_edges_weights`:

Examples

```
n = 300; noise = 0.05; seed = 1782;
theta <- seq(0, pi, length.out = n / 2)
x1 <- cos(theta) + rnorm(n / 2, sd = noise)
y1 <- sin(theta) + rnorm(n / 2, sd = noise)
x2 <- cos(theta + pi) + rnorm(n / 2, sd = noise)
y2 <- sin(theta + pi) + rnorm(n / 2, sd = noise)
X <- rbind(cbind(x1, y1), cbind(x2, y2))
y <- c(rep(0, n / 2), rep(1, n / 2))

nfeatures <- ncol(X)
i <- 1
clusters <- unique(y)
objcl <- which(y == clusters[i])
nuobjcl <- length(objcl)

noiseLabel <- -1
distX <- as.matrix(dist(X))^2
distXy <- distX[y != noiseLabel, y != noiseLabel]

mr <- matrix_mutual_reachability_distance(nuobjcl, distXy[objcl, objcl], nfeatures)
```

MST_Edges

*Function that finds the list of MST edges***Description**

Function that finds the list of MST edges

Usage

```
MST_Edges(G, start, G_edges_weights)
```

Arguments

G list of four elements: number of vertices, MST_edges (matrix of edges), MST_degrees (array of numbers), MST_parent (array of numbers)

start index of the first edge

G_edges_weights matrix of edges weights

Value

list of two elements: matrix of edges and array of degrees

Examples

```
n = 300; noise = 0.05;
seed = 1782;
theta <- seq(0, pi, length.out = n / 2)
x1 <- cos(theta) + rnorm(n / 2, sd = noise)
y1 <- sin(theta) + rnorm(n / 2, sd = noise)
x2 <- cos(theta + pi) + rnorm(n / 2, sd = noise)
y2 <- sin(theta + pi) + rnorm(n / 2, sd = noise)
X <- rbind(cbind(x1, y1), cbind(x2, y2))
y <- c(rep(0, n / 2), rep(1, n / 2))

nfeatures <- ncol(X)
i <- 1
clusters <- unique(y)
objcl <- which(y == clusters[i])
nuobjcl <- length(objcl)

noiseLabel <- -1
distX <- as.matrix(dist(X))^2
distXy <- distX[y != noiseLabel, y != noiseLabel]

mr <- matrix_mutual_reachability_distance(nuobjcl, distXy[objcl, objcl], nfeatures)

d_ucore_cl <- rep(0, nrow(X))
d_ucore_cl[objcl] <- mr$d_ucore
```

```
G <- list(no_vertices = nuobjcl, MST_edges = matrix(0, nrow = nuobjcl - 1, ncol = 3),
         MST_degrees = rep(0, nuobjcl), MST_parent = rep(0, nuobjcl))
g_start <- 1

mst_results <- MST_Edges(G, g_start, mr$G_edges_weights)
```

Index

`dbcv_index`, 2

`matrix_mutual_reachability_distance`, 3

`MST_Edges`, 4