

# Package ‘DisimForMixed’

July 21, 2025

**Type** Package

**Title** Calculate Dissimilarity Matrix for Dataset with Mixed Attributes

**Version** 0.2

**Date** 2016-03-08

**Author** Hasanthi A. Pathberiya

**Maintainer** Hasanthi A. Pathberiya <hasanthi@sjp.ac.lk>

**Imports** dplyr, cluster

**Description** Implement the methods proposed by Ahmad & Dey (2007) <[doi:10.1016/j.datak.2007.03.016](https://doi.org/10.1016/j.datak.2007.03.016)> in calculating the dissimilarity matrix at the presence of mixed attributes. This Package includes functions to discretize quantitative variables, calculate conditional probability for each pair of attribute values, distance between every pair of attribute values, significance of attributes, calculate dissimilarity between each pair of objects.

**License** GPL

**LazyData** TRUE

**RoxygenNote** 5.0.1

**NeedsCompilation** no

**Repository** CRAN

**Date/Publication** 2016-06-06 17:56:42

## Contents

calcCondProb . . . . .	2
calcDissimMat . . . . .	2
discretizeQuant . . . . .	3
distBetPairs . . . . .	4
findMax . . . . .	5
signifOfQuantVars . . . . .	6

<b>Index</b>	<b>8</b>
--------------	----------

---

calcCondProb                      *Calculate Conditional Probabilities.*

---

### Description

Takes in a data frame which contains only qualitative variables. Discretized quantitative variables, a mixture of qualitative variables and discretized quantitative variables are also accepted. Calculates conditional probabilities for each pair of attribute values in the data frame. Returns a data frame consists of J, A, B and C in columns where  $\Pr(A|B) = C$  and J is the column number in the input data frame corresponding to the values in A.

### Usage

```
calcCondProb(myDataAll)
```

### Arguments

myDataAll                      A data frame which includes qualitative variables OR discretized quantitative variables OR a mixture of qualitative variables and discretized quantitative variables in columns.

### Value

A data frame with four columns J, A, B and C in columns where  $\Pr(A|B) = C$  and J is the column number in the input data frame corresponding to the values in A.

### Examples

```
QualiVars <- data.frame(Qlvar1 = c("A","B","A","C"), Qlvar2 = c("Q","Q","R","Q"))
CalcForQuali <- calcCondProb(QualiVars)
QuantVars <- data.frame(Qnvar1 = c(1.5,3.2,4.9,5), Qnvar2 = c(4.8,2,1.1,5.8))
Discretized <- discretizeQuant(QuantVars)
CalcForQuant <- calcCondProb(Discretized)
AllQualQuant <- data.frame(QualiVars, Discretized)
CalcForAll <- calcCondProb(AllQualQuant)
```

---

calcDissimMat                      *Calculate Dissimilarity Matrix for Mixed Attributes.*

---

### Description

Takes in two data frames where first contains only qualitative attributes and the other contains only quantitative attributes. Function calculates the dissimilarity matrix based on the method proposed by Ahmad & Dey (2007).

**Usage**

```
calcDissimMat(myDataQuali, myDataQuant)
```

**Arguments**

myDataQuali     A data frame which includes only qualitative variables in columns.  
 myDataQuant     A data frame which includes only quantitative variables in columns.

**Details**

calcDissimMat is an implementation of the method proposed by Ahmad & Dey (2007) to calculate the dissimilarity matrix at the presence of both qualitative and quantitative attributes. This approach finds dissimilarity of qualitative and quantitative attributes separately and the final dissimilarity matrix is formed by combining both. See Ahmad & Dey (2007) for more details.

**Value**

A dissimilarity matrix. This can be used as an input to pam, fanny, agnes and diana functions.

**References**

Ahmad, A., & Dey, L. (2007). A k-mean clustering algorithm for mixed numeric and categorical data. *Data & Knowledge Engineering*, 63(2), 503-527.

**Examples**

```
QualiVars <- data.frame(Qlvar1 = c("A", "B", "A", "C", "C", "A"), Qlvar2 = c("Q", "Q", "R", "Q", "R", "Q"))
QuantVars <- data.frame(Qnvar1 = c(1.5, 3.2, 4.9, 5, 2.8, 3.1), Qnvar2 = c(4.8, 2, 1.1, 5.8, 3.1, 2.2))
DisSimMatCalcd <- calcDissimMat(QualiVars, QuantVars)

agnesClustering <- cluster::agnes(DisSimMatCalcd, diss = TRUE, method = "ward")
silWidths <- cluster::silhouette(cutree(agnesClustering, k = 2), DisSimMatCalcd)
mean(silWidths[,3])
plot(agnesClustering)

PAMClustering <- cluster::pam(DisSimMatCalcd, k=2, diss = TRUE)
silWidths <- cluster::silhouette(PAMClustering, DisSimMatCalcd)
plot(silWidths)
```

---

discretizeQuant     *Discretize Quantitative Variables.*

---

**Description**

Takes in a data frame which contains only Quantitative variables in columns. Standardize the variables. Discretize quantitative variables and returns discretized quantitative variables. Discretization was performed by equal width binning algorithm.

**Usage**

```
discretizeQuant(myDataQuant, noice = TRUE)
```

**Arguments**

**myDataQuant** A data frame which includes quantitative variables in columns.

**noice** Noice indicator. If `noice = TRUE` data standerdization is done by deviding the difference between data point and median of the variable by the range of the variable. If `noice = FALSE` data standerdization is done by deviding the difference between data point and mean of the variable by the standard deviation of the variable.

**Value**

A data frame consists of discretized quantitative variables.

**Examples**

```
QuantVars <- data.frame(Qnvar1 = c(1.5,3.2,4.9,5), Qnvar2 = c(4.8,2,1.1,5.8))
Discretized <- discretizeQuant(QuantVars)
```

---

<code>distBetPairs</code>	<i>Calculate Distance Between Attribute Values.</i>
---------------------------	---

---

**Description**

Takes in a data frame which contains only qualitative variables. Discretized quantitative variables , a mixture of qualitative variables and discretized quantitative variables are also accepted. Calculates distance between each pair of attribute values for a given attribute. This calculation is done according to the method proposed by Ahmad & Dey (2007).

**Usage**

```
distBetPairs(myDataAll)
```

**Arguments**

**myDataAll** A data frame which includes qualitative variables OR discretized quantitative variables OR a mixture of qualitative variables and discretized quantitative variables in columns.

**Details**

`distBetPairs` is an implementtion of the method proposed by Ahmad & Dey (2007) to find the distance between two catogorical values corresponding to a qualitative variable. This distance measure considers distribution of values in the data set. This function is also used to find the distance between discretized values corresponding to quantitative variables which are used in calculating the significance of quantitative attributes. See Ahmad & Dey (2007) for more details.

**Value**

A data frame with four columns J, A, B and C in columns where  $\text{Distance}(A, B) = C$  and J is the column number in the input data frame corresponding to the values in A.

**References**

Ahmad, A., & Dey, L. (2007). A k-mean clustering algorithm for mixed numeric and categorical data. *Data & Knowledge Engineering*, 63(2), 503-527.

**Examples**

```
QualiVars <- data.frame(Qlvar1 = c("A","B","A","C"), Qlvar2 = c("Q","Q","R","Q"))
library(dplyr)
distForQuali <- distBetPairs(QualiVars)
QuantVars <- data.frame(Qnvar1 = c(1.5,3.2,4.9,5), Qnvar2 = c(4.8,2,1.1,5.8))
Discretized <- discretizeQuant(QuantVars)
distForQuant <- distBetPairs(Discretized)
AllQualQuant <- data.frame(QualiVars, Discretized)
distForAll <- distBetPairs(AllQualQuant)
```

---

findMax

*Calculate Distance Between given Attribute Values by considering only a pair of attributes.*

---

**Description**

Takes in two lists  $A_i$  and  $A_j$ , representing values of two attributes, two values  $x$  and  $y$  from  $A_i$ . Quantitative attributes are accepted only after discretization. Calculates distance between  $x$  and  $y$  for  $A_j$  with respect to  $A_i$ .

**Usage**

```
findMax(Ai, Aj, x, y)
```

**Arguments**

$A_i$	A list consisting values of a selected attribute
$A_j$	A list consisting values of another selected attribute
$x$	Value from $A_i$
$y$	Another value from $A_i$

**Details**

findMax is the implementation of find\_max() function proposed by Ahmad & Dey (2007). See Ahmad & Dey (2007) for more details.

**Value**

distance between x and y for  $A_j$  with respect to  $A_i$ .

**References**

Ahmad, A., & Dey, L. (2007). A k-mean clustering algorithm for mixed numeric and categorical data. *Data & Knowledge Engineering*, 63(2), 503-527.

**Examples**

```
Attrib_i <- c("A", "B", "A", "C")
Attrib_j <- c("Q", "Q", "R", "Q")
xVal <- "A"
yVal <- "B"
QualiVars <- data.frame(Qlvar1 = c("A", "B", "A", "C"), Qlvar2 = c("Q", "Q", "R", "Q"))
library(dplyr)
distBetXY <- findMax(Attrib_i, Attrib_j, xVal, yVal)
```

---

signifOfQuantVars      *Calculate Significance of Quantitative Attributes.*

---

**Description**

Takes in two data frames where first contains only qualitative attributes and the other contains only quantitative attributes. Function calculates significance of quantitative attributes based on the method proposed by Ahmad & Dey (2007).

**Usage**

```
signifOfQuantVars(myDataQuali, myDataQuant)
```

**Arguments**

myDataQuali      A data frame which includes only qualitative variables in columns.  
myDataQuant      A data frame which includes only quantitative variables in columns.

**Details**

signifOfQuantVars is an implementation of the method proposed by Ahmad & Dey (2007) to calculate the significance of quantitative attributes. Significance of an attribute is an important fact to consider in the process of clustering. To calculate the significance quantitative attributes are discretized first. These significance values are used in calculating distance between any two numeric values of a quantitative attribute. See Ahmad & Dey (2007) for more details.

**Value**

A data frame with two columns A and B where A represents variable number and B represents significance of corresponding variable.

**References**

Ahmad, A., & Dey, L. (2007). A k-mean clustering algorithm for mixed numeric and categorical data. *Data & Knowledge Engineering*, 63(2), 503-527.

**Examples**

```
QualiVars <- data.frame(Qlvar1 = c("A","B","A","C"), Qlvar2 = c("Q","Q","R","Q"))
QuantVars <- data.frame(Qnvar1 = c(1.5,3.2,4.9,5), Qnvar2 = c(4.8,2,1.1,5.8))
SigOfQuant <- signifOfQuantVars(QualiVars, QuantVars)
```

# Index

calcCondProb, 2  
calcDissimMat, 2

discretizeQuant, 3  
distBetPairs, 4

findMax, 5

signifOfQuantVars, 6