Package 'RecAssoRules'

January 20, 2025

Type Package

Title Recursive Mining for Frequent Pattern and Confident Association
Rules

Version 1.0

Date 2020-11-14

Author Louis Raimbault [aut,cre],
A.Mokkadem [aut],
M.Pelletier [aut]

Maintainer Louis Raimbault <Louis.Raimbault@icloud.com>

Description Provides functions allowing the user to recursively extract frequent patterns and confi-
dent rules according to indicators of minimal support and minimal confidence. These func-
tions are described in ``Recursive Association Rule Mining'' Abdelkader Mokkadem, Mari-
ane Pelletier, Louis Raimbault (2020) <arXiv:2011.14195>.

License GPL (>= 2)

Imports Rcpp (>= 1.0.5)

LinkingTo Rcpp

URL https://github.com/LouisRaimbault/RecAssoRules-R

BugReports https://github.com/LouisRaimbault/RecAssoRules-R

NeedsCompilation yes

RoxygenNote 7.1.1

Repository CRAN

Date/Publication 2020-12-04 10:30:02 UTC

Contents

RecAssoRules-package   *Recursive mining for frequent itemSet and confident association rules.*

Description

This first version makes it possible to carry out the usual applications of association rule Mining.This by using recursive algorithms. All of the functions are using the C++ language. For more usage security, it mainly uses the std and stl libraries, reducing only very slightly the efficiency of the algorithms. It will be updated soon with functions allowing it to be applied on non-fixed databases, as well as a more practical interface.

---

prefrecrules   *Mining for frequent patterns and confident rules.*

Description

Returns a list with two objets : freqset a R DataFrame with 3 columns , the set-names , their support and relative support and rulesset a R Dataframe with 3 colums , the antecedant names , the consequent names and their confidence. When lauching the function, the BinaryMatrix is ordered in ascendant colsums order. For each column, the frequent itemSets and confident Rules containing the current tested item are extracted with the ARM traversal function.

Usage

```
prefrecrules (Bitmax  , varnames , relativeSup , Minconf )
```

Arguments

Bitmax	A binary DataFrame.
varnames	A vector of character or a Matrix character.
relativeSup	The wanted relative min Support value.
Minconf	The wanted min confident value.

Examples

```
data("Transacfruits")
## Import and build Transafruits as binary matrix.
binaryfruits = transactiontoBitmax (as.matrix(Transacfruits),",")
outsetrules = prefrecrules (binaryfruits, names(binaryfruits),0.02, 0.4)
## Mine frequents itemsets and confidentes rules with :
## relative min support of 0.02 and min confidence of 0.4.
```

---

prefrecset *Mining for frequent patterns.*

Description

Provides a R DataFrame with 3 columns , the set-names , their support and relative support. When lauching the function, The BinaryMatrix is ordered in ascendant colsums order. For each column, the frequent itemSets containing the current tested item are extracted with the traversal function.

Usage

```
prefrecset (Bitmax , varnames , relativeSup )
```

Arguments

Bitmax	A binary DataFrame
varnames	A vector of character or a Matrix character.
relativeSup	The wanted relative min Support value.

Examples

```
data("Transacfruits")
## Import and build Transafruits as binary matrix.
binaryfruits = transactiontoBitmax (as.matrix(Transacfruits),",",")
itemsets = prefrecset (binaryfruits, names(binaryfruits), 0.02)
## Mine frequents itemSet from Transacfruits with a relative min support of 0.02.
```

Transacfruits *Transacfruits Data Set*

Description

The Transacfruits dataset is a synthetic database. It thus represents fictitious transactions, simulating fruit purchases, among a random list.

Usage

```
data("Transacfruits")
```

Details

Transacfruits is made up of 1 columns and 100 rows. There are therefore 100 transactions. For each of them one or more fruits were purchased from a list of 15 fruits. It was generated by binomial simulation, for all transactions, there was for each fruit a fixed probability p of being bought.

| transactiontoBitmax | *Creation of a Binary DataFRame from transactions type* |

Description

Returns the corresponding Binary R DataFrame. If you imported your transaction as R DataFrame, please use the R function as.matrix(yourdataframe).

Usage

```
transactiontoBitmax(transac , deli)
```

Arguments

transac	a character vector or matrix.
deli	The character used to delimit the objects present in a transaction.

Examples

```
data("Transacfruits")
binaryfruits = transactiontoBitmax (as.matrix(Transacfruits),",")
## Import transaction data Transacfruits and create the correspondant binary matrix.
```

Index