Package 'RSarules'

October 15, 2016

Type Package

Title Random Sampling Association Rules from a Transaction Dataset

Version 1.0

Date 2016-10-11

Author Xiaoying Sun, Guoqi Qian and Yuehua Wu

Maintainer Xiaoying Sun <sunying@mathstat.yorku.ca>

Depends R (>= 3.3.1), arules (>= 1.4.1), Matrix (>= 1.2.6)

Imports methods, stats

Suggests MultiOrd

Description

Implements the Gibbs sampling algorithm to randomly sample association rules with one pre-chosen item as the consequent from a transaction dataset. The Gibbs sampling algorithm was proposed in G. Qian, C.R. Rao, X. Sun and Y. Wu (2016) <DOI:10.1073/pnas.1604553113>.

License GPL-3

Repository CRAN

NeedsCompilation no

Date/Publication 2016-10-15 11:21:10

R topics documented:

RSarules *Random Sampling Association Rules from a Transaction Dataset*

Description

Random sampling association rules according to a proposed probability distribution based on a function of support and confidence of the rules.

Usage

```
RSarules(data, M, ig, rhs, lhs_offset = NULL)
```

Arguments

data	a transaction dataset of any data structure which can be coerced into matrix (e.g., a binary matrix or data.frame). Each column contains an item and each row contains a transaction.
M	the number of association rules sampled from the transaction dataset.
ig	the value for the tuning parameter. See reference for more details.
rhs	the column number of an item to be the consequent of the sampled association rules.
lhs_offset	a vector of column numbers corresponding to a set of items that would be excluded in the antecedent of the sampled association rules. By default, all items excluding the consequent item could appear in the antecedent of the sampled association rules.

Value

A list contains the following components:

sampled_items	items appeared in the sampled rules and their frequencies. e.g. I 3 with frequency 0.1 means 10% sampled rules contain I 3 in their antecedents. They are ordered according to their frequencies.
sampled_rules	a transaction object contains M sampled rules.
measures	various measures for the sampled rules including support, confidenc e and importance in the transaction dataset and frequencies in the random sample.

Author(s)

Xiaoying Sun, Guoqi Qian and Yuehua Wu

Maintainer: Xiaoying Sun (sunying@mathstat.yorku.ca)

References

[1] G. Qian, C.R. Rao, X. Sun and Y. Wu. Boosting association rule mining in large datasets via Gibbs sampling. Proceedings of the National Academy of Sciences 113.18 (2016): 4958-4963. DOI: 10.1073/pnas.1604553113.

Examples

```
### simulation study: example 1

## generate data using R package 'MultiOrd'

set.seed(200)
library(MultiOrd)
library(arules)
l <- 5
n <- 100
mp <- rep(0.5, l-1)
bcor <- diag( x=1, nrow = l-1, ncol = l-1 )
bcor[1, l-1] <- 0.8
bcor[l-1, 1] <- 0.8
bcor[2, l-1] <- 0
bcor[l-1, 2] <- 0
bcor[3, l-1] <- 0.2
bcor[l-1, 3] <- 0.2
validation.CorrMat( mp, bcor)
dd <- generate.binary( n, mp, bcor)
data <- cbind(dd, 1- dd[, l-1])
colnames(data) <- c( paste( "I", 1:(l-2), sep = ""), "r1", "r2")

## Response being the last second item

rhs <- dim(data)[2]-1 # the last second item to be in the rhs
lhs_offset <- c( dim(data)[2])  # column numbers that are not contained in the lhs
M <- 10 # number of arules need to be sampled. M = 1000 in the reference paper.
ig <- 10 # the value for the tuning parameter 3, 6, 10
result <- RSarules( data = data, rhs = rhs, M = M , ig = ig, lhs_offset = lhs_offset )
result
inspect(result$sampled_rules)

## Response being the last second item

rhs2 <- dim(data)[2] # the last second item to be in the rhs
lhs_offset2 <- c( dim(data)[2]-1)  # column numbers that are not contained in the lhs
M <- 10 # number of arules need to be sampled. M = 1000 in the reference paper.
ig <- 10 # the value for the tuning parameter 3, 6, 10
result2 <- RSarules( data = data, rhs = rhs2, M = M , ig = ig, lhs_offset = lhs_offset2 )
result2
inspect(result2$sampled_rules)
```

Index