Package 'arulesNBMiner'

July 2, 2015

Version 0.1-5
Date 2015-07-02
Title Mining NB-Frequent Itemsets and NB-Precise Rules
Author Michael Hahsler
Maintainer Michael Hahsler <mhahsler@lyle.smu.edu>
Description NBMiner is an implementation of the model-based mining algorithm for mining NB-frequent itemsets presented in "Michael Hahsler. A model-based frequency constraint for mining associations from transaction data. Data Mining and Knowledge Discovery, 13(2):137-166, September 2006." In addition an extension for NB-precise rules is implemented.
Depends R (>= 2.10), arules (>= 0.6-6), rJava (>= 0.6-3)
URL http://R-Forge.R-project.org/projects/arules/,
       http://lyle.smu.edu/IDA/arules/
Imports methods, stats, graphics
SystemRequirements Java (>= 5.0)
License GPL-2
NeedsCompilation no
Repository CRAN
Date/Publication 2015-07-02 13:42:39

R topics documented:

Agrawal ................................................................. 2
NBMiner ............................................................. 3
NBMinerParameters ................................................ 4

Index 6
Description

This dataset is generated by the method described by Agrawal and Srikant (1994) using the reimplementation in arules which also retains the patterns used in the generation process.

Usage

data(Agrawal)

Format

The format is: transactions Agrawal.db itemsets Agrawal.pat

Details

Agrawal.db contains the dataset (1000 items/20000 transactions) and Agrawal.pat contains the patterns that were used to create the dataset.

References


Examples

data(Agrawal)

summary(Agrawal.pat)
summary(Agrawal.db)

## the data sets was generated with the following code
## Not run:
Agrawal.pat <- random.patterns(1000, nPats = 2000, method = "agrawal",
   lpats = 2, corr = 0.5, cmean = 0.5, cvar = 0.1, iWeight = NULL,
   verbose = FALSE)
Agrawal.db <- random.transactions(1000, 20000, method="agrawal",
   patterns = Agrawal.pat)

## End(Not run)
**NBMiner**

**NBMiner: Mine NB-Frequent Itemsets or NB-Precise Rules**

**Description**

Calls the Java implementation of the depth first search algorithm described in the paper in the references section to mine NB-frequent itemsets of NB-precise rules.

**Usage**

```r
NBMiner(data, parameter, control = NULL)
```

**Arguments**

- **data**: object of class transactions.
- **parameter**: a list of parameters (automatically converted into an object of class `NBMinerParameter`). Reasonable parameters can be obtained using `NBMinerParameters` (see details section).
- **control**: a list of control options (automatically converted into an object of class `NBMinerControl`). Currently only "verbose" and "debug" (both logical) are available.

**Details**

The parameters can be estimated from the data using `NBMinerParameters`.

**Value**

An object of class itemsets or rules (depending on the rules entry in parameter). The estimated precision is stored in the quality slot.

**References**


**See Also**

- `NBMinerParameters`, `transactions-class`, `itemsets-class`, `rules-class`

**Examples**

```r
data("Agrawal")

## mine
param <- NBMinerParameters(Agrawal.db, pi=0.99, theta=0.5, maxlen=5,
    minlen=1, trim = 0, verb = TRUE, plot=TRUE)
itemsets_NB <- NBMiner(Agrawal.db, parameter = param,
    control = list(verb = TRUE, debug=FALSE))
```
NBMinerParameters

Estimate Global Model Parameters from Data

Description

Estimate the global negative binomial data model used by the NBMiner and create an appropriate parameter object.

Usage

NBMinerParameters(data, trim = 0.01, pi = 0.99, theta = 0.5, minlen = 1, maxlen = 5, rules = FALSE, plot = FALSE, verbose = FALSE, getdata = FALSE)

Arguments

data the data as a object of class transactions.
trim fraction of incidences to trim off the tail of the frequency distribution of the data.
pi precision threshold $\pi$.
theta pruning parameter $\theta$.
minlen minimum number of items in found itemsets (default: 1).
maxlen maximal number of items in found itemsets (default: 5).
rules mine NB-precise rules instead of NB-frequent itemsets?
NBMinerParameters

plot the model?

verbose use verbose output for the estimation procedure.

getdata get also the observed and estimated counts.

Details

Uses the EM algorithm to estimate the global NB model for the data. The EM algorithm is used since the zero class (items which do not occur in the dataset) is not included in the data. The result are the two NB parameters \( k \) and \( a \), where \( a \) is rescaled by dividing it by the number of incidences in the data (this is needed by the NBMiner). Also the real number of items \( n \) is a result of the estimation.

theta and pi are just taken and added to the resulting parameter object.

Value

an object of class \texttt{NBMinerParameter} for \texttt{NBMiner}.

References


See Also

\texttt{NBMiner,transactions-class}

Examples

data("Epub")

param <- NBMinerParameters(Epub, trim = 0.05, plot = TRUE, verbose = TRUE)

param
Index

*Topic datasets
  Agrawal, 2

*Topic models
  NBMiner, 3
  NBMinerParameters, 4

Agrawal, 2

NBMiner, 3, 5
NBMinerControl-class (NBMiner), 3
NBMinerParameter-class (NBMiner), 3
NBMinerParameters, 3, 4