

Package ‘arulesViz’

April 2, 2012

Version 0.1-5

Date 2012-03-31

Title arulesViz - Visualizing Association Rules and Frequent Itemsets

Author Michael Hahsler and Sudheer Chelluboina

Maintainer Michael Hahsler <michael@hahsler.net>

Depends R (>= 2.14.0), arules (>= 1.0-5), MASS, scatterplot3d, vcd,seriation, igraph

Suggests iplots, Rgraphviz

Description Various visualization techniques for association rules and itemsets. The packages also includes several interactive visualizations for rule exploration. This package extends package arules.

License GPL (>= 2)

URL <http://r-forge.r-project.org/projects/arulesviz/>,
<http://lyle.smu.edu/IDA/arules/>

Copyright (C) 2011 Michael Hahsler and Sudheer Chelluboina

Repository CRAN

Date/Publication 2012-04-02 17:47:21

R topics documented:

abbreviate	2
plot	2
saveAsGraph	7

Index	8
--------------	----------

abbreviate	<i>Abbreviate function for item labels in transactions, itemMatrix and rules</i>
------------	--

Description

Methods to abbreviate long item labels in transactions and rules.

Usage

```
## S3 method for class 'transactions'
abbreviate(data, minlength = 4, ...,method = "both.sides")
## S3 method for class 'rules'
abbreviate(data, minlength = 4, ...,method = "both.sides")
## S3 method for class 'itemMatrix'
abbreviate(data, minlength = 4, ...,method = "both.sides")
```

Arguments

data	an object of class "transactions", "itemMatrix" and "rules"
minlength	characters allowed in abbreviation
method	apply to level and value (both.sides)
...	further arguments; currently unused

Author(s)

Sudheer Chelluboina and Michael Hahsler based on code by Martin Vodenicharov.

Examples

```
data(Adult)
Adult <- abbreviate(Adult, 9)
```

plot	<i>Plot method to visualize association rules and itemsets</i>
------	--

Description

This is the S3 method to visualize association rules and itemsets. Implemented are several popular visualization methods including scatter plots with shading (two-key plots), graph based visualizations, doubledecker plots, etc.

Usage

```
## S3 method for class 'rules'
plot(x, method = NULL, measure = "support", shading = "lift",
     interactive = FALSE, data = NULL, control = NULL, ...)
## S3 method for class 'itemsets'
plot(x, method = NULL, measure = "support",
     interactive=FALSE, data = NULL, control = NULL, ...)
```

Arguments

x	an object of class "rules" or "itemsets".
method	a string with value "scatterplot", "two-key plot", "matrix", "matrix3D", "mosaic", "doubledecker", "graph", "paracoord" or "grouped", "iplots" selecting the visualization method (see Details).
measure	measure(s) of interestingness (e.g., "support", "confidence", "lift") used in the visualization. Some visualization methods need one measure, others take a vector with two measures (e.g., scatterplot). In some plots (e.g., graphs) NA can be used to suppress using a measure.
shading	measure of interestingness used for the color of the points/arrows/nodes (e.g., "support", "confidence", "lift"). The default is "lift". NA can be often used to suppress shading.
interactive	enable interactive exploration (not implemented by all methods).
control	a list of control parameters for the plot. The available control parameters depends on the visualization technique (see Details).
data	the dataset (class "transactions") used to generate the rules/itemsets. Only "mosaic" and "doubledecker" require the original data.
...	further arguments are typically passed on to the used low-level plotting function.

Details

Most visualization techniques are described by Bruzese and Davino (2008), however, we added more color shading, reordering and interactive features. The following visualization method are available:

"scatterplot", "two-key plot" This visualization method draws a two dimensional scatterplot with different measures of interestingness (parameter "measure") on the axes and a third measure (parameter "shading") is represented by the color of the points. There is a special value for shading called "order" which produces a two-key plot where the color of the points represents the length (order) of the rule.

The list of control parameters for this method is

"main" plot title

"pch" use filled symbols: 20–25

"cex" symbol size

"xlim","ylim" limits

"jitter" a number greater than 0 adds jitter to the points

"col" color palette (default is 100 gray values.)

Interactive manipulations are available.

"matrix", "matrix3D" Arranges the association rules as a matrix with the itemsets in the antecedents on one axis and the itemsets in the consequent on the other. The interest measure is either visualized by a color (darker means a higher value for the measure) or as the height of a bar (method "matrix3D").

The list of control parameters for this method is

"main" plot title

"type" defines the way the data is rendered: "grid", "image" or "3D" (scatterplot3d)

"reorder" if TRUE then the itemsets on the x and y-axes are reordered to bring rules with similar values for the interest measure closer together and make the plot clearer.

"orderBy" specifies the measure of interest for reordering (default is the visualized measure)

"reorderMethod", "reorderControl", "reorderDist" seriation method, control arguments and distance method (default "euclidean") used for reordering (see `seriate()` method in **seriation**)

"col" a vector of n colors used for the plot (default: 100 gray values)

"xlim", "ylim" limits

Currently there is no interactive version available.

"grouped" Grouped matrix-based visualization (Hahsler and Chelloboina, 2011). Antecedents (columns) in the matrix are grouped using clustering. Groups are represented as balloons in the matrix.

The list of control parameters for this method is

"main" plot title

"k" number of antecedent groups (default: 20)

"aggr.fun" aggregation function can be any function computing a scalar from a vector (e.g., min, mean, median (default), sum, max). It is also used to reorder the balloons in the plot.

"col" color palette (default is 100 gray values.)

Interactive manipulations are available.

"graph" Represents the rules (or itemsets) as a graph.

Control arguments are

"main" plot title

"cex" cex for labels

"itemLabels" display item/itemset names instead of ids (TRUE)

"measureLabels" display values of interest measures (FALSE)

"precision" number of digits for numbers in plot.

"type" plot type: "items" or "itemsets"

"engine" graph layout engine: "igraph" (default) or "graphviz"

"layout" layout algorithm defined in **igraph** or **Rgraphviz** (default: layout.fruchterman.reingold for engine igraph and "dot"/"neato" for graphviz)

"arrowSize" [0,1]

"alpha" alpha transparency value (default .8; set to 1 for no transparency)

For the igraph engine the used plot function is `plot.igraph` in **igraph**. For graphviz the function `plot` in **Rgraphviz** is used. Note that Rgraphviz is available at <http://www.bioconductor.org/>. For the interactive version `tkplot` in **igraph** is always used.

... arguments are passed on to the respective plotting function (use for color, etc.).

"doubledecker", "mosaic" Represents a single rule as a doubledecker or mosaic plot. Parameter data has to be specified to compute the needed contingency table. Available control parameters are

"main" plot title

"paracoord" Represents the rules (or itemsets) as a parallel coordinate plot. Available control parameters are

"main" plot title

"reorder" reorder to minimize crossing lines.

"alpha" alpha transparency value

Currently there is no interactive version available.

"iplots" Experimental interactive plots (package **iplots**) which support selection, highlighting, brushing, etc. Currently plots a scatterplot (support vs. confidence) and several histograms. Interactive manipulations are available.

Value

Several interactive plots return a set of selected rules/itemsets. Other plots might return other data structures. For example graph-based plots return the graph (invisibly).

Author(s)

Michael Hahsler and Sudheer Chelluboina. Some visualizations are based on the implementation by Martin Vodenicharov.

References

Bruzzese, D. and Davino, C. (2008), Visual Mining of Association Rules, in Visual Data Mining: Theory, Techniques and Tools for Visual Analytics, Springer-Verlag, pp. 103–122.

Hahsler M. and Chelluboina S. (2011), Visualizing association rules in hierarchical groups. In 42nd Symposium on the Interface: Statistical, Machine Learning, and Visualization Algorithms (Interface 2011). The Interface Foundation of North America.

See Also

`scatterplot3d` in **scatterplot3d**, `plot.igraph` and `tkplot` in **igraph**, `seriate` in **seriation**

Examples

```
data(Groceries)
rules <- apriori(Groceries, parameter=list(support=0.001, confidence=0.5))
rules

## scatterplot
```

```

plot(rules)
## try: sel <- plot(rules, interactive=TRUE)

## Two-key plot is a scatterplot with shading = "order"
plot(rules, shading="order", control=list(main = "Two-key plot"))

## the following techniques work better with fewer rules
subrules <- rules[quality(rules)$confidence > 0.8]

## 2D matrix with shading
plot(subrules, method="matrix", measure="lift")
plot(subrules, method="matrix", measure="lift", control=list(reorder=TRUE))

## 3D matrix
plot(subrules, method="matrix3D", measure="lift")
plot(subrules, method="matrix3D", measure="lift", control=list(reorder=TRUE))

## matrix with two measures
plot(subrules, method="matrix", measure=c("lift", "confidence"))
plot(subrules, method="matrix", measure=c("lift", "confidence"),
control=list(reorder=TRUE))

## try: plot(subrules, method="matrix", measure="lift", interactive=TRUE, control=list(reorder=TRUE))

## grouped matrix plot
plot(rules, method="grouped")
plot(rules, method="grouped", control=list(k=30))
## try: sel <- plot(rules, method="grouped", interactive=TRUE)

## graphs only work with very few rules
subrules2 <- sample(rules, 10)
plot(subrules2, method="graph")
plot(subrules2, method="graph",
control=list(type="items"))
## try: plot(subrules2, method="graph", interactive=TRUE)
## try: plot(subrules2, method="graph", control=list(engine="graphviz", type="items"))

## parallel coordinates plot
plot(subrules2, method="paracoord")
plot(subrules2, method="paracoord", control=list(reorder=TRUE))

## Doubledecker plot only works for a single rule
oneRule <- sample(rules, 1)
plot(oneRule, method="doubledecker", data = Groceries)

## use iplots (experimental)
## try: sel <- plot(rules, method="iplots", interactive=TRUE)

## for itemsets
itemsets <- eclat(Groceries, parameter = list(support = 0.02))
plot(itemsets, method="paracoord", control=list(alpha=.5, reorder=TRUE))

```

```
plot(itemsets, method="graph")
```

saveAsGraph	<i>Save rules or itemsets as a graph description.</i>
-------------	---

Description

This function saves a rule set as a graph description in different formats (e.g., GraphML, dimacs, dot).

Usage

```
saveAsGraph(x, file, type=NULL, format="graphml")
```

Arguments

x	an object of class "rules" or "itemsets".
file	file name
type	see type in plot with method "graph" (e.g., "itemsets", "items").
format	file format (e.g., "edgelist", "graphml", "dimacs", "gml", "dot"). See <code>write.graph</code> in package igraph .

See Also

[plot](#), [write.graph](#) in **igraph**

Examples

```
data("Groceries")
rules <- apriori(Groceries, parameter=list(support=0.01, confidence=0.5))

saveAsGraph(rules, "rules.graphml")

## clean up
unlink("rules.graphml")
```

Index

- *Topic **file**
 - saveAsGraph, 7
- *Topic **hplot**
 - plot, 2
- *Topic **manip**
 - abbreviate, 2
- abbreviate, 2
- plot, 2, 7
- plot.igraph, 5
- saveAsGraph, 7
- scatterplot3d, 5
- seriate, 5
- tkplot, 5
- write.graph, 7