An Association Rule Mining Infrastructure for the R Data Analysis Toolbox

Michael Hahsler[1] **and Kurt Hornik**[2]

[1] Department of Information Systems and Operations
Vienna University of Economics and Business Administration
[2] Department of Statistics and Mathematics
Vienna University of Economics and Business Administration

Presented at the 30th Annual Conference of the German Classification Society
Berlin, March 9, 2006

Motivation

- The aim of association rule mining is to discover *interesting patterns* (e.g., association rules) in “large” databases containing *transaction data.*

- To support association rule mining in R, we need a suitable infrastructure which provides:

  1. Efficient handling transaction data and patterns.

  2. Capabilities to analyze and manipulate transaction data and patterns.

  3. Mining algorithms.

  4. Measures of interestingness.

Such an infrastructure is provided by **arules***.*

Outline of the Talk

1. Transaction data and association rules

2. The **arules** infrastructure

3. Example: Market basket analysis

Transaction Data

Example of market basket data:

transaction ID	items
1	milk, bread
2	bread, butter
3	beer
4	milk, bread, butter
5	bread, butter

transactions	milk	bread	butter	beer
1	1	1	0	0
2	0	1	1	0
3	0	0	0	1
4	1	1	1	0
5	0	1	1	0

Formally, let $I = \{i_1, i_2, \ldots, i_n\}$ be a set of n binary attributes called *items*. Let $\mathcal{D} = \{t_1, t_2, \ldots, t_m\}$ be a set of *transactions* called the *database*. Each transaction in $\mathcal{D}$ has an unique transaction ID and contains a subset of the items in I.

Transaction Data (2)

Transaction data can originate from various sources, e.g.:

- *POS-systems* collect large quantities of records (transactions) containing the products/product categories purchased during a shopping trip (*Market Baskets*).

  Used by retailers for *Market Basket Analysis* for, e.g., segmentation, cross-selling opportunities (Russell et al. 1997; Berry & Linoff 1997)

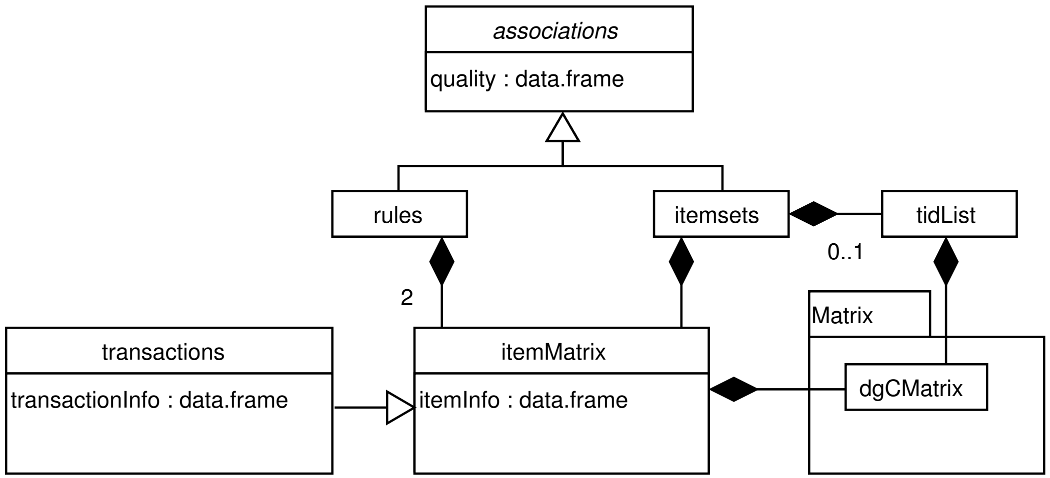
- Categorical and metric attributed of other data sources (e.g., *survey data*) can be mapped to binary attributes (Piatetsky-Shapiro 1991; Hastie et al. 2001).

  Used to discover interesting relationships between values of the attributes (e.g., between a certain age group and high income).

Association Rules

- A *rule* is defined as an implication of the form $X \Rightarrow Y$ where $X, Y \subseteq I$ and $X \cap Y = \emptyset$. The sets of items (for short *itemsets*) X and Y are called *antecedent* (left-hand-side or lhs) and *consequent* (right-hand-side or rhs) of the rule.
- To select "interesting" *association rules* (Agrawal et al. 1993) from the set of all possible rules minimum constraints for two measures are used:
  - The *support* $\mathrm{supp}(X)$ of an itemset X is defined as the proportion of transactions in the database which contain the itemset.
  - The *confidence* of a rule is defined $\mathrm{conf}(X \Rightarrow Y) = \mathrm{supp}(X \cup Y)/\mathrm{supp}(X)$.
- Typical rule: {bread, milk} $\Rightarrow$ {butter} ($\mathrm{supp} = 0.05, \mathrm{conf} = 0.6$)
- Efficient algorithms to find all association rules given the constraints are, e.g., Apriori, Eclat.

The arules Infrastructure



Simplified UML class diagram implemented in R (S4)

- Uses the *sparse matrix representation* (from package **Matrix** by Bates & Maechler (2005)) for transactions and associations.
- *Abstract associations class* for extensibility.
- Interfaces for *Apriori* and *Eclat* (implemented by Borgelt (2003)) to mine association rules and frequent itemsets.
- Provides *comprehensive analysis and manipulation capabilities* for transactions and associations (subsetting, sampling, visual inspection, etc.).

Example: Market basket analysis

Data Set

- 1 month (30 days) of real-world POS transaction[1] data from a typical local grocery outlet.
- Aggregated to product categories (e.g., “popcorn”).
- 9835 transactions with 169 different categories.

Goal of the store manager

- To obtain segment specific association rules to support promoting the product category “beef”.

[1] The data set included in package **arules** under the name `Groceries`.

Example: Segmentation

Find subsets of the database which represent different types of shopping behavior (e.g., small baskets at lunch time and rather large baskets on Fridays)

```
> library("arules")
> data("Groceries")

> s <- sample(Groceries, 2000)
> d <- dist(as(s, "matrix"), method = "binary")
```

For segmentation we use Partitioning Around Medoids (PAM) from package **cluster** (Maechler 2006) with $k = 8$.

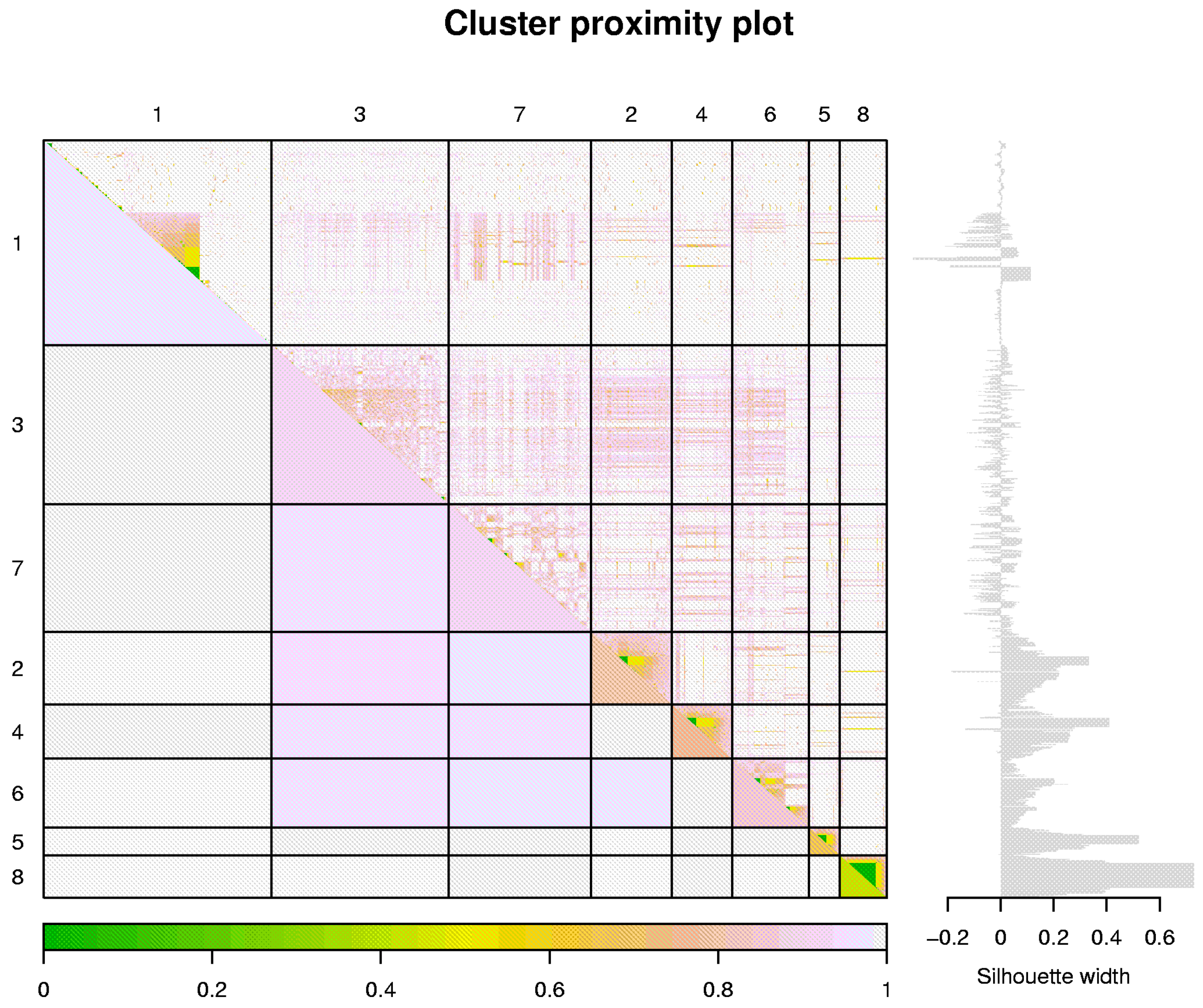
```
> library("cluster")
> labels <- pam(d, k = 8, cluster = TRUE)
```

Example: Segmentation (2)

Visual inspection with (re-ordered) dissimilarity matrix shading.

```
> library("cba")
> clu <-
+   cluproxplot(d,
+      labels)
```

(in package **cba** by Buchta & Hahsler (2006))



Example: Segmentation (3)

To predict labels for the whole data set based on the clustered sample, we use the nearest neighbor approach. Cross-distances are, e.g., implemented as the function `dists()` in package **cba**.

```
> xd <- dists(as(Groceries, "matrix") == 1,
+     as(s, "matrix") == 1, method = "binary")
> allLabels <- labels[max.col(-xd)]
```

We use the labels for all transactions (`allLabels`) to generate the list C of transaction data sets, one for each cluster.

```
> C <- split(Groceries, allLabels)
```

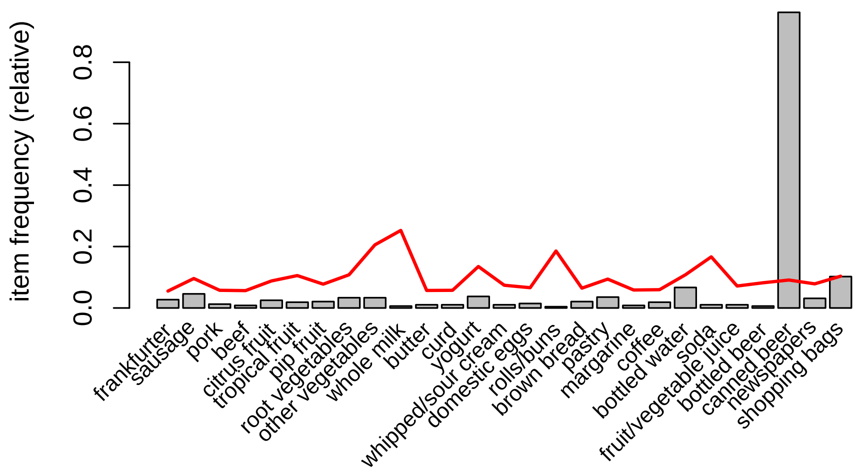
Example: Segmentation (4)

Inspect cluster profiles of two distinct clusters:

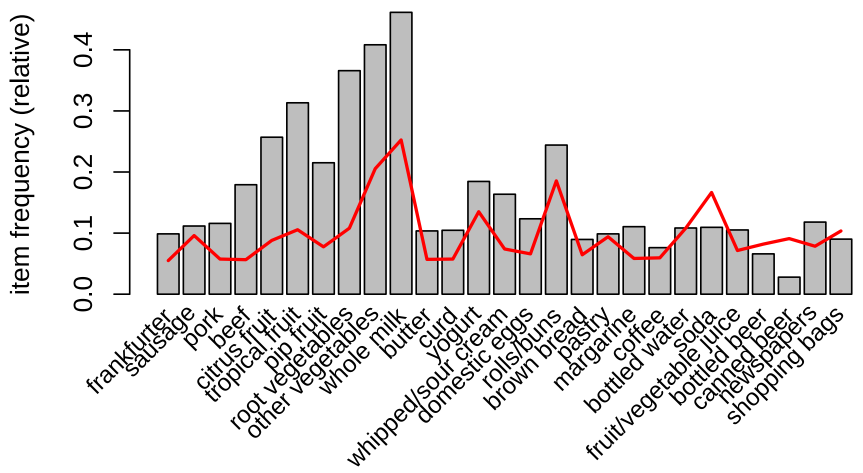
1. Cluster 8: Most compact cluster (highest avg. silhouette width)

2. Cluster 3: Largest average basket size

```
> itemFrequencyPlot(C[[8]], population = s, support = 0.05)
> itemFrequencyPlot(C[[3]], population = s, support = 0.05)
```



Cluster 8



Cluster 3

Example: Mining Association Rules

We mine association rules from the transactions in cluster 8 with a minimum support of 0.5% and a minimum confidence of 20%.

```
> rules <- apriori(C[[8]], parameter = list(support = 0.005,
+      confidence = 0.2), control = list(verbose = FALSE))
> rules

set of 13255 rules
```

In a second step, we find the rules which have the product category “beef” in the right-hand-side (equivalent to rule template * $\Rightarrow$ {beef}).

```
> beefRules <- subset(rules, subset = rhs %in% "beef")
> beefRules

set of 268 rules
```

Example: Mining Association Rules (2)

The store manager can now analyze the found 268 rules. As an example, we show the 3 rules with the highest confidence values.

```
> inspect(head(SORT(beefRules, by = "confidence"), n = 3))

  lhs                     rhs      support confidence  lift
1 {sausage,
    root vegetables,
    butter}            => {beef} 0.005411     0.6250 3.438
2 {pork,
    berries}           => {beef} 0.005411     0.5263 2.895
3 {root vegetables,
    whole milk,
    butter,
    rolls/buns}        => {beef} 0.005952     0.5238 2.881
```

Conclusion

The main properties of the flexible **arules** infrastructure are:

- Efficient storage of transaction data and associations in sparse matrix representation.
- A rich set of functions for analyzing and manipulation transaction data and associations.
- Interfaces to fast mining algorithms (Apriori, Eclat).
- Extensible class structure (e.g., for adding new types of associations).

The **arules** infrastructure provides the foundation for new applications. For example,

- computations with sets of associations,
- clustering itemsets or rules (Strehl et al., 1999),
- experiments with probabilistic models of transaction data (mixture models; Cadez et al. 2001).