Association Rules and the Negative Binomial Model

Seminar: Statistical Learning

Michael Hahsler
hahsler@ai.wu-wien.ac.at

Dept. of Information Business, WU-Vienna

Vienna, April 29, 2004

Contents

Motivation: Recommender Systems

- Produce *item-to-item recommendations* for Web Sites (e-commerce).
  - "Customers who bought these items also bought ..."
  - Displaying recommendations is virtually without additional cost.
  - Recommendations can help to simulate a virtual "shopping experience."
  - Shopper can be anonymous (no shopping history known)
- Recommendations based on online transaction data:
  - Purchases in Web stores (e.g., Amazon).
  - Document downloads in digital libraries (e.g., Elsevier's science direct).
  - Browsing a directory service (e.g., Google Directory, dmoz).

Association Rules: Problem definition

- Mining association rules from market basket data was first introduced by Agrawal et al. [1].

- The problem is to mine implications of the form $X \Rightarrow Y$ from a data base. where $X, Y \subseteq I$ and $X \cap Y = \emptyset$ are called the antecedent and the consequent of the rule.

- The data base is a set of transactions $\mathcal{D} = \{T_1, T_2, ..., T_j\}$ where each transaction contains a subset of the set of the available items $I = \{i_1, i_2, ..., i_n\}$.

- Measures of *significance* and *interest* are assigned to itemsets and rules with the aim to select only rules that satisfy constraints based on these measures.

Assoc. Rules: Measures of significance and interest

For the definitions we use estimated probabilities. For $Z \subseteq I$

$$P(Z) = \frac{count(Z)}{|\mathcal{D}|}$$

where $count(.)$ denotes the number of occurrences of an itemset and $|\mathcal{D}|$ is the number of transactions in the data base.

Agrawal et al. [1] define two measures for association rule mining:

$$supp(Z) = P(Z)$$

$$conf(X \Rightarrow Y) = P(Y \mid X) = \frac{P(X \cup Y)}{P(X)} = \frac{supp(X \cup Y)}{supp(X)}$$

Support and confidence are often also used as the absolute number of transactions (e.g., $supp^{abs}(Z) = count(Z)$)

Assoc. Rules: The minimum support constraint

An itemset $Z \subseteq I$ is only considered *significant* if

$$supp(Z) \geq \sigma$$

where σ is a user defined minimum support constraint. Z is then called a *frequent itemset* or *large itemset.* $\mathcal{F} = \{Z \subseteq I | supp(Z) \geq \sigma\}$ is the set of all frequent itemsets.

Rational:

- Items that appear more often in the data base are more important (e.g., they are responsible for a higher sales volume).
- Support is *downward closed (antimonotonicity)* and therefore can be used for reducing (pruning) the search space $\mathcal{P}(I)$ (search tree).

Problems:

- Rare item problem (infrequently purchased expensive items contribute most to the store's overall earnings).
- σ is set arbitrarily without knowledge of error rates.

Assoc. Rules: Minimum confidence constraint

A minimum confidence constraint γ is used to generate only *interesting* rules from the frequent itemsets with

$$conf(X \Rightarrow Y) \geq \gamma$$

where $Z \in \mathcal{F}, X \subset Z$ and $Y = Z \setminus X$.

Rational: conditional probability, directed

Problems:

- Sensitivity to the frequency of the consequent (a higher count for Y directly translates into a higher confidence value).

- γ is also set arbitrarily.

A simple stochastic item usage model

Base rule mining on a stochastic item usage model because:

- Strong regularities were found in transaction data (e.g., market baskets, web usage).

- Transaction data is known to have skewed distributions (i.e., problems with support and confidence).

- The model provides estimates of error rates (percentage of accepted spurious rules).

We suggest to use a simple and well-known mixture model for count data (Gamma-Poisson model, NB model) as a benchmark to detect rules.

A simple stochastic item usage model (cont.)

- Each item $i \in I$ has a latent rate λ at which the item is used.

- Over all items this rate varies according to a continuous random variable Λ.

- The distribution of R, the number of transactions the item i is used in the observed period, follows an independent Poisson process with the latent rate λ.

$$P(R = r | \Lambda = \lambda) = \frac{\lambda^{-r} e^{-\lambda}}{r!} \, for \, r = 0, 1, 2, ...$$

- The distribution of the number of transactions for all items is then a Poisson mixture model.

$$P(R = r) = \int_0^\infty \frac{\lambda^{-r} e^{-\lambda}}{r!} \, dG_\Lambda(\lambda) \, for \, r = 0, 1, 2, ...$$

A simple stochastic item usage model (cont.)

- Heterogeneity in the usage frequency among items is accounted for by the mixing distribution, a Gamma distribution with parameters $a > 0$ and $k > 0$.

$$f_\Lambda(\lambda) = \frac{e^{-\lambda/a}\lambda^{k-1}}{a^k\Gamma(k)} \; for \; \lambda > 0$$

- This results in a negative binomial (NB) distribution with parameters k (exponents) and $a = m/k$ (m represents the mean usage frequency).

$$P(R = r) = (1+a)^{-k}\frac{\Gamma(k+r)}{\Gamma(r+1)\Gamma(k)}\left(\frac{a}{1+a}\right)^r \; for \; r = 0, 1, 2, \ldots$$

  $P(R = 0)$ represents the proportion of items which were never used in the observed period.

A simple stochastic item usage model (cont.)

Although, the NB model (Gamma-Poisson model) simplifies reality considerably with its assumed Poisson processes and the Gamma mixing distribution, it is widely used in the literature for count data (see [3, pp. 223–224])

- accident statistics,
- birth-and -death processes,
- economics,
- library circulation,
- market research (repeat-buying theory),
- medicine and
- military applications.

Recently, it was also used in a similar form for Web usage [6].

Fitting the model: Datasets

We use 4 datasets:

- *WebView-1** and *WebView-2** contain several months of clickstream data for two e-commerce Web sites where each transaction consists of the product detail views during a session.

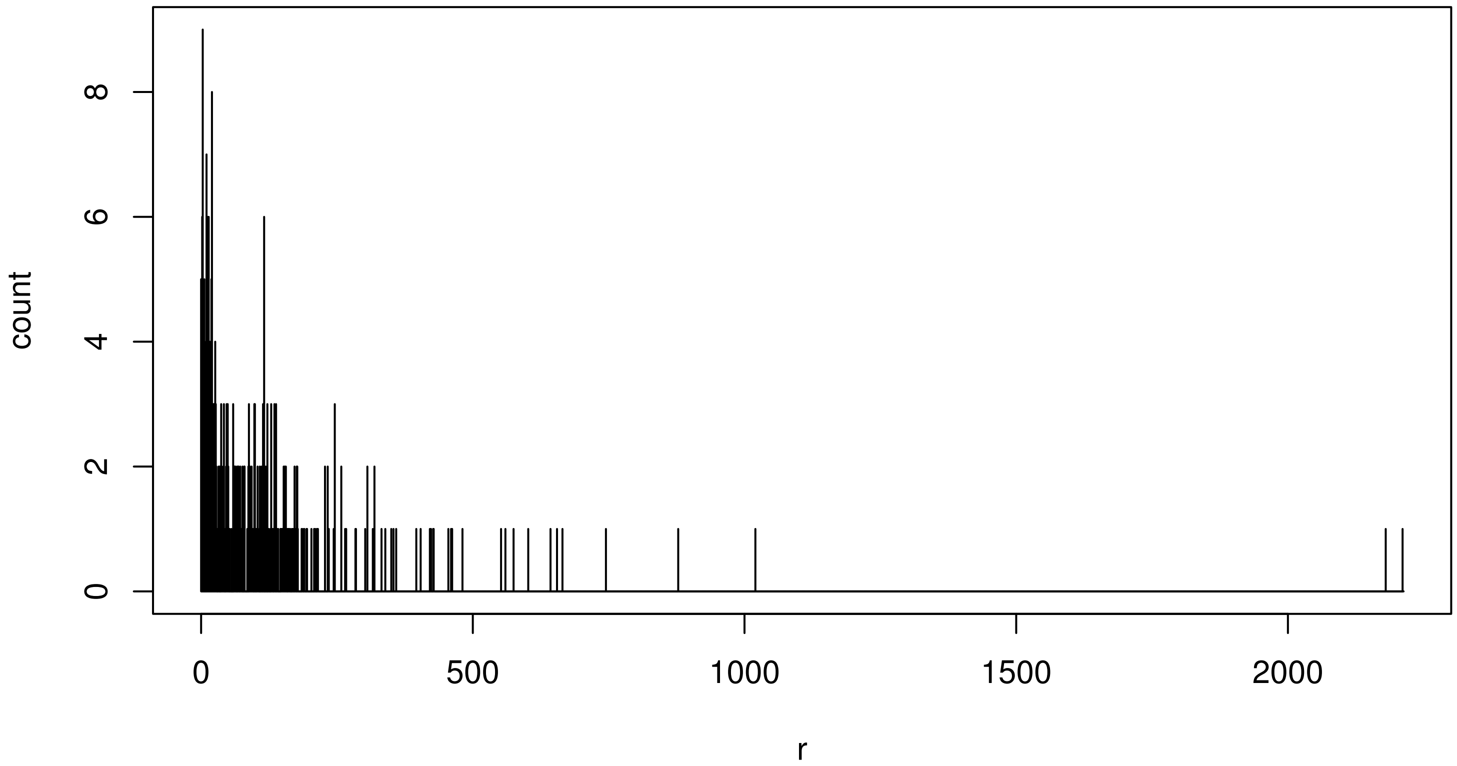
- *POS** is a point-of-sale dataset containing several years of data.

- *T10I4D100K* a widely used artificial dataset generated using the procedure described in Agrawal and Srikant [2].

* Provided by Blue Martini Software and used for the KDD Cup 2000 [4]

Fitting the model: Datasets (cont.)

Example: Observed counts $\hat{f}(.) * |I|$ for 20,000 transactions from WebView-1



Fitting the model: Estimation

Parameter estimation by the method of moments

$$\tilde{k} = \bar{x}^2/(s^2 - \bar{x})$$

$$\tilde{a} = \bar{x}/\tilde{k}$$

Challenges:

- Outliers in empiric data: Items with too high frequencies are not covered by the model.

- Zero-class is unknown: Transaction data does not contain information about items that are never used in the observation period.

Fitting the model: Estimation (cont.)

Proposed solutions for the estimation challenges:

- **Outliers:**
  We discard outliers by trimming a number of the items with the highest frequency from the three real-world datasets (e.g., 2.5% for the used datasets).

- **Unknown size of zero-class:**
  We iteratively used the method of moments to estimate the two parameters of the NB distribution and the Minimum χ^2 Estimation procedure to adapt the size of the zero-class.

Fitting the model: Results

	WebView-1	WebView-2	POS	T10I4D100K
Observed items	344	2,720	1,080	869
Trimmed items	9	80	55	*0*
Added zero-class	5	450	1,110	1
Used items	340	3,100	2,135	870
Item occurrences	34,146	70,391	53,740	201,883
$\bar{x}$	100.429	22.707	25.171	232.050
s^2	12027.676	1050.282	5104.761	50511.647
$\tilde{k}$	0.846	0.502	0.125	*1.071*
$\tilde{a}$	118.710	45.233	201.368	216.667
χ^2 p-value	0.0844	*0.00216*	0.0312	0.144

Samples with 20,000 transactions

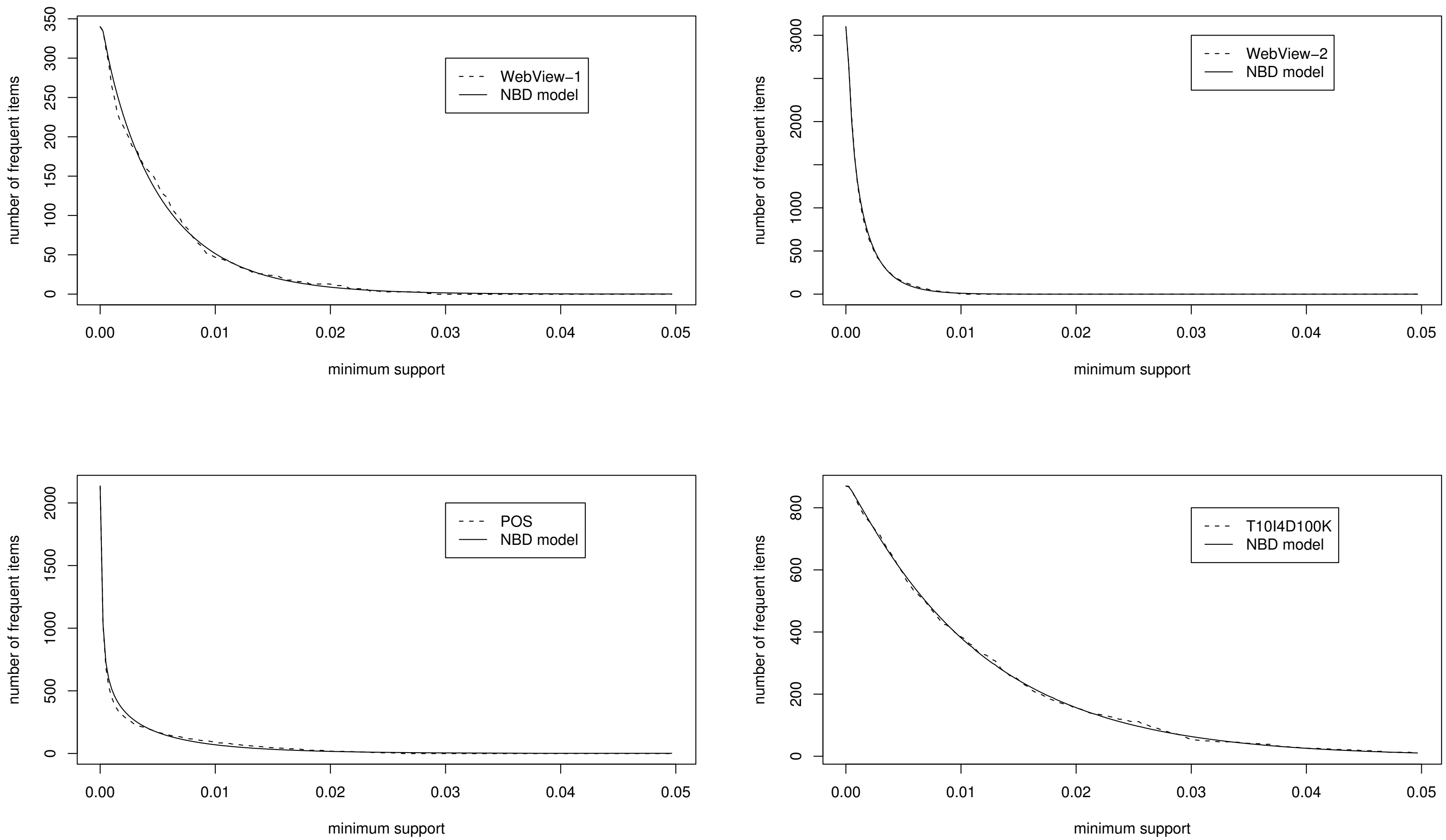
The model and support

- We established that $P(R = r)$, where R being a NB distributed random variable (with parameters k, a), models the probability of items being used $r = 0, 1, \ldots$ times in the dataset.

- Since the count r for an item i is its absolute support, R represents the distribution of support over all items $i \in I$.

- Therefore, the modeled proportion of items that pass a minimum support constraint σ^{abs} (frequent items) is given by $F_R(\sigma^{abs}) = P(R \geq \sigma^{abs})$.

The model and support (cont.)



The actual and predicted number of frequent items by minimum support.

Deriving a frequency constraint

- We now extend the model from single items to association rules
$$X \Rightarrow \{y_i\}$$
where $X \subseteq I$ is a fixed antecedent and $y_i \in I \setminus X$ represents all possible consequents.

- We can count the absolute support of these rules
$$supp^{abs}(X \cup \{y_i\}) = count(X \cup \{y_i\})$$
where we only need to consider the transactions that contain X.

- For all items y_i which are independent of the items in X, we expect that the distribution of the number of rules with a count r can be modeled by a random variable R_X with a NB distribution (assumption: $|X| \ll |I|$).

Deriving a frequency constraint (cont.)

We estimated already the parameters $\tilde{k}$ and $\tilde{a}$ for the distribution of R, representing the counts of all individual items.

For the rule model we need the parameter estimates for the NB-distributed random variable R_X.

Rescaling the parameters for X:

- The estimate scale parameter $\tilde{k}$ is not effected.
- The parameter $a = m/k$ has to be rescaled for the total number of possible counts in the transactions that also contain X relative to the number of possible counts in the whole dataset.

$$\tilde{a}' = \frac{\tilde{a}}{\sum_{T \in \mathcal{D}} |T|}$$

$$\tilde{a}_X = \tilde{a}' \sum_{\{T \in \mathcal{D} | T \supset X\}} |T \setminus X|$$

Deriving a frequency constraint (cont.)

- For rule mining we need to identify related items.

- If some items y_i are related with the items in X, these items will have a higher count in the transactions together with X than expected by the model, i.e., related items move towards the tail of the distribution.

- The task is to identify a count threshold σ_X^{abs} (an absolute minimum support on all rules with the antecedent X) that separates related consequents in the distribution's tail best from random items.

Deriving a frequency constraint (cont.)

Precision is a possible quality measure widely used for information retrieval and by the machine learning community [5]. Precision measures the proportion of predicted positive cases that are correct.

$$prec(\sigma_X^{abs}) = \frac{(1 - \hat{F}_X(\sigma_X^{abs})) - (1 - \tilde{F}_X(\sigma_X^{abs}))}{1 - \hat{F}_X(\sigma_X^{abs})} = 1 - \frac{1 - \tilde{F}_X(\sigma_X^{abs})}{1 - \hat{F}_X(\sigma_X^{abs})}$$

where $\tilde{F}_X(.)$ is the cumulative distribution function of the estimated random variable $\tilde{R}_X$ with parameters $\tilde{k}$ and $\tilde{a}_X$ and $\hat{F}_X(.)$ is the cumulative distribution function of the observations.

A suitable selection criterion for the threshold σ_X^{abs} is to allow only a percentage of falsely accepted rules. E.g., if for an application the maximum of acceptable spurious rules is 5% we can use the constraint minimum precision $\delta = 0.95$ to select σ_X^{abs}.

The task is to find for each X the consequents using a user defined precision threshold δ.

Deriving a frequency constraint: Example

```
# X={47961,47965}
# total items: 340
# k: 0.846, a: 0.159920927780706
# min precision: 0.95
# found r > 3
r   obs    model    prec
0   322  299.89726    -
1    11   34.98000    -
2     1    4.45142    -
3     0    0.58222 0.88811
4     2    0.07718 0.98515
5     1    0.01031 0.99702
6     2    0.00139 0.99947
7     0    0.00019 0.99978
8     1    0.00003 0.99997
# chosen consequents: 6

#Rules
{47961,47965} => {47953}, {47961,47965} => {47945}
{47961,47965} => {47973}, {47961,47965} => {47957}
{47961,47965} => {47949}, {47961,47965} => {47969}
```

Search space and downward closure

Minimum support possesses the downward closure property:
All subsets of a frequent itemset must also be frequent, i.e., a frequent itemset can only be constructed from frequent subsets.

This property is used to reduce the search space $\mathcal{P}(I)$ (which grows exponentially with $|I|$).

The model uses δ to chose an absolute minimum support σ_X^{abs} for all rules with the antecedent X. The chosen consequents are
$Y_X = \{y \in I \setminus X | supp^{abs}(X \cup \{y\}) \geq \sigma_X^{abs}\}$.

Generating new candidate antecedents by $X' = \{X \cup \{y\} | y \in Y_X\}$ guaranties that $supp^{abs}(X') \geq \sigma_X^{abs}$ for all X'.

At the same time for all the not chosen itemsets $X'' = \{X \cup \{y\} | y \in I \setminus Y_X\}$ we have $supp^{abs}(X'') < \sigma_X^{abs}$.

This follows directly from the downward closure property of support.

Mining algorithms

Depth-first search algorithm:

NB-DFS$(X, \mathcal{D}_X, |I|, \tilde{k}, \tilde{a}', \delta)$:

1. $\mathcal{L} = \emptyset$;
2. **for** all transactions $T \in \mathcal{D}_X$ **do**
3. **for** all $y \in T \setminus X$ **do**
4. **if** no tuple exists for y **then** add $\langle y, 1\rangle$ to set $\mathcal{L}$;
5. **else** $y.r$++ for tuple $\langle y, y.r\rangle$ in set $\mathcal{L}$;
6. **end**
7. **end**
8. $Y =$ NB-Select($\mathcal{L}, |I|, \tilde{k}, \tilde{a}', \delta$); // Select consequents
9. $R = \{\{X \Rightarrow y\}|y \in Y\};$
10. $C =$ NB-Gen(X, Y); // New antecedent candidates
11. **for** all $c \in C$ **do**
12. $\mathcal{D}_c = \{T \in \mathcal{D}_X | c \subseteq T\}$; // Conditional data base
13. $R_c =$ NB-DFS($c, \mathcal{D}_c, |I|, \tilde{k}, \tilde{a}', \delta$);
14. **end**
15. **return** $R \cup \bigcup_C R_c;$

Mining algorithms (cont.)

Select consequents:

NB-Select$(\mathcal{L}, |I|, \tilde{k}, \tilde{a}', \delta)$:
1. $r_{max} = 0; rescale = 0;$
2. **for** each tuple $\langle y, y.r \rangle \in \mathcal{L}$ **do**
3. $\quad n_{obs}[y.r]$++; // Frequency of observed counts
4. $\quad$ **if** $y.r > r_{max}$ **then** $r_{max} = y.r$; // Find maximum
5. $\quad rescale = rescale + y.r;$
6. **end**
7. **for** $(i = 0; i < r_{max}; i$++$)$ **do**
8. $\quad f_{NB}[i] = P(R_{NB} = i | k = \tilde{k}, a = \tilde{a}' * rescale);$
9. **end**
10. $f_{NB}[r_{max}] = P(R_{NB} \geq r_{max} | k = \tilde{k}, a = \tilde{a}' * rescale);$
11. $r = r_{max} + 1; precision = 1;$
12. **while** $(precision \geq \delta \wedge (r$--$) > 1)$ **do**
13. $\quad p = 1 - min\{|I| \sum_{i=r}^{r_{max}} f_{NB}[i] / \sum_{i=r}^{r_{max}} n_{obs}[i], 1\};$
14. **end**
15. **return** $\{y \in \mathcal{L} | y.r > r\}$; // Return set of consequents

Mining algorithms (cont.)

Generate new candidates using a global repository $\mathcal{R}$ to avoid visiting nodes (antecedents) several times:

NB-Gen(X, Y):
1. $C = \{c | y \in Y \wedge c = X \cup \{y\} \wedge c \notin \mathcal{R}\}$; // Also check the repository
2. **for** all $c \in C$ **do**
3. add c to $\mathcal{R}$;
4. **end**
5. **return** C;

Evaluation

1. Distribution of min. support over all rules and per antecedent size.

2. Impact of changing values of δ.

3. Algorithm complexity.

4. Quality evaluation: Comparison with the support-confidence framework (e.g., using ROC curves (Receiver Operating Characteristic) and *lift*).

Advantages and disadvantages

Advantages:

- Takes the structure of the dataset into account (count data has a skewed distribution).
- Uses a user set threshold on error rates rather than on counts.
- Chooses a suitable absolute support for each set of rules with the same antecedent which potentially gets smaller with antecedent size (deals better with the rare item problem).
- Directly generates rules without frequent itemsets.

Disadvantages:

- The model has to fit the data and parameters need to be estimated.
- The search space for rules is bigger than the search space for frequent itemsets.
- Downward closure cannot be applied to reduce the search space and, therefore, the concepts of maximal and closed itemsets are not applicable.

Open points and questions

1. NB parameter estimation (outliers, zero-class).

2. Downward closure property for antecedent generation.

3. Confidence bounds for count data.

4. Evaluation

References

[1] R. Agrawal, T. Imielinski, and A. Swami. Mining association rules between sets of items in large databases. In *Proc. of the ACM SIGMOD Int'l Conference on Management of Data*, pages 207–216, Washington D.C., May 1993.

[2] Rakesh Agrawal and Ramakrishnan Srikant. Fast algorithms for mining association rules in large databases. In Jorge B. Bocca, Matthias Jarke, and Carlo Zaniolo, editors, *Proc. 20th Int. Conf. Very Large Data Bases, VLDB*, pages 487–499, Santiago, Chile, Sept 1994.

[3] Norman L. Johnson, Samuel Kotz, and Adrienne W. Kemp. *Univariate Discrete Distributions*. John Wiley & Sons, New York, 2nd edition, 1993.

[4] Ron Kohavi, Carla Brodley, Brian Frasca, Llew Mason, and Zijian Zheng. KDD-Cup 2000 organizers' report: Peeling the onion. *SIGKDD Explorations*, 2(2):86–98, 2000.

[5] Ron Kohavi and Foster Provost. Glossary of terms. *Machine Learning*, 30(2–3):271–274, 1988.

[6] Sukekeyu Lee, Fred Zufryden, and Xavier Dreze. Modeling consumer visit frequency on the internet. In *34th Annual Hawaii International Conference on System Sciences (HICSS-34)-Volume 7*, 2001.