A Methodology for Filtering Association Rules

Alzira Faria
Instituto Superior de Engenharia do Porto
aff@isep.ipp.pt

Abstract

Basket data analysis is an important issue in the area of Artificial Intelligence and Decision Support Systems. Association rules are a model that represents co-occurrence of items in a transaction according to some support and confidence measures. However, sometimes the number of generated association rules is too large to be analyzed. A methodology is presented to highlight the strongest rules, using a filter. Experiment results show that this filter is efficient and capable of making basket data analysis easier to implement.

Keywords: association rules, filtering, artificial intelligence, decision support systems

Resumo

A análise de dados de cestos de compras é um assunto importante na área de Inteligência Artificial e Sistemas de Apoio à Decisão. As regras de associação são um modelo que representa co-ocorrência de itens numa transacção segundo determinados valores de suporte e confiança. No entanto, o número de regras geradas é, por vezes, suficientemente grande, dificultando a análise. Uma metodologia é apresentada para evidenciar as regras mais fortes, usando um filtro, preservando as restantes. Os resultados experimentais mostram que este filtro é eficiente e capaz de tornar a análise de dados de cestos de compras mais fácil de realizar.

Palavras-chave: regras de associação, filtragem, inteligência artificial, sistemas de apoio à decisão

1-Introduction

Knowledge Discovery in Databases (KDD) [Fayyad et al., 1996a] is a field that mixes the concepts of Artificial Intelligence and Decision Support Systems. Its main goal is to discover hidden useful information inside large databases, in order to transform static data into knowledge. This task is achieved by using algorithms that manipulate data and find either patterns inside data (description) or forecast unknown values (prediction), after some previous transformation of data (pre-processing).

The KDD process is divided into several steps that may be seen as cyclic [Fayyad et al., 1996b]. Before applying algorithms to data, we must select the subset of data where we want to perform the discovery. Then selected data is pre-processed to remove noise and outliers. The resulting data is transformed to fit a particular input for a given algorithm. According to [Pyle, 1999], this part is essential for good results on the KDD process. The application of algorithms to the

obtained data is called Data Mining (although this name is usually used to describe the whole process) [Witten & Frank, 2005; Berry & Linoff, 2004; Hand et al., 2001; Michalski et al., 1998]. Finally, results will be interpreted and presented, or we may integrate them and return to one of the previous phases.

The input of basket analysis is made up of special data, called basket data. After selecting the target data, and pre-processing, transformation plays a special role. Usually, each product sold in a market store is one record in a database table. In this case, we want to have all the items that were bought in a single transaction, which means that we must create a pivot table, where fields are all items we want to study, each record corresponds to a transaction, and values are binary data, representing the presence or absence of an item in the given transaction. Finally, in the Data Mining phase, specialized algorithms will be performed over the resulting basket data, and will produce a particular model called association rules [Agrawal et al., 1993]. The most widely used algorithm (which is the one that was used in this work) is Apriori [Agrawal & Srikant, 1994].

Association rules are particular representations of knowledge that are usually associated with the problem of basket analysis. In these problems, we try to discover useful information about the correlations between items in basket cases, such as "if customers buy the subset of items X then, under some support and confidence measures, they will also buy the subset of items Y".

The discovery of association rules performed by Apriori consists of extracting all the rules where both support and confidence are greater than given threshold values. The greater the number of available items the greater the diversity of transactions. This means that in such cases, support will usually be low, so we must also have a low threshold for support. On the other hand, confidence threshold should be high enough so that we can rely on the rule. However, each study is a particular case, so we must adjust these threshold values individually rather than fix single values for every dataset.

The most frequent problem with association rules is the adjustment of threshold values for support and confidence. Lower values will tend to produce more rules, while higher values can prevent important rules from being considered.

Several approaches were made, most of them from the Pareto curve analysis, considering the Support-Confidence plot [Bayardo & Agrawal, 1999; Brin et al., 1997; Liu et al., 1999; Silberschatz & Tuzhilin, 1996]. This approach is very intuitive and is quite simple to implement. It has also the advantage of being Apriori-independent, in the sense that it may be applied to the output of that algorithm.

A filter for reducing the number of association rules, based in the Pareto approach, is presented. There are two main results that must be discussed. One is the reduction in the set of generated rules: what is the percentage of rules that are obtained from the filter over the first Apriori set? The second one, which is more difficult, is the relevance of the rules obtained by the filter. Determining the relevance of the rules is somewhat ambiguous, so a pragmatic solution that was followed in the current work was to make a comparison with the decision rules generated by C5.0 [Quinlan, 1997].

In the next section the state-of-the-art about filtering association rules will be presented. In section 3, a new filter that is performed after the association rules are generated by Apriori to highlight the most important rules. In section 4, a case study will be shown, comparing the number of rules that are highlighted by the filter with the number of rules generated by Apriori. In section 5, analysis can easily be done with the resulting set of rules. A comparison with C5.0 will then be performed, in section 6. Finally, some conclusions will be discussed in section 7.

2-Association Rules

Formally, an association rule is expressed by the formula $X \Rightarrow Y$, where *X* and *Y* are mutually exclusive item sets ($X \cap Y = \varnothing$). The intuitive meaning of these rules is that transactions that contain set *X* tend to also contain set *Y*.

Each rule has two associated measures: support and confidence. Support is the ratio between the number of transactions that satisfy both *X* and *Y* and the total number of transactions *n*:

$$Support = \frac{|X \cup Y|}{n}$$

Confidence is the ratio between the number of transactions that satisfy both *X* and *Y* and the total number of transactions that satisfy *X*:

$$Confidence = \frac{|X \cup Y|}{|X|}$$

As was already referred, Apriori generates all the rules that satisfy both given support and confidence. We may ask ourselves: are there rules more important than others? The answer stated here is: all generated rules are important, but some are more important than others.
Consider a simple example of two association rules:

$$rule\ 1: A \cup B \Rightarrow C \quad (S_1, C_1)$$
$$rule\ 2: A \Rightarrow C \quad (S_2, C_2)$$

where *A*, *B* and *C* are sets of items, and S_i and C_i (i=1,2) are the respective support and confidence. It is easy to see that the second rule support is greater than or equal to the first, and it will only be equal if *B* occurs every time *A* and *C* occur. In the second rule, confidence is also greater than or equal to the first, so it will be a stronger rule. This means that *A* does not need *B* to imply *C*. But if support is the same and in the first rule, the confidence measure is greater than the second, so the first is the strongest. In the remaining case, where support and confidence take opposite directions, it is not so easy. In table 1 a summary of the comparison between the two association rules according to support and confidence is shown.

Table 1- Comparison between association rules according to support and confidence

Comparison	Strongest rule
$S_1 \leq S_2 \wedge C_1 \leq C_2$	rule 2
$S_1 = S_2 \wedge C_1 > C_2$	rule 1
$S_1 < S_2 \wedge C_1 > C_2$	Uncertain

This is a well-known relationship. In fact, this is a usual analysis that is made after the generation of association rules. It is important to make clear that both rule 1 and rule 2 are important, since they both satisfy support and confidence threshold values.

According to some authors, such as [Liu et al., 1999], if there are two rules with similar support and confidence, the simpler rule should be chosen. This means that in the last case of table 1, rule 2 should be considered the strongest rule.

Decision rules generated by C5.0 [Quinlan, 1997], are generated in a different way. A field of the database is chosen by considering a gain score. This score favors fields where all possible values have a similar number of records. As in association rules, one may consider a support value, the number of records that support the rule, and a confidence value, the percentage of correct classifications.

The correctness of the generated decision rules may be measured in several ways. The most common are:

- Splitting into two sets: a training set, used to generate the model, and the test set, for comparing predicted values with the real ones.
- Crossed validation: n tests with n subsets, corresponding to the test sets, with the remaining set being used as the training set.
- Leave one out: the same as cross validation, with n being the total number of records.

The comparison is made using a coincidence matrix (also called confusion matrix), where the main diagonal presents the correct prediction count. The wrong values are usually called false positives and negatives.

3-The New Association Rules Filter

The example in section 2 showed a simple case: one of the rules on left hand side contains the other. The filter that will now be presented (named FAR – Filtering Association Rules) will be applicable to all rules without restriction. This filter is based on the Pareto curve analysis, referred in section 1, and intends to reduce the number of association rules to consider in the analysis.

FAR does not interfere in the association rules generation, since it is performed after the output of Apriori. Support and Confidence threshold values are defined during the Apriori runtime and FAR will be constrained them.

After the generation of the association rules, two rules, *RS* (rule with best support) and *RC* (rule with best confidence) are chosen. If there is more than one with the best support or confidence, the rule where the other value is best will be chosen. If there is still more than one pick one of them randomly (for the filter, rules are not important, support and confidence are). If both rules have the same support and confidence, then all the rules that share the same values will be the strongest rules, and the filter stops immediately.

Consider (S_{RS}, C_{RS}) and (S_{RC}, C_{RC}) as the support and confidence of rules *RS* and *RC*, respectively, and S_{min} and C_{min} the minimum support and confidence. If *RS* and *RC* do not have the same support and confidence, we will have the case represented in figure 1.

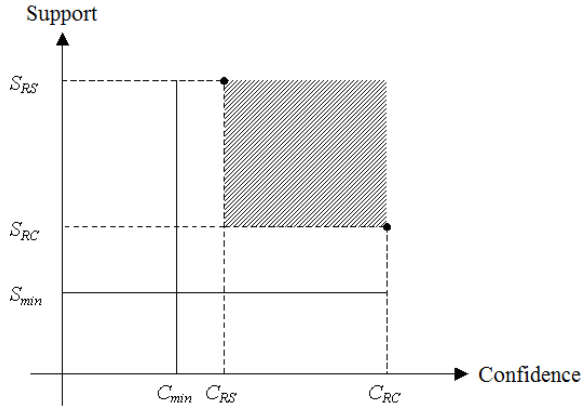

Figure 1- FAR first step

Only the rules that are inside the shadow range in figure 1, including the points of the best confidence and support, will be considered. FAR then considers another pair from the remaining rules, *RS2* and *RC2*, with the second best support and second best confidence, respectively, as in figure 2.

The process will continue until there are no more points to consider, or when all rules have the same support and confidence values. The obtained points refer to the rules with best support and confidence at each step. An important characteristic of these points is that sorting by support and by confidence are inverse. The set of association rules that are obtained by this filter are considered the best amongst all the rules. We may also want to sort these rules by order of importance. In this case, we may consider ordering by support or confidence or by some function combining both.

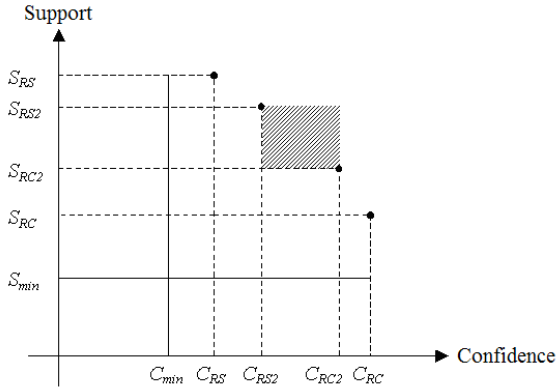

Figure 2- FAR second step

4- Case Study

To measure the efficiency of FAR, some experiments were made. An academic dataset with information about students' performance was our case study. The goal was to check correlations

between subjects of the same year and different semesters, and between subjects of consecutive years. The questions were:

1. Does performance of subjects from the first semester affect performance of subjects of the second semester?
2. Does performance of subjects from one year affect performance of subjects of the following year?

In figure 3 it is possible to see a set of association rules generated by Apriori from Clementine KDD tool [ISL, 1998]. Inside parenthesis there are the number of transactions that support the rule and respective percentage, and confidence. The threshold value of 20% for support was considered and 80% for confidence in all the experiments, in order to measure the efficiency uniformly. This may seem in contradiction with what was said in section 1, that we must adjust threshold values individually rather than fix them for every dataset, but in this case our goal is to test the filter not the rules. Figures 4, 5 and 6, show the evolution of FAR.

(1) X <= A (221:79.5%, 0.964)
(2) X <= B & A (218:78.4%, 0.968)
(3) X <= B & C & A (206:74.1%, 0.971)
(4) X <= B & C & D & A (198:71.2%, 0.98)
(5) X <= B & C & D (207:74.5%, 0.942)
(6) X <= B & C (217:78.1%, 0.931)
(7) X <= B & D & A (204:73.4%, 0.975)
(8) X <= B & D (216:77.7%, 0.931)
(9) X <= B (235:84.5%, 0.915)
(10) X <= C & A (207:74.5%, 0.971)
(11) X <= C & D & A (198:71.2%, 0.98)
(12) X <= C & D (213:76.6%, 0.925)
(13) X <= C (232:83.5%, 0.884)
(14) X <= D & A (205:73.7%, 0.976)
(15) X <= D (229:82.4%, 0.891)
(16) X <= E & A (208:74.8%, 0.966)
(17) X <= E & B & A (208:74.8%, 0.966)
(18) X <= E & B & C & A (201:72.3%, 0.97)
(19) X <= E & B & C & D (203:73.0%, 0.946)
(20) X <= E & B & C (211:75.9%, 0.934)
(21) X <= E & B & D & A (199:71.6%, 0.975)
(22) X <= E & B & D (210:75.5%, 0.933)
(23) X <= E & B (223:80.2%, 0.915)
(24) X <= E & C & A (201:72.3%, 0.97)
(25) X <= E & C & D & A (195:70.1%, 0.979)
(26) X <= E & C & D (209:75.2%, 0.928)
(27) X <= E & C (221:79.5%, 0.9)
(28) X <= E & D & A (199:71.6%, 0.975)
(29) X <= E & D (220:79.1%, 0.9)
(30) X <= E (240:86.3%, 0.858)
(31) X <= F & A (219:78.8%, 0.963)
(32) X <= F & B & A (216:77.7%, 0.968)
(33) X <= F & B & C & A (204:73.4%, 0.971)
(34) X <= F & B & C & D (206:74.1%, 0.942)
(35) X <= F & B & C (215:77.3%, 0.93)
(36) X <= F & B & D & A (203:73.0%, 0.975)
(37) X <= F & B & D (215:77.3%, 0.93)
(38) X <= F & B (231:83.1%, 0.918)
(39) X <= F & C & A (205:73.7%, 0.971)
(40) X <= F & C & D & A (197:70.9%, 0.98)
(41) X <= F & C & D (212:76.3%, 0.925)
(42) X <= F & C (228:82.0%, 0.89)
(43) X <= F & D & A (204:73.4%, 0.975)
(44) X <= F & D (226:81.3%, 0.898)
(45) X <= F & E & A (206:74.1%, 0.966)
(46) X <= F & E & B & A (206:74.1%, 0.966)
(47) X <= F & E & B & C (209:75.2%, 0.933)
(48) X <= F & E & B & D (209:75.2%, 0.933)
(49) X <= F & E & B (220:79.1%, 0.918)
(50) X <= F & E & C & A (199:71.6%, 0.97)
(51) X <= F & E & C & D (208:74.8%, 0.928)
(52) X <= F & E & C (218:78.4%, 0.904)
(53) X <= F & E & D & A (198:71.2%, 0.975)
(54) X <= F & E & D (218:78.4%, 0.904)
(55) X <= F & E (234:84.2%, 0.872)
(56) X <= F (260:93.5%, 0.835)

Figure 3- Association rules generated by Apriori

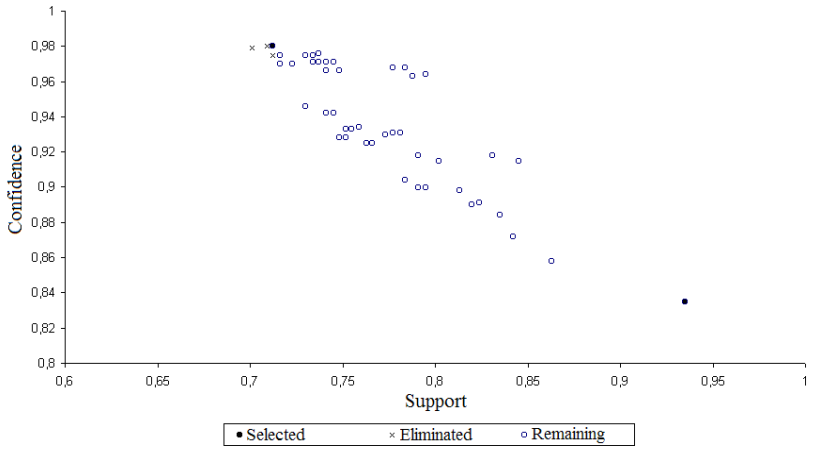

Figure 4- First step of FAR

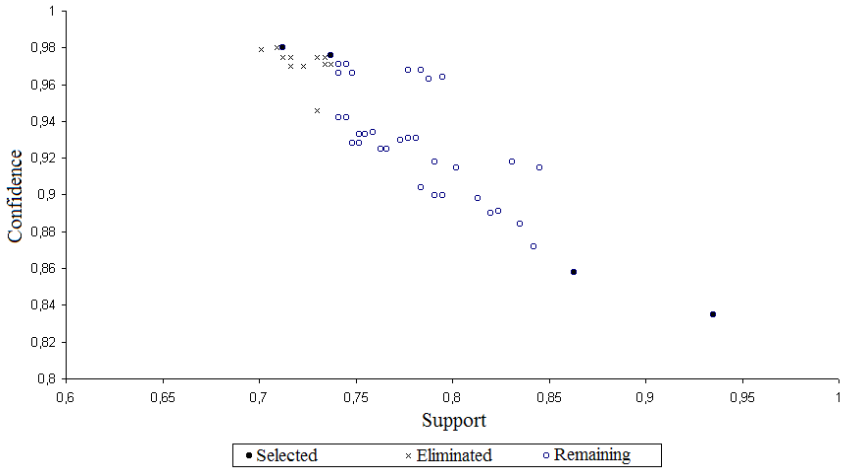

Figure 5- Second step of FAR

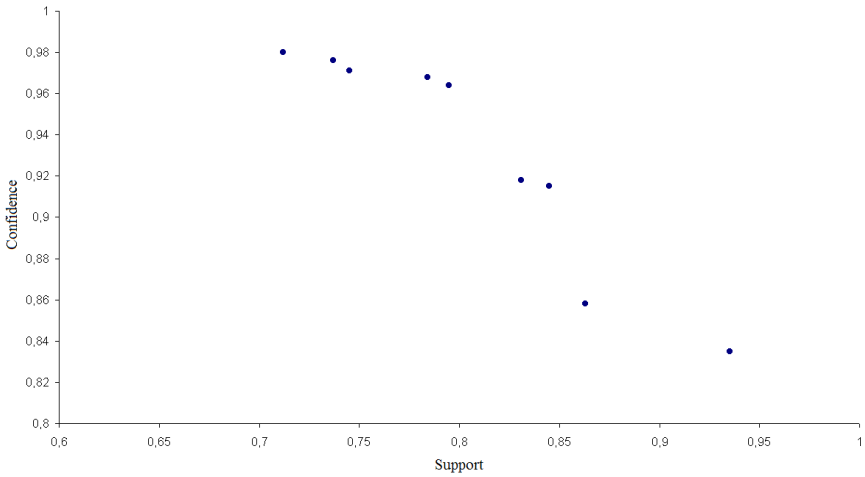

Figure 6- Final result

The final set of rules is shown in figure 7. As we may see from the example, the filter highlighted 10 rules from the original 56, which is much easier to analyze.

```
( 1) X <= B & C & D & A (198:71.2%, 0.98)
( 2) X <= C & D & A (198:71.2%, 0.98)
( 3) X <= D & A (205:73.7%, 0.976)
( 4) X <= C & A (207:74.5%, 0.971)
( 5) X <= B & A (218:78.4%, 0.968)
( 6) X <= A (221:79.5%, 0.964)
( 7) X <= F & B (231:83.1%, 0.918)
( 8) X <= B (235:84.5%, 0.915)
( 9) X <= E (240:86.3%, 0.858)
(10) X <= F (260:93.5%, 0.835)
```

Figure 7- Set of association rules after filtering

This example shows one of the sets generated for question 1, which is usually less than 100. Question 2 experiments usually generate more than 700 rules. In table 2, a summary of the results is presented.

Table 2- Summary of results with experiments

Question #	# initial rules	# rules after FAR	Percentage
1	1287	278	21.6%
2	33099	2842	8.6%
Total	**34386**	**3120**	**9.1%**

5-Support and Confidence Weight Analysis

The set of association rules obtained by the filter may now be analyzed. Consider the example in figure 7. The first two rules have the same support and confidence. Then, according to table 1, the second one is stronger, since B has no effect on support and confidence.

If we now consider rules 3, 4 and 5, we may notice that item A appears inside all the rules. One may take the hypothesis that item A is important, which is confirmed in rule 6. In fact, rule 6 has greater support and almost the same confidence (less than 1.2% when compared with rule 3, and less than 1.6% with rule 1, the one with the best confidence).

We believe, like most of the authors, that confidence is a better score to choose the strongest rule, since support tells us just how general the rule is. If we want to sort the rules as a combination of support and confidence we would say that confidence should have a higher weight. Table 3 shows some tests with weights for the rules in figure 7.

As we may observe in table 3, rule 6 is the only rule that wins in two of the scores: when support and confidence have the same weight (S+C) rules 6 and 8 are the best, and when the weight of confidence is twice the support (S+2C) rules 5 and 6 are the best. On the other hand, when the weight of support is twice the confidence, it is rule 10 that wins. Which of these results make sense?

Table 3.- Scoring rules

Rule #	Support(S)	Confidence(C)	S+C	2S+C	S+2C
1	71.2%	98.0%	1.69	2.40	2.67
2	71.2%	98.0%	1.69	2.40	2.67
3	73.7%	97.6%	1.71	2.45	2.69
4	74.5%	97.1%	1.72	2.46	2.69
5	78.4%	96.8%	1.75	2.54	**2.72**
6	79.5%	96.4%	**1.76**	2.55	**2.72**
7	83.1%	91.8%	1.75	2.58	2.67
8	84.5%	91.5%	**1.76**	2.61	2.68
9	86.3%	85.8%	1.72	2.58	2.58
10	93.5%	83.5%	1.77	**2.71**	2.61

Until now, it has not been said which subjects are being considered. In fact A is Calculus I, B is Linear Algebra, and X is Calculus II, all from Mathematics scientific area. C is Introduction to Programming, D is Circuits and Systems, E is Digital Electronics and F is Technical English, all from other scientific areas.

The dependence of X on A makes sense (rule 6). And it also makes some sense that rules 5 and 8 are very strong, since B is from the same scientific area as A and X. The explanation for the fact that rule 10 is so strong when support weight grows also makes sense: F is the subject where students achieve the best results.

6-Coincidence Matrix

Let us consider now a similar model, obtained with C5.0 (run inside Clementine): decision rules. Unlike association rules, decision rules are a set of deterministic rules that will lead us to a classification. There is no need for choosing the best rule, since they are applied in the right order and classify the unknown field. It is also possible to make an error analysis, splitting the data into two sets: the training set, which is used to generate the model, and the test set, which compares predicted values with the real ones.

For the example that is being followed, the decision rules that were generated by C5.0 inside Clementine, considering X as the unknown value, are in figure 8. The dataset was split into two equal sets: training and test.

This rule confirms that X result depends on the result of A. Notice that between brackets there is the number of records that support the rule and a correspondent confidence value. This confidence value is given by the Laplace ratio, which is $\frac{C+1}{T+2}$, where C is the number of true positives and T the number of records that support the rule. For instance, in the first example of figure 8, the confidence value is obtained from $\frac{108+1}{110+2} = \frac{109}{112} \approx 0.973$.

```
Rules for 1:
   Rule #1 for 1:
      if  A == 1
      then -> 1 (110, 0.973)

Rules for 0:
   Rule #1 for 0:
      if  A == 0
      then -> 0 (29, 0.806)

Default : -> 1
```

Figure 8- Decision rules obtained with C5.0.

After generating the decision rules, tests were made using the test set. The decision rules were applied to the test set and the predicted values were compared with the real values, resulting in the Coincidence Matrix (sometimes called Confusion Matrix) of figure 9, which shows that the generated model is very accurate (92.09% of correct predictions).

```
Results for output field X
   Comparing $C-X with X
      Correct   :      128    ( 92.09%)
      Wrong     :       11    (  7.91%)
      Total     :      139
      Coincidence Matrix
              $C-X
              0   1
      0        23  9
      1         2   105  Default : -> 1
```

Figure 9- Coincidence Matrix.

After obtaining all the results, comparisons were made in order to determine if association rules obtained after the filter have correspondence with the respective decision rules. Only datasets with more than 200 records were chosen, resulting in 24 comparisons. In 23 of the 24 cases (95.83%) there was a match.

Considering the tests of the decision rules over test sets, the correct prediction varied from 82.24% to 94.39%, with the average of 90.50%.

7- Conclusions

Results show that FAR algorithm is efficient, highlighting 21.6% of the association rules in sets of less than 100 rules, and 8.6% in sets of usually more than 700 rules. This also shows that when the size of the association rules set increases the performance of the algorithm is also better.

It is also shown from the comparison with C5.0 decision rules that the filter preserves the most important rules. In fact, in 95.83% of the cases, the set generated by the filter matches the

corresponding decision rules generated by C5.0. It is important to notice that the correct classification of C5.0 in the given dataset is very good (between 82.24% and 94.39%).

The filter chooses the most important rules, but all other rules may be considered in the analysis, since some correlations that are not as strong as the selected ones could also be important for supporting the decision. For instance, if we want to decide between two rules that were highlighted by the filter, it may be useful to look at the whole set generated by Apriori.

References

Agrawal, R., Imielinski, T., Swami, A., 1993. Mining Association Rules between Sets of Items in Large Databases. ACM SIGMOD Conference on Management of Data, 207-216. Washington, D. C.

Agrawal, R., Srikant, R., 1994. Fast Algorithms for Mining Association Rules. VLDB Conference. Santiago, Chile.

Bayardo Jr., R. J., Agrawal, R., 1999. Mining the Most Interesting Rules. KDD 1999: 145-154.

Berry, M. J. A., Linoff, G., 2004. Data Mining Techniques – For Marketing, Sales and Customer Support, Second Edition. John Wiley & Sons Ltd.

Brin, S., Motwani, R., Ullman, J., Tsur, S., 1997. Dynamic Itemset Counting and Implication Rules for Market Basket Data, In *Proc.of the 1997 ACM-SIGMOD Int'l Conf. on the Management of Data*, 255-264.

Fayyad, U., Piatetsky-Shapiro, G., Smyth, P., Uthurusamy, R., 1996. Advances in Knowledge Discovering and Data Mining. MIT Press, Cambridge, MA.

Fayyad, U., Piatetsky-Shapiro, G., Smyth, P., 1996. The KDD Process for Extracting Useful Knowledge from Volumes of Data. Communications of the ACM, 39 (11): 27-34.

Hand, D., Mannila, H., Smyth, P., 2001. Principles of Data Mining. MIT Press.

ISL, 1998. Clementine, User Guide, Version 5.0, Integral Solutions Limited.

Liu, B., Hsu, W., Ma, Y., 1999. Pruning and Summarizing the Discovered Associations, In *Proc. of the 5th International Conference on Knowledge Discovery and Data Mining*, KDD-99, 125-134.

Michalski, R. S., Bratko, I., Kubat, M., 1998. Machine Learning and Data Mining – Methods and Applications. John Wiley & Sons Ltd.

Pyle, D., 1999. Data Preparation for Data Mining. Morgan Kaufmann Publishers, Inc.

Quinlan, J. R., 1997. C5.0 Data Mining Tool. www.rulequest.com. Accessed January 2008.

Silberschatz, A., Tuzhilin, A., 1996. What Makes Patterns Interesting, In *Knowledge Discovery Systems. IEEE Trans. Knowl. Data Eng*. 8(6): 970-974.

Witten, I. H., Frank, E., 2005. Data Mining: Practical Machine Learning Tools and Techniques, Second Edition. Academic Press.