

Feature Selection-Ranking Methods In a Very Large Electric Database

Manuel Mejía-Lavalle, Guillermo Rodríguez-Ortiz, Gustavo Arroyo, Eduardo F. Morales¹

Instituto de Investigaciones Eléctricas, Gerencia de Sistemas Informáticos
Reforma 113, 62490 Cuernavaca, Morelos, México

¹ITESM - Cuernavaca

{mlavalle, gro, garroyo}@iie.org.mx, eduardo.morales@itesm.mx

Abstract. Feature selection is a crucial activity when knowledge discovery is applied to very large databases, as it reduces dimensionality and therefore the complexity of the problem. Its main objective is to eliminate attributes to obtain a computationally tractable problem, without affecting the quality of the solution. To perform feature selection, several methods have been proposed, some of them tested over small academic datasets. In this paper we evaluate different feature selection-ranking methods over a very large real world database related with a Mexican electric energy client-invoice system. Most of the research on feature selection methods only evaluates accuracy and processing time; here we also report on the amount of discovered knowledge and stress the issue around the boundary that separates relevant and irrelevant features. The evaluation was done using *Elvira* and *Weka* tools, which integrate and implement state of the art data mining algorithms. Finally, we propose a promising feature selection heuristic based on the experiments performed.

1 Introduction

Data mining is mainly applied to large amounts of stored data to look for the implicit knowledge hidden within this information. In other words, it looks for tendencies or patterns of behavior that allow to improve actual organizational procedures of marketing research, production, operation, maintenance, invoicing and others. To take advantage of the enormous amount information currently available in many databases, algorithms and tools specialized in the automatic discovery of hidden knowledge within this information have been developed; this process of non-trivial extraction of relevant information that is implicit in the data is known as Knowledge Discovery in Databases (KDD), where the data mining phase plays a central role in this process [1].

It has been noted, however, that when very large databases are going to get mined, the mining algorithms get very slow, requiring too much time to process the information and sometimes making the problem intractable. One way to attack this problem is to reduce the amount of data before applying the mining process [2]. In particular, the pre-processing method of feature selection applied to the data before mining has shown to be successful, because it eliminates the irrelevant or redundant attributes that cause the mining tools to become inefficient, but preserving the classification quality of the mining algorithm. Sometimes the percentage of instances correctly classified gets even higher

when using feature selection, because the data to mine are free of noise or data that cause that the mining tool to generate overfitted models [3].

In general, wrapper and filter methods have been applied to feature selection. Wrapper methods, although effective to eliminate irrelevant and redundant attributes, are very slow because they apply the mining algorithm many times, changing the number of attributes each execution time, as they follow some search and stop criteria [4]. Filter methods are more efficient using existing techniques such as decision trees algorithms, neuronal networks, nearest neighborhood, etc., that take into account dependencies between attributes. Another technique, called ranking method, uses some type of information gain measurement between individual attributes and the class, and it is very efficient [5]; however, because it measures the relevance of each isolated attribute, they cannot detect if redundant attributes exist, or if a combination of two attributes, apparently irrelevant when analyzed independently, can be transformed into relevant [6].

On the other hand, CFE (Federal Commission of Electricity in Mexico) faces the problem to accurately detecting customers that illicitly use energy, and consequently to reduce the losses due to this concept. At present time, a lot of historical information is stored in the Commercial System (SICOM), whose database was developed and it is maintained by CFE. SICOM was created mainly to register the users contract information, and the invoicing and collection data; this database has several years of operation and has a great amount of accumulated data (millions of records).

To make feasible the mining of this large database, in an effective and efficient way, in this paper we present an evaluation of different filter-ranking methods for supervised learning. The evaluation takes into account not only the classification quality and the processing time obtained after the filter application of each ranking method, but also it considers the discovered knowledge size, which, the smaller, the easier to interpret.

Also the boundary selection topic to determine which attributes must be considered relevant and which irrelevant is approached, since the ranking methods by themselves do not give this information, this decision is left without criterion. We propose an extension, simple to apply, that allows unifying the criterion for the attributes boundary in the different evaluated ranking methods.

Finally, based on the experimentation results, we propose a heuristic that looks for the efficient combination of ranking methods with the effectiveness of the wrapper methods. Although our work focuses on the SICOM data, the lessons learned can be applied to other real world databases with similar problems.

2 Related work

The emergence of Very Large Databases (VLDB) leads to new challenges that the mining algorithms of the 90's are incapable to attack efficiently. This is why new specialized mining algorithms for VLDB are required. According to [7], from the point of view of the mining algorithms, the main lines to deal with VLDB (scaling up algorithms) are: a) to design fast algorithms, optimizing searches, reducing complexity, finding approximate

solutions, or using parallelism; b) to divide the data based on the variables involved or the number of examples; and c) to use relational representations instead of a single table.

In particular, these new approaches in turn give origin to Data Reduction or Dimensional Reduction. Data Reduction tries to eliminate variables, attributes or instances that do not contribute information (or they do not contribute much information) to the KDD process, or to group the values that a variable can take (discretizing). These methods are generally applied before the actual mining is performed. Although in the 90's the pre-processing was minimum, and almost all the discovery work was left to the mining algorithm, every day we see more and more Data Pre-processing activities. This pre-processing allows the mining algorithm to do its work more efficiently (faster) and effectively (better quality).

In fact, the specialized literature mentions the *curse of dimensionality*, referring to the fact that the processing time of many induction methods grows dramatically (sometimes exponentially) with the number of attributes. Searching for improvements on VLDB processing power (necessary with tens of attributes and hundreds of thousands of instances), two main groups of methods have appeared: wrappers and filters.

The wrapper methods basic approach is to use the same induction algorithm to select the relevant variables and then to execute the classification task or mining process. The mining algorithm executes as many times as it changes the number of attributes for each run. With a 100 attribute dataset the total number of possible states and runs would reach 1.26×10^{30} , which tells us that to use an exhaustive method is out of consideration, except for databases with very few attributes.

On the other hand, filter methods use algorithms that are independent to the mining algorithm and they are executed previous to the mining step. Among filter methods are those algorithms for relevant variable selection, generally called feature selection, and the instance sampling methods, also known as sub sampling algorithms [8].

A great variety of filter methods exist for feature selection. Some authors consider the ID3 algorithm [9] (and its extensions) as one of the first proposed approaches to filter, even so ID3 is more used as a mining algorithm. Among the pioneering filter methods and very much cited are FOCUS [10], that makes an exhaustive search of all the possible attribute subgroups, but this is only appropriate for problems with few attributes, and RELIEF [11] that has the disadvantage of not being able to detect redundant attributes.

Koller [12] uses a distance metric called cross-entropy or KL-distance, that compares two probability distributions and indicates the error, or distances, among them, and obtains around 50% reduction on the number of attributes, maintaining the quality of classification and being able to significantly reduce processing times (for example, from 15 hours of a wrapper scheme application, to 15 minutes for the proposed algorithm). The final result is "sub optimal" because it assumes independence between attributes, which it is not always true. Piramuthu [6] evaluates 10 different measures for the attribute-class distance, using Sequential Forward Search (SFS) that includes the best attributes selected by each measure into a subset, such that the final result is a "better" attribute subset than the individual groups proposed by each method. However, the results are not compared with the original attribute set, and so it is not possible to conclude anything about the

effectiveness of each measure, and although SFS manages to reduce the search space, multiple mining algorithm runs varying the attribute subsets are necessary to validate the scheme, and this is computationally expensive.

SOAP is a method that operates only on numerical attributes [13] and has a low computational cost; it counts the number of times the class value changes with respect to an attribute whose values have been sorted into ascending order. SOAP reduces the number of attributes as compared to other methods; nevertheless it does not handle discrete attributes and the user has to supply the number of attributes that will be used in the final subset. Another filter scheme, SAPPP [5], handles continuous and discrete attributes; initially SAPPP selects an attribute subset and each time that increases the number of attributes uses a decision tree construction algorithm to evaluate if the added attributes in the subset are more relevant with respect to the previous tree. It verifies if they affect the classification quality (accuracy) and if they do not affect it, they are discarded (because they are irrelevant) and the process stops. A 30% reduction in processing time was obtained, maintaining the classification accuracy. In spite of everything, work must be done to solve how many instances to use at the beginning and the increment selection for each step.

Molina [14] tried to characterize 10 different methods to select attributes by measuring the impact of redundant and irrelevant attributes, as well as of the number of instances. Significant differences could not be obtained, and it was observed that, in general, the results of the different methods depended on the data being used. Stoppiglia [15] proposes to introduce an additional random variable to the database, such that, after the attribute ranking time, all those variables that obtained less scores than the random variable, will be considered irrelevant. This criterion represents an alternative to the statistical Fisher's test. The results show that the method obtains a good attributes selection, comparable to other techniques. This method is attractive because of simplicity, although more experiments are needed to prove its effectiveness, for example, it could be that most of the time the random variable manages only to eliminate or discriminate very few attributes (or none), so that the power to select attributes would be reduced. In section 4.3, we will explore this and other subjects.

Other proposals for feature selection explore the use of neural networks, fuzzy logic, genetic algorithms, and support vector machines [3], but they are computationally expensive. In general, it is observed that the methods that have been proposed: a) are verified with small, academic or simulated databases; b) obtain results that vary with the domain of the application; c) obtain greater quality of the result applying greater computational cost; d) depend on suitable tuning; and e) they do not evaluate the size of the extracted knowledge, which is a key factor to understand the phenomenon underlying the data.

3 The application domain

One the main CFE functions is to distribute to the costumers the electrical energy produced in the different generating plants in Mexico. Related to distribution, CFE faces

different problems that prevent it to recover certain amount of “lost income” from the 100% of the total energy for sale. At present CFE loses approximately 21% of the energy for distribution. These losses are mainly due to two kinds of problems: a) technical, and b) administrative. The technical energy losses are usually in the range of 10% and a great investment in new technologies would be needed in the distribution equipment to be able to reduce this percentage. The other 11% of the losses are due to administrative control problems, and they are classified in three categories of anomalies: a) invoicing errors, b) measurement errors, and c) illicit energy use or fraud. The first two have a minimum percentage impact so the big problem is the illicit use of energy, that is to say, people who steal the energy and therefore they do not pay for it.

CFE has faced this problem applying different actions (as to increase the frequency of measurement equipment readings of suspect customers, or to install equipment for automatic readings) and has managed to reduce the percentage due to illicit use losses, which represents a recovery of several million dollars. Since the problem has not been completely solved, it is important to attack it with other technologies and actions, using a knowledge discovery approach based on data mining to obtain patterns of behavior of the illicit customers. This alternative solution does not require a great deal of investment and it has been proven to be effective in similar cases, like credit card fraud detection [16].

The subject information to analyze is a sample of the SICOM database, a legacy system developed with the COBOL language, it contains around twenty tables with information about contracts, invoicing, and collection from customers across the nation. This system was not designed with the illicit users discovery in mind; nevertheless, it contains a field called *debit-type* in which a record is made if the debit is due to illicit use of energy. After joining three tables, including the one that has the *debit-type* field, a “mine” with 35,983 instances was obtained with the following attributes: *Permanent customer registry (RPU), Year, Month, debit-type, Digit, kWh, Energy, Cve-invoicing, Total, Status, Turn, Tariff, Name, Installed-load, Contract-load*, and others that altogether add up to 21 attributes. One of the values that the attribute *debit-type* can be assigned is “9”, which indicates an illicit use, and it is our class attribute. Various experiments were executed with this database to evaluate the different ranking methods as described next.

4 Evaluating ranking methods

4.1 Measuring the attributes degree of relevance

The application of filter-ranking methods to select features of a VLDB is adequate due to its low computational cost. We use *Elvira* [17] and *Weka* [18] tools, since they provide suitable and updated platforms for the easy execution of multiple experiments in a PC environment. In the presentation of the experiments the processing time has been left out because it was always very small, for example, *Elvira* obtains, in less of a second, the Mutual Information distance to measure the relevance of 21 attributes using 35,983 instances. The result is shown in the left column of Table 1.

Table 1. Ranking using *Elvira* (Mutual Information distance)

Traditional Ranking		Ranking with three random variables	
fctura	0.09097299304149882	fctura	0.09097299304149882
status	0.06121332572180206	status	0.06121332572180206
kwEen	0.051186334426340505	kwEen	0.051186334426340505
cCEto	0.045967636246832214	cCEto	0.045967636246832214
kwMen	0.0443061751909163	RAND3	0.04450328124055651
toMkw	0.04376718990743937	kwMen	0.0443061751909163
enrgia	0.04325196857770465	toMkw	0.04376718990743937
kwMcI	0.04308595013830481	enrgia	0.04325196857770465
toMcI	0.04302669641028058	kwMcI	0.04308595013830481
kwh	0.04259503495345594	toMcI	0.04302669641028058
total	0.042438776707532586	RAND2	0.04295118668801855
mes	0.04204718796227498	kwh	0.04259503495345594
toMcC	0.04163309856095569	total	0.042438776707532586
cIEen	0.038549970847028533	mes	0.04204718796227498
toMen	0.03831938680147813	toMcC	0.04163309856095569
cgInst	0.036173176514204305	RAND1	0.04031876955965204
cgCont	0.034291607355202744	cIEen	0.038549970847028533
cIMcC	0.02679884377613058	toMen	0.03831938680147813
anio	0.004512035977610684	cgInst	0.036173176514204305
tarifa	0.0010537446951081608	cgCont	0.034291607355202744
digito	7.321404042019974E-4	cIMcC	0.02679884377613058
		anio	0.004512035977610684
		tarifa	0.0010537446951081608
		digito	7.321404042019974E-4

Although in this case the attributes appear ordered according to their relevance, we lack of a uniform criterion to decide which attributes to select. We used the Stoppiglia criterion [15], but modifying it as follows: instead of using a single random variable, we added three, to observe how the ranking method maintains together, or not, the random variables in the set of ranked attributes, avoiding a possible bias introduced to the result by a single random variable, that in fact is a computational pseudo-random variable. The obtained result is shown in the right column of Table 1, where variables RAND3, 2 and 1 are the boundaries of the four subsets of attributes.

Following the same procedure, we applied different ranking methods to the database (a detailed explanation of the used “distances” can be found in [6] and [14]); the results are shown in Table 2. Also, the methods: Principal Component Analysis (PCA), Information Gain, Gain Ratio and Symmetrical were explored, and they produced similar results as Chi-Square, which means that they did not obtain a significant reduction on the number of attributes. From Table 2 we observe that, although some ranking methods agree in the selection of some attributes, in general, each method produces different attribute ordering, including the position for the three random variables. (This is a very interesting result, as we will see in Table 3).

Table 2. Application of different ranking measures

Euclidean distance	Matusita distance	Kullback-Leibler 1	Kullback-Leibler 2	Shannon entropy	Bhatta-charyya	Relief	OneR	Chi-Square
fctura	fctura	fctura	fctura	kwh	kwEen	anio	factra	factra
mes	kwEen	status	mes	enrgia	fctura	mes	status	status
cIMcC	kwMen	kwEen	status	total	kwMen	factra	anio	mes
anio	RAND3	cCEto	cgInst	tarifa	RAND3	digito	tarifa	kwEen
RAND3	status	RAND3	cgCont	cgInst	toMkw	RAND3	digito	kwMcI
tarifa	cCEto	kwMen	cCEto	cgCont	toMcI	RAND2	mes	kwh
digito	toMkw	toMkw	cIMcC	kwEen	enrgia	RAND1	cIMcC	toMcI
status	enrgia	enrgia	anio	toMcI	cCEto	status	cgCont	toMcC
RAND2	toMcI	kwMcI	kwEen	kwMen	total	cgInst	cgInst	total
cIEen	total	toMcI	RAND2	toMen	RAND2	tarifa	RAND1	toMen
cgInst	kwMcI	RAND2	RAND3	toMkw	kwMcI	cgCont	toMkw	enrgia
cgCont	RAND2	kwh	toMkw	kwMcI	toMcC	cCEto	RAND2	kwMen
RAND1	kwh	total	kwh	toMcC	RAND1	cIEen	cCEto	toMkw
cCEto	toMcC	mes	kwMen	cCEto	kwh	cIMcC	kwh	cCEto
toMcC	RAND1	toMcC	cIEen	cIEen	toMen	kwEen	toMcI	cIEen
kwMcI	toMen	RAND1	enrgia	status	mes	toMen	RAND3	cgInst
toMkw	mes	cIEen	total	cIMcC	status	total	total	cgCont
toMen	cIEen	toMen	kwMcI	anio	cIEen	toMcC	toMen	anio
kwMen	cgInst	cgInst	RAND1	RAND2	cgInst	toMcI	kwEen	cIMcC
toMcI	cgCont	cgCont	tarifa	RAND1	cgCont	kwh	enrgia	tarifa
kwEen	cIMcC	cIMcC	digito	RAND3	cIMcC	kwMcI	kwMen	RAND2
total	anio	anio	toMcC	digito	anio	kwMen	cIEen	RAND3
enrgia	tarifa	tarifa	toMcI	mes	tarifa	toMkw	kwMcI	RAND1
kwh	digito	digito	toMen	fctura	digito	enrgia	toMcC	digito

4.2 Performance Evaluation of the methods

In order to evaluate the methods, we applied the *J4.8* tree induction classifier (the Weka implementation of the last public version of *C4.5*) to the database "projected" on the attributes selected by each method. Table 3 shows the results. In all the cases, we always used the Weka's default parameters and the attributes of the first subset identified by the appearance of the first random variable (in section 4.3 we analyze this in more detail). The feature reduction column measures the number of attributes selected against the total number of attributes. The processing time is expressed in relation to the time required to obtain a tree that includes all the attributes of the database (complete case). The size of the discovered knowledge is measured by the number of leaves and the number of nodes of the induced tree. The classification quality appears as the percentage of instances correctly classified using the training data (accuracy) and also using a 10-fold cross validation test. A column is included that considers cost-benefit that it would be obtained if the discovered knowledge were applied by the organization, and assuming that each inspection has a cost of -2.5 units and that the obtained benefit of a correct prediction of

an illicit is of +97.5 units. The reported cost-benefit corresponds to the application of the above mentioned 10-fold cross validation test and it is calculated considering that the complete case obtains a 1000 units of benefit, and the results of the other methods are normalized with respect to the complete case.

Table 3. Evaluating ranking methods by inducing J4.8 trees

Method	Feature reduction	Time	Leaves / Nodes	Acc train / test	Cost-benefit (test)
Complete case	0	100	21 / 41	98.41 / 97.25	1000
Mutual Information	0.80	12	5 / 9	90.86 / 90.10	444
Euclidean distance	0.80	11	3 / 5	93.89 / 93.89	520
Matusita distance	0.86	8	2 / 3	90.58 / 90.21	507
Kullback-Leibler 1	0.80	11	5 / 9	90.86 / 90.10	444
Kullback-Leibler 2	0.57	14	17 / 33	98.26 / 97.50	1001
Shannon entropy	0.14	92	23 / 45	95.52 / 93.71	876
Bhattacharyya	0.86	9	2 / 3	90.18 / 90.21	507
Relief	0.80	12 + 9721	3 / 5	93.89 / 93.89	520
OneR	0.57	15	12 / 23	96.64 / 95.95	892

In Table 3, we observe that most of the methods obtain a reduction of the number of attributes greater than 0.50 and reduce the mining algorithm processing time in an order of magnitude; a special case is Relief, that unlike the other methods whose processing time is small, Relief requires a proportion of time 9721 times greater than the time required to induce the tree by using all the attributes. With respect to the size of the discovered knowledge it is observed that almost all the methods produce trees smaller than the complete case. On the other hand, although apparently all the methods do not affect too much on the accuracy of the discovered knowledge, the cost-benefit column highlights those methods that better impact on the prediction of the illicit energy use patterns.

4.3 Combination of ranking and wrapper methods

Although the ranking methods are very efficient, they have a flaw in that they do not take into account the possible interdependences between attributes. Observing the obtained results mention above, we propose a heuristic that looks for to overcome such a deficiency, combining the efficiency of the ranking methods, with the effectiveness of the wrapper methods. The heuristic involves the induction of a number of decision trees considering all subsets of attributes that a method produces (the subsets appear limited by the three random variables in Table 2). Applying the previous idea, we can observe, in a

computationally economic way, if some combination of attributes exists in the subsets that improves the obtained results as compared when using only the first attribute subset. For example, the application of KL-2 with three random variables produces three subsets. The induction trees produced by *J4.8* using the three subsets and a combination of these subsets are shown in Table 4. It is observed that, for this case, it does not exist a combination that significantly improves the results of the first subset, and this is why we can conclude that we have found a good solution, one that manages to reduce to the processing time and the knowledge size, without affecting the tree quality of prediction.

Table 4. Using feature subsets to induce *J4.8* trees

Feature subsets	Feature reduction	Time	Leaves / Nodes	Acc train / test	Cost-benefit (Test)
begin – RAND2	0.57	14	17 / 33	98.26 / 97.50	1001
RAND3–RAND1	0.66	12	1 / 1	79.42 / 79.45	-910
RAND1-end	0.76	11	1 / 1	79.42 / 79.42	-913
begin-RAND1	0.23	16	17 / 33	98.26 / 97.43	1001
RAND3-end	0.42	18	1 / 1	79.42 / 79.45	-910
begin-RAND2/RAND1-end	0.33	17	21 / 41	98.41 / 97.18	992

5 Conclusions and future work

The feature selection ranking methods are very efficient because they only need to calculate the relevance of each isolated attribute to predict the class attribute. The disadvantages of these methods are that no uniform criterion is provided to decide which attributes are more relevant than others, and that no mechanism is included to detect the possible interdependences between attributes. In this article the integration of three random variables to the database is proposed to avoid a possible bias introduced to the result if a single random variable is used. We observed that, although some ranking methods agree in the selection of some attributes, in general, each method produces different attribute ordering, including the position for the three random variables. This is a very interesting result. The three variables serve as subset boundaries and help to decide which attributes to select. Also, we propose to analyze the possible interdependences between attributes using the induction trees constructed on these subsets. These ideas have been proven to be successful in a real world electrical energy customer-invoice database. In the future these ideas are going to be applied to other databases and classifiers. In particular we are going to perform more simulations using the inclusion of multiple random variables to observe its utility like criterion within the feature selection area.

6 References

1. Frawley, W. et.al., Knowledge Discovery in DBs: An Overview, in Knowledge Discovery in Databases, Piatetsky-Shapiro, G. eds., Cambridge, MA, AAAI/MIT, 1991, pp. 1-27.
2. Pyle, D., Data preparation for data mining, Morgan Kaufmann, San Francisco, California, 1999.
3. Guyon, I., Elisseeff, A., An introduction to variable and feature selection, Journal of machine learning research, 3, 2003, pp. 1157-1182.
4. Kohavi, R., John, G., Wrappers for feature subset selection, Artificial Intelligence Journal, Special issue on relevance, 1997, pp. 273-324.
5. Leite, R., Brazdil, P., Decision tree-based attribute selection via sub sampling, Workshop de minería de datos y aprendizaje, Herrera, F., Riquelme, J. (eds), VIII Iberamia, Sevilla, Spain, Nov, 2002, pp. 77-83.
6. Piramuthu, S., Evaluating feature selection methods for learning in data mining applications, Proc. 31st annual Hawaii Int. conf. on system sciences, 1998, pp. 294-301.
7. Mitra, S., et.al., Data mining in soft computing framework: a survey, IEEE Trans. on neural networks, vol. 13, no. 1, January, 2002, pp. 3-14.
8. Domingos, P., When and how to sub sample, SIGKDD Explorations, Vol. 3, Issue 2, 2001, pp. 74-75.
9. Quinlan, J., Unknown attribute values in ID3, Int. conf. Machine learning, 1989, pp. 164-168.
10. Almuallim, H., Dietterich, T., Learning with many irrelevant features, Ninth nat. conf. on AI, MIT Press, 1991, pp. 547-552.
11. Kira, K., Rendell, L., The feature selection problem: traditional methods and a new algorithm, Tenth nat. conf. on AI, MIT Press, 1992, pp. 129-134.
12. Koller, D., Sahami, M., Toward optimal feature selection, Int. conf. on machine learning, 1996, pp. 284-292.
13. Ruiz, R., Aguilar, J., Riquelme, J., SOAP: efficient feature selection of numeric attributes, VIII Iberamia, workshop de minería de datos y aprendizaje, Spain, 2002, pp. 233-242.
14. Molina, L., Belanche, L., Nebot, A., Feature selection algorithms, a survey and experimental evaluation, IEEE Int. conf. on data mining, Maebashi City Japan, 2002, pp. 306-313.
15. Stoppiglia, H., Dreyfus, G., et.al., Ranking a random feature for variable and feature selection, Journal of machine learning research, 3, 2003, pp. 1399-1414.
16. S. Stolfo, W. Fan, W. Lee, A. Prodromidis, and P. Chan. Credit card fraud detection using meta-learning: Issues and initial results. Working notes of AAAI Workshop on AI Approaches to Fraud Detection and Risk Management, 1997.
17. www.ia.uned.es/~elvira/, 2003.
18. www.cs.waikato.ac.nz/ml/weka, 2003.