

DATA MINING LEARNING MODELS AND ALGORITHMS FOR MEDICAL APPLICATIONS

Plamena Andreeva¹, Maya Dimitrova¹, Petia Radeva²

¹ *Institute of Control and System Research, Bulgarian Academy of Sciences,
“Acad. G. Bonchev” str., Bl.2, P.O.Box 79, 1113 Sofia, BG,*

plamena@icsr.bas.bg, dimitrova@icsr.bas.bg

² *Computer Vision Center, Autonomous University Barcelona, Barcelona, Spain
petia@cvc.uab.es*

Abstract. Learning models are widely implemented for prediction of system behaviour and forecasting future trends. A comparison of different learning models used in Data Mining and a practical guideline how to select the most suited algorithm for a specific medical application is presented and some empirical criteria for describing and evaluating learning methods are given. Three case studies for medical data sets are presented and potential benefits of the proposed methodology for diagnosis learning are suggested.

Keywords: Data Mining, Machine Learning, Prediction, Rule extraction, Classification

1 INTRODUCTION

Information technology development has grown rapidly in the last years. The Data Mining (DM) technique has become an established method for improving statistical tools to predict future trends [1]. The wide range of applications from business tasks to scientific tasks has lead to a huge variety of learning methods and algorithms for rule extraction and prediction. The general tasks of classification, regression, clustering, or deviation analysis have a large number of solutions such as neural networks, decision tree learners, rule learners or Bayesian networks.

The aim of this study is to investigate different learning models and to give a practical guideline how to select the most suited algorithm for a specific application. Some of the well-known machine learning (ML) systems are investigated and different classification methods built in WEKA [5], See5 [3] and WizWhy [2] are tested with 3 medical datasets. Then practical criteria for evaluating different learning models are given and finally the results from our experiments are discussed.

1.1 Problem Formulation

In practical applications one has to decide which models and parameters may be appropriate for diagnosis and prediction problems. An algorithm proven useful for a medical database may show not to be useful in a cooperate database. The approaches used for data analysis are different: traditional statistical methods, neuronal nets (BrainMaker, NeuroShell), case-based reasoning (nearest neighbour), decision trees (See5/C5.0, Clementine), genetic methods (PolyAnalyst), machine learning algorithms and classifiers (WEKA).

For medical diagnosis there are many expert systems based on logical rules for decision making and prediction. The most precise rules are often with low completeness as they are special cases and do not bring any help in prediction and future trends. The real medical applications are systems with imprecision and uncertainty in logical rules. This is why fuzzy logic is well suited for decision-making and rule extraction. In the considered DM programs only See5/C5.0 and WEKA have classification with fuzzy threshold.

2 CLASSIFICATION METHODS

Usually, classification is a preliminary data analysis step for examining a set of cases to see if they can be grouped based on "similarity" to each other. Data analysis methods vary on the way how they detect patterns. The ultimate reason for doing classification is to increase understanding of the domain or to improve predictions compared to unclassified data.

Given a classification and a partial observation, one can always use the classification to make a statistical estimate of the unobserved attribute values and as the departure point for constructing new models, based on user's domain knowledge.

- The *decision-tree method* (used in systems CART, See5, Clementine) like the *nearest-neighbours* method, exploits clustering regularities for construction of decision-tree representation. It shows implicitly which variables are more significant with respect to classification decisions. The decision tree learning method requires the data to be expressed in the form of classified examples.

- The method of *Memory Based Reasoning* also called the *nearest neighbor method* finds the closest past analogs of the present situation and chooses the same solution. Such systems demonstrate good results in vastly diverse problems. Such systems do not create any models or rules summarizing the previous experience.

- *Genetic Algorithms* are powerful technique for solution of various combinatorial or optimization problems. They are more an instrument for scientific research rather than a tool for generic practical data analysis.

- *Nonlinear Regression Methods* (NR) are based on searching for a dependence of the target variable in the form of function. Such methods provide solutions with a larger statistical significance than neural networks do. This method has better chances of providing reliable solutions in medical diagnostics applications.

- *Support Vector Machines* methods are based on Structural Risk minimization principle from statistical learning theory. They generalise better than NN.

- *Maximum Likelihood Estimation (MLE)* deals with finding the set of models and parameters that maximises this probability. Unfortunately, MLE fails to provide a convincing way to compare alternate classifications that differ in class models or the number of classes. With statistical classification the usual procedure is to select the class, that most probably generates the partial observation and then use that class' distributions for the unknowns. This is generally satisfactory only if the probability of the chosen class is representative.

3 TECHNIQUES FOR FORECASTING

The Machine learning tools have been applied in a variety of medical domains to help solve diagnostic and prognostic problems. These tools enable the induction of diagnostic and prognostic knowledge in the form of rules or decision trees.

AutoClass C [4] is unsupervised Bayesian classification system that seeks a maximum posterior probability classification. In this program default class models are provided. AutoClass finds the set of classes that is maximally probable with respect to the data and model. Prediction requires having a training classification and a test database.

See5 program [0] uses decision tree (DT) for data analyses. The classifiers produced by See5 work with misclassification costs or penalties. The program can construct classifiers expressed as *decision trees* or as sets of *rules*. It uses fuzzy threshold for classifier construction. The experiments show that on average 10-classifier boosting reduces the error rate for test cases by about 25%.

The program system WizWhy [2] produces rules using "association rules" method. One of the main challenges of such a method is to validate each possible relationship. To calculate the *expected probability* WizWhy measures the dependency between symptom *A* and *B*. The output is a set of class descriptions and partial membership of the cases in the classes. This is very different from the

traditional problem formulation when the task is solved in two steps: first, a model for the systems dynamics is built, and then, on its basis, an optimal attractiveness index functions is calculated.

The software package WEKA [5] has number of ML tools for data analysis – Decision Trees, Naïve Bias, Decision Table, Sequential Model Optimization, NN, Linear Regression and Voting Features. The learning methods are called *classifiers*. WEKA also implements cost-sensitive classification. When a cost matrix is provided, the dataset will be reweighted. Because of its object oriented program code and good interface and several visual tools we prefer this program to the other three described above to conduct the experiments.

3.1. Practical Criteria for Evaluating ML

The human interpretability of rules and trees is a major benefit. We concentrated on decision tree learners and rule learners as they generate clear descriptions of how the ML method arrives at a particular classification. These models tend to be simple and understandable. In medical domains comprehensibility is particularly important.

Some of the criteria for evaluating machine learning methods and choosing an appropriate DM program are based on the goal to gain in usefulness of extracted information, to save time and computation costs, and to decrease the complexity. Except for WEKA, the other programs are tested only in Demo versions because of the limit to our academic purposes. In table 1 the presence and absence of the stated criteria is presented.

Table 1. Examination of demo version program for DM toward the stated criteria. This is done with regard to data analysis applications for medical data sets

Program/ Criterion	Useful- ness	Comp- lexity	Noisy data	Inconsist ency	Explana- tion
AutoClass	✗	✗	✓	✗	✗
WizWhy	✗	✓	✓	✓	✓
See5	✓	✗	✓	✓	✓
WEKA	✓	✗	✓	✗	✓

In *AutoClass* class probabilities are computed directly from the parameterized class distributions. The classes provide statistical models of instances' descriptions, where each class' s probability dominates all others. This version of *AutoClass* has no capability for searching the space of possible covariances. It has no visual tools and a poor explanation. For the purposes of our experiments it is not very useful.

Avoiding overfitting is one of the main features that differentiate between *WizWhy* and other prediction applications based on neural nets, decision trees or genetic algorithms. As a result, the accuracy level of *WizWhy* predictions is usually much higher in comparison with other approaches. The conclusive prediction' s probability is 88,3%In *WizWhy* predicted value is 42% and the issued rules are 126. The explanatory power is high, but the too many rules make the program not very useful. In the *BreastCancer* experiment it predicted class “*malignant*” with 93.3% probability, and the incorrectly classified cases are 6.

In *See5* a very simple *majority* classifier predicts that every new case belongs to the most common class in the training data. In this example, 365 of the 460 training cases belong to class “*benign*” so that a majority classifier would always optimize for “*benign*”. The decision tree has a lower error rate of 0.4% on the new cases, but this is higher than its error rate on the training cases. If boosting is used, this confidence is measured using an artificial weighting of the training cases and so does not reflect the accuracy of the rule.

The program package *WEKA* provides a number of tools for testing different classifiers. It has good explanatory and visual part. One great advantage is the object-oriented structure of the implemented learning models and algorithms. The program has sufficient interface and a variety of tools for transforming datasets. The unique feature of *WEKA* is the Distributed Experiments. The experimenter includes the ability to split and distribute an experiment to multiple hosts. We have

chosen to test the different classifiers in WEKA with *breast.cancer* dataset and the best results are achieved from the *NaiveBayes + Kernel* estimation algorithm. The incorrectly classified instances are 3. The same results are achieved with *SMO* classifier.

4 CASE STUDIES

For the experiment three data sets (from UCI ML repository <http://www.ics.uci.edu>) are used. The features of data sets are given in table 2.

Table 2. Features of the examined medical data sets, taken from UCI – ML Repository

Dataset	attributes	instances	Continuus	Classes
1. Breast Cancer	10	698	2	2
2. Diabetes Pima	9	768	8	2
3. IRIS	5	150	4	3

For the first dataset we use 460 randomly selected instances for training. The “*benign*” class in *breast.cancer* is 65,5%. When the class distribution is not well balanced the training set is locked so every one classification is done under the same condition.

For the second dataset the negative cases in the whole dataset are 65,1% and in the training set they are 63,70%, and 67,82% for testing. This gives us a well-spread distribution, by analogy with the first dataset. The difference in the third dataset is due to the well separated three classes. When classified with WEKA 100% correctly classified instances are achieved.

When tested with *DiabetsPima* dataset WEKA gives 76,71% accuracy with the *DecisionTable* classifier and 75,95% with *NaiveBayes*. The best result is with the *SMO* classifier – 76,34% accuracy. The only drawback is its increased time consumption. The *breast.cancer* data set (Wisconsin) has nonlinearly separated classes “*benign*” and “*malignant*” and is chosen for the testing dataset for a number of different classifiers available in WEKA. A detailed study of comparative experiments is performed and the results are shown in table 3.

Table 3. Results from the examined medical data set Breast.Cancer. *NaiveBayes´* indicates kernel estimation, Voted perceptron classifier obtains 127 perceptron

WEKA ML Classifier	Mean abs. error	Mean sqr error	Correctly Classified	Time
ZeroR	0,458	0,4726	66,39%	0,71 s
DecisionStump	0,123	0,2334	94,11%	0,49 s
Decision Table	0,0767	0,2343	94,95 %	3,84 s
IB1	0,0588	0,2425	94,12 %	0,94 s
j48.J48	0,0527	0,1695	96,64 %	3,35 s
kStar.KStar	0,0643	0,2025	94,54%	0,55 s
Logistic regress.	0.0409	0.1193	97,89 %	1,04 s
Naive Bayes	0,0282	0,1652	97.058 %	1,49 s
Naive Bayes´	0.0145	0.1129	98,74%	1,75 s
SMOptimization	0.1512	0.1799	98,74 %	39,1 s
Voted Perceptron	0.1092	0.3305	89.076 %	1,15 s
Neural Network	0,0405	0,1348	97,89%	11 s
Voting Feature []	0.3967	0.4095	96,22 %	0,55 s
AdaBoostM1	0.1221	0.1897	97.058 %	6.48 s

In WEKA *ZeroR* classifier simply predicts the majority class in the training data. Although it makes little sense to use this scheme for prediction, it can be useful for determining a baseline performance as a benchmark for other learning schemes. *DecisionStump* model builds a simple one-level binary decision tree (with an extra branch for missing values). *DecisionTable* produces a decision table using the wrapper method to find a good subset of attributes for inclusion in the table. This is done using a best-first search. *NaiveBayes* is a probabilistic classifier. By default it uses the

normal distribution to model numeric attributes. When kernel density estimators is used, this improves the performance. The *j48.J48* algorithm uses confidence threshold = 0,25. The *SMO* (*sequential minimal optimization* algorithm) is one of the fastest methods for learning but it works only for 2 classes. *VFI* (Classification by voting feature intervals) model uses intervals, which are constructed around each class for each attribute. Higher weight is assigned to more confident intervals, where confidence is a function of entropy. When no *WeightedConfidence* is selected, the correct classified instances increase by 1. The experiments gave higher accuracy (**98,74**) with *NaiveBayes'* and *SMO*, while the performance of the other classifiers excepting *ZeroR* were comparable. The *LogisticRegression* and *NN* algorithm came close with an accuracy of 97,89%.

5 CONCLUSIONS AND DISCUSSION

The improvement of new technologies rises data collection and accumulation. Without appropriate processing and interpretation this information remains useless. There are 4 standard methods for Data Mining: association, classification, clustering techniques and prediction. For most medical applications the logical rules are not precise but vague and the uncertainty is present both in premise and in the decision. For this kind of application a good methodology is the rule representation from decision-tree method, which is easily understood by the user. Therefore, the integration of fuzzy set and DM methods gives a much better and more exact representation of relationship between symptoms and diagnosis.

For the tested algorithms and classifiers in WEKA *Bayes* classifier and *SMO* model show the highest accuracy and the best correctly classified cases. In our experiments WEKA used only the most important attributes, and discarded the rest. In most cases the rules from *WizWhy* gave an overfit problem, there were too many specialized rules. This problem could be corrected by applying pruning. For *AutoClass*, the number of attributes has little influence on the number of classes generated, and its results require much interpretation by humans.

The experiments show that induced decision trees are useful for the analysis of the importance of clinical parameters and their combinations for the prediction of the diagnosis. We are also planning to provide new training examples for another medical datasets in order to derive simple rules for diagnosis determination, in time limiting conditions on a distributed information system. For this purpose the *Experimenter* in *Weka-3-1-9* will be used. The *Experimenter* includes the ability to split an experiment up and distribute it to multiple hosts. This works best when all results are being sent to a central data base and is appropriate to implement on the Web.

Acknowledgement.

This work has been partially supported by Grant No 809/98 of the Bulgarian Ministry of Education and Science and the Joint Research Project of ICSR-BAS and CVC-UAB "Intelligent user-computer interface for a cardiological diagnosis (CAD) system", 2000-2003.

REFERENCES

1. Data Mining Software, www.chel.com.ru/~rav/data_mining_software.html
2. Meidan, A., *WizWhy*[®] 3: A Data Mining tool for issuing predictions, summarising data, and revealing interesting phenomena, *White Paper*, web: www.wisoft.com
3. See5 RuleQuest Inc., www.rulequest.com
4. Stutz J., P. Cheeseman. Bayesian classification (autoclass): Theory and results. *In Advances in Knowledge Discovery and Data Mining*. AAAI/MIT Press, 1996
5. Witten Ian H., E. Frank, *Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations*, Ch. 8, © 2000 Morgan Kaufmann Publishers