6th Seminar on Neural Network Applications in Electrical Engineering, NEUREL-2002
Faculty of Electrical Engineering, University of Belgrade, Yugoslavia, September 26-28, 2002

Foundations of Predictive Data Mining

N. Jovanovic, V. Milutinovic, and Z. Obradovic, *Member, IEEE*

Abstract—**The aim of this paper is to introduce a novel reader to the topic of predictive data mining (DM) by discussing technical aspects and requirements of common mining tools. .A description of DM scope is followed by comparing DM to related data management and analysis techniques. This is followed by a discussion of a typical predictive DM process, and some of the more successful algorithms and software packages.**

Index Terms—**predictive data mining, knowledge discovery, data warehousing**

I. Introduction

We would like to encourage those who have tried to find "*data mining*" expression in the dictionary and failed. Everything is OK, the term is spelled correct, and the only problem why it is not easy to find it in dictionaries is that outside of statistics community the term is fairly new and is nowadays used in a different content from its prior occasional use in statistics.

Data mining (DM) is a process of efficiently extracting previously unknown interesting knowledge from large databases. It's a powerful technology with great potential to help users focus on the most important information stored in data warehouses or streamed through communication lines. DM can potentially answer questions that were too time consuming to resolve in the past. Also DM can predict future trends and behaviors, allowing us to make proactive, knowledge-driven decisions.

II. Data Mining

The amount of information in the world is estimated to double every 20 months. It is spread among data warehouses all around the world in different formats and on different security levels. Typical data warehouses today range in size up to terabytes, while applications with petabyte level data are emerging. Somewhere within these masses of data lies hidden information of strategic importance. The main question is how to find this needle in the haystack.

A promising approach to this problem is DM. It is a result of a long process of research and product development but it is not a magic wand. It won't supervise your data warehouse and notify you when it discovers an interesting pattern. On the other side it doesn't eliminate the need to know your problem as well as understand your data or analytical methods. It's just an assistant that can help you classify your data, predict later behavior, extract association rules or detect sequences. But with the efficient use of DM in your projects you could also reduce costs and increase revenues.

III. DM vs. Knowledge Discovery in Databases

The term "*data mining*" is used somewhat indiscriminately and is often applied to describe all of the tools employed to analyze and understand data. More specifically, however, DM is a part of a larger process called Knowledge Discovery in Databases (KDD). KDD incorporate several techniques: searching, statistical analysis, On-Line Analytical Processing (OLAP) and data mining.

IV. DM vs. OLAP

Data mining is also frequently identified with On-Line Analytical Processing (OLAP). As we shall see, they are very different techniques that can complement each other.

OLAP is part of the spectrum of decision support tools. Traditional query and report tools describe WHAT is in a database. OLAP goes further; it's used to answer WHY certain things are true. The user forms a hypothesis about a relationship and verifies it with a series of queries against the data. For example, an analyst might want to determine the factors that lead to loan defaults. He or she might initially hypothesize that people with low incomes are bad credit risks and analyze the database with OLAP to verify (or disprove) this assumption. In other words, the OLAP analyst generates a series of hypothetical patterns and relationships and uses queries against the database to verify them or disprove them.

The question is: How to form this hypothesis? It is a deductive process that is based on essential understanding of data and relations among them. What happens when the number of variables being analyzed is in the dozens or even hundreds? It becomes much more difficult and time-consuming to form a good hypothesis.

Data mining is different from OLAP because rather than verify hypothetical patterns, it uses the data itself to uncover such patterns. It is essentially an inductive process. For example, suppose the analyst who wanted to identify the risk factors for loan default were to use a data mining tool. The data mining tool might discover that people with high debt and low incomes were bad credit risks (as above), but it might go further and also discover a pattern the analyst did not think to try, such as that age is also a determinant of risk.

Here is where data mining and OLAP can complement each other. OLAP can be used to verify the results of data mining process. To achieve performance and useful output this cycle should to be supervised by analyst. This is important because the better you understand your data, the more effective the knowledge process will be.

V. DM Process

A systematic approach is essential to successful data mining. Many process models were designed to guide the analyst through a sequence of steps that will lead to good results. For example SPSS Clementine[21] uses the 5A's (Assess, Access, Analyze, Act and Automate) and SAS Enterprise Miner uses SEMMA (Sample, Explore, Modify, Model and Assess). A consortium of vendors and users of DM tools has been developing a specification called CRISP-DM (Cross-Industry Standard Process for Data Mining). We will give you a brief outline of these phases:

Project Understanding - this initial phase focuses on understanding the project objectives and requirements from a projects perspective, and then converting this knowledge into a data mining problem definition.

Data Understanding – starts with an initial data collection and proceeds with activities in order to get familiar with the data, to identify data quality problems, to discover first insights into the data, or to detect interesting subsets to form hypotheses for hidden information.

Data Preparation - covers all activities to construct the final dataset (data that will be fed into the modeling tool(s)) from the initial raw data.

Modeling – various modeling techniques are selected and applied, and their parameters are calibrated to optimal values. Typically, there are several techniques for the same data mining problem type so various data preparations have to be done.

Evaluation – before proceeding to final deployment of the model, it is important to more thoroughly evaluate the model, and review the steps executed to construct the model, to be certain it properly achieves the projects objectives. A key objective is to determine if there is some important projects issue that has not been sufficiently considered.

Deployment – gaining the model is generally not the end of the project. Even if the purpose of the model is to increase knowledge of the data, it will need to be organized and presented in a way that customer can use. Depending on the requirements, the deployment phase can range from generating a report to implementing a repeatable data mining process.

Keep in mind that while the phases appear in a list, the data mining process is not linear. You will inevitably need to loop back to previous steps.

VI. DM Technology

A designer of data mining tool should care at least about three technology aspects: appropriate statistical, modeling & learning algorithms; data collection & management techniques; and computing power.

Due to unstoppable progress in electronic industry processing speed is doubling approximately every 24-month. Also powerful workstations and parallel servers are more common and cheaper. It allows more data to be analyzed in shorter time. We can faster obtain more trustful conclusions and be competitive on the market.

For more information on data collection and management techniques, please check out our references about warehousing [25], [26].

VII. Predictive DM

The goal of data mining is to produce new knowledge that the user can act upon. DM does this by building a model of the real world based on data collected from a variety of sources which may include corporate transactions, customer histories and demographic information, process control data, and relevant external databases such as credit bureau information or weather data. The result of the model building is a description of patterns and relationships in the data that can be confidently used for prediction.

Before we select appropriate models and algorithms we have to focus on our goal: what is the ultimate purpose of mining this data? The next step is deciding on the type of prediction that's most appropriate: classification - predicting into what category or class a case falls, or regression: predicting what number value a variable will have (if it's a variable that varies with time, it's called time series prediction). Then you can choose the model type: a neural net to perform the regression, perhaps, and a decision tree for the classification.

In predictive models, the values or classes we are predicting are called the response, dependent or target variables. The values used to make the prediction are called independent variables. Predictive models are typically built, or trained, using data for which the value of the response variable is already known. This kind of training is sometimes referred to as supervised learning, because calculated or estimated values are compared with the known results. Data mining alternatives beyond the scope of this article include unsupervised learning, association identification and outlier detection.

Classification problems aim to identify the characteristics that indicate the group to which each case belongs. This pattern can be used both to understand the existing data and to predict how new instances will behave. Data mining creates classification models by examining already classified data (cases) and inductively finding a predictive pattern.

Regression uses existing values to forecast what other values will be. In the simplest case, regression uses standard statistical techniques such as linear regression. Unfortunately, many real-world problems are not simply linear projections of previous values. For instance, sales volumes, stock prices, and product failure rates are all very difficult to predict because they may depend on complex interactions of multiple predictor variables. Therefore, more complex techniques (e.g., logistic regression, decision trees, or neural nets) may be necessary to forecast future values.

Time series forecasting predicts unknown future values based on a time-varying series of predictors. Like regression, it uses known results to guide its predictions. Models must take into account the distinctive properties of time; especially the hierarchy of periods, seasonality, and calendar effects such as holidays, date arithmetic, and special considerations such as how much of the past is relevant.

VIII. Models and Algorithms

Each conference about data mining and/or knowledge discovery brings us dozens of papers that describes new ways how to mine data. They are response to rapid growth of subtle and sophisticated requirements on new data mining products.

Most of the models and algorithms discussed in this section can be thought of as generalization of the linear regression model. Much effort has been expended in the statistics, computer science and artificial intelligence to overcome the limitations of this basic model. The common characteristic of many of the newer technologies we will consider is that the pattern-finding mechanism is data-driven rather than user-driven.

Perhaps the most important thing to remember is that no one model or algorithm can or should be used exclusively. For any given problem, the nature of data itself will affect the choice of models and algorithms you choose. Consequently, you will need a variety of tools and technologies in order to find the best possible model.

IX. Neural Networks

Neural Networks are of particular interest because they offer a means of efficiently modeling large and complex problems in which there may be hundreds of predictor variables that have many interactions (actual biological neural networks are incomparably more complex). Neural networks may be used in classification problems (where the output is a categorical variable) or for regressions (where the output variable is continuous).

The most popular choice is a so called feed-forward neural network structure Fig. 1 that starts with an input layer, where each node corresponds to a predictor variable. These input nodes are connected to a number of nodes in a hidden layer. Each input node is connected to every node in the hidden layer. The nodes in the hidden layer may be connected to nodes in another hidden layer, or to an output layer. The output layer consists of one or more response variables.

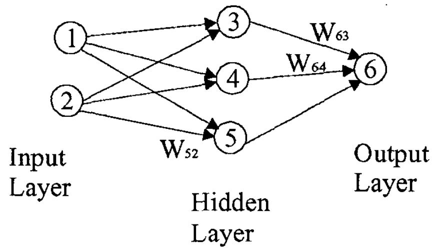

Fig. 1. A feedforward neural network with a single hidden layer

After the input layer, each node takes in a set of inputs, multiplies them by a connection weight Wij, adds them together, applies a function (called the activation or squashing function) to them, and passes the output to the node(s) in the next layer.)

$$out_i = function(\sum_{j=1}^{n}(W_{ij} * out_j)) \qquad (1)$$

In fact, if there is a linear activation function but no hidden layer, neural nets are equivalent to a linear regression; and with certain non-linear activation functions, neural nets are equivalent to logistic regression.

The connection weights (Wij) are the unknown parameters that are estimated by a training method. Originally, the most common training method was backpropagation. Newer methods include conjugate gradient, quasi-Newton, Levenberg-Marquardt, and genetic algorithms.

Neural networks with enough hidden nodes are universal approximators that will fit the training set if left to run long enough [6]. To avoid an overfitted neural network that will only work well on the training data, one must know when to stop training.

Some implementations address this problem by evaluating the neural network accuracy against the validation data periodically during training. Validation data set is obtained from initial data set and is used only in validation process. As long as the error rate on the validation set is decreasing, training continues. If the error rate on the validation data goes up, even though the error rate on the training data is still decreasing, then the neurall net may be overfitting the data. The graph in Fig. 2 illustrates how the validation data set helps to avoid overfitting.

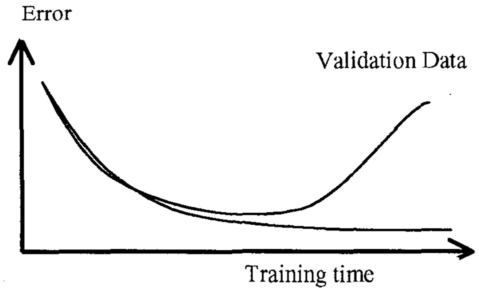

Fig. 2. Error on training and validation data as a function of training time

Since the goal of data mining is to make predictions on data other than the training set, you are clearly better off using a neural net that minimizes the error on the test data, not the training data.

Neural networks differ in philosophy from many statistical methods in several ways. First, parameters of neural networks are hidden inside the structure and can not be easily extracted and converted into explicit rationale. They require an extensive amount of training time. Once trained, however, they can provide predictions very quickly. They require very careful data cleansing, selection, preparation and pre-processing. For instance, neural nets require that all variables be numeric. Finally, neural networks tend to work best when the data set is sufficiently large and the signal-to-noise ratio is reasonably high.

X. Decision Trees

Decision trees are a way of representing a series of rules that lead to a class or value. Depending on the algorithm, each node of the tree may have two or more branches. Each branch will lead either to another decision node or to the bottom of the tree, called a leaf node. By navigating the decision tree you can assign a value or class to a case by deciding which branch to take, starting at the root node and moving to each subsequent node until a leaf node is reached. Each node uses the data from the case to choose the appropriate branch.

Decision trees models are commonly used in data mining to examine the data and induce the tree and its rules that will be used to make predictions. A number of different algorithms may be used for building decision trees including CHAID [24] (Chi-squared Automatic Interaction Detection), CART (Classification and Regression Trees), Quest [23], and C5.0 [27].

Decision trees are grown through an iterative splitting of data into discrete groups (CART are characterized by binary splits while CHAID allows n-way splits based on chi square test), where the goal is to maximize the "distance" between groups at each split. One of the distinctions between decision tree methods is how they measure this distance but the details of such measurement are beyond the scope of this article.

Decision trees that are used to predict categorical variables are called classification trees because they place instances in categories or classes. Decision trees used to predict continuous variables are called regression trees.

Trees can grow in size and became unintelligible, but more importantly they can overfit the data. Tree size can be controlled via stopping rules that limit growth [17], [10]. One common stopping rule is simply to limit the maximum depth to which a tree may grow. An alternative to stopping rules is to prune the tree. The tree is allowed to grow to its full size and then, using built-in heuristics or user intervention, the tree is pruned back to the smallest size that does not compromise accuracy. CART algorithm prunes trees by cross validating them to see if the improvement in accuracy justifies the extra nodes. Although a very popular choice in predictive DM, it is important to keep in mind that decision trees were observed to lead to very complex models when applied to large training data even when underlying relationships among attributes were simple. Practical approached to solving this problem were proposed at [12], [11].

Decision trees that are not limited to univariate splits could use multiple predictor variables in a single splitting rule. Such a decision tree could allow linear combinations of variables, also known as oblique trees.

Finally decision trees handle non-numeric data very well. This ability to accept categorical data minimizes the amount of data transformations and the explosion of predictor variables inherent in neural nets.

XI. Rule Induction

Rule induction is a method for deriving a set of rules to classify cases. Although decision trees can produce a set of rules, rule induction methods generate a set of independent rules that do not necessarily form a tree. Because the rule inducer is not forcing splits at each level, and can look ahead, it may be able to find different and sometimes better patterns for classification.

Unlike trees, the rules generated may not cover all possible situations. Also unlike trees, rules may sometimes conflict in their predictions, in which case it is necessary to choose which rule to follow. One common method to resolve conflicts is to assign a confidence to rules and use the one in which you are most confident. Alternatively, if more than two rules conflict, you may let them vote; perhaps weighting their votes by the confidence you have in each rule [1].

XII. K-Nearest Neighbor

When trying to solve new problems, people often look at solutions to similar problems that they have previously solved. K-nearest neighbor (k-NN) is a classification technique that uses advantages of this approach. It decides in which class to place a new case by examining some number — the "k" in k-nearest neighbor — of the most similar cases or. It assigns the new case to the same class to which most of its neighbors belong.

The first thing you must do to apply k-NN is to find a measure of the distance between attributes in the data and then calculate it. While this is easy for numeric data, categorical variables need special handling. For example, what is the distance between blue and green? You must then have a way of summing the distance measures for the attributes. You have to select the set of already classified cases to use as the basis for classifying new cases, predetermine size of a neighborhood for comparisons, and also decide how to count the neighbors themselves (e.g., you might give more weight to nearer neighbors than farther neighbors).

K-NN requires large computational efforts because the calculation time increases as the factorial of the total number of points. While it's a rapid process to apply a decision tree or neural net to a new case, k-NN requires that a new calculation be made for each new case.

K-NN models are very easy to understand when there are few predictor variables. They are also useful for building models that involve non-standard data types, such as text. The only requirement for being able to include a data type is the existence of an appropriate metric.

XIII. Genetic Algorithms

Genetic algorithms are not used to find patterns per se, but rather to guide the learning process of data mining algorithms such as neural nets. Essentially, genetic algorithms act as a method for performing a guided search for good models in the solution space [16].

They are called genetic algorithms because they loosely follow the pattern of biological evolution in which the members of one generation (of models) compete to pass on their characteristics to the next generation (of models), until the best (model) is found. The information to be passed on is contained in "chromosomes," which contain the parameters for building the model.

For example, in building neural net, genetic algorithms can replace backpropagation as a way to adjust the weights. The chromosome in this case would contain the weights. Alternatively, genetic algorithms might be used to find the best architecture, and the chromosomes would contain the number of hidden layers and the number of nodes in each layer.

While genetic algorithms are an interesting approach to optimizing models, they add a lot of computational overhead.

XIV. Logistic regression

Logistic regression is a generalization of linear regression. It is used primarily for predicting binary variables and occasionally multi-class variables. Due to a discrete response variable, model can not be created directly by linear regression. Therefore, instead of predicting whether the event itself will occur, we build the model to predict the logarithm of the odds of its occurrence. This logarithm is called the log odds or the logit transformation.

$$odds\_ratio = \frac{probability\_of\_an\_event\_occuring}{probability\_of\_an\_event\_not\_occuring} \quad (2)$$

The odds ratio has the same interpretation as in the more casual use of odds in games of chance or sporting events. When we say that the odds are 3 to 1 that a particular team will win a soccer game, we mean that the probability of their winning is three times as great as the probability of their losing. So we believe they have a 75% chance of winning and a 25% chance of losing.

Similar terminology can be applied to the chances of a particular type of customer (e.g., a customer with a given gender, income, marital status, etc.) replying to a mailing. Having predicted the log odds, you then take the anti-log of this number to find the odds. Odds of 62% would mean that the case is assigned to the class designated "1" or "yes," for example.

While logistic regression is a very powerful modeling tool, it assumes that the response variable is linear in the coefficients of the predictor variables. Furthermore, the modeler, based on his or her experience with the data and data analysis, must choose the right inputs and specify their functional relationship to the response variable. Doing this effectively requires a great deal of skill and experience on the part of the analyst.

XV. Choosing Software

In evaluating data mining tools you must look at a whole constellation of features, described below. You cannot put data mining tools into simple categories such as "high-end" versus "low-end" because the products are too rich in functionality to divide along just one dimension.

There are three main types of data mining products. First are tools that are analysis aids for OLAP. The next category includes the "pure" data mining products. These are horizontal tools aimed at data mining analysts concerned with solving a broad range of problems. The last category is analytic applications that implement specific business processes for which data mining is an integral part. For example, while you can use a horizontal data mining tool as part of the solution of many customer relationship management problems, you can also buy customized packages with the data mining imbedded. However, even packaged solutions require building and tuning models that match user's data.

Depending on particular circumstances — system architecture, staff resources, database size, problem complexity — some data mining products are likely, to be better suited than others to meet specific needs. Evaluating a data mining product involves learning about its capabilities in a number of key areas.

XVI. Conclusion

Predictive data mining offers great promise in helping us to uncover patterns hidden in the data that can be used to predict the behavior of customers, products and processes. However, predictive data mining tools need to be guided by users who understand the business, the data, and the general nature of the analytical methods involved.

It's vital to properly collect and prepare the data, and to check the models against the real world. Choosing the right data mining products means finding a tool with good basic capabilities, an interface that matches the skill level of the people who'll be using it, and features relevant to your specific problems.

References

[1] R. Agrawal, J. Schafer, "Pareallel Mining of Association Rules", IEEE Trans. On Knowledge and Data Eng., December 1996.

[2] C. Bishop, "Neural networks for pattern recognition", Ox-ford, 1994.

[3] L. Breiman, "Classification and regression trees", Chapman&Hall, 1984.

[4] I. Bruha, "Data Mining, KDD, and Knowledge Integration: Methodology and A case Study", SSGRR 2000.

[5] U. Fayyad, P. Shapiro, P. Smyth, R. Uthurusamy, "Advances in Knowledge Discovery and Data Mining", MIT Press, 1996.

[6] C. Glymour, D. Madigan, D. Pregibon, P. Smyht, "Statistical Themes and Lessons for Data Mining", Data Mining and Knowledge Discovery 1, 11-28, 1997.

[7] R. Hecht-Nilsen, "Neurocomputing", Addison-Wesley, 1990.

[8] N. Jovanovic, V. Milutinovic, F. Patricelli, "E-Business and E-Challenges: Datamining", IOS Press, 2002.

[9] A. Lazarevic, X. Xu, T. Fiez, Z. Obradovic, "Clustering-regression-ordering steps for knowledge discovery in spatial databases", In Proc. IEEE/INNS Int. Joint Conf. on Neural Networks, ISBN 0-7803-5532-6, Washington D.C., No. 346, 1999.

[10] N. Liu, H. Motoda, "Feature selection for knowledge discovery and data mining", Kluwer Academic Publishing, 1998.

[11] M. Mentha, R. Agraval, J. Rissanen, "SLIQ: A Fast Scalable Classifier for Data Mining", 5th Int. Conf. On Extending Database Technology (EDBT), Avignon, France 1996.

[12] T. Oates, D. Jensen, "Large Datasets Lead to Overly Complex Models: an Explanation and a Solution", In Proceedings of the Fourth International Conference on Knowledge Discovery and Data Mining, 294 - 298, 1998.

[13] T. Oates, and D. Jensen, "The Effects of Training Set Size on Decision Tree Complexity", In Proceedings of The Fourteenth International Conference on Machine Learning, 254 - 262, 1997.

[14] D. Pokrajac, T. Fiez, Z. Obradovic, "A Tool for Controlled Knowledge Discovery in Spatial Domains", In Proc. of 14.ESM Conference, 2000.

[15] D. Pyle, "Data Preparation for Data Mining", Morgan Kaufman, 1999.

[16] P. Werbos, "Beyond Regression: New tools for predicting and analysis in the behavioral sciences", Harvard University, Ph.D. Thesis, 1974.

[17] J. Yang, V. Honavar, "Feature subset selection using a genetic algorithm", Kluwer Academic 1998.

[18] T. Zhang, R. Ramakrishan, M. Livny, "BIRCH: An Efficient Data Clustering Method for Very Large Databases", Prof. of ACM SIGMOD Int. Conf. on Data Management, Canada 1996.

[19] K. Thearling, "An Introduction to Data Mining and Advanced DSS Technology", Available: http://www.thearling.com

[20] "CRISP-DM Process Model," Available: http://www.crisp-dm.org

[21] "SAS Enterprise Miner," Available: http://www.sas.com/products/miner

[22] "SPSS Clementine," Available: http://www.spss.com/clementine

[23] "Introduction to Data Mining and Knowledge Discovery," Available: http://www.twocrows.com

[24] "QUEST: Quick, Unbiased and Efficient Statistical Tree," Available: http://www.stat.wisc.edu/~loh/quest.html

[25] "CHAID: Chi-square Automatic Interaction Detector," Available: http://www.themeasurementgroup.com/Defintions/chaid.htm

[26] "The Data Warehousing Information Center," Available: http://www.dwinfocenter.org

[27] "The Data Warehousing Institute," Available: http://www.dw-institute.com

[28] "Data Mining Tool See5 and C5.0," Available: http://www.rulequest.com/see5-info.html