Construction of Single Classifier from Multiple Interim Classification Trees

Abdul Aziz[1] and Natalia Ahmed[2],
[1]The Institute of Management Sciences (Pak-Aims), Pakistan

[2]University of Hail, Kingdom of Saudi Arabia

Summary
Predicting the future is always a quest of mankind and thus Supervised Learning (predictive modeling, machine learning) is one of the most rapidly used techniques of Data Mining. Finding out patterns and measuring the accuracies of the foretell are very hot research areas these days. In this research paper, we have introduced a new method to generate a final optimal and accurate classifier from several interim classification trees for various samples of same dataset. This method saves plenty of time of passing test data from several trees because instead we will have an ultimate classifier from the merger of the interim trees, with the help of information gain theory. In this paper, the method is applied on the Drug Data used in SPSS Clementine demonstration. Above all, the proposed method is quite simple and easy to understand as well as easy to implement in practical environment.

Key words:
Data mining, Supervised learning, Machine learning, Classification trees.

1. Introduction

Discovering interesting patterns and establishing forecast based upon them has always been of common interest. Animal migration patterns are primary focus of hunters, farmers want more crops, so they find interesting factors in growth, and politicians are interested in patterns in voter opinion. Data mining/KDD expert make sense of data, ascertains the patterns that rule and summarizes them in suppositions that can be used for forecasting what will happen in new state of affairs. In data mining the whole processing till patterns finding is done with help of computers [13]. Such as a manufacturer is always curious about does he really knows his customers?, Who are his most profitable customers?, How can he attract more like them?, What do his customers really think about his products and services?, Who negatively affect his bottom line?, Who is leaving him for his competitor? Etc. and know what his customers are going to do, before they actually do. Such queries and desire to look deeper in data to increase profit etc. has been possible through predictive modeling. Knowingly and unknowingly people have been doing this for centuries but a formal field of knowledge discovery is the gift of the modern science. Associations and relationships have always been the in focus but supervised learning which has its roots in machine learning has not only helped in finding out patterns but also such predictions have helped a lot in crime analysis, medical diagnostic systems, risk of loan, stock market and several more because it automatically finds out relations among data features available and draws attention to interesting structures and relations [17].

Our focus at the moment is on Predictive modeling or supervised learning but let us first have a look on the scheme of this paper. Section 2 focuses on the predictive modeling, its meaning and various methods, algorithms and approaches used for supervised learning. In section 3 we propose a new most advantageous way to learn from the data set. This section will discuss about how some trees (which are created using the samples of the main data set) using Information gain ratio is amalgamated to put up one absolute predictive model. In recent years information gain is also being used for mutual information measure for info gain or info loss in two sets which are disjoint [21]. We suspend the details till section 3 and let us see what it is all on the subject of supervised learning.

2. RELATED WORK

There are many techniques of doing data mining [12] and here we are going to discuss them briefly, so first comes supervised learning when we have the perfect knowledge of the probable outcomes of the given problem. Next is clustering also named segmentation, in which most close or similar items or data is grouped in subsets. Through clustering we identify different groups which have no similarities but the items in each group are closely alike. After clustering, classification is made based upon resulting groups. K-means and Kohonen feature maps are popular clustering algorithms [26]. Another technique is Dependency Modeling, which calculates the probability density of the processes and for this purpose density estimation methods are used. There is one more approach

Manuscript received May 5, 2011
Manuscript revised May 20, 2011

for identifying interesting structures and it is Data Summarization and it finds out similarity between a few attributes in a subset of data, for this purpose vertical or horizontal slices are made and explored. Change and deviation detection method take care of sequence information of data, most of the time it is not explicitly done by other methods. On the whole all the techniques to do data mining are usually divided into supervised and unsupervised learning.

In this section we will see how predictive modeling is defined by various experts and how it is being done in different situations.

2.1 Supervised Learning

Supervised learning is also termed as predictive modeling and it is based upon machine learning theory. Machine learning technique addresses realistic problems effectively such as when there is no mathematical model of the problem or it is very costly to formulate one [17]. Supervised learning has its input and outputs and its objective is to predict the value of some particular attribute or feature (column value) of given data using other attributes under observation.

In such erudition we have the complete knowledge of the possible outcomes; e.g the famous benchmark "The Weather Problem" (shown in figure 1) has two possible values of an observation i-e 'play' or 'do not play'. These are formally called class values and there can be three, four or even more choices of class attribute whereas Class attribute is the feature that we want to predict for a specific observation values. These values of class attributes act like instructor and there are high chances to maintain a standard and to highlight any suspicious observation [16].

There can be two output spaces in predictive modeling; one is regression modeling and other is classification modeling. Regression modeling is used where the output is numeric or continuous whereas classification modeling is used where prediction will consists of discrete values [12]. Once we have the data set containing observations with their class attributes then we apply some algorithm and structure a decision tree or set of rules.

Outlook	Temp	Humidity	Windy	Class
Overcast	72	90	True	Play
Overcast	83	78	False	Play
Overcast	64	65	True	Play
Overcast	81	75	False	Play
Rainy	71	80	True	Don't play
Rainy	65	70	True	Don't play
Rainy	75	80	False	Play
Rainy	68	80	False	Play
Rainy	70	96	False	Play
Sunny	75	70	True	Play
Sunny	80	90	True	Don't play
Sunny	85	85	False	Don't play
Sunny	72	95	False	Don't play
Sunny	69	70	False	Play

Fig. 1. The Weather Data

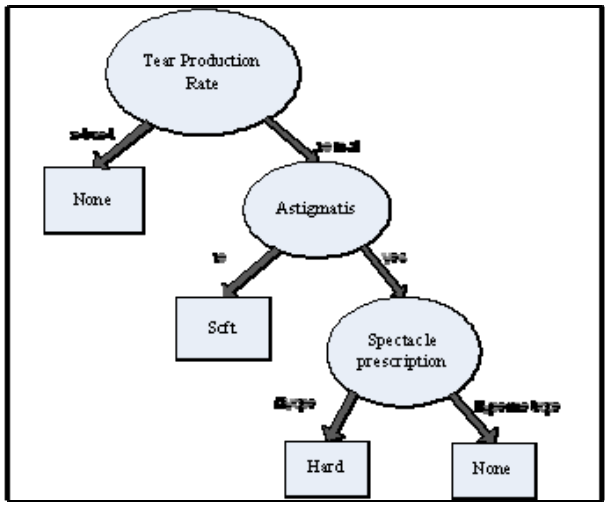


Fig. 2. Classification tree for contact lens data

A classifier tree can look like as shown in figure 2 indicating the rules for class attribute values of famous contact lens data [13]. Predictive modeling can be embodied in form of classification/ regression rules but usually classification rules are presented via decision tree which help in doing analysis for prospective supposition channels in the data that needs to be trimmed down [10]. There are several ways to perform regression and classification but one constraint that is usually associated with predictive modeling; supervised learning that is classification cannot be done without labeled data but now researches are also trying to use unlabeled data with labeled data to find out interesting classes, reason being we can get plenty of raw data to ponder upon but it needs more work to properly label it to hunt classes, as it is executed for text classification in [2] by taking positive and unlabeled data.

Industrial researchers are utilizing predictive modeling with the help of neural network (using regression methods) for Efficient Architectural Design Space Exploration, causing minimum error rate (1 to 2 %) predictions [3]. As conversed in [11], researchers have used supervised learning to improve the course of actions for collecting owing accounts receivables. Predictive modeling (classification) is used to predict the magnitude of holdup incase an invoice is unpaid. And the procedure of prediction proves that it highly increase the success factor. Supervised learning model of invoice consequence is predicted as follows; data is taken from previous invoices to form a predictive model to forecast that when a new invoice will be paid, in case no actions are taken. For

this purpose five classes are created: on time, 1-30 days late, 31-60 days late, 61-90 days late, and more than 90 days late (or 90+ days late). These five values are usually in all businesses related to payments etc [11]. Additionally, multi-level classification is another form of predictive modeling and parameter fitting optimization in multi-level classification is refined in [9], you can refer to this for further details.

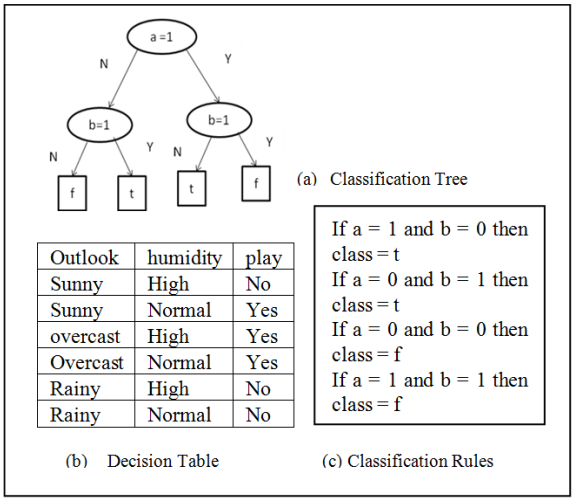
2.2 Techniques for Supervised Learning

Let us have a rapid view of various ways of doing classification (when we have the class attribute and its values). On the way to have a finest classifier, researchers have been formulating various techniques. There are algorithms that select among several attributes of data set to find the best attribute to split at different levels of trees, for example C4.5 by Quinlan [8] and CART by Breiman, univariate and multivariate respectively. C4.5 finds out the suitable attribute to split by calculating information gain from entropy theory and CART [14,15] uses Gini index to find the best to split. According to [4] there are several methods for supervised learning and among them comparison of ten is done, including SVMs, neural nets, logistic regression, naive bayes, memory-based learning, random forests, decision trees, bagged trees, boosted trees, and boosted stumps. These ten supervised learning methods are compared through various performance measures and a large evaluation is done. The results were very interesting; bagged trees, neural nets and random forests gave surprisingly good performance [4]. Let us discuss few of these methods, for example; Bagging is a method in which samples are taken from large data set and each sample has its own tree then test data is passed through all trees and majority result is accepted. Windowing is another method by Quinlan. Data is divided into test and training sets, training set is further divided in two subsets, classifier is generated from one and tested upon the other, all the misclassified records are made part of training set and same process is repeated, when maximum mis-classification is removed then model is applied to actual test data. Boosting (by Freund and Schapire in 1996 [27-29]) approach which gives high weights to misclassified records and generate and regenerate the predictive model and ultimately high accuracy is achieved [7].

Another classification technique is Random Forests (by Breiman (2001)), where several random trees are formed and there average is calculated from random classifiers. Randomness is applied to choose splitting variable and in deeper details a bit similar to boosting [18]. Random trees have established its strength in regression and classification for high dimensional data and provide with complete conditional spread of response values [18]. AdaBoost has many versions and extensively well-known algorithm [29]. AdaBoost and Random Forests are used by [5] for selecting variables and ranking them according to importance for Insolvency Risk and the model is based on empirical logistic regression. They have proven that predictive modeling has been fruitful for their goal. Some statisticians did their analysis and pointed out that step by step optimization in AdaBoost is similar to likelihood in logistic regression learning. Mease and Wyner in [22] highlighted some statistical weak spots in AdaBoost but Yoav Freund and Robert E. Schapire cleared them in [20] and said that AdaBoost has the ability of early stopping and it makes it more generalize for the given dataset . Bagging, Boosting and subspace methods are combined to get a good classification of data and ultimately decreasing biasness and dispersion of classifiers [19]. Some predictive learning methods come under supervised descriptive rule discovery including contrast set mining, emerging pattern mining and group discovery, a study upon this is performed in [6], discussing the importance of such classification framework in patient risk group detection in medicine , customer relation management and bioinformatics. Above is a mere discussion about the predictive modeling techniques, there are many more for classification learning. In next section we are going to discuss a little about the representation of the behavior of supervised learning.

2.3 Output of Supervised Learning

Human understandable visualization of all derived from supervised Learning is one of the very important steps. The patterns and behavior found through predictive learning are represented in various ways. Very common representations are decision tables, decision trees and decision rules/list. Decision tables are simple tables showing the rules for all class attribute values. Decision trees are a graphical representation of the rules deduced from the supervised modeling. Decision lists also state the rules of the given data set in form of text and are called classification rules [13]. The popular way to represent the resulting behavior of observations is through decision trees [10], which are commonly called supervised classifiers. Figure 3 shows various ways to represent the upshots of predictive modeling [13]. At advanced levels two and three dimensional maps are used to display the analysis. If the dataset has very high dimensionality then the classifiers can be represented in form of dimensional models. The figure 3 shows examples of various representations such as (a) classification tree, (b) decision table and (c) classification rules.



Outlook	humidity	play
Sunny	High	No
Sunny	Normal	Yes
overcast	High	Yes
Overcast	Normal	Yes
Rainy	High	No
Rainy	Normal	No

Fig. 3. Representing Classification leaning

3. OPTIMAL CLASSIFIER FROM MANY TREES

A good classifier depends upon mainly two factors one the chosen algorithm and second the training data set. So in this sense bagging is a right choice because it hits various samples to build classifiers and the concept is derived from "bootstrap aggregating"[1]. Bagging has shown impressive results for unstable classification models as well as stable classification models [23]. In this section we propose an optimal approach for merging several trees made from the different samples of same data set. First we will illustrate the logic of the proposition followed by a demonstration explaining each and every step. We have proposed a way to merge the various trees built after applying Bagging technique; all of such classifiers are steady with training observations but different [29]. It is sort of a hectic activity to pass each and every record from all the trees, so it is always optimal to have a final one classification model. Keeping in view this objective, we have formulated a procedure through which we can ultimately have a final decision tree or set of rules.

3.1 Attribute Mapping

The primary thing to do is to have all the trees generated by bagging method; additionally we will need the information about the attributes being used by 'n' number trees at all levels of tree i-e from root to leaf nodes. Such information can be gathered in a matrix.

TABLE 1 ATTRIBUTE MAPPING TABLE

	Tree 1	Tree 2	Tree 3	…Tree n
Attri at Root	A	E	C	B
Attri Left L1	C	A	D	D
Attri Right L1	E	C	B	A
……..	…..	……	…….	……
.....	…..	……	…….	…….
Attri Left Lk	B	D	E	C
Attri Right Lk	D	B	A	E

Table simply describes the trees from 1 to n on the column side and distribution of all the attributes used for branching from top to bottom (level-1after root and level-k, the last branching nodes) on the row side. Each tree is basically formed from an independent sample of the complete dataset. So attributes in all samples are same but the way they are used to do classification is somehow different in all the trees with respect to their samples.
Table 1 shows the mapping of attributes with their trees. In the above table we have basically accumulated all the information related to n trees as called 'interim trees' in our discussion. We need all this information because we need to merge n trees to get the final classifier for the test and future unseen data.

3.2 Building Final Model

This stage is focused to actually construct the ultimate tree. We have taken help of information theory in our model.
Attribute selection using information gain has been proved to be very effective in all fields and in medicine problems as well [24]. Figure 4 shows all the steps being performed in building single classifier. The formulas that we will use in section 3.3 are according to C4.5 for Information gain calculations and are shown below. The foundation of the outline of C4.5 is based upon HUNT's CLS algorithm to raise a decision tree [25].

$$\text{Entropy}(t) = -\sum_j p(j|t) \log_2 p(j|t)$$

Decrease in Entropy is called information gain.

$$\text{Gain}_{split} = \text{Entropy}(p) - [\sum_{i=1 to k} n_i/n \text{ Entropy}(i)]$$

The algorithm takes the information from the Attribute Mapping Table figure 1 as the foundation step and builds the tree from root to leaf nodes. This average is matched with the attribute's information gain and the closest one is chosen to be the branching node at the specific level. This process is repeatedly performed unless all the branching node and attributes are passed and utilized. There are few checks that algorithm will perform during this process, for example if two or three trees among n trees have same attribute at the specific level then the distinct will be taken to calculate the average. And if some attribute has already been selected at some above node in final model then it

will not be used while calculating information gain for new splitting node.

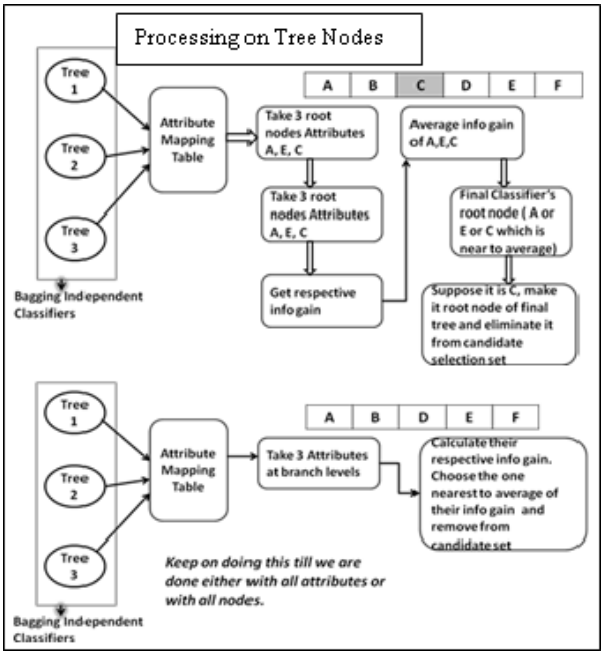


Fig. 4. Pictorial Representation of Classifier Builder

1. Create an empty Tree 'F'
2. Take a *newNode* 'z' in a tree 'F' *Node →{root, left-node 1…n, right-node 1…n}*
3. Extract all attributes in n trees matching the level of Node 'z'
4. Make a *Candidate Attributes Set* by finding distinct attributes from step 2.
5. Check if any of the attributes have been taken as split node/nodes already in previous iterations,
6. Exclude the attribute/attributes found at step 5 from *Candidate Attributes Set* and make a *Refined Candidate Attributes Set*.
7. Now calculate the information gain of all the attributes in *Refined Candidate Attributes Set* respectively and take average.
8. Choose the attribute with info gain closest to average as the splitting node 'z'
9. Record this attribute and its level in tree in 'Used Attributes Set'.
10. Repeat steps 2----9 for all nodes and all attributes till no more attributes or nodes are left to merge.
11. Tree 'F' is the final decision tree.

Fig. 5. Final Classifier Builder

1. All n trees are created from same data set, although samples are taken.
2. Data set is fairly large in terms of attributes (many attributes)
3. *Final Classifier Builder* follows univariate approach. Initially the final tree is assumed to have no nodes.
4. Numeric and nominal attributes are treated through same procedure. Discretisation will be done for numeric attributes.

Fig. 6. Assumptions set

Performing all this procedure in the end will provide us with a final tree or 'ultimate classifier' model for the training data sets of the n trees. The sub-steps of this whole process are described in form of an algorithm in figure 5.

Every learning algorithm has assumptions as said by 'Rob Schapire' in a tutorial on Boosting at Princeton University, so assumptions corresponding to these steps are mentioned in figure 6. The algorithm is stated at a very high abstraction level. So programming and implementation details are hidden, intention is to keep it simple and easy to understand. The algorithm described in figure 5 is explained in section 3.3. Let us have a detailed explanation in section 3.3 with the help of the example having some observations for class attribute upon which we will build our interim trees and then final classifier.

Every learning algorithm has assumptions as said by 'Rob Schapire' in a tutorial on Boosting at Princeton University, so assumptions corresponding to these steps are mentioned in figure 6. The algorithm is stated at a very high abstraction level. So programming and implementation details are hidden, intention is to keep it simple and easy to understand. The algorithm described in figure 5 is explained in section 3.3. Let us have a detailed explanation in section 3.3 with the help of the example having some observations for class attribute upon which we will build our interim trees and then final classifier.

3.3 Example

In this section we are going to take a real example to explain the whole proposition. For simplicity we have taken three different trees with five attributes. It is the same Drug data set used in the SPSS Clementine Demonstration. There are total 200 records that we have taken and two samples are drawn from these observations. Attributes are age, blood pressure, cholesterol, Sodium NA, potassium K and class attribute Drug. Class attribute can be drug A, B, C, X or Y. Two classifiers are already there after applying simple bagging technique. The first step is to make the Attribute Mapping Table similar to the one in table 1. Table 2 shows that how the data features are taken by different trees on different levels of trees.

TABLE 2 ATTRIBUTE MAPPING TABLE

	Tree 1	Tree 2
Attri at Root	Age	K
Attri Left L1	Na	Na
Attri Right L1	K	Age
Attri Left L2	Cholesterol	BP

Now the second step is to start creating a new /final tree. According to the algorithm shown in figure 5 we will start from root level. According to interim trees we have two attributes from which we have to select one attribute to be at the root node. We will calculate the info gain ratio of Age and K in their respective samples (as shown below).

(a) Info gain ration "K" in Tree2 = 0.530019419
(b) Info gain ration "Age" in Tree1 = 0.488954144

Now take the average of (a) and (b) = 0.5094867815 so attribute 'K' will be selected as the root node of final tree 'F'. Now move towards the next level. We are left with Age, Na, BP and Cholesterol. At left level-1

(c) Info gain ratio of Na in Tree 2 = 0.048891514
(d) Info gain ration of Na in Tree 1 = 0.564901758

As the attribute is same so we will not calculate any average and will just accept "Na" as the left level1 branching node of tree 'F'. After this we have Age, BP and Cholesterol. If we see at the right level1 we have 'K' and 'Age'. 'K' is already taken as the root node so at this level we have no choice other than selecting 'Age'. So 'Age' will go towards right level 1 as branching node. For left level 2, we now have Cholesterol and BP. Their respective info gain is as follows: -

(e) Info gain ratio of cholesterol in Tree 1 = 0.28989237
(f) Info gain ratio of BP in Tree 2 = 0.38249133

Now take the average of (e) and (f) = 0.33619185 so the choice is 'BP'. BP will be the splitting node at left level 2. As cholesterol is left so it will be the last level branching node of the final tree 'F'. The order of attributes in the final classifier is 'K', 'Na', 'Age', 'BP' and 'Cholesterol'. Info gain ratios are calculated according to C4.5 as follows: -

1. Info gain of whole data set is calculated (separate for each sample)
2. Then each attribute's info gain is calculated in its respective sample.
3. And finally the gain ratio is calculated for each attribute with respect to its data set sample.

1. $\text{Info}(t) = -\sum_{j=1 \text{ to } k} f(C_j, S)/|S| \ . \log_2 f(C_j, S)/|S|$
2. $\text{Info}_x(t) = -\sum_{i=1 \text{ to } n} [\,|T_i|/|T|\,] \ . \text{Info}(T_i)\ ]$
3. $\text{Gain}(t) = \text{Info}(t) - \text{Info}_x(t)$
4. $\text{Split Info}(t) = -\sum_{i=1 \text{ to } n} [\,|T_i|/|T|\,] \ . \log_2 [\,|T_i|/|T|\,]]$
5. $\text{Gain Ratio}(t) = \text{Gain}(t) \ / \text{SplitInfo}(t)$

f(Cj,S) number of samples in S that belong to class Cj (out of k possible classes) and |S| represents the total number of samples in the set S. x refers to the splitting

Fig.7. Information Gain Calculations

The formulas that are used to calculate the information gain ratio are stated in figure 7. We have seen that applying a simple yet an efficient algorithm we have reached a single tree and can easily check the results and accuracy of test data using this one classifier instead of numerous.

4. CONCLUSION

The eventual goal of learning is to have a final classifier model for the problem in focus. A single classification model is much more valuable both in terms of time and processing cost. If we have multiple models for one dataset, we will have to pass our test data from all these models and aggregate the independent results but a single classifier saves a lot of time and makes it easy to achieve the ultimate target.

Focusing the mentioned reason we have proposed a method to form one classifier from multiple classification trees from the same data set in order to decrease the complexity of passing through data from various classification trees to get the final result. Consequently, it will not only be efficient but it may produce more accurate and consistent results. It will be far easier to locate the misclassifications and cost sensitive errors. At the same time it is quick to add modifications to the classifier for modified classification rules. In nutshell our approach gives a boost to typical bagging method and makes it more promising and effective as well as efficient.

REFERENCES

[1] Kristina Machova, Frantisek Barcak, Peter Bednar Acta Polytechnica Hungarica,"A bagging method using decision trees in the role of base classifiers," Acta Polytechnica HungaricaVol. 3, No. 2, 2006
[2] Bing Liu Yang Dai Xiaoli Li, Wee Sun Lee Philip S. Yu "Building text classifiers using positive and unlabeled examples" Third IEEE International Conference on Data Mining (ICDM'03), 2003.
[3] Engin ¨Ipek, Sally A. Mckee, Karan Singh, Rich Caruana Bronis R. De Supinski and Martin Schulz, "Efficient architectural design space exploration via predictive modeling", ACM. Trans. Architect. Code Optim. 4, 4, Article 20 (January 2008).
[4] Rich Caruana Alexandru Niculescu-Mizil, "An empirical comparison of supervised learning algorithms" Appearing in Proceedings of the 23 rd International Conference on Machine Learning, Pittsburgh, PA, 2006.
[5] Rohan A. Baxter, Mark Gawler, Russell Ang, "Predictive model of insolvency risk for Australian corporations" appeared at the Sixth Australasian Data Mining Conference (AusDM 2007), Gold Coast, Australia. Conferences in Research and Practice in Information Technology (CRPIT), Vol. 70, 2007
[6] Petra Kralj Novak , Nada Lavrac Jozef Stefan, "Supervised Descriptive Rule Discovery: A unifying survey of contrast

set, emerging pattern and subgroup mining", Journal of Machine Learning Research 10 (2009) 377-403.
[7] S. Mendelson, A.J. Smola (Eds.): "An introduction to boosting and leveraging", Advanced Lectures on Machine Learning, LNAI 2600, 2003 pp. 118–183.
[8] J. Ross Quinlan C4.5: "Programs for machine learning". Morgan Kaufmann Publishers, Inc., 1993.
[9] Matthias W. Seeger Max Planck, Institute for Biological Cybernetics, "Cross-validation optimization for large scale structured classification kernel methods", Journal of Machine Learning Research 9 (2008) 1147-1178.
[10] Vassilios S. Verykios1, Elisa Bertino2, Igor Nai Fovino2 Loredana Parasiliti Provenza2, Yucel Saygin3, Yannis Theodoridis "State-of-the-art in privacy preserving data mining " SIGMOD Record, , March 2004 Vol. 33, No. 1.
[11] Sai Zeng IBM T.J., Ioana Boier-Martin, Prem Melville, Christian A. Lang, Conrad Murphy "Using predictive analysis to improve invoice-to-cash collection", KDD'08, Las Vegas, Nevada, USA, August 24–27, 2008.
[12] P. S. Bradley Usama M. Fayyad O. L. Mangasarian "Mathematical programming for data mining: formulations and challenges", Journal on Computing 11, 1999, 217-238.
[13] Ian H. Witten, Eibe Frank "Data Mining Practical Machine Learning Tools and Techniques" published by Morgan Kufman chapter 1 2005 Second Edition.
[14] L. Breiman, J. Friedman, C. J. Stone, and R. A.Olshen. "Classification and regression trees". Chapman & Hall/CRC, 1984
[15] Saeed Abu-Nimeh1, Dario Nappa, Xinlei Wang, and Suku Nair "A comparison of machine learning techniques for phishing detection APWG eCrim researchers summit", Pitts burgh, PA, USA October 4-5, 2007.
[16] K. P. Adhiya S. R. Kolhe Sandip S.Patil "Tracking and identification of suspicious an abnormal behaviors using supervised machine learning technique" ICAC3'09, Mumbai, Maharashtra, India January 23–24, 2009.
[17] Yong Wang Margaret Martonosi Li-Shiuan Peh "Predicting link quality using supervised learning in wireless sensor networks" Second International Workshop on Multihop Ad Hoc Networks: from Theory to Reality (REALMAN.06) ACM, 2006.
[18] Nicolai Meinshausen "Quantile regression forests", Journal of Machine Learning Research 7 (2006) 983–999.
[19] Nicolas Garcıa-Pedrajas Cordoba, Colin Fyfe Nonlinear "Boosting projections for ensemble construction" Journal of Machine Learning Research 8 (2007) 1-33.
[20] Yoav Freund and Robert E. Schapire Response to Mease and Wyner, "Evidence contrary to the statistical view of boosting", 2008, JMLR 9:131–156.
[21] Tao Hong Ashok Samal Leen-Kiat Soh "Computing information gain for spatial data support" ACM GIS '08, Irvine, CA, USA November 5-7, 2008.
[22] David Mease Abraham Wyner Evidence "Contrary to the statistical view of boosting" Journal of Machine Learning Research 9 (2008) 131-156
[23] Faisal Zaman Hideo Hirose "A robust bagging method using median as a combination rule" Proceedings of the 2008 IEEE 8th International Conference on Computer and Information Technology Workshops (2008) Pages 55-60
[24] Yue Huang Paul McCullagh Norman Black Roy Harper "Feature selection and classification model construction on type 2 diabetic patients' data" , Artificial Intelligence in Medicine Volume 41 , Issue 3 (November 2007)
[25] Mehmed Kantardzic "Data Mining: Concepts, Models, Methods, and Algorithms" published by John Wiley & Sons 2003 Chapter 7
[26] Pieter Adriaans and Dolf Zantinge Two Crows Corporation, "Introduction to Data Mining and Knowledge Discovery", Third Edition (Potomac, MD: Two Crows Corporation, 1999); Data Mining (New York: Addison Wesley, 1996)
[27] Robert E. Schapire "Theoretical views of boosting. In computational learning theory" Fourth European Conference, EuroCOLT'99, 1999 pages 1-10.
[28] Robert E. Schapire, "A brief introduction to boosting." In Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence, 1999
[29] John C. Henderson Eric Brill "Bagging and boosting a Treebank parser" ACM International Conference Proceeding Series; Vol. 4 Proceedings of the 1st North American chapter of the Association for Computational Linguistics conference table of contents Seattle, Washington 2000, Pages: 34 – 41.

Dr. Abdul Aziz did his M.Sc. from University of the Punjab, Pakistan in 1989; M.Phil and Ph.D in Computer Science from University of East Anglia, UK. He secured many honors and awards during his academic career from various institutions. He is currently working as full Professor at the Pak-American Institute of Management Sciences, Pakistan. He is the founder and Chair of Data Mining Research Group at Pak-Aims. Dr. Aziz has delivered lectures at many universities as guest speaker. He has published large number of research papers in different refereed international journals and conferences. His research interests include Knowledge Discovery in Databases (KDD) - Data Mining, Pattern Recognition, Data Warehousing and Machine Learning. He is member of editorial board for various well known journals and international conferences including IEEE publications.

Natalia Ahmed is a faculty member at University of Hail, Kingdom of Saudi Arabia. Her research interest includes Data Mining and Advance Database Management Systems. She has published her research work in various well known journals.