A HIGHER EDUCATION PREDICTIVE MODEL USING RANDOM FOREST ALGORITHM

[1]Saravanan N, [2]Manjula D
[1]Assistant Professor, [2]II MCA Student
[1 2]Department of Computer Application,
[1 2]Priyadarshini Engineering College, Vaniyambadi, Vellore, Tamilnadu, India

Abstract : Although the educational level of population has improved in the last decades, the statistics keep Portugal at Europe's tail end due to its high student failure rates. On the other hand, the fields of Business Intelligence (BI)/Data Mining (DM), which aim at extracting high-level knowledge from raw data, offer interesting automated tools that can aid the education domain. The present work intends to approach student achievement in secondary education using BI/DM techniques.. The two core classes (i.e. Mathematics and Portuguese) were modeled under binary/five-level classification and regression tasks. Also, four DM models (i.e. Decision Trees, Random Forest, Neural Networks and Support Vector Machines) and three input selections (e.g. with and without previous grades) were tested. The results show that a good predictive accuracy can be achieved, provided that the first and/or second school period grades are available. As a direct outcome of this research, more efficient student prediction tools can be developed, improving the quality of education and enhancing school resource management.

This paper proposes the use of data available at some university to access the variables that can best predict student progression. We combine virtual learning environment(VLE) and management information systems student records datasets and apply the random forest(RF) algorithm to ascertain which variables. RF demand useful in this case because of the large amount of data available for analysis.

***IndexTerms* - Business Intelligence in Education, Classification and Regression, Decision Trees, Random Forest**

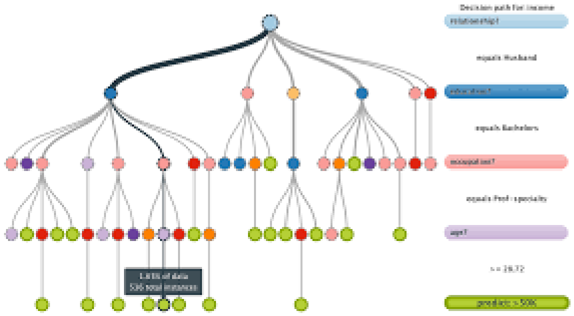
I. INTRODUCTION

Education is a key factor for achieving a long-term economic progress. During the last decades, the Portuguese educational level has improved. However, the statistics keep the Portugal at Europe's tail end due to its high student failure and dropping out rates. For example, in 2006 the early school leaving rate in Portugal was 40%for 18 to 24 year olds, while the European Union average value was just 15% (Eurostat 2007). In particular, failure in the core classes of Mathematics and Portuguese (the native language) is extremely serious, since they provide fundamental knowledge for the success in the remaining school subjects (e.g. physics or history). On the other hand, the interest in Business Intelligence (BI)/Data Mining (DM) (Turban et al. 2007), arose due to the advances of Information Technology, leading to an exponential growth of business and organizational databases. All this data holds valuable information, such as trends and patterns, which can be used to improve decision making and optimize success. students of Singapore for remedial classes. The input variables included demographic attributes (e.g. sex, region) and school performance over the past years and the proposed solution outperformed the traditional allocation procedure. In 2003 (Minaei-Bidgoli et al. 2003), The two core classes (i.e. Mathematics and Portuguese) will be modeled under three DM goals:

i) binary classification (pass/fail);
ii) classification with five levels (from I very good or excellent to V - insufficient); and
iii) Regression, with a numeric output that ranges between zero (0%) and twenty (100%).

Random Forests:

Random Forests(tm) is a trademark of Leo Breiman and AdeleCutler and is licensed exclusively to Salford Systems for the commercial release of the software.



Overview:

We assume that the user knows about the construction of single classification trees. Random Forests grows many classification trees. To classify a new object from an input vector, put the input vector down each of the trees in the forest. Each tree gives a classification, and we say the tree "votes" for that class. The forest chooses the classification having the most votes (over all the trees in the forest).

Each tree is grown as follows:

1. If the number of cases in the training set is N, sample N cases at random - but with replacement, from the original data. This sample will be the training set for growing the tree.

2.If there are M input variables, a number m<<M is specified such that at each node, m variables are selected at random out of the M and the best split on these m is used to split the node. The value of m is held constant during the forest growing.

3. Each tree is grown to the largest extent possible. There is no pruning.

In the original paper on random forests, it was shown that the forest error rate depends on two things:

The correlation between any two trees in the forest. Increasing the correlation increases the forest error rate.

The strength of each individual tree in the forest. A tree with a low error rate is a strong classifier. Increasing the strength of the individual trees decreases the forest error rate.

Features of Random Forests:

- It is unexcelled in accuracy among current algorithms.
- It runs efficiently on large data bases.
- It can handle thousands of input variables without variable deletion.
- It gives estimates of what variables are important in the classification.
- It offers an experimental method for detecting variable interactions.

Random forests algorithm:

The random forests algorithm (for both classification and regression) is as follows:

1. Draw ntree bootstrap samples from the original data.

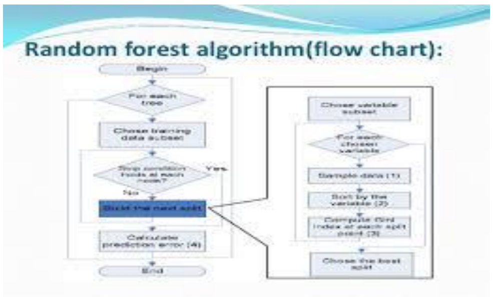
2. For each of the bootstrap samples, grow an unpruned classification or regression tree, with the following modification: at each node, rather than choosing the best split among all predictors, randomly sample mtry of the predictors and choose the best split from among those variables. (Bagging can be thought of as the special case of random forests obtained when mtry = p, the number of predictors.)

3. Predict new data by aggregating the predictions of the ntree trees (i.e., majority votes for classification, average for regression).

An estimate of the error rate can be obtained, based on the training data, by the following:

1. At each bootstrap iteration, predict the data not in the bootstrap sample (what Breiman calls “out-of-bag”, or OOB, data) using the tree grown with the bootstrap sample.

2. Aggregate the OOB predictions. (On the average, each data point would be out-of-bag around 36% of the times, so aggregate these predictions.) Calculate the error rate, and call it the OOB estimate of error rate.


Random forest algorithm(flow chart):
Begin
For each tree
Chose training data subset
Stop condition holds at each node?
Yes
No
Build the next split
Calculate prediction error (4)
End
Chose variable subset
For each chosen variable
Sample data (1)
Sort by the variable (2)
Compute Gini index at each split point (3)
Chose the best split

How random forests work:

- To understand and use the various options, further information about how they are computed is useful.
- Most of the options depend on two data objects generated by random forests.
- When the training set for the current tree is drawn by sampling with replacement, about one-third of the cases are left out of the sample.
- This oob (out-of-bag) data is used to get a running unbiased estimate of the classification error as trees are added to the forest. It is also used to get estimates of variable importance.
- After each tree is built, all of the data are run down the tree, and proximities are computed for each pair of cases.
- If two cases occupy the same terminal node, their proximity is increased by one.
- At the end of the run, the proximities are normalized by dividing by the number of trees.
- Proximities are used in replacing missing data, locating outliers, and producing illuminating low-dimensional views of the data.

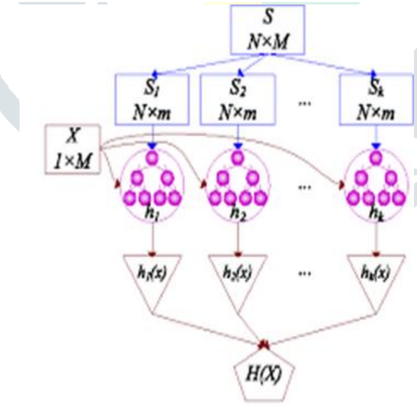
Proximities:

These are one of the most useful tools in random forests. The proximities originally formed a NxN matrix.

After a tree is grown, put all of the data, both training and oob, down the tree. If cases k and n are in the same terminal node increase their proximity by one. At the end, normalize the proximities by dividing by the number of trees.

Users noted that with large data sets, they could not fit an NxN matrix into fast memory. A modification reduced the required memory size to NxT where T is the number of trees in the forest. To speed up the computation-intensive scaling and iterative missing value replacement, the user is given the option of retaining only then largest proximities to each case. an NxN matrix, the computational burden may be time consuming. We advise taking nxn considerably smaller than the sample size to make this computation faster.

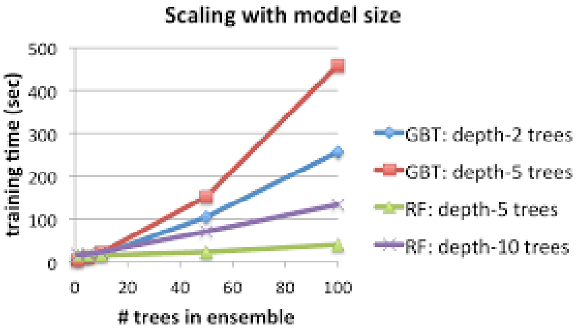
. Plotting the second scaling coordinate versus the first usually gives the most illuminating view.



rocess of the construction of the RF algorithm

Scaling:

The proximities between cases n and k form a matrix {prox(n,k)}. From their definition, it is easy to show that this matrix is symmetric, positive definite and bounded above by 1, with the diagonal elements equal to 1. It follows that the values 1-prox(n,k) are squared distances in a Euclidean space of dimension not greater than the number of cases. For more background on scaling see "Multidimensional Scaling" by T.F. Cox and M.A. Cox.



In metric scaling, the idea is to approximate the vectors x(n) by the first few scaling coordinates. This is done in random forests by extracting the largest few eigenvalues of the cv matrix, and their corresponding eigenvectors . The two dimensional plot of the ith scaling coordinate vs. the jth often gives useful information about the data. The most useful is usually the graph of the 2nd vs. the 1st. Since the Eigen functions are the top few of

Prototypes:

Prototypes are a way of getting a picture of how the variables relate to the classification. For the jth class, we find the case that has the largest number of class j cases among its k nearest neighbors, determined using the proximities. Among these k cases we find the median, 25th percentile, and 75th percentile for each variable. The medians are the prototype for class j and the quartiles give an estimate of is stability. For the second prototype, we repeat the procedure but only consider cases that are not among the original k, and so on. When we ask for prototypes to be output to the screen or saved to a file, prototypes for continuous variables are standardized by subtracting the 5th percentile and dividing by the difference between the 95th and 5th percentiles. For categorical variables, the prototype is the most frequent value. When we ask for prototypes to be output to the screen or saved to a file, all frequencies are given for categorical variables.

Unsupervised learning:

In unsupervised learning the data consist of a set of x -vectors of the same dimension with no class labels or response variables. There is no figure of merit to optimize, leaving the field open to ambiguous conclusions. The usual goal is to cluster the data - to see if it falls into different piles, each of which can be assigned some meaning.

The approach in random forests is to consider the original data as class 1 and to create a synthetic second class of the same size that will be labeled as class 2. The synthetic second class is created by sampling at random from the univariate distributions of the original data. Here is how a single member of class two is created - the first coordinate is sampled from the N values {x(1,n)}. The second coordinate is sampled independently from the N values {x(2,n)}, and so forth.

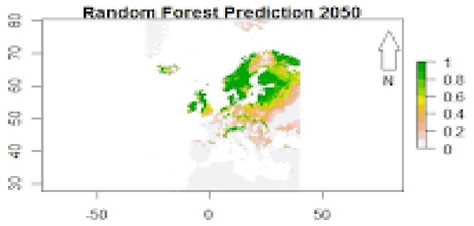
Thus, class two has the distribution of independent random variables, each one having the same univariate distribution as the corresponding variable in the original data. Class 2 thus destroys the dependency structure in the original data. But now, there are two classes and this artificial two-class problem can be run through random forests. This allows all of the random forests options to be applied to the original unlabeled data set.

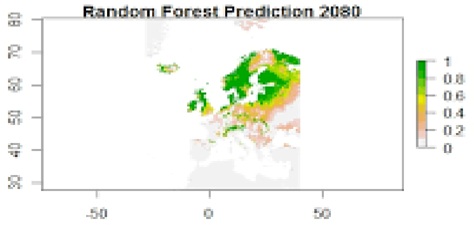
If the oob misclassification rate in the two-class problem is, say, 40% or more, it implies that the x -variables look too much like independent variables to random forests. The dependencies do not have a large role and not much discrimination is taking place. If the misclassification rate is lower, then the dependencies are playing an important role.

Formulating it as a two class problem has a number of payoffs. Missing values can be replaced effectively. Outliers can be found. Variable importance can be measured. Scaling can be performed (in this case, if the original data had labels, the unsupervised scaling often retains the structure of the original scaling). But the most important payoff is the possibility of clustering.

Balancing prediction error:

In some data sets, the prediction error between classes is highly unbalanced. Some classes have a low prediction error, others a high. This occurs usually when one class is much larger than another**.** Then random forests, trying to minimize overall error rate, will keep the error rate low on the large class while letting the smaller classes have a larger error rate





Variable importance:

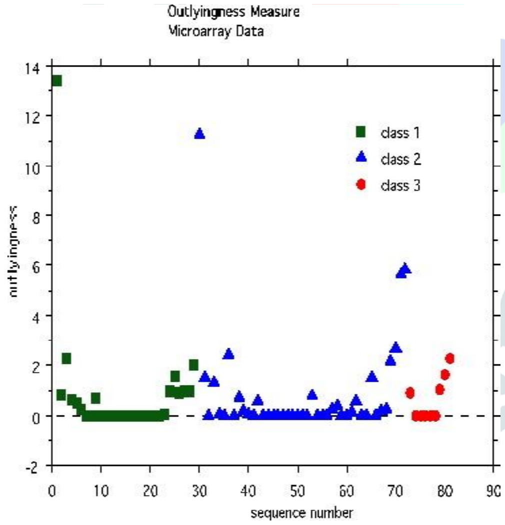
The variable importance are critical. The run computing importance is done by switching imp =0 to imp =1 in the above parameter list.

The output has four columns:

- Gene number
- The raw importance score
- The z-score obtained by dividing the raw score by its standard error
- The significance level.

Outliers:

An outlier is a case whose proximities to all other cases are small**.** There are two possible outliers-one is the first case in class 1, the second is the first case in class 2.



Conclusion:

This paper lists a high scope for the students to decide for the brighter future with specific and accurate analysis. It will help the educational system to monitor the students' performance in a systematic way. In this paper a simple data mining based prediction model were presented.in order to assists academic stakeholders to improve academic performance which is the main goal of study.

References

[1] Eurostat, 2007. Early school-leavers. http://epp.eurostat.ec.europa.eu/.

[2] Flexer A., 1996. Statistical Evaluation of Neural Networks Experiments: Minimum Requirements and Current Practice. In Proceedings of the 13th European Meeting on Cybernetics and Systems Research. Vienna, Austria, vol. 2, 1005–1008.

[3] Hastie T.; Tibshirani R.; and Friedman J., 2001. The Elements of Statistical Learning: Data Mining, Inference, and Prediction. Springer-Verlag, NY, USA. Kotsiantis S.; Pierrakeas C.; and Pintelas P., 2004. Predicting Students' Performance in Distance Learning

[4] Using Machine Learning Techniques. Applied Artificial Intelligence (AAI), 18, no. 5, 411–426.

[5] Applications in Higher Education. New Directions for Institutional Research, 113, 17–36.

[6] Ma Y.; Liu B.; Wong C.; Yu P.; and Lee S., 2000. Targeting the right students using data mining. In Proc. of 6th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. Boston, USA, 457–464.

[7] Minaei-Bidgoli B.; Kashy D.; Kortemeyer G.; and Punch W., 2003. Predicting student performance: an application of data mining methods with an educational web-based system. In Proc. of IEEE Frontiers in Education. Colorado, USA, 13–18.

[8] Pardos Z.; Heffernan N.; Anderson B.; and Heffernan C., 2006. Using Fine-Grained Skill Models to Fit Student Performance with Bayesian Networks. In Proc. of 8th Int. Conf. on Intelligent Tutoring Systems. Taiwan.

[9] Pritchard M. and Wilson S., 2003. Using Emotional and Social Factors To Predict Student Success. Journal of College Student Development, 44, no. 1, 18–28.
Development Core Team, 2006. R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria, ISBN 3900051-00-3, http://www.R-project.org.