International Journal of Research in Advent Technology (IJRAT) Special Issue
E-ISSN: 2321-9637
Available online at www.ijrat.org
National Conference on "Role of Information Technology in Social Innovations"
26th & 27th February 2019

Estimation of Future Result of Students

Anushika Singh[1], Chandan Singh Kholiya[2], Bindu Garg[3]
Department of Computer Engineering[1,2,3],Bharati Vidyapeeth (Deemed to be University) College of Engineering, Pune[1,2,3]
Email: anushikasingh1@gmail.com[1] ,cskholiya96@gmail.com[2]

Abstract- Over several years, many statistical tools have been used to analyze and predict students' performance from different point of view. One of the major challenges today is to predict the paths of students for higher education through the educational process. Prediction analysis in earlier stage depends on many different factors. But Data mining techniques could be more useful and easier for this kind of job. Data mining techniques are widely used in educational field to find new hidden patterns from student's data. And these hidden patterns which are discovered can be used to understand the problem arise in the educational field. Data Mining (DM), or Knowledge Discovery in Databases (KDD), is a process to extract meaningful and useful information from large set of raw data. Data mining techniques apply various methods in order to discover and extract patterns from stored databased on collected students' information, different data mining techniques need to be used. The purpose of this project is used data mining software for the prediction of final student mark based on parameters in the given dataset. The dataset contains information about a school's students and their various social aspects and marks.

Index Terms-Weka, prediction, student result

1. INTRODUCTION

Data Mining (DM) is a growing topic in the field Computer Sciences. Due to the large amounts of data and the urgent need to convert such data into useful information and knowledge, data mining has gained a great importance in the information industry and in society in recent years. Data Mining focuses on the extraction of hidden knowledge from various data warehouses, data marts, and repositories. Large amount of data becomes useless without its proper use. Knowledge data discovery (KDD) is similar to data mining but they are really different in an essential point. Data mining and knowledge discovery is used to derive common expressions of characteristics that are shared by all elements in a dataset. They both have techniques that can be used to extract useful information from large amount of data in the database. The results of applying the DM algorithms on any given or manual-generated dataset is known as Rule Discovery. There are mainly two types of rules, the production rules and the association rules. The production rules are a common formalism for expressing knowledge in expert systems. Decision Trees rules can be also transformed into the production rules. The association rules were used to find a relationship among sales of different items from the analysis of a big data.

Educational data mining is an emerging field in the area of data mining. In this competitive world, the education department also uses data mining to explore and analyse student performance, predict their results to prevent bad grades and focus on getting better results. The quality of education needs to be improved and educational data mining is a tool for this improvement. Student's performance depends on various things like social, personal, economic and other environmental factors. Educational institutes may utilize the outcome of the experimental results to understand the trends and behaviours in students' performance which may help to design new strategies for better performance of students. There are a number of classification algorithms: Decision Tree, K-Nearest neighbour, Naïve Bayes, Random Forest, Support Vector Machines etc. In this research, we are going to use some of them for mining the academic students' performance: J48, NaiveBayes, RandomForest, RandomTree, ZeroR, DecisionTable, SMO classification algorithms. Classification is one of the predictive tasks and is the most commonly used data mining technique in predicting the students' performance in educational institutes. Several attributes were considered in our study. First, we found the high influence attributes. We removed the unnecessary attributes from the dataset to extract useful and meaningful information through feature selection. It makes the mining process faster, valuable and meaningful. In the study, final marks are selected as dependent attribute. WEKA (Waikato Environment for Knowledge Analysis) is used as the data mining tool for study. WEKA is an open source tool written in Java that is widely used by the data miners. WEKA implements most of the machine learning algorithms and visualizes its results as well.

2. LITERATURE REVIEW

We did a background study to review similar existing systems used to perform student performance analysis. Some existing system are chosen because these systems are similar to the proposed system.

The Prediction of Students' Academic Performance Using Classification Data Mining Techniques [1] - It is a framework to predict the academic performance of the first-year bachelor students of computer science course. The data collected contained various aspects of students' records including previous academic records, family background and demographics. It contained 8 years of data. Three classifiers viz. Naïve Bayes, Decision Tree and Rule-Based classifiers are applied to find the academic performance of students. The analysis showed that Rule Based classifier was the best among the other classifiers as its accuracy was found to be 71.3% for that dataset.

Faculty Support System (FSS) [2] - FSS is able to analyse the students' data dynamically as it is able to update of students' data dynamically with the flow of time to create or add a new rule. Classification technique is used to predict the students' performance. FSS focus on the identification of factors that contribute to performance of students in a particular course.

Prediction of Students Outcome Using Data Mining Techniques [3] - It is a data model to predict student's future learning outcomes using senior students' dataset. They compared the data mining classification algorithms and found that J48 algorithm was best suited for such job based on their data.

A Comparative Study to Predict Student's Performance Using Educational Data Mining Techniques [4] - Author conducted a study to find that high influence attributes may be selected carefully to predict student performance. Feature selection may be used before classification for such job. They used Bayesian Network and Decision Tree algorithms for classification and prediction of student performance. The Feature Selection methods showed that student's attendance and Grade Point Average in the first semester topped the list of features. When the accuracy rate was considered, the Bayesian Network outperformed the Decision Tree classification in their case.

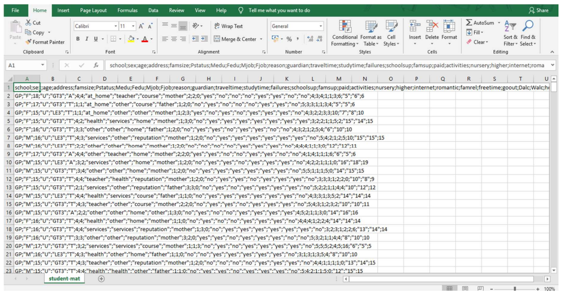
Analysis of Students' Performance by Using Different Data Mining Classifiers [5] - Author used WEKA tool to evaluate the performance of the university students. He found that the accuracy of the classifier algorithms depends upon size and nature of data. The author used Naïve Bayesian Network, Bayes, J48, Neural Network, and ID3 classification techniques. He found that Bayesian Network works better than others in terms of accuracy.

Predicting Students Yearly Performance using Neural Network [6] – The author used Cumulative Grade Point Average (CGPA) for prediction of students' yearly performance. The dataset used was from Bangabandhu Sheikh Mujibur Rahman Science and Technology University students' records. The authors used neural network technique for prediction and it was compared with the real CGPA of the student.

3. DESIGN AND IMPLEMENTATION

We implemented the system using the following steps:-

Dataset: - This data is taken from secondary education of two Portuguese schools. The data attributes include student grades, demographic, social and school related features and it was collected by using school reports and questionnaires. The dataset was in csv format.

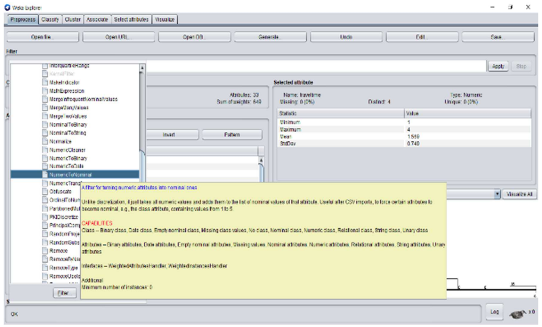


```
school;sex;age;address;famsize;Pstatus;Medu;Fedu;Mjob;Fjob;reason;guardian;traveltime;studytime;failures;schoolsup;famsup;paid;activities;nursery;higher;internet;roma
```

```
school;sex;age;address;famsize;Pstatus;Medu;Fedu;Mjob;Fjob;reason;guardian;traveltime;studytime;failures;schoolsup;famsup;paid;activities;nursery;higher;internet;romantic;famrel;freetime;goout;Dalc;Walc;he
GP;"F";18;"U";"GT3";"A";4;4;"at_home";"teacher";"course";"mother";2;2;0;"yes";"no";"no";"no";"yes";"yes";"no";"no";4;3;4;1;1;3;6;"5";"6";6
GP;"F";17;"U";"GT3";"T";1;1;"at_home";"other";"course";"father";1;2;0;"no";"yes";"no";"no";"no";"yes";"yes";"no";5;3;3;1;1;3;4;"5";"5";6
GP;"F";15;"U";"LE3";"T";1;1;"at_home";"other";"other";"mother";1;2;3;"yes";"no";"yes";"no";"yes";"yes";"yes";"no";4;3;2;2;3;3;10;"7";"8";10
GP;"F";15;"U";"GT3";"T";4;2;"health";"services";"home";"mother";1;3;0;"no";"yes";"yes";"yes";"yes";"yes";"yes";"yes";3;2;2;1;1;5;2;"15";"14";15
GP;"F";16;"U";"GT3";"T";3;3;"other";"other";"home";"father";1;2;0;"no";"yes";"yes";"no";"yes";"yes";"no";"no";4;3;2;1;2;5;4;"6";"10";10
GP;"M";16;"U";"LE3";"T";4;3;"services";"other";"reputation";"mother";1;2;0;"no";"yes";"yes";"yes";"yes";"yes";"yes";"no";5;4;2;1;2;5;10;"15";"15";15
GP;"M";16;"U";"LE3";"T";2;2;"other";"other";"home";"mother";1;2;0;"no";"no";"no";"no";"yes";"yes";"yes";"no";4;4;4;1;1;3;0;"12";"12";11
GP;"F";17;"U";"GT3";"A";4;4;"other";"teacher";"home";"mother";2;2;0;"yes";"yes";"no";"no";"yes";"yes";"no";"no";4;1;4;1;1;1;6;"6";"5";6
GP;"M";15;"U";"LE3";"A";3;2;"services";"other";"home";"mother";1;2;0;"no";"yes";"yes";"no";"yes";"yes";"yes";"no";4;2;2;1;1;1;0;"16";"18";19
GP;"M";15;"U";"GT3";"T";3;4;"other";"other";"home";"mother";1;2;0;"no";"yes";"yes";"yes";"yes";"yes";"yes";"no";5;5;1;1;1;5;0;"14";"15";15
GP;"F";15;"U";"GT3";"T";4;4;"teacher";"health";"reputation";"mother";1;2;0;"no";"yes";"yes";"no";"yes";"yes";"yes";"no";3;3;3;1;2;2;0;"10";"8";9
GP;"F";15;"U";"GT3";"T";2;1;"services";"other";"reputation";"father";3;3;0;"no";"yes";"no";"yes";"yes";"yes";"yes";"no";5;2;2;1;1;4;4;"10";"12";12
GP;"M";15;"U";"LE3";"T";4;4;"health";"services";"course";"father";1;1;0;"no";"yes";"yes";"yes";"yes";"yes";"yes";"no";4;3;3;1;3;5;2;"14";"14";14
GP;"M";15;"U";"GT3";"T";4;3;"teacher";"other";"course";"mother";2;2;0;"no";"yes";"yes";"no";"yes";"yes";"yes";"no";5;4;3;1;2;3;2;"10";"10";11
GP;"M";15;"U";"GT3";"A";2;2;"other";"other";"home";"other";1;3;0;"no";"yes";"no";"no";"yes";"yes";"yes";"yes";4;5;2;1;1;3;0;"14";"16";16
GP;"F";16;"U";"GT3";"T";4;4;"health";"other";"home";"mother";1;1;0;"no";"yes";"no";"no";"yes";"yes";"yes";"no";4;4;4;1;2;2;4;"14";"14";14
GP;"F";16;"U";"GT3";"T";4;4;"services";"services";"reputation";"mother";1;3;0;"no";"yes";"yes";"yes";"yes";"yes";"yes";"no";3;2;3;1;2;2;6;"13";"14";14
GP;"F";16;"U";"GT3";"T";3;3;"other";"other";"reputation";"mother";3;2;0;"yes";"yes";"no";"yes";"yes";"yes";"no";"no";5;3;2;1;1;4;4;"8";"10";10
GP;"M";17;"U";"GT3";"T";3;2;"services";"services";"course";"mother";1;1;3;"no";"yes";"no";"yes";"yes";"yes";"yes";"no";5;5;5;2;4;5;16;"6";"5";5
GP;"M";16;"U";"LE3";"T";4;3;"health";"other";"home";"father";1;1;0;"no";"no";"yes";"yes";"yes";"yes";"yes";"no";3;1;3;1;3;5;4;"8";"10";10
GP;"M";15;"U";"GT3";"T";4;3;"teacher";"other";"reputation";"mother";1;2;0;"no";"no";"no";"no";"yes";"yes";"yes";"no";4;4;1;1;1;1;0;"13";"14";15
GP;"M";15;"U";"GT3";"T";4;4;"health";"health";"other";"father";1;1;0;"no";"yes";"yes";"no";"yes";"yes";"yes";"no";5;4;2;1;1;5;0;"12";"15";15
```

Selecting attributes: - Out of the 30 attributes in our dataset we selected different combination of attributes out of which we finally selected 11 attributes which gave the best results for our dataset.

Preprocessing: - Preprocessing is the first step of evaluation of any Weka project. We selected the source file from our system and converted it to be able to read by the system. Different filters are used in weka to perform data cleaning. First we used the remove filter to remove the rest of the attributes. Then for the association rules we needed to convert numerical values to nominal values so we used the NumericToNominal filter for that conversion. Below is the picture of the preprocessing tab of our project before and after applying the filters.



Attributes: 33
Sum of weights: 649

Selected attribute
Name: traveltime   Type: Numeric
Missing: 0 (0%)   Distinct: 4   Unique: 0 (0%)

Statistic	Value
Minimum	1
Maximum	4
Mean	1.569
StdDev	0.749

A filter for turning numeric attributes into nominal ones.

Unlike discretization, it just takes all numeric values and adds them to the list of nominal values of that attribute. Useful after CSV imports, to force certain attributes to become nominal, e.g., the class attribute, containing values from 1 to 5.

CAPABILITIES
Class -- Binary class, Date class, Empty nominal class, Missing class values, No class, Nominal class, Numeric class, Relational class, String class, Unary class

Attributes -- Binary attributes, Date attributes, Empty nominal attributes, Missing values, Nominal attributes, Numeric attributes, Relational attributes, String attributes, Unary attributes

Interfaces -- WeightedAttributesHandler, WeightedInstancesHandler

Additional
Minimum number of instances: 0

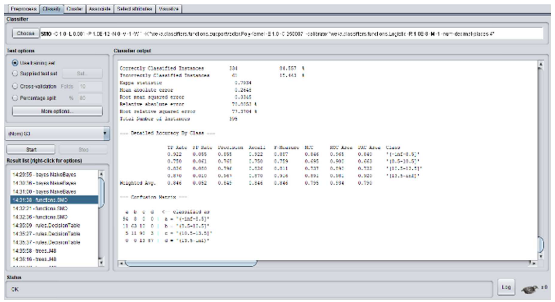
OK

Classification: - Classifiers are used to predict nominal and numeric quantities in Weka. In our project we used different classifiers like J48, RandomForest, RandomTree, NaiveBayes, DecisionTree etc. We can predict result in weka using three methods which are as follows: -

i).Training and test set

Steps for using training and test set are as follows: -

a) First select the classifier which you want to use to classify the data.
b) Select “Use training set” in the test options in classify tab.
c) Click “Start” to start the training.
d) Now select the “Supplied test set” from the test options. Now a dialog box would appear on the screen.
e) Click on “Open File” and select the test file (the test file and training file must have same attributes).
f) Now click on close to close the dialog box.
g) Click on “Start” to start the testing.



SMO -C 1.0 -L 0.001 -P 1.0E-12 -N 0 -V -1 -W 1 -K "weka.classifiers.functions.supportVector.PolyKernel -E 1.0 -C 250007" -calibrator "weka.classifiers.functions.Logistic -R 1.0E-8 -M -1 -num-decimal-places 4"

● Use training set
○ Supplied test set
○ Cross-validation Folds 10
○ Percentage split % 80

14:29:55 - bayes.NaiveBayes
14:30:36 - bayes.NaiveBayes
14:31:00 - bayes.NaiveBayes
14:31:38 - functions.SMO
14:32:21 - functions.SMO
14:32:36 - functions.SMO
14:35:09 - rules.DecisionTable
14:35:27 - rules.DecisionTable
14:35:37 - rules.DecisionTable
14:35:58 - trees.J48
14:36:16 - trees.J48

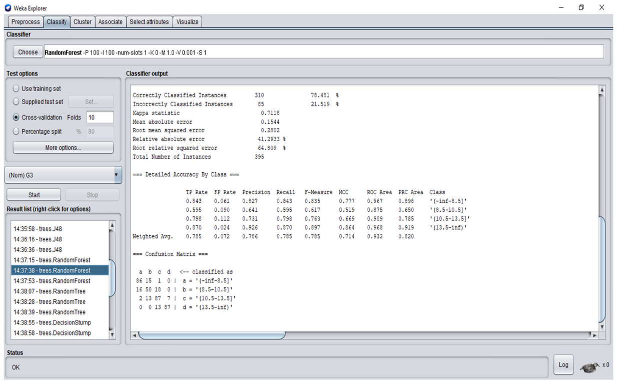
```
Correctly Classified Instances         334               84.557  %
Incorrectly Classified Instances        61               15.443  %
Kappa statistic                          0.7934
Mean absolute error                      0.2648
Root mean squared error                  0.3345
Relative absolute error                 70.8053 %
Root relative squared error             77.3704 %
Total Number of Instances              395

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.922    0.055    0.855      0.922    0.887      0.846    0.965     0.840     '(-inf-8.5]'
                 0.750    0.061    0.768      0.750    0.759      0.695    0.900     0.663     '(8.5-10.5]'
                 0.826    0.080    0.796      0.826    0.811      0.737    0.890     0.722     '(10.5-13.5]'
                 0.870    0.010    0.967      0.870    0.916      0.891    0.981     0.920     '(13.5-inf)'
Weighted Avg.    0.846    0.052    0.849      0.846    0.846      0.795    0.934     0.790

=== Confusion Matrix ===

  a  b  c  d   <-- classified as
 94  8  0  0 |  a = '(-inf-8.5]'
 11 63 10  0 |  b = '(8.5-10.5]'
  5 11 90  3 |  c = '(10.5-13.5]'
  0  0 13 87 |  d = '(13.5-inf)'
```

OK

ii).Cross-validation

Steps for using cross-validation are as follows: -

a) First select the classifier which you want to use to classify the data.
b) Select “Cross-validation” in the test options in classify tab.
c) Enter the number of folds (default 10)
d) Click “Start” to start the classification.



RandomForest -P 100 -I 100 -num-slots 1 -K 0 -M 1.0 -V 0.001 -S 1

○ Use training set
○ Supplied test set
● Cross-validation Folds 10
○ Percentage split % 80

14:35:58 - trees.J48
14:36:16 - trees.J48
14:36:36 - trees.J48
14:37:15 - trees.RandomForest
14:37:38 - trees.RandomForest
14:37:53 - trees.RandomForest
14:38:07 - trees.RandomTree
14:38:28 - trees.RandomTree
14:38:39 - trees.RandomTree
14:38:55 - trees.DecisionStump
14:38:58 - trees.DecisionStump

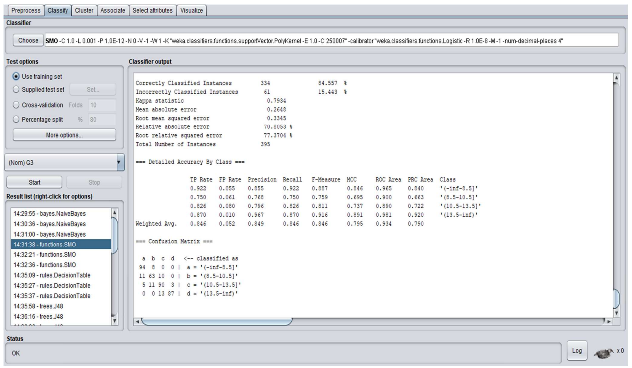
```
Correctly Classified Instances         310               78.481  %
Incorrectly Classified Instances        85               21.519  %
Kappa statistic                          0.7118
Mean absolute error                      0.1544
Root mean squared error                  0.2802
Relative absolute error                 41.2933 %
Root relative squared error             64.809  %
Total Number of Instances              395

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.843    0.061    0.827      0.843    0.835      0.777    0.967     0.898     '(-inf-8.5]'
                 0.595    0.090    0.641      0.595    0.617      0.519    0.875     0.650     '(8.5-10.5]'
                 0.798    0.112    0.731      0.798    0.763      0.669    0.909     0.785     '(10.5-13.5]'
                 0.870    0.024    0.926      0.870    0.897      0.864    0.968     0.919     '(13.5-inf)'
Weighted Avg.    0.785    0.072    0.786      0.785    0.785      0.714    0.932     0.820

=== Confusion Matrix ===

  a  b  c  d   <-- classified as
 86 15  1  0 |  a = '(-inf-8.5]'
 16 50 18  0 |  b = '(8.5-10.5]'
  2 13 87  7 |  c = '(10.5-13.5]'
  0  0 13 87 |  d = '(13.5-inf)'
```

OK

iii).Percentage Split

Steps for using percentage split are as follows: -

a) First select the classifier which you want to use to classify the data.
b) Select “Percentage split” in the test options in classify tab.
c) Enter the percentage of data you want to use for training and rest of the data is used for testing the trained data.
d) Click “Start” to start the classification.



SMO -C 1.0 -L 0.001 -P 1.0E-12 -N 0 -V -1 -W 1 -K "weka.classifiers.functions.supportVector.PolyKernel -E 1.0 -C 250007" -calibrator "weka.classifiers.functions.Logistic -R 1.0E-8 -M -1 -num-decimal-places 4"

● Use training set
○ Supplied test set
○ Cross-validation Folds 10
○ Percentage split % 80

14:29:55 - bayes.NaiveBayes
14:30:36 - bayes.NaiveBayes
14:31:00 - bayes.NaiveBayes
14:31:38 - functions.SMO
14:32:21 - functions.SMO
14:32:36 - functions.SMO
14:35:09 - rules.DecisionTable
14:35:27 - rules.DecisionTable
14:35:37 - rules.DecisionTable
14:35:58 - trees.J48
14:36:16 - trees.J48

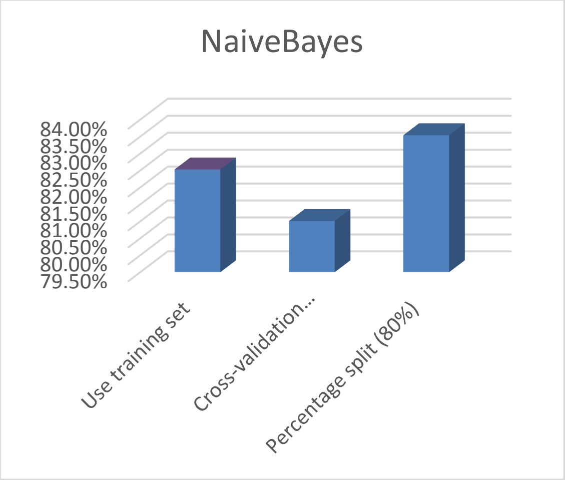
```
Correctly Classified Instances         334               84.557  %
Incorrectly Classified Instances        61               15.443  %
Kappa statistic                          0.7934
Mean absolute error                      0.2648
Root mean squared error                  0.3345
Relative absolute error                 70.8053 %
Root relative squared error             77.3704 %
Total Number of Instances              395

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.922    0.055    0.855      0.922    0.887      0.846    0.965     0.840     '(-inf-8.5]'
                 0.750    0.061    0.768      0.750    0.759      0.695    0.900     0.663     '(8.5-10.5]'
                 0.826    0.080    0.796      0.826    0.811      0.737    0.890     0.722     '(10.5-13.5]'
                 0.870    0.010    0.967      0.870    0.916      0.891    0.981     0.920     '(13.5-inf)'
Weighted Avg.    0.846    0.052    0.849      0.846    0.846      0.795    0.934     0.790

=== Confusion Matrix ===

  a  b  c  d   <-- classified as
 94  8  0  0 |  a = '(-inf-8.5]'
 11 63 10  0 |  b = '(8.5-10.5]'
  5 11 90  3 |  c = '(10.5-13.5]'
  0  0 13 87 |  d = '(13.5-inf)'
```

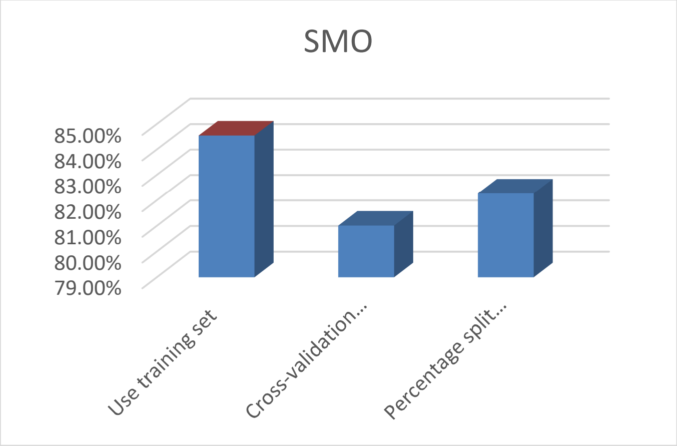
OK

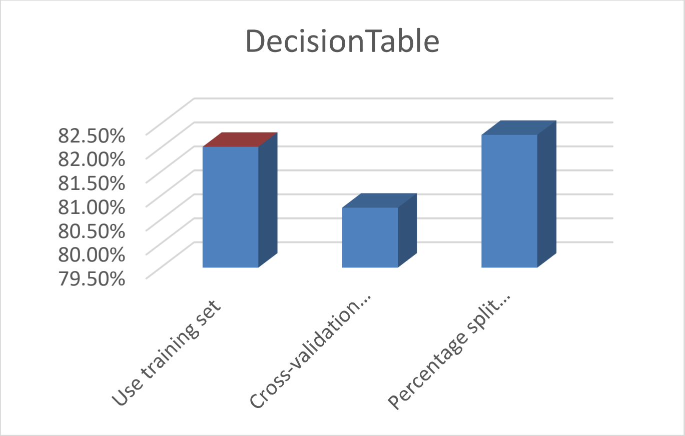
4. RESULTS

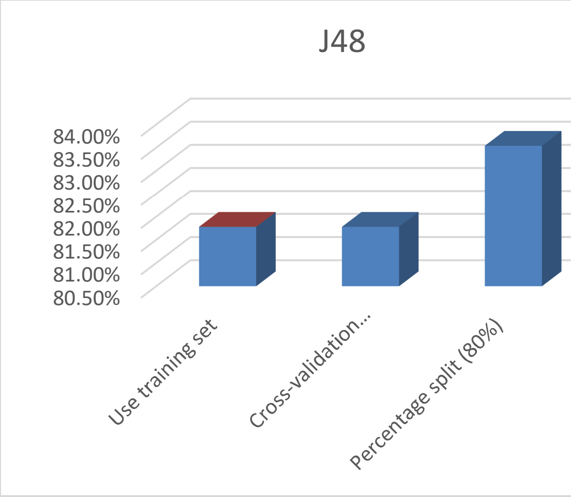
As we used different classification techniques to predict the results with different testing options each technique produced different results. The table of the techniques used and their results are as follow: -

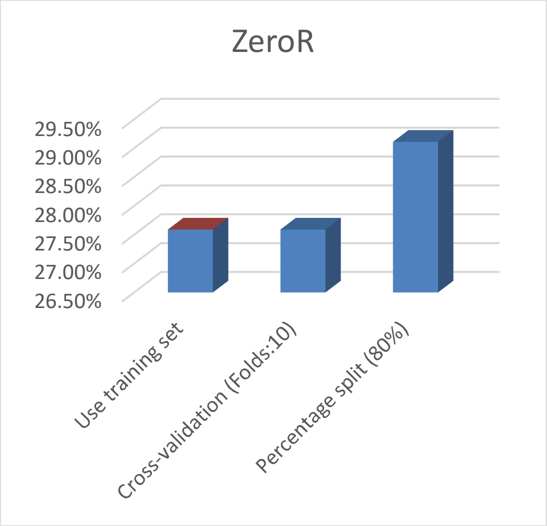
Classifier	**Test Option**	**Accuracy**
NaiveBayes	Use training set	82.53%
	Cross-validation (Folds:10)	81.01%
	Percentage split (80%)	83.54%
SMO	Use training set	84.55%
	Cross-validation (Folds:10)	81.01%
	Percentage split (80%)	82.27%
DecisionTable	Use training set	82.02%
	Cross-validation (Folds:10)	80.75%
	Percentage split (80%)	82.27%
J48	Use training set	81.78%
	Cross-validation (Folds:10)	81.78%
	Percentage split (80%)	83.54%
RandomForest	Use training set	98.73%
	Cross-validation (Folds:10)	78.48%
	Percentage split (80%)	81.01%
RandomTree	Use training set	98.73%
	Cross-validation (Folds:10)	72.65%
	Percentage split (80%)	65.82%
ZeroR	Use training set	27.59%
	Cross-validation (Folds:10)	27.59%
	Percentage split (80%)	29.11%

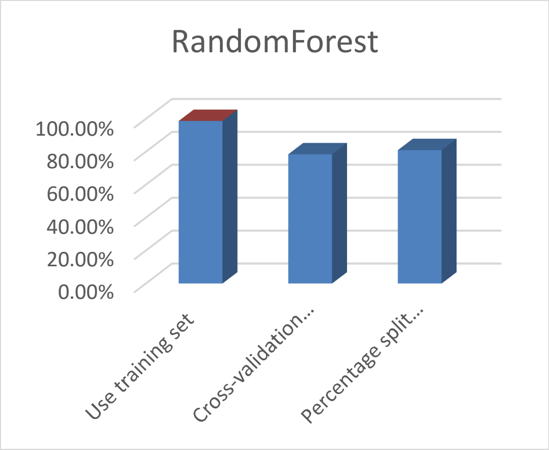


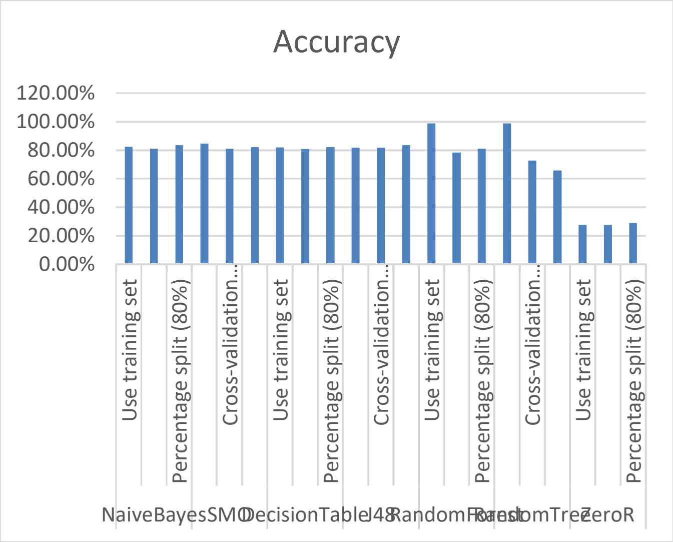


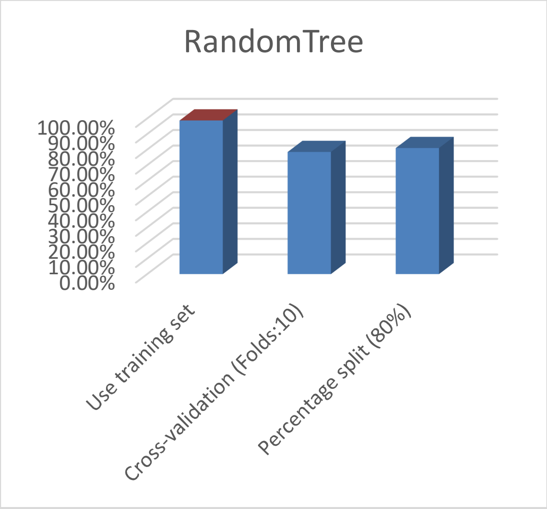












5. CONCLUSION

In this paper a dataset from a Portuguese school has been taken and analyzed. In total there were 649 instances with 33 attributes. Out of which 11 attributes were selected. Weka 3.8 was used as a data mining tool. Many different classifiers were used to predict the results such as J48, NaiveBayes, RandomForest, RandomTree, ZeroR, DecisionTable, SMO. Based on the accuracy and the classification error one may conclude that RandomForest was the most suitable algorithm for the dataset.

REFERENCES

[1] Ahmad, F., N.H. Ismail, and A. Abdulaziz, The Prediction of Students' Academic Performance

Using Classification Data Mining Techniques. Applied Mathematical Sciences, 2015. 9(129)

[2] J. Shana, and T. Venkatacalam, “A framework for dynamic Faculty Support System to analyse student course data”, International Journal of Emerging Technology and Advanced Engineering, Vol. 2, No. 7, 2012, pp.478-482.

[3] Sumitha, R. and E.S. Vinothkumar, Prediction of Students Outcome Using Data Mining Techniques. International Journal of Scientific Engineering and Applied Science (IJSEAS), 2016. 2(6)

[4] Khasanah, A.U. and Harwati, A Comparative Study to Predict Student’s Performance Using Educational Data Mining Techniques. IOP Conf. Series: Materials Science and Engineering, 2017. 215(012036)

[5] Almarabeh, H., Analysis of Students’ Performance by Using Different Data Mining Classifiers. I.J. Modern Education and Computer Science, 2017

[6] Sikder, M.F., M.J. Uddin, and S. Halder, Predicting Students Yearly Performance using Neural Network: A Case Study of BSMRSTU. 5th International Conference on Informatics, Electronics and Vision (ICIEV), 2016.