1.ABSTRACT

Data mining is the evolving process of identifying and extracting the hidden information from a data warehouse. Data Mining is widely used in business, medical, engineering and educational areas for analyzing existing data, identifying measures for improvement and also forecasting the future prospects. This study covers the application of data mining in education for predicting the academic performance of the students. Educational Data Mining(EDM) plays a dominant role in the data mining era. There is an essential need to identify effective algorithms for predicting the student's performance. The dataset used in this research is taken from the University of Minho, Mathematics department which consists of 33 attributes and 650 observations. The algorithms used in this study to predict the academic performance of the students are Decision tree, Random Forest, Gradient Boosted tree, Logistic Regression and Multilayer Perceptron. Ensemble model is created by combining the Decision tree, C5.0 and Random Forest along with cross validation to improve the accuracy of the algorithms. The use of effective EDM techniques and tools would enable educators to improve the process by identifying any existing lacunae. EDM helps in developing a warning system for identifying weak student's prior and give adequate training to improve the academic performance of the students.

Keywords:

Student Performance Prediction, Educational Data Mining, Data Mining Technique, Academic Performance, Decision tree, Random Forest, Gradient Boosted tree

2. INTRODUCTION

Data Mining is process of analyzing the important information from a large set of data and come up with the prediction model. Data Mining is also called as Knowledge Discovery in Databases (KDD). The data mining plays an important role in all the fields like medical, airline, banking sector, movies, scientific information and numerous new data types. Data mining can be used to solve real time problems. Educational Data Mining (EDM) is the emerging technique for developing the prediction model with the help of available dataset, and extract the prediction of students' academic performance using machine learning technique. The prediction model acts like a warning system which is used to identify the weak students. EDM is a new field of research in data mining. The recent increase in online learning by the students have led to the progression in development of EDM. The highly reputed educational institution mainly focuses on improving the performance of the students in order to retain the standard rank of the institution, hence they train the students in such way that they perform well in academics and extra-curricular activities. The data mining is classified into:

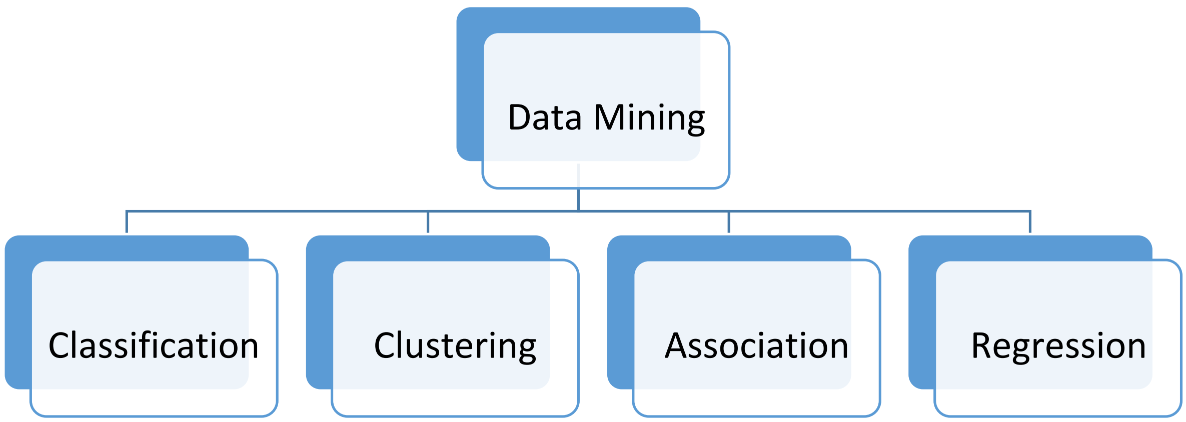


Figure 1: Data Mining Techniques

By using Educational data mining technique, the educational institutions can predict the performance of the students and identify low performing student early enough to overcome their difficulties in learning and improve their learning outcomes. Day by day the volume of the data is increased, hence there are different data mining algorithms which are used for predicting the performance of the students like supervised and unsupervised techniques to get the maximum accuracy. The supervised method is categorized into Classification or Categorization and Regression. The unsupervised method is categorized into Clustering and Association. Some of the algorithms which are popularly used in prediction are Decision tree, Multilayer Perceptron, Logistic Regression, Random forest, Gradient boosted trees, ID3 and J48. This study comprises of implementing different data mining techniques which are used in predicting the academic performance of the students. Two ensemble model is created, one model using Random Forest, Decision Tree and C5.0. Second model using Random Forest, Logistic Regression and Gradient Boosting. Major data mining techniques which are used for predicting the student's performance are shown below:

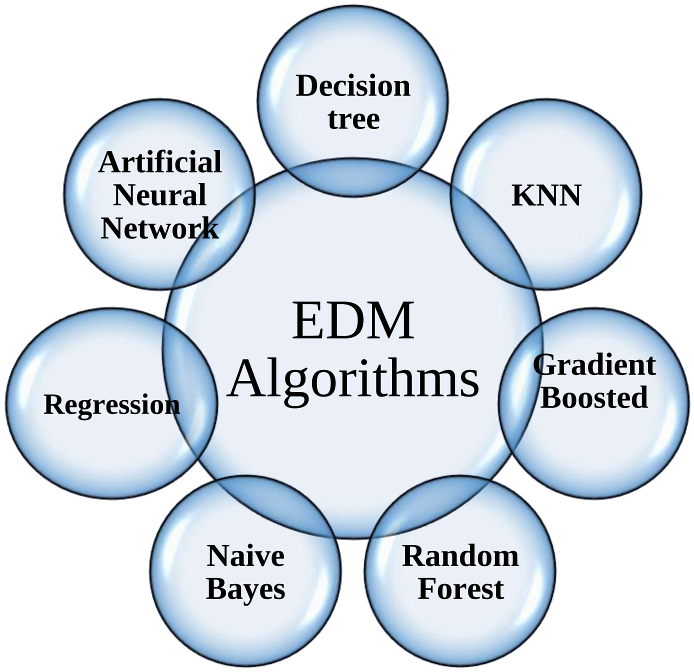


Figure 2: Data Mining Algorithm

RELATED WORK

Students' social activities and background details were used by Ching-Chieh Kiu [1] in the year 2018 for predicting the student's academic performance. This study used several data mining technique has been applied on predicting the accuracy of each dataset, of which decision tree J48 has given the best accuracy rate. The decision tree model has achieved 95% accuracy rate compared to other data mining technique. The dataset was divided into three subsets, students background details, student's social activities and student course work. The algorithms were implemented in WEKA tool. A comparative study on decision tree and random forest was done by Fergie Joanda and Reymon in the year 2018. The dataset was collected using questionaries' from the computer science students. It had 249 instances out of which 3 different classes of data are used. The study proves that decision tree gives more accurate result of 66.9% compared to random forest which gave 61.44%. The comparison was implemented using WEKA tool [2].

A comparative study was done on Naïve Bayes, Decision Tree, K-Nearest Neighbor and Discriminant Analysis by Samuel, Nor Bahiah and Siti Mariyam in the year 2019. The study was done to identify the best data mining technique for predicting the student's academic performance. It used 10 datasets from the University of California Irvine Repository. The decision tree out performed with the accuracy of 81.94% compared to other data mining techniques. The accuracy of other techniques was Naïve Bayes-73.61 %, KNN-80.56 % and Discriminant Analysis -77.78% accuracy rate. The tool used in the study was WEKA. In future hybrid metaheuristics algorithms will be for feature selection on the student data [3]. According to the study of Nongnuch Ketui, Warawut Wisomka and Kanitha Homjun in year 2019, Gradient boosted trees has given the best accuracy compared to other classification data mining techniques like Decision Tree, Weighted Decision Tree, Iterative Dichotomiser 3 (ID3) and Random Tree. WEKA is the data mining tool which is used for implementing the data mining techniques. A raw dataset was collected from the Rajamangala University of Technology Lanna Nan. The gradient boosted tree and decision tree gave good accuracy rate of 92.31% and 91.03% compared to other techniques. The Weighted Decision Tree 84.14%, ID3-89.66% and Random Tree-84.14% accuracy rate. The classification technique is widely used in predicting the student's performance [4].

According to Romero, Cristóbal, et al [5] SMO gives more accurate result compared to other techniques like BayesNet, Naïve Bayes Simple and EM. The paper used four different datasets and each dataset produced its own accuracy. The algorithms were implemented using WEKA tool. The SMO produced 82.4% accuracy result and BayesNet produced accuracy result of 81.5% accuracy. The Naïve Bayes Simple produced an accuracy of 82.4% and EM has 80.7% accuracy rate. All the algorithms performed equally good. In the year 2014 [6], Hu, Ya-Han, Chia-Lun Lo, and Sheng-Pao Shih made a study on predict students' online learning using C4.5, CART and LGR. The WEKA tool was used for implementation. The dataset used was learning portfolio data and C4.5 produced more accurate result of 93.4%. Yu, Liang-Chih, et al [7] used Sentiment analysis in order to predict the student's academic performance in the year 2018. The study used unstructured dataset. The WEKA tool was used for implementing the sentiment analysis which produced an 76% accuracy rate.

According to Deepika, K., and N. Sathvanaravana Support Vector Machine produces more accurate result compare to Linear Regression and Random forest. The study used student dataset of various academic disciplines of higher educational institutions in Kerala, India. The Linear Regression produced 89.96% accuracy and Random forest produced 89.98% and Support Vector Machine produced 91.43% accuracy result [8]. A comparative study was done in the year 2018 by Uzel and Vahide Nida on Multilayer Perceptron, Random Forest, Naïve Bayes, Decision Tree and Voting classifiers. The study used an educational dataset (xAPI) which is generated from an e-learning system includes 480 instances and 16 attributes. The Voting classifiers has highest accuracy of 80.6% [9]. The Artificial Neutral Network outperformed the decision tree and Naïve Bayes. The study used dataset from Kalboard 360 e–learning system with 500 instances and 17 attributes. The Decision Tree has

71.1% accuracy, Naïve Bayes has 67.5% accuracy and ANN has highest accuracy of 78.1% [10]. According to Rawat, Keshav Singh, and I. V. Malhan a hybrid classification gives more accurate result for predicting the student's academic performance. The study used data set of Department of Computer Science with 27 instances and 11 attributes. The Decision Tree produced 86.7% and KNN produced 87.5%, ANN produced 81.3% and NB produced 87.5%, Hybrid produced highest accuracy rate of 93.3% [11].

PAPER	EDM TECHNIQUE	TOOLS	DATASET	RESULT ACCURACY
[1] Kiu, Ching-Chieh. "Data Mining Analysis on Student's Academic Performance through Exploration of Student's Background and Social Activities." *2018 Fourth International Conference on Advances in Computing, Communication & Automation (ICACCA)*. IEEE, 2018.	1.Naïve Bayesian(NB) 2.Multilayer Perceptron 3. Decision Tree(DT) 4.J48 5. Random Forest	WEKA	Used 395 instances with 33 attributes that described performance in Mathematics subjects	DT-95% NB-76%
[2] Kaunang, Fergie Joanda, and Reymon Rotikan. "Students' Academic Performance Prediction using Data Mining." *2018 Third International Conference on Informatics and Computing (ICIC)*. IEEE, 2018.	1.Decision Tree(DT) 2. Random Forest(RF)	WEKA	Used 249 records with 3 different classes	DT -66.9% RF-61.14%
[3] Ajibade, Samuel-Soma M., Nor Bahiah Ahmad, and Siti Mariyam Shamsuddin. "An Heuristic Feature Selection Algorithm to Evaluate Academic Performance of Students." *2019 IEEE 10th Control and System Graduate Research Colloquium (ICSGRC)*. IEEE, 2019.	1.Naïve Bayes (NB) 2.Decision Tree (DT) 3. K-Nearest Neighbor (KNN) 4. Discriminant Analysis (DISC)	WEKA	Used 10 different datasets that are gotten from the University of California Irvine (UCI) Repository.	NB-73.61% DT-81.94 % KNN-80.56 % DISC-77.78%

PAPER	EDM TECHNIQUE	TOOLS	DATASET	RESULT ACCURACY
[4] Ketui, Nongnuch, Warawut Wisomka, and Kanitha Homjun. "Using Classification Data Mining Techniques for Students Performance Prediction." *2019 Joint International Conference on Digital Arts, Media and Technology with ECTI Northern Section Conference on Electrical, Electronics, Computer and Telecommunications Engineering (ECTI DAMT-NCON)*. IEEE.	1.Decision Tree 2.Weighted Decision Tree(WDT) 3.Iterative Dichotomiser 3 (ID3) 4.Random Tree 5.Gradient Boosted Trees	WEKA	Education Division of Rajamangala University of Technology Lanna Nan (RMUTL Nan) for gave the raw dataset	DT-91.03% WDT-84.14% ID3-89.66% RT-84.14% GBT-92.31%
[5] Romero, Cristóbal, et al. "Predicting students' final performance from participation in on-line discussion forums." *Computers & Education* 68 (2013): 458-472.	1.SMO 2.BayesNet 3.NaiveBayesSimple 4.EM	Meerkat ED SNAPP	Used four different student dataset	SMO -82.4% BayesNet - 81.5% NaiveBayes82 .4% EM -80.7%
[6] Hu, Ya-Han, Chia-Lun Lo, and Sheng-Pao Shih. "Developing early warning systems to predict students' online learning performance." *Computers in Human Behavior* 36 (2014): 469478.	1.C4.5 2. CART 3. LGR.	WEKA	Used learning portfolio data	C4.5-93.4% CART-76.9% LGR.-95%
[7] Yu, Liang-Chih, et al. "Improving early prediction of academic failure using sentiment analysis on self-evaluated comments." *Journal of Computer Assisted Learning* 34.4 (2018): 358-365.	1.Sentiment Analysis	WEKA	Used unstructured data	Sentiment Analysis-76%

PAPER	EDM TECHNIQUE	TOOLS	DATASET	RESULT ACCURACY
[8] Deepika, K., and N. Sathvanaravana. "Analyze and Predicting the Student Academic Performance Using Data Mining Tools." *2018 Second International Conference on Intelligent Computing and Control Systems (ICICCS)*. IEEE, 2018.	1.Linear Regression 2.Random forest 3.SVM	WEKA	Used student dataset of various academic disciplines of higher educational institutions in Kerala, India	LR-89.96% RF -89.98% SVM-91.43%
[9] Uzel, Vahide Nida, Sultan Sevgi Turgut, and Selma Ayşe Özel. "Prediction of Students' Academic Success Using Data Mining Methods." *2018 Innovations in Intelligent Systems and Applications Conference (ASYU)*. IEEE, 2018.	1.Multilayer Perceptron(MP) 2. Random Forest 3.NaïveBayes 4.Decision Tree 5.Voting classifiers(VC)	WEKA	Used an educational dataset which includes 480 instances and 16 attributes	MP -78.3 % RF-76.6% NB -67.7% DT -75.8% VC-80.6%
[10] Siddiqui, Isma Farrah, and Qasim Ali Arain. "ANALYZING STUDENTS'ACADEMIC PERFORMANCE THROUGH EDUCATIONAL DATA MINING." *3C Tecnologia* (2019).	1.Decision Tree 2.Naïve Bayes 3.Artificial Neural Network	WEKA	Used dataset from Kalboard 360 e–learning system with 500 instances and 17 attributes	DT-71.1% NB -67.5% ANN -78.1%
[11] Rawat, Keshav Singh, and I. V. Malhan. "A Hybrid Classification Method Based on Machine Learning Classifiers to Predict Performance in Educational Data Mining." *Proceedings of 2nd International Conference on Communication, Computing and Networking*. Springer, Singapore, 2019.	1.Decision tree 2.KNN 3.Artificial neural network 4.Naïve Bayes 5.Hybird	WEKA	Used data set of Department of Computer Science with 27 instances and 11 attributes	DT-86.7% KNN-87.5% ANN-81.3% NB-87.5% Hybird-93.3%

Decision tree has accuracy level from 66.9% to 95% and Random Forest has accuracy level from 61.14% to 89.98%. Naïve Bayes has accuracy level from 67.5% to 82.4% and KNN has accuracy level from 80.56% to 87.5%. ANN has accuracy level from 78.1% to 81.3% and Discriminant Analysis has accuracy level from77.78%. ID3 has accuracy level from 89.66% Weighted Decision Tree has accuracy from 84.14%. Gradient Boosted Trees has accuracy level of 92.31% and Sentiment Analysis has accuracy level 76%. Linear Regression has accuracy level 89.96% and Support Vector Machine has accuracy level 91.43%. Multilayer perceptron has 78.3% accuracy and Voting classifier has 80.6% accuracy. CART has 76.9% accuracy and LGR has 95% accuracy.

PROPOSED WORK

In this experiment, the dataset used consists of 650 instances with 33 attributes that describes the performance in Mathematics subjects. The attributes of the dataset are divided into four subsets:

1) Students background with 18 attributes
2) Student social activities with 12 attributes
3) Student coursework results with 2 attributes
4) Important values with 18 attributes

These subsets attributes will be used to predict final grade(G3). G3 is a Numeric datatype with range of 1 – 10 used to measure student performance on their final grade. The subset attributes will be evaluated under models: 2-level classification (Pass / Fail). Important attributes are selected using variable Importance function in order to enhance the accuracy of the algorithm. Ensembling is applied on the Important attributes. The algorithms which are used in study are listed below

DECISION TREE

It's one of the most powerful classification algorithm with tree like structure. Decision trees are a type of Supervised Machine Learning where the data is continuously split according to a certain parameter. A decision tree is a tree-shaped diagram used to determine a course of action. Each branch of the tree represents a possible decision, occurrence, or reaction. It has a root node, sub node and leaf node. The root node is the starting node of the tree followed by sub node which is used to make decisions and finally the leaf node which gives the end result of the classification.

RANDOM FOREST

Random forest is a supervised learning algorithm which is used for both classification as well as regression. As we know that a forest is made up of trees and more trees means more robust forest. Similarly, random forest algorithm creates decision trees on data samples and then gets the prediction from each of them and finally selects the best solution by means of voting. It is an ensemble method which is better than a single decision tree because it reduces the over-fitting by averaging the result.

LOGISTIC REGRESSION

Logistic regression is a statistical algorithm and it is mainly used for Binary classification problems (problems with two class values). Logistic regression is used to describe data and to explain the relationship between one dependent Binary variable and one or more Nominal, ordinal, interval or ratio-level independent variables.

GRADIENT BOOSTING

Gradient boosting is one of the most powerful techniques for building predictive models. The idea used in gradient boosting is a weak learner can be modified to become better. It uses a collection of decision tree which is built sequentially one after the other based on the result of the first tree the next tree performance is improved. Trees are constructed in a greedy manner, choosing the best split points based on purity scores like Gini or to minimize the loss

MULTILAYER PERCEPTRON

In the Multilayer perceptron, there can be more than one linear layer (combinations of neurons). If we take the simple example the three-layer network, first layer will be the input layer and last will be output layer and middle layer will be called hidden layer. The input data is provided into the input layer and take the output from the output layer. The number of the hidden layer can be increased as much as needed, to make the model more complex according to our task.

ENSEMBLE WITH TREE

Ensembling is a technique of combining two or more algorithms of similar or dissimilar types called base learners. This is done to make a more robust system which incorporates the predictions from all the base learners. Ensemble methods allows to produce better predictions compared to a single model. Popular technique used in Ensembling is Boosting and Bagging. This study uses two different Ensemble model. One ensemble model is created using Random Forest, logistic Regression and Gradient Boosting, other created using Decision tree, Random Forest and C5.0. Three types of concepts are used in Ensembling to combine the result which are listed below.

AVERAGING

It's defined as taking the average of predictions from models in case of regression problem or while predicting probabilities for the classification problem.

Model1	Model2	Model3	AveragePrediction
45	40	65	50

MAJORITY VOTE

It's defined as taking the prediction with maximum vote / recommendation from multiple models predictions while predicting the outcomes of a classification problem.

Model1	Model2	Model3	VotingPrediction
1	0	1	1

WEIGHTED AVERAGE

Different weights are applied to predictions from multiple models then taking the average which means giving high or low importance to specific model output.

	Model1	Model2	Model3	WeightAveragePrediction
Weight	0.4	0.3	0.3	
Prediction	45	40	60	48

3. SYSTEM ANALYSIS

ABOUT THE TOOL

R is a language and environment for statistical computing and graphics. R is a programming language developed by Ross Ihaka and Robert Gentleman in 1993. R possesses an extensive catalog of statistical and graphical methods. It includes machine learning algorithm, linear regression, time series, statistical inference to name a few. Most of the R libraries are written in R, but for heavy computational task, C, C++ and Fortran codes are preferred. R is an integrated suite of software facilities for data manipulation, calculation and graphical display. It includes

- An effective data handling and storage facility
- A suite of operators for calculations on arrays, in particular matrices,
- A large, coherent, integrated collection of intermediate tools for data analysis
- Graphical facilities for data analysis and display either on-screen or on hardcopy
- A well-developed, simple and effective programming language which includes conditionals, loops, user-defined recursive functions and input and output facilities.

Many users think of R as a statistics system. We prefer to think of it as an environment within which statistical techniques are implemented. R can be extended (easily) via packages. There are about eight packages supplied with the R distribution and many more are available through the CRAN family of Internet sites covering a very wide range of modern statistics. R is used for Statistical inference, Data analysis and Machine learning algorithm.Data science is shaping the way companies run their businesses. Without a doubt, staying away from Artificial Intelligence and Machine will lead the company to fail. The big question is which tool/language should you use?They are plenty of tools available in the market to perform data analysis. Learning a new language requires some time investment. The picture below depicts the learning curve compared to the business capability a language offers. The negative relationship implies that there is no free lunch. If you want to give the best insight from the data, then you need to spend some time learning the appropriate tool, which is R.On the top left of the graph, you can see Excel and PowerBI. These two tools are simple to learn but don't offer outstanding business capability, especially in term of modeling. In the middle, you can see Python and SAS. SAS is a dedicated tool to run a statistical analysis for business, but it is not free. SAS is a click and run software.

REQUIREMENT SPECIFICATION

HARWARE

The windows64 bit operating system with x64 based processor and 4GB RAM

SOFTWARE

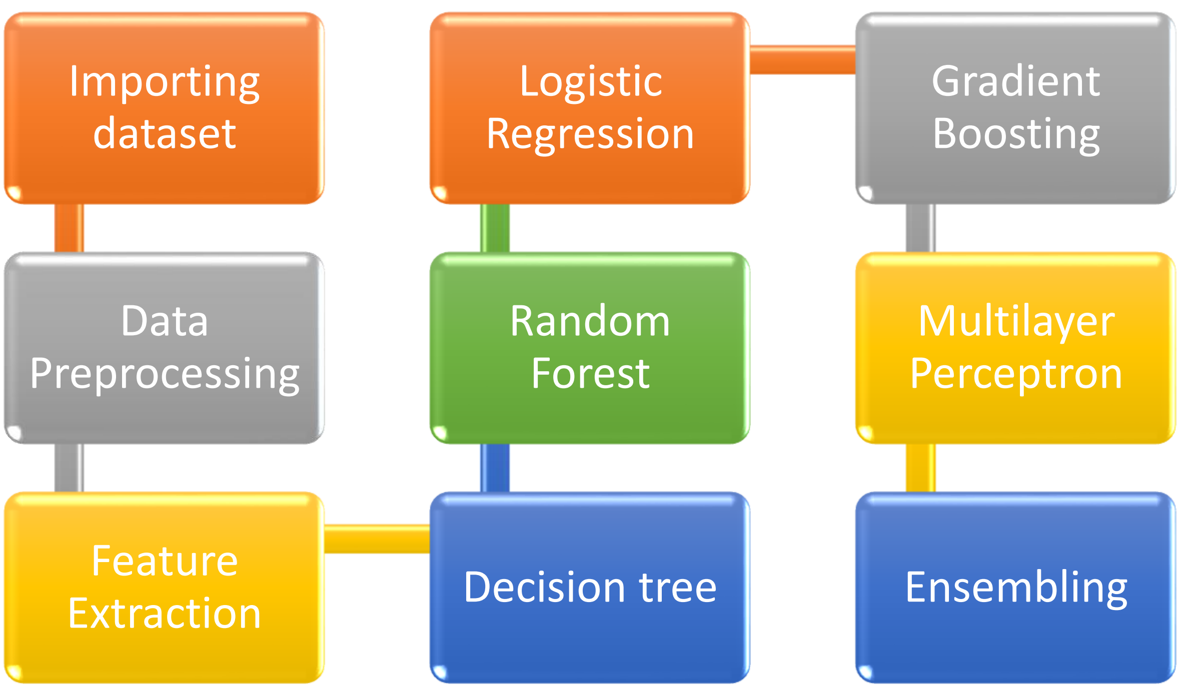
R studio is the software tool which is used for implementing the machine learning algorithms. R is an open source software which has lots of statistical packages installed. It also has different packages for implementing specific algorithms.

DATASET

The dataset used in this study is taken from the University of Minho, Mathematics department which consists of 33 attributes and 650 observations. This dataset includes G1, G2, G3 major attributes and G3 containing the average grade of G1 and G2.

4.SYSTEM DESIGN

TAXONOMY OF THE SYSTEM



DATABASE DESIGN

STUDENT BACKGROUND			
Attribute	**Description**	**Type**	**Value**
sex	gender of student	Binary	male | female
school	school of student		Mousinho da Silveira | Gabriel Pereira
address	type of student’s home address		rural | urban
Pstatus	cohabitation status of parent		living together | apart
famsize	size of family		≤ 3 | > 3
schoolsup	extra educational school support		yes | no
famsup	educational support from family		yes | no
Mjob	job of mother	Nominal	- at home - civil services - teacher - health care related - other
Fjob	job of father		
reason	reason to choose this school		- close to home - school reputation - course preference - other
guardian	guardian of student		- father |mother |other
Medu	education of mother	Numeric	0# none 1# primary education 2# 5th to 9th grade 3# secondary education 4# higher education
Fedu	education of father		
famrel	quality of family relationships		very bad (1) to excellent (5)
age	age of student		15 - 22
traveltime	travel time from home to school		< 15 min 15 to 30 min 30 min. to 1 hour > 1 hour
studytime	weekly study time		< 2 hours 2 to 5 hours 5 to 10 hours > 10 hours
failures	number of failures in past class		n if 1 ≤ n < 3, else 4

STUDENT SOCIAL ACTIVITIES			
Attribute	**Description**	**Type**	**Value**
activities	extra-curricular		
higher	plans for higher education		
internet	home internet access		
nursery	nursery school attended		
paidclass	extra paid classes		
romantic	in romantic relationship	Binary	yes | no
absences	absences from school		
health	status of current health		
freetime	free time after school	Numeric	very low (1) to very high (5)
goout	outing with friends		
Dalc	consume alcohol in weekday		
Walc	consume alcohol in weekend		0 - 93

STUDENT COURSEWORK RESULT			
Attribute	**Description**	**Type**	**Value**
GI	1st grade period	Numeric	0 - 20
G2	2nd grade period		

IMPORTANT ATTRIBUTES			
Attribute	**Description**	**Type**	**Value**
failures	number of failures in past class	Numeric	n if 1 ≤ n < 3, else 4
studytime	weekly study time	Numeric	1# < 2 hours 2# 2 to 5 hours 3# 5 to 10 hours 4# > 10 hours
G2	2nd grade period	Numeric	0 – 20

Attribute	Description	Type	Value
absences	absences from school	Numeric	very low (1) to very high (5)
goout	outing with friends	Numeric	very low (1) to very high (5)
Wal–c	consume alcohol in weekend	Numeric	0 – 5
Dalc	consume alcohol in weekday	Numeric	0 – 5
traveltime	travel time from home to school	Numeric	0# < 15 min 1# 15 to 30 min 2# 30 min. to 1 hour 3# > 1 hour
famrel	quality of family relationships	Numeric	very bad (1) to excellent (5)
Fedu	education of father	Numeric	0# none 1# primary education 2# 5th to 9th grade 3# secondary education 4# higher education
reason	reason to choose this school	Nominal	- close to home - school reputation - course preference - other
guardian	guardian of student	Nominal	- father |mother |other
Fjob	job of father	Nominal	- at home - civil services - teacher - health care related - other
Mjob	job of mother	Nominal	
higher	plans for higher education	Binary	yes | no
internet	home internet access	Binary	yes | no
paidclass	extra paid classes	Binary	yes | no

SCREEN DESIGN

➢ **The implementation of Decision tree is given below**

````
data_test <- create_train_test(stud, 0.8, train = FALSE)
dim(data_train)
dim(data_test)


prop.table(table(data_train$class))

#STEP4: Build the model

tree<-rpart(class~sex+school+Pstatus+failures+studytime+Medu+Fedu+traveltime+Fjob+Mjob+
famrel+reason+Pstatus+famsize+schoolsup+famsup+age+guardian, data=data_train,
method='class')
rpart.plot(tree,extra=106,cex=.7,roundint = FALSE)

#STEP5: Make a prediction
pred<-predict(tree, data_test, type = 'class')

table_mat <- table(data_test$class, pred)
table_mat

#STEP6: Measure performance
#ACCURACY=TP+TN/TP+TN+FP+FN

accuracy_Test <- sum(diag(table_mat)) / sum(table_mat)
print(paste('Accuracy for train', accuracy_Test))


```
````

➢ **The implementation of Random Forest is given below**

```
#STEP3: SPLIT THE DATASET
student$class<-as.factor(student$class)
set.seed(200)
data1_set_size= floor(nrow(student)*0.80)
index<-sample(1:nrow(student),size=data1_set_size)

training<-student[index,]
testing<-student[-index,]



#STEP4: Build the model
rf<-randomForest(class~sex+school+address+Pstatus+failures+studytime+Medu+Fedu+travelti
me+Fjob+Mjob+famrel+reason+Pstatus+famsize+schoolsup+famsup+age+guardian,
data=training, mtry=10, ntree=300, importance=TRUE)
rf
plot(rf)
result<-data.frame(testing$class, predict(rf, testing[,1:32], type="response"))
result
plot(result)


#ACCURACY OF THE MODEL
prediction<-predict(rf,testing,type="class")

ConfusionMatric<-table(prediction,testing$class)
ConfusionMatric
```
```

- **The implementation of Multilayer Perceptron is given below**

```r
stud <- stud[sample(1:nrow(stud),length(1:nrow(stud))),1:ncol(stud)]
StudentValues <- stud[,1:18]
StudentTargets <- decodeClassLabels(stud[,19])
stud <- splitForTrainingAndTest(StudentValues, StudentTargets, ratio=0.20)
na.omit(stud$inputsTrain)
na.omit(stud$targetsTrain)

#STEP4:BUILT THE MULTILAYER PERCEPTRON MODEL

model <- mlp(
  stud$inputsTrain, stud$targetsTrain, size=10, learnFuncParams=c(0.1),
             maxit=50, inputsTest=stud$inputsTest, targetsTest=stud$targetsTest)

model
weightMatrix(model)
extractNetInfo(model)
par(mfrow=c(2,2))
plotIterativeError(model)

predictions <- predict(model,stud$inputsTest)

plotRegressionError(predictions[,2], stud$targetsTest[,2])


confusionMatrix(stud$targetsTrain,fitted.values(model))
table_mat<-confusionMatrix(stud$targetsTest,predictions)
table_mat
accuracy_Test <- sum(diag(table_mat)) / sum(table_mat)
print(paste('Accuracy for test', accuracy_Test))
```

- **The implementation of Gradient Boosting is given below**

```r
    return (data[-train_sample, ])
  }
}

data_train <- create_train_test(stud, 0.8, train = TRUE)
data_test <- create_train_test(stud, 0.8, train = FALSE)
dim(data_train)
dim(data_test)
prop.table(table(data_train$class))



model <- train(class~sex+school+address+Pstatus+failures+studytime+Medu+Fedu+traveltime
+Fjob+Mjob+famrel+reason+Pstatus+famsize+schoolsup+famsup+age+guardian, data =
data_train, method ="xgbTree", trControl = trainControl("cv", number=10))

model$bestTune

# Make predictions on the test data
predicted.classes <-predict(model,data_test)
head(predicted.classes)

# Compute model prediction accuracy rate
mean(predicted.classes == data_test$class)
```

- **The implementation of Logistic Regression is given below**

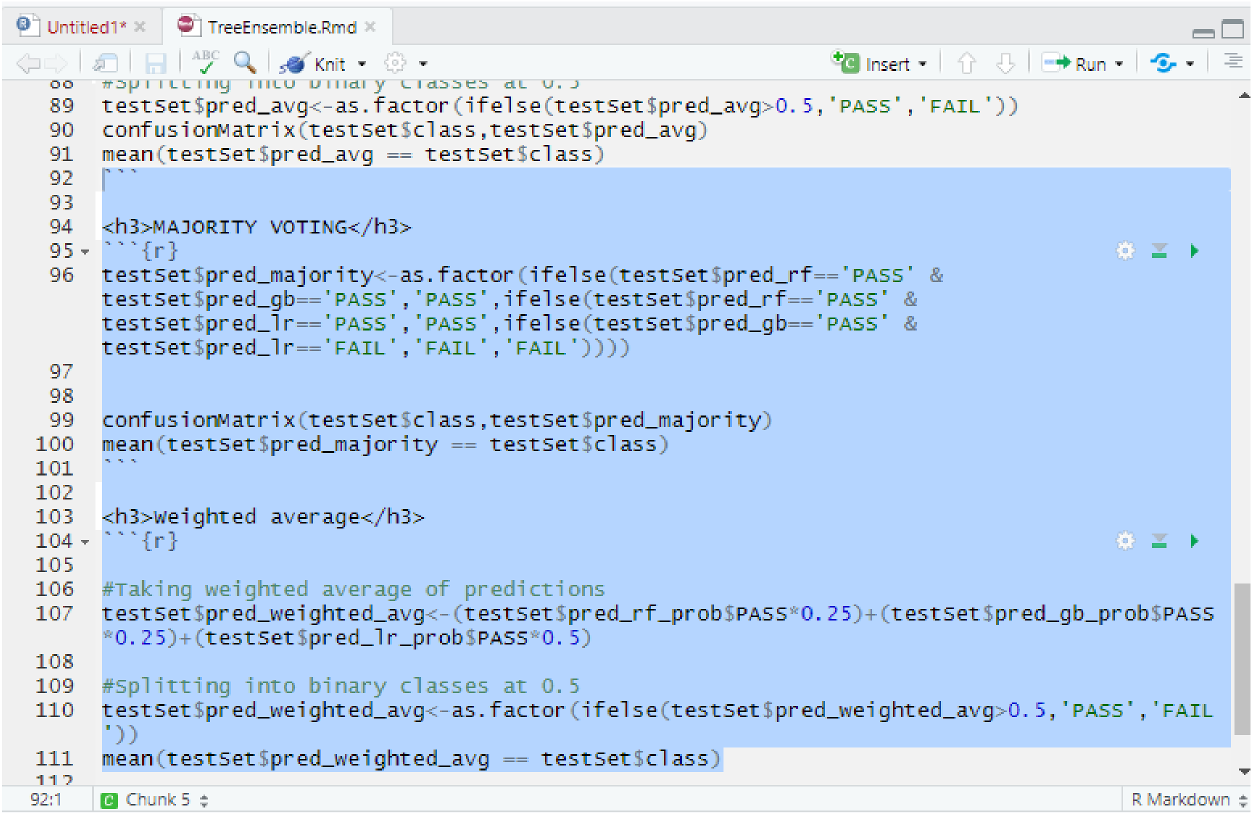
```r
test_pass <- pass[-pass_train, ]
test_fail <- fail[-fail_train, ]
test<- rbind(test_pass, test_fail)  # row bind the pass and fail

table(test$class)
table(train$class)

mymodel<-glm(class~sex+school+address+Pstatus+failures+studytime+Medu+Fedu+traveltime+Fjob+Mjob+famrel+reason+Pstatus+famsize+schoolsup+famsup+age+guardian, family='binomial', data=train,maxit=100)

mymodel
summary(mymodel)

restest<-predict(mymodel,test,type="response")

ROCRPred<-prediction(restest,test$class)
ROCRPref<-performance(ROCRPred,"tpr","fpr")

plot(ROCRPref,colorize=TRUE,print.cutoffs.at=seq(0.1, by=0.1))

confmatrix<-table(Actual_Value=test$class,Predicted_Value=restest>=0.5)
confmatrix
accuracy<-sum(diag(confmatrix))/sum(confmatrix)
print(paste("Accuracy of the test",accuracy))

```

- **Implementing Ensemble**

```r

```
<h3>AVERAGING</h3>
```{r}
#Predicting the probabilities
testSet$pred_rf_prob<-predict(object = model_rf,testSet[,predictors],type='prob')
testSet$pred_gb_prob<-predict(object = model_gb,testSet[,predictors],type='prob')
testSet$pred_lr_prob<-predict(object = model_lr,testSet[,predictors],type='prob')

#Taking average of predictions
testSet$pred_avg<-(testSet$pred_rf_prob$PASS+testSet$pred_gb_prob$PASS+testSet$pred_lr_prob$PASS)/3

#Splitting into binary classes at 0.5
testSet$pred_avg<-as.factor(ifelse(testSet$pred_avg>0.5,'PASS','FAIL'))

confusionMatrix(testSet$class,testSet$pred_avg)
mean(testSet$pred_avg == testSet$class)


```

<h3>MAJORITY VOTING</h3>
```{r}
testSet$pred_majority<-as.factor(ifelse(testSet$pred_rf=='PASS' & testSet$pred_gb=='PASS','PASS',ifelse(testSet$pred_rf=='PASS' & testSet$pred_lr=='PASS','PASS',ifelse(testSet$pred_gb=='PASS' & testSet$pred_lr=='FAIL','FAIL','FAIL'))))
```

➢ **Implementing Ensemble with Tree algorithms**




```
#Splitting into binary classes at 0.5
testSet$pred_avg<-as.factor(ifelse(testSet$pred_avg>0.5,'PASS','FAIL'))
confusionMatrix(testSet$class,testSet$pred_avg)
mean(testSet$pred_avg == testSet$class)
```


<h3>MAJORITY VOTING</h3>
```{r}
testSet$pred_majority<-as.factor(ifelse(testSet$pred_rf=='PASS' & testSet$pred_gb=='PASS','PASS',ifelse(testSet$pred_rf=='PASS' & testSet$pred_lr=='PASS','PASS',ifelse(testSet$pred_gb=='PASS' & testSet$pred_lr=='FAIL','FAIL','FAIL'))))


confusionMatrix(testSet$class,testSet$pred_majority)
mean(testSet$pred_majority == testSet$class)
```

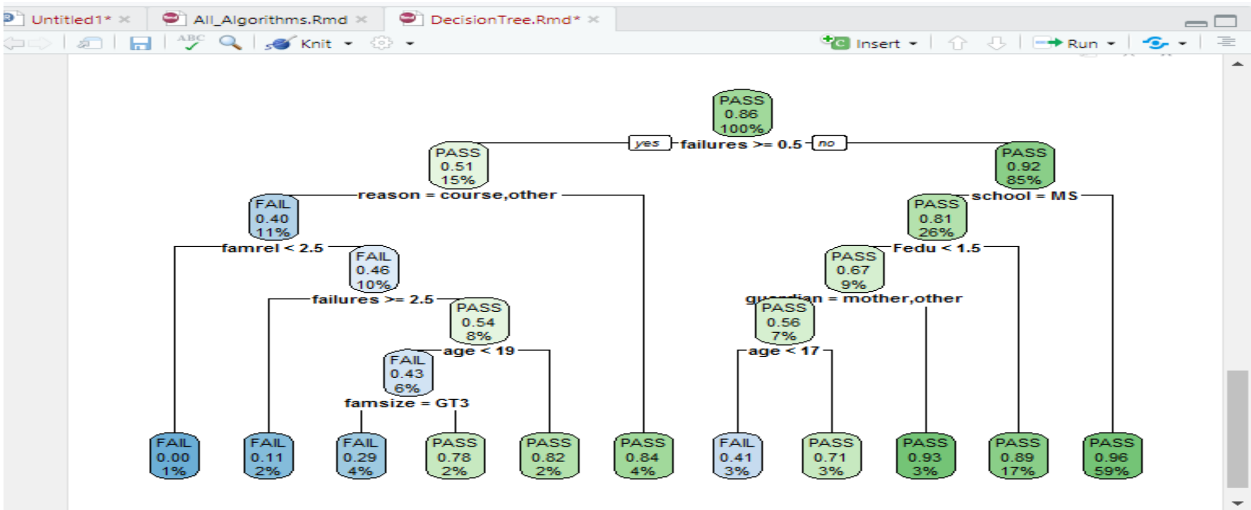
<h3>Weighted average</h3>
```{r}

#Taking weighted average of predictions
testSet$pred_weighted_avg<-(testSet$pred_rf_prob$PASS*0.25)+(testSet$pred_gb_prob$PASS*0.25)+(testSet$pred_lr_prob$PASS*0.5)

#Splitting into binary classes at 0.5
testSet$pred_weighted_avg<-as.factor(ifelse(testSet$pred_weighted_avg>0.5,'PASS','FAIL'))
mean(testSet$pred_weighted_avg == testSet$class)
```

REPORT

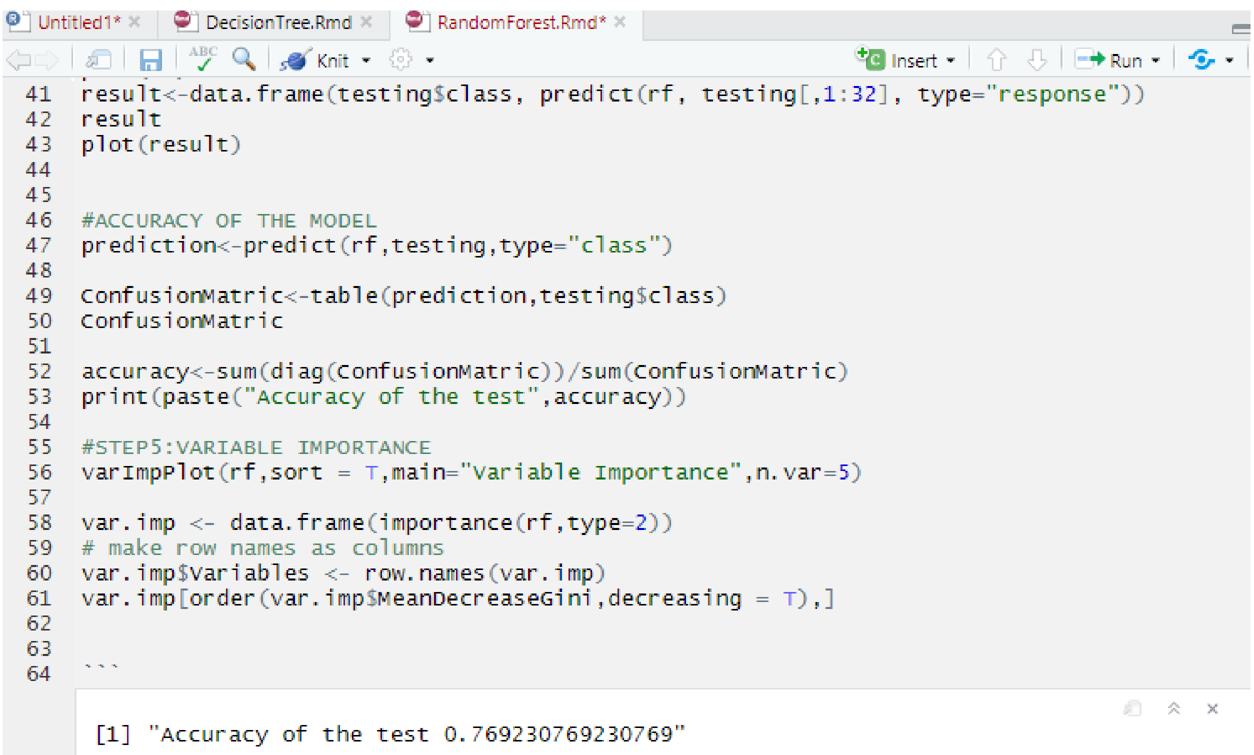
The implementation of decision tree with background details gave 73% accuracy, with social activities attributes 78% accuracy, with course work attributes 93% accuracy and with important attributes it gave 93% accuracy. The implementation of Random forest with background details gave 76% accuracy, with social activities attributes 78% accuracy, with course work attributes 91% accuracy and with important attributes it gave 91% accuracy. The implementation of Logistic Regression with background details gave 87% accuracy, with social activities attributes 83% accuracy, with course work attributes 93% accuracy and with important attributes it gave 92% accuracy. The implementation of Gradient Boosting with background details gave 79% accuracy, with social activities attributes 74% accuracy, with course work attributes 93% accuracy and with important attributes it gave 92% accuracy. The implementation of Logistic Regression with background details gave 83% accuracy, with social activities attributes 83% accuracy, with course work attributes 90% accuracy and with important attributes it gave 93% accuracy.

EXPERIMENTAL RESULTS

- **Output of the decision tree**



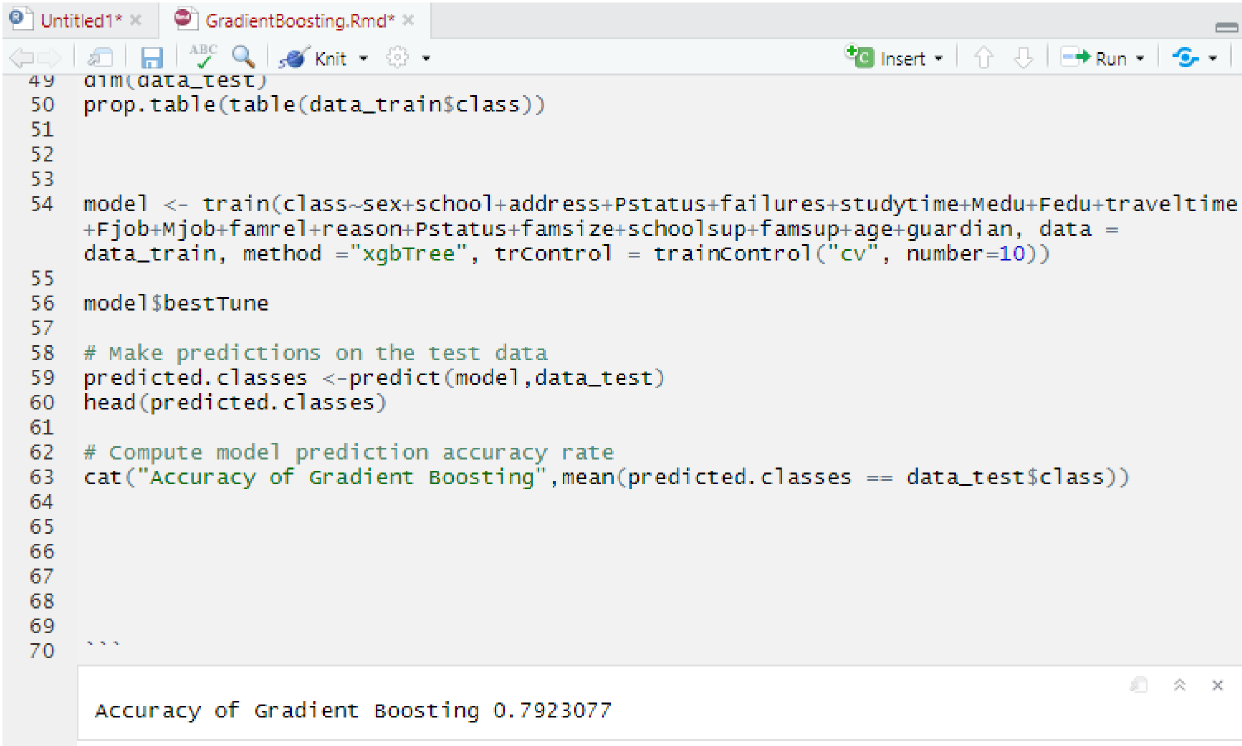
- **Output of Random forest**



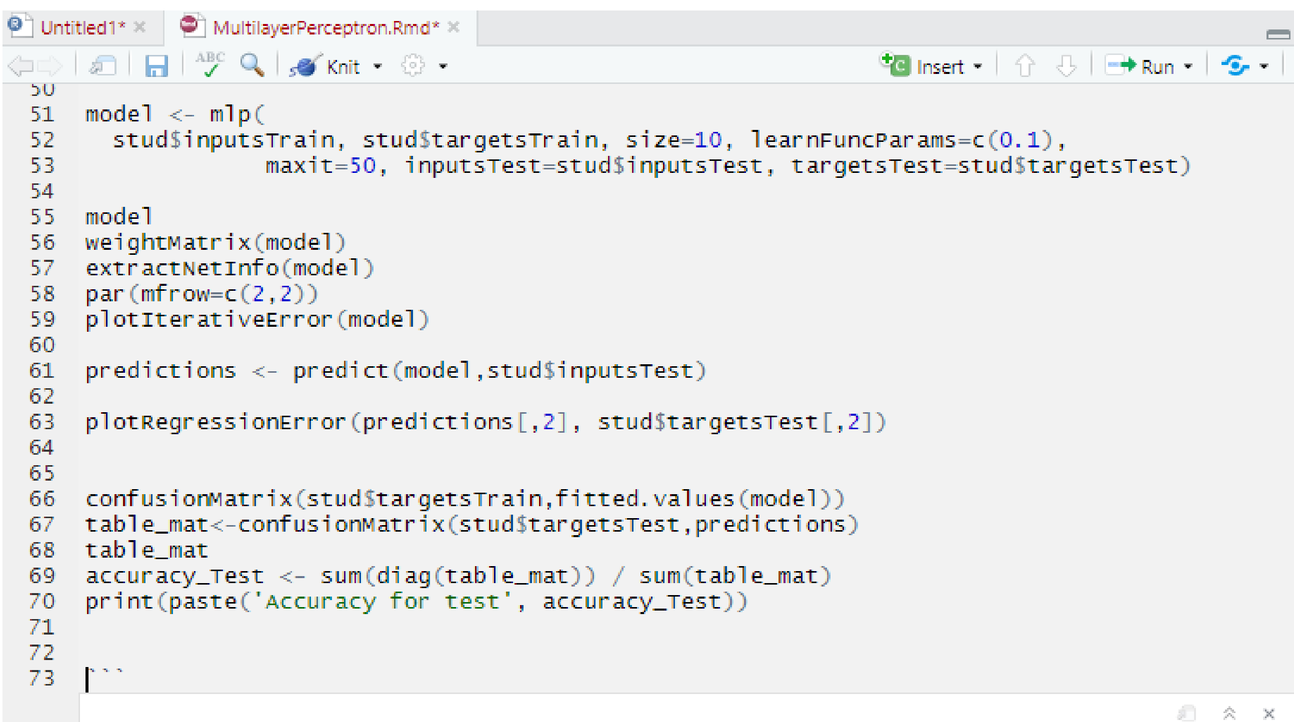

```
result<-data.frame(testing$class, predict(rf, testing[,1:32], type="response"))
result
plot(result)


#ACCURACY OF THE MODEL
prediction<-predict(rf,testing,type="class")

ConfusionMatric<-table(prediction,testing$class)
ConfusionMatric

accuracy<-sum(diag(ConfusionMatric))/sum(ConfusionMatric)
print(paste("Accuracy of the test",accuracy))

#STEP5:VARIABLE IMPORTANCE
varImpPlot(rf,sort = T,main="Variable Importance",n.var=5)

var.imp <- data.frame(importance(rf,type=2))
# make row names as columns
var.imp$Variables <- row.names(var.imp)
var.imp[order(var.imp$MeanDecreaseGini,decreasing = T),]


```

[1] "Accuracy of the test 0.769230769230769"
```
```

➢ **Output of Gradient Boosting**



➢ **Output of Multilayer Perceptron**



- **Output of Logistic Regression**

```r
table(test$class)
table(train$class)

mymodel<-glm(class~sex+school+address+Pstatus+failures+studytime+Medu+Fedu+traveltime+Fjob+Mjob+famrel+reason+Pstatus+famsize+schoolsup+famsup+age+guardian,
family='binomial', data=train,maxit=100)

mymodel
summary(mymodel)

restest<-predict(mymodel,test,type="response")

ROCRPred<-prediction(restest,test$class)
ROCRPref<-performance(ROCRPred,"tpr","fpr")

plot(ROCRPref,colorize=TRUE,print.cutoffs.at=seq(0.1, by=0.1))

confmatrix<-table(Actual_Value=test$class,Predicted_Value=restest>=0.5)
confmatrix
accuracy<-sum(diag(confmatrix))/sum(confmatrix)
print(paste("Accuracy of the test",accuracy))
```

```
[1] "Accuracy of the test 0.871794871794872"
```

- **Output of Ensemble**

MAJORITY VOTING

```r
testSet$pred_majority<-as.factor(ifelse(testSet$pred_rf=='PASS' & testSet$pred_gb=='PASS','PASS',ifelse(testSet$pred_rf=='PASS' & testSet$pred_lr=='PASS','PASS',ifelse(testSet$pred_gb=='PASS' & testSet$pred_lr=='FAIL','FAIL','FAIL'))))


confusionMatrix(testSet$class,testSet$pred_majority)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction FAIL PASS
##       FAIL   14    6
##       PASS    0  109
##
##                Accuracy : 0.9535
##                  95% CI : (0.9015, 0.9827)
##     No Information Rate : 0.8915
##     P-Value [Acc > NIR] : 0.01065
##
##                   Kappa : 0.7977
##
##  Mcnemar's Test P-Value : 0.04123
##
##             Sensitivity : 1.0000
##             Specificity : 0.9478
##          Pos Pred Value : 0.7000
##          Neg Pred Value : 1.0000
##              Prevalence : 0.1085
##          Detection Rate : 0.1085
##    Detection Prevalence : 0.1550
##       Balanced Accuracy : 0.9739
##
##        'Positive' Class : FAIL
##
```

➢ **Output of Tree Ensemble**

MAJORITY VOTING

```
testSet$pred_majority<-as.factor(ifelse(testSet$pred_rf=='PASS' & testSet$pred_gb=='PASS','PASS',ifelse(testSet$pred_rf=='PASS' & testSet$pred_lr=='PASS','PASS',ifelse(testSet$pred_gb=='PASS' & testSet$pred_lr=='FAIL','FAIL','FAIL'))))


confusionMatrix(testSet$class,testSet$pred_majority)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction FAIL PASS
##       FAIL   17    3
##       PASS    2  107
##
##                Accuracy : 0.9612
##                  95% CI : (0.9119, 0.9873)
##     No Information Rate : 0.8527
##     P-Value [Acc > NIR] : 6.419e-05
##
##                   Kappa : 0.849
##
##  Mcnemar's Test P-Value : 1
##
##             Sensitivity : 0.8947
##             Specificity : 0.9727
##          Pos Pred Value : 0.8500
##          Neg Pred Value : 0.9817
##              Prevalence : 0.1473
##          Detection Rate : 0.1318
##    Detection Prevalence : 0.1550
##       Balanced Accuracy : 0.9337
##
##        'Positive' Class : FAIL
```

5.IMPLEMENTATION

The dataset used in this study is taken from the University of Minho, Mathematics department which consists of 33 attributes and 650 observations. This dataset includes G1, G2, G3 major attributes and G3 containing the average grade of G1 and G2. The response attribute class is created using G3 attribute. This class attribute contains only the pass and fail values, the students who got more than 10 in G3 they are placed into pass class and students who got less than 10 in G3 are placed in fail category. The dataset is divided into three groups: Background details, Social activities and Course work. The Background details contain Sex, School, Address, Pstatus, famsize, schoolsup, famup, Mjob, Fjob, Fedu, Medu, Reason, Guardian, Famrel, Age, Travel time, Study time and Failures. The Social activities contain Activities, Higher, Internet, Nursery, Paid Class, Romantic, Absences, Health, Free time, Gout, Dalc and Walc. The course work contains G1 and G2 attribute. From the 33 attributes 14 attributes are selected as important attributes by using the Variable Importance function in order to improve the accuracy of the algorithms.

SCREENSHOT

Untitled1* × | logesticRegression.Rmd × | dataset ×

Filter

	failures	studytime	famrel	traveltime	guardian	higher	absences	goout	Dalc	Walc	internet	paid	Fedu	Mjob	Fjob	reason	G2
1	0	2	4	2	mother	yes	4	4	1	1	no	no	4	at_home	teacher	course	11
2	0	2	5	1	father	yes	2	3	1	1	yes	no	1	at_home	other	course	11
3	0	2	4	1	mother	yes	6	2	2	3	yes	no	1	at_home	other	other	13
4	0	3	3	1	mother	yes	0	2	1	1	yes	no	2	health	services	home	14
5	0	2	4	1	father	yes	0	2	1	2	no	no	3	other	other	home	13
6	0	2	5	1	mother	yes	6	2	1	2	yes	no	3	services	other	reputation	12
7	0	2	4	1	mother	yes	0	4	1	1	yes	no	2	other	other	home	12
8	0	2	4	2	mother	yes	2	4	1	1	no	no	4	other	teacher	home	13
9	0	2	4	1	mother	yes	0	2	1	1	yes	no	2	services	other	home	16
10	0	2	5	1	mother	yes	0	1	1	1	yes	no	4	other	other	home	12
11	0	2	3	1	mother	yes	2	3	1	2	yes	no	4	teacher	health	reputation	14
12	0	3	5	3	father	yes	0	2	1	1	yes	no	1	services	other	reputation	12
13	0	1	4	1	father	yes	0	3	1	3	yes	no	4	health	services	course	13
14	0	2	5	2	mother	yes	0	3	1	2	yes	no	3	teacher	other	course	12
15	0	3	4	1	other	yes	0	2	1	1	yes	no	2	other	other	home	14
16	0	1	4	1	mother	yes	6	4	1	2	yes	no	4	health	other	home	17

Showing 1 to 17 of 649 entries, 18 total columns

```r

test_pass <- pass[-pass_train, ]
test_fail <- fail[-fail_train, ]
test<- rbind(test_pass, test_fail)  # row bind the pass and fail

table(test$class)
table(train$class)

mymodel<-glm(class~sex+school+address+Pstatus+failures+studytime+Medu+Fedu+traveltime+Fjob+Mjob+famrel+reason+Pstatus+famsize+schoolsup+famsup+age+guardian, family='binomial', data=train,maxit=100)

mymodel
summary(mymodel)

restest<-predict(mymodel,test,type="response")

ROCRPred<-prediction(restest,test$class)
ROCRPref<-performance(ROCRPred,"tpr","fpr")

plot(ROCRPref,colorize=TRUE,print.cutoffs.at=seq(0.1, by=0.1))

confmatrix<-table(Actual_Value=test$class,Predicted_Value=restest>=0.5)
confmatrix
accuracy<-sum(diag(confmatrix))/sum(confmatrix)
print(paste("Accuracy of the test",accuracy))

```
```
```

SAMPLE CODE

```
chisq.test(class,Fjob)

chisq.test(class,Mjob)

chisq.test(class,paid)

chisq.test(class,guardian)

cor(Fedu,G3)

cor(traveltime,G3)

cor(absences,G3)

cor(Walc,G3)

cor(Dalc,G3)

create_train_test <- function(data, size = 0.8, train = TRUE) {

  n_row = nrow(data)

  total_row = size * n_row

  train_sample  <- 1: total_row

  if (train == TRUE) {

    return (data[train_sample, ])

  } else {

    return (data[-train_sample, ])

  }

}


data_train <- create_train_test(stud, 0.8, train = TRUE)

data_test <- create_train_test(stud, 0.8, train = FALSE)


rf<-randomForest(class~failures+studytime+G2+higher+absences+goout+Walc+famre
l+reason+Fedu+internet+Fjob,data=data_train, mtry=12, ntree=500, importance=T
RUE)

rf


mymodel<-glm(class~failures+studytime+G2+higher+absences+goout+Walc+famrel+re
ason+Fedu+internet+Fjob, family='binomial', data=data_train,maxit=100)

model <- mlp(stud$inputsTrain, stud$targetsTrain, size=5, learnFuncParams=c(0
.1),

  maxit=50, inputsTest=stud$inputsTest, targetsTest=stud$targetsTest)
```

6.CODE REVIEW & TESTING

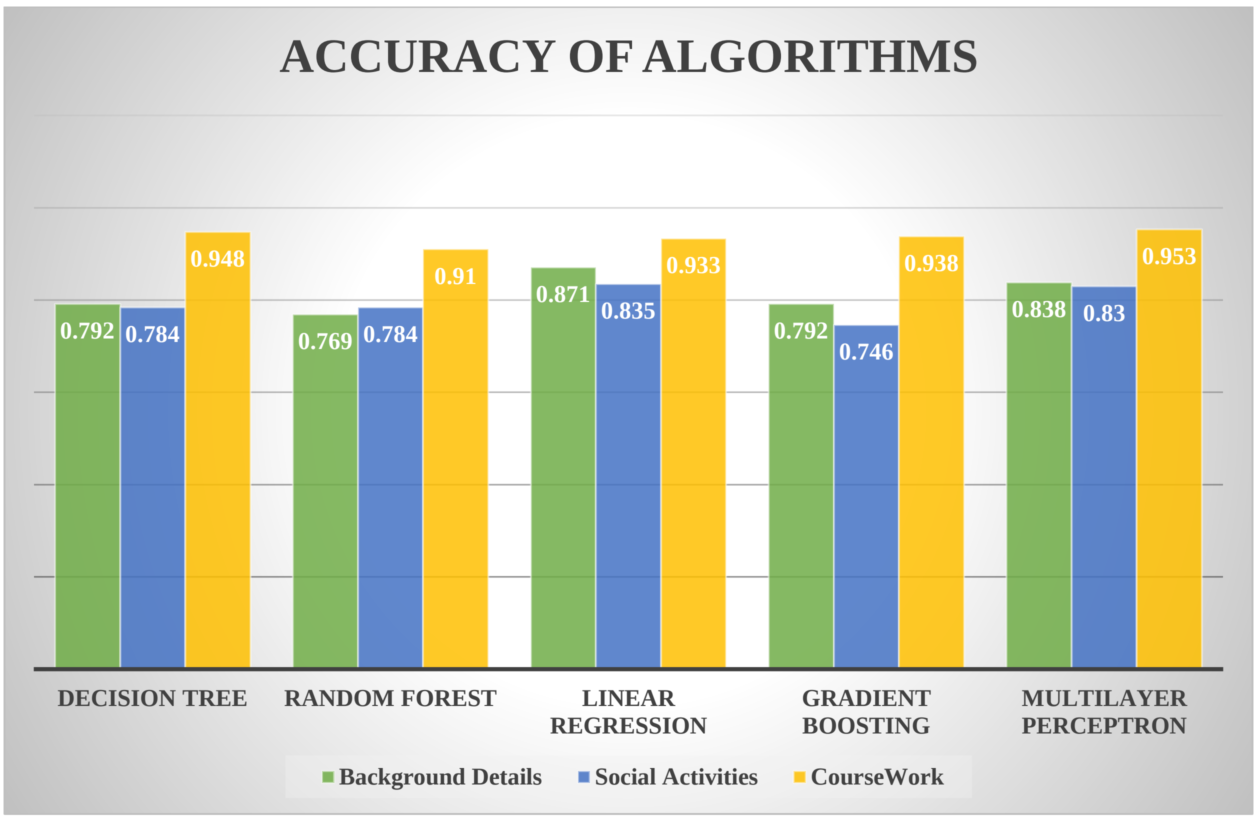
Individual Algorithms Accuracy				
S.no	**Algorithm**	**Background Details**	**Social Activities**	**Course**
1	Decision Tree	0.792	0.784	0.93
2	Random Forest	0.769	0.784	0.91
3	Linear Regression	0.871	0.835	0.933
4	Gradient Boosting	0.792	0.746	0.938
5	Multilayer Perceptron	0.838	0.83	0.9

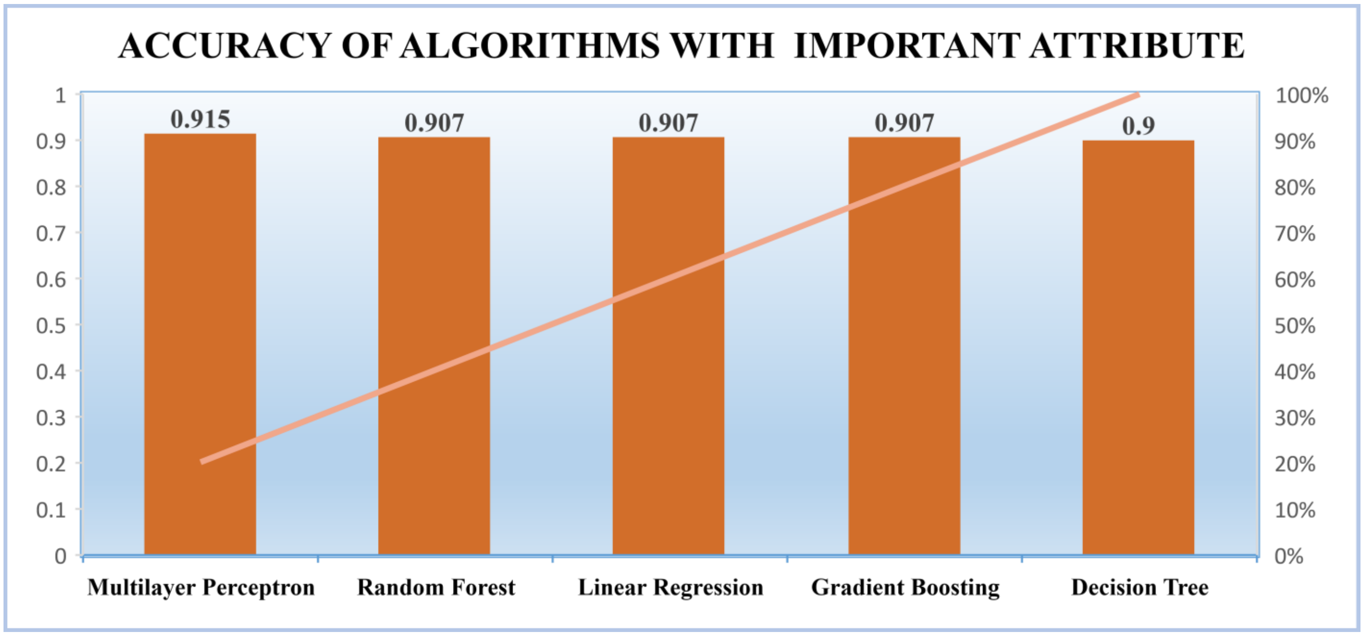
Individual Algorithms Accuracy with Important Attributes		
S.no	**Algorithm**	**Accuracy**
1	Decision Tree	0.93
2	Random Forest	0.915
3	Linear Regression	0.923
4	Gradient Boosting	0.923
5	Multilayer Perceptron	0.93

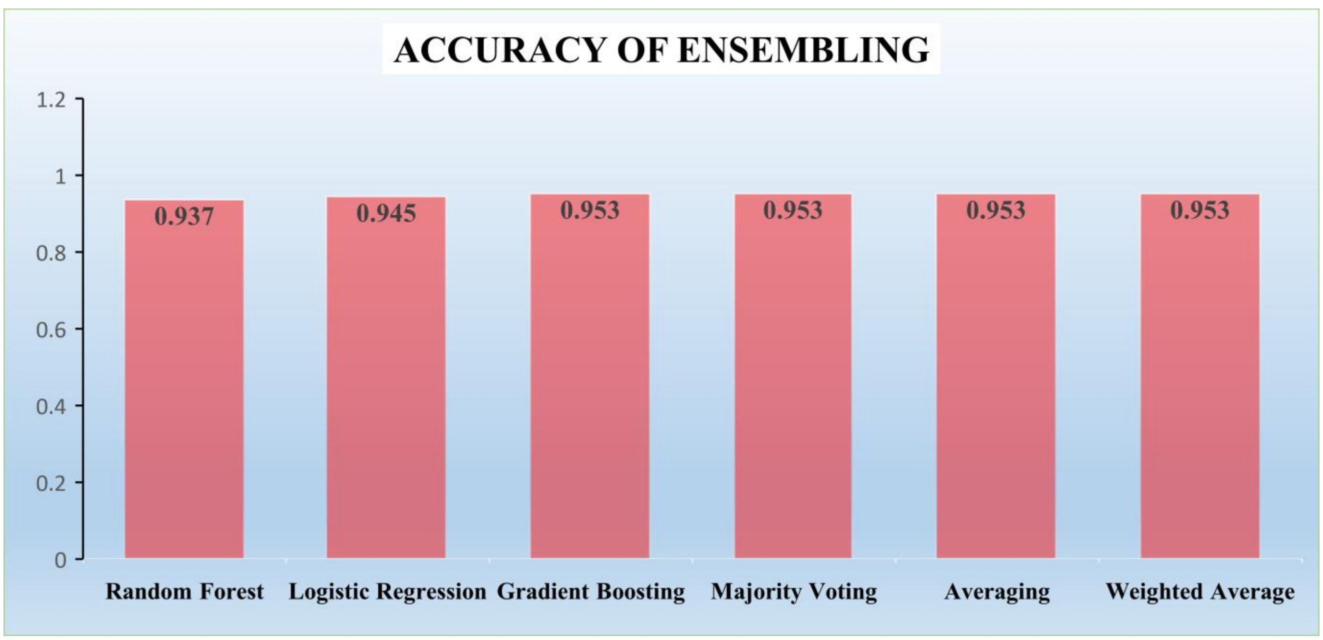
Ensemble with Important attributes		
S.no	**Algorithm**	**Accuracy**
1	Random Forest	0.937
2	Logistic Regression	0.945
3	Gradient Boosting	0.953
Averaging		**0.953**
Majority Voting		**0.953**
Weighted Average		**0.953**

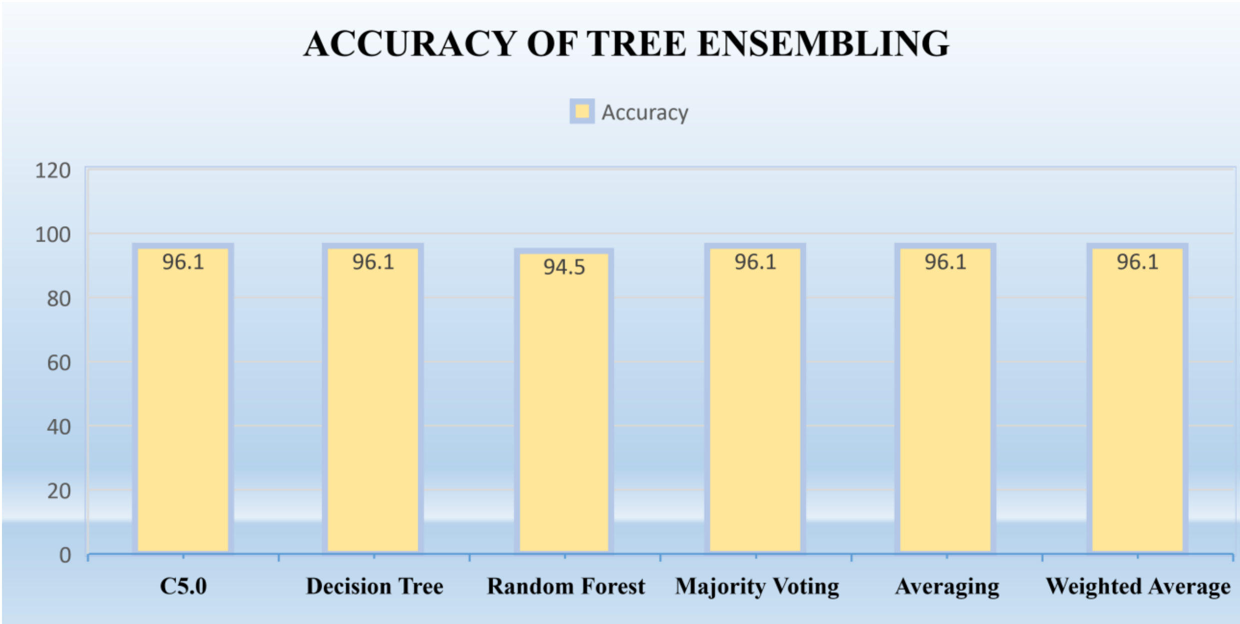
Ensemble Trees with Important attributes		
S.no	**Algorithm**	**Accuracy**
1	C5.0	0.961
2	Decision Tree	0.961
3	Random Forest	0.945
Averaging		**0.961**
Majority Voting		**0.961**
Weighted Average		**0.961**

CHARTS




ACCURACY OF ALGORITHMS WITH IMPORTANT ATTRIBUTE
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
100%
90%
80%
70%
60%
50%
40%
30%
20%
10%
0%
0.915
0.907
0.907
0.907
0.9
Multilayer Perceptron
Random Forest
Linear Regression
Gradient Boosting
Decision Tree


ACCURACY OF ENSEMBLING
1.2
1
0.8
0.6
0.4
0.2
0
0.937
0.945
0.953
0.953
0.953
0.953
Random Forest
Logistic Regression
Gradient Boosting
Majority Voting
Averaging
Weighted Average



7.CONCLUSION

Educational data mining is the interesting field of research for educationalist. With the help of EDM the educational institutions can be benefitted by identifying the week student's and give adequate training for improving the performance of the student. The classification technique is the popular data mining technique used in Educational Data Mining. This study implemented five classification algorithms which are Decision tree, Random Forest, Logistic Regression, Gradient Boosting and Multilayer Perceptron. The Logistic Regression out performed other algorithms with an accuracy of 87% using Background, Social with an accuracy of 83.5% and Course work attributes with 93%. And Multilayer Perceptron and Decision tree out performed other algorithms with an accuracy of 93% using Important attributes. Ensembling with Gradient Boosting, Random Forest and Logistic Regression gave accuracy of 95%. The Tree Ensembling using Decision tree, Random forest and C5.0 gave accuracy of 96%. As the age increases the failure rate also increases, the female students face less failures compared to male. If the study time is greater than equal to 3 hours the student can escape from failure.

8.REFERENCES

[1] Kiu, Ching-Chieh. "Data Mining Analysis on Student's Academic Performance through Exploration of Student's Background and Social Activities." *2018 Fourth International Conference on Advances in Computing, Communication & Automation (ICACCA)*. IEEE, 2018.

[2] Kaunang, Fergie Joanda, and Reymon Rotikan. "Students' Academic Performance Prediction using Data Mining." *2018 Third International Co-nference on Informatics and Computing (ICIC)*. IEEE, 2018.

[3] Ajibade, Samuel-Soma M., Nor Bahiah Ahmad, and Siti Mariyam Shamsuddin. "An Heuristic Feature Selection Algorithm to Evaluate Academic Performance of Students." *2019 IEEE 10th Control and System Graduate Research Colloquium (ICSGRC)*. IEEE, 2019.

[4] Ketui, Nongnuch, Warawut Wisomka, and Kanitha Homjun. "Using Classification Data Mining Techniques for Students Performance Prediction." *2019 Joint International Conference on Digital Arts, Media and Technology with ECTI Northern Section Conference on Electrical, Electronics, Computer and Telecommunications Engineering (ECTI DAMT-NCON)*. IEEE.

[5] Romero, Cristóbal, et al. "Predicting students' final performance from participation in on-line discussion forums." *Computers & Education* 68 (2013): 458-472.

[6] Hu, Ya-Han, Chia-Lun Lo, and Sheng-Pao Shih. "Developing early warning systems to predict students' online learning performance." *Computers in Human Behavior* 36 (2014): 469-478.

[7] Yu, Liang-Chih, et al. "Improving early prediction of academic failure using sentiment analysis on self_evaluated comments." *Journal of Computer Assisted Learning* 34.4 (2018): 358-365.

[8] Deepika, K., and N. Sathvanaravana. "Analyze and Predicting the Student Academic Performance Using Data Mining Tools." *2018 Second International Conference on Intelligent Computing and Control Systems (ICICCS)*. IEEE, 2018.

[9] Uzel, Vahide Nida, Sultan Sevgi Turgut, and Selma Ayşe Özel. "Prediction of Students' Academic Success Using Data Mining Methods." *2018 Innovations in Intelligent Systems and Applications Conference (ASYU)*. IEEE, 2018.

[10] Siddiqui, Isma Farrah, and Qasim Ali Arain. "ANALYZING STUDENTS'ACADEMIC PERFORMANCE THROUGH EDUCATIONAL DATA MINING." 3C Tecnologia (2019).

[11] Rawat, Keshav Singh, and I. V. Malhan. "A Hybrid Classification Method Based on Machine Learning Classifiers to Predict Performance in Educational Data Mining." Proceedings of 2nd International Conference on Communication, Computing and Networking. Springer, Singapore, 2019.