

Classification of Student Performance Dataset using Machine Learning Algorithms

K. Maheswari, P. Deepalakshmi, K. Ponmozhi

Abstract: The scope of this research work is to identify the efficient machine learning algorithm for predicting the behavior of a student from the student performance dataset. We applied Support Vector Machines, K-Nearest Neighbor, Decision Tree and Naïve Bayes algorithms to predict the grade of a student and compared their prediction results in terms of various performance metrics. The students who visited many resources for reference, made academic related discussions and interactions in the class room, absent for minimum days, cared by parents care have shown great improvement in the final grade. Among the machine learning techniques we have used, SVM has shown more accuracy in terms of four important attribute. The accuracy rate of SVM after tuning is 0.80. The KNN and decision tree achieves the accuracy of 0.64, 0.65 respectively whereas the Naïve Bayes achieves 0.77.

Keywords : Classification, Decision Tree , KNN , Machine Learning , Naïve Bayes , Student Performance and SVM,

I. INTRODUCTION

Today, large volume of data is available in the educational institution databases. These databases contain student details, faculty (teaching and non-teaching) details and employer details. Analyzing and monitoring the data is a complex manual process and time consuming also. So, there is a need for an efficient system to do such process effectively. We consider prediction of student performance, a part of the about said process using machine learning techniques in this paper.

The design and development of data mining classification models is used to predict students' academic performance using various parameters. The source of the dataset taken for this study is available at <https://www.kaggle.com/aljarah/xAPI-Edu-Data>. The characteristics of considered dataset are multivariate, categorical with 16 attributes with 480 instances. The attributes are gender, NationalITy, PlaceofBirth, StageID, GradeID, SectionID, Topic, Semester, Relation, raisedhands, VisITedResources, AnnouncementsView, Discussion, ParentAnsweringSurvey, ParentschoolSatisfaction,

Revised Manuscript Received on December 16, 2019.

* Correspondence Author

Dr.K.Maheswari*, Department of Computer Applications, Kalasalingam Academy of Research and Education, Krishnankovil-Virudhunagar, India. Email: maheswarisnr@gmail.com

Dr.P.Deepalakshmi, Department of Computer Science and Engineering, Kalasalingam Academy of Research and Education, Krishnankovil-Virudhunagar, India. Email: deepa.kumar@klu.ac.in

Dr.K.Ponmozhi, Department of Computer Applications, Kalasalingam Academy of Research and Education, Krishnankovil-Virudhunagar Email: chezhian71.p@gmail.com

StudentAbsenceDays and Class.

The gender attribute in the dataset has 64% male and 36% female. The student's population from Kuwait is 37%, Jordan is 36% and others are 27%. The dataset with gender distribution (in percentage) in the nationality is shown in Fig. 1. The dataset is cleaned by applying Random Forest preprocessing algorithm with caret package using R software. The checking of Not Available (NA) values or empty values and missing values are done.

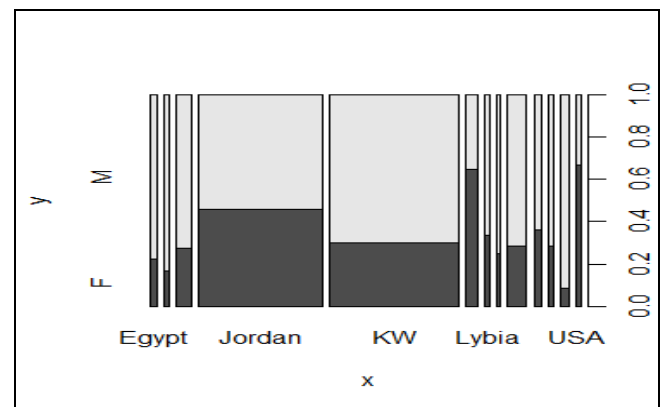


Fig. 1 Gender distribution based on Nationality in the dataset

The rest of the paper is organized as follows. The next section discusses the review of literature followed by methodology section. Then experimental result section discusses the setup and results in detail. Finally, we conclude and provide recommendations in conclusion section.

II. REVIEW OF LITERATURE

Dorina Kabakchieva et al [1] describes classification techniques that she used for student performance prediction. It includes Rule Learner, Decision Tree Classifier, Neural Network and Nearest Neighbor Classifier. The author analyzed the performance of these algorithms and conveyed that neural network model performs well among four algorithms and achieved higher accuracy of 73.59%. The decision tree model achieves 72.74%, KNN model achieves 70.49% and oneR Rule learner achieves 67.46% accuracy. The author recommended that the attributes, university examination score and the failure number at the first year level have higher influence in the classification.

CH.M.H.Sai Baba et al [2] suggested to increase the number of parameters used for classification. The attributes considered by the author were students 10th standard mark percentage, inter percentage and B.Tech first year percentage. The true prediction considers all these marks because the prediction is whether the

student will get job or not. The more number of student details like social networking interest, communication skill and others in the form of large amount of data with more attributes are needed for better prediction. Classification algorithm is applied on this dataset and prediction is done for whether the student gets placed or not.

Karunendra Verma et al [3] focused on improving slow learners performance by providing remedial classes to them. This study is applied for B.Sc computer science students with the attributes Semester - I marks, Semester - II marks, attendance during remedial teaching. The algorithms considered by the author was Decision Tree algorithm C4.5 (J48), Bayesian classifiers (Naïve Bayes and BayesNet), Nearest Neighbour algorithm (IBk) and Two rule learners - OneR and JRip. The accuracy achieved by the author was 52% to 67% and algorithm for Decision Tree C4.5(J48) showed more accuracy than the other algorithms.

Mashaal Al luhaybi et al [4] focused on the academic performance of level 2 students by applying C4.5 and Naïve Bayes. The attributes considered were admission, course related data and level 1 final grades. The author developed a model for Computer Science and Information Technology students. The Naïve Bayes and C4.5 algorithms were applied on the data set. The result achieved was 88.48 and 84.29 respectively. The author concluded Naïve Bayes performs well than C4.5 for this particular dataset.

Bhavesh Patel et al [5] discussed about the student performance by using the attributes for classification as Higher Secondary marks, FatherIncome, Lab Hours, Attendance, Internet_Uses_Learning Internal_Marks, Theory_Marks, Practical_Knowledge, Assignment_Marks , PrevSemMarks Extra curricular activities and number of Siblings. The author implemented different classification algorithms for analyzing. The algorithm J48 yielded higher accuracy with minimum error rate.

Edin Osmanbegović et al [6] evaluated the performance of teaching and learning. Multi-Layer perception, Neural Network, Naïve Bayes and decision tree algorithms are applied on summer semester results for the students of University of Tuzla during the academic year 2010 -2011. Naïve Bayes performed well than the other algorithms applied by the author. The author suggested that the model can help the students to improve their performance and reduce failing ratio.

Sajida Perveen et al [7] implemented decision tree classifier for diabetic patients to predict three types of age groups. The AdaBoost bagging and J48 algorithms are applied for Canadian dataset. The AdaBoost algorithm with ensemble performs well than the other.

Sajida Perveen et al [8] dealt with course teacher performance using student data. This is another method of improving educational standard in educational Institutions. The CGPA and internal assessment attributes were used to predict student performance. J48, Multilayer perceptron, Naïve Bayes and sequential minimal optimization algorithms were applied on the dataset collected from university of California, Irvine. J48 Decision tree performs more for the attribute evaluation process. The lowest and least important attributes are identified which is not used for classification.

From the literature study, we observed that a need for a system to analyze the large student database with minimum time. This study focuses on predicting the performance of

student performance using different attributes by applying Machine learning algorithms.

III. METHODOLOGY

The xAPI-Educational Mining Dataset with a size of 6 KB is analyzed in this article using the classification algorithms, SVM [10], Naïve Bayes [12], Decision Tree, KNN [11].The SVM[10] algorithm is mining classification algorithm which is applied to analyze the data set. The algorithm is given below.

Algorithm 1 : KNN Support Vector Machine

Input : Training set x_i , Weight W , Bias b

Output : Classified Data y_i

1. Let $S=(x_i,y_i)$ where $i= 1,2,\dots,N$ where N is the number of samples
2. Each point $x_i \in$ one of the two classes $(-1,+1)$,
3. Hyper plane divides S into 2 parts such that points with the same class label falls in one part and the other points on the other part of the hyperplane.
4. A hyper plane is identified by $W.x_i + b = 0$.
5. $W.x_i + b \geq 1$ if $y_i = +1$ and $W.x_i + b \leq -1$ if $y_i = -1$
6. Repeat steps 3-5 until SVM obtains optimal hyper plane with maximum margin on both sides.

KNN is a simple classification algorithm whose working principle is based on the similarity measure. This algorithm is measured by distance function. The distance functions used for continuous variables are Euclidean, Manhattan, Minkowski .

Algorithm 2 : KNN

Input : Training set X_i , Class Label C

Output : Finds closest neighbors of X_i

1. For $i= 1,2,\dots, N$, let X_i denotes features of i^{th} point in the dataset and let C_i be the class label for X_i
2. Calculate the distance between test data and training data of each sample using Euclidean, Hamming and Manhattan Distance.
3. Sort the calculated distances in increasing order.
4. Choose the first K distances from the sorted distance list, with $K > 0$ always.
5. Find the points corresponding to the distance obtained.
6. Let K_i represents the number of data points which belong to the i^{th} class.
7. Compare K_i and K_j , if K_i is greater, then place X_i in class C_i .

When the data is continuous, Euclidean distance is preferred and calculated as in Equ. (1).

$$d(x, y) = \text{Square Root}(x - y)^2 \quad (1)$$

For categorical values, Hamming Distance (HD) is used as in Eq.(2)

$$HD = \sum_{i=1}^k X_i - Y_i \quad (2)$$

where x and y are distance parameters. If both are equal, the distance is 0, otherwise the distance is 1. HD is Hamming Distance.

Decision Tree starts with root node and further branched as left and right child. The more interpretation is achieved by decision tree because the classification is shown visually.

Algorithm 3 : Decision Tree

Input : Training set X_i , Weight W, Bias b

Output: Constructed Decision Tree

1. Choose the best or most powerful attribute for root node.
2. Split the dataset or training set into sub trees.
3. Each sub tree contains the same value for the attribute
4. The steps 1,2 and 3 are repeated for each sub tree until there is no more node to branch.

Naïve Bayes algorithm works based on the probability of Bayes Theorem. This algorithm converts the dataset into frequency table and likelyhood table. Frequency table is a list of items with its occurrence. The number of times each item has occurred is represented in frequency table. Likelyhood table is a joint probability distribution of observed data. The features are independent to each other in this algorithm. Because of this nature, it quickly learns higher dimensional attributes with minimum training data. Hence, this algorithm is used in many real life applications such as medical diagnosis, spam filtering, and speech, image data analysis.

Algorithm 4 : Naïve Bayes

Input : Training set, Test Set

Output: Class

1. Convert the training set into frequency table .
2. Transform frequency table into likelihood table.
3. Apply Naïve Bayes equation to find the posterior probability. The equation is

$$P(m/N) = \frac{P(N/m)P(n)}{P(N)}$$

where

- P(m/N) posterior probability
- P(n) class prior probability
- P(N) predictor prior probability
- P(N/m) Likelyhood

4. The maximum value obtained by posterior probability P(m/N) is considered to be the outcome of prediction.

IV. EXPERIMENTAL RESULTS

The importance of attributes by varimp method for student performance dataset is evaluated and is shown in Fig 2. After data collection, the feature selection process is used before applying classification algorithms. If the data set has more number of features, it is very difficult to apply all the features for classification. So, choosing of important feature is done with the help of varimp function in R software. The 16

attributes are ordered by maximum importance across the classes in Fig.2.

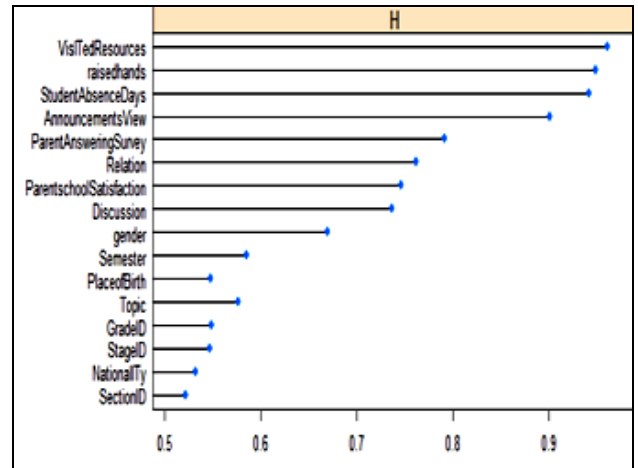


Fig. 2. Important variable prediction

The recursive feature selection was applied on the dataset to improve the accuracy of feature selection process. The 10 fold cross validation of outer resampling method was done to select the top five variables out of 16. The attributes selected are StudentAbsenceDays, VisITedResources, raisedhands, AnnouncementsView and ParentAnsweringSurvey.

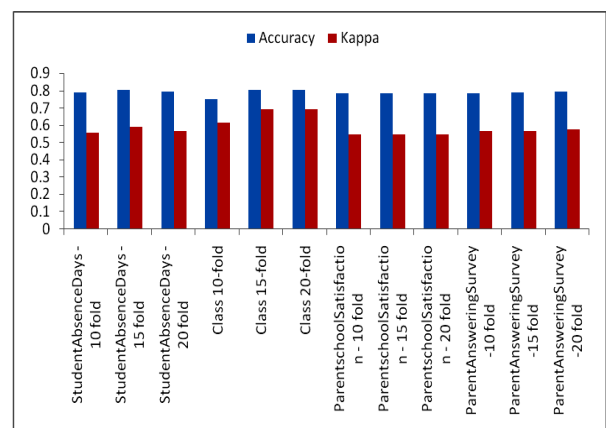


Fig.3 SVM accuracy

SVM Linear Kernel C-classification algorithm with cost 1 and gamma 0.01612903 of 480 samples is implemented on these top 5 attributes. This setup is applied for various attributes and accuracy of the attribute StudentAbsenceDays has shown increased performance as in Fig. 3. Kappa is a measure of agreement of two features or two measures. Its value ranges from 0 to 1. Kappa value 1 implies that there is a perfect agreement between the two. The possible interpretations of kappa are

- Poor agreement takes the value Less than 0.20
- Fair agreement ranges from 0.20 to 0.40
- Moderate agreement lies between 0.40 to 0.60
- Good agreement is from 0.60 to 0.80
- Very good agreement range is 0.80 to 1.00

Naïve Bayes algorithm is applied for each attribute of the dataset individually to determine the performance of classification algorithm. The attribute class versus StudentAbsenceDays, class, ParentschoolSatisfaction and ParentAnsweringSurvey were analyzed. The accuracy attained are 0.77, 0.74, 0.74 and 0.72 respectively and the same is shown in Fig.4. When it is applied for four attributes together the performance gets improved drastically.

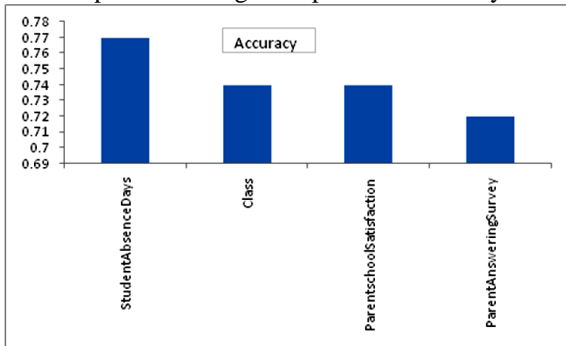


Fig 4. Naive Bayes accuracy

The K nearest neighbor algorithm is applied for the attributes Class, ParentschoolSatisfaction, ParentAnsweringSurvey, StudentAbsenceDays with different K values. For K, values are varied within the range from 1 to 10. Accuracy is used to select the optimal model using the largest K value. Hence, the K value 9 has shown more accuracy than the other values. Accuracy is used to select the optimal model using the largest K value. and is shown in table 1.

Table -1: Selection of optimal K value

K	Accuracy	Kappa
1	0.6436908	0.4658610
2	0.6457741	0.4689438
3	0.6416074	0.4622289
4	0.6436908	0.4658610
5	0.6436908	0.4656153
6	0.6416074	0.4624745
7	0.6436908	0.4651509
8	0.6436908	0.4656153
9	0.6457741	0.4689449
10	0.6416074	0.4625250

The optimal KNN model was selected by using the predicted accuracy value and their neighbor is shown in Fig. 5.

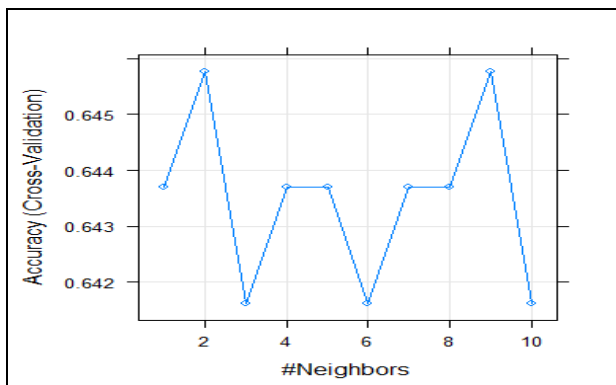


Fig. 5. KNN accuracy for K in the range 1 to 10

The selection of K value is tried for 11 to 20. The accuracy and kappa value is less than the value achieved by K=9. So, again the optimal K value is 9 which is shown in Fig. 6.

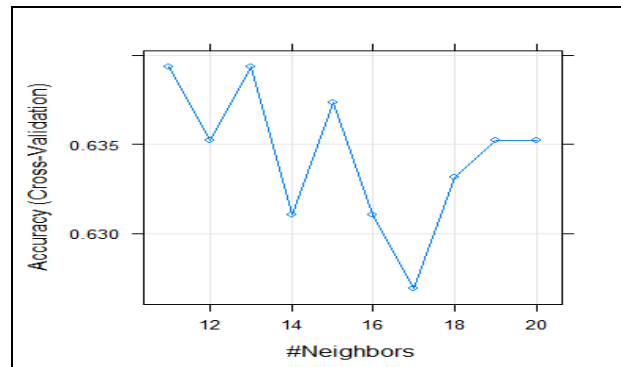


Fig. 6. KNN accuracy for K in the range 11 to 20

Again the KNN training is applied for K value ranging from 21 to 30. It does not show any improvement for this dataset as found in Fig. 7.

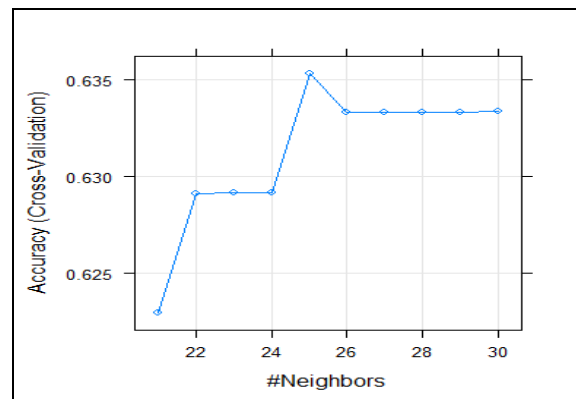


Fig. 7. KNN accuracy for K in the range 21 to 30

Class, StudentAbsenceDays, ParentAnsweringSurvey and ParentSchoolSatisfaction attributes are used for training. The optimal value chosen for these attributes is K=9. The same KNN model is applied for only two attribute Class and ParentSchoolSatisfaction. The accuracy and Kappa value is 0.4373445 and 0.08699107 is shown in Fig 8. Kappa is increased but accuracy is shown very less value. If the accuracy is not improved for different attributes and different functions, then the training can be stopped at that point and can try for another algorithm. So, the next model decision tree is directed.

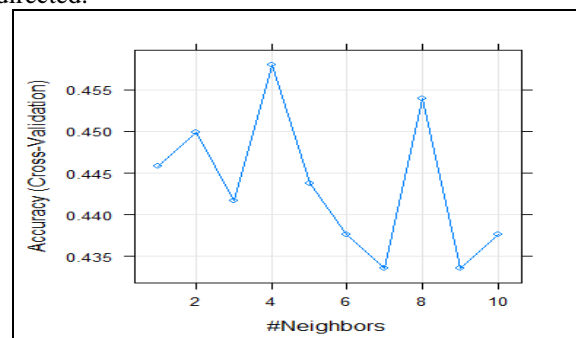


Fig. 8. Class with ParentschoolSatisfaction

The classification algorithm decision tree is implemented using the top 5 attributes of student performance dataset. The prediction is presented in confusion matrix represented in Fig. 9.

Prediction	High	Low	Medium
High	116	2	84
Low	1	92	21
Medium	25	33	106

Fig. 9. Confusion Matrix

The accuracy achieved is 0.65 with Precision 0.82, Recall 0.57 and F-measure 0.67. The high (H) is the class which predicts the students in higher class of 116. The root node error is a misclassification error. There are 480 samples in the dataset, out of these samples, the wrongly predicted samples are 269. The error rate is 0.56042 and is shown in figure 9.

$$RNN = \frac{WPS}{TNS} \quad (3)$$

where RNN is Root Node Error, WPS is Wrongly Predicted Samples and TNS is Total Number of Samples. Therefore, $RNN = \frac{269}{480} = 0.56042$. The number of splits, reerror, xerror and xstd are shown in table 1 and was plotted in Fig. 10.

Table 2 Error Rate

CP	NSPLIT	RELERRO R	XERROR	XSTD
1	0	1.00000	1.00000	0.04042
0.167286				4
2	1	0.83271	0.89963	0.04072
0.118959				1
3	2	0.71375	0.81784	0.04058
0.096654				1
4	3	0.61710	0.65056	0.03920
0.010000				1

Relative error will not be taken into account for pruning. The relative error and Xerror is reduced when the tree grows to more levels. If number of level is less, then the error rate will also be less. In this work, more number of levels in classification. So, error rate is also more.

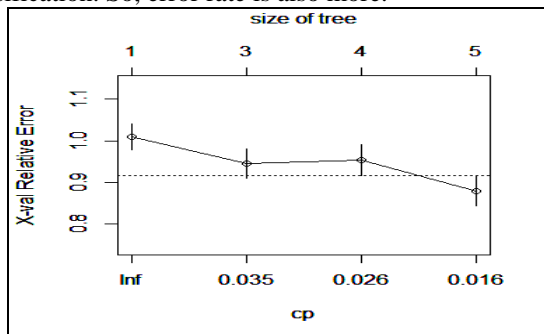


Fig.10. Relative error

The number of splits of a decision tree is shown in Fig. 11. Splitting is defined as the process of dividing a node into sub nodes. Decision tree splits the training data set in to smaller

samples repeatedly. X relative error is an error of cross-validation. If the number of levels of tree increases, cross validation error increases. Number of splits with R-square mean error and X Relative error is shown in Fig. 12.

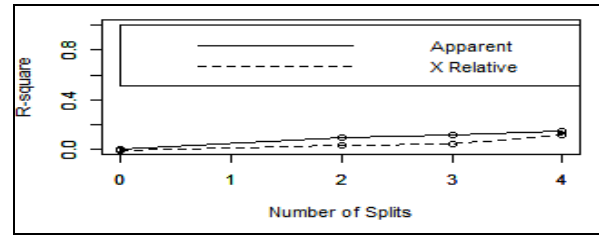


Fig.11. Splits of decision tree with R-square

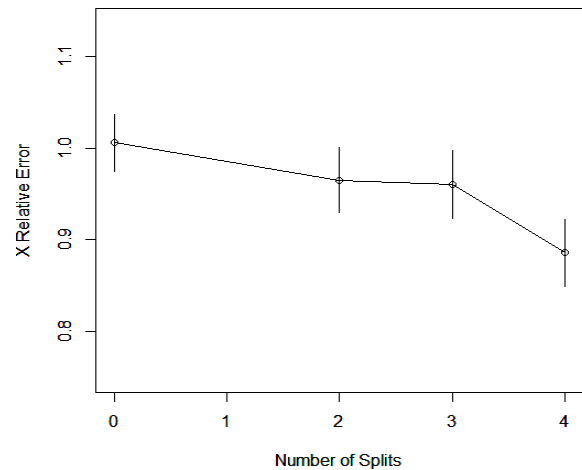


Fig.12. Splits of decision tree with X-Relative Error

The decision tree after pruning is shown in Fig. 13. Pruning is a process of reducing the levels of decision tree. The purpose of reducing the size of the tree is to increase the accuracy. To avoid overfitting of decision tree, the decision tree is pruned. More levels in the above tree is reduced and is shown in Fig. 13.

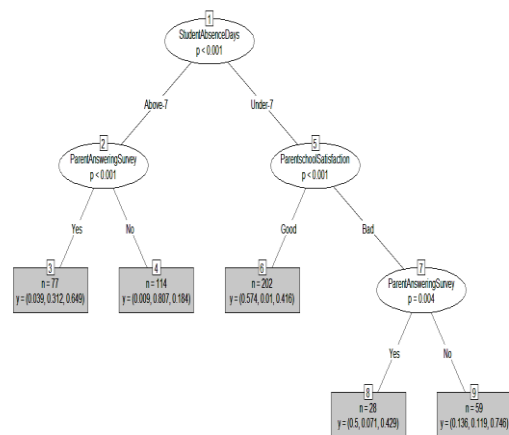


Fig.13. Decision Tree after pruning

The comparison results of the classification algorithms considered in this paper are plotted in Fig.14. SVM performs well for this dataset

compared to other algorithms because SVM is well suited for two class classification problem. There is no overfitting issue and complicated decision boundary in SVM. It uses kernel mapping for efficient computation and training. It generates minimum error rate, and hence, the accuracy is maintained higher. Naive Bayes performance is based on the probability model. The Naïve Bayes classification selects the class label with the highest probability. The largest probability is always associated with the correct class label. It needs high confidence to reach the highest probability frequency value 1.

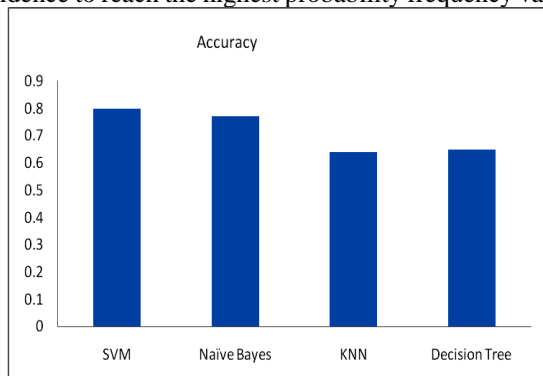


Fig.14. Accuracy chart

KNN performance depends on the K value. It is suitable for homogeneous features. It requires fast and expensive testing. KNN is a lazy training algorithm. Hence, the accuracy achieved is less. Decision Tree is a supervised learning algorithm. A small change in the data reflects more in the final estimated decision tree. The issues of decision tree is over fitting, partitioning and handling non numerical data. This issues will affects and degrade the performance.

V. CONCLUSION

From the comparative results of various data mining classification algorithms for student performance dataset, it is observed that SVM performs well than other algorithms. Among the 16 features in the dataset, features like gender, Nationality, PlaceofBirth, StageID, GradeID, SectionID, Topic, Semester, Relation did not have much impact on the performance of the student. But, the student absent days have influence on the performance of the student. Some improvement in terms of tuning, pruning, splitting is needed for improving the performance. The future work can concentrate on hybrid algorithms or other advanced algorithms to improve the accuracy.

REFERENCES

1. Dorina Kabakchieva ,” Student Performance Prediction by Using Data Mining Classification Algorithms”, International Journal of Computer Science and Management Research Vol 1 Issue 4 November 2012, ISSN 2278-733X.
2. CH.M.H.Sai Baba, AkhilaGovindu , Mani Krishna Sai Raavi, Venkata Praneeth Somisetty,” Student Performance Analysis Using Classification Techniques”, International Journal of Pure an Applied Mathematics, Volume 115 No. 6 2017, 1-7.
3. Karunendra Verma, Arjun Singh, Purushottam Verma,” A Review on Predicting Student Performance Using Data Mining Method”, International Journal OF Current Engineering and Scientific Research (IJCESR), ISSN (Print): 2393-8374, (Online): 2394-0697, Volume-3, Issue-1, 2016
4. Mashael Al luhaybi, Allan Tucker and Leila Yousefi ,” The Prediction of Student Failure Using Classification Methods: A Casestudy”, pp. 79–90, 2018. © CS & IT-CSCP 2018
5. Bhavesh Patel , Chetan Gondaliya,” Student Performance Analysis Using Data Mining Technique”, International Journal of Computer Science and Mobile Computing, Vol.6 Issue.5, May- 2017, pg. 64-71.
6. Edin Osmanbegović , Mirza Suljić ,Data Mining Approach For Predicting Student Performance, Economic Review – Journal of Economics and Business, Vol. X, Issue 1, May 2012. Citations :155
7. Sajida Perveen, Muhammed Shahbaz, Aziz Guergachi,Karim Keshavjee, “Performance Analysis of Data Mining Classification Techniques to Predict Diabetes” , Science Direct Elsevier,Vol 82,pages: 1-142,2016.
8. Ahmed Mohamed, Ahmet Rizaner, Ali Hakanulusoy,”Using Data Mining to Predict Instructor Performance” 12 th international Conference on application of fuzzy systems and soft computing,ICAFS,2016,29-30,Aug 2016,Vienna.
9. K. Maheswari, P. Packia Amutha Priya , "Analysis and Implementation of Text Mining for Different Documents", International Journal of Scientific Research in Science and Technology(IJSRST), Print ISSN : 2395-6011, Online ISSN : 2395-602X, Volume 3, Issue 5, pp.109-113, May-June-2017,URL : <http://ijsrst.com/ICASCT2518>
10. Dr.K.Maheswari,Ms.P.Packia Amutha Priya “Classification of Twitter Data Set using SVM and KSVM”, Published in international Journal of Pure and Applied Mathematics, Volume 118 No. 7 2018,PP: 675-680, ISSN: 1311-8080 (printed version); ISSN: 1314-3395.
11. K.Maheswari,“Improving Accuracy of Sentiment Classification Analysis in twitter Data Set Using knn” published in International Journal of Research and Analytical Reviews,Vol 5, Issue 1, PP:422-425/E ISSN 2348-1269 Print ISSN 2349-5138,UGC Approved Journal.
12. Dr.K.Maheswari,Ms.P.Packia Amutha Priya “Predicting Customer Behavior in Online Shopping Using SVM Classifier”, presented paper in 2017 IEEE International Conference on Intelligent Techniques in Control, Optimization & Signal Processing, INCOS'17, published in IEEE Xplore , 01 March 2018.

AUTHORS PROFILE



Dr. K .Maheswari received her B.sc (Computer Science) from Madurai Kamaraj University and MCA. M.Phil. from Bharathidasan University. She has completed her Ph.d at Bharathiar University. She is currently working as an Associate Professor in the Department of Computer Applications, Kalasalingam Academy of Research and Education. She has 23 years of teaching experience. She has presented research papers in several national and international conferences. She has published many research papers in various international journals. Her research interest is VoIP , network security and Data Mining.



Dr. P. Deepalakshmi is currently working as a Professor in Department of Computer Science and Engineering at Kalasalingam Academy of Research and Education (KARE), Virudhunagar, Tamilnadu, India. She is also serving as Dean, School of Computing. Her research interest includes Optimization Techniques, Network Routing, Distributed Computing, Network Security, Data Analytics, Machine Learning Techniques. She also takes care of KARE ACM student chapter as faculty mentor. Contact her at deepa.kumar@klu.ac.in



Dr. K. Ponmozhi is currently working as Assistant professor in the Department of computer applications, Kalasalingam Academy of Research and Education, Srivilliputhur, TamilNadu, India. She received her Bachelors degree in computer science from Madurai Kamaraj University, her MCA from Bharathidasan University. She received her Ph.D in computer science from Mother Teresa women's University. She has more than 20 years of teaching experiences. She published many articles in journals and conferences both in National level and International Level. Her area of interest is networking, computer graphics, image processing, cognitive science.

