WWW.IJRPR.COM

International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

Predicting Students Academic Performance Using Supervised Learning

Ngene C C[1], Okolo C C[2], Asogwu E C[1], Iwegbuna O N[1], Belonwu T S[1]

[1]Department of Computer Science,Nnamdi Azikiwe University, Awka, Anambra State, Nigeria

[2]Electronics Development Institute,Federal Ministry of Sci and Tech, Anambra state, Nigeria

ABSTRACT

Timely prediction of the academic performance of postgraduate students is an important topic in the area of postgraduate Studies. Ability to predict student's performance provides an opportunity for improvement in educational sector in that it helps to design mechanisms that can be used to improve the educational standard and avoid dropout of students. The research work presented a Naïve Bayes classification and Support Vector Machine (SVM) algorithm to predict Postgraduate students' academic performance using demographic factors and first semester examination scores. The dataset used in this research involves 145 Masters Students of 2014/2015 session as well as their demographic attributes and first semester result. Using Naïve Bayes and Support Vector machine (Sequential Minimum Optimization- SMO) supervised learning algorithm with twenty two identified features, the prediction showed an accuracy of 84.14% and 88.97% respectively

Keywords: Classification, Predictive Model, Supervised learning, Naïve Bayes, data mining

INTRODUCTION

Machine learning is an application of artificial intelligence (AI) that provides systems the ability to automatically learn and improve from experience without being explicitly programmed. *It centers on the development of computer programs* that can access data and use it learn for themselves. Machine learning is categorized into supervised learning, unsupervised learning and reinforcement learning. In supervised learning, the model is trained using labeled data (the input and the desired output are provided in order to make them learn and establish relationships between given inputs and outputs. Once the model gets trained it can start making a prediction or decision when new data is given to it. Supervised learning can be classified into Classification (when the output is a category such as true or false, cat or dog, rich or poor etc) and Regression (when the output is a real, computed value such as the price or weight). Supervised learning algorithms include K-Nearest Neighbour, Naïve Bayes, Decision Trees, Support Vector Machine, Random Forest, Logistic Regression etc. In Unsupervised Learning, the model is given a dataset which is neither labelled nor classified. There is no training phase where labeled data is fed to the learning algorithm in order to train the model, instead the algorithm has to figure out things by itself .The model explores the data and draws inferences from datasets to define hidden structures from unlabelled data. Two types of unsupervised learning are Clustering and Association. Unsupervised learning algorithms include K-Means Clustering, Hierarchical Clustering, Principal Component Analysis etc.

Reinforcement learning is a kind of Machine Learning that works on the concept of Reward maximization whereby system that is to be trained, learns on its own based on its previous experiences and outcomes while doing a similar kind of a job. The system takes a decision, learns from the feedback and takes better decisions in the future.

Naive Bayes is a powerful supervised machine learning algorithm that is used for classification. It is an extension of the Bayes theorem and is called "Naïve" because the classifier assumes that the input features/ attributes that go into the model are independent of each other therefore a change in one input feature won't affect the others. Naïve Bayes can be used for binary classification and multi-classification. It requires less training data. It is an eager learning classifier and is quite fast in its execution; hence it can be used for making real-time predictions.

Supervised learning involves some steps. The first step involves gathering the labelled training data. This is the output that provides feedback to the algorithm. Next the labeled data is split into three sets: Training and Test data. Training dataset is the dataset that is used to train the model. The model learns from this data. Once the machine learning algorithm learns the underlying patterns of the training data, it needs to be tested on fresh data that it has never seen before, but which still belongs to the same distribution as the training data. This is called the Test data. If the model performs well on the test data then it is considered as a machine learning model that generalizes the dataset of interest. The main aim of this study is to classify student's academic into three classes: Poor, Average and Good with the help of collected features such as age, gender, city of residence, marital status, mode of study, employment status, family size and first semester result. These features and its respective values were collected via questionnaires and the records unit of the University. Naïve Bayes and Support Vector Machine classification algorithm were applied to the dataset and classify the student

into their respective category. This will enable teachers and academic advisors predict student's academic standing and offer academic advising based on their performance.

LITERATURE REVIEW

Altujjar et al., (2016) built a predictive model based on records of female students using the ID3 decision tree algorithm to reveal the courses affecting low academic performance at the IT department, King Saud University, Riyadh, Saudi Arabia. They built several models based on ID3 decision tree algorithm. They divided the dataset into three groups to build separate model for each group. Results suggested that the classification model based on performance in the second year was the most accurate. The student performance in IT 221 and the two programming courses, CSC111 and CSC113, was a great indicator of student level of achievement. Badr et al., (2016) built an application to predict student's performance in a programming course based on their previous performances in specific mathematics and English courses. In addition, the model aimed to reduce dropout rates by helping students predict their performance in programming courses before enrolling for them. Two experiments were conducted using the CBA rule-generation algorithm. They first used student's grades in two English courses and two mathematics courses, which generated four rules with accuracy of 62.75%. The second used student's grades only in two English courses, generating four rules with accuracy of 67.33%. The results showed that student's performance in English courses had a significant predictive effect on their performance in the programming course. The literature review about predicting performance mentioned above show that it is possible to predict performance of students with a reasonable accuracy. The above mentioned works used only academic record to predict their result. However, both academic result and demographic attributes is used in this work. This aspect differ our works from other works.

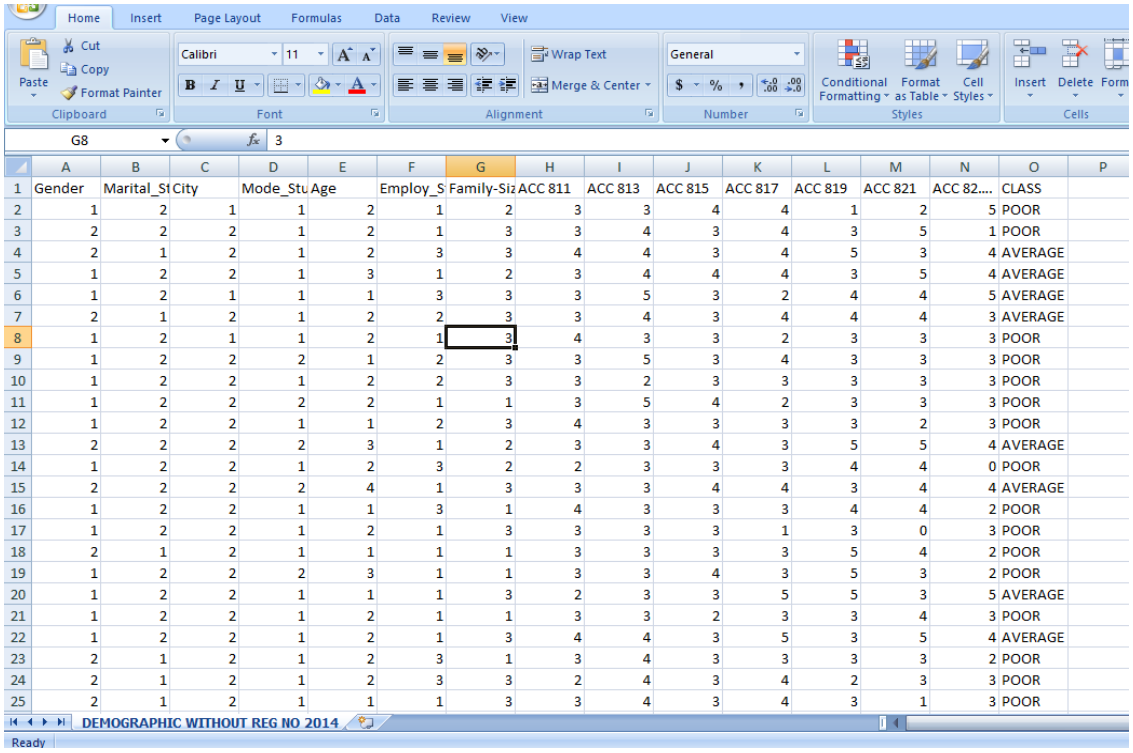
MATERIALS AND METHODS

The Data mining tools and technique that were used are Classification technique and WEKA (Waikato Environment for Knowledge Analysis (WEKA) tool. Data was collected from multiple sources: via questionnaire and academic records unit Nnamdi Azikiwe Univesrity. Records of students were collected from the Postgraduate School while their demographic data was collected via questionnaires. The experiment was carried out using the standard data sets of Postgraduate students using 145 instances with only 14 attributes (gender, marital status, city, mode of study, age bracket, employment status, family size, ACC 811, ACC 813, ACC 815, ACC 817, ABB 819, ACC 821, and Elective Course). The 15th attribute in represents the class that is to be predicted by the algorithm. The data was then loaded into (Waikato Environment for Knowledge Analysis (WEKA) environment. Weka is an open source data mining tools that contains collection of machine learning algorithms for data mining tasks. The algorithms can either be applied directly to a dataset or called from the user's own Java code. Weka contains tools for data pre-processing, classification, regression, clustering, association rules, and visualization. It is also well-suited for developing new machine learning schemes. Table 3.0 shows a summary of the attributes and the scaling factors used for the research while Table 3.1 shows the sample data that was used for the analysis. The values from the given sample were transformed into an excel format with an extension of csv for machine readable task.

Table 1: Attributes and Their Scaling Factors

Attribute	**Scaling Factor**
Gender	(1) Female (2) Male
Marital Status	(1)Single (2) Married (3) Divorced
City	(1)Rural (2) Urban
Mode of Study	(1)Part time (2) Full Time
Age Bracket	(1)20-30 (2) 31-40 (3) 41-50 (4) 50 and above
Employment Status	(1)Full Time (2) Part Time (3) Unemployed
Family Size	(1)1-3 (2) 4-6 (3) 7-9 (4) 10 and above
ACC 811	Grade Point for ACC 811 (A: 5, B: 4, C: 3, D: 2, E: 1, F: 0)
ACC 813	Grade Point for ACC 813 (A: 5, B: 4, C: 3, D: 2, E: 1, F: 0)
ACC 815	Grade Point for ACC 815 (A: 5, B: 4, C: 3, D: 2, E: 1, F: 0)
ACC 817	Grade Point for ACC 817 (A: 5, B: 4, C: 3, D: 2, E: 1, F: 0)
ACC 819	Grade Point for ACC 819 (A: 5, B: 4, C: 3, D: 2, E: 1, F: 0)
ACC 821	Grade Point for ACC 821 (A: 5, B: 4, C: 3, D: 2, E: 1, F: 0)
ACC 823 (Elective)	Grade Point for ACC 813 (A: 5, B: 4, C: 3, D: 2, E: 1, F: 0)
Class	Good: 4.00-5.00, Average: 3.50-4.00, Low: 0.00-3.49

Table 2: Dataset for Implementation of the new System



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1	Gender	Marital_St	City	Mode_Stu	Age	Employ_S	Family-Siz	ACC 811	ACC 813	ACC 815	ACC 817	ACC 819	ACC 821	ACC 82....	CLASS
2	1	2	1	1	2	1	2	3	3	4	4	1	2	5	POOR
3	2	2	2	1	2	1	3	3	4	3	4	3	5	1	POOR
4	2	1	2	1	2	3	3	4	4	3	4	5	3	4	AVERAGE
5	1	2	2	1	3	1	2	3	4	4	4	3	5	4	AVERAGE
6	1	2	1	1	1	3	3	3	5	3	2	4	4	5	AVERAGE
7	2	1	2	1	2	2	3	3	4	3	4	4	4	3	AVERAGE
8	1	2	1	1	2	1	3	4	3	3	2	3	3	3	POOR
9	1	2	2	2	1	2	3	3	5	3	4	3	3	3	POOR
10	1	2	2	1	2	2	3	3	2	3	3	3	3	3	POOR
11	1	2	2	2	2	1	1	3	5	4	2	3	3	3	POOR
12	1	2	2	1	1	2	3	4	3	3	3	3	2	3	POOR
13	2	2	2	2	3	1	2	3	3	4	3	5	5	4	AVERAGE
14	1	2	2	1	2	3	2	2	3	3	3	4	4	0	POOR
15	2	2	2	2	4	1	3	3	3	4	4	3	4	4	AVERAGE
16	1	2	2	1	1	3	1	4	3	3	3	4	4	2	POOR
17	1	2	2	1	2	1	3	3	3	3	1	3	0	3	POOR
18	2	1	2	1	1	1	1	3	3	3	3	5	4	2	POOR
19	1	2	2	2	3	1	1	3	3	4	3	5	3	2	POOR
20	1	2	2	1	1	1	3	2	3	3	5	5	3	5	AVERAGE
21	1	2	2	1	2	1	1	3	3	2	3	3	4	3	POOR
22	1	2	2	1	2	1	3	4	4	3	5	3	5	4	AVERAGE
23	2	1	2	1	2	3	1	3	4	3	3	3	3	2	POOR
24	2	1	2	1	2	3	3	2	4	3	4	2	3	3	POOR
25	2	1	2	1	1	1	3	3	4	3	4	3	1	3	POOR

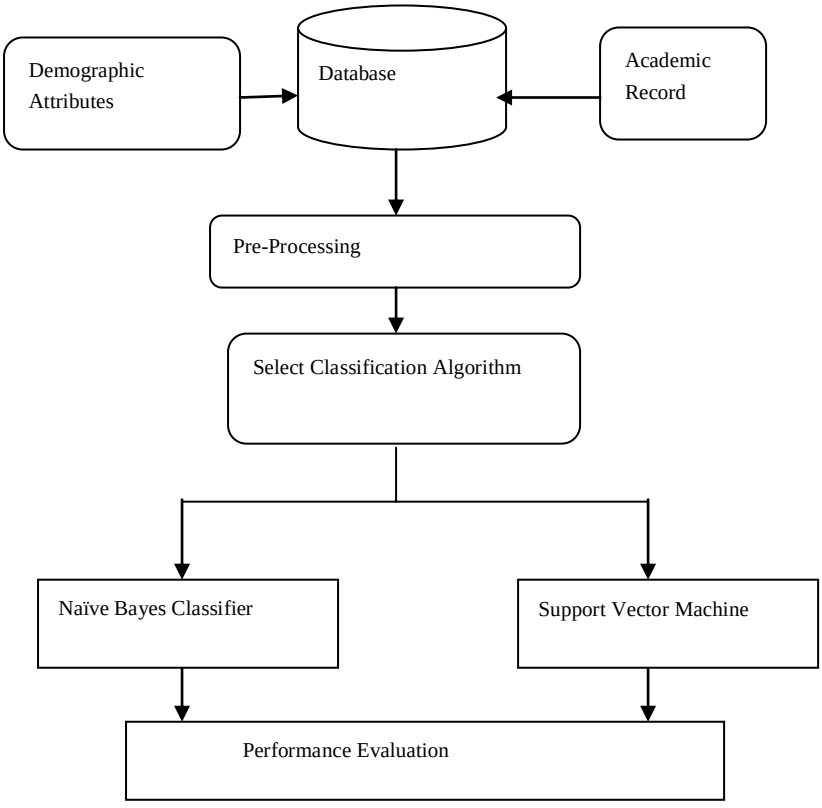


Figure 1: Graphical representation of the proposed System

RESULTS AND ANALYSIS

These loaded data was thereafter pre-processed to transform the data from the raw form to a much more usable or desired form before feeding in the algorithms. It involved data cleaning (removal of noisy and irrelevant data from the collection), data integration, data normalization (transforming numerical values such as GPA parameters to nominal or categorical class), and data selection (selecting those features which are more informative or relevant). Two well known classification algorithms, Naïve Bayes and Support Vector Machine (Sequential Mimimum Optimization SMO) were applied to the data. The two algorithms were also tested using the 10 Cross Validation and percentage split (90% training data and 10% testing data). These measures were used to assess the predictive performance of the model and to judge how it performed to the test data. The Correctly and Incorrectly classified Instances after implementing listed algorithm using 10-fold cross validation and percentage split are given in tables. The results including those correctly classified , kappa statics , time to build the model, precision, recall and F-Measure are also shown in table 3 and 4.

Table 3: Performance metrics results (10 Folds Cross Validation)

Data Mining Technique	Correctly Classified	Incorrectly Classified	Kappa Statistics	Time Taken to build the model
Naïve Bayes	84.1379%	15.8621%	0.6946	0 sec
SMO	88.9655%	11.0345%	0.7841	0.13 sec

Table: 4: Filtered Result using 10 Folds Cross Validation

Algorithm	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
Naïve Bayes	0.850	0.077	0.932	0.850	0.769	0.939	0.889	0.938	POOR
	0.900	0.200	0.761	0.900	0.824	0.690	0.884	0.781	AVERAGE
	0.000	0.007	0.000	0.000	0.000	0.016	0.850	0.190	GOOD
Weighted Average	0.841	0.125	0.829	0.841	0.832	0.709	0.913	0.847	
SMO	0.925	0.077	0.937	0.925	0.931	0.847	0.924	0.908	POOR
	0.917	0.129	0.833	0.917	0.873	0.779	0.894	0.798	AVERAGE
	0.000	0.000	0.000	0.000	0.000	0.000	0.500	0.034	GOOD
Weighted Average	0.890	0.096	0.862	0.890	0.875	0.789	0.897	0.832	

Table 3 and 4 shows the result on the analysis from the 10 folds Cross validation. The entire data set was used for the cross validation. 84.1379% and 88.9655% were correctly classified for Naïve Bayes and SMO respectively. Also the time taken to build the models were 0sec and 0.13sec for Naïve Bayes and SMO respectively

Table 5: **Performance metrics results (Percentage Split- 90% Training and 10% Test)**

Data Mining Technique	Correctly Classified	Incorrectly Classified	Kappa Statistics	Time Taken to build the model	Time Taken to test the model
Naïve Bayes	92.8571%	7.1429%	0.8627	0 sec	0 sec
SMO	100%	0%	1	0.002sec	0 sec

Table 6: Filtered Result using 10 Folds Cross Validation

Algorithm	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
Naïve Bayes	1.000	0.000	1.000	1.000	1.000	1.000	1.000	1.000	POOR
	0.833	0.000	1.000	0.833	0.909	0.861	0.938	0.944	AVERAGE
	0.000	0.071	0.000	0.000	0.000	0.000	-	-	GOOD
Weighted Average	0.929	0.000	1.000	0.929	0.961	0.940	0.973	0.976	
SMO	1.000	0.000	1.000	1.000	1.000	1.000	1.000	1.000	POOR
	1.000	0.000	1.000	1.000	1.000	1.000	1.000	1.000	POOR
	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	GOOD
Weighted Average	1.000	0.000	1.000	1.000	1.000	1.000	1.000	1.000	

Table 5and 6 shows the result on the analysis from the using Percentage Split (90% Training and 10% Test Data). The entire data set was used for the cross validation. 92.8571%% and 100%% were correctly classified for Naïve Bayes and SMO respectively. Also the time taken to build the models was 0sec and 0.002sec sec for Naïve Bayes and SMO respectively. The result from the Kappa Statistics also showed that the models performed well.

(1) Accuracy is the total number of values that are classified correctly.
(2) Precision is the number of the predicted positive values that were correct
(3) Recall is the number of positive values that were correctly identified

Using Naïve Bayes and Support Vector machine (Sequential Mimimum Optimization- SMO) supervised learning algorithm with twenty two identified features and 145 instances, the prediction showed an accuracy of 84.14% and 88.97% for Naïve Bayes and SMO respectively using 10 Folds Cross Validation.

Using Percentage Split (90% Training and 10% Test Data), the the prediction showed an accuracy of 92.86% and 100% for Naïve Bayes and SMO respectively.

CONCLUSION

Predicting student performance using supervised learning (Classification technique) can be useful to the managements in many environments. This study makes use of two supervised learning algorithms; Naïve Bayes and Support Vector Machine (using SMO) algorithm to predict the final performance of the students based on their demographic attribute and their previous performances. From the results Support Vector Machine performed better than Naïve Bayes in predicting student performance

REFERENCES

[1] Altujjar, Y., Altamimi, W., Al-Turaiki, I., & Al-Razgan, M.(2016), Predicting Critical Courses Affecting Students Performance: A Case Study. Procedia Computer Science 82 (2016) 65 71.

[2] Badr, G., Algobail, A., Almutairi, H., & Almutery, M. (2016). Predicting Students Performance in University Courses: A Case Study and Tool in KSU Mathematics Department. Procedia Computer Science 82 (2016) 80 89.

[3] Jadhav, S., & Channe, H. (2017). Comparative Study of K-NN , Naive Bayes and Decision Tree Classification Techniques. vol. 5, no. 1, pp. 2014–2017, 2016.

[4] Kumar, M., & Singh, J., & Handa, D. (2017). Literature Survey on Student's Performance Prediction in Education using Data Mining Techniques. *International Journal of Education and Management Engineering*. 6. 40-49. 10.5815/ijeme.2017.06.05.

[5] Makhtar, M., Nawang, H., & Shamsuddin, S. (2017). Analysis on students performance using Naïve Bayes classifier. J. Theor. Appl.Inf. Technol. 95(16):3993–3999, 2017.