ISSN:
Print - 2277 - 0593
Online - 2315 - 7461

Journal of Natural Science, Engineering and Technology

PREDICTING STUDENTS' GRADE SCORES USING TRAINING FUNCTIONS OF ARTIFICIAL NEURAL NETWORK

B. A. KALEJAYE[1], O. FOLORUNSO[2] AND O. L. USMAN[1]

[1]Department of Computer Science, Tai Solarin University of Education, Nigeria.
[2]Department of Computer Science, Federal University of Agriculture, Abeokuta, Nigeria.

***Corresponding author:** kalejaiyebiola@yahoo.com **Tel:** +2348062955460

ABSTRACT

The observed poor quality of graduates of some Nigerian Universities in recent times has been traced to non-availability of adequate mechanism. This mechanism is expected to assist the policy maker project into the future performance of students, in order to discover at the early stage, students who have no tendency of doing well in school. This study focuses on the use of artificial neural network (ANN) model for predicting students' academic performance in a University System, based on the previous datasets. The domain used in the study consists of sixty (60) students in the Department of Computer and Information Science, Tai Solarin University of Education in Ogun State, who have completed four academic sessions from the university. The codes were written and executed using MATLAB format. The students' CGPA from first year through their third year were used as the inputs to train the ANN models constructed using ***nntool*** and the Final Grades (CGPA) served as a target output. The output predicted by the networks is expressed in-line with the current grading system of the case study. CGPA values simulated by the network are compared with the actual final CGPA to determine the efficacy of each of the three feed-forward neural networks used. Test data evaluations showed that the ANN model is able to predict correctly, the final grade of students with 91.7% accuracy.

Key words: *Students, Academic Performance, Neural Network, Training Functions, University, CGPA*

INTRODUCTION

The art of the using artificial neural network in predicting student performance is characterized by the application of Artificial Intelligence techniques. It seems natural to use analogies when making decision or prediction, as by definition they contain information about how people have behaved in similar situation in the past. According to Kokinov (2003), we may explain human behaviour by assuring that decisions are made by analogy (logic form reasoning) with previous experiences, hence students early performance in tertiary institutions can be used to predict the similarities or otherwise of the same students during their specific period of training. In view of the fact that students play important part in the university customs, student performance has long been a key issue for all kinds of educational organizations.

So far, the adoption of information technology makes student's information management easy and efficient (Zhang and Patuwo, 1998). Advising students on their class performance and motivating them in order to continue to improve their performance is an integral part of every instruction. The mechanisms that would achieve the above aim required a method competent to accurately predicting student achievement as early as possible. According to Lykourentzou *et al*., (2009), learners' performance prediction can help to identify the weak learners at early stages and properly assist them to cope with their study. Based on this fact, Huang (2011) submitted that a valid model for predicting student academic performance is needed and will be helpful in designing and implementing pedagogical and instructional interventions, in order to enhance teaching and learning. Predicting future class performance is a difficult process and every attempt to automate the task must overcome a number of challenges. To address these constraints, Karamouzis (2000) reported the development of a prototype electronic system called FIG (Final Grade). The study evaluated FIG's predictive power.

Several other researchers have applied Artificial Intelligence techniques to predict student performance before or after being admitted into universities. For instance, Lykourentzou *et al.*, (2009) predicted the final grades of students in e-learning courses with multiple feed-forward neural networks using self-designed multiple-choice test data of students of as input. In a similar manner, Oladokun, Adebanjo and Charles-Owaba (2008) applied multilayer perceptron neural network for predicting the possible performance of candidates being considered for admission into Engineering Course of the University of Ibadan and concluded that ANN model is able to correctly estimate the performance of more than 70% of prospective students. However, Abass *et al.*, (2011) applied another technique of Artificial Intelligence (AI) i.e., case-base reasoning (CBR) to predict student academic performance based on the previous datasets using 20 students in the Department of Computer Science, TASUED as the study domain. The high correlation coefficient observed between the actual student CGPA and the CBR prediction also proved the efficiency of AI techniques in this type of assignment.

Hence this study is to tackle the problem of academic failure by seeking ways to make the process more effective, efficient and reliable. Specifically the study seeks to investigate the possibility of using neural network model to forecast the performance of a student before graduating the student.

REVIEW OF LITERATURE

Substantial studies based on neural networks have been conducted on data from schools, colleges, and distance-education courses, aiming at predict student achievement. Junemann, Lagos, and Arriagada (2007) employed neural networks to estimate future student performance based on students' family, social, and wealth characteristics as inputs. The aforementioned study focused on predicting the learning outcomes of 15-year-old students on reading, mathematics and science courses. Wang and Mitrovic (2002) applied neural networks to forecast the number of errors that a trainee will make using problem-specific attributes and the trainees' current level as input variables. This estimation was, however, used on a single examination to optimize the choice of the problems the student was asked to solve subsequently in the same examination.

The prediction of academic outcomes, using college-student data, was studied by Cripps (1996). In this work, various demographic features (age, gender, and race) as well as college entrance examination results were used to train a neural network in order to predict student programme completion final grade. In the study, no new data on what occurred throughout student progress were used to dynamically improve this estimation.

However, Sheel *et al.*, (2001) compared neural networks and statistical models to cluster students into two distinct groups using a single mathematical placement test. Student data obtained from distance education were used by Kalles and Pierrakeas (2006) and Kotsiantis, Pierrakeas, and Pintelas (2004) to predict success or failure of students in final exams using multiple approaches including neural networks. The data covered demographic information, homework grades, and plenary class meeting attendance levels. Lykourentzou *et al.*, (2009) estimated early, the final grades of students in e-learning courses using multiple feed-forward neural networks and multiple-choice test data administered to the students of National Technical University, Athens, Greece as input. In similar manner, Oladokun, Adebanjo and Charles-Owaba (2008) applied multilayer perceptron network to predict the likely performance of candidates being considered for admission into Engineering Course of the University of Ibadan using various influencing factors such as O'level scores, matriculation exam scores, age on admission, parental educational background and others as input variables. The results indicated that neural network was able to correctly estimate the performance of more than 70% of prospective students. Apart from neural networks, other techniques of AI have been applied on data from educational sector, most of which aiming at predicting learning achievement. Adeleke, Ruzaini and Hongwu (2013) predicted the risk status of newly admitted students of Computer Science of Universiti Malaysia, Pohang using fuzzy logic system and predictive factors such as secondary school results strength, number of sittings, mode of entry, parent literacy and others. Results of the study suggested that AI techniques are capable to handle uncertainty that is associated with students' performance, so as to determine their strength and weaknesses. Four machine learning models: learning tree, bagging, random forest and boosting have been used to predict academic performance of students by Hudson and Christiano (2014) with results showing high prediction accuracy. In Hudson, Christiano and Diego (2014), random forest was integrated with psychometrics to predict learning achievements of high school students. In this work, neural network is used to estimate students' final grade in the university.

Inspired by the structure of the brain, a neural network consists of a set of highly interconnected artificial neural units, called Processing Elements (PE). Each unit is designed to mimic its biological counterpart. Each accepts a weighted set of inputs and responds with an output. Neural Networks address problem that are often not easy for traditional computers to solve, typically, speech and pattern recognition, time-series forecasting, scheduling, regression and function approximation. An Artificial Neural Network is the area of science that deals with methods and system for information processing using neural network is called neurocomputation (Adepoju *et al.*, 2007).

The origin of the neural network can be

traced to 1940s when two researchers, Warren McCulloch and Walter Pitts, tried to build a model to replicate how biological neurons work. Though the focus of this research was on the anatomy of the brain, it turns out that this model introduced a new approach for solving technical problems outside biological science.

Feed-Forward Neural Networks

A feed-forward neural network (FFNN), shown in Fig.1, is a backpropgation network which allows signal to flow one way only, from input to output layer. There is no feedback mechanism (loops). FFNNs tend to be straight forward network that connect inputs with outputs. In addition, FFNN consists of one or more hidden layers of neurons in which neural connections, called *synapses*, do not form a directed cycle (Haykin, 1999; Lykourentzou *et al.*, 2009; Usman and Adenubi, 2013). The information moves only forward, from the input to the output nodes. During its learning phase, the network is presented with a set of examples called the training set. Each example consists of an input vector and the corresponding output vector. This type of learning is known as the supervised learning. The goal of the FFNN training is to minimize the sum of square error (SSE) or more recently, mean square error (MSE) between its actual and target outputs, by adjusting the network synaptic weights and neuron biases. More particularly, these network parameters are adjusted based on the backpropagation algorithm discussed later.

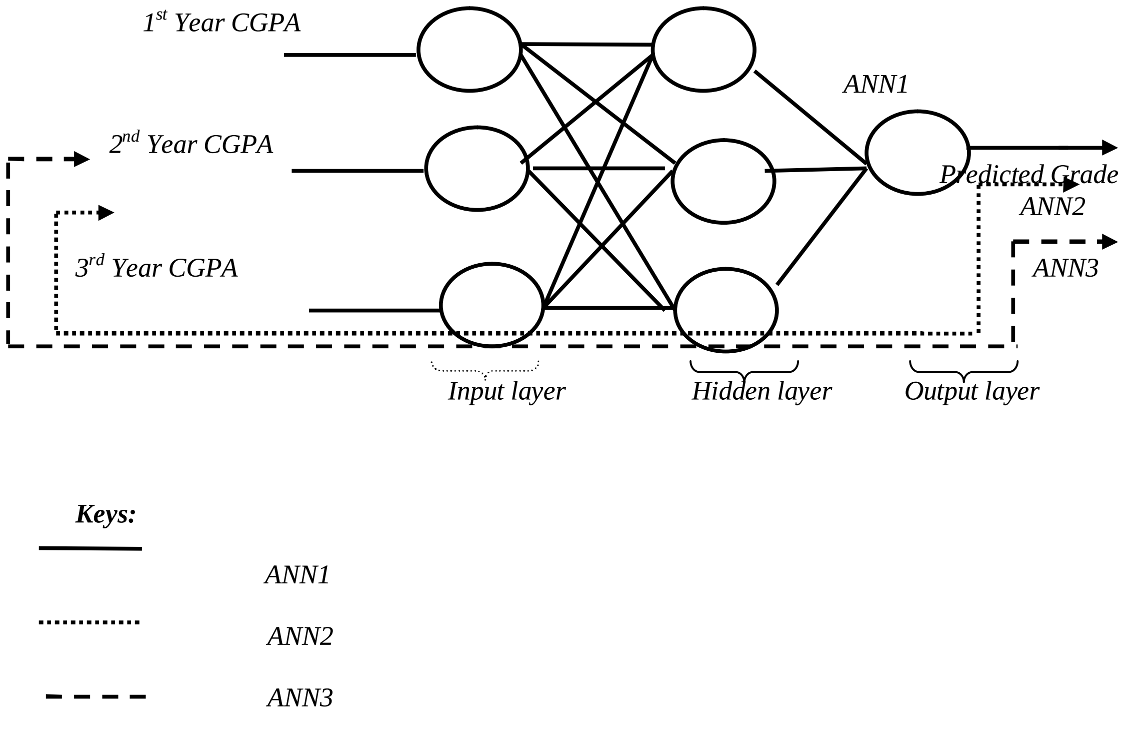

Fig.1: A simple FFNN

METHODOLOGY

Method of Grading Student's Cumulative Grade Point Aggregate (CGPA)

There are several grading systems depending on the culture of a University. Some universities like University of Ibadan use 7-Grading scale, others use 5-Grading scale. In this research, the method of grading the ANN simulated output is based on the current grading system used by the case study and is summarized in Table 1.

Table 1: Output Data Transformation for ANNs

S/N	Cumulative Grade Point Average (CGPA)	Class of Degree	Classification of ANN Output
1	4.50-5.00	First Class	Distinction
2	3.50-4.49	Second Class (Upper Division)	Very Good
3	2.40-3.49	Second Class (Lower Division)	Good
4	1.50-2.39	Third Class	Fair
5	1.00-1.49	Pass	Fail
6	Below 1.00	Fail	Fail

Therefore, artificial neural network (ANN) model used in this study is aimed at achieving predictive means of the final grade likely to be the results of the CGPA computation that can be classified into the divisions, summarized in the table 1.

Methodology for predicting students' academic performance with ANN

The methodology for predicting students' performance and designing a tool for performing this task is clearly divided into ten (10) recognizable steps discussed below. The flowchart in Fig.2 captured the first eight (8) steps while step 9 and 10 are elaborated further in Fig.3.

1. **Identify the purpose of performance prediction:** The purpose of predicting student's academic performance is to enable stakeholders in university provide better educational services as well as customize assistance according to students' predicted level of performance.

2. Data Collection, Description and Representation: The data used in this research work was sourced from Department of Computer and Information Science, Tai Solarin University of Education, Ogun State, Nigeria. This data consists of Cumulative Grade Point Aggregate (CGPA) of 60 randomly selected students who have completed their academic programme with the university. The samples are in the age range (17-25 years), cut-across all intelligent levels, and are exposed to the same learning experience of the case study. This data was carefully studied and synchronized into a form suitable for coding within the framework of ANN modelling. The data are represented below:

i. The Input Variable: The input variables are the dataset used as input to the ANN models, as well as the target values used to compare the predicted values against the reality. The first three (3) sessions CGPA values were used as inputs while the final CGPA values served as target values.

ii. The Output Variable: The output variable represents the final CGPA of a student on graduation. The classification of output variable is shown in Table 1.

3. **Creating the Network using appropriate tool and software:** The network created in this work is a typical feed-forward neural network (FFNN) described in above. Three feed-forward neural networks labeled ANN1, ANN2 and ANN3 respectively were created and implemented to predict student final CGPA, by approximating the function that maps students' GCPA in their first three sessions to their final CGPA. For the implementation of the neural networks, the ***nntool*** of MATLAB (R2008a) software was used.

4. Determination of the network topology, training function, adjustment of synaptic weight and other parameters: The network topology describes the arrangement nodes of the neural network. Choosing the topology of the neural network is a difficult decision (Emuoyibofarhe *et al.*, 2003; Oladokun *et al.*, 2006; Usman and Adenubi, 2013). The network topologies available for are numerous; each with its inherent strengths and weaknesses. For example, some networks trade off speed for accuracy, while some are capable of handling static variables and not continuous ones. Hence, in order to arrive at an appropriate network topology, various options were considered. Due to the nature of this data, which is static and not sufficiently large to enable the use of complex topologies, the FFNN (3-3-1) was selected (Fig.1). By this topology, there are three neurons at the input layer, three neurons at the hidden layer and only one neuron at the output layer. The experiment was also conducted in three phases, selecting three different training functions (one for each neural network) namely, ***Trainlm, Traingdm*** and ***Traingda***. The objective is to determine the best training function that can adequately minimize the square errors in the responses of the neural network. ***Trainlm*** function updates weight and bias values of FFNN according to Levenberg-Marquardt optimization, whereas ***Traingdm*** and ***Traingda*** functions update FFNN weight and bias values according to the Gradient Descent optimization. ***Taringdm*** is a gradient descent with momentum, whereas ***Taringda*** is a gradient descent with adaptive learning rate.

5. Training the network: Trainings were implemented in the MATLAB Neural Network Toolbox. During the training, for each one of the three series of experiment, 1000 training iterations were maintained and the best trained network was kept. Computation time for the training phase of each network did not exceed 25 minutes. Table 2 shows the characteristic of ANN models before training session.

6. Testing, validating and simulating results to new inputs: After training and cross validation, the networks were tested to simulate new responses. The results of this phase are presented in the section. During experiments, 65% of the entire dataset was used for the training, 20% for testing and the remaining 15% for cross validation.

7. Evaluating the predicted outputs or responses: The results were evaluated to determine the level of prediction accuracy. Criteria for evaluating the output include Computational time, means square errors (MSE) and percentage accuracy.

Parameters	Values
Training Algorithms	Levenberg-Marquardt, Gradient Descent
Training functions	TRAINLM, TRAINGDM, TRAINGDA
Performance function	Mean square error (MSE)
Adaptation learning function	LEARNGDM
Number of layers	3
Number of Hidden layer nodes	10
Transfer function	TANSIG
Epoch	1000 Iterations

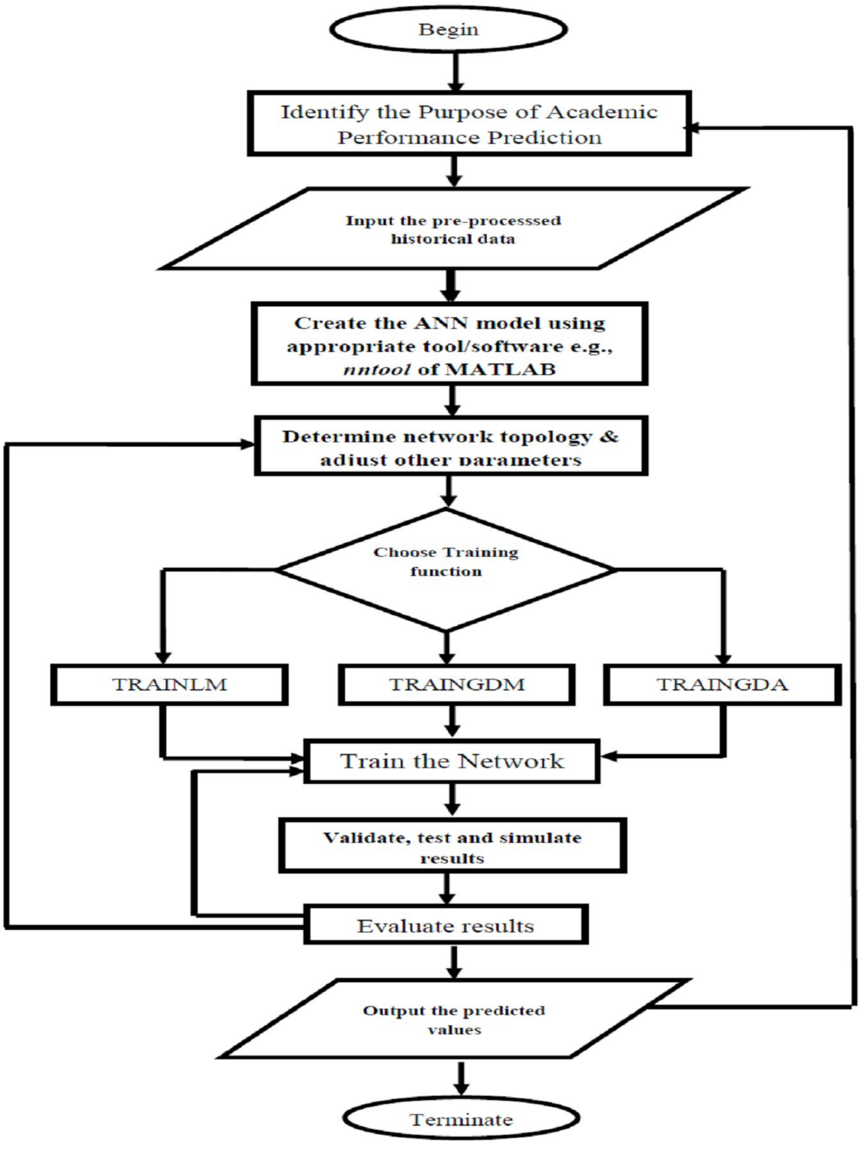

Fig.2: Methodology for predicting students' performance

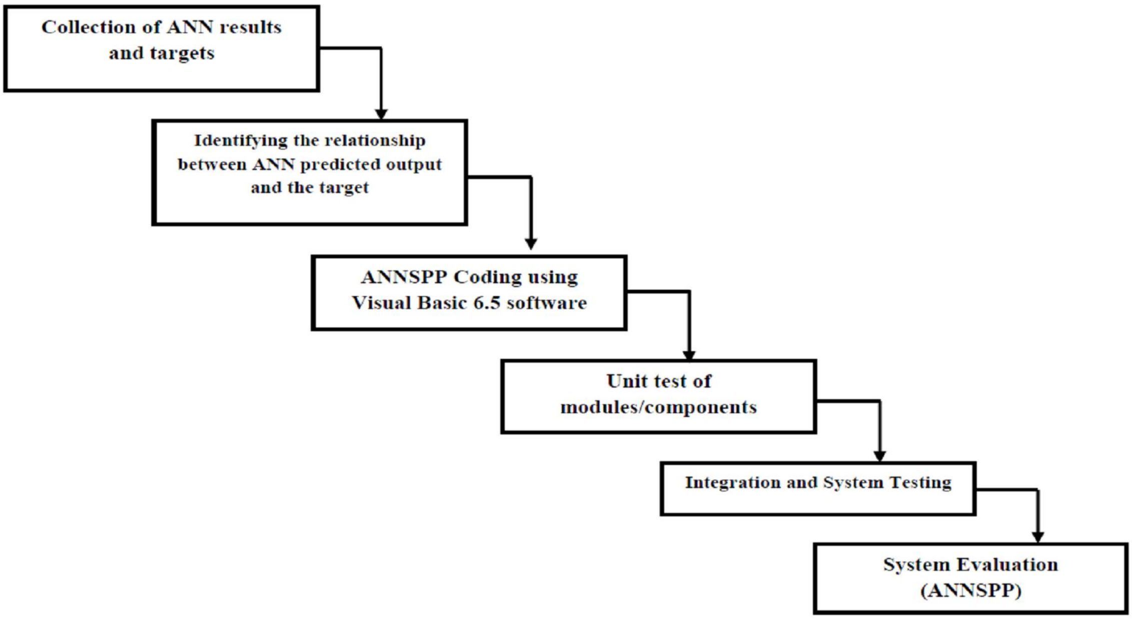

Fig.3: Waterfall representation of a process for developing ANNSPP

Method of developing Artificial Neural Network for students' performance prediction

This section explains the last two steps of the methodology for predicting students' final grades using ANN shown in Fig.2. The method for developing Artificial Neural Network for Students' Performance Prediction (ANNSPP) follows strictly the ***Waterfall Model*** of process development approach. The steps are as follows;

1. **Assemble results predicted by ANN and target final grades:** Immediately after the ANN experiments and all necessary computation, the predicted outputs generated by the best trained network and the target outputs were collected and prepared for coding a system for performing students' performance prediction.

2. Determine the relationship between ANN results and target outputs: The relationship between these outputs were identified and analyzed to a form that makes coding easy in the context of Visual Basic. Visual Basic is an event-driven tool that allows the developer to develop Windows application and has the ability to handle fixed and dynamic variable (Usman and Adenubi, 2013).

3. Use the relationship identified to code ANNSPP using appropriate software: As established above, the codes for developing ANNSPP were written in Visual Basic.

4. Perform unit test of individual module/component of ANNSPP: Several modules or components of ANNSPP are coded and tested separately to ensure they perform the required functions integration. This is done in terms of *"form creation"* in Visual Basic environment.

5. Integrate the modules or components and conduct general system testing: Several modules of ANNSPP are integrated and tested to ensure they function as a unit. The output of this phase is a system which is capable of predicting the student's final grade in the university.

6. Evaluate the performance of ANNSPP: The performance of ANNSPP was analyzed and evaluated and the results shown in the next section. Fig.4 depicts the possible interaction between the user and ANNSPP interface. Arrow that moves from the user to the system indicates user making prediction request by supply the necessary information to the system while the one that comes out from the system indicates the system's response to the request made. The third arrow that moves from the administrator into the system indicates that the administrator provides the necessary backend assistant.

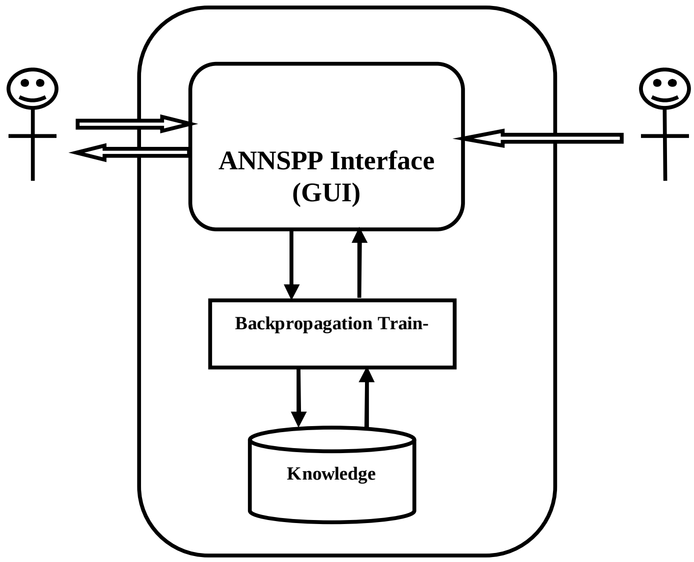

Fig.4: Possible Interaction between User and ANNSPP

Backpropagation algorithm for training FFNN

Backpropagation involves evaluating the error between the predicted outputs and the known values of the training data set (Folorunso *et al.*, 2010; Yegnanarayana, 2012). The objective of training FFNN is to minimize a cost function specifically defined as the mean square error (MSE) between its actual and target outputs, by adjusting the synaptic weights and nodes biases. The error signal at the output of neuron ***j*** at iteration ***n*** is defined according to Haykin (1999) as:

$$\epsilon_j(n) = d_j(n) - y_j(n) \tag{1}$$

The summation of instantaneous value of the error energy for node ***j*** is defined as:

$$E = \frac{1}{2}\sum_{j \in C} \varepsilon^2(n) \tag{2}$$

The induced local field $V_j(n)$ produced at the input of the activation function associated with node j is defined by:

$$V_j(n) = \sum_{i=1}^{m} w_{ji}(n) y_i(n) \quad (3)$$

where

$$V_j(n) = \varphi_j(V(n)) \quad (4)$$

Note: $w_{ji}(n)$ = synaptic weight, $y_i(n)$ = functional signal

Differentiating *eqn(2)* with respect to $\varepsilon_j(n)$, we get:

$$\frac{\partial E}{\partial \varepsilon_j(n)} = \varepsilon_j(n) \quad (5)$$

Differentiating *eqn(3)* with respect to $y_i(n)$, we obtain:

$$\frac{\partial \varepsilon_j(n)}{\partial y_i(n)} = -1 \quad (6)$$

Also, differentiating *eqn(4)* with respect to , we get;

$$\frac{\partial y_i(n)}{\partial V_j(n)} = \varphi_j'(V_j(n)) \quad (7)$$

Finally, differentiating *eqn(3)* with respect to , yields;

$$\frac{\partial V_j(n)}{\partial w_{ji}(n)} = y_i(n) \quad (8)$$

In a similar manner like *least mean square (LMS)* algorithm, backpropagation algorithm applies a correction $\Delta w_{ji}(n)$ to the synaptic weight $w_{ji}(n)$, which is proportional to the partial derivative i.e. $\frac{\partial E}{\partial w_{ji}(n)}$

$$\Delta w_{ji}(n) = \frac{\partial E}{\partial w_{ji}(n)} = \frac{\partial E}{\partial \varepsilon_j(n)} \cdot \frac{\partial \varepsilon_j(n)}{\partial y_i(n)} \cdot \frac{\partial y_i(n)}{\partial V_j(n)} \cdot \frac{\partial V_j(n)}{\partial w_{ji}(n)} \quad (9)$$

By using *eqn(6)* to *eqn(8)* in *eqn(9)*, we obtain;

$$\Delta w_{ji}(n) = \frac{\partial E}{\partial w_{ji}(n)} = -\varepsilon(n)\varphi_j'\left(V_j(n)\right) y_i(n) \quad (10)$$

Therefore, the correction $\Delta w_{ji}(n)$ applied to $w_{ji}(n)$ is defined by the delta rule:

$$\Delta w_{ji}(n) = -\eta \frac{\partial E}{\partial w_{ji}(n)} = w_{ji}^{new} - w_{ji}^{old} \quad (11)$$

where η is the learning rate parameter, E is the error parameter and the minus sign signifies *"gradient descent"* in weight space.

RESULTS AND DISCUSSION

As established in the previous sections, three feed-forward neural networks, namely ANN1, ANN2 and ANN3 were implemented to predict student final grade in a University, by approximating the function that maps students' CGPA in the first three sessions to their final CGPA. Table 3 presents the simulated results obtained from the experiment. According to the table, all numerical values are given in 2d.p. which stipulates the standard way of expressing student's CGPA in the University. The performance plot (Fig.5) of *Trainlm* function showed that, it is efficient at minimizing the mean square error (MSE) between ANN1 responses and the actual students' final grade while the regression plot of Fig.6 showed that student final grade prediction is possible at a correlation coefficient (R) value equals to 0.95146. ANN1 took 8 sec to complete the training in this experiment at epoch equals 14 iterations. From Table 4, ANN1 correctly predicted the final grades of 52 students out of 60 that were used in the experiment with percentage accuracy of 86.67%.

The results from the training of a FFNN (ANN2) with *Traingdm* function showed a slight improvement in the performance of neural network, as the model was able to predict the final grade of 53 students out of 60 students that were used in the experiment, thereby increasing the percentage accuracy of neural network model to 88.33%. From the graph in Figs. 7 and 8, one can deduce that the training function is slightly inefficient at minimizing the performance criterion between its responses and the actual students' final grades though, the correlation coefficient (R) value is high (0.965469). Another economic disadvantage of this model is that, the model spent very long period of time (24.27 minutes) for its training session, leading to the usage of maximum epoch set. This implies that the training was completed at epoch equals to 1000 iterations.

Finally, the result from the training of FFNN (ANN3) with *Traingda* function was found to be the best when compared with the known values. This is clear from the classification of CGPA shown in the last column of Table 3. From above analysis, one can conclude that this training function is more efficient than the previous ones. The model (ANN3) generated from it is very efficient at minimizing the mean square error between its simulated responses and the actual students' final grades, evident from the performance plot of Fig.9. With this training function, ANN3 was able to correctly predict the final grade of 55 students out of 60 students used in the experiment, thereby, further increasing the percentage accuracy to 91.67%. The regression plot of Fig.10 showed that student final grade prediction is possible at a correlation coefficient (R) value equals to 0.990930. With this function, FFNN took 2 seconds to complete the training with epoch value of 68 iterations. Obviously, the predicted outputs of this model was used in the development and coding ANNSPP. For better understanding and simplicity, the results displayed in Table 3 have been summarized in Table 4.

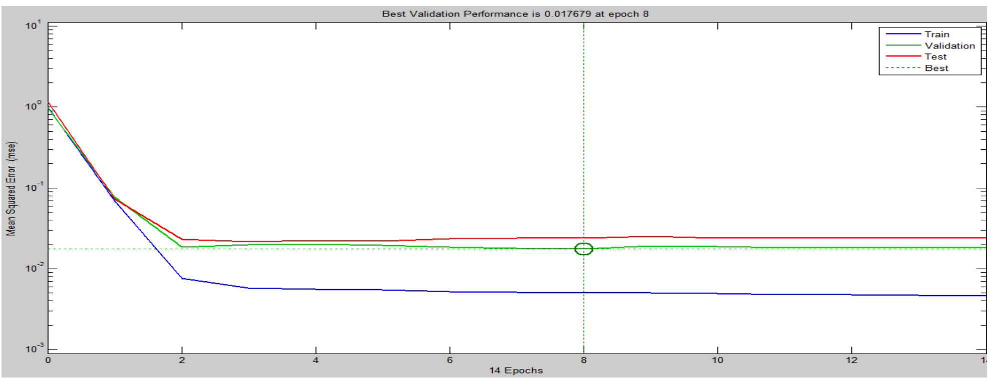

Fig.5: ANN1 Training (TRAINLM)

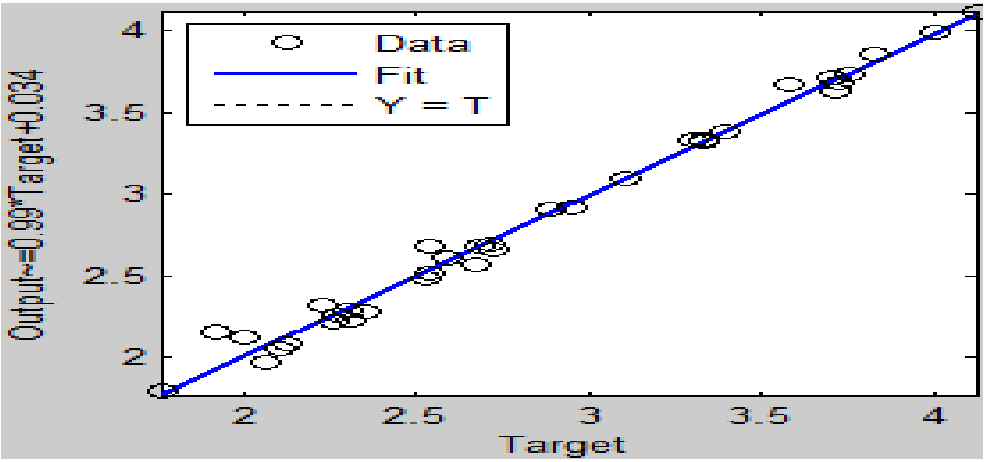

Fig.6: Testing ANN1 (Training = 0.951459)

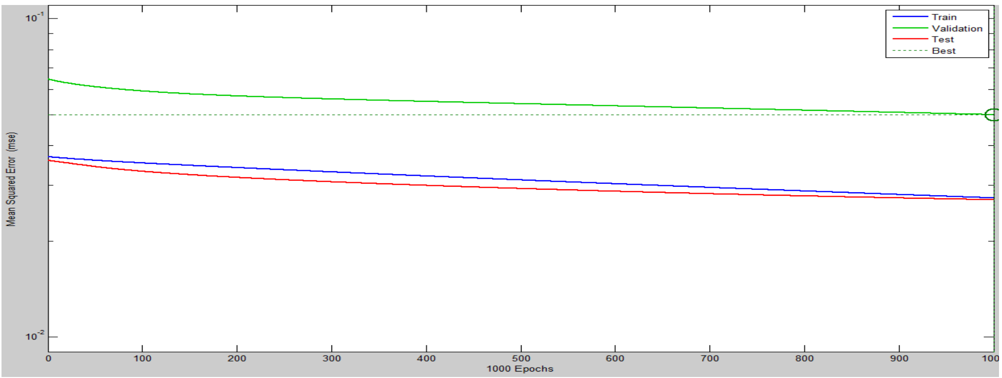

Fig.7: ANN2 Training (TRAINGDM)

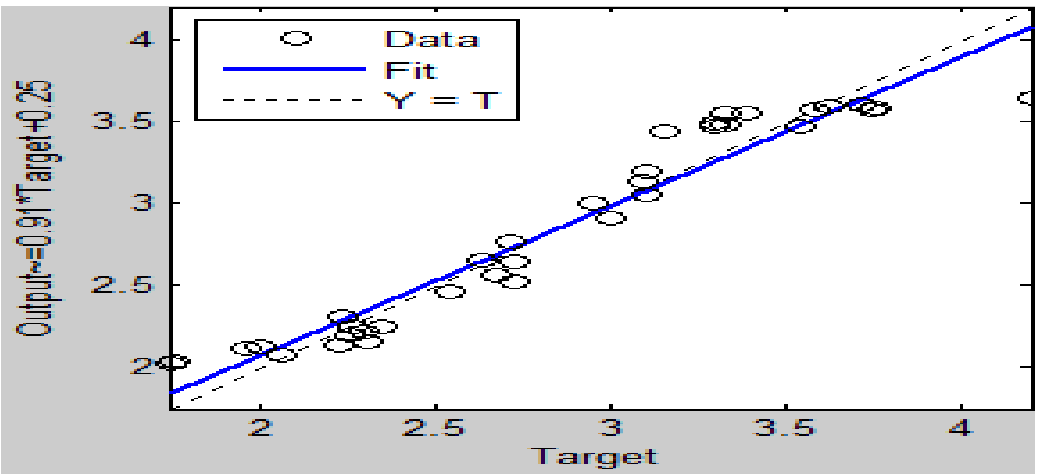

Fig.8: Testing ANN2 (Training = 0.965469)

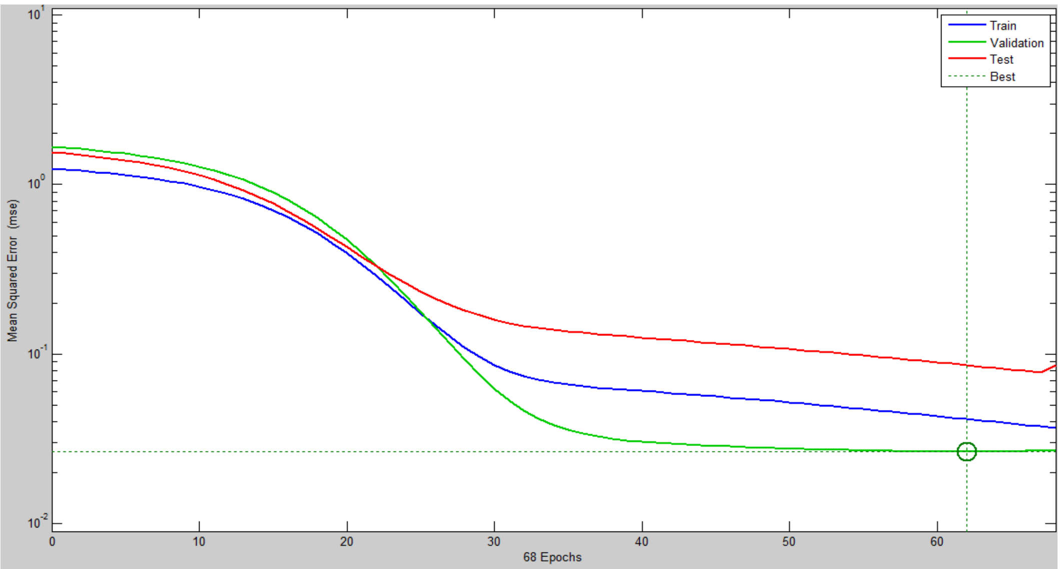

Fig. 9: ANN2 Training (TRAINGDA)

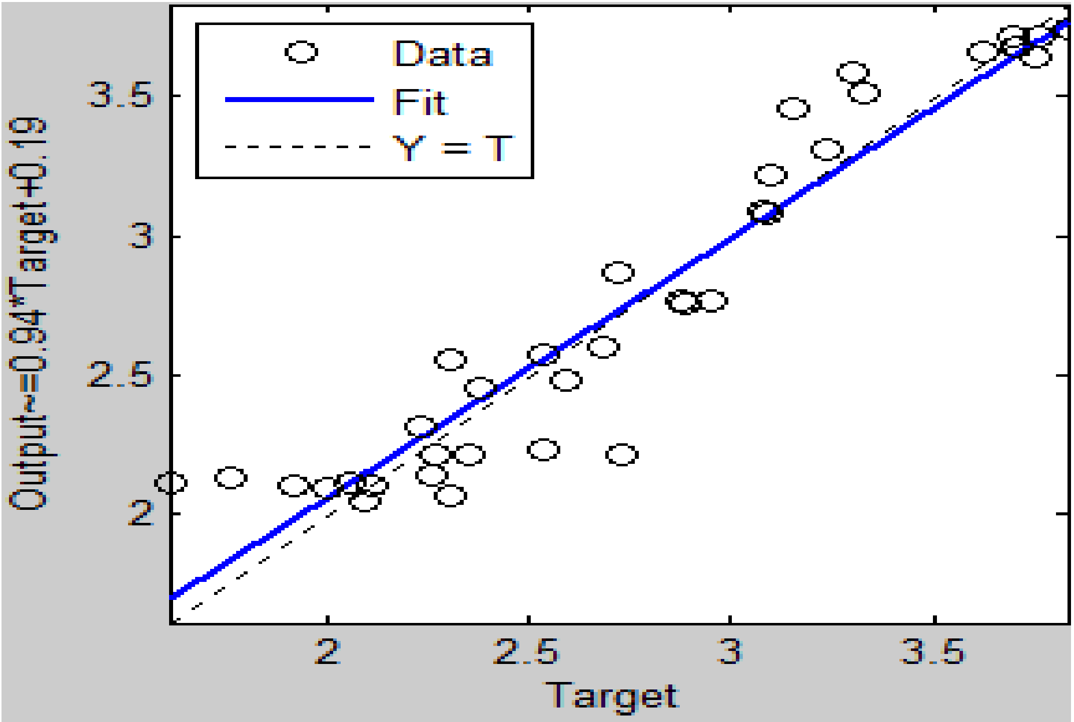

Fig.10: Testing ANN3 (Training = 0.990930)

Table 3: Comparison of ANNs Results with known Target Values

Students ID	Target Final CGPA	Output Classification	ANN1 Outputs Trainlm	Output Classifiction	ANN2 Outputs Traingdm	Output Classfication	ANN3 Outputs Traingda	Output Classification
Student1	2.38	Fair	2.55	Good	2.48	Good	2.39	Fair
student2	4.20	Very Good	3.99	Very Good	3.65	Very Good	4.11	Very Good
Student3	3.08	Good	3.03	Good	3.18	Good	3.09	Good
Student4	3.39	Good	3.35	Good	3.55	Very Good	3.47	Good
Student5	2.30	Fair	2.25	Fair	2.23	Fair	2.33	Fair
Student6	4.12	Very Good	3.95	Very Good	3.66	Very Good	4.10	Very Good
Student7	2.11	Fair	2.02	Fair	2.10	Fair	2.13	Fair
Student8	2.88	Good	2.86	Good	2.79	Good	2.86	Good
Student9	2.68	Good	2.68	Good	2.62	Good	2.70	Good
Student10	3.30	Good	3.73	Very Good	3.60	Very Good	3.41	Good
Student11	2.53	Good	2.53	Good	2.35	Fair	2.52	Good
Student12	1.83	Fair	1.83	Fair	2.04	Fair	2.14	Fair
Student13	2.35	Fair	2.24	Fair	2.24	Fair	2.21	Fair
Student14	2.54	Good	2.53	Good	2.46	Good	2.49	Fair
Student15	3.30	Good	3.40	Good	3.47	Good	3.46	Good
Student16	2.31	Fair	2.47	Good	2.49	Good	2.54	Good
Student17	2.23	Fair	2.29	Fair	2.30	Fair	2.31	Fair
Student18	3.83	Very Good	3.87	Very Good	3.57	Very Good	3.81	Very Good
Student19	3.15	Good	3.16	Good	3.44	Good	3.22	Good
Student20	3.76	Very Good	3.75	Very Good	3.58	Very Good	3.74	Very Good
Student21	2.89	Good	3.59	Very Good	2.94	Good	2.74	Good
Student22	2.63	Good	2.66	Good	2.65	Good	2.70	Good
Student23	2.59	Good	2.58	Good	2.59	Good	2.59	Fair
Student24	3.30	Good	3.39	Good	3.49	Good	3.42	Good
Student25	3.54	Good	3.45	Good	3.47	Good	3.43	Good
Student26	3.58	Good	3.67	Very Good	3.58	Very Good	3.72	Very Good
Student27	2.31	Fair	2.21	Fair	2.15	Fair	2.08	Fair
Student28	2.06	Fair	1.94	Fair	2.08	Fair	2.14	Fair
Student29	3.33	Good	3.31	Good	3.48	Good	3.31	Good
Student30	3.10	Good	3.12	Good	3.20	Good	3.22	Good
Student31	2.22	Fair	2.11	Fair	2.13	Fair	2.13	Fair
Student32	3.09	Good	3.07	Good	3.13	Good	3.08	Good
Student33	4.00	Very Good	3.91	Very Good	3.62	Very Good	4.22	Very Good
Student34	3.00	Good	2.92	Good	2.91	Good	2.87	Good
Student35	3.33	Good	3.20	Good	3.55	Very Good	3.39	Good

Student36	3.71	Very Good	3.63	Very Good	3.56	Very Good	3.70	Very Good
Student37	2.00	Fair	2.12	Fair	2.13	Fair	2.11	Fair
Student38	2.27	Fair	2.51	Good	2.24	Fair	2.29	Fair
Student39	1.96	Fair	2.04	Fair	2.11	Fair	1.92	Fair
Student40	2.67	Good	2.54	Good	2.56	Good	2.66	Good
Student41	3.75	Very Good	3.73	Very Good	3.57	Very Good	3.77	Very Good
Student42	3.70	Very Good	3.65	Very Good	3.61	Very Good	3.74	Very Good
Student43	3.24	Good	3.27	Good	3.32	Good	3.32	Good
Student44	1.76	Fair	1.81	Fair	2.03	Fair	1.80	Fair
Student45	3.62	Very Good	3.63	Very Good	3.60	Very Good	3.68	Very Good
Student46	2.73	Good	2.57	Good	2.52	Good	2.80	Good
Student47	1.61	Fair	1.76	Fair	2.01	Fair	1.62	Fair
Student48	2.26	Fair	2.22	Fair	2.19	Fair	2.27	Fair
Student49	2.71	Good	2.67	Good	2.77	Good	2.74	Good
Student50	1.92	Fair	2.15	Fair	2.16	Fair	1.89	Fair
Student51	3.10	Good	2.93	Good	3.05	Good	3.15	Good
Student52	2.30	Fair	2.66	Good	2.21	Fair	2.30	Fair
Student53	1.74	Fair	1.78	Fair	2.02	Fair	1.73	Fair
Student54	2.95	Good	2.88	Good	3.00	Good	2.93	Good
Student55	3.24	Good	3.16	Good	3.26	Good	3.22	Good
Student56	2.72	Good	2.65	Good	2.65	Good	2.86	Good
Student57	3.72	Very Good	3.66	Very Good	3.61	Very Good	3.75	Very Good
Student58	2.13	Fair	2.07	Fair	2.09	Fair	2.16	Fair
Student59	2.09	Fair	2.08	Fair	2.09	Fair	2.07	Fair
Student60	2.54	Good	3.66	Very Good	2.65	Good	2.55	Good

Student36	3.71	Very Good	3.63	Very Good
Student37	2.00	Fair	2.12	Fair
Student38	2.27	Fair	2.51	Good
Student39	1.96	Fair	2.04	Fair
Student40	2.67	Good	2.54	Good
Student41	3.75	Very Good	3.73	Very Good
Student42	3.70	Very Good	3.65	Very Good
Student43	3.24	Good	3.27	Good
Student44	1.76	Fair	1.81	Fair
Student45	3.62	Very Good	3.63	Very Good
Student46	2.73	Good	2.57	Good
Student47	1.61	Fair	1.76	Fair
Student48	2.26	Fair	2.22	Fair
Student49	2.71	Good	2.67	Good
Student50	1.92	Fair	2.15	Fair
Student51	3.10	Good	2.93	Good

Table 4: Summary of ANNs Performance

	ANN1		ANN2		ANN3	
Summary	Original Grade	Predicted Grade	Original Grade	Predicted Grade	Original Grade	Predicted Grade
Distinction	0	0	0	0	0	0
Very Good	10	14	10	14	10	11
Good	29	29	29	26	29	27
Fair	21	16	21	20	21	22
Fail	0	0	0	0	0	0
Total Samples	60	60	60	60	60	60
Correct		52		53		55
Incorrect		8		7		5
% Accuracy of the prediction		86.67%		88.33%		91.67%

The ANNSPP designed was then implemented to predict the final grades of some students when supplied with unknown CGPA values and the example of the predicted results are shown in the interface displayed in Fig.11.

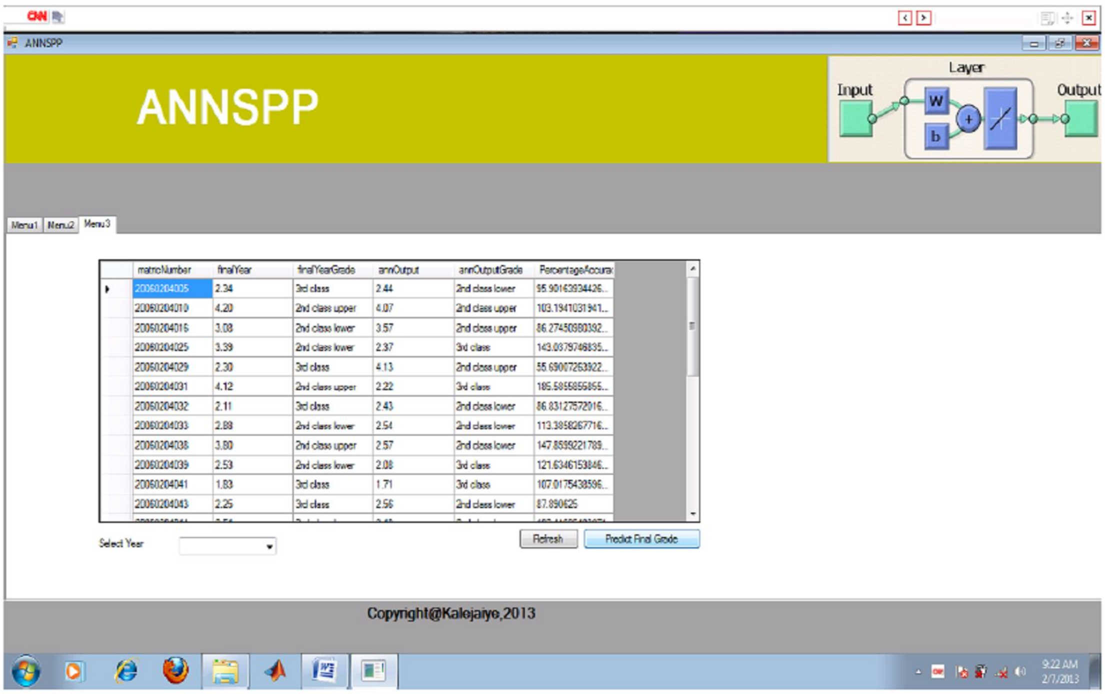

Fig.11: Prediction Using ANNSPP

From the various tests performed on the results of the training, validation and test results, it is confirmed that Artificial Neural Network (ANN) performs quite impressible in estimating the Final Grades of students in University. Both the percentage accuracies and correlation coefficients are good evidences of the fact that, given appropriate dataset at its disposal, the ANNSPP designed can guarantee students' learning outcome prediction accuracy and help the stakeholders to discover at early stage, students who have no tendency of doing well in tertiary institutions, thereby prevent continuous waste of scarce resources on such student, in addition to motivating the good ones. The study also corroborates earlier researches that have reported the effectiveness of ANN in regression and time-series forecast of learners' achievements at various level of education, examples of which are shown in Fig.12.

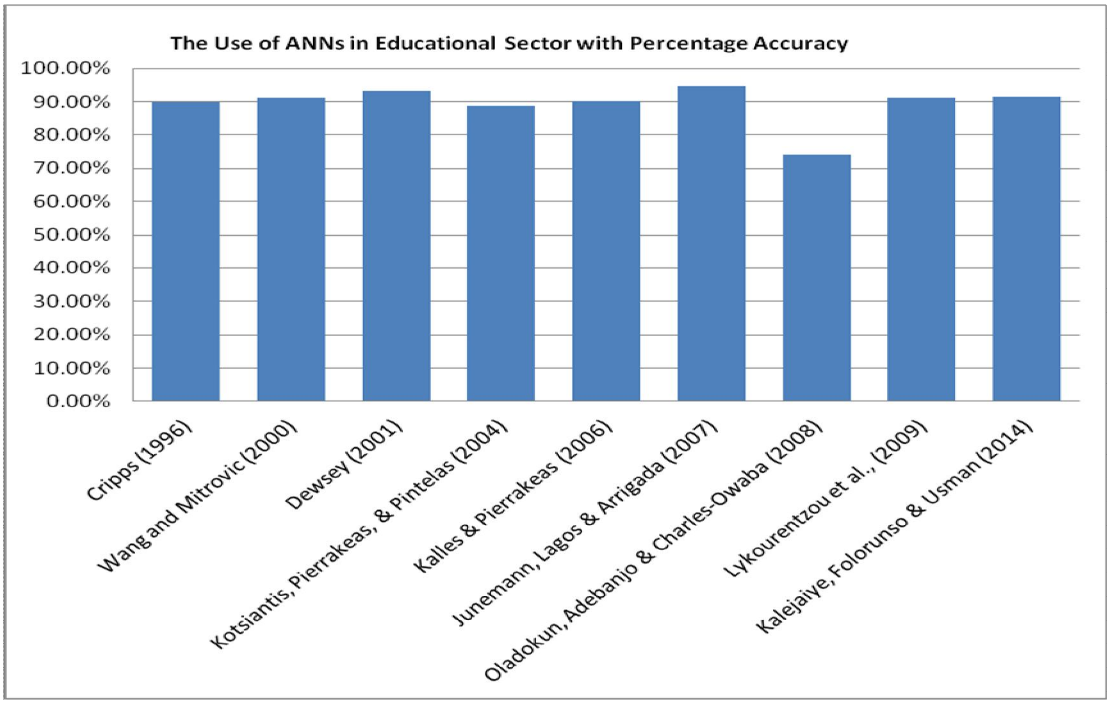

Fig.12: Comparison of Achievements of ANN models

CONCLUSION AND FUTURE RESEARCH

The result obtained from the study actually showed that the Artificial Neural Networks are capable to predict the performance of students in the university and can be used to develop a predictive tool. This is due to a little but not all that significance errors which exists between the training values and the ANN simulated values. Suffice to say that the ±0.3 experienced in the study can be reduced if the number of times the Cumulative Grade Point Aggregate (CGPA) values is used in the training session increases. From the results, it can be concluded that ANN is 91.7% efficient at predicting student final academic performance (Fig.12). The information technology has brought many innovations into the education sector, therefore, the result of the research can be used to formulate a policy that concerns students' learning outcomes. However, future investigation may focus on developing a new and more robust ANN prediction models while establishing database to collect large volume of students' performance data in Universities for further research on ANN abilities in forecasting the stochastic nature that is associated with such data.

REFERENCES

Abass,O., Oyekanlu ,E.A., Alaba, O.B, Longe. O.B., 2011. Forecasting Student Academic Performance using Case-Base Reasoning. *International conference on ICT for Africa*. 23-26th march, 2010 Otta Nigeria, P.105-112

Adepoju, G.A., Ogunjuyigbe, S.O.A., Alawode K.O., 2007. Application of Neural Network to Load Forecasting in Nigerian Electrical Power System". *The Pacific Journal of Science and Technology*.8(1):68-72 .

Adeleke, R.A., Ruzainin, A.A, Hongwu, Q., 2013. Risk Status Prediction and Modelling of Students' Academic Achievement-A Fuzzy Logic Approach. Research Inventy: International Journal of Engineering and Science, 3(11):7-14.

Cripps, A. 1996. Using artificial neural nets to predict academic performance. Paper presented at the 1996 *ACM Symposium on Applied Computing,* Philadelphia, PA.

Emuoyibofarhe, O.J., Reju, A.S., Bello, B.O., 2003. Application Of ANN To Optimal Control Of Blending Process In A Continuous Stirred Tank Reactor(CSTR)". Science Focus; *An International Journal Of Biological And Physical Science.* 3:84-91.

Folorunso, O., Akinwale, A.T., Asiribo, O.E., Adeyemo, T.A., 2010. Population Prediction
Using Artificial Neural Network. *Africa Journal of Mathematics and Computer Science Research,* 3(8):155-162.

Haykin, S.1999, Neural Networks: A Comprehensive Foundation (2nd Edition). *Macmillan College Publishing* Company, New York.

Huang, S., 2011. Predictive Modeling and Analysis of Student Academic Performance in an Engineering Dynamics Course. *All Graduate Theses and Dissertations,* School of Graduate Studies, Utah State University, DigitalCommon@USU, Paper 1086, http://digitalcommons.usu.edu/etd/1086.

Hudson, F.G., Christiano, M.A.G., 2014. Four Machine Learning Methods to Predict Academic Achievement of College Students: A Comparison study. *REVISTA ELECTRONICA DE PSICOLOGIA EDUCACAO E SAUDE, ANO4*, 1:68-101.

Hudson, F.G., Christiano, M.A.G., Diego, A., 2014. Predicting Academic Achievement of High-School students using Machine Learning. *Journal of Scientific Research: Psychology*, 5:2046-2057.

Junemann, M.A.P., Lagos, P.A.S., Arriagada, R.C., 2007. Neural networks to predict schooling failure/success. In *J. Mira & J.R. Álvarez (Eds.), Lecture Notes in Computer Science*, *Nature Inspired Problem-Solving Methods in Knowledge Engineering.* Berlin: *Springer*, 4528: 571–579.

Kalejaye, 2014. Artificial Neural Network Model for Predicting Students' Academic Performance. A thesis Submitted in Fulfilment of the requirement for the Degree of

Master of Education in the Department of Computer Science, Tai Solarin University of Education, Ijebu-Ode, Ogun State.

Kalles, D., Pierrakeas, C., 2006. Analyzing student performance in distance learning with genetic algorithms and decision trees. *Applied Artificial Intelligence*, 20, 655–674.

Karamouzis, S., 2000. Using Analogies to predict Student performance. *In Proc. IASTED's Artificial Intelligence and soft comp.* Banf, Canada, P.355-360.

Kokinov, B., 2003. Analogy in decision-making social interaction and emergent rationality. *Behavioral and Sciences*.26, 167-168.

Kotsiantis, S., Pierrakeas, C., Pintelas, P., 2004. Predicting students' performance in distance learning using machine learning techniques. *Applied Artificial Intelligence,* 18, 411–426.

Lykourentzou, I., Giannoukos, I., Mpardis, G., Nikolopoulos, V., Loumos, V., 2009. Early and Dynamic Student Achievement Prediction in E-Learning Courses Using Neural Networks. *Journal of the American Society for Information Science and Technology*, 60(2):372-380.

MATLAB, 2008. MATLAB Environment, from http://www.mathworks.com/products/matlab/

Oladokun, V.O., Charles-Owaba, O.E., Nwaozuru, C.S., 2006. An Application of Artificial Neural Network to Maintenance Management. *Journal of Industrial Engineering International*. South Tehran, 2(3):19-26.

Oladokun, V.O Adebanjo, A.T., Charles-Owaba, O.E., 2008. Predicting Students Academic Performance using Artificial Neural Network: A case study of an Engineering Course. *Pacific Journal of Science and Technology*, 9(1):72-79.

Sheel, S.J., Vrooman, D., Renner, R.S., Dawsey, S.K. 2001. A comparison of neural networks and classical discriminant analysis in predicting students' mathematics placement examination scores. InV.N.Alexandrov, J.J. Dongarra, B.A. Juliano, R.S. Renner, & C.J.K. Tan (Eds.), *Lecture Notes in Computer Science*, Computational Science—ICCS 2001. Berlin: *Springer* 2074: 952–957.

Usman, O.L., Adenubi, A.O., 2013. Artificial Neural Network (ANN) Model for Predicting Students' Academic Performance. *Journal of Science and Information Technology (JOSIT)*, 11(2):23-37.

Wang,T., Mitrovic, A. 2002. Using neural networks to predict students' performance. Paper presented at the *International Conference on Computers in Education*, Auckland, New Zealand.

Yegnanarayana, B., 2012. Artificial Neural Networks. *PHI Learning Private Limited, M-97, Connaught Circus*, New Delhi-110001, ISBN-978-81-203-1253-1.

Zhang, G., Patuwo, B.E 1998, Forecasting with Artificial Neural Networks: The State of the Art. *International Journal of Forecasting*. 14:35-62.

(Manuscript received: 3rd November, 2014; accepted: 19th February, 2015)