

Using Artificial Neural Networks to Diagnose Heart Disease

Ahmad Rufai

Sokoto State University, Sokoto
Nigeria
Department of Computer Science

U. S. Idriss

Federal University Gashua,
Nigeria
Department of Computer Science

Mahmood Umar

Sokoto State University, Sokoto
Nigeria
Department of Computer Science

ABSTRACT

This paper presents an application of Artificial Neural Networks (ANN) in predicting patient coronary heart disease status. Multilayer perceptron (MLP) which is a type of ANN architecture was used to develop the proposed model. Several experiments were carried out to determine the network optimal parameters. Overall, the optimised ANN system achieved a very high diagnosing accuracy of 92.2%, proving its usefulness in support of diagnosis process of coronary heart disease.

General Terms

Neural networks, Coronary heart disease, Multilayer perceptron (MLP).

Keywords

Artificial neural networks, Multilayer perceptron, Back-propagation algorithm, Coronary heart disease, Principal Component Analysis

1. INTRODUCTION

Coronary heart disease (CHD) is a term used to refer to the blockage of one's heart blood supply or interruption caused by a build-up of fatty substances in the coronary arteries of an individual. CHD is sometimes referred to as ischaemic heart disease and is seen as a major cause of death in the World [1]. CHD has been identified as a disease that is incurable but can only be managed to help patient live longer. Unlike other diseases such as yellow fever, cold and flu that can be detected by doctors through looking at patients eyes, Detection of CHD as described by the NHS involves series of stages. The first stage is a risk assessment for cardio-vascular diseases and heart attack which is carried out if a doctor suspects patient is at risk of developing the disease. During this stage, the doctor questions patient about his/her medical and family history, blood pressure and cholesterol level. Some doctors often ask questions about the patient's life style.

The second stage of diagnosing CHD is a series of laboratory test taking to confirm the presence of the disease. These tests are normally electrocardiogram (ECG), X-rays, blood test, coronary angiography, radionuclide test and magnetic resonance imaging (MRI). Since CHD is identified as incurable disease that can only be managed, it has become necessary for patients to know their CHD status in time. This can be achieved if an accurate diagnosis can be made promptly to patients suspected of being at risk. This implies reducing the diagnosis time. Thus a medical decision support system can be developed to assist doctors in this complex diagnosis process. In recent years, there has been an increasing interest into the use of artificial neural networks (ANNs) in the field of medical diagnosis. [2] described ANN

as a distributed network of computing elements that is modeled after the biological neural system and may be implemented as a computer software program. ANN has also been defined as a model of reasoning based on the human brain that is capable of learning through the use of experience to improve its performance [3]. Researchers have shown an increased interest into the application of ANN models in diagnosis and prediction of outcome of disease. As a result, a considerable amount of literature has been published on its application in disease diagnosis and prediction.[4] compared classification performances of three ANN models namely, multi-layer perceptron (MLP), radial basis function(RBF) and self-organizing feature maps (SOFM) with two other data mining techniques namely; logistic regression (LR), classification and regression tree (CART) in predicting the presence of coronary artery disease (CAD). However, the results of the experiments are somewhat confusing as they were presented in terms of ROC curves, Hierarchical Cluster Analysis (HCA) and Multidimensional Scaling (MDS) rather than the more popular percentage of accuracy approach. [5] employed artificial neural network to model the demographic characteristics of antenatal clinic attendees in South Africa and their HIV status. [6] employed artificial neural network to predict breast cancer survivability.[7] used both standard feedforward networks, Multilayer Perceptron (MLP), Radial Basis Function (RBF), and a hybrid network, Conic Section Function Neural Network (CSFNN) to diagnose hepatitis disease. The study concluded that the CSFNN was very successful in predicting hepatitis. In [8], an integrated model consisting of Genetic Algorithm (GA) and MLP was used for diagnosing of stroke disease. The GA was used to reduced the dimensionality of the networks input features. The results of the study show that the network's classification accuracy was better with the fewer input features.

The above studies all confirmed the wide used of ANN as decision support tool in the field of medical diagnosis with heart disease inclusive. However, far too little attention has been paid to the data understanding and data preparation stage of the above studies. This stage is an important stage in building predictive models. Thus, in this study, data pre-processing would be included so as to analyze how this will influence the accuracy of the proposed ANN model. Another area of concern that has not been given attention to in the previous studies was the elimination of correlated information in the sample data. Most of the real world data samples used to train ANNs contains correlated information caused by intersecting input instances. This tends to cause confusion over ANNs during learning process[9].Thus, Principal Components Analysis will be implemented to eliminate the correlated information within the training data and also analyze how this will influence the ANN accuracy.

2. ARTIFICIAL NEURAL NETWORKS

Artificial neural network consists of highly interconnected processors also known as neurons, similar to biological neurons in the brain. The neurons are connected with one another through weighted links passing signals from one neuron to another. The weights in ANN are similar to synapses of biological neural networks and they are the means of long-term memory in ANN, they give an idea of how important a neuron input is. ANN learns through constant adjustment of these weights, each neuron receives a number of inputs signals through its connection but only produces one output. Each time a neuron receives an input signals from its input links; it computes a new activation level and sends it as an output signal through the output link [3].

2.1 The Multilayer Perceptron

MLP is one of the most frequently used neural network architecture [10]. MLP is a supervised neural network architecture with one input layer, one or more hidden layer and one output layer. The input layer receives external inputs and redistributes them to the neurons in the hidden layer. The hidden layer is made up of computational neurons, these neurons detect the features hidden in the input data. The features detected by neurons in hidden layer are then used by output layer to determine the final output of the network. With one or two hidden layers, MLP can approximate virtually any input-output map [11]. An example of an MLP with two hidden layers and one output node is shown in Fig. 1.

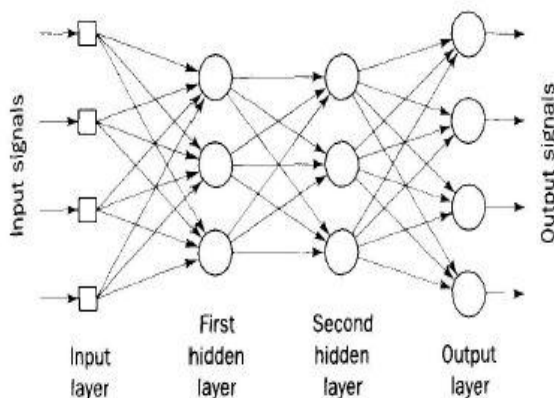


Fig 1 : MLP (Negnevitsky, 2002)

2.2 Back-Propagation Training Algorithm(BP)

Once the architecture of an MLP is determined, the weight of the network have to be computed through a training process. The training is based on a training pattern and desired network output. Back-Propagation algorithm (BP) is the most widely used training algorithm for MLP [12]. The algorithm was introduced by Bryson and Ho in 1969. It is one of gradient descent algorithms use to reduce the performance function through updating the networks weights by moving them along the negative gradients of the performance function. BP works in two phases. First, a training input pattern is presented to the network input layer. The input pattern is then propagated to other layers of the network until an output is produced by the output layer. If the actual output is different from the desired output, an error is calculated and then propagated backwards through the network from the output layer to the input layer

modifying the network weights through the process [3]. For full details of how the BP algorithm works, see [3].

BP can be implemented in two ways: first through an incremental training, where the gradient is computed and weights are updated after each input is applied to the network. Second, through batch mode, where all the inputs are applied to the network before the weights are updated. The gradients calculated at each training examples are added together to determine the change in the weights and biases. The incremental training is usually seen as more efficient and also faster for systems with more training samples [11].

When using BP algorithm, a network performance can be checked using sum squared error. This error is computed for each training pattern and the back-propagation algorithms attempts to minimize the average of this criterion over the set of training patterns [13]. This average is the Mean Squared Error (MSE) between the actual output a_k and the target output t_k which equation is given in formula below:

$$MSE = \frac{1}{2} \sum_{k=1}^{n_k} (t_k - a_k)^2$$

A network is considered to have converged when the value of MSE in an entire pass through all training sets or epoch is adequately small [3]. This value might differ from problem to problem.

Although the BP algorithm is widely-used; its limitation is that, its calculations are very extensive, resulting in slow training. The used of momentum constant in the delta rule of the BP algorithm is the simplest method to avoid this problem. According to [3] the inclusion of momentum constant in back-propagation algorithm has a stabilizing effect on the training. It achieves this by slowing down the process when the learning surface exhibits peaks and valleys.

3. METHODOLOGY

The following stages summaries the research methodology adopted in this study.

Problem identification. At this stage, references where collected related to coronary heart disease. Problems relating to the disease diagnosis were also established.

Data Collection and preprocessing. At this stage data essential for diagnosis of heart disease was obtained from the UCI Machine Learning Repository. The data was pre-processed and normalized.

Design and Implementation. MATLAB neural network toolbox was utilized to design and implement the ANN system. Several experiments were carried out to determine the optimal number of hidden layers, number of neurons per layer, the input data, the training algorithms, the activation function of the ANN system.

Discussion and Result: Discussion were made on the prediction accuracy of the ANN model.

Fig 2 shows the result of our experiment, we found that for both sigmoid and hyperbolic tangent activation function,

4. EXPERIMENTAL RESULTS

4.1 Input layer

In this study, the input data essential for diagnosis of heart disease were obtained from the UCI Machine Learning Repository (2007). The data had 303 instances 14 attributes, including 1 class attribute. The values of output classification were pre-processed from (1, 2, 3) to (0, 1). Also, the attributes of the input dataset contain values that were falling within different ranges. To make the ANN training more efficient and obtain high accuracy, we normalized and pre-processed the dataset. Some of the input attributes were normalized to fall between 0-1 while those with values already in smaller range were left the same. The total number of instances was also reduced to 297 after cleaning, which sees the elimination of instances with missing values. The dataset was divided into 80% for training and 20% for testing. However the effect of dataset division will be tested during the experimentation.

4.2 Hidden Layer

The number of neurons in the hidden layer of an ANN has great influence on the network performance, specifically the network's ability to generalize from the training data to the unknown examples[10]. Often if the number of neurons in the hidden layer is high, the network may determine only the decision boundaries in a vector space that are undeservedly influenced by specific properties in the training data, this is known as over fitting. if hidden neurons are set low , the network might not fully recognize the structures present in the training data a phenomenon called under fitting. Thus it is recommended to experiment with many values until a suitable value is found [14]. As a result, we conduct an experiment to determine the best hidden layer neurons for our network. We experiment with both hyperbolic tangent and sigmoid activation functions. The following graph shows the result of our experiments.

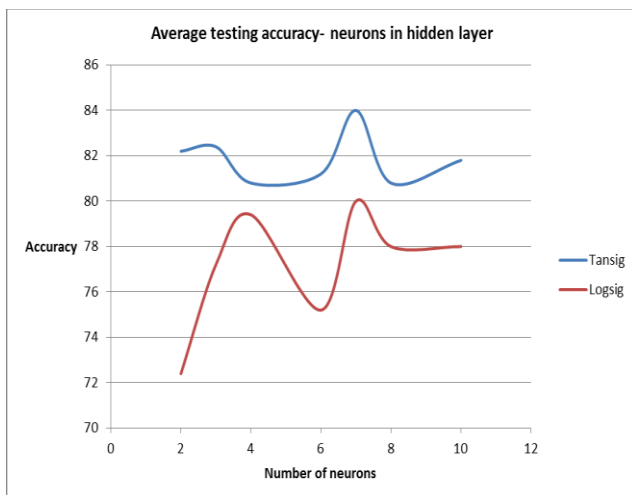


Fig 2: Number of neurons in the hidden layer

4.3 Learning Rate and Momentum Constant

The learning rate controls the amount by which the weights and biases of a network are change during training. If the learning rate is set too high, the algorithm may become unstable. If the learning rate is set too small, the algorithm may take time to converge. On the other hand, the momentum

seven neurons resulted in the best accuracy. This value is therefore adopted for our study.

constant is a parameter introduced in the delta rule of back-propagation algorithm to speed up training. The performance of MLP trained with BP algorithm is greatly influenced by proper setting of both learning rate and momentum constant [15]. Better network performance can be achieved by careful selection of these two parameters. However, it is not practical to determine the optimal values for these parameters without an experiment. **Figure 3&4** shows the details of the experiment we conducted to find the best combination of both learning rate and momentum constant values that can increase both the network's training time and performance.

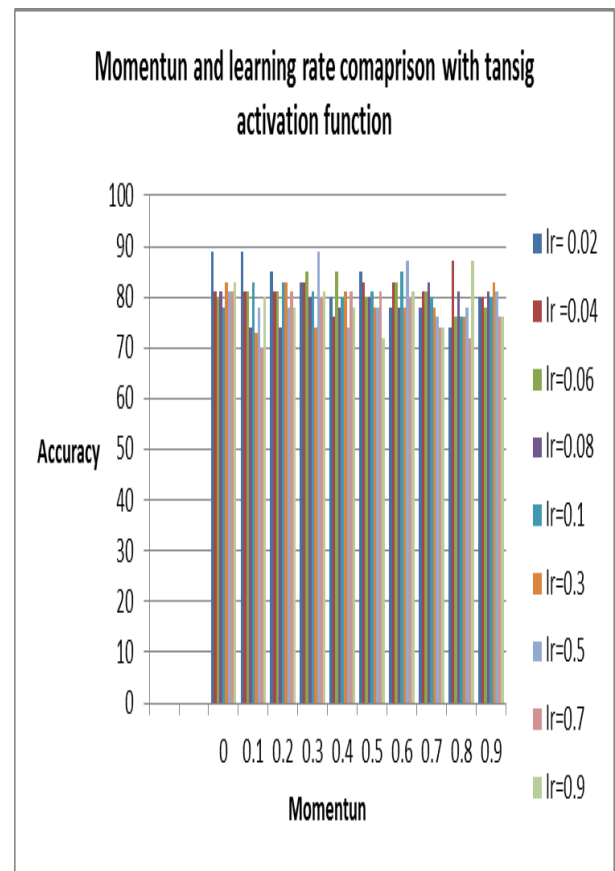


Figure 3: Learning rate versus momentum constant using Tansig activation function

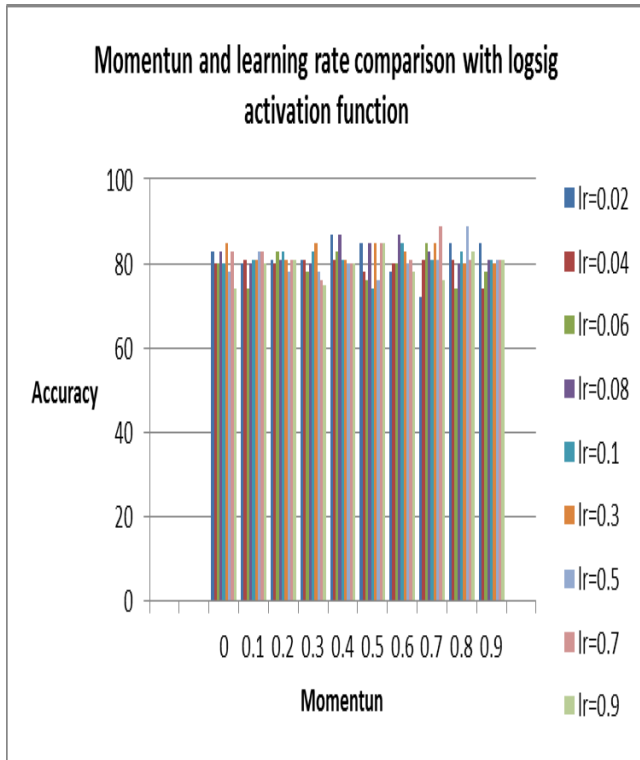


Figure 4: Learning rate versus momentum constant using logsig activation function.

Figure 3 and 4 are graphical representations of the comparison between learning rate and momentum constant for both sigmoid and hyperbolic tangent activation functions. For hyperbolic tangent, it's a clear that the highest accuracy was produced when the learning rate was 0.02 and momentum constant was 0. However, similar result was produced when the momentum was 0.1. Nevertheless; the graph appears to be different for sigmoid activation function with highest accuracy achieved at learning rate 0.7 and momentum constant 0.7. Similarly, same result was produced with learning rate 0.5 and momentum constant 0.8. Due to this inconsistency, the value of the momentum and learning rate was determined after another experiment that seeks to find the optimal activation function for the ANN.

4.4 Activation Function

There are many activation functions that are often used to compute the output of a neuron, the most popular ones among them with practical applications are; sign, step, linear and sigmoid functions [3]. While the learning algorithm selected for this research is BP, experiments was conducted on two activation functions namely, basic sigmoid and sigmoid represented with hyperbolic tangent. The outcome of this experiment was used to determine the best activation function for the ANN. Each activation function was tested with it best momentum and learning rate values found on the previous experiments. Figure 5 shows the results of our experiments.

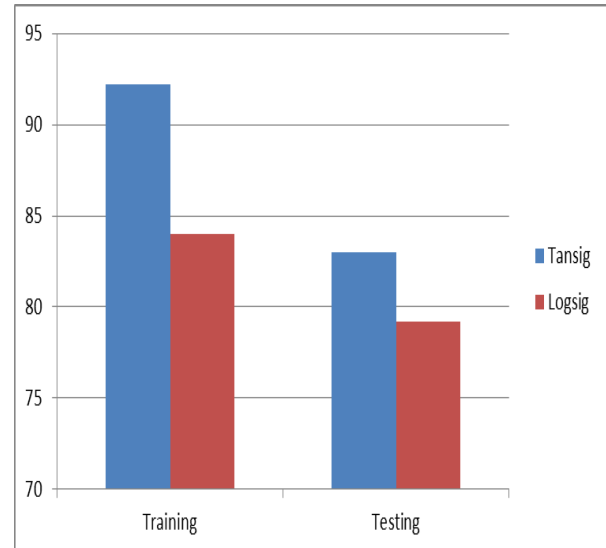


Figure 5: Experimentation on activation function

From the figure above, the hyperbolic tangent produced the highest accuracy of 92.2 in training and 83 in testing. As a result, the ANN was implemented using the hyperbolic tangent.

4.5 Data Division

We carried out an experiment to analyse the relationship between data division and accuracy of the network. The result of this experiment was used to determine the suitable data division for the ANN. Figure 6 shows a graph demonstrating our findings.

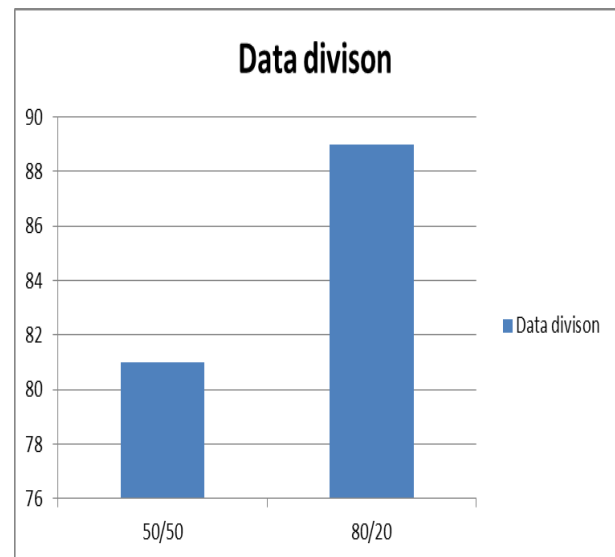


Figure 6: Experimentation on data division

Key points about the graph

50/50- 50% used for training and 50% used for testing

80/20- 80% used for training and 20% used for testing.

From the above graph; it is evident that 80/20 produced the best accuracy, implying that the higher the percentage of data used in training the better the testing accuracy.

Based on the results and that of the previous experiments, we conclude that the most suitable ANN model for predicting heart disease using the UCI machine learning data is:

1. 7 neurons in the hidden layer
2. 0.02 as the learning rate
3. 0.1 as the momentum constant
4. Hyperbolic tangent as the activation function
5. 80/20 as the data division

4.6 Principal Component Analysis

Having found the most suitable ANN, we conducted an experiment to discover if the input data contains correlated information and also analyse how this may influence the network accuracy. This was done using principal component analysis (PCA) as explained earlier in the design section of this paper. During this experiment, the input data was normalised, so that it will have zero mean and unity variance. A PC variance value of 0.05 was also used. This means that PCA will eliminate those principal components which contribute less than 5% to the total variation in the data set. After the experiment, it was found that the number of attributes within the dataset was reduced from 13 to 9, implying that there is indeed correlation between these attributes. Nevertheless, it was later discovered that although PCA eliminates some redundant information within the dataset, it has a significant impact on the testing accuracy. The average accuracy of the network without PCA was 83% and was later reduced to 82.2% after applying PCA. Therefore, it can be concluded that all the principal components used in this dataset before the application of PCA are very vital in predicting the likelihood of individual having heart disease. Hence, it was concluded that PCA will not be used for the final implementation of the ANN system due to its lower accuracy influence on the ANN.

4.7 Speed and Accuracy Comparison between different training algorithms

The following is the final experiment carried out during the implementation process of the ANN. The aim of the experiment was to find the best training algorithm that gives the best accuracy.

Table 1: Experimentation on training algorithm

Training Function	Technique	Time	Epochs	Goal Met	Accuracy
TrainGD	Gradient Descent	58s	10000	No	89
TrainGDM	Gradient Descent with momentum	1.6 mins	10000	No	89
TrainGDA	Variable Learning rate	58s	9507	Yes	81
TrainGD with Momentum	Variable Learning rate	52s	7909	Yes	81

TrainRP	Rprop	11s	341	Yes	81
TrainCGF	Fletcher-powell CG	12s	635	Yes	83
TrainCGP	Polak-Ribiere CG	4s	270	Yes	83
TrainCGB	Powell-Beale CG	18S	978	Yes	81
TrainLM	Levenberg-Marquardt	14s	22	Yes	76

shows the effect of different training algorithms on the networks performance. It is clear that, the fastest training algorithm for this problem is trainrp. However, the testing accuracy of trainrp is lower than that of traingd and traingdm. It can be concluded that though both traingd and traingdm were very slow and also felt to reach the performance goal, this algorithms however perform well on this particular data producing the highest accuracy than others.

5. CONCLUSION

In this study, we have shown that a Multilayer Perceptron trained with back-propagation algorithm is able to predict patient's heart disease status from real world data obtained from UCI machine learning repository. We have also shown the effect of the network's different parameter values on its prediction accuracy. Specifically, the authors have used principal component analysis to show that all the 14 attributes of the dataset are vital for heart disease diagnosis. We have shown that our system is able to achieve high prediction accuracy of 92.2%, proving the usefulness of the proposed model in medical diagnosis.

6. ACKNOWLEDGMENTS

Our thanks goes to the UCI machine learning repository team for making the heart disease patient data available online.

7. REFERENCES

- [1] NHS (2011) Coronary heart disease. Available at: <http://www.nhs.uk/conditions/coronary-heart-disease/Pages/Introduction.aspx> (Accessed: 22/12/11)
- [2] Shi L and Wang XC. 2010. Artificial neural networks: Current applications in modern medicine. IEEE, 3837
- [3] Negnevitsky, M (2002) Artificial Intelligence: A guide to intelligent systems. 1st edition. Pearson.
- [4] Kurt, I., Ture, M. & Kurum, A. T, (2008). Comparing performances of logistic regression, classification, regression tree and neural networks for predicting coronary artery disease. Expert Systems with Applications, 34, 366–374. [online]. Available at: <http://www.sciencedirect.com> (Accessed: 21 March 2018)
- [5] Sibanda, W and Pretorius, P. (2011). Novel Application of Multi-Layer Perceptrons (MLP) Neural Networks to Model HIV in South Africa using Seroprevalence Data

- from Antenatal Clinics. *International Journal of Computer Applications*, 35(2). Available at: www.ijcaonline.org (Accessed: 24 March 2018]
- [6] Delen, D., Walker, G., & Kadam, A. (2005). Predicting breast cancer survivability: A comparison of three data mining methods. *Artificial Intelligence in Medicine*, 34(2), 113–127. Available at: <http://www.sciencedirect.com> (Accessed: 20 March 2018]
- [7] Ozyilmaz, L. & Yildirim, T. (2003). Artificial neural networks for diagnosis of hepatitis disease. In: *Proceedings of the International Joint Conference on Neural Networks*, Vol 1, 586 – 589. July 20-24. Istanbul, Turkey
- [8] Shanthi, D., Sahoo, .G, & Saravanan, N.(2008). Input Feature Selection using Hybrid Neuro-Genetic Approach in the Diagnosis of Stroke Disease. *International Journal of Computer Science and Network Security* 8(12).
- [9] Junita M.S and Brian S.H.2008. Improved Neural Network Performance Using Principal Component Analysis on Matlab. *International Journal of The Computer , The Internet and Management* 16(2).pp1-8
- [10] Hongmei, Y., Yingtao,J., Jun,Z., Chenglin,P. & Qingui, Lui (2006). A Multilayer perceptron-based Medical decision System for heart disease diagnosis . *Expert Systems with Applications* 30(2006),271-281.
- [11] Khemphila,A. and Boonjink, V. (2010) Comparing performances of logistic regression, decision trees , and neural networks for classifying heart disease patients.In:CISIM,2010 International conference on computer information systems and industrial management application. Krakow 8-10 Oct. 2010
- [12] Mutasem Khalil,S.A, Khairuddin, B,O. and Shahrul, A.N, (2009). Back Propagation Algorithm: The Best Algorithm Among the Multi-layer Perceptron Algorithm. *International Journal of Computer Science and Network Security* 9(4).
- [13] Bernard, W., Aaron, G., and Youngsik, K & Dookum, P. (2013). The No-Prop algorithm: A new learning algorithm for the multlayer neural network. *Neural Networks* 37(2013) 182-188. Available at: www.elsevier.com/locate/neunet. [Accessed: 02 September 2018]
- [14] Kavzoğlu, Taşkin (2001) An investigation of the design and use of feed-forward artificial neural networks in the classification of remotely sensed images. PhD thesis, University of Nottingham., viewed 15 October 2017 < <http://eprints.nottingham.ac.uk/13872/>>
- [15] Attoh-Okine, N.O. (1999). Analysis of learning rate and momentum term in back propagation neural network algorithm trained to predict pavement performance, *Advances in Engineering Software* 30(4), p 291-302