Novel Application of Multi-Layer Perceptrons (MLP) Neural Networks to Model HIV in South Africa using Seroprevalence Data from Antenatal Clinics

Wilbert Sibanda
North-West University, Vaal Triangle Campus
VAN ECK BLVD, VANDERBIJLPARK, 1900,
South Africa

Philip Pretorius
North-West University, Vaal Triangle Campus
VAN ECK BLVD, VANDERBIJLPARK, 1900,
South Africa

ABSTRACT
This paper presents an application of Multi-layer Perceptrons (MLP) neural networks to model the demographic characteristics of antenatal clinic attendees in South Africa. The method of cross-validation is used to examine the between-sample variation of neural networks for HIV prediction. MLP neural networks for classifying both the HIV negative and positive clinic attendees are developed and evaluated using validity and reliability of the test. Neural networks are robust to sampling variations in overall classification performance.

General Terms
Neural networks, HIV, multilayer perceptron (MLP)

Keywords
Multi-layer Perceptrons, neural networks, HIV/AIDS, seroprevalence data, antenatal

1. INTRODUCTION
HIV/AIDS is causing extra-ordinary problems to human health throughout Africa and the world in general (UNAIDS, 2009). To introduce control strategy and to plan political and economic policies, it is crucial to estimate and predict the magnitude of the spread of the epidemic in the society. There are different methods to accomplish this task. A population-based survey can give a clear picture on the status of the epidemic but such a survey is extremely hard and expensive (WHO).

The spread of the epidemic can be predicted from HIV incidence data however this requires a long follow-up period. The epidemic can also be predicted based on AIDS notification or AIDS mortality. This again can be unreliable as in most cases there might be report delays and flaws in the registration system.

It is well known fact that HIV spreads within a society through different biological, social and environmental factors and hence modeling these factors has the advantage to understand the spread of the HIV epidemic over a specific population. The model formulation process is very important, as the results are a reflection of the model. The model needs to extract as much information as possible so that it is a good representation of the population. The parameters and assumptions involved in building the model should be clearly stated and understood. The assumptions have a role in the process of assessing the correctness of the model and also the accuracy and efficiency of the parameters involved in formulating the model.

Artificial Neural Networks (ANNs) have been applied to an increasing number of real world problems of varying complexities (Patel J.L., Goyal R.K., 2007). Their greatest advantage is in solving problems that are too complex for conventional technologies, such as problems that do not have an algorithmic solution or for which an algorithmic solution is too complex to be found. In general, because of their derivation from the biological brain, ANNs are well suited to problems that people are good at solving, but for which computers are not. These problems include pattern recognition and forecasting. The later techniques require the recognition of the trends in data. Other advantages of neural networks include; adaptive learning (i.e. an ability to learn to do tasks based on the data given for training or initial experience) and self-organization (i.e. an ANN can create its own organization or representation of the information it receives during learning time and real-time operation).

In using neural networks, the entire available data set is usually randomly divided into a training (in-sample) set and a test (out-of-sample) set. The training set is used for neural network model building and the test set is used to evaluate the predictive capability of the model. While this practice is adopted in many studies, the random division of a sample into training and test sets may introduce bias in model selection and evaluation in that the characteristics of the test may be very different from those of the training. Cross-validation will be used to accurately describe the predictive performance of the neural networks. Cross-validation is a re-sampling technique which uses multiple random training and test sub-samples. The advantage of cross-validation is that all observations or patterns in the available sample are used for training the model. The cross-validation analysis will yield valuable insights on the reliability of the neural networks with respect to sampling variation.

Section 3 contains the variable description, the data used and the design of this study. Section 4 details the cross-validation results obtained in the study, while the concluding remarks are provided in section 5.

2. NEURAL NETWORKS FOR PATTERN CLASSIFICATION

2.1. Neural Networks

Neural networks are flexible, nonparametric modeling tools (Trentin and Freno, 2009). They can perform any complex function mapping with desired accuracy. An ANN is typically composed of several layers of many computing elements called nodes. Each node receives an input signal from other nodes or external inputs and after processing the signals locally through a transfer function, it outputs a transformed signal to other nodes or final result. In MLP, all nodes and layers are arranged in a feed-forward manner. The first layer is called the input layer where external information is received. The last layer is called the output layer where the network produces the model solution. In between, there are one or more hidden layers which are critical to ANNs to identify the complex patterns in the data. An example of an MLP with one hidden layer and one output node is shown in Fig. 1. This three-layer MLP is a commonly used ANN structure for two-group classification problems like HIV status prediction.

As in any statistical model, the parameters (weights) of neural network model need to be estimated before the network can be used for prediction purposes. The process of determining these weights is called training. The training process aims at changing the weights so as to minimize the error between the observed and the predicted outcomes, traditionally given by the sum of their squared differences across all observations (patterns) scrutinized in one iteration (epoch) of training. During training, the calculated error is backpropagated to the network and the weights are accordingly adjusted so as to improve the network prediction. The backpropagation algorithm is an iterative method based on gradient descent on the error surface reaching the minimal error possible (Zimmermann, Minin, and Kusherbaeva, 2011). The weights are continually modified as a function of the change in the error, and the amount of weight change is determined by a learning parameter.

The training phase is a critical part in the use of neural networks for classification problems. The network training is a supervised one in that the desired or target response of the network for each input pattern is always known a-priori.

2.2 The Multilayer Perceptron Neural Network Model

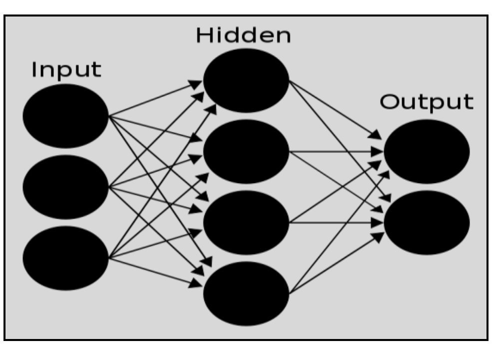

Figure 1: A Multilayer Perceptron (MLP) Neural Network Model

As stated in section 2.1, Fig. 1 illustrates a perceptron network with three layers. This network has an input layer (on the left) with three neurons, one hidden layer (in the middle) with three neurons and an output layer (on the right) with three neurons. There is one neuron in the input layer for each predictor variable.

Input Layer - A vector of predictor variable values is presented to the input layer. At the input layer, the values are distributed to each of the neurons in the hidden layer. In addition to the predictor variables, there is a constant input called the bias that is fed to each of the hidden layers. The bias is multiplied by a weight and added to the sum going into the neuron.

Hidden Layer - At the hidden layer, the value from each input neuron is multiplied by a weight, and the resulting weighted values are added together producing a combined value. The weighted sum is fed into a transfer function. The outputs from the hidden layer are distributed to the output layer.

Output Layer - On arrival at the output layer, the value from each hidden layer neuron is multiplied by a weight, and the resulting weighted values are added together producing a combined value. Thereafter, the weighted sum is fed into a transfer function, which outputs its own value.

2.3. Training Multilayer Perceptron Networks

The goal of the training process is to find the set of weight values that will cause the output from the neural network to match the actual target values as closely as possible. There are several issues involved in designing and training an MLP perceptron network such as:

- Selecting the number of hidden layers to use in the network.
- Deciding how many neurons to use in each hidden layer.
- Finding a globally optimal solution that avoids local minima.
- Converging to an optimal solution in a reasonable period of time.
- Validating the neural network to test for overfitting.

2.3.1 Selecting the Number of Hidden Layers

For nearly all problems, one hidden layer is sufficient. Using two hidden layers rarely improves the model, and it may introduce a greater risk of converging to a local minima. There is no theoretical reason for using more than two hidden layers. Three layer models with one hidden layer are recommended.

2.3.2 Deciding how many neurons to use in the hidden layers

One of the most important characteristics of a perceptron network is the number of neurons in the hidden layer(s). If an inadequate number of neurons are used, the network will be unable to model complex data, and the resulting fit will be poor.

If too many neurons are used, the training time may become excessively long, and, worse, the network may *over-fit* the data. When overfitting occurs, the network will begin to model random noise in the data. The result is that the model fits the training data extremely well, but it generalizes poorly to new, unseen data. Validation must be used to test for this.

2.3.3 Finding a globally optimal solution

A typical neural network might have a couple of hundred weights whose values must produce an optimal solution. The output of a neural network as a function of the inputs is often highly nonlinear, making the optimization process complex. The plot of error as a function of the weights for neural networks will most likely appear as a rough surface with many local minima such as shown in fig. 2.

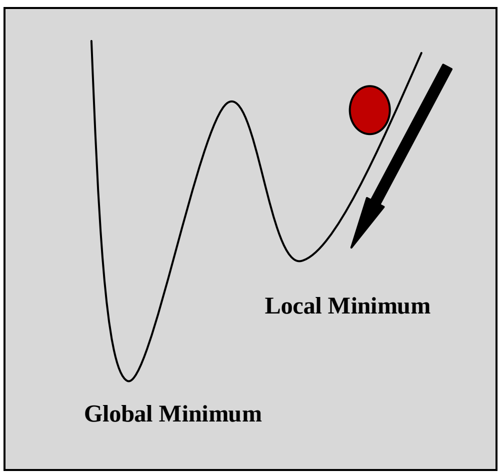

Figure 2: Plot of error as a function of the weights

Optimization methods such as steepest descent and conjugate gradient are highly susceptible to finding local minima if they begin the search in a valley near a local minimum. They have no ability to see the big picture and find the global minimum. Several methods have been tried to avoid local minima. The simplest is just to try a number of random starting points and use the one with the best value. A more sophisticated technique called *simulated annealing* improves on this by trying widely separated random values and then gradually reducing the random jumps in the hope that the location is getting closer to the global minimum. Conjugate gradient usually finds the optimum weights quickly, but there is no guarantee that the weight values it finds are globally optimal.

2.3.4 Converging to the Optimal Solution using Conjugate Gradient

For this research, given a set of randomly-selected starting weight values, we selected conjugate gradient algorithm to optimize the weight values.

The process to refine the weight values includes, running a set of predictor variable values through the network using a tentative set of weights, computing the difference between the predicted target value and the actual target value for this case, averaging the error information over the entire set of training cases, propagating the error backward through the network and computing the gradient of the change in error with respect to changes in weight values and making adjustments to the weights to reduce the error. Each cycle is called an epoch.

Because the error information is propagated backward through the network, this type of training method is called backward propagation. The backpropagation training algorithm was first described by Rumelhart and McClelland in 1986. Backpropagation using gradient descent often converges very slowly or not at all. While backpropagation with gradient descent is still used in many neural network programs, it is no longer considered to be the best or fastest algorithm. Neurosolutions software uses the conjugate gradient algorithm to adjust weight values using the gradient during the backward propagation of errors through the network. Compared to gradient descent, the conjugate gradient algorithm takes a more direct path to the optimal set of weight values. Usually, conjugate gradient is significantly faster and more robust than gradient descent. Conjugate gradient also does not require the user to specify learning rate and momentum parameters.

3. DESIGN OF STUDY

Since ANNs are used to study the relationship between the likelihood of being HIV positive or negative, two important questions need to be addressed:

i. What is the appropriate neural network architecture for this particular data set?

ii. How robust is the neural network performance in predicting the HIV status in terms of sampling variability?

For the first question, there are no definite rules to follow since the choice of the architecture also depends on the classification objective. For example, if the objective is to classify a given set of objects as well as possible, then a larger network may be desirable. However, if the network is to be used to predict the classification of unseen objects, then a larger network is not necessarily better. For the second question, a cross-validation approach was used to investigate the robustness of the neural networks in HIV status prediction.

3.1. Measures and Sample

This study utilizes a total of six quantitative demographic characteristics as variables, namely parity, gravidity, mother's age, father's age, educational level and syphilis. The qualitative characteristics such as race and province were not included in this study. Out of a total of 31 808 individuals, 4 000 HIV positive and 4 000 HIV negative individuals were randomly selected from the 2007 South African annual antenatal clinics (ANC) seroprevalence data. The 8 000 subjects selected from the 2007 antenatal data were extensively randomized to reduce bias using Neurosolution^R randomization function.

Gravidity is defined as the number of pregnancies, complete or incomplete, experienced by a female, while parity denotes the number of times the individual has given birth. The HIV status is binary coded; a 1 represents positive status, while a 0 represents a negative status. The complete specifications of the demographic characteristics are illustrated in Table 1.

Table 1: Summary of Input and Output Variables

Demographic Characteristics	Specifications	
Mother's age	13 – 45 years	**Inputs**
Father's age	15 – 55 years	
Educational level	0 – 13	
Syphilis (RPR)	0 (negative) & 1 (positive)	
Gravidity	0 – 10	
Parity	0 – 8	
HIV status	0 (negative) & 1 (positive)	**Output**

Prior to training the neural networks, the data columns were portioned into either input or desired output. The data rows were segmented into one of the following three groups; training, cross validation and testing as shown in Table 2.

Table 2: Data Tagging

Group	Description
Training	Data used by the neural network to learn from
Cross validation	Data used to evaluate performance during learning
Testing	Data used to evaluate performance after training

3.2. Design of Neural Network Model

ANNs are characterized by their architectures. As stated in Section 2, neural network architecture refers to the number of layers, nodes in each layer and the number of arcs. Networks with one hidden layer are generally sufficient for most problems including classification. All networks used in this study will have one hidden layer. For this classification problem, the number of input nodes is the number of predictor variables. For example, for the prediction of the HIV status, the networks will have six input nodes in the first layer corresponding to six predictor variables. Node biases will be used in the output nodes and logistic activation function will be specified in the networks.

For this study data preprocessing is conducted, though some studies suggest that (Shanker et al, 1996) data preprocessing is not beneficial for a classification exercise. Neural network training is a nonlinear nonconvex minimization problem and hence global solutions cannot be guaranteed. To reduce the likelihood of being trapped in a bad local minima, the ANNs were each trained 50 times by using 50 sets of randomly selected initial weights and the best solution of weights among the 50 runs is retained for a particular network architecture.

3.3. Cross-validation

The cross-validation method is employed to examine the neural network performance in HIV status prediction in terms of sampling variation. Cross-validation is a useful statistical technique to determine the robustness of a model. One simple use of the cross-validation idea is consisted of randomly splitting a sample into two sub-samples of training and test sets. The training sample is used for model fitting and/or parameter estimation and the predictive effectiveness of the fitted model is evaluated using the test sample. Since the best model is designed to fit one sub-sample, it often estimates the true error rate overly optimistically (Efron and Gong, 1983). The solution to this problem is to use the five-fold cross-validation by carrying out the simple cross-validation five times. In this study, a fivefold cross-validation is used as proposed by Zhang et al., 1997. The total sample is divided into five equal and mutually exclusive portions. Training will be conducted on any four of the five portions. Testing will be performed on the remaining part. As a result, five overlapping training samples are constructed and testing is also performed five times. The average test classification rate over all five partitions is a good indicator for the out-of-sample performance of a classifier.

3.4. Sensitivity Analyses

Sensitivity analysis assesses the effect that each of the network inputs has on the network output, thus providing a feedback as to which input channels are the most significant. Sensitivity analysis provides an opportunity to prune the input space by removing the insignificant channels, reducing the size and complexity of the network. Sensitivity analysis is therefore a method for extracting the cause and effect relationship between the inputs and outputs of the network.

4. RESULTS

4.1. Number of Neurons in the Hidden Layer

The average prediction (test set) percentages for each configuration are represented in Fig. 3. We can see that the performance increases with the number of neurons in the hidden layers, for HIV positive individuals; 66% prediction with one neuron, 69%, 71%, 72% and 74% prediction respectively for two, four, five and ten hidden layers. The prediction performance decreased for HIV negative individuals as the number of hidden layers increased. Based on the different responses to increases in the number of hidden layers between HIV negative and positive individuals, this research resorted to using the only one hidden layer for prediction purposes.

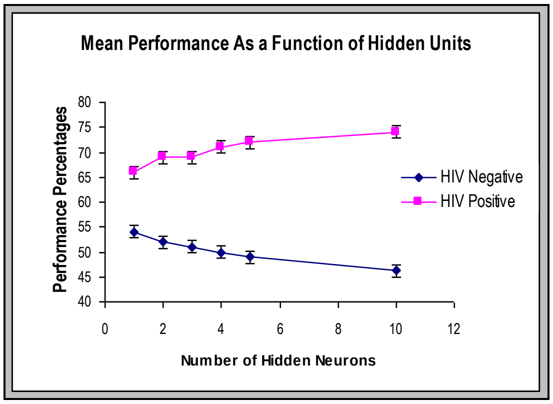

Figure 3: Mean Performance as a function of the hidden unit

4.2. Number of Iterations

The MSE between observed values and values estimated by the network declined very rapidly from a high starting value to about 0.35 after 150 iterations in the training set (Fig.4). In the validation set, a similar variation was observed, with minimum values close to 0.35. Values of MSE stabilized after 150 iterations.

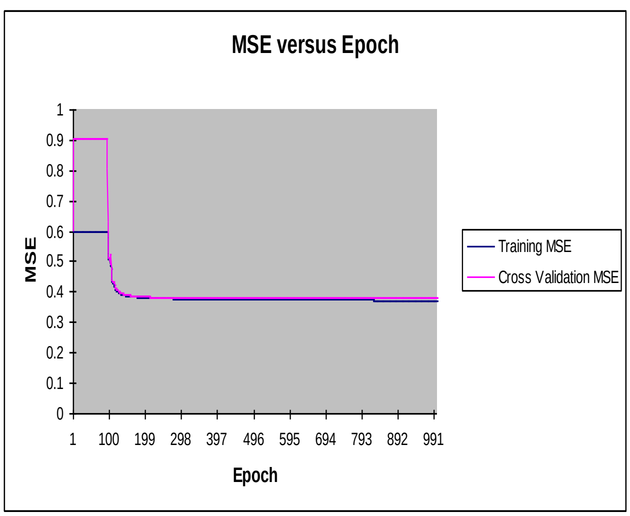

Figure 4: MSE as a function of the training iteration number

The percentages of correct classifications increased slowly for HIV positive individuals up to 1 000 iterations (Fig. 5. In this study, training of the network was stopped at 150 iterations, to avoid further deterioration in the classification of HIV negative individuals.

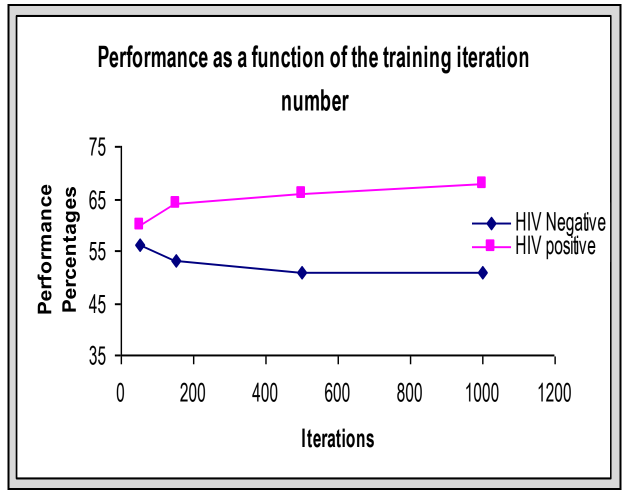

Figure 5: Performance (percentage of correctly classified records) as a function of the training iteration number (epoch)

Cross-validation results on the predictive performance of neural networks are given in Table 3. Across the five small test sub-samples, overall classification rate of neural networks ranges from 66% to 74% for HIV positive and 46% to 54% for HIV negative individuals from 1 to 10 hidden nodes. For HIV positive prediction, neural networks give an average of 66% across the five sub-samples using only one hidden layer compared to 54% for HIV negative prediction.

Table 3: Cross-validation results on the predictive performance for the five small subsamples

Hidden Nodes	Sub-samples										Mean ±SD (HIV)	
	1		2		3		4		5			
	-	+	-	+	-	+	-	+	-	+	-	+
1	53	64	54	69	57	65	52	67	53	64	**54±** 1.9	**66±** **2**
2	49	70	55	68	52	72	50	70	52	69	**52±** 2.6	**69±** 1.9
3	50	67	51	73	54	68	48	72	51	69	**51±** 2.7	**69±** 3.2
4	48	70	53	72	52	71	47	73	50	71	**50±**	**71±**

											1.6	1.9
5	49	68	50	76	51	71	47	73	49	71	**49±** 1.6	**72±** 2.6
6	49	69	59	59	53	70	54	70	54	71	**54± 0.2**	**68± 0.2**
7	46	72	49	77	45	79	46	76	47	77	**47± 0.4**	**76± 02**
8	46	73	48	77	48	74	46	75	48	76	**47± 0.2**	**75± 0**
9	47	72	40	82	48	74	45	76	47	79	**45**	**77**
10	40	73	46	76	47	76	46	75	46	72	**46±** 0.5	**74±** 1.8

4.3. Sensitivity

The sensitivity test showed that mother's age and the father's age had the greatest effect on the HIV status of the antenatal clinic attendees. Gravidity and syphilis had the lowest effect as shown in Fig. 6.

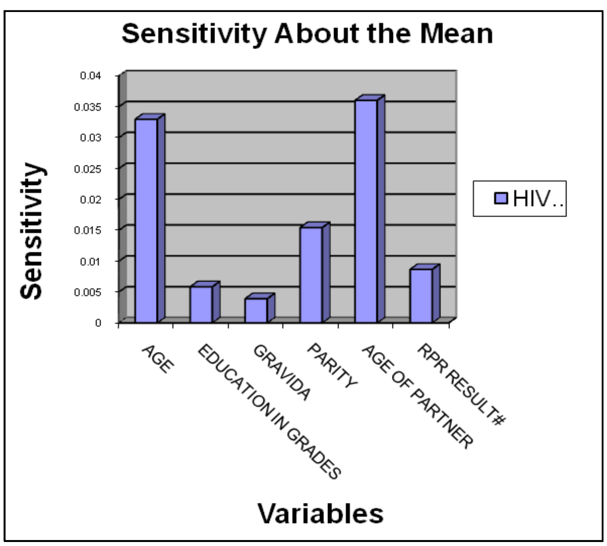

Figure 6: Sensitivity Test Results

5. CONCLUSION

The MLP has been shown to be a useful tool for prediction, function approximation and classification. The practical benefits of a modeling system that can accurately reproduce any measurable relationship are huge. The benefits of the MLP approach are particularly apparent in applications where a full theoretical model cannot be constructed, and especially when dealing with non-linear systems. The numerous difficulties in implementing, training and interpreting the MLP must be balanced against the performance benefits when compared to more traditional, and often inappropriate, techniques. It is indeed clear that the full benefits that neural networks offer can only be realized through a fundamental understanding of the basic theory.

6. ACKNOWLEDGEMENTS

Wilbert Sibanda acknowledges Doctoral funding from South African Centre for Epidemiological Modeling (SACEMA), Medical Research Council of South Africa (MRC) and the North West university (Vaal Triangle). Special thanks to South African National Department of Health for the 2006 and 2007 antenatal seroprevalence data.

7. REFERENCES

[1] Efron B., Gong G. 1983. A leisurely look at the bootsrap, the jackknife and cross-validation. American Statistician. vol. 37, 36-48.

[2] Patel J.L, Goyal R.K. 2007. Applications of artificial neural networks in medical science. Curr Clin Pharmacol. (Sep. 2007), 2(3), 217-26.

[3] Trentin E. and Freno A. 2009. Unsupervised nonparametric density estimation: A neural network approach. Proceedings of International Joint Conference on Neural Networks, Atlanta, Georgia, USA. (Jun. 2009), 14-19.

[4] UNAIDS. 2009. AIDS Epidemic Update, www.unaids.org

[5] World Health Organization (WHO) and the Joint United Nations Programme. Guidelines for measuring national HIV prevalence. UNAIDS/WHO Working Group on Global HIV/AIDS and STI surveillance

[6] Zimmermann H.G., Minin A. and Kusherbaeva A. 2011. Comparison of the complex valued and real valued neural networks trained with gradient descent and random search algorithms. ESANN 2011 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 27-29 April 2011.