

https://dspace.org/

Measuring confidence of missing data estimation for HIV classification

Mistry, Jaisheel

2009-07-27T12:16:58Z
http://hdl.handle.net/10539/7124

Downloaded from a DSpace repository. https://dspace.org/

Measuring Confidence of Missing Data Estimation for HIV Classification

Jaisheel Mistry

A dissertation submitted to the Faculty of Engineering and the Built Environment, University of the Witwatersrand, in fulfillment of the requirements for the degree of Master of Science in Engineering.

Johannesburg 2008

Declaration

I declare that this dissertation is my own, unaided work, except where otherwise acknowledged. It is being submitted for the degree of Master of Science in Engineering in the University of the Witwatersrand, Johannesburg. It has not been submitted before for any degree or examination in any other university.

Signed this _____ day of _____ 20_____

_____

Jaisheel Mistry

Abstract

Computational intelligence methods have been applied to classify pregnant women's HIV status using demographic data from the South African Antenatal Seroprevalence database obtained from the South African Department of Health. Classification accuracies using a multitude of computational intelligence techniques ranged between 60% and 70%. The purpose of this research is to determine the certainty of predicting the HIV status of a patient. Ensemble neural networks were used for the investigation to obtain a set of possible solutions. The predictive certainty of each patients predicted HIV status was computed by giving the percentage of most dominant outputs from the set of possible solutions. Ensembles of neural networks were obtained using boosting, bagging and the Bayesian approach. It was found that the ensemble trained using the Bayesian approach is most suitable for the proposed predictive certainty measure. Furthermore, a sensitivity analysis was done to investigate how each of the demographic variables influenced the certainty of predicting the HIV status of a patient.

Acknowledgements

I would like to thank my supervisor Prof. Tshilidzi Marwala for his valuable guidance, encouragement and support throughout the study. He not only proposed the project, but also sourced funding for it. I would like to thank Dr. Fulufhelo Vincent Nelwamondo for all his assistance with proof reading and giving useful feedback regarding my research. I would like to thank my friends Fulufhelo Andrew Netshiongolwe, Linda Mthembu and Rofhiwa Musehane for proof reading this dissertation. Thanks to the National Research Foundation (NRF) for providing funding. I would also like the South African Department of Health for allowing access to their database.

Table of Contents

Declaration i

Abstract ii

Acknowledgements iii

List of Figures vii

List of Nomenclature ix

Chapter 1: Missing Data Estimation for HIV/AIDS model 1

1.1 Introduction 1

1.2 Missing Data Estimation 2

1.2.1 Substitution Methods 2

1.2.2 Hot deck imputation 3

1.2.3 Regression Methods 3

1.2.4 Expectation Maximization 4

1.3 Confidence Measurements 4

1.3.1 Confidence Intervals 4

1.3.2 Predictive Certainty 5

1.3.3 Standard Deviations 5

1.4 Antenatal Dataset 5

1.4.1 Variables 7

1.4.2 Data Preparation 8

1.5 Modeling the HIV Pandemic 9

1.6 Neural Networks 10

1.7 Outline of Dissertation 12

1.8 List of publications 13

Chapter 2: Missing Data Estimation using Computational Intelligence 15

2.1 Introduction 15
2.2 Missing Data Mechanisms 15
2.2.1 Missing at Random 15
2.2.2 Missing Completely at Random 16
2.2.3 Non- Ignorable (Not Missing at Random) 16
2.3 Autoencoder and Autoassociative Neural Network 16
2.4 Genetic Algorithm 17
2.5 Computational Intelligence Missing Data Estimation 18
2.6 Conclusion 20
Chapter 3: Autoencoder Neural Network for HIV Classification 21
3.1 Introduction 21
3.2 Network Training 21
3.3 Experiment Methodology 24
3.4 Results 24
3.5 Conclusion 27
Chapter 4: Optimization Methods 28
4. 1 Introduction 28
4.2 Markov Chain Monte Carlo 28
4.3 Simulated Annealing Optimization 29
4.4 Experimental Methodology 31
4.5 Results and Discussion 32
4.6 Conclusion 33
Chapter 5: Ensemble Based Neural Network Systems 34
5.1 Introduction 34
5.2 Ensemble Generation 35

5.2.1 Bagging 35
5.2.2 Boosting 37
5.2.3 Bayesian Approach 39
5.3 Experimental Methodology 40
5.4 Results 41
5.5 Conclusion 44
Chapter 6: Demographic influences for HIV classification 46
6.1 Introduction 46
6.2 Causal Influences for HIV Virus 46
6.3 Experimental Methodology 47
6.4 Results 47
6.5 Conclusion 50
Chapter 7: Disscusion and Conclusion 51
7.1 Summary of findings 51
7.2 Recommendations for future work 53
Chapter 8: References 54

List of Figures

Figure 1.1Illustration of a single layer MLP Neural Network..... 11
Figure 2.1Autoassociative Neural Network..... 17
Figure 2.2 General Computational intelligent method used to estimate missing data..... 20
Figure 3.1 Percentage Error for Neural Network Training for Hidden Nodes Ranging from 1 to 22 22
Figure 3.2 Accuracy for Neural Network Training for Hidden Nodes Ranging from 1 to 22..... 22
Figure 3.3 Training Accuray for diiferent amount of training cycles..... 23
Figure 3.4 Training error for different amounts of training cycles..... 24
Figure 3.5 Distribution of Estimation Accuracy for Neural Networks trained to certain testing accuracy..... 25
Figure 5.1 Distribution of the Predictive Certainty Values for the Ensemble trained using the Bagging Training Method..... 42
Figure 5.2 Distribution of the Predictive Certainty Values for the Ensemble trained using the Boosting Training Method..... 43
Figure 5.3 Distribution of the Predictive Certainty Values for the Ensemble trained using the Bayesian Training Method..... 43

List of Tables

Table 1.1 Summary of Data fields for the HIV Dataset 7
Table 1.2 The Unary Coding scheme used for the Province Variable..... 9
Table 3.1 Autoencoder Neural Networks used for the Investigation..... 26
Table 3.2 Over-trained and Under-trained Autoencoder Neural Networks 26
Table 5.1 Accuracy of the Different Ensembles for Predictive Certainty Ranges..... 44
Table 6.1 Average change in Predictive Certainty (%) for HIV positive and HIV negative 48
Table 6.2 Average change in Predictive Certainty(%) for HIV positive and HIV negative patients for changes in Race Type(RT)..... 48
Table 6.3 Average change in Predictive Certainty(%) for HIV positive and HIV negative patients for changes in Age Group 48
Table 6.4 Average change in Predictive Certainty (%) for HIV positive and HIV negative patients for changes in Partners Age Group..... 49
Table 6.5 Average change in Predictive Certainty(%) for HIV positive and HIV negative patients for changes Gravidity Ranging from 0 to 5 49
Table 6.6 Average change in Predictive Certainty(%) for HIV positive and HIV negative patients for changes Parity Ranging from 0 to 5 49
Table 6.7 Average change in Predictive Certainty (%) for HIV positive and HIV negative patients for changes in Province(P) 49

List of Nomenclature

AANN	Autoassociative Neural Network
ANN	Autoencoder Neural Network
CI	Computational Intelligence
GA	Genetic Algorithm
HMC	Hybrid/Hamiltonian Monte Carlo
MCMC	Markov Chain Monte Carlo
MH	Metropolis Hasting
MLP	Multi Layer Perceptron
NN	Neural Networks
PC	Predictive Certainty
PCA	Principal Component Analysis
RBF	Radial Basis Function
RNN	Recurrent Neural Network
SA	Simulated Annealing
SA	Sensitivity Analysis
SCG	Scaled Conjugate Gradient

Chapter 1: Missing Data Estimation for HIV/AIDS model

1.1 Introduction

Modern technology such as the internet allows for data to be collected in a variety of ways and this often leads to excess data. It is necessary to use this data for decision making and extracting useful information, however there is a common problem of missing data. This occurs in databases and for real time systems such as machinery with sophisticated control systems and the data may go missing during the data-retrieval stage where data are not collected using a standardized procedure, this makes the data to be intentionally left unanswered or forgotten [1]. Data in a database may also be lost during the data-storage process where there may be a break in transmission or data distortion occurs over a noisy transmission channel. For industrial machinery, sensor failure in modern control systems also leads to missing data. Hence it is difficult to make decisions or gather useful information. To make inferences from data, it is essential that one uses complete data and therefore it is necessary to estimate the missing values [2].

Many methods are used for estimating missing data. Some of these methods include zero substitution, mean substitution, interpolation, extrapolation, look up tables and statistical regression. Other missing data estimation methods are found in [2-6].

A new computational intelligent (CI) method using autoassociative neural networks (AANN) and genetic algorithms (GA) have been used to successfully estimate missing data [7]. This CI missing data estimation method uses the neural networks to learn interrelationships within the data and the GA is used to minimize an error function between the input and output of the neural network. The method presented in [7] gives the estimated value of the missing data but this method gives no indication of the confidence of such estimates. This method was tested on unseen test sets to show that they worked accurately. It was found in [8] that these methods worked better than using stand alone neural network (NN) classifiers. However it was found

that the accuracy on the test data was not particularly high for a certain HIV dataset when estimating a patient's HIV status [8-10].

The first aim of the research is to estimate the HIV status of a patient and give a confidence measure to the predicted value. The research is carried out using an existing computational intelligence missing data estimation method. A set of estimates will be collected using optimization techniques that give a set of possible optimum values. The first approach is to estimate the missing values using the optimization technique. The second approach is by using an ensemble of neural networks. Research is done to determine which approach is most suitable for giving relevant confidence information when classifying the HIV status of a patient.

The rest of the dissertation is organized as follows: Section 1.2 of this chapter gives a brief background on missing data estimation. Section 1.3 discusses different methods of expressing confidence measurements. The HIV dataset used for the investigation in this dissertation is discussed in section 1.4. A brief background on existing methods of modeling the HIV pandemic is then presented in section 1.5. Section 1.6 introduces the neural networks used extensively for the various investigations presented in this dissertation. Finally an outline of this dissertation is given in section 1.7. Section 1.8 lists the publications made by the author when doing the research presented in this dissertation.

1.2 Missing Data Estimation

It was mentioned earlier that there are various methods for estimating missing data. The computational intelligence missing data estimation method is explained in chapter 2. Popular methods for dealing with missing data include substitution, hot deck imputation, regression methods and imputation using the expectation maximization process [2]. These methods are briefly explained and discussed.

1.2.1 Substitution Methods

Two types of popular substitution methods include mean substitution or zero substitution. Zero substitution deals with placing a zero as the estimate value [7].Using this method is not sensible because it may have no relevance to the data type that is to be estimated. For example,

substituting a person's age with zero is not sensible. For mean substitution the missing value is estimated to be the mean of the variable for all available cases. Mean substitution has the high likelihood of producing biased estimates, and hence it is also not recommended. Mean substitution may also result in values that are not sensible. Suppose there is a variable x which can either hold the value of 1, 2, 3 or 4. Now suppose we have three complete entries which take on the value of 1, 3 and 4. The mean value for x equals 2.33 and this value has no significance because x can only be in 1 of the four possible states [7]. For the latter scenario it is necessary to use mode substitution for categorical variables. Mode substitution substitutes the most frequent or popular class for the categorical variable. Disadvantages of using mean and mode substitution increase when working with small sample sets. For larger data sets it is advantageous to use means and modes because they represent the most probable estimate.

1.2.2 Hot deck imputation

Hot deck imputation is a look up table method which works by finding a similar case as the one with the missing value. Suppose the variable x is missing in a given record, the x value is substituted with the x value of a record that has the same or similar values for the other fields. An advantage of this method is that the estimate values will be more sensible in that categorical variables will remain categorical and continuous variables will remain continuous. A disadvantage of using the hot deck method is that it is difficult to define similarity and the method is inefficient in cases where there is a large amount of uncertainty [2]. This occurs when there are large amounts of similar cases and the missing variable type differs significantly from each other.

1.2.3 Regression Methods

For regression methods, a regression equation based on complete data for a given variable is derived. The missing variable is treated as being dependent on the other variables and hence can be estimated using the regression equation. A polynomial equation may not be sufficient to model non-linear systems [7], hence more advanced methods are required to model such non-

linear systems. The methods presented in chapter 2 for missing data estimation using a neural network can be thought of as a complex regression model used to model non-linear systems. The advantages and disadvantages of this method are explained further in chapter 2.

1.2.4 Expectation Maximization

The Expectation Maximization (EM) Algorithm is an iterative process that consists of 2 steps [11]. The first step, known as the expectation (E) step, computes the value of the complete data log likelihood based upon the complete data cases and the algorithm's best guess as to which are the best statistical functions for the specified model. The second step, known as Maximization (M) step involves substituting expected values for the missing data obtained in the E step so as to maximize the likelihood function. The E and M steps are repeated iteratively until convergence is obtained [11]. The EM algorithm is thought to be a statistically sound method for estimating missing values [7]. The disadvantage however is that the algorithm does not add any uncertainty component to the estimate data [7].

1.3 Confidence Measurements

The second aim of the research in this dissertation is to give a confidence measure when estimating a person's HIV status. The confidence measure can either be given in terms of confidence intervals, predictive certainty or in terms of standard deviations [12-14]. These measures are discussed and explained in the following sections.

1.3.1 Confidence Intervals

Instead of estimating a particular value for a certain variable, an interval in which the estimate is likely to be in is given. The interval shows that the value of the variable is not precisely known but it is within a certain range [13]. The confidence interval that is estimated can be thought as a value plus an error range. The confidence interval may be computed by calculating the mean of the set of possible estimates and adding an approximate error value to the mean to obtain the limits of the confidence interval. Confidence Intervals are most suitable when estimating continuous value variables. In chapter 4 confidence intervals are computed for the set of possible values obtained from the optimization techniques for the computational intelligence

missing data estimation method. An advantage of having confidence intervals is that one knows a range of probable estimates. A disadvantage of such a measure is that sometimes the intervals are too large and hence the estimate can be a large range of possible values.

1.3.2 Predictive Certainty

Predictive certainty is the measure of frequency of the most likely estimate [13]. This measure is calculated by determining the percentage of the most dominant estimate from a set of possible estimates. Predictive certainty is more suitable for categorical variables. The use of predictive certainty measure is used in chapter 5. The predictive certainty measure can be thought of as the probability of the given estimate of being the correct estimate. It is disadvantageous to use this measurement for variables that have continuous values because the ranges of possible values have to be broken into smaller ranges that can be thought of as different categories. Defining the division boundaries must be done by intuition on the range of the variable.

1.3.3 Standard Deviations

Given a set of possible estimates, it is possible to compute the mean and standard deviation of the set. One can define the confidence of the estimate as the percentage of estimates that lie within one standard deviation of the mean [12, 14]. The mean is also taken as the estimate value. This method is more suitable for variables that have continuous values. This measure of confidence has been used in chapter 4 for the set of possible estimates. A disadvantage when using this method is that it is possible that there is a large standard deviation and hence the estimate can be a range of values.

1.4 Antenatal Dataset

The data used for the investigations carried out in this dissertation is from the South African antenatal seroprevalence survey [15]. The survey is completed by pregnant female patients that attend the antenatal clinics. The data from the survey has been made available by the South African Department of health. The survey has been taken annually from the year 2000.

The particular dataset used for the investigations is from the year 2001. Further details on the data and the preprocessing of data are further discussed in the following subsections.

1.4.1 Variables

The 2001 dataset has 9 demographic variables that are named in table 1.1

Table 1.1 Summary of Data fields for the HIV Dataset

Variable	Type	Range
Province	Integer	1-9
Race	Integer	1-5
Age	Integer	14-50
Education	Integer	1-4
Gravidity	Integer	1-5
Parity	Integer	0-5
Father's Age	Integer	14-50
HIV Status	Binary	0 or 1
RPR Test	Binary	0 or 1

Each of the variables is further explained as follows:

- Province is the provincial location of the patient. The patient resides in 1 of the 9 provinces of South Africa. These provinces are Gauteng, Kwazulu Natal, Limpopo, North West, Orange Free State, Western Cape, Eastern cape and Mpumalanga.
- The race of the patient is either African, White, Coloured or Asian.
- Age refers to the age of the pregnant female patient attending the antenatal clinic.
- Education refers to the maximum level of education the patient has attended. The maximum education level can either be none, primary, secondary or tertiary.
- Gravidity refers to the number of times the patient was previously pregnant.
- Parity refers to the number of children of the patient that are currently alive.
- Father's Age indicates the age of the patient's partner.
- HIV status is the HIV status of the Mother. The binary 1 indicates the mother tested HIV positive and the 0 indicates that the mother is HIV negative.

- RPR indicates the Rapid Plasma Reagin (Syphilis) test results of the patient. The binary 1 indicates that the patient has tested positive for the syphilis test and 0 indicates that the patient has no syphilis.

1.4.2 Data Preparation

Before using the data for neural network training, it is essential that that the data is preprocessed so that it is useful to model the dynamics of the system. The data preprocessing involved removing outliers, randomizing data, normalizing data and encoding certain variables.

Records with outliers for the age and age of husband were removed from the dataset. The age and age of father variable was restricted between 14 to 50 years.

The dataset supplied by the South African Department of health was sorted according to the provincial location. Hence the data was randomized by re-indexing each entry in the database. The re-indexing of the data is necessary so that the neural networks to be trained cover the dynamics of the entire country and not just a select few provinces.

The age, age of father, gravidity and parity variables were normalized between 0 and 1. Normalization is required so that all the variables are within the same range and not to bias the neural network. The normalization of variables is done as follows:

$$x_{norm} = \frac{x - x_{min}}{x_{max} - x_{min}} \quad (1)$$

where x_{max} is the maximum possible value for variable x and x_{min} is the minimum value for the variable x.

Variables such as province, race and education were encoded using the unary coding method. The example of how this encoding was done for the provincial variable is shown in the table 2. Similarly the education and race variables were encoded using 4 bits each.

Table 1.2 The Unary Coding scheme used for the Province Variable

Province	Integer Value	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7	Bit 8	Bit 9
Gauteng	1	1	0	0	0	0	0	0	0	0
Kwazulu Natal	2	0	1	0	0	0	0	0	0	0
Mpumalanga	3	0	0	1	0	0	0	0	0	0
Limpopo	4	0	0	0	1	0	0	0	0	0
North West	5	0	0	0	0	1	0	0	0	0
Northern Cape	6	0	0	0	0	0	1	0	0	0
Western Cape	7	0	0	0	0	0	0	1	0	0
Eastern Cape	8	0	0	0	0	0	0	0	1	0
Orange Free State	9	0	0	0	0	0	0	0	0	1

1.5 Modeling the HIV Pandemic

In 1982 Root-Berstein [16] defined the Acquired Immunodeficiency Syndrome (AIDS) for unusual immune system failure. It was then found that the Human Immunodeficiency Virus (HIV) was identified as the cause of AIDS. Besides identifying the virus, much research has been done to better understand the virus. The HIV virus has already claimed more than 20 million lives by the end of 2007. The HIV/AIDS virus has spread rapidly in South Africa which currently has the highest prevalence rate in the world. Some of the research on this problem includes investigating the causes of the HIV virus [17, 18], predicting the HIV status for risk analysis purposes and to better understand the risks of such a virus [19, 20]. In the field of bioinformatics HIV classifications using neural networks is presented in [18, 20-22].

The research in this dissertation focuses on investigating whether demographic and social characteristics influence the risk of HIV infection. HIV is predominantly transmitted sexually and hence, the person's HIV status is dependent on their social setting. In [23] it is said that social factors affect the risk of exposure to the virus and the probability of transmission. Hence, it is

necessary to use the sociological, cultural and economic factors in addition to biological factors to model the virus.

Using patient data to better understand and model the HIV epidemic is not uncommon. Work by Knorr and Srivastava [24] was done to model the intracellular and intercellular scale HIV dynamics of a person using patient data [24]. Lurie et al (1992). developed a decision analysis model for HIV testing using health workers and hospitals patient information [25]. Other models of the HIV virus that are based on patient data can be found in [26-29].

The research done in this dissertation uses the HIV model developed by Leke [23]. This model is based on the computational intelligence method for missing data estimation that is presented in [7]. The aim is to determine the confidence of predicting a patients HIV status using the demographic properties that were introduced in the previous section. In addition an investigation will be done to determine how the demographic properties influence the certainty of the prediction.

1.6 Neural Networks

A neural network (NN) is an information processing system that is inspired by the way the biological nervous system operates. Hence a neural network process information in similar manner to how the brain would process information [30]. Neural Networks can be thought of as a machine that is designed to simulate a particular way the human brain performs a particular task [30, 31].

NNs have gained popularity for pattern recognition as opposed to using conventional statistical theory. These neural networks have been used to model any kind of system that may be linear or non-linear. The NNs in this dissertation are used for the missing data estimation which is essentially classifying a patients HIV status using the demographic properties.

There are a variety of NN architectures and these include [30]:

- Multi layer Perceptron (MLP)
- Radial Basis Function (RBF)

- Recurrent Neural Network (RNN)
- Hierachial Mixture of Experts (HME)
- Self organizing maps (SOM)
- Hybrid Neural Network (HNN)

The research in this dissertation makes use of the MLP neural network. The MLP neural network consists of the input, hidden and output layer with each layer having a set of neurons or nodes. The neurons of each layer are interconnected to the proceeding layers neurons by weights. 2-Layer MLP NNs have the simplest architecture with a single layer of hidden nodes. Figure 1.1 illustrates the MLP neural network with its various components. MLP uses conventional feedforward optimization methods that are stable [30]. The MLP architecture is limited to 1 hidden layer because of the universal approximation theorem, which states that a 2-layer MLP NN is suitable enough to model any non-linear system and increasing the number of hidden layers becomes redundant [30].

The NN Matlab toolbox Netlab [31] is used for NN implementations in this dissertation.

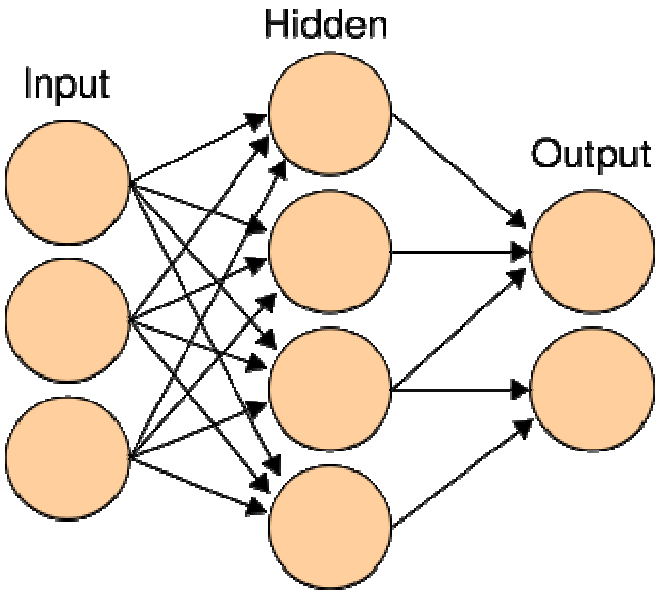


Figure 1.1Illustration of a single layer MLP Neural Network

1.7 Outline of Dissertation

As mentioned earlier the aim of the research is to determine the confidence of the predicted HIV status of a patient. The research aims to do this by using the existing computational intelligence missing data estimation method to estimate the patients HIV status. The confidence of the estimate is obtained using a set of possible estimates and calculating the percentage of the most dominant estimate. This means by calculating the percentage of the most dominant estimate present in the possible estimates. This percentage value will be referred to as the predictive certainty and this can be taken as the confidence of the estimate. In addition in chapter 6 the demographic influences of HIV/AIDS virus will be investigated for cases that could be estimated with a high predictive certainty. A brief outline of the dissertation is given below:

Chapter 2 provides a background on the computational intelligence method for missing data estimation. The Autoencoder neural network and genetic algorithm optimization method used for such estimation methods are also discussed.

Chapter 3 presents an investigation into finding a suitable autoencoder neural network in order to obtain good estimation accuracy for the HIV status of the patient. The results and conclusions of the investigation are then presented.

Chapter 4 gives a background on the Markov Chain Monte Carlo Metropolis Hastings algorithm and the Simulated Annealing optimization techniques. These optimization techniques are used with the best Autoencoder neural network obtained in the investigation in chapter 3 so as to obtain a set of possible estimates. The methods for determining the confidence of the estimates are presented. Results and conclusions that influence the methodology of investigation for the chapter 5 are also presented.

Chapter 5 provides a brief background on ensemble based systems for classification. In this chapter different methods of obtaining an ensemble for the modeling the HIV/AIDS pandemic using the demographic properties are investigated. A suitable method of obtaining an ensemble

that is capable of giving relevant confidence information by means of predictive certainty is obtained by using the Bayesian Framework for neural network training.

Chapter 6 presents an investigation to determine the demographic influences for classifying the HIV status of the patient. The investigation is done by investigating the change in predictive certainty when changing the different demographic properties.

Chapter 7 summarizes the findings of the research and presents suggestions for future work.

1.8 List of publications

The following publications were submitted and accepted while doing the research for this dissertation:

1. Jaisheel Mistry, Fulufhelo Vincent Nelwamondo and Tshilidzi Marwala, "Investigation of Autoencoder Neural Network accuracy for computational Intelligence Methods to estimate Missing Data", *IASTED* International Conference on Simulation and Modelling, Botswana, September 2008, pp 121

2. Jaisheel Mistry, Fulufhelo Vincent Nelwamondo and Tshilidzi Marwala, "Estimating Missing Data and Determining the Confidence of the Estimate Data", IEEE *International Conference on Machine Learning and Applications* (*ICMLA*' 08), San Diego, December 2008, accepted as a 4 page short paper for proceedings of the ICMLA conference 2008

3. Jaisheel Mistry, Fulufhelo Vincent Nelwamondo and Tshilidzi Marwala, "Investigating a Predictive Certainty measure for Ensemble Based HIV Classification Systems", 6th International Conference on Computational Cybernetics (*ICCC* 2008), Slovakia, November 2008, accepted for the proceedings of the ICCC conference

4. Jaisheel Mistry, Fulufhelo Vincent Nelwamondo and Tshilidzi Marwala, “Investigating Demographic Influences for HIV Classification using Bayesian Neural Networks”, 15th International Conference on Neuro-Information Processing (ICONIP), New Zealand, November 2008, accepted for 2 page paper in the conference proceedings and under review for lecture notes publication

Chapter 2: Missing Data Estimation using Computational Intelligence

2.1 Introduction

As mentioned in the previous chapter the aim of the research is to determine the confidence of predicting the HIV status of a patient. This chapter gives a background on the computational intelligence method used for estimating missing data which is used for doing the HIV status classification. This computational intelligence method is the method proposed by Abdella in [7] and makes use of autoassociative neural networks and genetic algorithms.

The concept of missing data mechanisms is explained in the next section. Section 2.3 discusses the Autoencoder and Autoassociative neural networks that are used in the computational intelligence missing data estimation method. The Genetic Algorithm optimization method is presented in section 2.4. The computational intelligence missing data estimation method is presented in section 2.5.

2.2 Missing Data Mechanisms

Before estimating missing data, it is important to understand why the data is missing. The reason for the data being missing is known as the missing data mechanism. The three main types of missing data mechanisms are Missing at Random (MAR), Missing Completely at Random (MCAR) and the Not Missing at Random (NMAR or non ignorable case) [2]. These mechanisms are explained further.

2.2.1 Missing at Random

A MAR data is missing value that has a probability of being missing in field X and this missing data is dependent on other fields in the database but not on field X itself. A simple example is that the person's education level might be missing in the database because of age, rather than because of the other educational levels in the database.

2.2.2 Missing Completely at Random

MCAR occurs if the probability of a missing value that belongs to field X is not related to the field X itself and not to any other field in the dataset. A patient's husband's age may be missing due the fact that the patient does not know. The patient's husband's age is not dependent on any of the other variables in the database.

2.2.3 Non- Ignorable

Non ignorable is also known as informatively missing or not missing at random. The non-ignorable case is when a missing data in field X is dependent on field X itself. A simple example is that the patient has not gone to school and hence does not want to fill in the field for the maximum education achieved.

In this dissertation the computational intelligence missing data estimation method is used to estimate the patients HIV status. This method can be used to estimate missing data regardless of the missing data mechanism. However, in this research it is assumed that the HIV status is missing at random.

2.3 Autoencoder and Autoassociative Neural Network

Autoassociative Neural Networks are neural networks that are trained to produce the output vector to be the same as the input vector. Autoencoder neural network is the same as autoassociative neural network and is derived from the term autoassociative neural network encoder. This implies that these neural networks have fewer hidden nodes than input nodes. The network therefore predicts the inputs as outputs, whenever an input is presented. Figure 2.1 illustrates an autoassociative MLP neural network. Applications of autoassociative neural networks can be found in [8, 32-36]. Autoencoder Neural Networks have fewer hidden nodes than input nodes and hence this characterizes a bottle-neck. This implies that the inputs to the network are projected to a smaller space resulting in redundant data being removed [10, 32]. The bottle-neck can be thought of as reducing dimensionality of the input data by taking account of the covariance and correlation of the various dimensions of data [32].

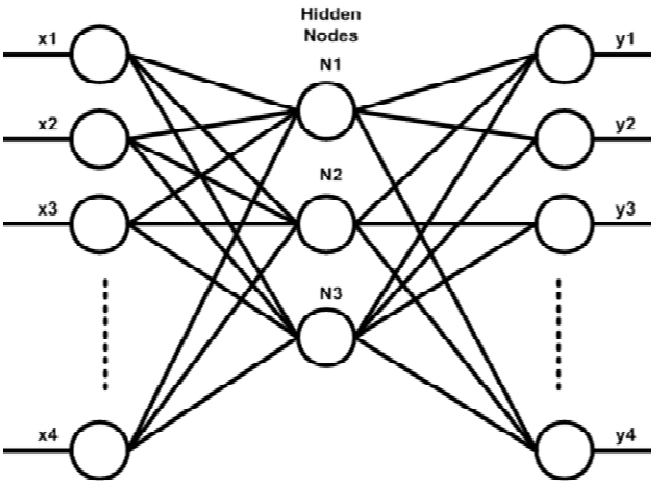


Figure 2.1Autoassociative Neural Network

2.4 Genetic Algorithm

Genetic algorithms (GA) are used to find optimum solutions to various difficult problems such as wire routing, scheduling, adaptive control, game playing, cognitive modeling, transportation problems and database query optimisation using principles of evolutionary biology [37-39]. GAs make use of biologically derived techniques such as inheritance, mutation, natural selection and crossover to discover optimal solutions to these difficult problems [38]. GA is used to find the maximum of a certain evaluation function. The maximum of the evaluation function is found by using the following procedure [37]:

1. Create an initial population of genes (possible solutions) by choosing random numbers
2. Using the evaluation function, evaluate the population of genes
3. Create a new population by selecting the fittest genes from the old population
4. Apply some genetic process such as mutation and crossover to create a new population
5. Using the evaluation function evaluate the new population
6. Repeat steps 3, 4 and 5 for a certain amount of generations

The result of running such an algorithm would give a population of possible optimal solutions as well as the fitness of each of the solution. If the variable parameters such as rate of crossover, type of crossover, initial population size and number of generations are correctly chosen it is found that the entire population converge to the global maximum of the evaluation function [37]. An advantage of such stochastic optimization technique is that the evaluation can be nonlinear because the gradient of the evaluation function is not required as is the case with conjugate gradient methods [38]. A disadvantage of using GAs is that they require computational power and time for each generation to take place.

2.5 Computational Intelligence Missing Data Estimation

CI methods used for estimating missing data are a well researched topic [1, 40-42]. A review of these methods is given in [1]. Figure 2.2 illustrates the general structure of the computational intelligence methods that are used to estimate missing data. The figure 2.2 illustrates that the database has data that have known data denoted by X_K and data that are unknown or missing data denoted by X_U. The system optimization process (see figure 2.2) would use an optimization technique to optimize an evaluation function. The evaluation function minimizes the error between the estimated unknown inputs (X'_U, X_K) and the output from the identification system F $\{X_U, X_K\}$. The identification system is a system that recognizes patterns or interrelationships of variables in the data. This identification system is obtained by training on a set of complete data; hence the dynamics of the system are identified. When a minimum error is calculated the corresponding X'_U is stored as the estimated value.

The error of the predicted output of the identification system is calculated as follows:

$$\text{error} = \begin{bmatrix} \overrightarrow{X_K} \\ \overrightarrow{X_U} \end{bmatrix} - F\left\{\begin{bmatrix} \overrightarrow{X_K} \\ \overrightarrow{X_U} \end{bmatrix}\right\} \qquad (1)$$

where the matrix $\begin{bmatrix} \overrightarrow{X_K} \\ \overrightarrow{X_U} \end{bmatrix}$ is representative of a single record of known data $\overrightarrow{X_K}$ and unknown data $\overrightarrow{X_U}$.The vector notation is employed to illustrate that there are a set of missing values and a set of unknown values. In order to keep the magnitude of the error positive, the error is squared to become:

$$\text{error} = \left(\begin{bmatrix} \overrightarrow{X_K} \\ \overrightarrow{X_U} \end{bmatrix} - F\left\{ \begin{bmatrix} \overrightarrow{X_K} \\ \overrightarrow{X_U} \end{bmatrix} \right\} \right)^2 \quad (2)$$

Note that the GA is to find the minimum error and hence the evaluation function is given as follows:

$$\text{evaluation function} = -\left(\begin{bmatrix} \overrightarrow{X_K} \\ \overrightarrow{X_U} \end{bmatrix} - F\left\{ \begin{bmatrix} \overrightarrow{X_K} \\ \overrightarrow{X_U} \end{bmatrix} \right\} \right)^2 \quad (3)$$

so that the genetic algorithm used is to find the maximum of the evaluation function.

This method of missing data estimation is preferred to the use of classifiers because of the feedback mechanism employed when using the optimisation process [10]. The benefit of this feedback is that the system prediction is less susceptible to noisy data sets [1]. Noisy data sets could be data sets that should be treated as outliers to the system. The use of this method for missing data estimation has the ability to estimate more than one missing field from a record [7]. Another advantage of this method is that a single multi classifier is needed for the identification system as shown in figure 2.2.

There is a range of non-linear modeling methods that are used for system identification (see figure 2.2). These tools include neural networks , fuzzy logic, neuro- fuzzy systems, rough-set models, neuro-rough models, Principal Component Analysis (PCA) compression mimic neural networks, Support Vector Machines (SVM), Decision Tree Algorithms and Hybrid Neural Network architectures [10]. The use of Genetic Algorithms, Particle Swarm Optimization and Simulated Annealing are some of the optimization methods used for the optimization process [43].

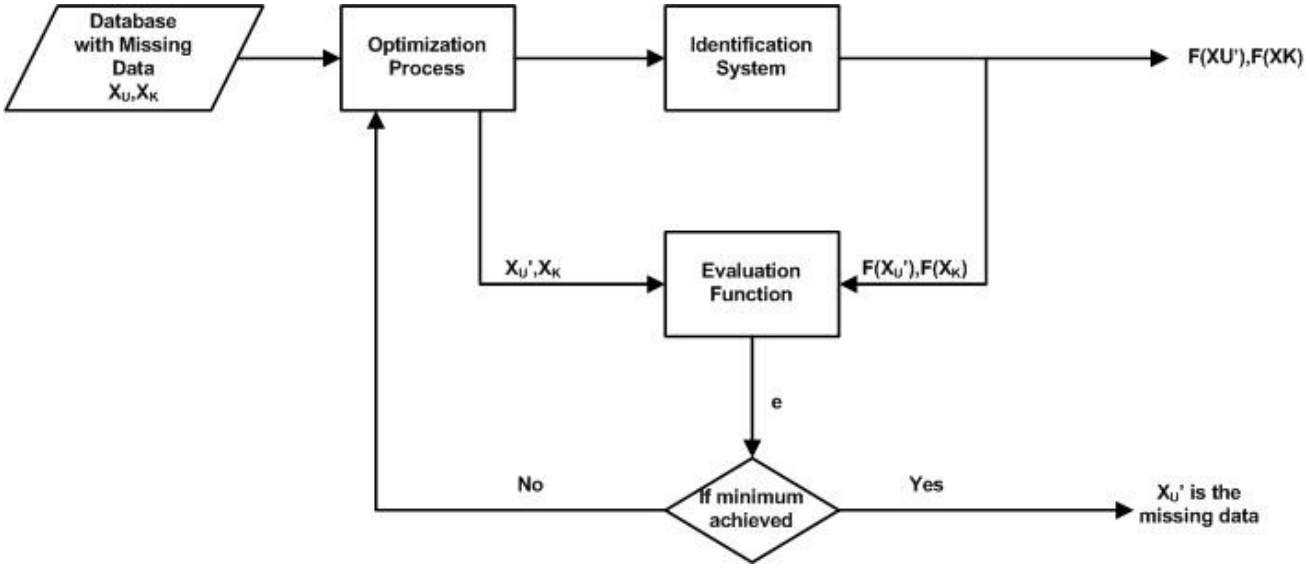


Figure 2.2 General Computational intelligent method used to estimate missing data

2.6 Conclusion

A brief background on missing data mechanisms was given. The computational intelligence missing data estimation method is used to estimate missing data that belongs to any one of these mechanisms. The autoassociative and Autoencoder neural network methods used for the missing data estimation method were explained further in this chapter.

Chapter 3: Autoencoder Neural Network for HIV Classification

3.1 Introduction

In chapter 2 autoencoder neural networks were introduced. Autoencoder neural networks are used to capture interrelationships between the different variables. The Autoencoder neural network is used in the computational intelligence method for estimating missing data. It is important to investigate what impact the accuracy of the Autoencoder Neural Network has on the accuracy of the missing data estimation method. In this chapter the relationship of the accuracy of the autoencoder neural network and the overall accuracy of the computational intelligence missing data estimation method is investigated. The impact of over-trained and undertrained autoencoder neural networks on the estimation method is also demonstrated.

3.2 Network Training

An investigation was done to investigate the number of hidden nodes that are required for the autoencoder neural network. The investigation was carried out by training the neural network and testing by calculating the average percentage error between the estimated value and the actual value. The average percentage error is calculated for the variables age, father's age, parity and gravidity. The average percentage accuracy of predicting the other binary inputs are also calculated. The simulation was run by training the autoencoder neural networks for 2500 cycles with a training data set of 6000 records. The accuracy and error percentages were calculated using a test dataset consisting of 2000 records. These simulations were done thrice to investigate variance of results. These results are presented in the figure 3.1 and figure 3.2.

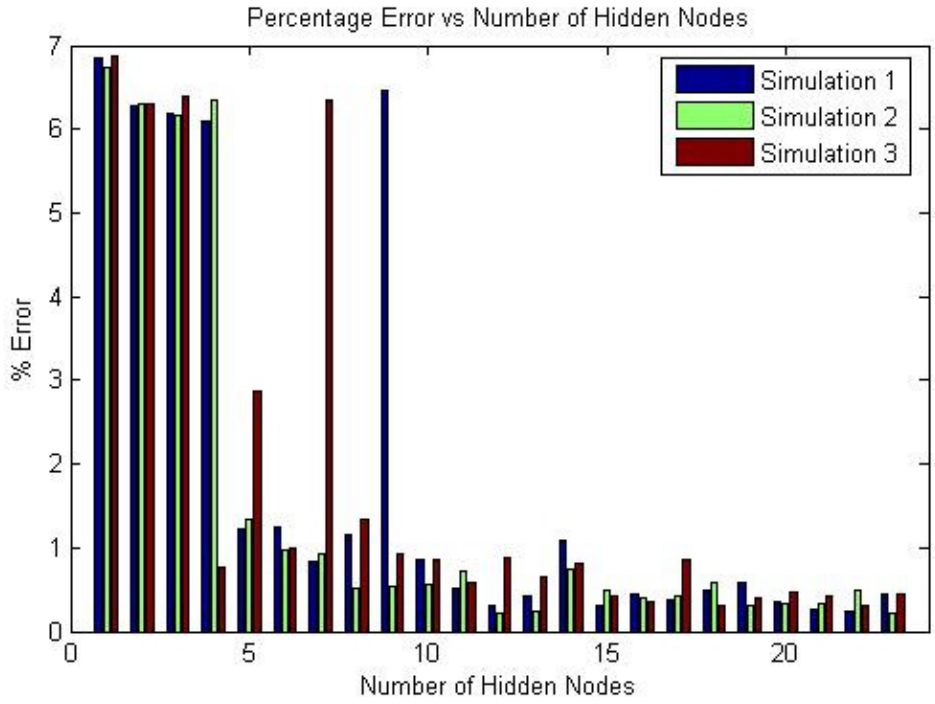


Figure 3.1 Percentage Error for Neural Network Training for Hidden Nodes Ranging from 1 to 22

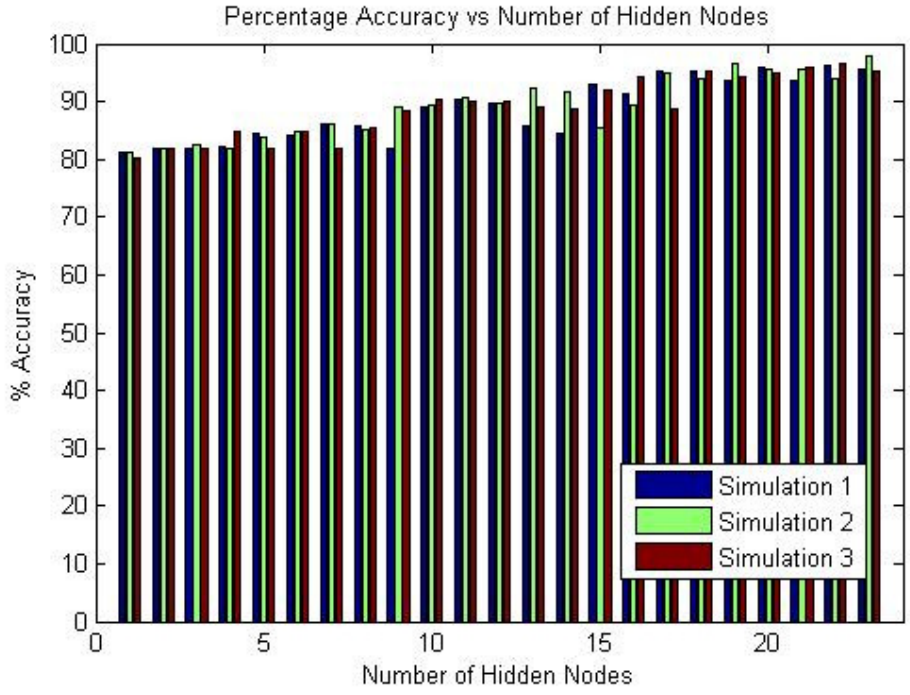


Figure 3.2 Accuracy for Neural Network Training for Hidden Nodes Ranging from 1 to 22

From figure 3.1 and figure 3.2 it is evident that the autoencoder neural network should have a minimum of 10 hidden nodes because the accuracies are higher than 80 percent. The

percentage error is significantly low when there are more than 10 hidden nodes. The variance in error and accuracy values for the three simulations are lower when there are 10 or more hidden nodes. Furthermore the percentage error is significantly low when there are more than 10 hidden nodes. The variance in error and accuracy values for the three simulations are lower when there are more than 10 hidden nodes.

In order to ensure that the neural networks are not over-trained, it is necessary to find the number of training cycles. The figure 3.3 and figure 3.4 illustrates the average percentage error and average percentage accuracy of the various autoencoder neural networks as the number of training cycles increases from 500 to 3000. It is evident that the autoencoder neural network models need to be trained with a minimum of 1500 cycles. The variance in error and accuracy is much less once the number of training cycles exceeds 1800 training cycles.

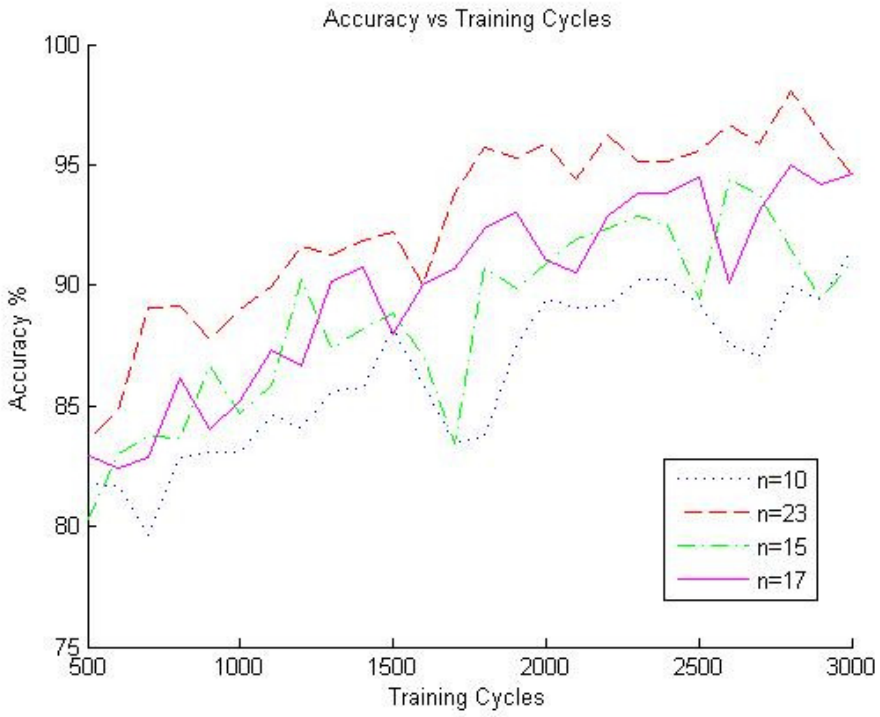


Figure 3.3 Training Accuray for diiferent amount of training cycles

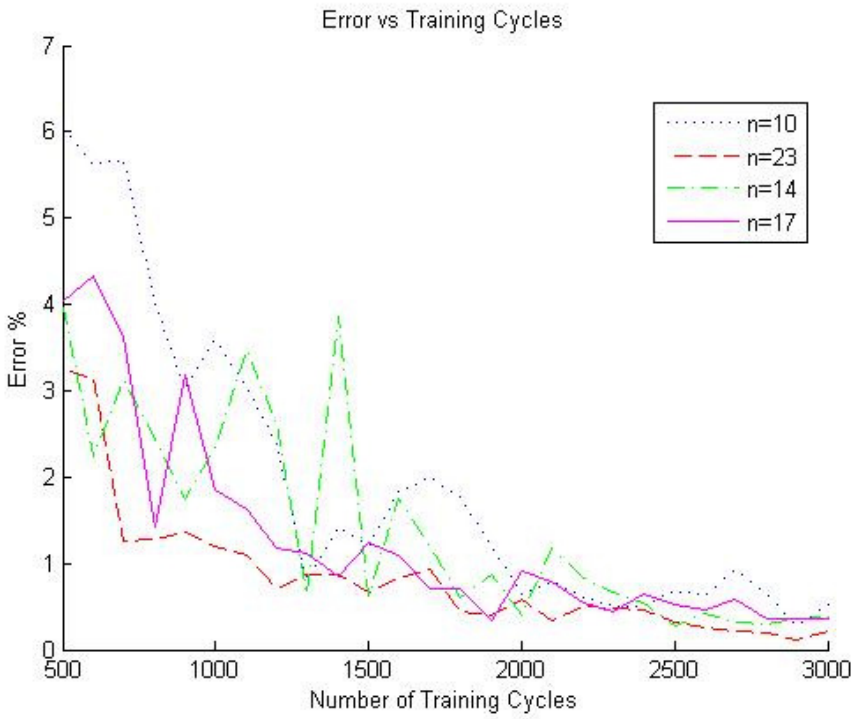


Figure 3.4 Training error for different amounts of training cycles

3.3 Experiment Methodology

To investigate the accuracy of the missing data estimation method that is presented in figure 2.2 the genetic algorithm is used for the optimization process. The Neural Networks have been trained to prediction accuracy levels greater than 80%. A special test data set consisting of 1000 records were used for the investigation. The HIV status field was set to be missing for all the records in the test set. This was done because the most common field that was missing in the database was the HIV status of the patient. By investigation it was found that the genetic algorithm should be set to run for 40 generations using a population size of 100 genes. The simple crossover method with a crossover probability of 60% was used in the genetic algorithm. For the CI missing data estimation method, the reader is referred to Chapter 2.

3.4 Results

The investigation was carried out and table 3.1 illustrates some of the autoencoder neural networks that were used for the investigation. Figure 3.5 illustrates the accuracy of the missing data estimation method for Autoencoder Neural Networks with different accuracies. In addition

table 3.2 gives the missing data estimation accuracy for some of the Autoencoder Neural Networks that are under-trained or over-trained.

Results indicate that the autoencoder neural network accuracy must be greater than 80% to yield estimation accuracy greater than 55%. The best results achieved are 65.3% accuracy for an autoencoder neural network which has 95.1% prediction accuracy. There is no direct relationship between the missing data estimation accuracy and the autoencoder neural network prediction accuracy because the neural network with 95% prediction accuracy was more accurate than a neural network with a prediction accuracy of 99%.

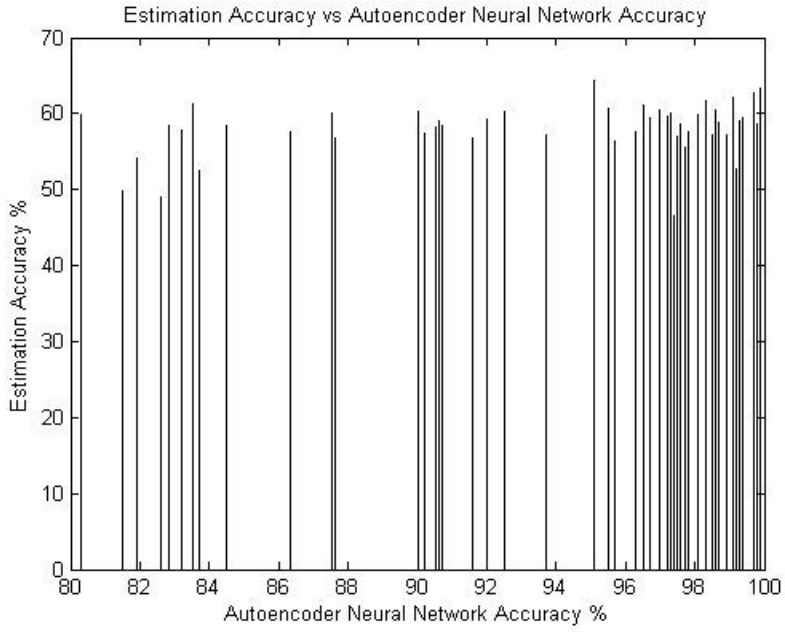


Figure 3.5 Distribution of Estimation Accuracy for Neural Networks trained to certain testing accuracy

Table 3.1 Autoencoder Neural Networks used for the Investigation

Hidden Nodes	Training Cycles	Testing Accuracy (%)	Estimation Accuracy (%)
14	2700	82.8	58.5
15	1900	87.5	60
16	2800	99.7	62.7
17	2100	95.1	65.3
18	1900	93.7	57.3
19	2300	84	63.1
20	2000	97.5	57
21	1700	95.5	60.7
22	2300	83.2	57.8
23	2500	99	63.3

Table 3.2 Over-trained and Under-trained Autoencoder Neural Networks

Hidden Nodes	Training Cycles	Testing Accuracy (%)	Estimation Accuracy (%)
15	1500	82	39.1
17	4000	100	45
18	1500	84.9	49.1
19	3500	100	44.6
22	3000	97.4	46.5
23	3500	100	30

The test data set used for the investigation was unbiased in that there were 500 cases of patients who were HIV positive and there were another 500 patients who tested HIV negative. Using zero substitution for such a problem would result in having 50 % estimation accuracy. The

results with accuracy of 65.1% were obtained by predicting 333 HIV positive patients and 318 HIV negative patients correctly.

The results obtained by using the autoencoder neural network with 95.1% prediction accuracy were better than that of the autoencoder neural network with 99% prediction accuracy. This result shows that there is no direct relationship between the autoencoder prediction accuracy and the missing data estimation accuracy. Hence it is essential to choose an autoencoder neural network that is properly trained and tested on test set. Future work for improving the estimation will be to measure the confidence of the predicted results. This can be done by calculating a sample set of possible estimate values and calculating the variance of the solution set.

3.5 Conclusion

The relationship between the Autoencoder Neural Network accuracy and the missing data estimation accuracy was investigated. The tests for the investigation were done using the South African antenatal seroprevalence database. It was found that the higher the accuracy of the autoencoder neural network, the better the estimation. It is however important to ensure that the autoencoder neural network be trained to achieve maximum accuracy but at the same time ensure the neural network is not over-trained because of the negative impact on the estimation method. The best result achieved was an estimation accuracy of 65% when the Autoencoder Neural Network was tested to have an accuracy of 95%. This research further justifies the use of computational intelligence methods for estimating missing data.

Chapter 4: Optimization Methods

4. 1 Introduction

In chapter 3 we used the computational intelligence missing data estimation method to estimate the HIV status of the patients. The computational intelligence method made use of the autoencoder neural network and the genetic algorithm optimisation technique. The most optimal solution from the genetic algorithm was chosen as the estimate for the HIV status. The genetic algorithm is an evolutionary algorithm and a population of possible solutions is given. In this chapter the set of optimal solutions from the genetic algorithm is analysed statistically to see if any useful confidence information can be extracted. In addition, the popular stochastic optimisation of Simmulated Annealing will also be used to obtain a set of possible estimates.

The research in this chapter primarily focuses on determining whether the set of optimal solutions obtained for the optimisation process for the missing data estimation method is useful for extracting confidence information. The confidence limits will be determined by determining the standard deviation from the optimal solution of the set of possible solutions. The Simulated Annealing and Genetic Algorithm optimisation techniques are used to determine whether the optimisation technique influence the confidence measurements. The best autoencoder neural network obtained in the previous chapter will be used for the identification system for the missing data estimation method found in figure 2.2.

The simulated annealing optimisation algorithm is an accelerated algorithm for the Markov Chain Monte Carlo Metropolis Hastings algorithm. The following section gives a brief introduction on Markov Chain Monte Carlo and this is followed by an explanation of the Simulated Annealing optimisation method. This is followed by the method of investigation and finaly the results are presented and disscussed.

4.2 Markov Chain Monte Carlo

Markov Chain Monte Carlo is used extensively for probabilistic machine learning [44]. Markov Chain Monte Carlo (MCMC) consist of a large class of sampling algorithms [44]. These sampling algorithms have been used in the fields of science, econometrics, physics and computer science

over the past 2 decades [4, 45, 46]. MCMC algorithms are used to solve integration and optimization problems that have large dimensional spaces.

Some of the popular problems that are solved using MCMC algorithms are [46]:

1. Bayesian inference and learning. This is used extensively for training neural networks.
2. Statistical Mechanics
3. Optimization problems
4. Penalized likelihood model selection

The aim of the MCMC algorithms is focused on drawing a set of samples from a target density defined in a high dimensional space. The set of samples can be obtained by using all possible values for the input parameters. This method is referred to as a Monte Carlo Integration. For High dimensional problems this is not feasible and hence approximations to the Monte Carlo Integration must be applied. MCMC algorithms are all algorithms that are approximations of performing the Monte Carlo integration problem. The Markov Chain property of MCMC algorithms refers to the fact that new samples are selected based on the previous set samples [46].

When solving optimization problems using the MCMC algorithms, the objective is to obtain samples that give most optimal solutions. Hence samples are initially selected randomly and iteratively new samples are obtained by updating the previous samples. New samples that are less likely to be optimal are rejected using certain probabilistic conditions. Hence this rejection sampling is important for obtaining optimal solutions [46]. The following section discusses the MCMC Metropolis Hasting algorithm which is a rejection sampling method used for the simulated annealing optimization algorithm.

4.3 Simulated Annealing Optimization

Simulated annealing was developed as an approach to finding a maximum of a complex function which may have local maxima and minima [44]. The objective of the algorithm is to find the global maximum. This algorithm avoids finding the local maxima as in the case with hill climbing algorithms. The analogy of the algorithm is derived from the annealing of the crystal as

the temperature decreases. Initially there is lots of movement and hence large amounts of the domain space is searched and as temperature cools the movement minimizes and hence smaller search spaces are explored.

As discussed in the previous section Simulated Annealing is an extension of the MCMC Metropolis Hasting Algorithm. The MCMC Metropolis Hasting algorithm is used to obtain a set of possible optimal solution to an unknown distribution using the following steps [46]:

1. Choose a random sample θ .
2. Obtain a new sample θ^* using θ. This can be done $\theta^* = \theta + \theta\varepsilon$ where ε is a small random number ranging between 0 and 1.
3. Compute the fitness of the two possible solutions using the evaluation function $f()$ which is to be maximized.
4. Compute $\alpha = \min(1, \frac{f(\theta^*)}{f(\theta)})$
5. Accept θ^* with probability α (if $\theta^* > \alpha$) and add it to the set of samples. If θ^* is rejected (if $\theta^* < \alpha$) then add θ again to the set of possible optimum solutions.
6. Repeat steps 2 to 5 until you have a sufficient amount of samples
7. Use the mean of the sample set as the optimal solution.

Steps 2 to 5 are repeated for N number of times to obtain a sample set with N solutions. Often the first few samples from the sample set are rejected. The period for which the samples are rejected is known as the burn in period.

The simulated annealing algorithm is an accelerated version of the MCMC Metroplis Hasting algorithm. The samples in the solution are made to converge quicker by changing the acceptance criteria α in step 3 of the algorithm. The acceptance criteria is changed to $\alpha = \min(1, \left[\frac{f(\theta^*)}{f(\theta)}\right]^{1/T(t)})$ where $T(t)$ is the number of samples currently in the sample set. $T(t)$ is often referred to as the temperature [44]. The idea is that initially the samples are accepted with reasonable high probability but as time $T(t)$ increases the acceptance probability decreases making it difficult for new samples to enter the solution set [45, 46].

4.4 Experiment Methodology

The aim of the investigations done in this chapter is to:

1. Determine whether useful confidence information can be gathered from the set of optimal solutions obtained from the optimization method that is used in the computational intelligence missing data estimation method presented in figure 2.2.
2. Determine whether the optimization method used has any impact on the confidence measurements.

The computational intelligence missing data estimation method presented in figure 2.2 will be used coupled with the genetic algorithm optimization technique and the simulated annealing optimization technique. The best autoencoder neural network obtained in chapter 3 will be used for the identification system as indicated in figure 2.2. The best autoencoder neural network is the one with 17 hidden nodes that achieved 65% accuracy for estimating the HIV status of the patient.

By investigation it was found that the genetic algorithm should be set to run for 40 generations using a population size of 100 genes. The simple crossover method with a crossover probability of 60% was used in the genetic algorithm. The computational intelligence missing data estimation method used is presented in chapter 2.

The simulated annealing algorithm was also used to obtain a set of possible solutions. The simulated annealing algorithm was run for 1000 cycles and the first 100 samples were removed from the sets.

Confidence information is derived from the set of possible solutions by calculating the mean and standard deviation of the sample set. The mean of the sample set is chosen as the optimal solution. The computed standard deviation from each solution set is set as confidence limit. In addition the percentage of samples from the solution set that are within 1 standard deviation from the mean is also computed.

4.5 Results and Discussion

The missing data estimation method achieved 63% accuracy when using the Simulated Annealing algorithm as compared to the 65% accuracy that was achieved in chapter 3 when using the Genetic Algorithm optimization technique.

The confidence limits are calculated to be 1 standard deviation of the set of possible estimates. The confidence limits and percentage of sample set that lie within one standard deviation of the mean are computed for each data record individually. For demonstration purposes, 10 records were selected to illustrate the confidence measurements against the actual solution. The 10 records that were used are illustrated in table 4.1.

Table 4-1Results obtained using the optimization methods

	Simulated Annealing			Genetic Algorithm		
Actual solution	Optimal Solution	Confidence Limits	% within 1 std deviation	Optimal Solution	Confidence Limits	% within 1 std deviation
1	0.69253	0.22428	69	0.87314	0.17881	72
0	0.72814	0.2196	68	0.59428	0.02066	99
0	0.25374	0.21568	70	0.46954	0.04699	95
1	0.91802	0.17476	86	0.77277	0.04154	98
0	0.4604	0.24254	70	0.041462	0.00589	93
1	0.08587	0.17504	86	0.10165	0.02812	98
1	0.89827	0.17904	85	0.5267	0.00526	98
1	0.08593	0.17535	86	0.10843	0.00898	98
1	0.57101	0.23824	70	0.62285	0.01163	98
0	0.91105	0.17574	85	0.773	0.01386	98

In table 4.1 it is evident that the optimal solutions for these two methods were often different. Further it was found that most of the estimates were not close to the binary 0 or 1 value. It was found that the optimal solutions obtained were values ranging between 0.2 and 0.8 depending on the data record being used. The confidence limits for the optimal solutions in some cases are over 0.2 and this is very high considering that the search space is between 0 and 1. The confidence limit of 0.2 is actually 20% of the search space.

From table 4.1 it can be seen that the confidence limits are small when using the Genetic Algorithm and most of the solutions lie within 1 standard deviation of the mean. The Genetic Algorithm converges quicker to the maximum and hence more solutions are closer to the optimal solution.

Furthermore we cannot interpret these values of 0.1 to 0.9 as the probability of a patient being HIV positive because the neural network has never been trained with such probabilities. Essentially Autoencoder neural network should be constrained to only one of two estimates (0 for HIV negative and 1 for HIV positive). Hence it is not necessary to use an optimization technique because the search space for estimating the patients HIV status is small. The optimization of the evaluation function can be done by brute force.

The Autoencoder neural network architecture using the brute force optimization yielded the same results as using standard neural network classifiers. We can also infer that the optimization techniques are not useful in giving a set of possible estimates because they have no domain knowledge.

4.6 Conclusion

Investigations were done to determine whether useful confidence information could be derived from the solution set obtained from the optimization technique used in the computational intelligence missing data method. The Genetic Algorithm and Simulated Annealing optimization methods were used. The confidence limits were set to be the standard deviation from the mean of the set of the optimal solutions. It was found that no useful confidence information can be derived from the solution set because the neural networks were not trained on probability values between 0 and 1. Hence the optimization can be done by brute force to evaluate 1 or 0. It is therefore necessary to use multiple classifiers to determine a set of possible solutions.

Chapter 5: Ensemble Based Neural Network Systems

5.1 Introduction

In chapter 3 we found the prediction accuracy for estimating the HIV status of patients using the autoencoder neural network to be 65%. Due to this low prediction accuracy, it is necessary to determine the certainty of the prediction. A common approach in pattern recognition and machine learning is to determine certainty or confidence by making use of ensembles of classifiers [47].

The research using ensemble classifiers or multiple classifiers has grown in popularity due to the increase in available computational power [47-51]. Using ensembles for decision making is more favorable because ensembles are thought to be more knowledgeable of a domain compared to a single classifier. Ensemble classifiers are favoured for statistical reasons because each classifier has different generalization capabilities [47].

Ensembles classifiers are more capable of classifying complex data sets where there may be an excess or shortage of data points. The divide and conquer approach employed by using ensembles helps in modeling nonlinear systems. A thorough review on ensemble based system is given in [47]. Two key components of ensemble based systems include the method for generates an ensemble and the method for aggregating decisions made by each classifier from the ensemble [47].

Approaches to generating an ensemble include the bagging and boosting approaches which are explained in the next section. Methods for aggregating outputs from ensembles include majority vote selection, undemocratic voting or weighted majority voting.

Hence, in this chapter, ensemble neural networks will be used to classify a patient's HIV status. In addition the predictive certainty of the ensembles will be measured by determining the percentage of the most likely prediction from all the possible neural networks in the ensemble. The bagging, boosting and Bayesian training methods will be used to obtain three different

ensembles. A background of the three previously mentioned training methods will be given. The method of the investigation is then given and the results and discussion are presented. This is followed by the final conclusions.

5.2 Ensemble Generation

A brief background on the bagging, boosting and Bayesian training methods for obtaining ensemble systems are presented.

5.2.1 Bagging

Bagging was introduced by [49] and it is one of the first algorithms for generating ensembles. Bagging is short for bootstrap aggregating. Bagging works by generating classifiers on different sets of data. Hence training data is divided up into subsets of data. These subsets of data are obtained by choosing smaller random samples of data from the training data set. This random sampling is done with replacement so that the subsets overlap one another.

There are variants of the bagging algorithm and these include random forests and Pasting Small Votes[47]. The Pasting Small Votes algorithm works where the large dataset is broken up into smaller bites and the classifiers are obtained by training using the bites. The datasets for bites are chosen randomly (RVotes) or by importance (IVotes) [50].The pasting small votes (IVote) bagging algorithm is used for investigations carried out in this chapter. Its pseudo code is given in figure 5.1.

The importance votes (IVote) works by giving importance to datasets that are misclassified. Hence the subsets consist of datasets that are difficult to classify rather than randomly sampled datasets. The outputs from the ensemble classifiers are aggregated using the majority vote method. The class with majority votes is chosen as the final decision for the ensemble.

Pasting Small Votes (Ivotes) Algorithm

Input:

Training data S with labels $\omega_i \in \Omega = \{w_1, \ldots, w_C\}$ representing C classes;

Weak learning algorithm **WeaklLearn**;

Integer T specifying the number of iterations;

Bitesize M, indicating the size of individual training subsets to be created.

Initialize:

Choose a random sample S_0 of size M from S.

Call **WeakLearn** with S_0, and receive the classifier h_0.

Evaluate h_0 on a validation dataset, and obtain error e_0 of h_0.

For t=1,…,T

Randomly draw an instance x from S according to uniform distribution.

Evaluate x using the ensemble of classifiers E_t and aggregate the outputs using the majority vote

If x is misclassified, place x in S_t. Otherwise place x in S_t with probability p

$$p = \frac{e_{t-1}}{1 - e_{t-1}}$$

Call **WeakLearn** with S_t and receive the hypothesis h_t

Evaluate h_t on validation dataset, and obtain error e_t of h_t. If $e_t > \frac{1}{2}$, return to step 4.

Add h_t to the ensemble E_t.

End

Figure 5.1: Pasting small votes (IVote) bagging algorithm [44]

5.2.2 Boosting

Boosting methods to generate ensembles were introduced by [52]. Boosting algorithms differ from bagging methods in that the smaller subsets are not obtained randomly. The basic principle of boosting is to build a strong classifier out of two weaker classifiers. The stronger classifier is obtained by training on subsets that were misclassified by either of the weak classifiers and datasets where the two classifiers have differing decisions.

A more advanced boosting algorithm is the AdaBoost algorithm and it was introduced in [52]. Each classifier is generated by using a weak learning method on the subset of data. The datasets for subsets used for training are chosen using a distribution. The distribution is updated each round so as to give more importance to misclassified data sets. The pseudo code for the AdaBoost algorithm used in this paper is given in figure 5.2.

AdaBoost Algorithm

Input:

Sequence of N examples $S=[(x_i, y_i)], i=1,...,N$ with labels $y_i \in \Omega, \Omega=\{w_1,...w_C\}$;

Weak learning algorithm **WeakLearn**;

Integer T specifying number of iterations.

Initialize $D_1(i)=\frac{1}{N}., i=1,...,N$

For t=1, 2,…, T:

Select a training data subset S_t, drawn from the distribution D_t.

Train classifier h_t with S_t using **WeakLearn**.

Calculate the error of

$h_t : e_t = \sum_{i:h_t(x_i)\neq y_i} D_t(i)$

Set $\beta_t = e_t/(1-e_t)$

Update distribution

$$D_t : D_{t+1}(i) = \frac{D_t(i)}{Z_t} \times \begin{cases} \beta_t, if\ h_t(x_i)=y_i \\ 1, \quad otherwise \end{cases}$$

Where $Z_t = \sum_i D_t(i)$ is a normalization constant chosen so that D_{t+1} becomes a proper distribution function.

Figure 5.2: AdaBoost Algorithm [10]

The AdaBoost algorithm generates a set of classifiers and combines the result using the weighted majority vote method. Hence classifiers that have lower errors or higher accuracies are given higher weighting when aggregating the outputs from the classifiers. The weighted

Majority Voting method is used to aggregate the outputs of each classifier. For a given instance x the total vote V_j for each class is calculated as follows [47]:

$$V_j = \sum_{t:h_t(x)=w_j} \log \frac{1}{\beta_t}, j = 1,...,C. \quad (1)$$

The class with the majority vote V_j is chosen as the final decision of the ensemble.

5.2.3 Bayesian Approach

Neural Networks can be trained using a variety of methods. Scaled conjugate gradient methods and Quasi-Newton methods are used to obtain the maximum likelihood weight values [31]. An alternate approach to training the neural networks is using the Bayesian Approach. Extensive work has been done by [53, 54] to train neural networks using the Bayesian approach. The problem of identifying the weights (w) is posed in Bayesian form as follows [13]:

$$P(w \mid D) = \frac{P(D \mid w)P(w)}{P(D)} \quad (2)$$

where $P(w)$ is the probability distribution function of the weight-space in the absence of any data, known as the prior probability distribution function, and $D \equiv (y_1, y_2, ..y_N)$ is a matrix containing the output data for the neural network. The quantity $P(w \mid D)$ is the posterior probability distribution after the data has been seen and $P(D \mid w)$ is the likelihood probability distribution function, while $P(D)$ is the normalization factor. Following the rules of probability theory, the distribution of output vector y may be written in the following form [53]:

$$P(y \mid D) = \int P(y \mid w)P(w \mid D)dy \quad (3)$$

The distribution in equation (3) is estimated using Markov Chain Monte Carlo (MCMC) methods [46]. In this dissertation, the Hybrid Monte Carlo (HMC) MCMC method is used. This method has been used quite extensively to solve complex engineering problems [53]. This method makes use of gradient information to reduce random walk behavior to speed up the exploration weight search space [44].The details of this technique, which are fairly abstract, are beyond the

scope of this dissertation but can be obtained, in [53]. The HMC implementation found in [8] was used for determining the weights of the neural network.

A key advantage of using Bayesian training is that a set of neural networks are obtained and training using the Bayesian approach reduces the likelihood of over-training [53].

Multi Layer Perceptron (MLP) architecture is used for generating the ensemble neural networks investigated in this chapter. Considering the unary coding scheme used, the Neural Network has 22 inputs with 1 output which is the HIV status of the patient.

The WeakLearn method is required for the bagging and boosting algorithms discussed in figure 5.1 and figure 5.2. The WeakLearn method is basically a training method that uses conventional neural network training methods. The WeakLearn method requires that the neural network be trained so that it has a prediction error of less than 50%. The Scaled Conjugate Gradient (SCG) training method was used for the WeakLearn method. By experiment it was found that the MLP neural network with 15 hidden nodes is sufficient to obtain 50% accuracy. The output activation function used was set to use the logistic function and the number of training cycles used was set to 70 when using a training dataset of 8000 records. Using the previously mentioned Weaklearn method the bagging and boosting ensembles were generated by setting the T parameter (see figure 5.1) to 100. Hence each ensemble consisted of 100 Neural Networks. The Bitesize M used in the bagging algorithm (see figure 5.2) was set to 60%.

5.3 Experimental Methodology

The Hybrid Monte Carlo (HMC) method was used for training to obtain the Bayesian ensemble of neural networks. The HMC training method gives a set of possible weights for certain number of hidden nodes. The weights were initially trained using the scaled conjugate gradient method to the early stopping point. The benefits of setting the weights to this prior values are discussed in [46].The burn in period was set to 300 cycles and samples were drawn from the following 200 cycles. Duplicate samples of weights were removed so that the ensemble contained only unique samples. The ensemble of neural networks was obtained by collecting a set of possible weights for MLP neural networks trained with 15 to 23 hidden nodes.

The predictive certainty is the measure of the percentage of the classifiers that agree with the ensembles total decision. Hence the predictive certainty (PC) for a data set x is defined as follows:

$$PC = 100 \times \frac{1}{N} \sum_{j=1}^{N} \begin{cases} 1 & if\ h_j(x) = E(x) \\ 0 & \end{cases} \quad (4)$$

where $E(x)$ is the decision of the ensemble which has N classifiers. For the Bagging and the Bayesian method that make use of majority vote aggregation method, the predictive certainty is always greater than 50% when dealing with two classes [47]. The predictive certainty may be less than 50% for the AdaBoost methods that make use of weighted majority voting [47].

5.4 Results

The three ensembles were generated as explained in the previous section. The three ensembles were tested using a test dataset of 2500 records. The overall prediction accuracy for the Bagging, Boosting and Bayesian ensemble is 63%, 63% and 65% respectively. Hence all three ensembles perform equally well.

The predictive certainty for each record in the test set were computed and the percentage of record that have a predictive certainty ranging between 90 to 100, 80 to 90, 70 to 80 , 60 to 70 , 50 to 60 and 50 below were computed and these results are presented in figure 5.3, figure 5.4 and figure 5.5.

It is evident that the Bayesian ensemble has fewer records that have a high predictive certainty. It is evident that the Bagging and Boosting methods have at least 55% of the records that can be predicted with a high predictive certainty. Although the Bagging and Boosting ensembles look favourable, it is necessary to see the correlation between the predictive certainty measure and prediction accuracy. In Table 5.1 we find the prediction accuracy for cases that have different range of predictive certainty.

In addition, it is evident that Bagging and Boosting ensembles have low prediction accuracy for records that have high predictive certainty. This may be due to the fact that these algorithms concentrate on building weak classifiers that are trained with data that were previously

misclassified. The Bayesian ensemble has a high accuracy for records that have high predictive certainty. For the Bayesian ensemble, records that have a predictive certainty greater than 70%, have an overall prediction accuracy of 88%.

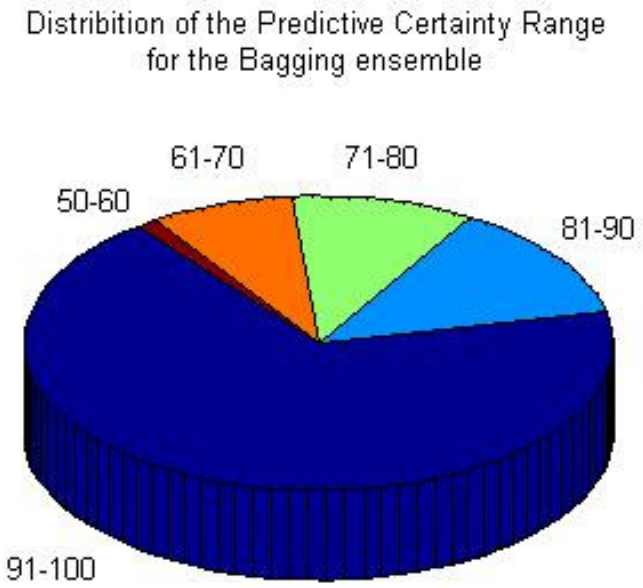


Figure 5.3 Distribution of the Predictive Certainty Values for the Ensemble trained using the Bagging Training Method

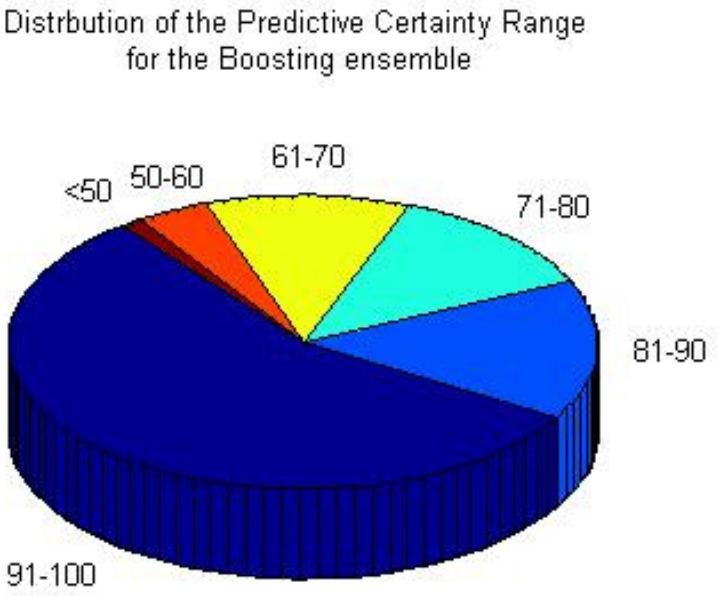


Figure 5.4 Distribution of the Predictive Certainty Values for the Ensemble trained using the Boosting Training Method

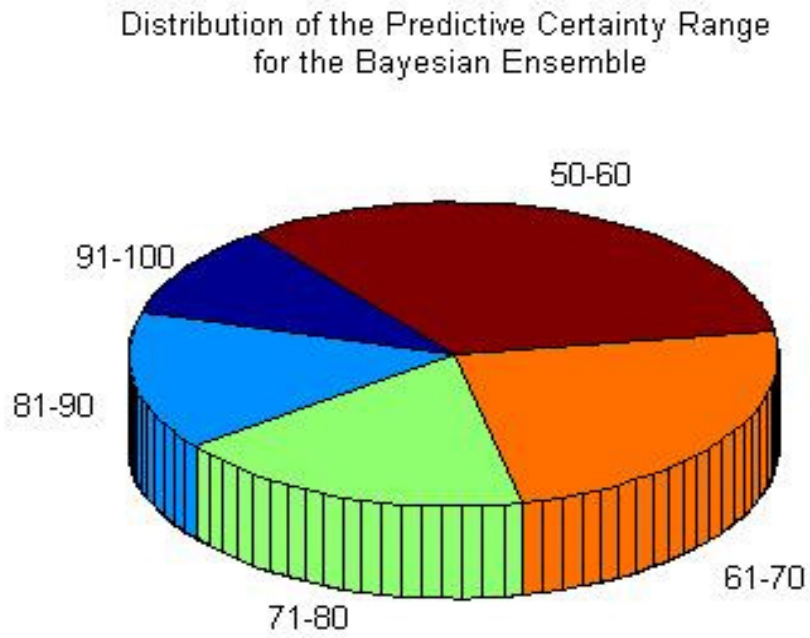


Figure 5.5 Distribution of the Predictive Certainty Values for the Ensemble trained using the Bayesian Training Method

Looking at the results in table 5.1 and figures 5.3 to 5.5 it is necessary to look at both the percentage of data that can be classified with high predictive certainty and one must look at the prediction accuracy for the different ranges of predictive certainty. A good ensemble would be the one that can classify a high percentage of data with high predictive certainty and the predictive accuracy which is also high.

Table 5.1 Accuracy of the Different Ensembles for Predictive Certainty Ranges

Predictive Certainty Range (%)	Bagging Ensemble Accuracy (%)	Boosting Ensemble Accuracy (%)	Bayesian Ensemble Accuracy (%)
91 - 100	68	70	94
81 - 90	57	59	85
71 - 80	49	54	88
61 - 70	54	52	56
50 - 60	55	49	42
<50	N/A	40	N/A

5.5 Conclusion

Three different neural network ensembles were obtained by using the boosting, bagging and Bayesian training methods. The predictive certainty was measured by calculating the percentage of the most dominant decision from each ensemble. All three ensembles are used to predict the HIV/AIDS status of the patient using the given demographic properties. The overall classification accuracy of each of the ensembles is similar to methods used in chapter 3. It was found that the ensemble trained using the Bayesian training method was most suited for obtaining a confident predictive certainty measure. The bagging and boosting methods

predictive certainty measure is not useful because these methods concentrate on training on previously misclassified data sets.

Chapter 6: Demographic influences for HIV classification

6.1 Introduction

Computational intelligence and artificial intelligence have been used successfully for decision making, clinical diagnosis, prognosis and prediction of outcomes. One such problem in this field is for better understanding the HIV/AIDS pandemic. Some of the research on this problem include investigating the causes of the HIV/AIDS virus [19, 55], predicting the HIV status for risk analysis purposes [8, 9] and to better understand the risks of such a virus [56]. In the field of bioinformatics HIV classifications using neural networks are presented in [20-22, 56].

In this chapter the demographic influences on the HIV status of a patient are investigated by using the computational intelligence missing data estimation method. The predictive certainty of the HIV status is found by using an ensemble of autoassociative neural networks. Using the ensemble of classifiers for obtaining a set of possible solutions allows for obtaining predictive certainty measures as suggested in [47]. The Bayesian approach will be used for training in order to obtain an ensemble of neural networks. The estimate from each neural network is aggregated using a voting scheme and the predictive certainty is measured by giving the percentage of the most likely estimate. The change in predictive certainty is measured for adjustments made to the possible states of the different demographic properties of the patients. The changes in predictive certainty will help one to better understand the demographic influences on classifying the HIV status of a patient. The following section discusses the method of determining the demographic influences of the HIV virus. The results of the investigation are presented and discussed.

6.2 Causal Influences for HIV Virus

The demographic influences of the HIV virus are investigated by means of a sensitivity analysis. In sensitivity analysis we compare the model output with the produced output for the modified input parameters. The sensitivity analysis that was done in this chapter, measures the changes in predictive certainty with a change of state of the various demographic influences. A similar

type of sensitivity analysis was done in [57] to determine the causal influences for interstate conflict. In this chapter the sensitivity analysis is done for records that can be predicted with a high predictive certainty. The changes in predictive certainty obtained when changing the variables helps understand the impact each variable has on classifying the HIV status of a patient. In this chapter the average change in predictive certainty is computed for all the records that could be predicted with a high predictive certainty.

6.3 Experimental Methodology

The Multi Layer Perceptron (MLP) architecture is used for the neural network investigated in this chapter. Considering the unary coding scheme used, the neural network has 23 inputs. The HMC training method gives a set of possible weights for certain number of hidden nodes. The weights were initially trained using the scaled conjugate gradient method until the early stopping point is reached. The benefits of setting the weights to this prior value are discussed in [46]. The burn in period was set to 300 cycles and samples were drawn from the next 200 cycles. Duplicate samples were removed so that the ensemble contained only unique samples. The ensemble of autoassociative neural networks was obtained by collecting a set of possible weights for MLP neural networks trained with 15 to 22 hidden nodes in order to increase the structural diversity [58].

6.4 Results

A set of 120 autoassociative neural networks were obtained using the HMC training. The predictive capabilities of these networks were tested using a validation dataset. It was found that the accuracy of these networks ranged between 77% and 96%.

The ensemble of neural networks classifiers yielded an overall accuracy of 68%. It was found that only 40% of the HIV status could be estimated with a predictive certainty greater than 70%. For the records that could be classified with a predictive certainty of 70% it was found that 88% of these records were correctly estimated. The method discussed in section 3 is used for determining how the patient's maximum education achieved, race group, age, age of husband,

gravidity, parity and provincial location influenced the HIV classification. These results are presented in tables 6.1 to 6.7.

From table 6.1 to 6.7 it is evident that there is a bigger change in predictive certainty for patients that are classified as HIV positive. This is because there are fewer cases where patients are classified as HIV positive. A change in predictive certainty of greater than 40% would result in the patients HIV status changing. These large changes in predictive certainty are more evident in the tables 6.1, 6.2 and 6.7.

Therefore, there is reasonable amount of uncertainty for predicting a patients HIV status when using education level, race group and provincial location. Given the high degree of uncertainty when using these variables and their particular state, one should be less confident at classifying such a patients' HIV status.

Table 6.1 Average change in Predictive Certainty (%) for HIV positive and HIV negative

HIV Status	Uneducated	Primary	Secondary	Tertiary
Negative	39	11	3	15
Positive	18	12	0	42

Table 6.2 Average change in Predictive Certainty(%) for HIV positive and HIV negative patients for changes in Race Type(RT)

HIV Status	RT1	RT2	RT3	RT4
Negative	33	2	9	24
Positive	2	41	58	51

Table 6.3 Average change in Predictive Certainty(%) for HIV positive and HIV negative patients for changes in Age Group

HIV Status	GP1	GP2	GP3	GP4	GP5	GP6	GP7	GP8
Negative	3	3	2	3	4	7	10	15
Positive	11	4	2	6	9	10	11	14

Table 6.4 Average change in Predictive Certainty (%) for HIV positive and HIV negative patients for changes in Partners Age Group

HIV Status	GP1	GP2	GP3	GP4	GP5	GP6	GP7	GP8
Negative	2	1	2	2	4	7	10	15
Positive	4	2	2	5	11	17	23	27

Table 6.5 Average change in Predictive Certainty(%) for HIV positive and HIV negative patients for changes Gravidity Ranging from 0 to 5

HIV Status	0	1	2	3	4	5
Negative	11	6	3	2	2	5
Positive	9	1	9	20	25	31

Table 6.6 Average change in Predictive Certainty(%) for HIV positive and HIV negative patients for changes Parity Ranging from 0 to 5

HIV Status	0	1	2	3	4	5
Negative	6	3	3	6	10	12
Positive	1	10	20	25	21	34

Table 6.7 Average change in Predictive Certainty (%) for HIV positive and HIV negative patients for changes in Province(P)

HIV Status	P1	P2	P3	P4	P5	P6	P7	P8	P9
Negative	19	47	51	14	24	2	20	29	23
Positive	2	2	15	15	3	33	24	28	31

This research was aimed at better understanding the demographic influences for HIV classifications. Although all the demographic variables influence the ability to classify the HIV status of a patient, the research shows that certain variables such as race group, educational level and provincial location influence the ability to classify the HIV status of a patient with high

degree of uncertainty. Similar research was done in [19], where inverse neural networks were used for adaptive control of the HIV virus. In [19] gravidity and educational level were assumed influential and used to build inverse neural networks that helped determine how they affect the risk of being HIV positive. The research in this paper verifies this and also shows that some of the other variables are influential to such an extent that there is a high degree of uncertainty in predicting the HIV status of a patient.

6.5 Conclusion

The demographic influences for classifying the HIV status of patients was investigated using an ensemble of neural networks obtained by using the Bayesian training method. The high variability in predictive certainty shows that the ensemble of neural networks used to model the HIV system performs classification with a high degree of uncertainty. It is evident that changing the demographic properties of HIV positive patients has more variability in predictive certainty. It was found that education level, race group and provincial location are the most influential variables for the HIV status classification and the age of mother and her partner have minimal influence on classifying a patients HIV status. The methods applied in this chapter help with knowledge discovery regarding the demographic influences for HIV classification.

Chapter 7: Discussion and Conclusion

7.1 Summary of findings

Methods to determine how confident computational intelligence missing data estimation methods are for classifying the HIV status of a patient were investigated. The South African Antenatal Seroprevalence database was used for the investigations carried out for the research. The predictive certainty, statistical mean and standard deviations was used for quantifying the confidence of estimates. Two approaches were also used to obtain a set of probable HIV status of the patient.

The first approach involved using optimization techniques that give a set of possible outcomes. The mean of this set is used as the optimal outcome and the standard deviation is used to indicate the confidence limits of the optimal estimate. Both the simulated annealing and genetic algorithm optimization techniques were used. It was found that the optimal solutions for these two methods were often different. Further observations showed that most of the estimates were not close to the binary 0 or 1 value. It was found that the optimal solutions obtained were values ranging between 0.3 and 0.5 depending on the data record being used. A plot of the evaluation function that is used by the optimization methods indicates that the set of possible solutions have a random distribution. Furthermore we cannot interpret these values of 0.1 to 0.9 as the probability of a patient being HIV positive because the neural network has never been trained with such probabilities.

Essentially when using autoencoder neural network only one of two estimates (0 for HIV negative and 1 for HIV positive) can be used by the optimization technique. Hence, it is not necessary to use an optimization technique because the search space for estimating the patients HIV status is small. The optimization of the evaluation function can be done by brute force (using all possible solutions from the sample space) . The autoencoder neural network architecture using the brute force optimization yielded the same results as using standard neural network classifiers. We can also conclude that the optimization techniques are not useful in giving a set of possible estimates because they have no domain knowledge.

Hence it was decided to use standard neural network classifiers for HIV classification. These standard classifiers also employ simpler architecture because they have fewer hidden and output nodes. The second approach focused on using an ensemble of neural network classifiers to obtain a set of possible estimates of the HIV status of a patient. The use of an ensemble of neural network allows for each neural network to make a decision on its domain knowledge of the HIV system. Three different ensembles of neural networks were generated using the Bagging, Boosting and Bayesian algorithms. The outputs from each ensemble were aggregated using the majority vote scheme. The predictive certainty was defined to be the percentage of the most popular estimate. It was found that the predictive certainty confidence measure is only useful for the ensemble that was obtained using the Bayesian training method. It was found that only 40% of the records could be predicted with a predictive certainty greater than 70%. If the predictive certainty of the estimate exceeded 70%, the prediction accuracy was 88%. Hence it is concluded that the predictive certainty is a suitable measure to quantify the confidence of the estimate obtained from the ensemble of neural networks which was obtained by using the Bayesian training method.

To better understand the HIV pandemic, an investigation was done to determine which of the demographic variables influence classifying the HIV status of patients. The investigation was done by means of a sensitivity analysis. The sensitivity analysis involves determining the changes in predictive certainty when one of the input parameters was changed. The sensitivity analysis was only done for records that could be predicted with a predictive certainty of greater than 80%. The high variability in predictive certainty shows that the ensemble of neural networks used to model the HIV system performs classification with a high degree of uncertainty. It is evident that changing the demographic properties of HIV positive patients has more variability in predictive certainty. It was found that education level, race group and provincial location are the most influential for the HIV status classification and the age of mother and her partner have minimal influence on classifying a patients HIV status.

7.2 Recommendations for future work

The research focused primarily on determining the confidence or certainty of predicting a patients HIV status. Future work needs to be conducted to determine the confidence or certainty when multiple variables have missing values. This problem is very relevant as there are numerous records in the data base that have multiple fields of missing data. A means of indicating the confidence as a single value or as an individual value for each estimated field needs to be investigated. The research needs to be conducted on other data sets to verify the use of Bayesian neural networks and the predictive certainty measure as a good method to obtain a useful confidence measure.

The research done in chapter 6 can be furthered by using the predictive certainty changes to model the uncertainties of the HIV system. Basically the results in chapter 6 can be used to train neural networks to predict the probability of a patient being HIV positive instead of just classifying the patients HIV status. This model can be verified with the actual data and will help justify the predictive certainty as being a relevant measure for confidence.

Chapter 8: References

[1] F. V. Nelwamondo and T. Marwala, "Key issues on computational intelligence techniques for missing data imputation - A review," *in the Conference Proceedings of the World Multi Conference on Systemics, Cybernetics and Informatics (WMSCI'08)*, pp. 35, 2008

[2] D. B. RUBIN, "Inference and missing data," *Biometrika,* vol. 63, pp. 581, 1976.

[3] D. B. Rubin, "Multiple imputation for nonresponse in surveys," J. Wiley & Sons, New York, 1987.

[4] R. J. A. Little and D. B. Rubin, "Statistical Analysis With Missing Data," *Technometrics,* vol. 45, pp. 364, 2003.

[5] P. D. Allison, "Missing data techniques for structural equation modeling: Structural equation modeling," *Journal of Abnormal Psychology(1965),* vol. 112, pp. 545, 2003.

[6] P. D. Allison, "Multiple Imputation for Missing Data: A Cautionary Tale," *SOCIOLOGICAL METHODS AND RESEARCH,* vol. 28, pp. 301, 2000.

[7] M. Abdella and T. Marwala, "The use of genetic algorithms and neural networks to approximate missing data in database," *Computational Cybernetics, 2005.ICCC 2005.IEEE 3rd International Conference on,* pp. 207, 2005.

[8] B. B. Leke, T. Tetty and T. Marwala, "*Autoencoder networks for HIV* classification," *Current Science,* vol. 91, pp. 1467-1473, 2006.

[9] B. Leke-Betechuoh, T. Marwala, T. Tim and M. Lagazio, "Prediction of HIV status from demographic data using neural networks," in *2006 IEEE International Conference on Systems, Man and Cybernetics,* pp. 2339, 2007.

[10] J. Mistry, T. Marwala and F. V. Nelwamondo, "Investigation of autoencoder neural network accuracy for computational intelligence methods to estimate missing data," in Conference Proceedings of the World Multi Conference on Systemics, Cybernetics and Informatics (WMSCI'08), pp. 41, 2008.

[11] A. P. Dempster, N. M. Laird and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society,* vol. 39, pp. 1-38, 1977.

[12] E. Kreyszig, *Advanced Engineering Mathematics.* ,8th ed.Wiley New York, 1993.

[13] S. Gibilisco, *Statistics Demystified.* McGraw-Hill Professional, 2004.

[14] L. J. Stephens, *Advanced Statistics Demystified.* McGraw-Hill Professional, 2004.

[15] SAHealthdept, "South African Department of Health HIV Syphilis survey data 2001," 2002.

[16] R. Root-Bernstein, "The evolving definition of AIDS, Rethinking AIDS," http ://www.virusmyth.com/*aids*/hiv/rrbdef.htm, 1998.

[17] J. P. Van Geertruyden, J. Menten, R. Colebunders, E. Korenromp and U. D'Alessandro, "The impact of HIV-1 on the malaria parasite biomass in adults in sub-Saharan Africa contributes to the emergence of antimalarial drug resistance," *Malar J.,* vol. 7, pp. 134, 2008.

[18] I. Bonet, M. M. Garcia, Y. Saeys, V. D. Peer and R. Grau, "Predicting human immunodeficiency virus (HIV) drug resistance using recurrent neural networks," in *2nd International Work-Conference on the Interplay between Natural and Artificial Computation, IWINAC 2007*, pp. 234, 2007.

[19] T. Marwala, B. B. Leke and T. Tettey, "Using inverse neural network for HIV adaptive control," *International Journal of Computational Intelligence Research,* vol. 3, pp. 11, 2007.

[20] M. Wang, J. Zheng, Z. Chen and Y. Shi, "Classification methods for HIV-1 medicated neuronal damage," in *2005 IEEE Computational Systems Bioinformatics Conference, Workshops and Poster Abstracts,* pp. 31, 2005.

[21] E. A. Berger, R. W. Doms, E. -. Fenyo, B. T. M. Korber, D. R. Littman, J. P. Moore, Q. J. Sattentau, H. Schuitemaker, J. Sodroski and R. A. Weiss, "New classification for HIV-1," *Nature,* vol. 391, pp. 240, 1998.

[22] A. Srisawat and B. Kijsirikul, "Using associative classification for predicting HIV-1 drug resistance," in *Proceedings - HIS'04: 4th International Conference on Hybrid Intelligent Systems,* pp. 280, 2005.

[23] B. B. Leke, "HIV ANALYSIS USING COMPUTATIONAL INTELLIGENCE,", PHD Thesis, University of Witwatersrand, 2008.

[24] A. L. Knorr and R. Srivastava, "Evaluation of HIV-1 kinetic models using quantitative discrimination analysis," *Bioinformatics,* vol. 21, pp. 1668, 2005.

[25] P. Lurie, K. Phillips, A. Avins, J. Kahn, R. Lowe, P. Franks and D. Ciccarone, "Decision analysis models for HIV testing of health care workers and hospital-based patients," *Proceedings of 8th International AIDS Conference,* 1992.

[26]USCensusBureau, "HIV/AIDS Surveillance Data Base Installation," http://www.census.gov/ipc/www/hsbhome.html, 2004.

[27] WorldBankResources, "Optimizing the Allocation among HIV Prevention Interventions,", http://www.worldbank.org/html/extdr/toc.html, 2002.

[28] FutureGroup, "AIDS Impact Model for Busines: AIM-B,"futuresgroup.com/aim/form1.cfm, 2002.

[29] AIDSCAP, "A tool for Estimating Intervention Effects on the Reduction of HIV Transmission," , http://www.iaen.org/models/avert/avert10.zip, 1998.

[30] C. M. Bishop, *Neural Networks for Pattern Recognition.* Oxford: Oxford University Press, pp. 482, 1995.

[31] I. Nabney, *NETLAB: Algorithms for Pattern Recognition.* Springer, 2002.

[32] M. A. Kramer, "Autoassociative neural networks," *Comput. Chem. Eng.,* vol. 16, pp. 313, 1992.

[33] C. V. Kropas-Hughes, M. E. Oxley, S. K. Rogers and M. Kabrisky, "Autoassociative-Heteroassociative Neural Networks," *Eng Appl Artif Intell,* vol. 13, pp. 603, 2000.

[34] O. Toygar and A. Acan, "An analysis of appearance-based statistical methods and autoassociative neural networks on face recognition," in *Proceedings of the International Conference on Artificial Intelligence, IC-AI 2003,* pp. 292 , 2003.

[35] K. Jang, E. B. Bartlett and R. M. Nelson, "Measuring retrofit energy savings using autoassociative neural networks," *ASHRAE Trans,* vol. 102, pp. 412, 1996.

[36] T. Denoeux and M. Masson, "Principal component analysis of fuzzy data using autoassociative neural networks," *IEEE Trans. Fuzzy Syst.,* vol. 12, pp. 336, 2004.

[37] C. R. Houck, J. Joines and M. Kay, "A Genetic Algorithm for Function Optimization: A Matlab Implementation," *NCSU-IE TR,* vol. 95, 1995.

[38] D. E. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning.* Addison-Wesley Longman Publishing Co., Inc. Boston, MA, USA, 1989.

[39] Z. Michalewicz, *Genetic Algorithms Data Structures= Evolution Programs.* Springer, 1996.

[40] K. Yang, J. Li and C. Wang, "Missing values estimation in microarray data with partial least squares regression," in *ICCS 2006: 6th International Conference on Computational Science,* pp. 662, 2006.

[41] H. Feng, G. Chen, C. Yin, B. Yang and Y. Chen, "A SVM regression based approach to filling in missing values," in *9th International Conference on Knowledge-Based Intelligent Information and Engineering Systems, KES 2005,* pp. 581, 2005.

[42] W. Bridewell, P. Langley, S. Racunas and S. Borrett, "Learning process models with missing data," in *17th European Conference on Machine Learning, ECML 2006,* pp. 557, 2006.

[43] B. Crossingham and T. Marwala, "*Using Genetic Algorithm to Optimise Rough Set Partition Sizes for HIV Data Analysis*," *Studies in Computational Intelligence,* 2007.

[44] C. Andrieu, N. De Freitas, A. Doucet and M. I. Jordan, "An introduction to MCMC for machine learning," *Mach. Learning,* vol. 50, pp. 5, 2003.

[45] G. Casella, J. Ferrándiz, D. Peña, D. R. Insua, J. M. Bernardo, P. A. García-López, A. González, J. Berger, A. P. Dawid and T. J. Diciccio, "Statistical inference and Monte Carlo algorithms," *TEST,* vol. 5, pp. 249, 1996.

[46] A. Vehtari, S. Sarkka and J. Lampinen, "On MCMC sampling in bayesian MLP neural networks," in *International Joint Conference on Neural Networks (IJCNN'2000),* pp. 317, 2000.

[47] R. Polikar, "Ensemble based systems in decision making," *IEEE Circuits and Systems Magazine,* vol. 6, pp. 21, 2006.

[48] J. Li and N. Khaneja, "Ensemble control of linear systems," in *46th IEEE Conference on Decision and Control 2007, CDC,* pp. 3768, 2008.

[49] L. Breiman, "Bagging predictors," *Macine Learning,* vol. 24, pp. 123, 1996.

[50] L. Breiman, "Pasting small votes for classification in large databases and on-line," *Mach. Learning,* vol. 36, pp. 85, 1999.

[51] N. Boonyanunta and P. Zeephongsekul, "Improving the predictive power of AdaBoost: A case study in classifying borrowers," in *16th International Conference on Industrial and Engineering Applications of Artificial Intelligence and Expert Systems, IEA/AIE 2003 Proceedings,* pp. 674, 2003.

[52] R. E. Schapire, "A Brief Introduction to Boosting," *INTERNATIONAL JOINT CONFERENCE ON ARTIFICIAL INTELLIGENCE,* vol. 16, pp. 1401, 1999.

[53] R. M. Neal, "Probabilistic inference using Markov chain Monte Carlo methods," *Computer Science Tech Report Crg-Tr-93-1, University of Toronto,* 1993.

[54] D. J. C. MacKay, "Bayesian Methods for Neural Networks: Theory and Applications," *Course Notes for Neural Networks Summer School. Available Online at: Http://wol.Ra.Phy.Cam.Ac.uk/mackay/cpi4.Ps.Gz,* 1995.

[55] C. W. Lee and J. Park, "Assessment of HIV/AIDS-related health performance using an artificial neural network," *Information and Management,* vol. 38, pp. 231, 2001.

[56] B. L. Betechuoh, T. Tim, T. Marwala and M. Lagazio, "Using genetic algorithms versus line search optimization for HIV predictions," *WSEAS Transactions on Information Science and Applications,* vol. 3, pp. 684, 2006.

[57] T. Tettey and T. Marwala, "Conflict modelling and knowledge extraction using computational intelligence methods," in *INES 2007 - 11th International Conference on Intelligent Engineering Systems,* pp. 161, 2007.

[58] L. M. Masisi, T. Marwala and F. V. Nelwamondo, "The effect of structural diversity of an ensemble of classifiers on classification accuracy," in the Proceedings of the IASTED Simulation and Modeling Conference, Botswana, pp. 130, 2008.