HIV ANALYSIS USING COMPUTATIONAL INTELLIGENCE

Brain Leke Betechuoh

A thesis submitted to the Faculty of Engineering and the Built Environment, University of the Witwatersrand, Johannesburg, in fulfilment of the requirements for the degree of Doctor of Philosophy.

Johannesburg, February 2008

Declaration

I declare that this thesis is my own, unaided work, except where otherwise acknowledged. It is being submitted for the degree of Doctor of Philosophy to 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__

_____

Brain Leke Betechuoh.

Abstract

In this study, a new method to analyze HIV using a combination of autoencoder networks and genetic algorithms is proposed. The proposed method is tested on a set of demographic properties of individuals obtained from the South African antenatal survey. The autoencoder model is then compared with a conventional feedforward neural network model and yields a classification accuracy of 92% compared to 84% obtained for the conventional feedforward model. The autoencoder model is then used to propose a new method of approximating missing entries in the HIV database using ant colony optimization. This method is able to estimate missing input to an accuracy of 80%. The estimated missing input values are then used to analyze HIV. The autoencoder network classifier model yields a classification accuracy of 81% in the presence of missing input values. The feedforward neural network classifier model yields a classification accuracy of 82% in the presence of missing input values. A control mechanism is proposed to assess the effect of demographic properties on the HIV status of individuals, based on inverse neural networks, and autoencoder networks-based-on-genetic algorithms. This control mechanism is aimed at understanding whether HIV susceptibility can be controlled

by modifying some of the demographic properties. The inverse neural network control model has accuracies of 77% and 82%, meanwhile the genetic algorithm model has accuracies of 77% and 92%, for the prediction of educational level of individuals, and gravidity, respectively. HIV modelling using neuro-fuzzy models is then investigated, and rules are extracted, which provide more valuable insight. The classification accuracy obtained by the neuro-fuzzy model is 86%. A rough set approximation is then investigated for rule extraction, and it is found that the rules present simplistic and understandable relationships on how the demographic properties affect HIV risk. The study concludes by investigating a model for automatic relevance determination, to determine which of the demographic properties is important for HIV modelling. A comparison is done between using the full input data set and the data set using the input parameters selected by the technique for the HIV classification. Age of the individual, gravidity, province, region, reported pregnancy and educational level were amongst the input parameters selected as relevant for classification of an individual's HIV risk. This study thus proposes models, which can be used to understand HIV dynamics, and can be used by policy-makers to more effectively understand the demographic influences driving HIV infection.

Acknowledgements

I would like to thank my supervisor, Prof Tshilidzi Marwala, for his assistance and tremendous insight into the project. I also thank the computational intelligence research group students for proof reading my conference and journal papers, as well as providing insights into the research. I thank my parents, Mr. Leke Casimir and Mrs. Leke Agatha, as well as my sisters, Mrs. Tasong Gwendoline and Sydonie, and my brothers, Clarence and Collins for their extreme support. To my brother-in-law, Dr. William Tasong and my nieces Cindy and Jennifer. I thank my friends Patrick, Mphake, Nkandu, Socrates, Andile, Yemi, Danny and Kenneth. Last but not least I thank GOD for giving me the strength and conviction to carry on.

This work was performed in the Computational Intelligence Group at the School of Electrical and Information Engineering at the University of the Witwatersrand. The Group is funded by the National Research Foundation and the Department of Trade and Industry's THRIP programme. Their financial support is much appreciated.

I dedicate this thesis to my Dad, Mr. Leke Betechuoh Casimir, and my Mum, Mrs. Leke Agatha Fonkeng, for their endless support throughout my studies.

Contents

List of Figures

List of Tables

Chapter 1

Introduction to HIV Analysis with Computational Intelligence

1.1 Introduction

Acquired Immunodeficiency Syndrome (AIDS) was first defined in 1982 (Root-Bernstein 1998) to describe the first cases of unusual immune system failure that were identified in the previous year. The Human Immunodeficiency Virus (HIV) was later identified as the cause of AIDS. Since the identification of the virus and the disease, very little has been effective in stopping the spread. AIDS is now an epidemic, which at the end of 2003 had claimed an estimated 2.9 million lives (Poundstone et al. 2004). Epidemiology examines the role of host, agent and environment to explain the incidence and transmission of disease. Risk factor epidemiology examines the individual (demographic and

social) characteristics of individuals and attempts to determine the factors that place an individual at risk of acquiring a disease (Poundstone et al. 2004). In this study, the demographic and social characteristics of individuals and their behaviour are used to determine the risk of HIV infection; this is referred to as "biomedical individualism" (Poundstone et al. 2004; Fee and Krieger 1993). The prevalence of infectious diseases is dependant on the nature of the disease transmission. HIV is primarily transmitted sexually, hence the HIV status in one person is dependant on that of others as well as exposures to other individuals. Social factors therefore affect the risk of exposure, as well as the probability of transmission of the disease and are necessary to understand and model the disease. By identifying the individual risk factors that lead to the disease, it is possible to modify social conditions which give rise to these factors, and thus design effective HIV intervention policies (Poundstone et al. 2004). Traditional control techniques (vector control, environmental control, curative cure) are insufficient for this epidemic whose speed is predominantly determined by the sociological, cultural and economic factors rather than simply biological factors (WorldBank 2002). It is thus imperative to develop models that take into account the sociological, cultural and economic factors. Analytical models based on mathematical models of HIV dynamics have been proposed to try and understand the spread of HIV and the contraction thereof. Knorr and Srivastava (2005) proposed a model to evaluate the intracellular and intercellular scale HIV dynamics of a person using available patient data. Lurie et al. (1992) developed a decision analysis model for HIV testing of health workers and hospital-based patients, which incorporated key

elements of clinical decision-making. They yielded a model, which permitted the evaluation of the economic burden of various policies and the identification of data, which are critical for decision-making. Other models that have been developed include HIV/AIDS Surveillance database (USCensusBureau 2004), the allocation by cost (ABC) model (WorldBankResources 2002), and AVERT, which is a tool for estimating intervention effects on the reduction of HIV transmission, developed by the Family Health International AIDS Control and Prevention (AIDSCAP) Project (AIDSCAP 1998). Another model, AIM-B (AIDS Impact Model for Business) developed by the Futures Group Europe in conjunction with the Global Business Council on HIV and AIDS, was developed to help managers analyse how HIV/AIDS is affecting their business and project, and how it will affect them in the future (FutureGroup 2002). HIV/AIDS intervention are currently being designed and carried out in the developing world, due to the huge infection rates. Sub-Saharan Africa is the most affected region by the HIV/AIDS pandemic according to WHO statistics (UNAIDS 2006), thus it is important to develop intervention policies. These interventions are sometimes evaluated based on controlled trials. The process of designing and evaluating the intervention policies can, however, be quite difficult and time-consuming using conventional mathematical models due to the epidemiologic complexity of HIV/AIDS. It is thus necessary to have a model, which provides assistance to those responsible for implementing prevention studies. In this chapter, the data set to be used for the study is presented. A background on artificial neural networks is also presented, which has been used in HIV/AIDS analysis. Two network architectures are presented which

are the multilayer perceptron (MLP) and the radial basis function (RBF). A background on genetic algorithms is also presented since it is used in this study to create a model for HIV analysis and classification. In conclusion, questions not previously tackled by other models are presented and the contributions of this work are discussed.

1.2 Data to be Used

1.2.1 Data Source

Demographic and medical data to be used for this research, came from the South African antenatal seroprevalence survey of 2001 (HealthDept 2005). This is a national survey, and any pregnant women attending selected public health care clinics participating for the first time in the survey were eligible to participate. Anonymity is guaranteed. The antenatal seroprevalence surveys are used as the main source of HIV prevalence data worldwide, reasons for this are that antenatal clinics are found throughout the world, and pregnant women are ideal candidates for the study as they are sexually active.

1.2.2 Missing Data

Out of the total data set cases, 5964 complete cases were selected, out of 6106 cases (97.68%) and the incomplete entries (142 cases - 2.33%) were discarded.

1.2.3 Variables

The variables obtained in the study are: *race, region, age of the mother, age of the father, education level of the mother, gravidity, parity, province of origin, Rapid Plasma Reatin (RPR), region of origin, regional weighting parameter (WTREV) and* and *HIV status* (HealthDept 2005). The qualitative variables such as race and region are converted into integer values. The age of mother and father are represented in years. The integer value representing education level represents the highest grade successfully completed, with 13 representing tertiary education. Gravidity is the number of pregnancies, complete or incomplete, experienced by a female, and this variable is represented by an integer between 0 and 11. Parity is the number of times the individual has given birth, (for example, multiple births are counted as one) and this is not the same as gravidity. Both these quantities are important, as they show the reproductive activity as well as the reproductive health state of the women. RPR refers to a screening test for syphilis for which HIV may cause a false positive. The HIV status is binary coded; a 1 represents positive status, while a 0 represents negative status. Thus the total number of input variables is 10, shown in Table 1.1.

Table 1.1: Summary of input and output variables

Input Variables	Type	Range
Age Group	Integer	14-50
Age Gap	Integer	1-7
Education	Integer	0-13
Gravidity	Integer	0-11
Parity	Integer	0-40
Province	Integer	1-9
Race	Integer	1-5
Region	Integer	1-36
RPR	Integer	0-2
WTREV	Continuous	0.638-1.2743
Output Variables		
HIV Status	Binary	0 or 1

1.2.4 Outliers

Age is the only variable with outliers. The standard age bracket used in demographic studies relating to female fertility is 14-50 in African countries, and this was used to extract outliers in mother's age.

1.2.5 Data set Used

The dataset was divided into three sets; training, validation and testing sets. The sets were created by dividing the huge dataset into three equivalent small datasets of 1988 entries each. The inputs used were; *age of female, age gap, educational level of female, gravidity, parity, province of origin, race, Rapid Plasma Reatin (RPR)*and *region of origin.* The training set is balanced to consist of an equal number of positive outcomes as negatives, by duplicating the positive entries. An alternative to oversampling the minority class is to assign distinct costs to training examples, or by undersampling the majority class (Hudson and Cohen 2000). Due to the limited size of the dataset, over sampling the positive cases was used rather than undersampling the negative cases to account for the biasing of the data set. The original training set consisted of more negatives than positives with a ratio of 3:1. If the neural network had been trained on this biased dataset, the predicted outcome would always have been negative. This data was randomized and the inputs were scaled between 0 and 1.

1.3 Neural Networks

The recent rapid advances in neural network technology in many pattern recognition systems, as opposed to the conventional statistical theory, have been attributed to the ability of these neural networks to model any kind of system,

be it linear or non-linear. Due to the difficulty and complexity of all the various statistical methods employed and the high level of expertise required for such methods such as; moving averages and regression methods, there has been a significant increase in usage of neural networks. This increase has also been due to the fact that neural networks can be applied to virtually every field in the industry, such as the medical field e.g. AIDS modelling, engineering e.g. control of the product quality. Neural network has gathered enormous momentum in recent years and this field of study is currently being introduced in many universities with the industry demanding more products, which need neural networks. This document constitutes a neural network design for:

- HIV classification from demographic properties
- Estimating missing data in the HIV demographic database
- Understanding the influence of demographic properties on HIV susceptibility and
- Obtaining the relevance of demographic properties on HIV predictability.

Neural networks (NNs) were first introduced in the early 1940s based on the understanding of neurology. An artificial neural network is a network consisting of neurons and paths connecting the neurons (Bishop 1995). They are interconnected assemblies of simple processing nodes whose functionality is loosely based on the animal neuron. NNs can also be defined as generalizations of classical pattern-oriented techniques in statistics and engineering areas

of signal processing, system identification and control. Fig. 1.1 (Bishop 1995) shows a neural network model with the major components of the network. Each input is multiplied by weights along its path and the weighted inputs are then summed and biased. This weighted input is then biased by adding a value unto the weighted input. The output of the summation is sent into a function which the user specifies (linear, logistic). The output of the function block is fed to the output neuron.

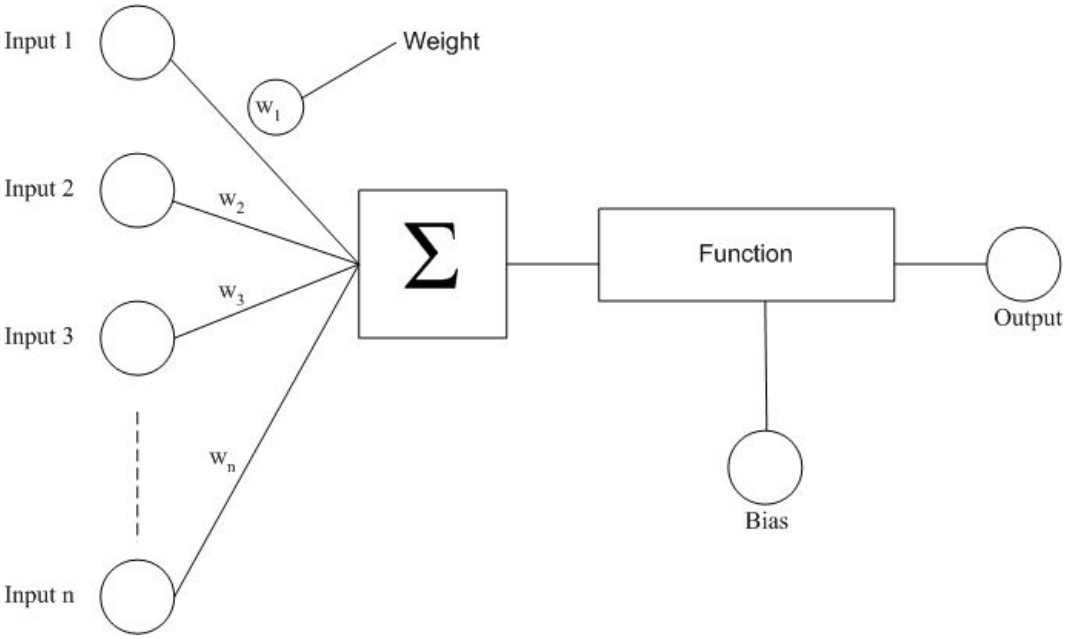

Figure 1.1: Architecture of a neuron

Neural networks (NNs) consist of simple processing units which communicate with each other by sending signals over a large number of weighted connections. The various aspects of the NN models are; neurons (a set of processing units); a state of activation for every unit, equivalent to the output of the unit; connection between the units (each connection is defined by a weight which

determines the signal the unit j has on unit k); a propagation rule (this determines the effective input of a unit from its external inputs); an external input or bias for each unit; and a learning rule. NNs are adaptable systems that can learn relationships through repeated presentation of data, and are capable of generalizing to new, previously unseen data. For Fig. 1.1, the NN output equation is (Bishop 1995):

$$Output_k = \sum_j w_{ij} y_j + b_k \tag{1.1}$$

Where w_j represents the j-th layer's weights, b represents the bias at the node, y_j represents the output at the j-th layer's node and k represents the output node.

Neural network has been motivated by the fact that scientists are challenged to use machines more effectively for tasks currently solved by humans (Smith 2003; Orr 2006; Bishop 1995). Neural networks assist in systems where an algorithmic solution cannot be formulated. NN possess the property of adaptive learning which is the ability to learn how to do tasks based on the data given for training or initial experience (Bishop 1995). They can create their own organization or representation of the information it receives during learning time from the data observed and also possess the ability to represent any function and are known as universal approximators (Bishop 1995). They are insensitive to noise or unreliable data. There is also no restriction on the output type in neural networks and require very short computational times for modelling of systems.

Statistical techniques on handling data have many drawbacks which neural networks do not possess (Smith 2003; Orr 2006). They impose restrictions on the number of input data which NNs do not. The regressions are performed using simple dependency functions (linear and logarithmic), which are quite unrealistic. There is no need for intensive mathematical methods to transform data for NN models meanwhile statistical methods require intensive mathematical transformations. NNs are non-linear hence are better able to account for complexity of human behaviour and also give tolerance to missing or erroneous values.

The integration of neural networks into the modern environment is a major issue in industry. These results from the fact that NN sometimes become unstable when applied to large scale problems and they also neglect the effect of noise hence would tend not to react appropriately to sharp changes. There is also the problem that neural networks are viewed as black boxes whose rules are unknown.

1.3.1 Neural Network Architectures

There exist many kinds of network architectures, such as: (Haykin 1994)

- Multi-layer perceptron (MLP)
- Radial Basis Functions (RBF)

- Recurrent Neural Networks (RNN)
- Hierarchical Mixture of Experts (HME) and
- Self-Organizing Maps (SOM)

Multilayer Perceptron

The simplest network architecture consists of a single layer with directed inputs, weighted connections to the output unit. These are very simple learning algorithms which find the weights for linear and binary activation functions. However, these algorithms can only work for a limited number of functions. The limitations are overcome by adding one or more layers, known as hidden layers which are nonlinear units between the input and the output. The architecture is a feedforward structure whereby each unit receives inputs only from the lower layers units. Gradient methods are used to find the sets of weights that work accurately for the practical cases. Backpropagation is also used to compute derivatives, with respect to each weight in the network, of the error function. The error function generally used in the neural network computation is the squared difference between the actual and desired outputs. The activities for each unit are computed by forward propagation through the network, for the various training cases. Starting with the output units, backward propagation through the network is used to compute the derivatives of the error function with respect to the input received by each unit. The representation of such a network is shown in Fig. 1.2 (Bishop 1995). The learning algorithm

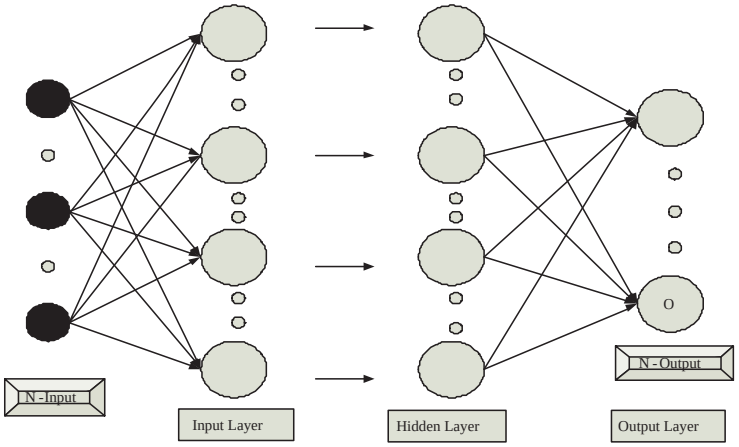

Figure 1.2: Two-Layer Multilayer Perceptron Neural Network

and number of iterations determines how good the error on the training set is minimized meanwhile the number of learning samples determines how good the training samples represent the actual function. In multi-layer perceptron, a number of layers are fully connected. The input to the activation function then becomes a scalar product of the layer weight vector w_i and input i, that is (Bishop 1995):

$$Output = actfun(w_i \times i) \tag{1.2}$$

The different kinds of activation functions with their equations are as shown in Table 1.2. The perceptron learning rule is a method for finding the weights in a network. The perceptron has the property that if there exist a set of weights that solve the problem, then the perceptron will find these weights. This rule follows a regression approach, that is, given a set of inputs and output values, the network finds the best mapping from inputs to outputs. Given an input value which was not in the set, the trained network can predict the most likely output value. This ability to determine the output for an input the network was

Table 1.2: Table of activation functions with the respective functions

Name	Function
Linear	A
Sigmoid	$\frac{1}{1+e^{-a}}$
Tanh	$\frac{e^{a}-e^{-a}}{e^{a}+e^{-a}}$
Exp	e^{a}
Softmax	$\frac{e^{a}}{\sum_{j} e_{j}^{a}+e_{j}^{-a}}$

not trained with is known as generalization. Multi-layer networks are known as approximators. Two-layer networks with a sigmoid transfer function in the hidden layer and linear transfer functions in the output layer can approximate any function provided a sufficient number of hidden units are available (Bishop 1995). These hidden units make use of non-linear activation functions.

Radial Basis Function

These kinds of networks consist of 2 layers, stacked together. The first layer with a Gaussian activation function and the second layer with a linear activation function. These networks are fast in training because the first layer can be initialised with meaningful values and the second layer is found through matrix inversion techniques (Haykin 1994). An iterative optimization technique is then used to refine the solution. The computation nodes of the hidden layers of such a network are different and serve a different purpose from the output layer of the network as opposed to the MLP where the hidden and output layers

share a common neuron model (Hassoun 1995). The hidden layer, as discussed above, for the RBF network is non-linear and the output layer is linear hence the inability to approximate non-linear functions whereas in MLP both layers are non-linear (Haykin 1994). The RBF network has the architecture shown in Fig. 1.3 represented by the following equations (Bishop 1995)

$$y_k(x) = \sum_{j=0}^{n} w_{jk}\phi_j(x) + b_j \tag{1.3}$$

and

$$\phi_j(x) = exp(-\frac{\parallel x - \mu_j \parallel^2}{2\sigma_j^2}) \tag{1.4}$$

Where μ represents the centres and σ represents the widths of the network (training parameters to be optimised).

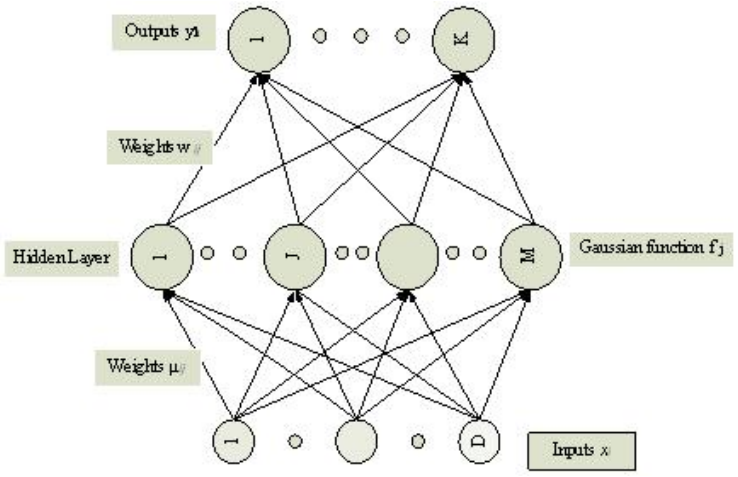

Figure 1.3: Architecture of a Radial Basis Function Neural Network

Recurrent Neural Networks

In these networks, there is the presence of recurrent or loop connections. These recurrent connections can, however, be unfolded to form feed-forward neural

networks (Olurotimi 1994). These networks make efficient use of time varying information but are, however, complex to design. This complexity arises from the fact that in order to use backpropagation algorithms with such architecture, there is a need to make the architecture feed-forward first, hence adding some computational expense (Manolios and Fanelli 1994). The inputs and outputs of this architecture are of arbitrary length sequences of vectors, not vectors. This also makes the handling of the input and outputs difficult to follow (Connor et al. 1994).

Hierarchical Mixture of Experts

These networks are built out of modules, experts and gates, of which can be any of the other neural network types (Bishop 1995). The experts work on the problem in a small domain; meanwhile, the gates mix the opinions of the experts. The building of structure is data driven which poses a problem since as the structure would tend to fit the particular data it was trained for hence leading to over-fitting, which is a phenomenon to be avoided.

Self-Organizing Map Networks

SOM is mainly used in the biomedical field such as in coronary heart risk assessment. It is relatively easy to implement and evaluate and is computationally cheap (Bishop 1995). However, SOM has the problem of overcrowding and underutilization of the neurons in the network due to the fact that the

size and shape of the network is fixed before the training phase begins.

1.3.2 Remarks on Network Architectures

The above sections have discussed briefly the different architectures available for neural network. Each section has given the short-falls of the various networks. MLP are, however, the most appropriate network architecture for the project at hand since RBF networks require more parameters than MLP, RNN are complex to design due to the fact that they need to be unfolded (Bishop 1995), HME networks lead to over-fitting of the data and SOM networks have the problem of overcrowding and underutilization of the neurons in the network (Haykin 1994; Bishop 1995).

1.4 Genetic Algorithms

Genetic algorithms (GAs) are algorithms that are used to find global approximate solutions to complex problems, which are inspired by biological evolutions (Michalewicz 1996). GA is inspired by Darwin's theory of natural evolution (Holland 1975). GAs have been proven to yield good results for various optimization problems such as scheduling routines, adaptive control, transportation problems, travelling salesman problems and optimal control problems. In genetic algorithms, the learning process is considered as a competition among a population of individuals (Davis 1991). The individuals are selected based

on the evaluation of a fitness function, which evaluates the contribution of the individuals to the next generation of solutions. New populations of individuals forming the candidate solutions are then created through the genetic processes of reproduction and mutation.

GA was chosen as the main optimization technique in this research due to its superiority over the other optimization techniques (Michalewicz 1996). GAs differ from conventional optimization methods since it focuses on a population of candidate solutions rather than on a single candidate solution. The population of candidate solutions go through a process of reproduction of individuals which favour the individuals with better fitness values than the other individuals in the previous generation (Michalewicz 1996). GAs offer an alternative method to the existing conventional optimization methods, where these methods are inappropriate. Genetic algorithms are able to provide more feasible and optimum solutions than conventional optimization methods. They transform the optimization problem into an appropriate form unlike other evolutionary programs that leave the problem unchanged.

The genetic algorithm is implemented as follows:

1. Generate a population of candidate solutions randomly
2. Calculate the fitness values for each of the candidates in the population
3. Perform genetic processes (reproduction, crossover, and mutation) on the fittest individuals to generate new candidate solutions.

4. Evaluate the new fitness values for the new population of candidate individuals

5. Iterate to step 3 until the optimum solution is found.

Another important step in the implementation process is the determination of the various parameters required by genetic algorithm such as the population size, and the probabilities of applying genetic operators. These operations include; Encoding, Evaluation, Crossover, Mutation and Decoding. These operations are performed on the population of candidate solutions until a stopping criterion is attained. The standard stopping criteria used to stop the procedure is a given number of iterations, which is also known as the number of generations. This ensures that the procedure eventually stops even if a global optimum or a convergence point is not reached. Another stopping criterion for the procedure is when the best solution does not change over a specified number of iterations. This happens when an optimum solution has been found. The third stopping criterion is when the average fitness of the generation is the same or is close to the fitness of the best solution. More details on genetic algorithm background can be found in Goldberg (1989); Davis (1991).

In this research, the genetic algorithms operative parameters such as; Crossover parameter, mutation parameter, algorithm, and training steps was obtained through experimentation. For HIV classification, the gene representation was binary meanwhile for the other continuous optimization tasks using fitness functions, the continuous gene was used.

1.5 Statement of the Problem

Section 1.1 gave an introduction on HIV/AIDS. A lot of data is collected all over the world from HIV/AIDS surveys. Statistical and Analytical methods have been proposed to better understand the spread of HIV, which as statistics show is rampant. The emergence of Artificial Intelligence methods (Neural networks in particular) in recent years have offered a computational approach to modelling, which is likely to be very beneficial. These methods are capable of using large data sets and deriving relationships from these data. A question that thus arises is: how do we use computational intelligence to model HIV from the existing demographic data? The demographic data collected in the various antenatal clinics worldwide are survey based. These surveys sometimes contain missing entries whereby an individual does not fill some of the questions posed. For instance, in Table 1.1, the educational level of the first record and the age of the fourth record may not be available. A question that arises in this case is how do we know the educational level for the first record? Are there any mechanisms in place for HIV modelling to predict or approximate the missing data based on the relationship that exists between the variables in the database? What impact does the missing data have on the overall modelling? Upon creating a model for HIV modelling using computational intelligence, another important question that arises is: what are the effects of changes in the demographic properties on the HIV susceptibility of individuals? Is it possible to create a model to understand how changing the demographic properties affect the HIV susceptibility? The demographic surveys use generic

questions for the individuals, some of which are not necessarily important for the modelling process. A question that arises is: Is it possible to create a model which depicts the importance and relevance of the different parameters? The last question recognized is: do neuro-fuzzy models, which have not been investigated for HIV modelling offer better results than other conventional neural networks models and how do the rules obtained by the neuro-fuzzy model compare to rough set rules?

The aim of this research is thus to:

1. Create a model based on computational intelligence to model HIV from demographic data.

2. Create a model to estimate missing data in the HIV database and to understand the impact of such missing data

3. Create a computational model to understand how the demographic properties influence the HIV susceptibility of individuals.

4. Create a model based on neuro-fuzzy networks to model HIV from demographic data and compare with a model based on rough sets for rules extraction.

5. Create a model which depicts the relevance and importance of the survey parameters with respect to HIV modelling, and reduces the demographic input space.

The methods presented earlier, which have been implemented for HIV using standard statistical analysis, have not investigated the use of autoencoder networks for classification. To the best of our knowledge, these methods do not investigate the impact of missing information in their analysis. These methods also have a shortcoming in that even though some use demographic properties, the influence of these properties are seldomly investigated. The relevance of the HIV parameters used, as well as significant information, such as rules, contained in the database for HIV analysis have been seldom looked into.

1.6 Importance of the Research

As earlier stated in previous sections, demographic data are collected worldwide to better understand the spread of HIV. Different analytical and statistical models have been developed to model HIV as presented in Section 1.2. In this research, computational intelligence methods are utilized to model HIV and offer another approach into HIV modelling. In surveys, it is virtually impossible to have a complete database with no missing entries. Missing entries affect modelling to a great effect since most models depend on complete datasets. The other main contributions of this research are:

- Introducing a new research direction into missing data approximation and analysis through neural networks, evolutionary computing and swarm intelligence (Ant Colony Optimization).

- Providing a model to understand the effect of demographic properties on HIV susceptibility, which can subsequently be used by decision-makers and policy-makers to understand how to control the spread of HIV through demographic properties more effectively.

- Investigating the use of neuro-fuzzy models, rough sets approximation and fuzzy rule extraction for HIV modelling.

- Providing a model which diminishes the effective demographic input dimensionality required for HIV modelling. This is important since it results in less time being spent capturing information, which is not important for HIV modelling, thereby saving resources. This model also yields how relevant the demographic input parameters are in driving HIV infection.

1.7 Structure of the Thesis

Chapter 1 of this thesis has presented the research question as well as the research contributions. A background on the tools to be used for the proposed methodologies has been proposed. These tools include neural networks and genetic algorithms. A thorough analysis on neural networks as well as genetic algorithms has been presented. The dataset to be used in this research has also been presented in this chapter.

Chapter 2 describes the proposed method for modelling HIV from demographic

properties. A background on modelling HIV using computational intelligence is presented. The proposed method for modelling HIV is then presented, as well as an introduction to autoencoder neural networks, which are used in this model. The results obtained are then presented and a conclusion is drawn.

Chapter 3 presents a proposed method to estimate missing data in the HIV database using ant colony optimization. The chapter begins by presenting missing data models that have been previously proposed. Ant Colony Optimization (ACO) is then introduced and the methodology proposed is presented. The results obtained are then presented together with remarks and conclusions.

Chapter 4 presents a methodology proposed to understand the demographic influences on HIV susceptibility. An adaptive control model is proposed and a background on adaptive control implementation is presented. The results for this section are then presented, together with concluding remarks.

Chapter 5 presents a proposed method for modelling HIV from demographic characteristics using neuro-fuzzy modelling. A background on neuro-fuzzy models and rough set theory is first presented. The neuro-fuzzy method proposed is then presented and compared to the method implemented in Chapter 2. The fuzzy rules extracted are also presented herein and the rules extracted are compared to those extracted from a rough set model approach. Concluding remarks are then drawn.

Chapter 6 presents an automatic relevance determination methodology based on computational intelligence to determine which of the parameters obtained from the surveys are important for the HIV modelling process. The relative relevance of the demographic parameters is presented and the input space is reduced. A background on automatic relevance determination is presented. The results obtained are then presented together with concluding remarks.

Finally, Chapter 7 presents the overall conclusion, which shows how the research questions have been answered. Furthermore, possible future research work is proposed in this chapter.

1.8 Publications

From this thesis, the following journal and conference publications were made:

1. Leke, B., Marwala, T. and Tettey, T.: 2006, Autoencoder Networks for HIV Classification, *Current Science Journal*. **91**(11), 1467-1473.

2. Leke, B., Marwala, T. and Tettey, T.: 2007, Using Inverse Neural Networks for HIV Adaptive Control. *International Journal of Computational Intelligence Research*, **3**(1), 11 - 15.

3. Leke, B., Tim, T., Marwala, T. and Lagazio, M.: 2006, Using genetic algorithms versus line search algorithm optimization for HIV predictions, *WSEAS Transactions on Information Science and Applications*, **4**(3),

684-690.

4. Leke, B., Marwala, T., Tim, T. and Lagazio, M.: 2006, Prediction of HIV status from demographic data using neural networks. *Proceedings of the 2006 IEEE International Conference of Systems, Man and Cybernetics*, Taipei, Taiwan, 2339-2344.

5. Leke, B. and Marwala, T.: 2006, Ant colony optimization for missing data estimation, *Proceedings: Pattern Recognition Association of South Africa*, Parys, 183-188, ISBN 10: 0-620-37384-9.

6. Leke, B., Marwala, T., Tim, T. and Lagazio, M.: 2006, A comparative study between genetic algorithms and line search algorithm optimization for HIV predictions. *Proceedings of the 5th WSEAS International Conference on Artificial Intelligence, Knowledge Engineering and Data Bases*, Madrid, Spain, 231-236.

7. Leke, B., and Marwala, T.: 2007, HIV Modeling from Demographic Data Using Neuro-fuzzy modeling and Fuzzy Rule Extraction. *IEEE International Conference of Fuzzy Systems*, Imperial College, London, United Kingdom. (Submitted)

8. Leke, B., Marwala, T., Manana, J.: 2007, Computational Intelligence for HIV Modelling, IJCNN, Orlando, USA, *Under review.*

9. Leke, B., and Marwala, T.: 2006, Estimation of Missing Entries in HIV Database Using Computational Intelligence. *Submitted to Journal of Computers in Biology and Medicine*, Under Review

10. Leke, B., Marwala, T. and Vilakazi, B.: 2006, Understanding Demographic Influences on HIV Susceptibility. *Submitted to Epidemiologic Reviews*. Under Review.

Chapter 2

Autoencoder Networks for HIV Classification

2.1 Introduction

This section introduces a new method to analyze HIV using a combination of autoencoder networks and genetic algorithms. A model is also proposed based on conventional feedforward neural network. The proposed method is tested on a set of demographic properties of individuals obtained from the South African antenatal survey. The autoencoder network model was found to outperform the conventional feedforward neural network models, as a much better classifier. Neural networks have been successfully used for medical informatics, for decision making, clinical diagnosis, prognosis, and prediction of outcomes such as in Tandon et al. (2006), Alkan et al. (2005), Sawa and Ohno-Machado

(2003), Szpurek et al. (2005), and Tan and Pan (2005) and for classification. Marwala (2001) used a probabilistic committee of neural networks to classify faults in a population of nominally identical cylindrical shells and obtained an accuracy of 95%, in classifying eight classes of fault cases. Ohno-Machado (1996) depicted the limitation on the accuracy of the neural network model due to lack of data balance and increased the accuracy by using sequential neural networks. Lisboa (2002) assessed the evidence of healthcare benefits in using neural networks. Fernandez and Caballero (2006) used artificial neural networks to model the activity of cyclic urea HIV-1 protease inhibitors. They showed that artificial neural networks were capable of representing the non-linearity in the HIV model. Lee and Park (2001) applied neural networks to classify and predict the symptomatic status of HIV/AIDS patients based on publicly available HIV/AIDS data. A study was also performed to predict the functional health status of HIV/AIDS patients defined as "in good health" or "not in good health", using neural networks (Sardari and Sardari 2002). Laumann and Youm (1999) used the racial and ethnic group differences to model the prevalence of the disease and succeeded in relating the demographic properties to the transmission of the disease. Poundstone et al. (2004) related demographic properties to the spread of HIV. Poundstone's work justifies the use of such demographic properties in creating a model to predict the HIV status of individuals, as is done in this study. The above models concluded that ANN performed well in HIV classification problems. The methodology presented here aims at using demographic and social factors, to predict the HIV status of an individual, using autoencoder neural networks.

2.2 Background

2.2.1 Autoassociative Networks

Autoassociative networks are models where the network is trained to recall the inputs (Lu and Hsu 2002). This network thus predicts the inputs as outputs, whenever inputs are presented. These networks have been used in a number of applications such as Atalla and Inman (1998), Frolov et al. (1995), Smauoi and Al-Yakoob (2003), and Hines et al. (1998). An autoassociative neural network encoder (or simply known as autoencoder) consists of an input and output layer with the same number of inputs and outputs, hence the name autoassociative, combined with a narrow hidden layer (Lu and Hsu 2002). The networks will be trained using HIV/AIDS demographic data. The hidden layer attempts to reconstruct the inputs to match the outputs, by minimizing the error between the inputs and the outputs when new data is presented. The narrow hidden layer forces the network to reduce any redundancies, but still allows the network to detect non-redundant data. However, it must be noted that for missing data estimation it is absolutely crucial that the network must be as accurate as possible and that this accuracy is not necessarily realized through few hidden nodes as is the case when these networks are used for data compression. It is therefore crucial that some process of identifying the optimal architecture be used. Genetic algorithm is used in this study to find the optimal autoencoder architecture by finding the global optimum solution

(Holland 1975). Preliminary research showed that genetic algorithms outperformed line search optimization methods (Leke, Marwala, Tim and Lagazio 2006a; Leke, Tim, Marwala and Lagazio 2006). The auto-encoder neural network architecture used in this study is shown in Fig. 2.1.

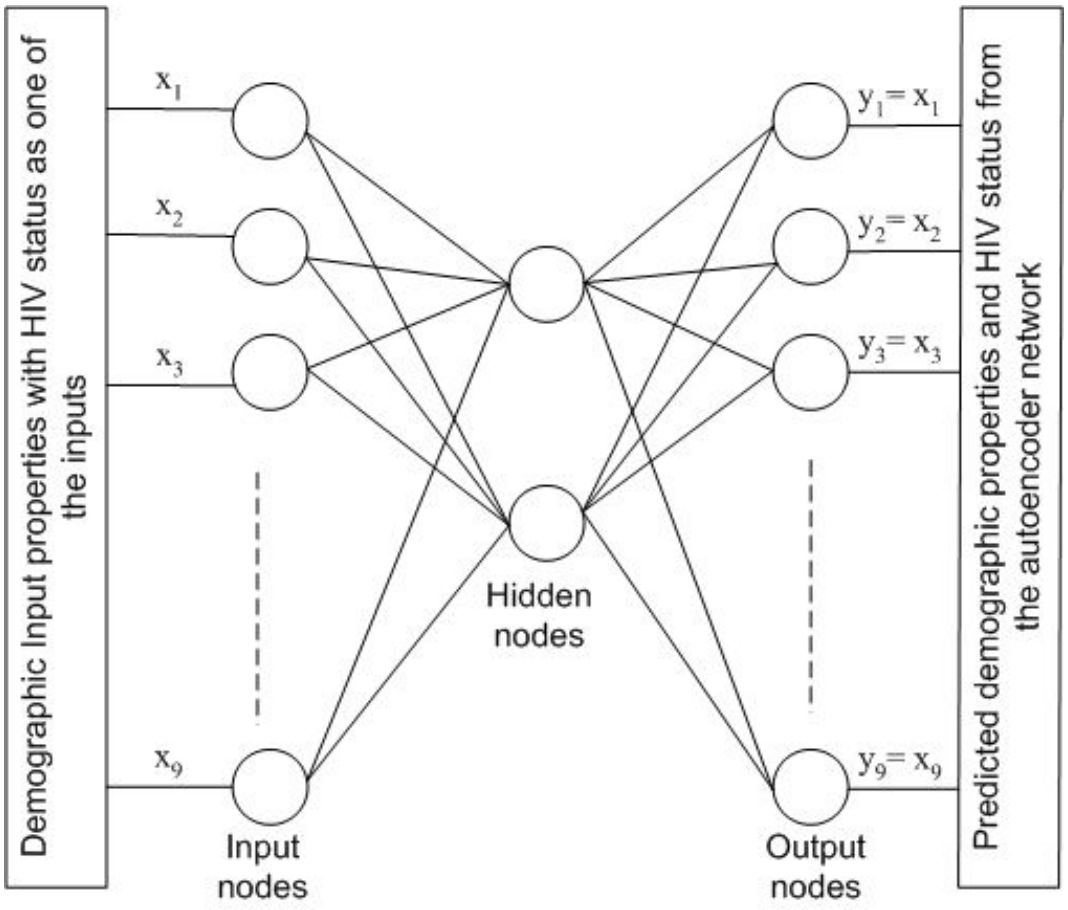

Figure 2.1: Architecture of an Autoassociative Neural Network

2.2.2 Classification as a Statistical Pattern

The goal of our classification is to develop an algorithm, which will assign an individual, represented by a vector x describing the demographic, social and

behavioural characteristics of that individual, to one of the HIV classes, C1 or C2 (where C1, C2 represents the status of an individual, which may be positive or negative). The data on which the model is based upon contains demographic examples of individuals, as well as the classes to which those individuals belong. The output of the classification system is assigned to the variable y. The classification model is therefore required to map the inputs $x_1, ..., x_d$ to the output y. A mathematical function describes this mapping, and since it cannot be explicitly determined, the data is used to determine the parameters. This can be written as follows:

$$\{y\} = f(\{x\}, \{w\}) \tag{2.1}$$

Here w is the mapping weights and x represents the demographic input parameters and y represents the HIV status. In this study, autoencoder neural networks are used to obtain the functional mapping, and supervised learning is used to obtain the parameters. In the case of autoencoder networks, the networks are trained to recall the inputs, hence the functional mapping equation can be represented as:

$$\{x\} = f(\{x\}, \{w\}) \tag{2.2}$$

The purpose of the classification model is to design the decision surface to assign new inputs to one of the classes (Bishop 1995).

2.2.3 The Confusion Matrix

The mean square error (MSE)is insufficient as a classification accuracy measure, as it indicates only an error function that can be minimized by optimization methods, but does not give an indication of the classification accuracy. In medical diagnosis in particular, it is necessary for a more detailed accuracy analysis, including the number of false positives, false negatives, true positives and true negatives. The confusion matrix shows the cross-classification of the predicted class against the true class. By splitting misclassifications into the different cells of the matrix, it is possible to assign a cost of making that particular misclassification (Hand et al. 2001). The confusion matrix is given in Eqn 2.3 (Hand et al. 2001).

$$\begin{pmatrix} TN & FP \\ FN & TP \end{pmatrix} \tag{2.3}$$

Where: TN = True Negatives, FP = False Positives, FN= False Negatives and TP = True Positives.

The rows represent the true classes and the columns represent the predicted classes. The ideal solution has no false positives, nor false negatives, so the diagonal entries are at a maximum. Usually, as is true for this case, the cost of misclassification is difficult to determine. Using the quantities in the confusion matrix, it is possible to derive the Receiver Operating Characteristic or ROC curve. It is also possible to get the accuracy for the measurements from the confusion matrix which will be used to qualify the network and the results obtained.

2.2.4 Receiver Operator Characteristics and Accuracy

The neural network classifiers produce a continuous output indicative of the probability that the element belongs to a class. A threshold is applied to convert this output to predict class membership, and the value of the threshold affects performance. For an instance in a two-class classifier there are four possible outcomes: true positive, where the instance is positive and is classified as positive; true negative, where the instance is negative and is classified as negative; false positive, where the instance is negative but is classified as positive; and false negative, where the instance is positive but is classified as negative. These outcomes are often summarised in a confusion matrix, where the entries along the major diagonal represent correct decisions, and the entries off the diagonal are the errors. Other quantities are derived from the possible outcomes. The True Positive Rate (hit rate or sensitivity) is defined in Eqn 2.4, and the False Positive Rate (false alarm rate), or specificity is defined in Eqn 2.5 (Hand et al. 2001; Lavrac 1999).

$$TruePositiveRatio = \frac{TP}{TP + FN} \tag{2.4}$$

$$FalsePositiveRatio = \frac{FP}{FP + TN} \tag{2.5}$$

The True Positive Ratio (Hand et al. 2001; Lavrac 1999) is plotted against the False Positive Ratio for different threshold values. The accuracy in general is the number of correctly classified out of the total number of cases. An ROC curve demonstrates several things:

1. It shows the tradeoff between sensitivity and specificity (any increase in

sensitivity will be accompanied by a decrease in specificity).

2. The closer the curve follows the left-hand border and then the top-border of the ROC space, the more accurate the test.

3. The closer the curve comes to the 45-degree diagonal of the ROC space, the less accurate the test.

4. The slope of the tangent line at a cutpoint gives the likelihood ratio (LR) for that value of the test.

5. The area under the curve is a measure of the accuracy. The accuracy depends on how well the test separates the group being tested into the two classes. The curve always goes through two points (0,0 and 1,1). (0,0) is where the classifier finds no positives, that is the classifier always gets the negative cases right but gets all positive cases wrong. (1,1) is where the classifier finds no negatives, that is the classifier gets all the positives right but gets all the negatives wrong.

A typical example of an ROC curve is shown in Fig. 2.2.

From Fig. 2.2, there are three ROC curves. Curve A has a larger area under the curve and is considered as the best classifier amongst the three, meanwhile curve C has an area under the curve (AUC) of 0.5 and is the worst classifier amongst the three. An AUC of 0.5 means that the model classifiers half of the samples right, which is the probability of guessing right giving two choices. Hence, the curve C represents a model which can be obtained by guesswork.

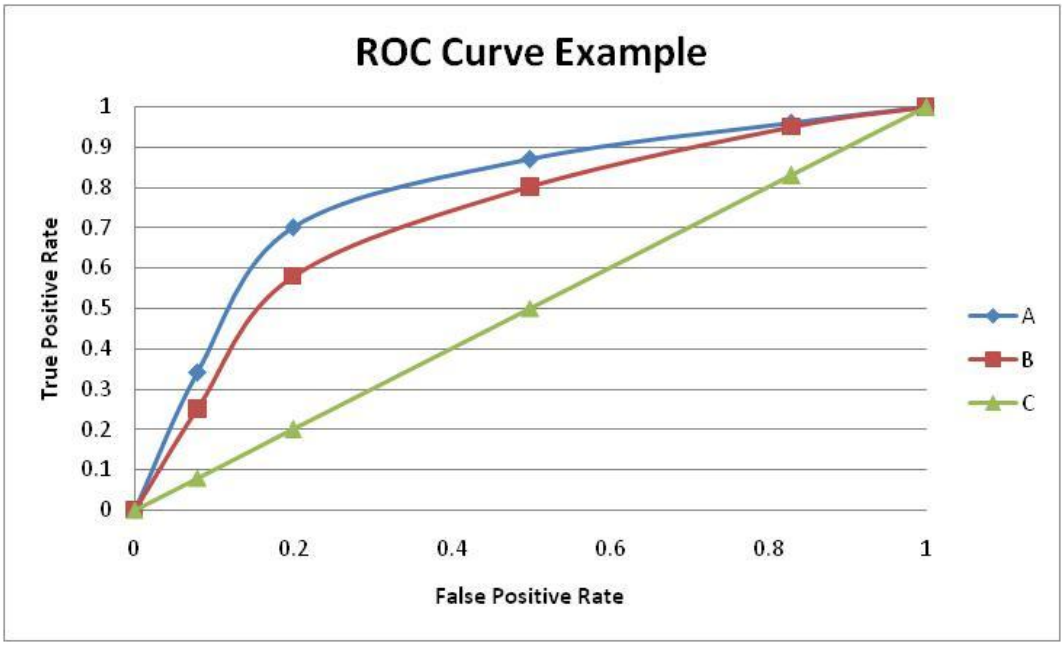

Figure 2.2: ROC Curve Example

The AUC can be computed using the trapezoidal rule in mathematics. A perfect classifier moves from point (0,0) to point (0,1) and then moves from point (0,1) to point (1,1) such that the AUC is 1. The closer the AUC tends towards 1, the better the classifier.

Model Selection

Plotting the True Positive Ratio on the y-axis against the False Positive Ratio on the x-axis results in the ROC curve, which depicts the trade-offs between true positives and the costs (false positives). Perfect classification occurs at the point (0,1) on the ROC space, while (0,0) indicates that the classifier never issues positive classifications, and (1,1) represents a classifier always issuing positive classification (Fawcett 2003). The threshold can therefore be selected

according to the misclassification costs: if a classification should only be made if there is strong evidence, then a classifier in the lower left hand side should be selected. Conversely, if the aim is for the classifier to be sensitive to possible positive cases, the upper right hand corner shows the classifiers that make positive classifications even if evidence is low.

Model Evaluation

Random performance manifests as the diagonal line y = x, and a classifier guessing 50% of the time is positioned at (0.5,0.5) on the plot. The area under the ROC curve (AUC) has often been used to compare classifiers, with a maximum of area of 1 and minimum of 0. The AUC is a useful method since it does not depend on the decision threshold, and is invariant to prior class probabilities. Since all random classifiers appear on the diagonal, all performing classifiers should have an area greater than 0.5. Statistically, the AUC is the probability that the classifier will rank a randomly chosen positive instance higher than a randomly chosen negative instance (Fawcett 2003). The area under the ROC curve is a measure of the difference between the distributions of the positives and negative classes (Chan et al. 2002). In this study, the area under the ROC represents the probability that the test will produce a value for a randomly chosen diseased subject that is greater than the value for a randomly chosen healthy subject. A perfect test has unity area, while a test that produces a random result has an area of 0.5. In some instances, it is possible that classifiers with larger AUCs are suboptimal classifiers, and the classifiers

on the convex hull of the curve are evaluated for accuracy performance. The slope of the curve indicates class distribution in that segment of the ranking, which means that a diagonal segment indicates that locally, random behaviour occurs. A more convex curve with greater area under the curve therefore indicates good separation ability of the classifier. Concavities also indicate worse than random behaviour, and these occur locally (Flach 2004).

2.3 Methodology

The literature review showed that models for HIV prediction and classification have been developed using conventional feedforward neural networks architectures and have worked well. However, it was found from the literature review that autoencoder networks have not been applied to HIV modelling, for prediction and classification. Our work thus focuses on proposing a methodology for HIV classification from demographic properties using autoencoder neural networks and genetic algorithms. Support Vector Machine (SVM) methods and Decision Trees (DT) methods have been applied for missing data estimation, such as in Ssali and Marwala (2008), Jagannathan and Wright (2008) and Twala et al. (2008). Our work also focuses on comparing the proposed autoencoder method to a conventional feedforward neural networks model, by creating a feedforward MLP neural network model and comparing the results with the autoencoder network model results.

2.3.1 HIV Classification Using Autoencoder Networks

The NETLAB toolbox (Nabney 2003) was used to create and train an autoencoder MLP architecture. This toolbox has a 2-layer MLP network, which according to literature review (Bishop 1995) is capable of modelling any complex relationship, such as the HIV model. The network implemented consisted of an input layer, representing different demographic inputs and the HIV status, mapped to an output layer representing the same characteristics as the input layer via the hidden layer. The network was thus trained to recall itself (predict the demographic inputs). This network is shown in Fig. 2.1. One of the input nodes in Fig. 2.1, x_2, represented the HIV status of individuals, which was ultimately represented by one of the output nodes, y_2, as well. The neural network equation can be written as in Eqn 2.1. Since the network is trained to recall the demographic inputs, the output vector (predicted demographic properties) obtained will be approximately equal to the input vector x (actual demographic properties). An error, however, exists between the input vector x and the output vector y, which can be expressed as the difference between the input and output vector. This error is formulated as (Abdella and Marwala 2005):

$$e = x - y \tag{2.6}$$

Substituting for y from Eqn 2.1 into Eqn 2.6 we get

$$e = x - f(x, w) \tag{2.7}$$

In our work, a minimum and non-negative error is required. This can be obtained by squaring the error function in Eqn 2.7 to obtain

$$e = (x - f(x, w))^2 \tag{2.8}$$

To predict the HIV status of individuals, the HIV status input, in the input vector x was assumed as an unknown input, while the demographic input properties were considered as the known inputs. When the input vector x has unknown elements, the input vector set can be categorized into x known represented by x_k and x unknown represented by x_u. Rewriting Eqn 2.8 in terms of x_k and x_u, we obtain

$$e = (\begin{Bmatrix} x_u \\ \\ x_k \end{Bmatrix} - f(\begin{Bmatrix} x_u \\ \\ x_k \end{Bmatrix}, w))^2 \tag{2.9}$$

Here x_u represents the HIV status of the individual, which is unknown, x_k represents the demographic input parameters of the individuals in Table 1.1, w represents the weight vector that maps the autoencoder network input vector x to the same input vector x. An estimated value for the HIV status is then obtained by minimizing Eqn 2.9 using a genetic algorithm (GA). GA was chosen because it finds a good approximation to the global optimum solution (Davis 1991). GA, however, always finds the maximum value. To cater for this, the negative of Eqn 2.9 was used as the fitness function for the GA. The error function to be minimized is thus

$$e = -(\begin{Bmatrix} x_u \\ \\ x_k \end{Bmatrix} - f(\begin{Bmatrix} x_u \\ \\ x_k \end{Bmatrix}, w))^2 \tag{2.10}$$

Where f represents the functional mapping in the MLP network depicted by Eqn. 1.1, with a linear activation function. This estimated value from the

autoencoder network and genetic algorithm was a continuous value representing the HIV status. A threshold was thus required to convert the HIV output node value to a binary value, representative of the HIV class of the individual. Fig. 2.3 shows the implementation of this proposed model in a flowchart.

2.3.2 HIV Classification Using Neural Networks

In this model, the NETLAB toolbox (Nabney 2003) was used to create and train an MLP neural network architecture. The network implemented consisted of an input layer, representing different demographic inputs of an individual, mapped to an output layer representing the HIV status of an individual via the hidden layer. The network thus mapped the demographic inputs of individuals to the HIV status. This network is shown in Fig. 2.4. The neural network equation can be written as in Eqn 2.1. In this model, however, the output vector represents the HIV status of the individual. The network is thus trained to find the relationship between the HIV status of the individual and the individual's demographic input properties. An error, however, exists between the individual's predicted HIV status (output vector) y and the individual's actual HIV status (target vector) during training, which can be expressed as the difference between the target and output vector. For the neural network HIV classification, the mean square error function between the target output vector and the output vector y is insufficient as a classification

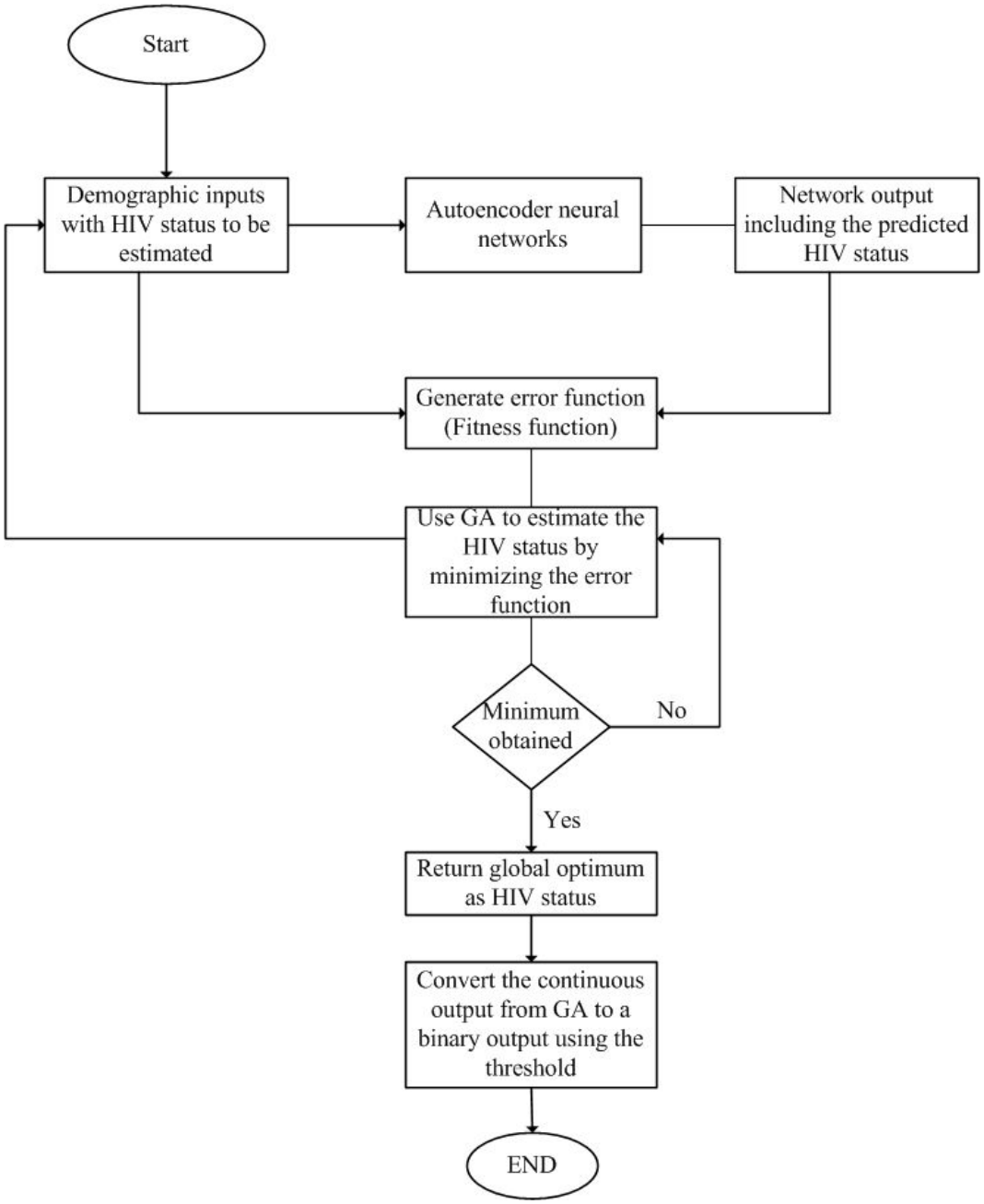

Figure 2.3: Flowchart of the proposed model

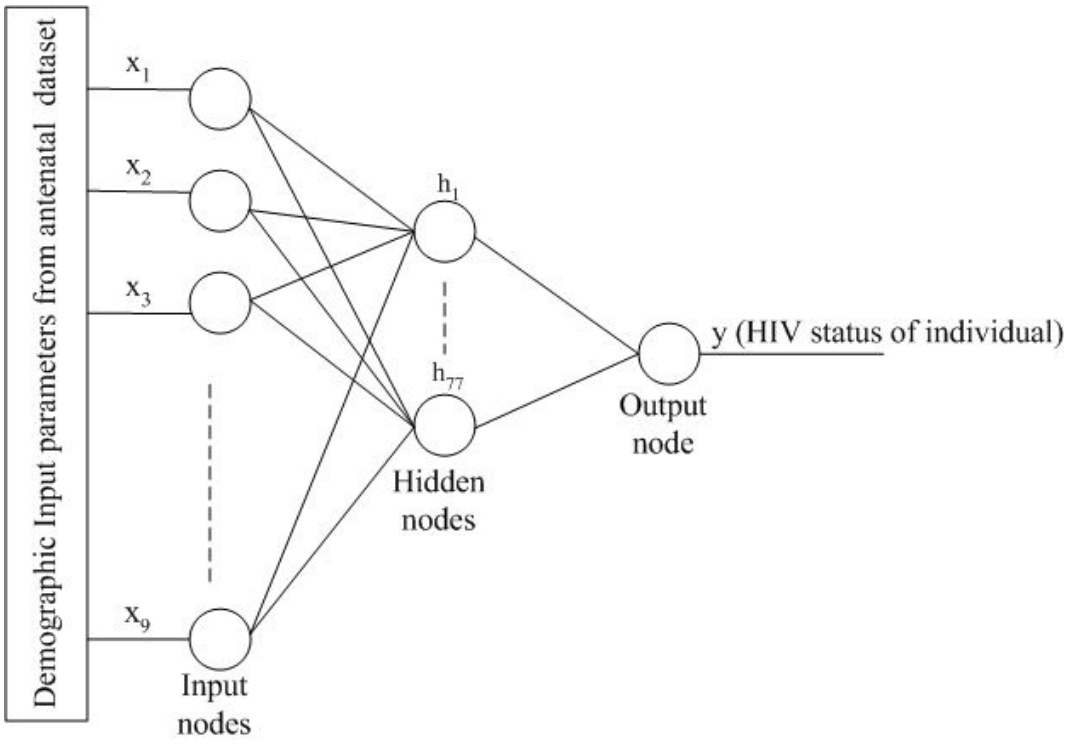

Figure 2.4: MLP network used in this study

accuracy measure, as it only indicates the total number of correct classifications. A confusion matrix was thus constructed and the accuracy was obtained from the confusion matrix. The accuracy can be formulated as (Hand et al. 2001)

$$Accuracy = \frac{TN + TP}{TN + TP + FN + FP} \tag{2.11}$$

Here TN = True Negatives (where network predicts an HIV negative person as negative), FP = False Positives (where network predicts an HIV negative person as positive), FN = False Negatives (where network predicts an HIV positive person as negative) and TP = True Positives (where network predicts an HIV positive person as positive).

The accuracy function was then used as the fitness function in the genetic

algorithm to obtain the optimal neural network parameters(Leke, Marwala, Tim and Lagazio 2006b). GA was used since it finds the maximum value of the fitness function, which was required in this case. GA was also used to obtain the threshold value to convert the continuous network output to a binary value representative of HIV, by minimizing the fitness function. The genetic algorithm parameters and how they were chosen are discussed in Section 1.4.

2.4 Results and Discussions

The Demographic and medical data, used in this study, came from the South African antenatal seroprevalence survey of 2001 (HealthDept 2005). This is a national survey, and any pregnant women attending selected public health care clinics participating for the first time in the survey were eligible. The variables obtained are shown in Table 1.1. A total of 1986 training inputs were provided for the network. The genetic algorithm, used for the autoencoder network model proposed in this study and the neural network model, used arithmetic cross-over, non-uniform mutation and normalized geometric selection. The probability of cross-over was chosen to be 0.75 as suggested in Marwala and Chakraverty (2006). The probability of mutation was chosen to be 0.0333 as recommended by Marwala and Chakraverty (2006). Genetic algorithm had a population of 40 and was run for 150 generations. The first experiment investigated the use of autoencoder networks for HIV classification. An autoencoder network with 9 inputs and 9 outputs was constructed

and several numbers of hidden units were investigated, using Matlab (MATLAB 2004). A GA was used to obtain the optimum number of hidden units and yielded an optimum number of hidden units of 2, hence the structure 9 - 2 - 9. Linear optimization using the mean square error versus hidden units was also investigated. As shown in Fig. 2.5, the linear optimization yielded 6 hidden units as the optimal network that gives the best prediction since the error does not change significantly from 6 units onwards (the difference in error is about 8.5 from 6 hidden units to 20 hidden units). It must be noted,

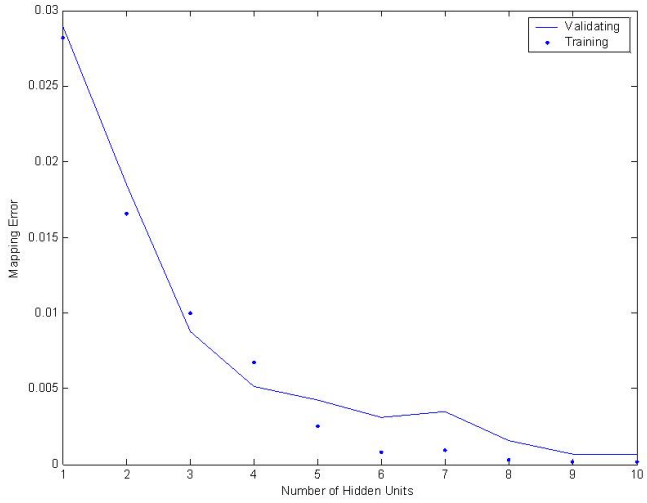

Figure 2.5: Plot of RMS versus Hidden Units

however, that it is generally assumed that the best autoencoder network is the one that has the lowest possible number of hidden units (Kramer 1991). A hidden unit of 2 was thus used as the optimal autoencoder network number of hidden units. The performance analysis for the autoencoder network model is based on classification accuracy and the area under the ROC curve. The proposed autoencoder network model obtained an HIV classification accuracy

of 92%. The confusion matrix obtained for the above network is as shown in Table 2.1. The ROC curve for this classification is shown in Fig. 2.6 and the

Table 2.1: Confusion Matrix of Autoencoder Neural Network Classifier

Confusion Matrix	Predicted Positive	Predicted Negative
Actual Positive	899	94
Actual Negative	65	928

area under the curve was computed as 0.86, thus giving a very good classifier according to Monash (2006).

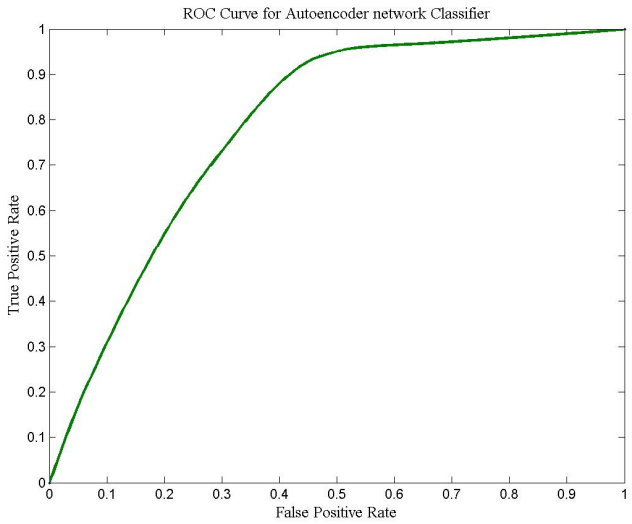

Figure 2.6: ROC curve for autoencoder network classifier

The second experiment investigated the use of conventional feedforward neural network MLP architecture to classify the HIV status of an individual using the demographic input properties. The MLP was constructed with 9 inputs and 1 output. A GA was then used to obtain the optimal structure and yielded an optimal number of hidden units of 77, hence the structure was 9 - 77 - 1.

The performance analysis for this network model is also based on classification accuracy and the area under the ROC curve. This network gave an accuracy of 84%. The confusion matrix obtained for the above network is as shown in Table 2.2. The ROC curve obtained for this classification is shown in Fig.

Table 2.2: Confusion Matrix of Feedforward MLP Neural Network Classifier

Confusion Matrix	Predicted Positive	Predicted Negative
Actual Positive	680	313
Actual Negative	0	993

2.7 and the area under this ROC curve obtained was 0.8, which according to Monash (2006) is a very good classifier. The reason why autoencoder networks

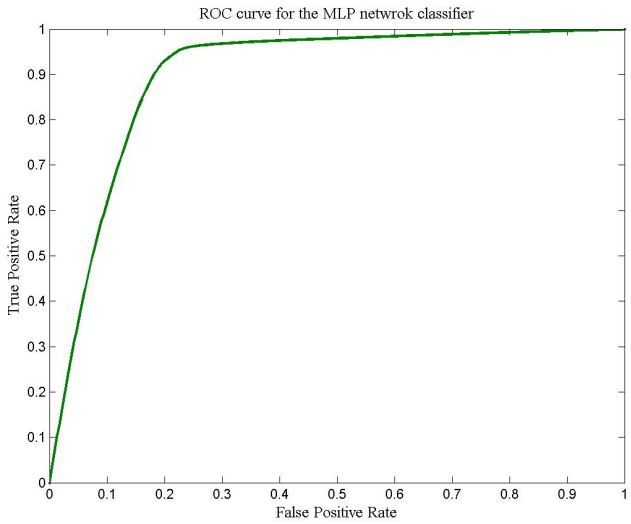

Figure 2.7: ROC curve for the MLP network classifier

performed better than the conventional feedforward neural network can be attributed to the fact that the autoencoder network focuses on characterizing

the positive classes independently of the negative classes, whereas the conventional feedforward neural networks may overlook under-represented classes (Leke et al. 2007a). The difference in performance can also be attributed to the fact that in the autoencoder network, classification is done by choosing the best fitting model using probability distributions, i.e. the class of the network with the smallest reconstruction error meanwhile conventional feedforward neural networks just map an input vector to an output vector using scenario and encodes the classes directly. This plays a role because, for non-linear models such as the HIV model, it is usually difficult to compute the derivatives for the scenarios since they require that we integrate all the possible representations that could have been used for each particular observed input vector.

The results obtained can be summarized as shown in Table 2.3.

Table 2.3: Summary of Results Obtained

Model	Accuracy	Area Under ROC Curve	Classifier Type
Autoencoder Network	92	0.86	Very Good
Feedforward Network	84	0.8	Good

2.5 Conclusion

A method based on autoassociative neural networks and genetic algorithms is proposed to classify the HIV status of an individual from demographic properties. This method is proposed in order to investigate whether using autoencoder networks improves the accuracy of classification, of an individual's HIV status, from demographic properties. The proposed method is tested on an HIV data set obtained from the South African antenatal seroprevalence survey of 2001. The method is then compared to a conventional feedforward neural network model, implemented using the MLP architecture. A classification accuracy of 92% was obtained for the autoencoder network compared to 84% obtained for the conventional feedforward neural network model implementation. The area under the ROC curve for the autoencoder network classifier was computed as 0.86 compared to 0.8 computed for the conventional feedforward neural network classifier. The result of this study thus suggests that autoencoder network models are more accurate and better classifiers for the HIV model than conventional feedforward neural network models, since autoencoder networks focus on characterizing the positive classes independently of the negative classes, whereas the conventional feedforward neural networks may overlook under-represented classes.

Chapter 3

Estimation of Missing Entries in HIV Database Using Computational Intelligence

3.1 Introduction

This chapter introduces a method to estimate missing input values for HIV analysis, using a combination of autoencoder networks and ant colony optimization. The estimated missing input values are then used in an autoencoder network and genetic algorithm classifier model to analyze HIV. The estimated missing input values are also used in a feedforward neural network classifier model to analyze HIV. In the previous chapter, the use of artificial neural

networks in the classification of the HIV status of individuals from their demographic properties was investigated. Good results were achieved by using only complete data cases with no missing data values. In our database, one common missing data value is the educational level of the female, which was missing from 142 individuals (2.3% of our demographic database). The values may not have been recorded due to the female not filling the highest educational level attained. Since this parameter is included in the neural network classification models developed in our previous study, a model is proposed to estimate this missing input parameter, given that the demographic parameters *age group, age gap, region of origin, province, race, gravidity, rapid plasma reatin (RPR)* and *parity*, are known. In the literature review, there is no method proposed thus far that investigates the use of autoencoder networks for missing data values estimation for the HIV model. From the literature review, there is also no method that has been proposed for missing data estimation using ant colony optimization. The aim of this chapter will thus be to propose a new method, which is based on autoassociative models combined with ant colony optimization to estimate missing data in the HIV demographic properties database. These estimated values are then used to classify the HIV status of individuals, and to also quantify the impact of missing data on HIV classification. Two models for classification, based on autoencoder networks and conventional feedforward networks, are analyzed and compared to obtain the most missing data (noise) resistant network model.

Neural network systems usually handle only complete input data cases and

have been applied to data estimation. Junninen et al. (2004) applied neural networks for the imputation of missing values in air quality data sets, and commented on the substitution of mean values of the data to replace the missing data. Junninen et al. (2004) recommended neural networks for imputation of missing data as the more effective model. Gabrys (2002) proposed a missing data estimation method using neuro-fuzzy neural networks, which classified the data successfully in the presence of missing data. Pesonen et al. (1998) inspected different substitution methods for the replacement of missing data values for use in neural network based decision support systems. Pesonen et al. (1998) concluded that neural networks could be used for the estimation of missing data values in the database. Tolle et al. (2000) used artificial neural networks to estimate missing data values for predicting blood concentration levels of pharmaceutical agents in humans, and showed that neural networks outperformed the other methods investigated. Khalil et al. (2001) showed that neural networks are a reasonable alternative for replacing missing data values in a streamflow data set. Lint et al. (2005) presented a state-space neural network model and showed that it yielded good results on both synthetic and real data for missing data models. Other methods that have been applied to missing data estimation include Generative Topographic Mapping as in Vellido (2006) and nearest neighbour methods as in Pesonen et al. (1998). Reports and publications showed models based on neural networks for estimating missing data values in the HIV model and the impact of such missing data have, however, not been investigated. In this study neural networks are used with genetic algorithms (Davis 1991; Michalewicz 1996) and ant colony optimization

(ACO) (Bullnheimer et al. 1998; Dorigo and Gambardella 1997).

Other methods that have also been applied to data imputation for the missing data problem include; Decision trees and Support Vector Machines. C4.5 decision trees have been applied to the problem by Lakshminarayan et al. (1999) for the treatment of missing data in a database. In this case, however, the database was largely made up of continuous attributes, which C4.5 does not naturally handle. Background on C4.5 decision trees can be found in Han and Kamber (2000) and Quinlan (1993). Support vector machines (SVM) have also been applied to the missing data problem, such as in Pelckmans et al. (2005) where the Least-squares SVM was applied to handle missing data. Honghai et al. (2005) also investigated an SVM regression based approach for filling in missing values, but the documented results were for largely continuous databases as well, which differ from the HIV dataset.

3.2 Ant Colony Optimization

Ant colony optimization (ACO) is a branch of swarm intelligence, which makes use of the behavioral simulation of ants. ACO simulates the collective habits of ants - ants searching for food, and bringing their discovered food back to the nest. Ants have poor vision and communication, thus the key to the group effectiveness is pheromone - a chemical substance deposited by ants as they travel (McMullen 2001). ACO was first proposed by Dorigo and Gambardella

as a multi-agent approach for difficult combinatorial optimization problems such as traveling salesman problem (Dorigo and Gambardella 1997), where the ant that finds the shortest path to the food will have the strongest pheromone trail, faster than the ants that choose a longer path (Haupt and Haupt 2004). This path is thus the optimal path since other ants will be attracted to the shorter path. ACO has been applied for feature subset selection using neural networks as in Sivagaminathan and Ramakrishnan (2007). ACO has also been applied in vehicle routing problems (Bullnheimer et al. 1998) and graph coloring (Costa and Hentz 1997). Publications and reports reviewed, however, showed that ACO used in data estimation, to the best of our knowledge, had not been investigated. In this study, ACO is thus used to estimate missing inputs in the HIV data set. More details on ACO can be found in Dorigo and Gambardella (1997); Haupt and Haupt (2004). The next section presents the proposed methodology.

3.3 Methodology

The literature review showed that neural networks have been used for missing data estimation and obtained better results compared to other methods that exist such as replacing the missing values, with mean of known values. Literature review also showed that autoencoder networks combined with genetic algorithms have been used for missing data estimation as in Abdella and Marwala (2005). However, it was found from literature review that autoencoder

networks have not been applied to HIV modeling, for estimation of the missing data. Quantification of the impact of such missing data values in the HIV model have also not been investigated in any literature review. Our work thus focuses on proposing a methodology for missing data estimation in an HIV demographic database, using autoencoder neural networks. The method proposed for missing data estimation is based on ant colony optimization, which has not been applied in missing data estimation. Our work also focuses on classifying the HIV status of individuals from the estimated missing data values obtained from the missing data method, using firstly autoencoder neural networks, and secondly conventional feedforward neural networks. The two classification models are then compared to obtain which of these models is more resistant to the effects of missing data, and thus the more noise resistant model.

3.3.1 Missing Data Estimation Using Autoencoder Networks and Ant Colony Optimization

In this section, an overview of the proposed method is presented in Fig. 3.1. Firstly, a set of ants is initialized. These ants represent possible solutions for the missing data value. The population size, which is specified by the user, determines the number of ants (Nants) to be initialized. The population size must be sufficient to explore all the potential solutions. A population size of 100 is chosen for this study. This was found as the optimal number of ants

during the optimization process. Each ant is then used to represent the input parameter to be estimated and is propagated through an autoencoder neural network. A cost function is then generated, which is the sum-of-squared-errors between the predicted output from the autoencoder network and actual output required, which is the inputs of the autoencoder network. This can be formulated as (Dorigo and Gambardella 1997; Leke and Marwala 2006):

$$cost^k = \frac{\sum_{n=1}^{8}(f\left\{\begin{array}{c} x_{ant}^k \\ , w \\ x_d \end{array}\right\}_n - X_n)^2}{8} \tag{3.1}$$

Here x_{ant}^k represents the k-th ant, and x_d represents the known demographic input properties, w represents the mapping weights of the autoencoder network, X_n represents the n-th actual demographic input, f(o) represents the autoencoder function and $cost_k$ is the cost function of the k -th ant. The cost function is obtained by getting the average sum-of-square error for the known parameters and the values predicted by the network. The closer the predicted values are to the real values, the more accurate the missing data information. Eqn. 3.1 is divided by 8, since there are 8 known parameters and 1 unknown parameter. The cost function is then used to obtain the pheromone for each ant as follows (Dorigo and Gambardella 1997; Leke and Marwala 2006):

$$Phmone^k = \frac{Q}{cost^k} \tag{3.2}$$

Here Q is a constant (the exploitation probability factor) to be optimized (in this study, 0.028(Haupt and Haupt 2004)), and $Phmone^k$ is the pheromone concentration of the k -th ant. The ants are then run through iterations where the pheromone is updated using the update rule (Dorigo and Gambardella

1997; Haupt and Haupt 2004)

$$Phmone_t^k = (1-\xi)Phmone_{t-1}^k + \varepsilon * Phmone_{best} + Phtemp^k \tag{3.3}$$

Here $Phmone_t^k$ is the k-th ant's pheromone from iteration 1 to t,$Phmone_{t-1}^k$ is the pheromone from iteration 1 to (t-1), $Phmone_{best}$ is the pheromone of the ant with the least cost in the current iteration t, $Phtemp^k$ is the k-th ant's pheromone for each iteration t, $0 < \xi < 1$ is the decay constant of the pheromone trail (in this study, 0.8 (Sivagaminathan and Ramakrishnan 2007)) and ε is the best path weighting constant (in this study, 5 (Haupt and Haupt 2004)). The optimal ant, which is the estimated missing data, is then chosen as the ant with the greatest pheromone trail, after all the iterations. The number of iterations, used in this study, is 200, since iterations above this value yielded insignificant changes to the estimation accuracy.

3.4 Results and Discussion

The demographic and medical data, used in this study, came from the South African antenatal seroprevalence survey of 2001 (HealthDept 2005). This is a national survey, and any pregnant women attending selected public health care clinics, participating for the first time in the survey, were eligible. The variables obtained are shown in Table 1.1. A total of 1986 training inputs were provided for the network. Amount of 142 demographic input entries were incomplete. The genetic algorithm, used for the autoencoder network classification model, and the feedforward neural network classification model,

used the same parameters as in Chapter 2, Section 2.4. The ant colony had a population of 100 and was run for 200 iterations.

The first experiment investigated the use of autoencoder networks and ant colony optimization to estimate missing input in the demographic data set model. Input values were discarded from the input data set for experimental purposes, and were then used to obtain the estimation accuracy of the model. An autoencoder neural network structure was created with 9 inputs, 2 hidden layers and 9 output layers, using Matlab (MATLAB 2004). Linear optimization using the mean square error versus hidden nodes was investigated. As shown in Fig. 2.5, the linear optimization yielded 6 hidden nodes as the optimal network that gives the best prediction since the error does not change significantly from 6 units onwards (the difference in error is about 8.5 from 6 hidden nodes to 20 hidden nodes). It must be noted, however, that it is generally assumed that the best autoencoder network is the one that has the lowest possible number of hidden nodes (Kramer 1991). A hidden unit with 2 hidden nodes was thus used as the optimal autoencoder network number of hidden nodes. The proposed method estimated missing input values to an accuracy of 80%. The correlation between the two data sets was 0.84, which also further confirms the consistency of this model. Ant colony optimization is used for missing data estimation as the optimization method, rather than GA in this model, in order to ensure that the autoencoder missing data model and the autoencoder classification model which uses GA are uncorrelated and decoupled.

The second experiment investigated the use of autoencoder networks for HIV

classification in the presence of missing data. An autoencoder network with 9 inputs and 9 outputs with 2 hidden layers (optimal) was constructed (Section 2.3.1), using Matlab (MATLAB 2004). The performance analysis for the autoencoder network model is based on classification accuracy and the area under the ROC curve. The estimated missing demographic data obtained from the first experiment, was combined with the known demographic data from the antenatal data set, to classify the HIV status of individuals. The ROC curve for the classifier is shown in Fig. 2.6. The area under the curve was computed as 0.86, thus giving a very good classifier according to Monash (2006). The accuracy of the model with missing data presence is 81%. Previous research carried out by the authors showed that the classification accuracy of the autoencoder network for a case with no missing data was 92%. The effect of the presence of missing data on the autoencoder network is quantified by a difference of 11, which is quite substantial.

The third experiment investigated the use of a conventional feedforward neural network MLP architecture to classify the HIV status of an individual, in the presence of missing data. The MLP was constructed with 9 inputs representing demographic characteristics, and 1 output representing HIV class (Section 2.3.2). A GA was then used to obtain the optimal structure and yielded an optimal number of hidden units of 77, hence the structure was 9 - 77 - 1. The performance analysis for this network model is also based on classification accuracy and the area under the ROC curve. The estimated missing demographic data obtained from the first experiment, was combined with the known

demographic data from the antenatal data set, to classify the HIV status of individuals. The ROC curve obtained for the classifier is shown in Fig. 2.7. The area under this ROC curve was computed as 0.8, which again according to Monash (2006) is a very good classifier. The accuracy obtained for the classification in the presence of missing data is 82%. Previous research carried out by the authors showed that the classification accuracy of the conventional feedforward network for a case with no missing data present was 84%. The effect of the presence of missing data on the feedforward neural network caused a difference in accuracy of 2%, which is not really significant.

The results thus show that the effect of missing data is more significant on the autoencoder network classification model rather than on the feedforward neural network classification model. It is hypothesized that this may be due to the lower effective dimension of the autoencoder network classifier, which results in high correlation between the input parameters for output classification (this is mathematically proven in the subsequent pages of this chapter). For estimation of parameters where the output is known, autoencoder networks proved to be very effective since the network structure ensures the parameters are reconstructed to suit the output. For classification of the HIV status from missing data, however, if a slightly wrong estimate is yielded for the missing input value, this also causes the predicted parameters to be affected due to the high correlation of the input parameters in the narrow hidden layer. In comparison, the feedforward neural network model assigns significance of input

parameters during learning through the weights and biases of the hidden layers. The input parameters yield a decoupled effect on the output, even though they are combined ultimately in the network. If the missing input values were slightly wrongly estimated, the pattern generated by the feedforward neural network will not be significantly affected, since the other decoupled known input values influence the pattern independently. The error analysis can be demonstrated using the following analysis. Consider a network represented by Fig. 3.2. The network hidden layer equation can be represented by (Bishop 1995):

$$y_1' = w_{11}x_1 + w_{12}x_2 + w_{13}x_3 + b_{11} \tag{3.4}$$

$$y_2' = w_{21}x_1 + w_{22}x_2 + w_{23}x_3 + b_{12} \tag{3.5}$$

$$y_3' = w_{31}x_1 + w_{32}x_2 + w_{33}x_3 + b_{13} \tag{3.6}$$

The hidden layer has an activation function which is a hyperbolic tangent function represented by (Bishop 1995):

$$f(x) = \tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \tag{3.7}$$

Using McLaurin's Expansion for a function f(x), which is:

$$f(x) = f(0) + x \star f'(0) \tag{3.8}$$

Knowing that:

$$f'(x) = \frac{4}{e^x + e^{-x2}} \tag{3.9}$$

We get f(0)=0 and f'(0)=1. Thus,

$$f(x) \approx \tanh(x) \approx x \tag{3.10}$$

by McLaurin's approximation where we are effectively assuming a linear activation function. Thus: $y"_1, y"_2, y"_3$ can be represented by: y'_1, y'_2, y'_3. Replacing x_1, x_2 and x_3 by $x_1 + \Delta x_1$, $x_2 + \Delta x_2$ and $x_3 + \Delta x_3$ respectively, Eqn 3.4, Eqn 3.5 and Eqn 3.6 become

$$y'_1(x + \Delta x) = w_{11}(x_1 + \Delta x_1) + w_{12}(x_2 + \Delta x_2) + w_{13}(x_3 + \Delta x_3) + b_{11} \quad (3.11)$$

$$y'_2(x + \Delta x) = w_{21}(x_1 + \Delta x_1) + w_{22}(x_2 + \Delta x_2) + w_{23}(x_3 + \Delta x_3) + b_{12} \quad (3.12)$$

$$y'_3(x + \Delta x) = w_{31}(x_1 + \Delta x_1) + w_{32}(x_2 + \Delta x_2) + w_{33}(x_3 + \Delta x_3) + b_{13} \quad (3.13)$$

With a linear output function, we get:

$$y = w'_1 y"_1 + w'_2 y"_2 + w'_3 y"_3 + b_2 \quad (3.14)$$

Substituting and simplifying yields:

$$E = (y(x + \Delta x) - y(x)) = \sum_{n=1}^{3} w'_n \sum_{k=1}^{3} w_{nk} \Delta x_k \quad (3.15)$$

This results in a linear combination thus demonstrating that dependence of feedforward neural network on the uncertainty is linear.

If we now consider the network represented by Fig. 3.3, the hidden layer equations obtained are Eqn 3.4 and Eqn 3.5. Using the hyperbolic tangent function Eqn 3.7 as the hidden layer activation function and approximating with the McLaurin's series as in Eqn 3.10. Replacing x_1, x_2 and x_3 by $x_1 + \Delta x_1, x_2 + \Delta x_2$ and $x_3 + \Delta x_3$ respectively in Eqn 3.4 and Eqn 3.5 we get Eqn 3.11 and Eqn 3.12. The outputs with a linear activation function for the network is (Bishop 1995):

$$y_1 = w'_{11} y'_1 + w'_{12} y'_2 + b_{21} \quad (3.16)$$

$$y_2 = w'_{21}y'_1 + w'_{22}y'_2 + b_{22} \tag{3.17}$$

$$y_3 = w'_{31}y'_1 + w'_{32}y'_2 + b_{23} \tag{3.18}$$

Replacing y'_1 and y'_2 by Eqn 3.11 and Eqn 3.12 respectively,

$$\begin{aligned} y_1(x+\Delta x) = {} & w'_{11}(w_{11}(x_1+\Delta x_1) + w_{12}(x_2+\Delta x_2) + w_{13}(x_3+\Delta x_3) + b_{11}) \\ & + w'_{12}(w_{21}(x_1+\Delta x_1) + w_{22}(x_2+\Delta x_2) + w_{23}(x_3+\Delta x_3) + b_{12}) \end{aligned} \tag{3.19}$$

$$\begin{aligned} y_2(x+\Delta x) = {} & w'_{21}(w_{11}(x_1+\Delta x_1) + w_{12}(x_2+\Delta x_2) + w_{13}(x_3+\Delta x_3) + b_{11}) \\ & + w'_{22}(w_{21}(x_1+\Delta x_1) + w_{22}(x_2+\Delta x_2) + w_{23}(x_3+\Delta x_3) + b_{12}) \end{aligned} \tag{3.20}$$

$$\begin{aligned} y_3(x+\Delta x) = {} & w'_{31}(w_{11}(x_1+\Delta x_1) + w_{12}(x_2+\Delta x_2) + w_{13}(x_3+\Delta x_3) + b_{11}) \\ & + w'_{32}(w_{21}(x_1+\Delta x_1) + w_{22}(x_2+\Delta x_2) + w_{23}(x_3+\Delta x_3) + b_{12}) \end{aligned} \tag{3.21}$$

Thus

$$\begin{aligned} y_1(x+\Delta x) = {} & y_1(x) + w'_{11}(w_{11}\Delta x_1 + w_{12}\Delta x_2 + w_{13}\Delta x_3) \\ & + w'_{12}(w_{21}\Delta x_1 + w_{22}\Delta x_2 + w_{23}\Delta x_3) \end{aligned} \tag{3.22}$$

$$\begin{aligned} y_2(x+\Delta x) = {} & y_2(x) + w'_{21}(w_{11}\Delta x_1 + w_{12}\Delta x_2 + w_{13}\Delta x_3) \\ & + w'_{22}(w_{21}\Delta x_1 + w_{22}\Delta x_2 + w_{23}\Delta x_3) \end{aligned} \tag{3.23}$$

$$\begin{aligned} y_3(x+\Delta x) = {} & y_3(x) + w'_{31}(w_{11}\Delta x_1 + w_{12}\Delta x_2 + w_{13}\Delta x_3) \\ & + w'_{32}(w_{21}\Delta x_1 + w_{22}\Delta x_2 + w_{23}\Delta x_3) \end{aligned} \tag{3.24}$$

Thus

$$\begin{aligned} y_n(x+\Delta x) = {} & y_n(x) + w'_{n1}(w_{11}\Delta x_1 + w_{12}\Delta x_2 + w_{13}\Delta x_3) \\ & + w'_{n2}(w_{21}\Delta x_1 + w_{22}\Delta x_2 + w_{23}\Delta x_3) \end{aligned} \tag{3.25}$$

For the above network, the error is quantified as (Leke and Marwala 2006):

$$E = \sum_{i=1}^{3}(y_{pred}^{(i)} - y_{act}^{(i)})^2 = \sum_{i=1}^{3}\xi_i^2 \tag{3.26}$$

Where the y_{act} is equivalent to the inputs. Here y_{pred} is equal to $y_n(x + \Delta x)$ and y_{act} is equivalent to $y_n(x)$;

$$\xi_n = y_n(x + \Delta x) - y_n(x) = w'_{n1}(w_{11}\Delta x_1 + w_{12}\Delta x_2 + w_{13}\Delta x_3) + w'_{n2}(w_{21}\Delta x_1 + w_{22}\Delta x_2 + w_{23}\Delta x_3) \tag{3.27}$$

$$\xi_n = A\Delta x_1 + B\Delta x_2 + C\Delta x_3 \tag{3.28}$$

Where A $= w'_{n1}w_{11} + w'_{n2}w_{21}$, B $= w'_{n1}w_{12} + w'_{n2}w_{22}$ and C $= w'_{n1}w_{13} + w'_{n2}w_{23}$.

$$\xi_n^2 = A^2\Delta x_1^2 + B^2\Delta x_2^2 + C^2\Delta x_3^2 + 2AB\Delta x_1\Delta x_2 + 2AC\Delta x_1\Delta x_3 + 2BC\Delta x_2\Delta x_3 \tag{3.29}$$

Since as the weights w, are small the product of four weights w4 will converge towards a value, K. Thus AB $\approx$ AC $\approx$ BC $\approx A^2 \approx B^2 \approx C^2 \approx$ K. Also the product of the uncertainty is a constant, P.

Thus

$$P = \Delta x_1\Delta x_2\Delta x_3 \tag{3.30}$$

Substituting Eqn (3.30) in Eqn (3.29), we get

$$\xi_n^2 = K\Delta x_1^2 + \Delta x_2^2 + \Delta x_3^2 + \frac{2P}{\Delta x_3} + \frac{2P}{\Delta x_2} + \frac{2P}{\Delta x_1} = K\sum_{i=1}^{3}\Delta x_i^2 + \frac{2P}{\Delta x_i} \tag{3.31}$$

Substituting Eqn (3.31) in Eqn (3.26) yields (Leke and Marwala 2006):

$$E = \sum_{n=1}^{3}K_n\sum_{i=1}^{3}\Delta x_i^2 + \frac{2P_n}{\Delta x_i} \tag{3.32}$$

From Eqn 3.15, it is found that the feedforward network has a linear output dependence on the uncertainty in input data meanwhile from Eqn 3.32, it is

found that the presence of uncertainty in the input data yields a non-linear quadratic relationship. This error analysis of the two networks, thus shows that the results obtained in this study makes sense and is logical, due to the network architectures for the autoencoder and feedforward neural network. Feedforward neural networks are thus more noise resistant than the autoencoder networks.

Ant colony optimization (ACO) was used in this chapter, for estimation of the missing entries in the database. Genetic algorithm (GA) could also be used, however, it was not preferred so as to ensure a decoupling of the estimation model, which uses GA and the missing data model, which uses ACO. Also, the genetic algorithm tends to converge slowly compared to other optimization methods according to the literature (Marwala and Chakraverty 2006). The ACO converged in approximately 320s compared to the GA's approximately 1200s, thus about 4 times the convergence time. Also, the complexity and multi-dimensionality of GA is much higher than that of the ACO, which makes use of the pheromone on the trail. ACO uses a simple update rule as shown in Eqn 3.3 meanwhile GA on the other hand makes use of genetic processes such as crossover, mutation, and reproduction. The accuracy obtained by the GA model for estimation of the missing parameters was 82%, which was slightly better than that of the ACO of 80%, however, the time cost was significantly better for the ACO.

The results obtained can be summarized in Tables 3.1 and 3.2.

Table 3.1: Summary of Results

Model	Accuracy	Correlation	Time	Conclusion
AENN + ACO	80	0.84	320s	Recommended: Computation Good
AAENN + GA	82	0.82	1200s	Computationally Expensive

Table 3.2: Summary of Prediction Results

Model	Accuracy	AUC	Noise Effect	Conclusion
AENN + Missing Data	81	0.86	11	More sensitive to noise
FFNN + Missing Data	82	0.8	2	Recommended:Less sensitive

Note: AENN refers to Autoencoder network and FFNN refers to feedforward neural network.

3.5 Conclusion

A method based on autoassociative neural networks and ant colony optimization is proposed to estimate missing input values in the South African antenatal demographic data set for HIV classification. This method is proposed in order to investigate whether missing input values in the demographic survey have an impact on the classification of the HIV status of individuals. The model proposed estimated input values in the demographic data set with an accuracy of 80%. An autoencoder network classification model and a feedforward neural network classification model are then investigated, to quantify the impact of missing data on these models. The missing input estimates obtained from the

autoencoder missing data model, were fed into the autoencoder HIV classification network model. This network classifier had an area under the ROC curve of 0.86. The network classifier obtained a classification accuracy of 81% in the presence of missing data, compared to an accuracy of 92% obtained for a case without missing data. The missing input estimates obtained from the autoencoder missing data model, were then fed into the HIV feedforward neural network classification model. The network classifier had an area under the ROC curve of 0.8. The network classifier obtained a classification accuracy of 82% in the presence of missing data, compared to an accuracy of 84% obtained for a case without missing data. The result of this study thus suggests that even though autoencoder network models are more accurate and better classifiers for the HIV model than conventional feedforward neural network models, feedforward neural network models perform better in the presence of missing data. Feedforward neural networks are thus more noise resistant than autoencoder networks, due to the fact that the narrow hidden layer of the autoencoder networks causes the input parameters to be highly correlated, meanwhile the feedforward network architecture model ensures that the effect of the input parameters on the output is decoupled and independent.

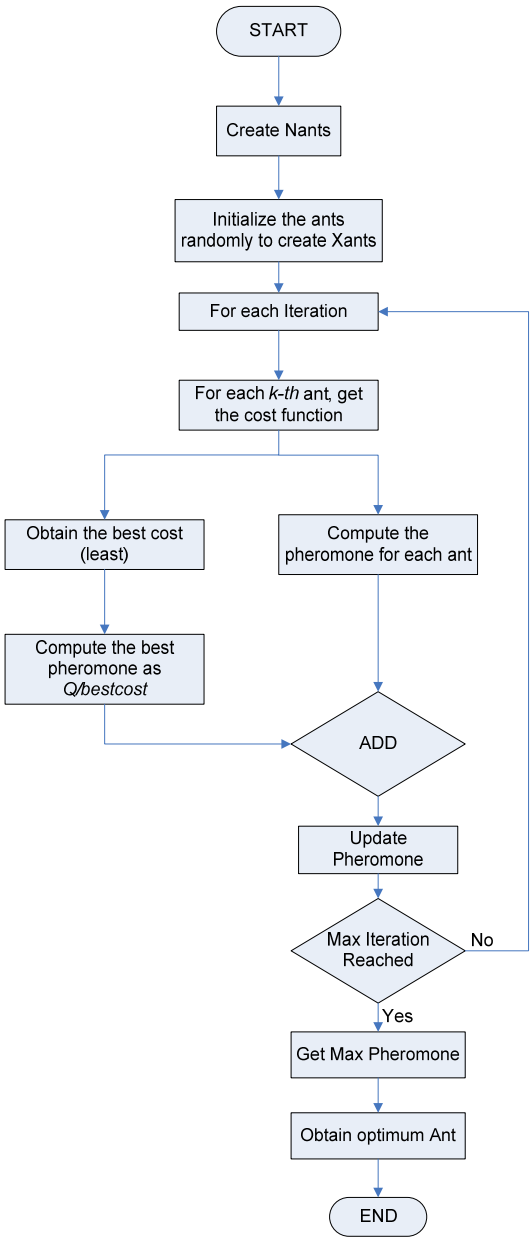

Figure 3.1: Flow Chart of Missing Data Estimation Model

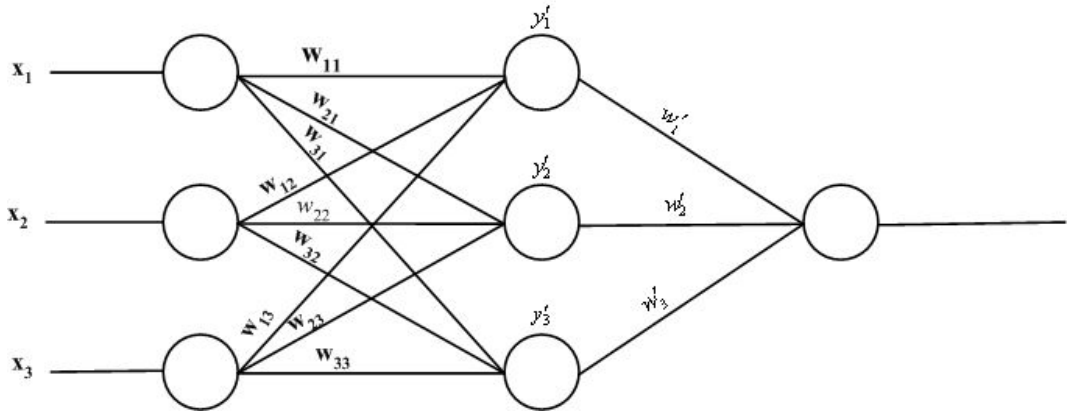

Figure 3.2: Feedforward neural network used for error analysis

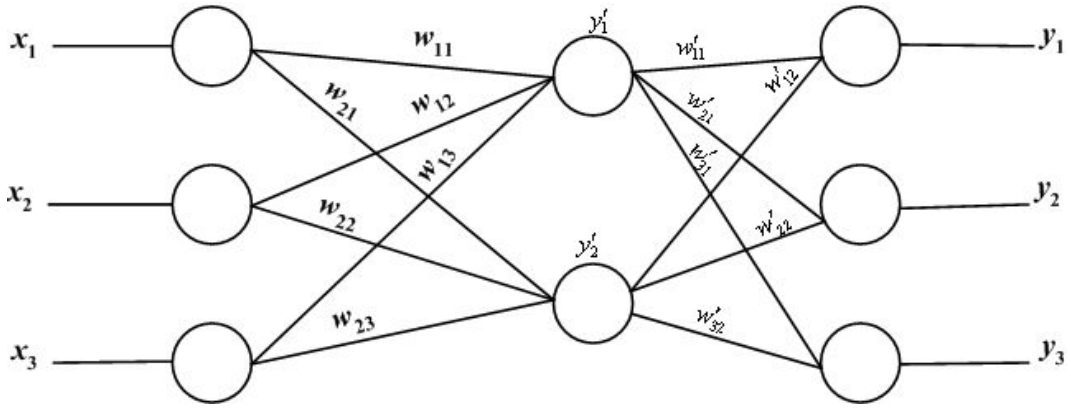

Figure 3.3: Autoencoder neural network used for error analysis

Chapter 4

Understanding Demographic Influences On HIV Susceptibility

4.1 Introduction

In this chapter, a control mechanism is proposed to assess the effect of demographic properties on the HIV status of individuals, based on inverse neural networks, and autoencoder networks-based-on-genetic algorithms. This control mechanism is aimed at understanding whether HIV susceptibility can be controlled by modifying some of the demographic properties. In this chapter, the educational level and gravidity are the demographic parameters in study.

The educational level represents the highest grade successfully completed by the individuals. This parameter can be controlled by enforcing better educational policies. Gravidity is the number of pregnancies, complete or incomplete, experienced by a female. This parameter can be controlled by using a condom and the authors believe this parameter can be used by policy makers to understand the HIV spread. The data set used for this chapter, which was obtained from the South African antenatal seroprevalence survey, comprised of; race, age of the female, age of the partner, educational level, gravidity, province of origin, region of origin and HIV status. Other parameters in the data set are not controllable, since the race, age or place of origin of an individual cannot be modified. The control mechanisms are thus proposed to assess whether the educational level or gravidity can be used to control HIV/AIDS susceptibility. The educational level and gravidity are chosen because these two variables are the only modifiable parameters in the data set and policy can be instituted to affect these variables unlike the other parameters, which are not modifiable by policy. The reason for generating the control models is to understand whether the demographic parameters can be modified to diminish the likelihood of contracting HIV. The first control model is implemented using inverse neural networks and the second model is implemented using autoencoder networks based on genetic algorithms. These models are explained in detail in the next section.

4.2 Methodology

The literature review shows that neural networks have been used for HIV/AIDS modeling and worked well for such models, such as in Poundstone et al. (2004). It was also found from the literature review that neural networks have been used for HIV classification and prediction of HIV status of patients, from symptoms using demographic properties and have yielded good results as in Lee and Park (2001) and Sardari and Sardari (2002). The work in this chapter focuses on creating a model to understand the impact of demographic properties' changes on HIV status of individuals, rather than just relating these properties to the HIV status of the individuals. From the literature review, it was found that little had been done in proposing computational models for HIV control, which could be used to understand how the demographic properties relate to the HIV status of individuals, as well as understand how these demographic influences can be modified to reduce the risk of HIV. An adaptive controller model is thus proposed to help understand how modification of demographic properties can affect HIV risk. The adaptive control model is implemented using inverse neural networks, and autoencoder networks-based-on-genetic algorithms. This study's objectives are thus to:

1. Generate an inverse neural network model to predict a demographic parameter given the HIV status and other demographic parameters, thus modeling output-input relationship.

2. Generate a model using autoencoder networks and genetic algorithms to

predict a missing demographic parameter given the other demographic parameters and the HIV status of the individual.

4.2.1 Generating Inverse Neural Network Model to Predict the Demographic Parameter

The datasets presented in Chapter 1, Section 1.2 were used to create a neural network model. In this model, one of the inputs (educational level) in Fig. 2.1 is replaced by the output (HIV status); meanwhile the output in Fig. 2.1 becomes the replaced input (educational level) (Leke et al. 2007b). This is known as an inverse neural network, since it relates the output to an input. A second inverse neural network was then created with the output being the gravidity. A genetic algorithm was then used to optimize the network parameters (hidden nodes, α, and training cycles). The network was made up of an MLP network comprising of 9 inputs (the educational level in the demographic properties data set being replaced by the HIV status) and 1 output (educational level). The number of hidden units returned by the genetic algorithm was 18, with a weight decay coefficient (alpha) of 0.254 and 984 training cycles, which was found after training, validation and testing. The model for the system is as shown in Fig. 4.1. The demographic inputs are used in the neural network model represented by "autoencoder networks" in Fig. 4.1 to predict the HIV status. If the HIV status is predicted as positive, then the inverse neural network model is used to predict the input parameter value (educational level or

gravidity) required to make the status negative, by replacing the HIV value, in the input data set, by 0.

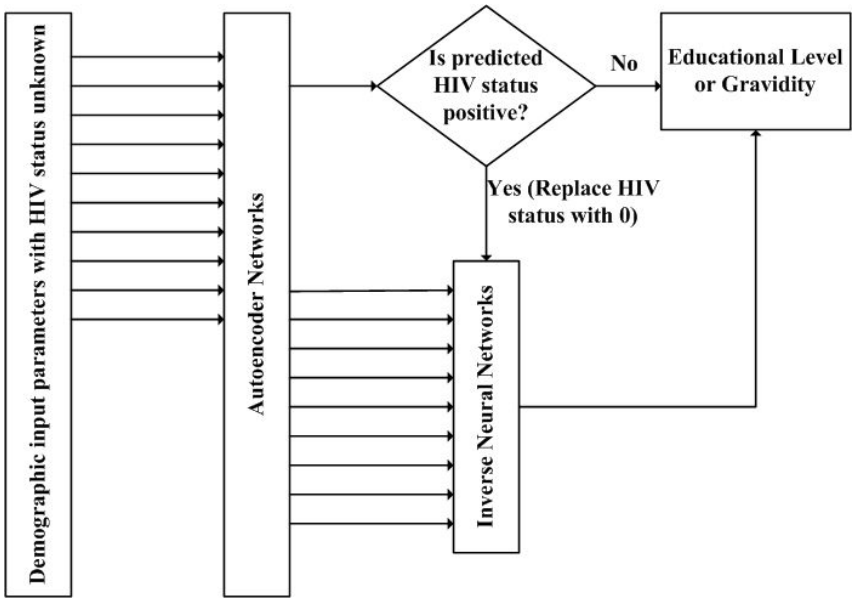

Figure 4.1: Inverse neural network control model

4.2.2 Generating the Genetic Algorithm Model to Estimate the Missing Demographic Parameter

The aim of this model is to assess the educational level that will yield a negative HIV status for an individual. Initially, the prediction model was used to predict the status of individuals from the demographic input data set. For every set of inputs, if the prediction yielded negative, then that educational level or gravidity is kept as the right demographic parameter for that individual. If the output from the prediction model, however, yields a positive status, then the

educational level or gravidity is discarded and estimated using an autoencoder-based-on-genetic algorithm model. A network model is thus created, which is similar to the network model created in Chapter 2. In this model, however, to predict the educational level or gravidity required, the demographic parameter in question (represented by x_3 in Fig. 2.1 and ultimately y_3 in the output node) in the input vector x is assumed as an unknown input, x_u while the other demographic input properties and the HIV status are considered as the known inputs, x_k. These are used in Eqn 2.10 and genetic algorithm is used to minimize Eqn 2.10 thereby obtaining the educational level or gravidity from the HIV status and the demographic properties of the individuals. Fig. 4.2 shows the implementation of this proposed model in a flowchart.

4.2.3 Generating the Model for HIV Control

The overall model is then created using the datasets in Chapter 1, Section 1.2. The first model uses genetic algorithms and autoencoder networks, to predict the educational level or gravidity, using a data estimation model, as explained in Section 4.2.2. The second model uses inverse neural networks, to predict the educational level or gravidity, from the other demographic properties and the HIV status required, as explained in Section 4.2.1. Both models are implemented in Matlab (MATLAB 2004). The structure implemented is shown in Fig. 4.3. The two models are used to generate demographic parameters required to yield a negative status for an individual whose status is predicted

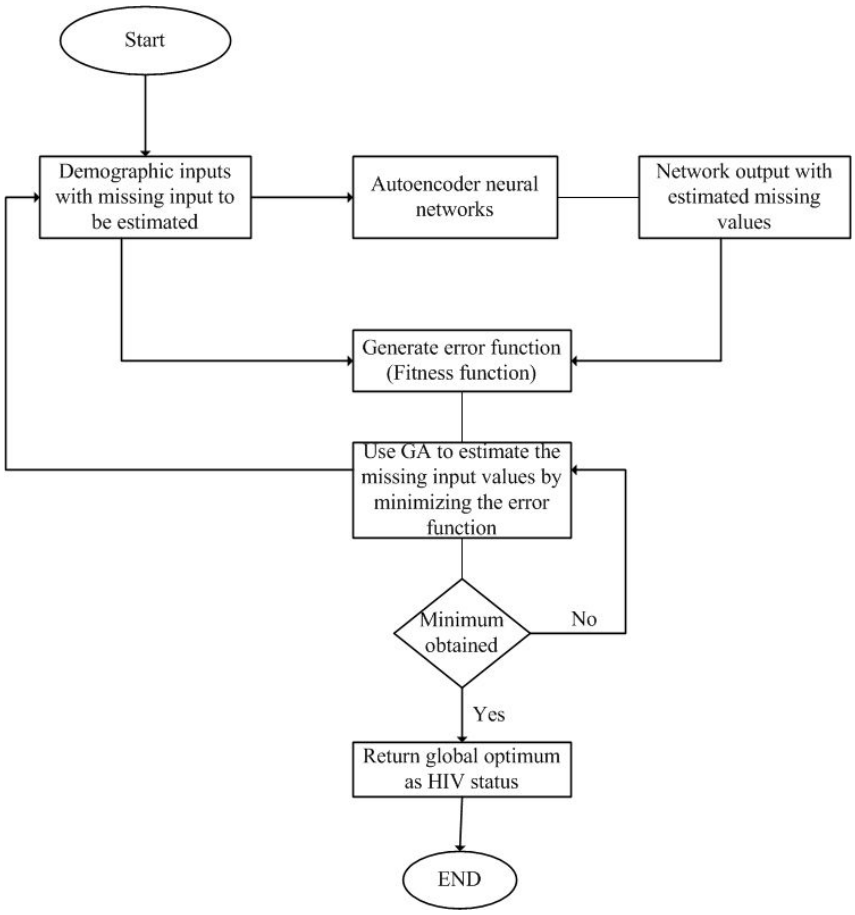

Figure 4.2: Flow chart of the proposed autoencoder missing data estimation model

as positive, thereby finding the demographic parameter which is required for an individual to be less prone or susceptible to contracting HIV. For the two models, a demographic input dataset is sent into a prediction model, implemented in Chapter 2 of this thesis. The predicted HIV status of the network is then verified, and if the result yielded is 1 (positive), the control estimation model, the inverse neural network model implemented in Section 4.2.1, or the autoencoder-based-on-genetic algorithm model, implemented in Section 4.2.2, or a combination of the two models, is then used with the HIV status being replaced with a zero. The educational level thus required for the HIV status to

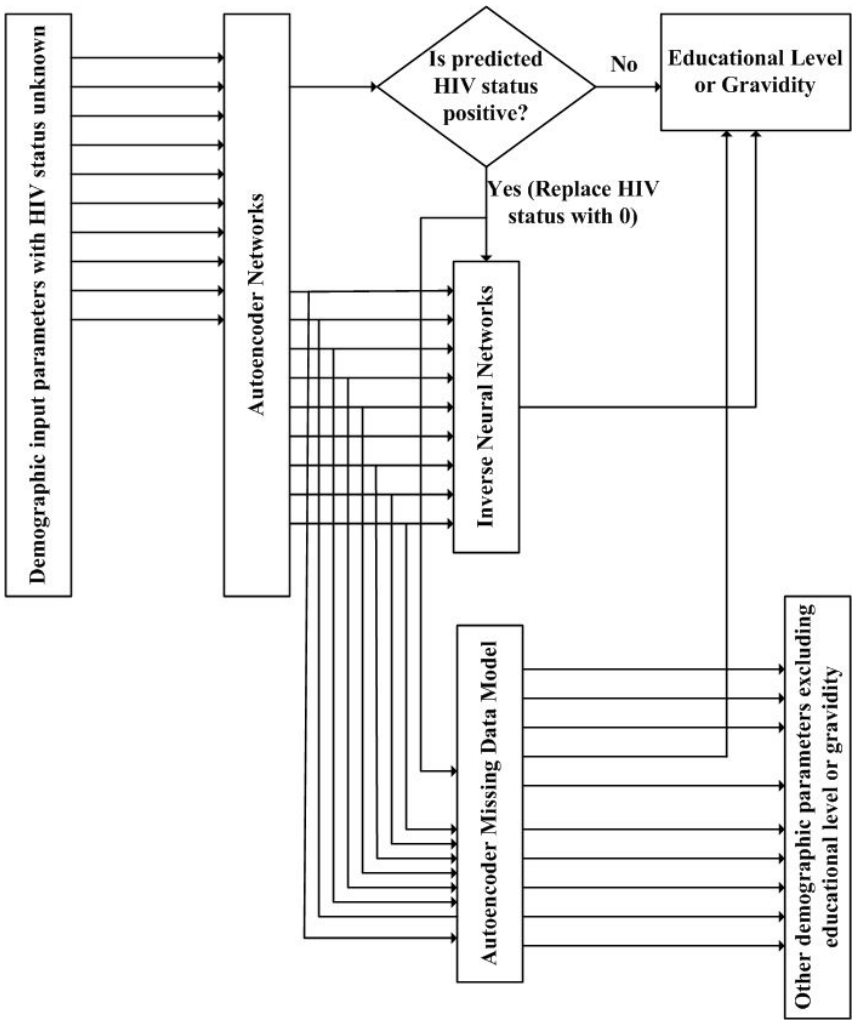

Figure 4.3: Structure of the HIV control model

be zero (negative) is then obtained. The genetic algorithm is used to estimate the required parameter here by minimizing an error equation in Eqn. 2.10.

4.3 Testing the Procedure

The dataset presented in Chapter 1 was used. An autoencoder network comprised of 10 inputs (9 inputs representing the demographic parameters and 1

input representing the HIV status) and 10 outputs was constructed. An inverse neural network comprised of 9 inputs (8 inputs representing the demographic parameters and 1 input representing the HIV status) and 1 output (representing the demographic input to be predicted), was also constructed. GA was used to choose the optimal number of hidden units for both networks. On implementing the GA; the arithmetic cross-over, non-uniform mutation and normalized geometric selection were used. Bounds were set based on maximum and minimum values obtained from the data. The probability of cross-over was chosen to be 0.75 as suggested in Marwala and Chakraverty (2006) and probability of mutation was chosen to be 0.0333 by Marwala and Chakraverty (2006). The GA had a population size of 40 and was run for 100 generations. The GA yielded 2 hidden units as the optimal network that gives the best prediction and classification of the HIV status, from the demographic properties for the autoencoder network model. The GA also yielded 18 hidden units as the optimal network that gives the best prediction of the educational level, from the HIV status and other demographic properties, for the inverse neural network model. The networks were trained using scaled conjugate gradient method (Bishop 1995) with error backpropagation algorithms.

The first experiment investigates the use of the data set to predict the HIV status of individuals from demographic characteristics. The performance analysis for the HIV prediction model is based on classification accuracy and the area under the ROC curve. Experimentation was performed on a computer with a processing power of 3.2 GHz. The optimal number of hidden nodes for

the autoencoder network was 2; hence the structure was 10-2-10. This network gave an accuracy of 92% on the test data sets. The confusion matrix obtained for the above network is as shown in Table 2.1. The ROC curve obtained for the classifier is shown in Fig. 2.6. The area under the ROC curve was found to be 0.86, which according to Monash (2006) is an excellent classifier.

The second experiment used the same dataset as in the first experiment to generate a control model. A feedforward neural network was created with one of the demographic inputs (educational level or gravidity) replaced by the HIV status. The demographic input was then used as the output. The inverse neural network model was thus created. The optimal number of hidden nodes for the inverse neural network model obtained by the genetic algorithm was 18, thus yielding a 9 - 18 - 1 structure. The training time was 46.9980s. When used to predict the educational level of individuals from the other demographic parameters, this network gave an accuracy of 77%. When used to predict the gravidity of individuals from the other demographic parameters, this network gave an accuracy of 82%.

The third experiment consisted of cases where one of the inputs (educational level or gravidity) to the neural network was assumed to be unknown and then estimated using autoencoder networks-based-on-GA. On implementing the GA; the arithmetic cross-over, non-uniform mutation and normalized geometric selection were used. Bounds were set based on maximum and minimum values obtained from the data. The GA had a population size of 20 and was

run for 150 generations. When used to predict the educational level of individuals from the other demographic parameters, this network gave an accuracy of 77%. When used to predict the gravidity of individuals from the other demographic parameters, this network gave an accuracy of 92%.

The last experiment used a combination of the inverse neural networks and the autoencoder-based-on-genetic algorithm models to generate a control model. A majority voting combination scheme was used whereby the training accuracies were used as the weightings of the combination. When used to predict the educational level of individuals from the other demographic parameters, this network gave an accuracy of 74%. When used to predict the gravidity of individuals from the other demographic parameters, this network gave an accuracy of 89%.

The source of errors in the experiment is mainly data set related errors, due to the data being biased towards one class, and neural network training. To minimize the data set errors, reliable data was replicated for the class with less data. To minimize the neural network training errors, standard procedures were used for training, generalization and testing of neural networks. The effects of these errors were, however, minimal on the overall predictability.

The results are summarized in Table 4.1

Table 4.1: Summary of Results

Model	Training Time	Accuracy	Recommended
Inverse NN	46.9980s	77(Educ), 82(Grav)	X
Autoencoder	60.016s	77(Educ), 92 (Grav)	Y
INVNN + Autoencoder	70.3s	74(Educ), 89 (Grav)	X

4.4 Conclusion

In this study, a method based on neural networks and genetic algorithms is proposed to investigate how demographic properties can be used to understand the HIV susceptibility of individuals. The model aims at obtaining the educational level or gravidity, which will make individuals less prone and susceptible to HIV contraction using demographic characteristics. A classifier was first developed using autoencoder networks to classify the HIV status of individuals based on demographic properties, and had a classification accuracy of 92% and an area under the curve of 0.86 for the ROC curve. An adaptive control model was then generated and implemented in Matlab. This model was implemented using inverse neural networks, and using autoencoder networks-based-on-genetic algorithms. The models were tested on a set of demographic characteristics from the South African antenatal data set. The proposed model is able to estimate the educational level using GA and inverse neural networks. The inverse neural network model yields faster results compared to the GA model. The slowness of the genetic algorithm vis--vis the

inverse neural networks model may be attributed to the fact that genetic algorithms tend to converge slowly (Marwala and Chakraverty 2006; Davis 1991). An accuracy of 77% was obtained by the inverse neural network model and an accuracy of 77% was obtained by the genetic algorithm model, for the educational level prediction. An accuracy of 82% was obtained by the inverse neural network model as opposed to 92% for the genetic algorithm model, for the gravidity prediction. The lower accuracy of the inverse neural network model may be attributed to the fact that inverse neural network models depend significantly on the correctness of the model, meanwhile autoencoder networks depend on the correlation of the input parameters due to the narrow hidden layers. A model can thus be developed using inverse neural networks, to effectively assess the demographic parameters required by individuals, in order to control the susceptibility to HIV of individuals. A model can also be developed using autoencoder neural networks and genetic algorithms, to effectively assess the demographic parameters required by individuals. The results of this study show that gravidity is a highly controllable parameter due to its high predictability accuracy from the other demographic properties. The results of this study also show that the educational level can be controlled even though not as effectively as gravidity, due to lower predictability accuracy. This study thus provides a means to understand how the demographic properties affect HIV spread, and can be used by decision-makers and policy-makers to understand the effects of demographic influences on HIV contraction.

Chapter 5

HIV Modelling using Neuro-fuzzy Method, Rough Sets and Rule Extraction

5.1 Introduction

This chapter introduces a new method of using neuro-fuzzy modeling to an HIV modeling and classification problem. The chapter uses the Takagi-Sugeno fuzzy model for HIV analysis. The neuro-fuzzy model is then compared to a rough set approximation for HIV analysis. This rough set approximation is based on a formulation by Tettey et al. (2007). The defuzzification of the neuro-fuzzy

model rule base, and the rules extracted by the rough set approximation, ensures transparency and overcomes the black box-like nature of artificial neural networks. The neuro-fuzzy and rough set rules extracted will be presented. The chapter will conclude by understanding the effect of missing data on the neuro-fuzzy model and comparisons will be drawn with the feedforward model presented in Chapter 2.

Analytical models based on statistical and computational intelligence models have been proposed to model and understand the spread of HIV. These techniques have been applied on quantitative measures collected over the years. A survey of the work performed on HIV analysis was presented in Chapter 2. A neuro-fuzzy model, as well as a rough set model, is now presented as an alternative technique for modeling the disease from demographic data. The methods presented in Chapter 2 had shortcomings in that, even though neural networks perform well for classification purposes, they have a black box nature and the intrinsic output relationship to the input parameters are not easy to interpret. This results in different interpretations by different researchers for the same problem. Background review showed that, to the best of our knowledge, neuro-fuzzy models have not been applied to HIV modeling of the antenatal HIV database and it is anticipated that the results will be more advantageous than neural networks models. Rough set models have also seldom been used for HIV analysis and the formulation investigated in this chapter is based on work by Tettey et al. (2007). Such models offer more significant results in that the models yield rules, which are causal and more interpretable than the

black box results obtained by neural networks. The weights extracted from the neural network during optimization, offers no understanding as to how the demographic characteristics affect the risk of contracting the disease. Neuro-fuzzy models, which offer more intuitive understanding, are presented in the next section. The neuro-fuzzy model will be used to model the risk of HIV from demographic properties, which has fairly accurate prediction ability while offering rules, which better explain the significance of the demographic characteristics on the HIV risk. Rough set theory, which is the basis for formulating the rough set approximation is also presented in the next section. The rough set model yields rules, which will be compared to the neuro-fuzzy model rules. The effect of the presence of missing data in the data set on the neuro-fuzzy model's classification accuracy is further studied.

5.2 Background

5.2.1 Fuzzy Systems and Neuro-fuzzy Modelling

A fuzzy rule-based model suitable for approximation of many systems and functions is the Takagi-Sugeno (TS) fuzzy model (Takagi and Sugeno 1985). Fuzzy logic concepts offer methods of modeling imprecise models, for complex models. Fuzzy set theory derives its background for approximating information and generating uncertain decisions, from human reasoning. Fuzzy logic is based on the theory of fuzzy sets, which relates to classes of objects with

unsharp boundaries in which membership is a matter of degree. The membership function is described by an arbitrary curve suitable from the point of view of simplicity, convenience, speed and efficiency. The process of formulating the mapping from a given input to an output using fuzzy logic is called the fuzzy inference (Jang 1993). The basic structure of any fuzzy inference system is a model that maps characteristics of input data to input membership functions, input membership function to rules, rules to a set of output characteristics, output characteristics to output membership functions, and the output membership function to a single-valued output or a decision associated with the output (Jang et al. 2002). In rule-based fuzzy systems, the relationships between variables are represented by means of fuzzy if-then rules e.g. "If antecedent proposition then consequent proposition". Depending on the particular structure of the consequent proposition, three main types of fuzzy models are distinguished as: (1) Linguistic (Mamdani Type) fuzzy model (Zadeh 1973), (Mamdani 1977) (2) Fuzzy relational model (Yi and Chung 1993) (3) Takagi-Sugeno (TS) fuzzy model (Takagi and Sugeno 1985). A major distinction can be made between the linguistic model, which has fuzzy sets in both antecedents and consequents of the rules, and the TS model, where the consequents are (crisp) functions of the input variables. Fuzzy relational models can be regarded as an extension of linguistic models, which allow for different degrees of association between the antecedent and the consequent linguistic terms. In this work, the TS fuzzy model is employed to develop a model for classifying HIV from demographic characteristics. The TS fuzzy models are relatively easy to identify and their structure can be readily analyzed. In

the TS fuzzy model, the rule consequents are usually taken to be either crisp numbers or linear functions of the inputs (Bersini and Bontempi 1997)

$Rule_i$:IF x is A_i THEN $y_i = \{A\}_i^T + \{B\}_i$, i=1,2,...,M

where $x \in \Re^n$ is the antecedent and $y_i \in \Re^n$ is the consequent of the *i-th* rule. In the consequent, $\{A\}_i$ is the parameter vector and $\{B\}_i$, is the scalar offset. The number of rules is denoted by M and A_i is the antecedent fuzzy set of the *i-th* rule defined by the membership function (Takagi and Sugeno 1985):

$$\mu_i(x) = \Re^n \rightarrow [0, 1] \tag{5.1}$$

The fuzzy antecedent in the TS fuzzy model is normally defined as an and-conjunction by means of the product operator (Takagi and Sugeno 1985)

$$\mu_i(x) = \prod_{j=1}^{p} \mu_i^j(x_j) \tag{5.2}$$

Where x_j is the *j-th* input variable in the p-dimensional input space,and μ_{ij} the membership degree of x_j to the fuzzy set describing the *j-th* rule. $\mu_{ij}(x)$ is the overall truth value of the *i-th* rule. For the input x, the total output y of the TS model is computed by aggregating the individual rule contributions (Mamdani 1977)

$$y = \sum_{i=1}^{M} \mu_i(x) \cdot y_i \tag{5.3}$$

Where μ_i is the normalized degree of fulfillment of the antecedent clause of

rule R_i (Takagi and Sugeno 1985).

$$\widehat{\mu}_i(x) = \frac{\mu_i(x)}{\sum_{j=1}^{M} \mu_j(x)} \tag{5.4}$$

The y_i are called the consequent functions of the M rules and are defined by (Takagi and Sugeno 1985):

$$y_i = w_{i0} + w_{i1}x_1 + w_{i2}x_2 + \cdots + w_{ip}x_p \tag{5.5}$$

Where w_{ij} are the linear weights for the *i-th* rule consequent function. A fuzzy rule-based system can be viewed as a layered network similar to Radial basis function (RBF) artificial neural networks (Tettey and Marwala 2006). The parameters such as membership functions and consequent parameters are to be optimized, using training algorithms inherited from neural networks such as gradient descent methods. There are two approaches to training the neuro-fuzzy models (Tettey and Marwala 2006):

1. Fuzzy rules can be extracted from expert knowledge and used to create a model. The parameters of the models are then refined using data from the system to be modeled.

2. The number of rules can be determined from the quantitative data set using a model selection technique. The parameters are then optimized using data from the system to be modeled. The TS model, also considered as universal approximators, is most popular with data-driven identification, such as the HIV model.

Fuzzy Rules Extraction

As presented in Section 5.2.1, the fuzzy network realizes the inference mechanism of a Takagi-Sugeno fuzzy model, based on a collection of rules of the form (Bersini and Bontempi 1997)

$Rule_i$:IF x is A_i THEN $y_i = \{A\}_i^T + \{B\}_i$, i=1,2,...,M

To realize the fuzzy inference mechanism, a network with three layers is used.

1. Layer L_1. Units in this layer receive input values $(x_1, x_2, ..., x_n)$ and act as fuzzy sets representing the terms of the corresponding input variable. Nodes in this layer are arranged into H groups: each group representing the IF-part of a fuzzy rule. Each node $i_k \epsilon L_1$ receives the input variable concerned, that is x_i and computes the membership value $\mu_{ik}(x_i)$ that specifies the degree to which the input value x_i belongs to the fuzzy set A_i^k, defined by a Gaussian membership function (Figueiredo and Gomide 1999):

$$\mu_{ik}(x_i) = e^{-(x_i - w_{ik})^2/\sigma_{ik}^2} \tag{5.6}$$

   where w_{ik} and σ_{ik} are the center and width of the Gaussian function. Hence the output node $i_k \epsilon L_1$ is in the range [0, 1] and is computed by the following function (Figueiredo and Gomide 1999):

$$f_{ik}^{(1)}(x_i) = e^{-(x_i - w_{ik})^2/\sigma_{ik}^2} \tag{5.7}$$

2. Layer L_2. The number of nodes in this layer is equal to the number

of fuzzy rules. A node in this layer represents a fuzzy rule; for each node, there are n fixed links from the input nodes representing the *IF*-part of the fuzzy rules. The *kth* node performs the *AND* operation for precondition matching of the *kth* rule by a product operator (Figueiredo and Gomide 1999; Bersini and Bontempi 1997); thus the output of this node is:

$$f_k^{(2)}(\bar{x}) = \prod_{i=1}^{n} f_{ik}^{(1)}(x_i) \tag{5.8}$$

3. Layer L_3. Nodes in this layer represent the output variables of the system. Each node j acts as a defuzzifier and computes the output values according to the following equation (Figueiredo and Gomide 1999):

$$f_j^{(3)}(\bar{x}) = \frac{\sum_{k=1}^{N} f_k^{(2)}(\bar{x}).\nu_{kj}}{\sum_{k=1}^{N} f_k^{(2)}(\bar{x})} \tag{5.9}$$

where ν corresponds to the consequent weights, $\bar{x}$ is a vector containing the input parameters, and N is the number of fuzzy rules.

The weights of the network correspond to the Gaussian membership function parameters $\{w_{ik}\}, \{\sigma_{ik}\}$ and to the consequents ν_{kj}. In other words, each node $k \epsilon L_2$ is associated to two weight vectors $\bar{w}_k = (w_{1k}, ..., w_{nk}), \bar{\sigma}_k = (\sigma_{1k}, ..., \sigma_{nk})$ and one consequent weight vector $\bar{\nu}_k = (\nu_{k1}, ..., \nu_{km})$. Upon obtaining these parameters the rules can be written as (Bersini and Bontempi 1997):

$Rule_k$: IF (x_1 is A_1^k) AND...AND (x_n is A_n^k) THEN (y_i is ν_{k1}) AND...AND (y_m is ν_{km}), which is expressed mathematically as: $y_i = \{A\}_i^T + \{B\}_i$, i=1,...,m

5.2.2 Rough Set Theory

The rough sets theory provides a technique of reasoning from vague and imprecise data (Goh and Law 2003). The technique is based on the assumption that information of interest is associated somehow with *some information* of the universe of the discourse (Komorowski et al. 1999; Yang and John 2006). Objects with the same information are *indiscernible* in the view of the available information. An elementary set consisting of indiscernible objects forms a basic granule of knowledge. A union of elementary sets is referred to as a crisp set, otherwise the set is considered to be rough. The next few subsections briefly introduce concepts that are common to rough set theory.

Information System

An information system (Λ), is defined as a pair $(\mathbf{U}, A)$ where $\mathbf{U}$ is a finite set of objects called the universe and A is a non-empty finite set of attributes as shown in Eqn. 5.10 below (Yang and John 2006):

$$\Lambda = (\mathbf{U}, A) \tag{5.10}$$

Every attribute $a \in A$ has a value which must be a member of a value set V_a of the attribute a.

$$a : \mathbf{U} \rightarrow V_a \tag{5.11}$$

A rough set is defined with a set of attributes and the indiscernibility relation between them. Indiscernibility is discussed next.

Indiscernibility Relation

Indiscernibility (I) relation is one of the fundamental ideas of rough set theory (Grzymala-Busse 1992). Indiscernibility simply implies similarity (Goh and Law 2003). Given an information system Λ and subset $B \subseteq A$, B determines a binary relation $I(B)$ on $\mathbf{U}$ (Goh and Law 2003):

$$(x, y) \in I(B) \quad iff \quad a(x) = a(y) \tag{5.12}$$

for all $a \in B$ where $a(x)$ denotes the value of attribute a for element x. Eqn. 5.12 implies that any two elements, x and y, that belong to $I(B)$ should be identical from the point of view of a.

Information Table and Data Representation

An Information Table (IT) is used in rough sets theory as a way of representing the data. The data in the IT are arranged based on their condition attributes

($\mathcal{C}$) and a decision attribute ($\mathcal{D}$). Condition attributes and decision attribute are analogous to the independent variables and a dependent variable (Goh and Law 2003). These attributes are divided into $C \cup \mathcal{D} = Q$ and $C \cap \mathcal{D} = \emptyset$. An example of an IT is given in Table 5.1.

Data is represented by a table where each row represents an instance, sometimes referred to as an object. Every column represents an attribute which can be a measured variable. In Table 5.1, HIV is the decision variable whereas, *race*, *education*, *gravidity*, *parity* and *ages* of both parents are the condition attributes. This kind of a table is also referred to as Information System (Komorowski et al. 1999).

Decision Rules Induction

Rough sets analysis also involve generating decision rules for a given IT. The rules are normally determined based on condition attributes values (Goh and Law 2003). The rules are presented in an *if* CONDITION(S)-*then* DECISION format.

Set Approximation

There are various properties of rough sets that have been presented in the literature (Pawlak 1991). Some of the properties are discussed below.

Table 5.1: Extract of the HIV database used

	Race	Educ	Gravid	Parity	Age	Father	HIV
$obj^{(1)}$	1	11	1	2	35	41	0
$obj^{(2)}$	2	13	1	0	20	22	0
$obj^{(3)}$	3	10	2	0	28	27	1
$obj^{(4)}$	2	12	1	1	20	33	1
$obj^{(5)}$	3	9	6	2	28	28	0
$obj^{(6)}$	2	9	2	1	26	27	0
$obj^{(7)}$	2	7	1	0	15	35	0
$obj^{(8)}$	1	0	4	3	26	28	0
$obj^{(9)}$	4	7	1	0	15	29	1
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
$obj^{(n)}$	1	11	1	0	20	22	1

Lower and Upper Approximation of Sets

The lower and upper approximations are defined on the basis of indiscernibility relation discussed previously. The lower approximation is defined as the collection of cases whose equivalent classes are contained in the cases that need to be approximated whereas the upper approximation is defined as the collection of classes that are partially contained in the set that needs to be approximated (Rowland et al. 1998).

Let **concept** X be defined as a set of all cases defined by a specific value of

the decision and x represent an object. Any finite union of elementary set, associated with B is called a $B - definable$ set (Grzymala-Busse 1992). The set X is approximated by two $B - definable$ sets, referred to as the B-lower approximation denoted by $\underline{B}X$ and B-upper approximation, $\overline{B}X$. The B-lower approximation is defined as (Grzymala-Busse 1992),

$$\{x \in \mathbf{U} | [x]_B \subseteq X\} \tag{5.13}$$

and the B-upper approximation is defined as (Grzymala-Busse 1992):

$$\{x \in \mathbf{U} | [x]_B \cap X \neq \emptyset\} \tag{5.14}$$

where $[x]_B$ denotes an equivalent class of $I(B)$ containing the variable x. There are other methods that have been reported in the literature for defining the lower and upper approximations for a completely specified decision tables. It follows from the properties that a crisp set is only defined if $\underline{B}(X) = \overline{B}(X)$. Roughness therefore is defined as the difference between the upper and the lower approximation.

Rough Membership Functions

Rough membership function is a function $\mu_A^x : \mathbf{U} \rightarrow [0, 1]$ that when applied to object x, quantifies the degree of overlap between set X and the indiscernibility

set to which x belongs. The rough membership function is used to calculate the plausibility and is defined as (Grzymala-Busse 1992)

$$\mu_A^X(X) = \frac{|[X]_B \cap X|}{|[X]_B|} \tag{5.15}$$

5.2.3 Rough Sets Rule Extraction and Analysis

Let us assume an input space with n objects, each with m attributes. The output of the algorithm is a set of certain and possible rules and the algorithm is presented Algorithm 1 (Tettey et al. 2007).

5.3 Methodology

The literature review showed that models for HIV prediction and classification have been developed using conventional feedforward neural networks architectures and have worked well. However, it was found from the literature review that neuro-fuzzy models have not been applied to HIV modeling, for prediction and classification. Our work thus focuses on proposing a methodology for HIV classification from demographic properties using Takagi-Sugeno neuro-fuzzy models. Then the proposed TS neuro-fuzzy model is compared to a conventional feedforward neural networks model, by creating a feedforward MLP neural network model and comparing the results with the TS neuro-fuzzy

model results. It was also found from literature review that little work had been done using rough set approximation for HIV analysis. Rough set have an advantage of yielding more explicit and interpretable rules. The rough set rules extracted will be compared to the neuro-fuzzy rules extracted. The final objective of this Chapter is to investigate the effect of missing data on the classification accuracy. This effect is also compared to the effect of missing data on the feedforward neural network model.

5.3.1 Creation of TS Neuro-Fuzzy Model

A Takagi-Sugeno neuro-fuzzy model was created and trained with demographic data obtained from the South African antenatal seroprevalence survey of 2001 (HealthDept 2005). This is a national survey data set, and any pregnant women attending selected public health care clinics participating for the first time in the survey were eligible. The demographic input variables obtained include; age of mother, age of partner, educational level of mother, gravidity (number of complete or incomplete pregnancies), parity (number of complete pregnancies), province of origin, race of mother, and region of origin. The qualitative variables such as the province of origin, race of mother and region of origin were encoded to integers. The output of the model was the HIV status, which was encoded using an integer scheme, whereby a 1 represents a positive HIV status meanwhile a 0 represents a negative HIV status. The network thus looked as in Fig. 5.1, with parameters defined in earlier equations

and P is defined by Eqn 5.3.

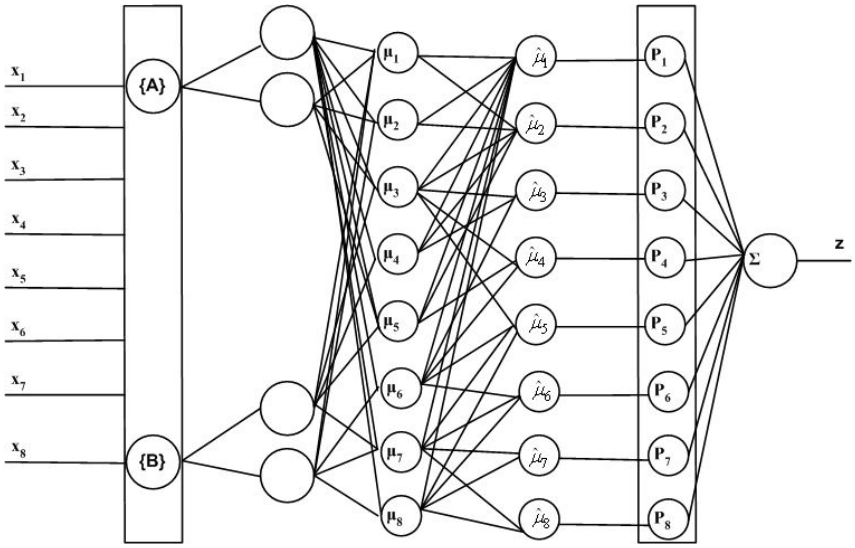

Figure 5.1: Neuro-fuzzy Network Structure

The number of membership functions was then optimized. Fig. 5.2 shows the relationship between mean square error (MSE) and number of clusters meanwhile Fig. 5.3 shows the relationship between the accuracy of classification and the number of clusters.

Fig. 5.2 and Fig. 5.3 show that two clusters gave the least MSE as well as the highest accuracy and was thus the optimal number of clusters. This is further confirmed by Table 5.2, which shows that with a cluster number of 2, that is, two fuzzy rules, the least variance was obtained using a 10-fold cross validation data set.

The fuzziness parameter was then optimized as shown in Fig. 5.4. From Fig. 5.4, it can be deduced that a fuzziness parameter of 2.0 yielded the best accuracy of classification. The lowest accuracy was obtained for a fuzziness

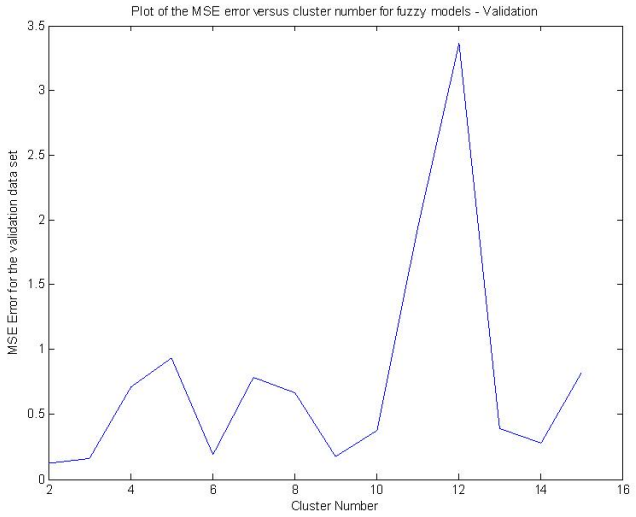

Figure 5.2: Plot of MSE Error versus Cluster parameter

parameter of 3.

A TS neuro-fuzzy model was thus created with 2 clusters and a fuzziness parameter of 2.0. This neuro-fuzzy model had 8 demographic inputs and one output depicting the HIV status of individuals. The TS neuro-fuzzy model was compared to a neural network feedforward model. The Feedforward model used for comparison was presented in Chapter 2 Section 2.3.2.

5.3.2 Rough Sets Formulation

The process of fuzzy rule extraction requires several steps. A summary of the steps that have been taken to formulate the rough set approximations and obtain rules, is shown in Fig. 5.5. The remainder of the section explains the details of the process.

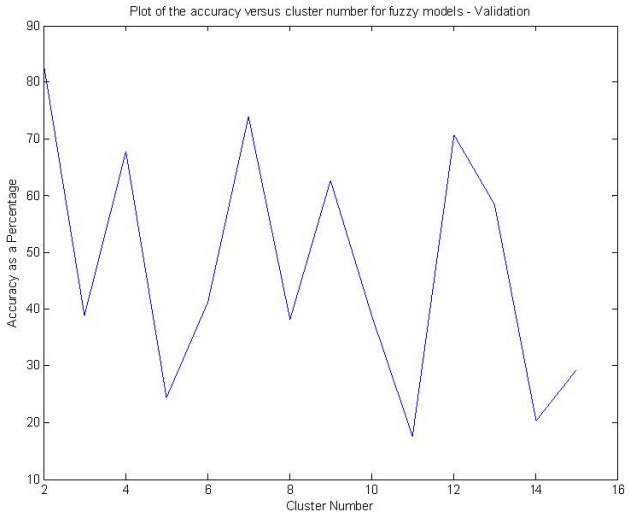

Figure 5.3: Plot of Accuracy versus cluster number

Data Preprocessing

The HIV/AIDS data that is used in this work is obtained from a survey performed on pregnant women (HealthDept 2005). Like all data in raw form, there are several steps that need to be taken in order to ensure the data is in usable form. There are several types of outliers that have been identified in the data. Firstly, some of the data records were not complete. This is probably due to the fact that the people being surveyed omitted certain information and also errors made by the person who manually recorded the surveys onto a spreadsheet. This together resulted in certain entries being incomplete. All such entries were deleted from the data. The second form of outliers were from incorrectly entered variables. For instance *Gravidity* is defined as the number of times a woman has been pregnant and *parity* is described as the number of times a woman has given birth. Given that the survey was for pregnant women

Table 5.2: Variances of the MSE error with respect to cluster number

Cluster Number	Average Error	Variance
2	0.12514	0.000077746
3	0.16327	0.00019226
4	0.71186	0.00032077
5	0.93128	0.12206
6	0.18855	0.00021238
7	0.78362	0.0012216
8	0.66805	0.0002408
9	0.17207	0.0002148
10	0.37949	0.0014
11	1.9294	0.0072989
12	3.3632	0.033302
13	0.3931	0.00039738
14	0.27769	0.0015327
15	0.82516	0.0037677

any instance where the women had a *gravidity* value of zero but a *parity* value greater than zero was deleted. Furthermore, cases where a woman had a *parity* value greater than the *gravidity* were also removed.

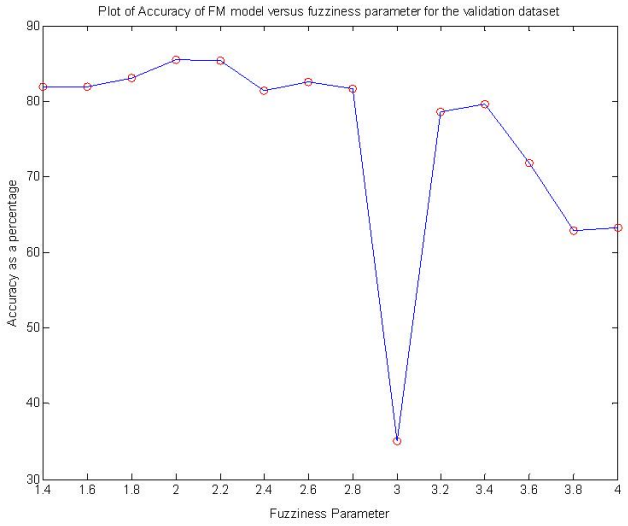

Figure 5.4: Plot of Accuracy versus Fuzziness parameter

Figure 5.5: Steps required to formulate rough set approximation and rule extraction

Variable Discretisation

The discretisation defines the granularity with which we would like to analyse our universe of discourse. If one chooses to discretise the variables into a large number of categories the rules extracted are more complex to analyse. Therefore, if one would like to use the rough sets for rule analysis and interpretation rather than for classification it is advisable that the number of categories be as small as possible. A GA based model to obtain the number of categories was implemented by Crossingham and Marwala (2007). For the purposes of this work the input variables have been discretised into four categories. A

description of the categories and their definition is shown in Table 5.3.

Table 5.3: A table showing the discretised variables.

Race	**Age Mother**	**Educ**	**Grav**	**Parity**	**Age Father**	**HIV Stat**
White	Teen	None	Zero	Zero	Teen	Neg
African	Young	Prim	Low	Low	Young	Pos
Asian	Mature	Sec	High	High	Mature	-
Coloured	Old	Uni	V. High	V. High	Old	-

5.4 Results and Discussions

The demographic and medical data, used in this study, came from the South African antenatal seroprevalence survey of 2001 (HealthDept 2005). Eight demographic input variables were used; age of mother, age of partner, educational level of mother, gravidity, parity, province of origin, race of mother, and region of origin. The genetic algorithm, used for the optimizing the feedforward neural network model parameters used arithmetic cross-over, non-uniform mutation and normalized geometric selection (Davis 1991). The probability of cross-over was chosen to be 0.75 as proposed in Marwala and Chakraverty (2006). The probability of mutation was chosen to be 0.0333 as recommended by Marwala and Chakraverty (2006). Genetic algorithm had a population of 40 and was run for 150 generations. The first experiment investigated the use of TS neuro-fuzzy models for HIV modeling. Two clusters were used, which as stated earlier was found to be the optimal number of clusters for the HIV model. A

fuzziness parameter of 2 was also used, which was found as the most accurate during the optimization stage. The proposed TS neuro-fuzzy model obtained an HIV classification accuracy of 86%. The confusion matrix obtained for the above model is shown in Table 5.4. The ROC curve for this classification is

Table 5.4: Confusion Matrix of TS Neuro-fuzzy classifier Classifier

Confusion Matrix	Predicted Positive	Predicted Negative
Actual Positive	355	145
Actual Negative	0	500

shown in Fig. 5.6 and the area under the curve was computed as 0.82, thus giving a very good classifier according to Monash (2006).

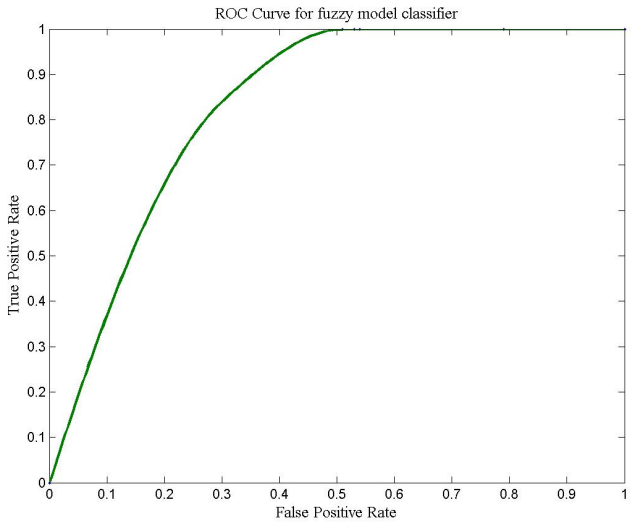

Figure 5.6: ROC curve of Fuzzy Model classifier

The second experiment investigated the use of conventional feedforward neural network MLP architecture to classify the HIV status of an individual using the demographic input properties. The MLP was constructed with 8 inputs and 1 output. A GA was then used to obtain the optimal structure and yielded

an optimal number of hidden units of 77; hence the structure was 8 - 77 - 1. The performance analysis for this network model is also based on classification accuracy and the area under the ROC curve. This network gave an accuracy of 84%. The confusion matrix obtained for the above network is as shown in Table 5.5. The ROC curve obtained for this classification is shown in Fig.

Table 5.5: Confusion Matrix of Feedforward MLP Classifier

Confusion Matrix	Predicted Positive	Predicted Negative
Actual Positive	680	313
Actual Negative	0	993

5.7 and the area under this ROC curve obtained was 0.8, which according to

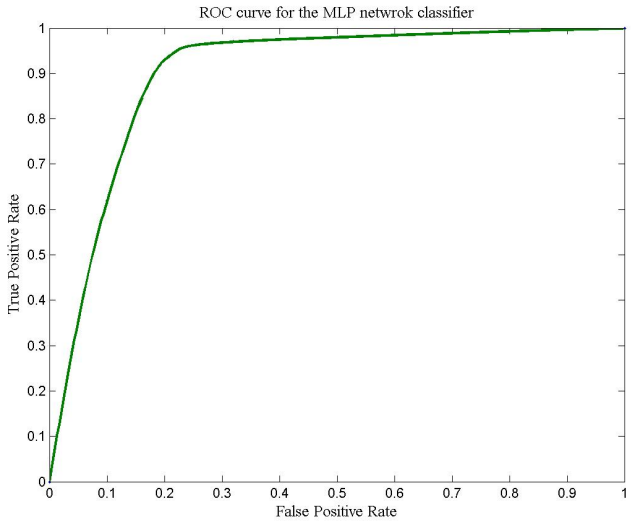

Figure 5.7: ROC curve of feedforward MLP model

Monash (2006) is a very good classifier.

The last experiment investigated the effect of missing data on the classification accuracy of the two models. In the database, one common missing data value is

the educational level of the female, which was missing from 142 individuals (8% of our demographic database). Since this parameter is included in the neural network classification models developed in this study, a model to estimate this missing input parameter using ant colony optimization (ACO), given that the demographic parameters age group, region of origin, age gap, gravidity, parity, province and race, are known was used. More details on this can be found in Chapter 3, as well as in Leke and Marwala (2006). The estimated missing demographic data was then combined with the known demographic data from the antenatal data set, to classify the HIV status of individuals. The accuracy obtained for the HIV classification in the presence of missing data was 83% for the TS neuro-fuzzy model compared to 86% for the case where no data is missing. The effect of the presence of missing data on the TS neuro-fuzzy model thus caused a difference in accuracy of 3%, which is not really significant.

Similarly, the accuracy obtained for the HIV classification in the presence of missing data was 82% for the feedforward neural network model compared to an accuracy of 84% for a case with no missing data. The effect of the presence of missing data on the feedforward neural network thus caused a difference in accuracy of 2%, which is also not really significant.

The membership functions obtained for the various inputs are as shown in Fig. 5.8.

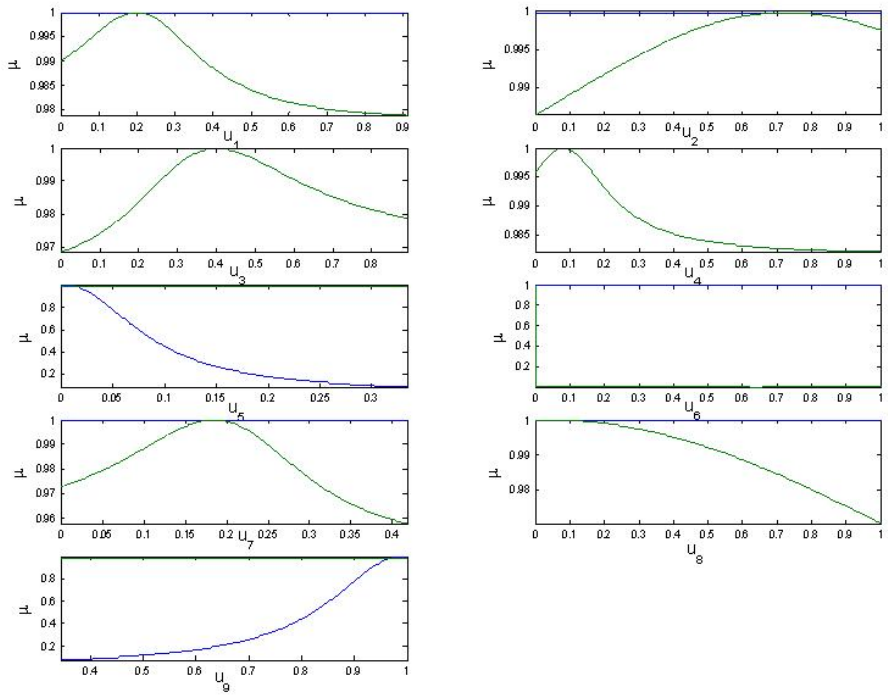

Figure 5.8: Membership functions for various inputs

5.5 Rules Extraction

In this section, the rules extracted by the neuro-fuzzy model and the rough set model are presented. A comparison is then done on the rules obtained by the two models.

5.5.1 Fuzzy Rule Extraction

The TS neuro-fuzzy model used for HIV classification in the previous section can also be used for rule extraction. The background on fuzzy rules extraction was presented in Section 5.2.1. Two fuzzy rules can be extracted from the

model, which are shown below:

1. If u_1 is A_{11} and u_2 is A_{12} and u_3 is A_{13} and u_4 is A_{14} and u_5 is A_{15} and u_6 is A_{16} and u_7 is A_{17} and u_8 is A_{18} then

$$y(k) = 0.645u_1 - 0.233u_2 + 1.833u_3 - 1.846u_4 - 0.164u_5 - 2.64u_6 + 0.024u_7 - 1.312u_8 + 1.63 \tag{5.16}$$

2. If u_1 is A_{21} and u_2 is A_{22} and u_3 is A_{23} and u_4 is A_{24} and u_5 is A_{25} and u_6 is A_{26} and u_7 is A_{27} and u_8 is A_{28} then

$$y(k) = -0.563u_1 + 0.004u_2 + 0.335u_3 + 0.269u_4 - 0.615u_6 - 0.146u_7 - 0.755u_8 + 1.309 \tag{5.17}$$

The symbols from u_1 to u_8 are the input vector which are age of mother (u_1, age of partner (u_2, educational level of mother (u_3, gravidity (u_4, parity (u_5, province of origin (u_6, race of mother (u_7, and region of origin (u_8. The rest of the symbols are as previously defined. The rules extracted can be converted so that they are represented in the commonly used linguistic terms. However, only the antecedent of the fuzzy inference models can be translated into linguistic terms. The consequent parts of the rule are still represented as mathematical expressions. The translated fuzzy rules, after pruning the parameters (to remove redundant fuzzy sets) can be written as follows:

1. If Gravidity is high *AND* Education is low *AND* Parity is low *AND* Age gap between the partners is high then

$$y(k) = 0.291u_1 - 0.782u_2 - 0.575u_3 - 1.221u_4 + 1.808 \tag{5.18}$$

2. If Gravidity is low *AND* Education is high *AND* Parity is high *AND* Age gap between the partners is low then

$$y(k) = -4.16u_1 + 0.613u_2 + 4.845u_3 + 0.949u_4 - 0.085 \tag{5.19}$$

This model can be validated using expert knowledge of the problem. For example, if the gravidity is high, the educational level of the individual is low, the parity is high, and the Age gap between the partners is high, there is a high chance that the individual will be HIV positive. If values of high Gravidity and Age gap which have membership values of 1 and low values of education and parity with a membership value of 0 are used, the model gives a prediction of 0.8786, which is higher than the threshold 0.52 obtained from the ROC curve. The neuro-fuzzy model thus offers a model which effectively classifies the HIV status of individuals from demographic data with an added advantage of rules extraction. Also, the slightly higher accuracy can be attributed to the fact that neuro-fuzzy networks have the ability to incorporate existing domain knowledge as well as to establish relationships from the data. This is an advantage over feedforward networks.

5.5.2 Rough sets rule extraction

Applying the rough sets definitions of the HIV gives 130 unique discernible cases and 95 indiscernible cases. This means the data is only representative of 225 combinations of the variables out of the total possible unique sets of

4096. The 130 discernible cases are part of the lower approximation set and can be used to form rules which are assumed to always hold. Examples of seven extracted discernible rule are shown below:

1. **If** Race = AF **and** Mother's Age = Young **and** Education = Secondary **and** Gravidity = Low **and** Parity = Low **and** Father's Age = Young **then** Probability of being HIV Negative is High

2. **If** Race = AF **and** Mother's Age = Mature **and** Education = Secondary **and** Gravidity = Low **and** Parity = Low **and** Father's Age = Young **then** Probability of being HIV Negative is High

3. **If** Race = AF **and** Mother's Age = Mature **and** Education = Primary **and** Gravidity = Low **and** Parity = Low **and** Father's Age = Mature **then** Probability of being HIV Positive is High

4. **If** Race = AF **and** Mother's Age = Old **and** Education = Primary **and** Gravidity = Low **and** Parity = Low **and** Father's Age = Young **then** Probability of being HIV Positive is High

5. **If** Race = CO **and** Mother's Age = Teenager **and** Education = Primary **and** Gravidity = Low **and** Parity = Low **and** Father's Age = Teenager **then** Probability of being HIV Positive is High

6. **If** Race = CO **and** Mother's Age = Mature **and** Education = Tertiary **and** Gravidity = Medium **and** Parity = Low **and** Father's Age = Mature **then** Probability of being HIV Negative is High

7. **If** Race = WH **and** Mother's Age = Young **and** Education = Secondary **and** Gravidity = Low **and** Parity = Low **and** Father's Age = Young **then** Probability of being HIV Negative is High

The rest of the 95 rules can only be stated with a certain plausibility. Examples of five indiscernable rules which are stated with a given plausibility are shown below. The plausibility of the risk of HIV is stated in terms of a positive HIV/AIDS status. Since the negative and positive status are mutually exclusive, the plausibility of a negative HIV/AIDS status is found by $\mu_A^X(X)_{neg} = 1 - \mu_A^X(X)_{pos}$.

1. **If** Race = AF **and** Mother's Age = Teenager **and** Education = Secondary **and** Gravidity = Low **and** Parity = Low **and** Father's Age = Young **then** HIV is Positive with plausibility = 0.3333

2. **If** Race = CO **and** Mother's Age = Young **and** Education = Secondary **and** Gravidity = Low **and** Parity = Low **and** Father's Age = Mature **then** HIV is Positive with plausibility = 0.1786

3. **If** Race = AF **and** Mother's Age = Young **and** Education = Secondary **and** Gravidity = Low **and** Parity = Low **and** Father's Age = Young **then** HIV is Positive with plausibility = 0.3529

4. **If** Race = AF **and** Mother's Age = Young **and** Education = Secondary **and** Gravidity = Low **and** Parity = Low **and** Father's Age = Young **then** HIV is Positive with plausibility = 0.2

5. **If** Race = WH **and** Mother's Age = Young **and** Education = Secondary **and** Gravidity = Low **and** Parity = Low **and** Father's Age = Young **then** HIV is Positive with plausibility = 0.3071

The rules extracted from the data are assumed to be sufficiently representative of the social context of South Africa. Accepting that the data describes the South African context, the data can then be used to identify critical areas where policies need to be put in place to control the disease spread. For example, rule no. 3 tells us that within the dataset black matured couples with a low education level always gave a high probability of being HIV positive. This can indicate the need for the Government to strengthen efforts of reducing illiteracy amongst the mature black population.

5.5.3 Rules Comparison

It can be seen from Section 5.5.2 that the rules extracted using the rough set approach are more explicit and easy to interpret than those extracted using the neuro-fuzzy approach presented in Section 5.5.1. The disadvantage of the TS neuro-fuzzy is that the consequent expressions are expressed in a mathematical form, which need expert knowledge of the problem domain to be understood, meanwhile the rough set rules are already in a simple understandable form. It seems on the other hand that the granularity of the variables is compromised when data is discretised for the rough set approach. The obvious consequence

of this can be that the classification accuracy is affected.

Algorithm 1: Proposed algorithm for classification tasks

input : Condition and Decision Attributes

output: Certain and Possible Rules

forall (*conditionattributes* $1 \rightarrow n$) **do**

← Compute the lower approximation, for each subset B, with q attributes for each X_l as:

$\underline{B}(X_l) = \{(obj^{(i)}) | 1 \leq i \leq n, obj^{(i)} \in X_l, B_k^c(obj^{(i)}) \subseteq X_l, 1 \leq k \leq |B(obj^{(i)}|\}$

$\longrightarrow$ where $B(obj^{(i)})$ is a set of equivalent classes derived from attribute subset B, whereas $B_k^c(obj^{(i)})$ is certain part of the k^{th} equivalent class in $B(obj^{(i)})$.

approximations of each subset, B.

← Remove certain rules with condition parts which are more specific.

← Compute the upper approximations of each subset B, with q attributes as: $\overline{B}(X_l) = \{(obj^{(i)}, symbol^{(i)}) | 1 \leq i \leq n, B_k^c(obj^{(i)}) \cap X_l \neq \emptyset, B_k^c(obj^{(i)}) \not\subset X_l, 1 \leq k \leq |B(obj^{(i)}|\}$

← Calculate the plausibility measures of each equivalent classes in an upper approximation for each X_l as: $(B_k^c(obj^{(i)})) = \frac{|B_k^c(obj^{(i)}) \cap X_l|}{|B_k^c(obj^{(i)})|}$

← Derive the possible rules from the upper approximations of each subset, with the plausibility measure recalculated.

← Remove possible rules with conditions parts that are more specific and plausibility measure less or equal to those of other possible or estimated objects.

← Output certain rules and possible rules

end

The results obtained are summarised in Table 5.6.

Table 5.6: Summary of Results

Method	Accuracy	Effect of Missing Data	Ease of Rules
FeedForward Network	84	82	Black-Box
TS Neuro-fuzzy	86	83	Mathematical
Rough Set Model	-	-	Simple Linguistic

5.6 Conclusion

A background on HIV analysis using computational intelligence has been presented in this chapter. The previous models lacked transparency and possessed a black box-like nature. A model based on Takagi-Sugeno neuro-fuzzy models, using training methods adapted from neural networks is proposed for HIV classification using the South African antenatal demographic data set. The model proposed has a classification accuracy of 86% compared to a classification accuracy of 84% for the conventional feedforward neural network model. The effect of missing data on the TS neuro-fuzzy model is then analyzed. It is found that this model obtains a classification accuracy of 83% in the presence of missing data. The feedforward network obtains an accuracy of 82% in the presence of missing data. The TS neuro-fuzzy model is thus slightly more accurate than the feedforward neural network model, and is as much resistant to

the presence of noise as the feedforward model. The impact of missing data is, however, not too significant. The TS neuro-fuzzy model is further expressed as fuzzy rules by defuzzification, into common linguistic terms that are readable. Rough sets are then used to analyse the HIV database. It was observed that the rules extracted using the rough set approach are more explicit and easy to interpret than those extracted using the neuro-fuzzy approach. The disadvantage of the TS neuro-fuzzy is that the consequent expressions are expressed in a mathematical form. It seems on the other hand that the granularity of the variables is compromised when data is discretised for the rough set approach. The obvious consequence of this can be that the classification accuracy can be affected. The results of this study thus suggest that the TS model can be used as a classification model for HIV analysis and offers more insight into HIV modeling than the artificial neural networks black boxes. The results also show that rough sets offer more plausible and understandable rules, which can be used to understand the effect of demographic properties on HIV risk.

Chapter 6

Automatic Relevance Determination

A very important factor in HIV classification using demographic characteristics for machine intelligence is the availability of reliable data. The data obtained from statistical and health monitoring structures are very diverse and sometimes irrelevant. Neural networks have been applied in diverse fields and are a powerful tool for complex mappings and modeling of non-linear relationships. In this work, Bayesian framework for artificial neural networks (ANN), known as automatic relevance determination (ARD) method is created to obtain the relative relevance of a large data set of variables from the antenatal clinic data. The viability of this technique is analyzed by selecting optimum input parameters for the neural network model. Zhang et al. (2006) comment on the fact that the performance of a neural network can be improved by reducing the number of input variables correctly, but sometimes even at the cost of losing

some useful input information. The problem of irrelevant input parameters can be resolved by identifying the input parameters that are not important to the performance of the networks, and then removing the less important input parameters from the input data set as in automatic relevance determination (ARD). ARD will thus be used to reduce the input space, and also to know how important each variable is on driving HIV risk (by depicting the relevance of each parameter on HIV risk).

6.1 Introduction

The study of the influence of demographic and social characteristics of HIV by using traditional statistical methods is complicated and time-consuming. In addition, for analytic models to be developed, a priori information about the structure of the mathematical relationships between the input and output variable is needed. This relationship for the HIV scenario appears to be non-linear and discrete, and difficult to handle with standard statistical techniques. If new input variables are to be added to the model, the use of conventional multiple linear regressions will become inappropriate, whereas if non-linear regression is to be used, an explicit function should be provided in advance. Moreover, the latter procedures are static in the sense that the nature of the model cannot be changed. Similarly, computer regression programs cannot learn or become smarter.

An alternative way to avoid the above problems is to employ artificial neural networks (ANN). ANN is an inter-connected structure of processing elements, offering an effective alternative to more traditional statistical techniques in many scientific fields. Since neural networks are highly non-linear and require no prior assumptions concerning the data relationships, they have become a useful tool to tackle medical (HIV) modeling. A drawback of ANN is the usual need to divide the data set into three subsets (training, testing and validation sets) which may become a problem if only few data are available. The Bayesian method of automatic relevance determination (ARD) (Hajmeer and Basheer 2003; Neal 1996) for multilayer perceptron networks (MLP) provides the relative importance of different inputs to the ANN and avoids the need to use separate testing and validation data, thanks to the inclusion of regularization coefficients inside the ANN structure. The methodology presented here aims at using demographic and social factors, to predict and classify the HIV status of an individual using multi-layer perceptron neural network. This methodology first determines the relevant input parameters from the pool of parameters available and then uses these parameters to create a model for HIV classification of an individual with the demographic inputs.

For learning to be successful, information that is known a priori to be irrelevant must be discarded, as otherwise we will be deceived by the chance associations that we are likely to encounter if we search for relationships with a huge number of input variables. The question that arises is; Can the data be used to determine the degree to which each input is relevant, and thereby avoid both

the cost of not using inputs that are useful, and the cost of being misled by chance associations with inputs that have little relevance?

One approach is to select a subset of input variables using some criterion that balances data fit and model complexity. When the number of inputs is large, considering all possible subsets is infeasible, but variables can be added one at a time (forward selection) or removing them one at a time (backward elimination). This approach has long been used in the context of linear regression. Bonnlander (1996) reviews variable selection methods for neural networks, and develops one based on mutual information.

Variable selection seems to be a dubious procedure, since if we think an input *might* be relevant, we will usually also think that it is probably at least a *little bit* relevant. Situations where we think all variables are either highly relevant or not relevant at all seem uncommon (though not, of course, impossible). Accordingly, we should usually not seek to eliminate some variables, but rather to adjust the degree to which each variable is considered relevant. We may hope that a method of this sort for determining the relevance of inputs will be able to avoid paying undue attention to the less relevant inputs, while still making use of the small amount of extra information that they provide.

This is the philosophy behind the Bayesian method of Automatic Relevance Determination (ARD) (Mackay 1998; Neal 1996) for multilayer perceptron networks. In this method a hyperparameter is associated with each input, which controls the size of the weights associated with connections out of that

input, as shown in Fig 6.1 (Lopez et al. 2005).

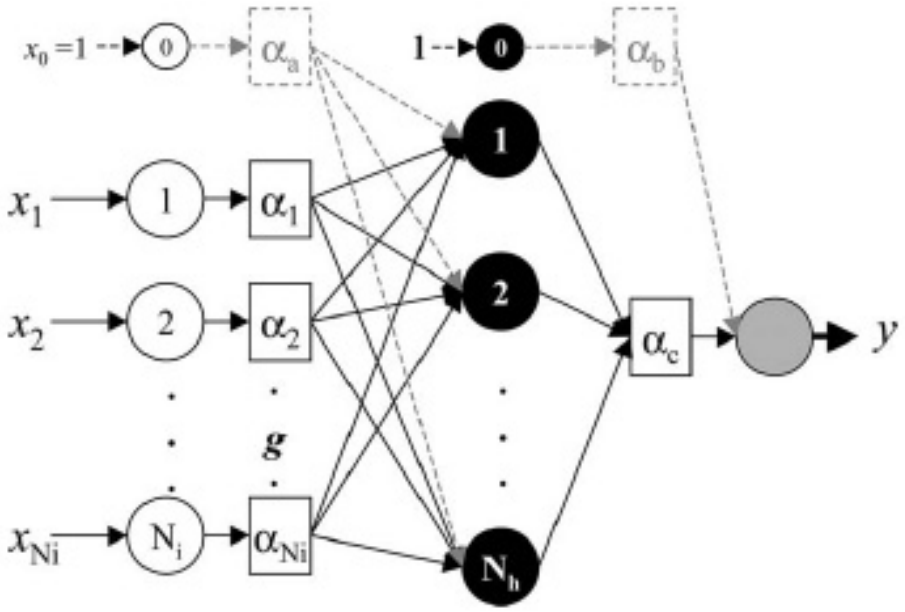

Figure 6.1: Graphical representation of the multilayer perceptron with automatic relevance determination. The hyperparameters $\{\alpha_1, ..., \alpha_{N_i}\}$ control the weights connecting each input to the hidden layer

If the hyperparameter for an input is small, weights for that input will likely be small, and the input will therefore have only a small effect on the network's predictions. These relevance hyperparameters are set in a Bayesian inference fashion, according to the resulting probability of the observed data. Bayesian inference can be defined as follows (Neal 1996):

Given a prior distribution $\pi(\theta)$, a conditional distribution $p(x|\theta)$ and data x, the posterior distribution is computed as follows (Neal 1996):

$$p(\theta|x) = \frac{p(\theta, x)}{p(x)} = \frac{\pi(\theta)p(x|\theta)}{\int \pi(\theta)p(x|\theta)d\theta} \tag{6.1}$$

The posterior distribution, $p(\theta|x)$ is then used to make inferences. For classification, a maximum *a posteriori* (MAP) estimate is used defined as (Neal 1996):

$$\hat{\theta} = argmaxp(\theta|x) \tag{6.2}$$

For estimation, the expected quadratic error is minimized as follows (Neal 1996):

$$\hat{\theta} = E(\theta|x) = \int p(\theta|x)d\theta \tag{6.3}$$

This chapter focusses on a classification problem, thus uses eqn. 6.2.

More on ARD is presented in the next section.

The purpose of this chapter is firstly to apply an automatic relevance determination method for neural network, using Bayesian inference, to determine the relevance of demographic properties for HIV classification; secondly, to propose a good classifier of HIV/AIDS based on relevant demographic data; and thirdly, to compare a classifier using all the input parameters with a classifier which uses only the relevant parameters.

6.2 Background

This section presents a background on the methodologies that have been implemented for relevance determination such as:

- One-Way Analysis of Variance (ANOVA)(MATLAB 2004; Lantz 2007)

and

- Automatic Relevance Determination (ARD) (Neal 1996; Mackay 1998)

6.2.1 One-Way Analysis of Variance

The purpose of one-way analysis of variance (ANOVA) is to find out whether data from several groups have a common mean. That is, to determine whether the groups are actually different in the measured characteristic. ANOVA is used for hypothesis testing in simple regression, multiple regression and comparison of means. There exists a variation when the data values are not identical, which can be as a result of the model or the factor. This variation is the sum-of-squares of the deviations of the actual values from the mean of those values. The variation or sum of squares is abbreviated as SS. The ANOVA method also makes use of the degree of freedom (df), which are the number of values that are free to vary once certain parameters have been established. The sample variance (MS) is obtained by dividing the variations by the degree of freedom, that is (Lantz 2007):

$$MS = \frac{SS}{df} \tag{6.4}$$

The ANOVA table is composed of rows, each row represents one source of variation. For each source of variation

- The variation is in the SS column

- The degrees of freedom is in the df column
- The variance is in the MS column
- The MS value is found by dividing the SS by the df

The table is structured as shown in Table 6.1. One-way ANOVA is a simple

Table 6.1: A Basic Analysis of Variance Table

Source	SS	df	MS	F	p
Between	Data	Data	= SS(B)/df(B)	= MS(B)/MS(W)	
Within	Data	Data	= SS(W)/df(W)		
Total	= SS(B)+SS(W)	= df(B)+df(W)	= SS(T)/df(T)		

special case of the linear model. The one-way ANOVA form of the model is (MATLAB 2004).

$$y_{ij} = \alpha_{ij} + \varepsilon_{ij} \tag{6.5}$$

where:

- y_{ij} is a matrix of observations in which each column represents a different group.
- α_{ij} is a matrix whose columns are the group means. (The "dot j" notation means that α applies to rows of the j-th column. That is, the α_{ij} is the same for all i values.)

- ε_{ij} is a matrix of random disturbances. The model is based on the fact that the columns of y are a constant plus a random disturbance. The aim is thus to know if the constants are all the same.

The standard ANOVA table as shown in Table 6.1, has columns for the sums of squares, degrees of freedom, mean squares (SS/df), F statistic, and p-value. The table obtained for this study will be commented on in Section 6.4. The one-way ANOVA table uses the grand mean defined as (Lantz 2007):

$$\bar{\bar{x}} = \frac{\sum_{i=1}^{k} n_i \bar{x}_i}{\sum_{i=1}^{k} n_i} \tag{6.6}$$

where $\bar{\bar{x}}$ is the weighted (grand) mean of the individual sample means, $\bar{x}_i$ is the mean of factor i and n_k is the kth factors number of parameters. The grand mean is the average of all the values when the factor is ignored. The between group variation, SS(B) in Table 6.1 is the variation between each sample mean and the grand mean, defined by (Lantz 2007):

$$SS(B) = \sum_{i=1}^{k} n_i \left(\bar{x}_i - \bar{\bar{x}}\right)^2 \tag{6.7}$$

The within group variation, SS(W) is the weighted total of the individual variations. The weighting is done with the degrees of freedom (df). The df for each sample is one less than the sample size for that sample. SS(W) is defined as (Lantz 2007):

$$SS(W) = \sum_{i=1}^{k} df_i s_i^2 \tag{6.8}$$

The between group df, df(B) is one less than the number of groups. The data

in this study has 9 groups, so the df(B) is 8. The within group df, df(W) is the sum of the individual df's of each group, which is the number of samples minus one. The variance MS(B) and MS(W) is calculated as in equation 6.4. The MS(Within) is also known as the pooled estimate of the variance since it is a weighted average of the individual variances, sometimes abbreviated as s_p^2. The p value in Table 6.1 is the area to the right of the test statistic (F). F is defined by (Lantz 2007):

$$F = \frac{MS(B)}{MS(W)} \tag{6.9}$$

If the p-value is less than a significance level, then the null hypothesis, which states that the means of all the groups are the same, can be rejected. The significance of the null hypothesis being rejected is that at least one of the groups has a different mean. The groups with different means can be obtained using box plots.

6.2.2 Automatic Relevance Determination

Feature selection methods are often classified into wrappers and filters, the difference being in whether or not the method uses the output of the classifier in order to select the features. Wrapper methods usually work by evaluating the classifier on subsets of the feature space, using some sort of greedy algorithm to organize the search of the large number of possible feature combinations. Filter methods, on the other hand, generally use unsupervised methods to select features (Rencher 1995). In the ARD, the contribution of each input

feature to the output function is divided by a separate length scale (Gold et al. 2005). The larger the length scale the smaller the contribution, which that feature will make to the output function. For this reason the length scales produced by the hyperparameter tuning algorithm can be used for feature selection simply by eliminating those features with the largest length scales. While feature selection via ARD is technically a wrapper approach because the output of the classifier on the training set is used in the hyperparameter tuning algorithm, it is distinct from traditional wrapper approaches because it avoids searching the space of feature combinations and instead proceeds directly to an appropriate feature set using principles designed to improve generalization performance. Feature detection using ARD was originally proposed for back-propagation and radial basis function (RBF) network (Mackay 1998).

This method of detecting the variable influence using neural networks is an extension of Bayesian regularization, which is based on a probabilistic interpretation of the network training procedure. Neural Networks have an error function associated with their predictability. This may be expressed mathematically as follows:

$$y = f(w; x) + \varepsilon \tag{6.10}$$

Where y is the actual output desired, f is the output predicted by the network, ε is the error, w are the weights and x is a vector of inputs. There is uncertainty in the training of the networks (Bishop 1995). This uncertainty in the training of the networks can be associated with the fact that the assignment of the weights is done by randomization. Wherever used from now on, p(.)

would denote the statistical probability. The network weights are considered as random variables. The problem of identifying the weights and biases in neural networks may be posed in the Bayesian framework as (Bishop 1995):

$$p(w|D) = \frac{p(D|w)p(w)}{p(D)} \tag{6.11}$$

(2) where $p(w)$ is the probability distribution function of the weight-space in the absence of any data, also known as the prior distribution function and D $= (y_1, , y_N)$ is a matrix containing the HIV data. The quantity $p(w|D)$ is the posterior probability distribution function after the antenatal data have been seen, $p(D|w)$ is the likelihood function and $p(D)$ is the normalization function also known as the *evidence* (Rank 2003).

Regularization techniques generally alter the objective function to be minimized during the network training, by adding penalty terms (regularizers) to avoid the overfitting phenomenon and thereby develop models with better generalization properties. The objective function to be minimized is as follows (Liitiainen 2006):

$$I = \gamma E_1 + \xi E_2 = \frac{\gamma}{2}\|w\|^2 + \frac{\xi}{2}\sum_{i=1}^{N} e_i^2 \tag{6.12}$$

where $e_i = (y_i - f(x_i, w))$, γ and ξ are hyperparameters to be optimized, E_1 and E_2 are functions defined on the right hand side of the equation, and $f(x_i, w)$ is defined as in chapter 1 eqn. 1.1 as (Bishop 1995):

$$y_k = f_{outer}\left(\sum_{j=1}^{M} w_{kj}^{(2)} f_{inner}\left(\sum_{i=1}^{d} w_{ij}^{(1)} x_i + w_{j0}^{(1)}\right) + w_{k0}^{(2)}\right) \tag{6.13}$$

Here, $w_{ji}^{(1)}$ and $w_{kj}^{(2)}$ indicate weights in the first and second layer, respectively, going from input i to hidden unit j, M is the number of hidden units, d is the

number of output units while $w_{j0}^{(1)}$ and $w_{k0}^{(2)}$ indicate the biases for the hidden unit j and the output unit k. In this chapter, the function $f_{outer}(\bullet)$ is linear while f_{inner} is a hyperbolic tangent function.

Note that two hyperparameters are used to get a Bayesian interpretation. ARD is based on Bayesian inference on three levels. In what follows, H denotes the model and D is the data. We assume no prior knowledge of the problem which means that flat priors are used whenever necessary.

First Level of Inference

Assume that the sample (x_i, y_i) is independent. Recall the cost (Liitiainen 2006):

$$I = \gamma E_1 + \xi E_2 \tag{6.14}$$

In the first level the hyperparameters γ and ξ are assumed to be fixed. We assume the prior $p(w) \sim exp(-\gamma \|w\|^2)$. For the first observation, we assume (Liitiainen 2006):

$$p(y_i|x_i, w, b, \xi, H) \sim exp(-\frac{\xi}{2} e_i^2) \tag{6.15}$$

This is a model with a Gaussian prior and a Gaussian noise model. With the assumptions, we get (Liitiainen 2006):

$$p(w, b|D, \gamma, \xi, H) \sim exp(-I(D, \gamma, \xi, w, b)) \tag{6.16}$$

It follows that given the hyperparameters, finding the maximum likelihood for $p(w, b|D, \gamma, \xi, H)$ is equivalent to minimizing the cost function I.

Second Level of Inference

In the second level $p(\xi, \gamma|D, H)$ is examined. $p(\xi, \gamma|D, H)$ can be written as (Liitiainen 2006):

$$p(\xi, \gamma|D, H) = \int p(D|w, b, H)p(w, b|\xi, \gamma, H)p(\xi, \gamma|H)dwdb \tag{6.17}$$

A non-informative prior for the hyperparameters is assumed (Liitiainen 2006). This can be solved in closed form. Thus no approximation is needed on the second level. Using the previously derived formula in eqn. 6.16, we get (Liitiainen 2006):

$$p(\xi, \gamma|D, H) \sim \frac{\gamma^{n_f/2}\xi^{N/2}}{|H|^{-1/2}} exp(-I(w, b)) \tag{6.18}$$

Here H is the Hessian of the cost function and n_f is the dimension of the space in which f maps the inputs. Typically nf $\gg 1$ and the Hessian H is not available as such. However, it turns out that this is not a problem. By using the cross-entropy error between the network prediction and the output of the training data defined in eqn. 6.19, it is possible to derive a maximum likelihood cost function for the hyperparameters (Bishop 1995).

$$E = -\beta \sum_{n=1}^{N} \sum_{k=1}^{K} \{t_{nk} ln(y_{nk}) + (1 - t_{nk}) ln(1 - y_{nk})\} + \frac{\bar{\alpha}}{2} \sum_{j=1}^{W} w_j^2 \tag{6.19}$$

The cross-entropy function is chosen because it has been found to be more suited to classification problems than the sum-of-square of error cost function (Bishop 1995). In eqn. 6.19, n is the index for the training pattern, β is the data contribution to the error and k is the index for the output units. The second term in eqn 6.19 is the regularisation parameter and it penalises weights

of large magnitudes (Bishop 1995). This regularisation parameter is called the weight decay vector and its coefficient, $\bar{\alpha}$, determines the relative contribution of the regularisation term on the training error. This regularisation parameter ensures that the mapping function is smooth. y_k represents the equation in eqn. 6.13.

Third Level of Inference

Recall that H denoted the model structure (including model parameters, selected inputs). In the third level we write (assuming non-informative priors) (Liitiainen 2006):

$$p(D|H) = \int p(D|\gamma, \xi, H)p(\xi, \gamma|H)d\xi d\gamma \sim p(D|\gamma_{MAP}, \xi_{MAP}, H)D_\gamma D_\xi \quad (6.20)$$

The terms D_γ and D_ξ are the second derivatives of the second level cost function at the optimum, MAP stands for maximum *a posteriori*.

Input Selection

Now that we can evaluate the evidence $p(D|H)$ of models, input selection is easy. A combination of inputs is evaluated by doing the three levels of inference to calculate model parameters and hyperparameters. Scaling of input variables is implemented in the same way.

ARD methods have the advantage that they are based on a sound probabilistic theory instead of rules of thumb for determining the hyperparameters of the

regularizers (Papadokonstantakis et al. 2006). In the ARD framework for MLP, it assigns one hyperparameter for each n input variable (input neuron), which can control the magnitude of weight fanning out from the input neuron (Wang and Lu 2006). ARD background can be found in Lopez et al. (2005) and Papadokonstantakis et al. (2006).

6.3 Methodology

Background reading showed that models for HIV classification using neural networks have been proposed as presented in Chapter 2 and worked well using demographic properties. However, it was realized that no automatic relevance determination models have been applied to HIV modeling. Detecting the relevant parameters avoids redundancy, and it is believed that this will play a role in obtaining a better classifier for the classification models. In this study, a method using multi-layer perceptron neural networks and scaled conjugate gradients is investigated. The method uses the demographic properties of individuals and determines which of these properties is important for the prediction of the individual's HIV status. The ARD methodology has been presented in Section 6.2.2. The results from this model will be assessed with results obtained from the one-way analysis of variance (ANOVA).

Upon obtaining the relevant parameters using ARD Bayesian regularisation approach, a neural network model is then created using the relevant parameters

and trained to predict the risk of HIV from demographic properties. The classification accuracy of this new network model is compared with the already created model presented in Chapter 2 Section 2.3.2, with the full data set. The area under the ROC curves is also compared to evaluate which of the models is a better classifier.

6.4 Testing the Procedure

From the data presented in Chapter 1, 9 parameters were selected. The multilayer perceptron network with 9 inputs representing the demographic parameters, and 1 output representing the HIV status of individuals, was constructed and several numbers of hidden units were used and implemented in Matlab (MATLAB 2004).

The first experiment used the ARD Netlab implementation (Nabney 2003), which uses scaled conjugate gradient (Bishop 1995; Haykin 1994) to obtain the relevant parameters from the data set required to predict the risk of HIV for an individual. The weights were penalised with a regularisation parameter (α) of 0.01 and a co-efficient of data error (β), defined in eqn. 6.19, of 50.0. This method yielded Table 6.2 as the weightings associated to the different inputs. Table 6.2 shows that the parameters Age, Gravidity and Education are of high relative significance, meanwhile the parameters Province, Region and rapid plasma reatin(RPR) are of medium relative relevance. The parameters

Table 6.2: Automatic Relevance With Multi-layer Perceptron and Scaled Conjugate Gradient

Variable	Weights(Inverse Variance)	Inverse Weights	Relative Weights
Age	1.19446	0.8372	79
Gravidity	5.09001	0.1965	19
Parity	7.66687	0.013	1.2
Province	20.45520	0.0489	5
Race	94.59406	0.0106	1
Region	10.56756	0.0946	9
RPR	8.98063	0.1114	11
WTREV	130227180.94823	0.00	0
Education	3.05223	0.3276	31

Parity, Race and Regional weighting parameter (WTREV) are of low relative relevance and can be ignored.

The second experiment investigated the use of the relevant parameters obtained from the previous experiment for HIV modeling, i.e. predicting the HIV status of individuals from relevant demographic characteristics. The optimal number of hidden nodes for the neural network was 9, hence the structure was 6 - 9 - 1. The confusion matrix shown in Table 6.3 is obtained for the classification.

This model had an accuracy of 76%. The ROC curve obtained for this network

Table 6.3: Classifier Confusion Matrix for ARD Classification

Confusion Matrix	Predicted Positive	Predicted Negative
Actual Positive	654	339
Actual Negative	138	855

model is shown in Fig. 6.2 and the area under this ROC curve is 0.9, which according to Monash (2006) is an excellent classifier.

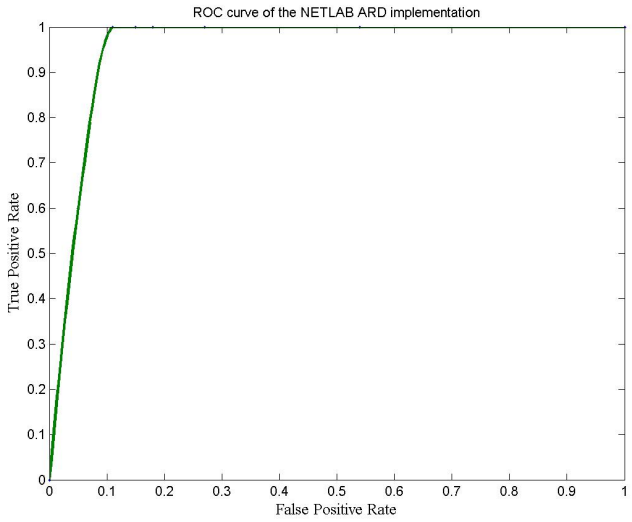

Figure 6.2: ROC Curve for the Netlab ARD network)

The third experiment used the full data set, with all the 9 inputs present, for HIV modeling, i.e. predicting the HIV status of individuals from all the demographic characteristics. The optimal number of hidden nodes for the neural network was 77, hence the structure was 9 - 77 - 1. This network gave an accuracy of 84% on the test data sets. The confusion matrix obtained for the above network is as shown in Table 6.4. The ROC curve obtained for this classification is shown in Fig. 6.3 and the area under this ROC curve obtained

Table 6.4: Classifier Confusion Matrix for All Inputs Classification

Confusion Matrix	Predicted Positive	Predicted Negative
Actual Positive	738	255
Actual Negative	85	908

was 0.8, which according to Monash (2006) is a very good classifier.

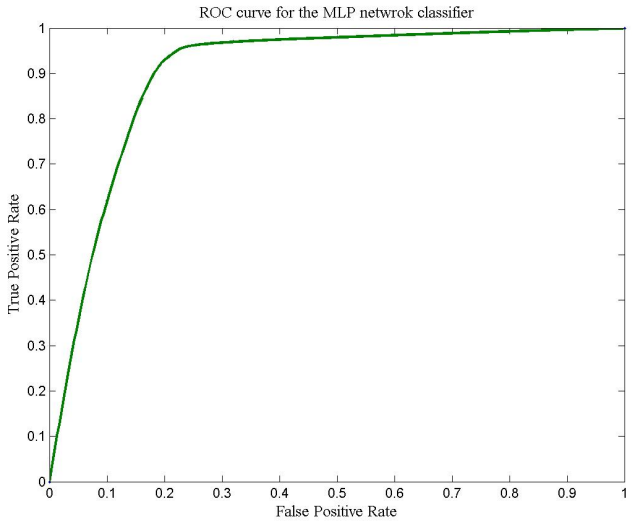

Figure 6.3: ROC curve for the MLP network classifier

The fourth experiment investigated the use of one-way ANOVA to determine the relevant parameters. The standard ANOVA table has columns for the sums of squares, degrees of freedom, mean squares (SS/df), F statistic, and p-value. This was implemented in Matlab using the *anova*1 function (MATLAB 2004). In this study, the p-value is 0. This is less than the significance level of 0.05, thus rejecting the null hypothesis. This is a strong indication that the data set values from the different data points are not the same (different means).

An F statistic of 2562.47 is obtained for the data points. The p-value returned by anova1 depends on assumptions about the random disturbances ε_{ij} in the model equation (MATLAB 2004). Graphical assurance that the means are dissimilar can be obtained by looking at the box plots in Fig. 6.4. Fig. 6.4

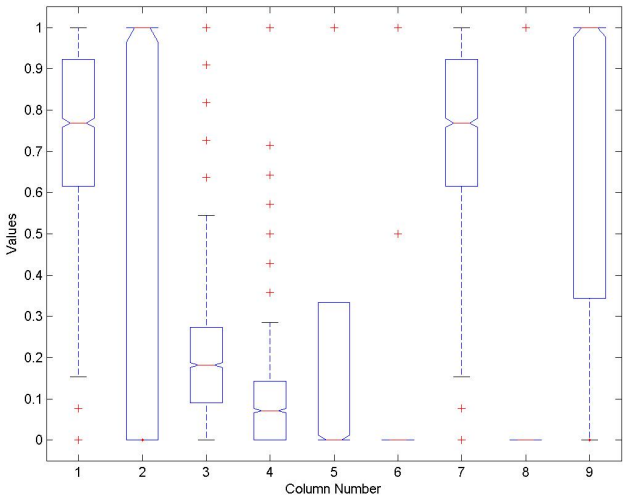

Figure 6.4: ANOVA Mean Values Box Plot

shows that the means are dissimilar for all parameters except for columns six and eight which represent race and rapid plasma reatin (RPR). These two parameters can thus be eliminated. This is inline with the ARD findings as well, which found race parameter as an irrelevant parameter.

The ARD method thus suggests that Parity, Race and regional weighting parameter (WTREV) can be ignored when dealing with the analysis of HIV from the demographic properties. The ARD method also suggests that Age, Gravidity and Education are of high relative significance meanwhile Province, Region and Rapid Plasma Reatin (RPR) are of medium relative significance. Using

the one-way ANOVA, which suggests that race is irrelevant, a conclusion can be drawn that the risk of HIV is not dependent on the race of an individual.

The results obtained are summarised in Table 6.5.

Table 6.5: Summary of Results

Model	Inputs Not Relevant	Overall Finding
MLP ARD	Parity, Race, WTREV	Race can be ignored.
One-Way Anova	Race, RPR	Race can be ignored.

6.5 Discussion and Conclusion

A method based on multi-layer perceptron neural networks and scaled conjugate gradients is investigated for automatic relevance determination of the demographic inputs for HIV modeling in this chapter. This procedure was tested on an HIV data set obtained from South African antenatal seroprevalence survey of 2001. A data set comprising of 9 inputs was used and it was found that only 6 inputs were relevant for the classification and modeling, thus the other 3 inputs were discarded. The one-way ANOVA method was used to assess the functionality of the ARD for selection of the relevant parameters. A network model was then created with these 6 inputs and yielded a classification accuracy of 76%. The area under the ROC curve was computed as 0.9, thus an excellent classification. The full dataset obtained a classification accuracy

of 84% and the area under the curve was computed as 0.80, thus a very good classification. The area under the ROC curve (AUC) is a good measure of classifier performance. The ARD data set model AUC is 0.9, meanwhile the AUC of the full data input parameters model is 0.8. This thus shows that identifying the relevant parameters for modeling is important before the modeling process begins. The area under the ROC curve for the ARD data was higher than that of the full data inputs by 0.10, thus a better classifier.

Chapter 7

Conclusion and Further Recommendations

7.1 Conclusive Remarks

This chapter concludes the research carried out by presenting an overview of all the results obtained. The objectives of this research as stated in Chapter 1 Section 1.5 were to:

1. Create a model based on computational intelligence to model HIV from demographic data.

2. Create a model to estimate missing data in the HIV database and to understand the impact of such missing data

3. Create a computational model to understand how the demographic properties influence the HIV susceptibility of individuals,

4. Create a model based on neuro-fuzzy networks to model HIV from demographic data and create a model based on rough sets for rules extraction,

5. Create a model, which depicts the relevance and importance of the demographic parameters with respect to HIV modelling, and this reduces the input space dimensionality.

The research was carried out using Computational intelligence, which as shown in Chapter 1 Section 1.1 has been applied in medical informatics. A model was created in Chapter 2 to classify the HIV status of individuals based on a demographic dataset presented in Chapter 1 Section 1.2. This model used autoencoder neural networks together with genetic algorithms and yielded a classification accuracy of 92% compared to an accuracy of 84% obtained from a conventional feedforward neural network model. The Area under the ROC curve obtained for this model was 0.86, which showed that this was a very good classifier. The first objective of the research was thus achieved by creating this model. This model possessed novelty in that this research showed that autoencoder networks-based-on-genetic algorithms can be applied in HIV modelling. The findings are summarized in Table 2.3.

When analysing the demographic data set in Chapter 1 Section 1.2, it was realized that there were missing entries in the database. This could have been due to the individuals submitting incomplete forms. The model created for

HIV classification using autoencoder networks could, however, only work with complete data entries. It was thus imperative that a model to estimate the missing entries be created. The most frequently missing parameter was the educational level of the individuals, which accounted for 88% of the missing entries. A methodology was thus created using autoencoder networks and ant colony optimization (ACO), which research review showed had not yet been applied for missing data approximation. This was thus the novelty of this chapter. The methodology was able to estimate missing data to an accuracy of 80%. The impact of missing data on the overall predictability was analyzed. The autoencoder network had an uncertainty effect of 11% on the predictability, with a classification accuracy of 81% in the presence of missing data compared to a classification accuracy of 92% for a case with no missing data. The feedforward neural network had an uncertainty effect of 2% on the predictability, with a classification accuracy of 82% in the presence of missing data compared to a classification accuracy of 84% for a case with no missing data. The feedforward neural network was thus more noise resistant than the autoencoder network. The results were found to be coherent with an error analysis performed on both networks in the presence of uncertainty in the data. This analysis is performed in Chapter 3 Section 3.4. The second research objective was thus met. The results are summarized in Tables 3.1 and 3.2.

Chapter 4 focussed on understanding the effect of demographic influences on HIV susceptibility. In this chapter a methodology was proposed to understand

how modifying certain modifiable demographic parameters of individuals affects their HIV status. An adaptive control model was implemented using an inverse neural network and an autoencoder-based-on-genetic algorithms model and the two models were compared. The two modifiable parameters in this study were the educational level and the gravidity, which were defined in Chapter 1 Section 1.2. An accuracy of 77% was obtained by the inverse neural network model and an accuracy of 77% was obtained by the genetic algorithm model, for the educational level prediction. An accuracy of 82% was obtained by the inverse neural network model as opposed to 92% for the genetic algorithm model, for the gravidity prediction. A model can thus be developed using inverse neural networks, to effectively assess the demographic parameters required by individuals, in order to control the susceptibility to HIV of such individuals. A model can also be developed using autoencoder neural networks and genetic algorithms, to effectively assess the demographic parameters required by individuals. The results of this study show that gravidity is a highly controllable parameter due to its high predictability accuracy from the other demographic properties. The results of this study also show that the educational level can be controlled even though not as effectively as gravidity, due to lower predictability accuracy. This chapter thus successfully meets the third objective. The results are also summarized in Table 4.1.

Literature review also showed that neuro-fuzzy models had not been applied to HIV modelling. It was believed that the neuro-fuzzy model will offer more

significant results when applied to HIV classification compared to neural networks, which have a black box-like nature. A neuro-fuzzy model was thus implemented in Chapter 5. This model yielded a classification accuracy of 86%, compared to an accuracy of 84% for the feedforward neural network and an accuracy of 92% for the autoencoder network. In the presence of missing data, the neuro-fuzzy model yielded an accuracy of 83%. The neuro-fuzzy model was thus more noise resistant than the autoencoder network model but comparatively the same as the feedforward model. The neuro-fuzzy model, however, had an added advantage of fuzzy rules extraction which was presented in Chapter 5 Section 5.5.1. A rough set approximation model was also investigated in this chapter and the results showed that this model yields simpler and more understandable rules compared to the neuro-fuzzy model. A background on rough set was first presented and the rough set formulation was also presented. The fourth objective of the research was thus met in this chapter. The results of this chapter are summarized in Table 5.6.

The last objective of this research was to investigate a methodology to obtain the relevant parameters for HIV modelling from the available demographic parameters in the antenatal database. This was to ensure that redundant parameters are not used in the model as this will lead to greater computation times as well as greater cost in collecting all the parameters. A method based on multi-layer perceptron neural networks and scaled conjugate gradient method was investigated in Chapter 6. This method was able to prune the data set size from 9 parameters to 6 parameters. The method showed that

the parameters Age, Gravidity and Education are of high relative significance, meanwhile the parameters Province, Region and Rapid plasma reatin (RPR) are of medium relative relevance. The method also showed that the parameters Parity, Race and Regional weighting parameter (WTREV) are of low relative relevance and can be ignored. The area under the ROC curve for the classifier proposed by this model was 0.9 compared to 0.8 obtained from the classifier with all the input parameters present. The pruned data set classifier was thus a better classifier than the latter. The ARD method thus reduced the input space, and depicted the relevance of the various parameters. The result is summarized in Table 6.5.

The summary of all the findings in this research are presented in Table 7.1.

Note: E represents Education, G represents Gravidity, ACO represents Ant Colony Optimization, NN represents Neural networks, AE represents Autoencoder networks, ARD represents Automatic Relevance Determination, MLP represents Multilayer Perceptron and GA represents genetic algorithms in Table 7.1.

All the objectives of this research were met. Models were proposed and tested on demographic data obtained from the South African antenatal Clinics and these model yielded good accuracies. Each model presented herein also has novelty in HIV modelling since extensive research reviews showed such models had not been previously implemented.

Table 7.1: Summary of Results

Model	Accuracy	Characteristics
Feedforward Prediction	84	Less sensitive to noise. Better for missing input models.
AE Prediction	92	Best predictor model. Worst sensitivity to noise.
AE + ACO Missing Data	80	Slightly lower accuracy but much better computation times.
AE + GA Missing Data	82	Higher accuracy but computationally expensive.
Inverse NN Demographic	77(E), 82 (G)	Lower Gravidity accuracy but better computation times.
AE + GA Demographic	77(E), 92(G)	Higher Gravidity accuracy but computationally expensive.
Neurofuzzy	86	Mathematically inclined rules, which need background to extract.Also not too sensitive to noise.
Rough Set	–	Very simplistic linguistic rules,which can be readily understood.
MLP ARD	76	Together with One-way ANOVA, found that Race can be ignored.

7.2 Further Work and Reccomendations

As further work, fuzzy ARTMAP principles can be applied for HIV modelling. The question being addressed here would be: given an individual with demographic characteristics D and a set of known families F, we must classify the individual's demographic characteristics into one of the families F. The aim thus is to group individuals with the same demographic properties into the same family, since they have similar structural and functional properties. Thus if an unknown individual's demographic property, D_i is found to belong to some family F_i, we can infer the structure and function of F_i. The pattern recognition process begins with the data acquisition stage. A subset of features is then selected, using genetic algorithms, and is then classified using a wide range of classification techniques. The fuzzy ARTMAP classifier should be introduced because it is believed this classifier may demonstrate significant benefits over other established classifiers in current literature. An online learning classifier should also be introduced, as it is believed this may also significantly improve on the classification performance of current classifiers. To the best of our knowledge, neither the fuzzy ARTMAP, nor the online learning, have been considered in the context of HIV classification and modeling.

In the literature review, there is no method proposed thus far that investigates the use of fuzzy ARTMAP or online learning for HIV modeling. Research into the applicability of fuzzy ARTMAPS in HIV modelling may thus propose a new method, which is based on fuzzy ARTMAP models combined with online

learning to classify the HIV status of an individual based on demographic properties.

As a recommendation, the use of multi-agents for HIV modelling may also offer a more significant view to understand the spread of the disease as well as to understand how inter-relationships contributes to the dynamics of the disease. Agent modelling is being used in stock price modelling to understand the interaction between stocks as well as the effects these interactions have on the overall predictability. Application of such agent modelling techniques could offer insights into HIV classification of individuals based on other individuals, which other current models do not possess.

References

Abdella, M. and Marwala, T.: 2005, The use of genetic algorithms and neural networks to approximate missing data in database, *Computing and Informatics* **24**, 577–589.

AIDSCAP: 1998, A tool for Estimating Intervention Effects on the Reduction of HIV Transmission, v1.0. `http://www.iaen.org/models/avert/avert10.zip`.

Alkan, A., Koklukaya, E. and Subasi, A.: 2005, Automatic seizure detection in EGG using logistic regression and artificial neural network, *Journal of Neuroscience Methods* **148**(2), 167–176.

Atalla, M. and Inman, D.: 1998, On model updating using neural networks, *Mechanical Systems and Signal Processing* **12**(1), 135–161.

Bersini, H. and Bontempi, G.: 1997, Now comes the time to defuzzify neuro-fuzzy models, *Fuzzy Sets and Systems* **90**, 161–169.

Bishop, C.: 1995, *Neural Networks for Pattern Recognition*, Oxford University Press, Oxford, UK.

Bonnlander, B.: 1996, *Nonparametric Selection of Input Variables for Connectionist Learning: PhD Thesis*, University of Colorado, Department of Computer Science. `http://www.cs.colorado.edu/~brianb/`.

Bullnheimer, B., Hartl, R. and Strauss, G.: 1998, Applying the ant system for the vehicle routing problem, *Meta-heuristics Advances and Trends in Local Search Paradigms for Optimizations* pp. 109–120.

Chan, K., Lee, T.-W., Sample, P., Goldbaum, M., Weinreb, R. and Sejnowski, T.: 2002, Comparison of machine learning and traditional classifiers in glaucoma diagnosis, *IEEE Transactions on Biomedical Engineering* **49**(9), 963–974.

Connor, J., Martin, R. and Atlas, L.: 1994, Recurrent neural networks and robust time series prediction, *Neural Computation* **6**(6), 1154–1172.

Costa, D. and Hentz, A.: 1997, Ants can color graphs, *Journal of the Operational Research Society* **48**, 295–305.

Crossingham, B. and Marwala, T.: 2007, *Using Genetic Algorithm to Optimise Rough Set Partition Sizes for HIV Data Analysis: Studies in Computational Intelligence*, Vol. 78, Springer-Verlag.

Davis, L.: 1991, *Handbook of Genetic Algorithms*, Van Nostrand, New York, USA.

Dorigo, M. and Gambardella, L.: 1997, Ant colonies for the traveling salesman problem, *BioSystems* **43**, 73–81.

Fawcett, T.: 2003, Roc graphs: Notes and practical considerations for data mining researchers, *Technical Report HPL-2003-4*, Intelligent Enterprise Technologies Laboratory, HP Laboratories.

Fee, E. and Krieger, N.: 1993, Understanding AIDS: historical interpretations and limits of biomedical individualism, *American Journal of Public Health* **83**, 1477–1488.

Fernandez, M. and Caballero, J.: 2006, Modeling of activity of cyclic urea HIV-1 protease inhibitors using regularized-artificial neural networks, *Journal of Bioorganic and Medicinal Chemistry* **14**, 280–294.

Figueiredo, M. and Gomide, F.: 1999, Design of fuzzy systems usaing neuro-fuzzy networks, *IEEE Transactions on Neural Networks* **10**(4), 815–827.

Flach, P.: 2004, The many faces of roc analysis in machine learning, icml '04 tutorial on roc analysis, *Tutorial: International Conference of Machine Learning*. `http://www.cs.bris.ac.uk/~flach/ICML04tutorial/`.

Frolov, A., Kartashov, A., Goltsev, A. and Folk, R.: 1995, Quality and efficiency of retrieval for Willshaw-like autoassociative networks II Recognition Network, *Computation in Neural Systems* **6**(4), 535–549.

FutureGroup: 2002, AIDS Impact Model for Busines: AIM-B. `http://www.futuresgroup.com/aim/form1.cfm`.

Gabrys, B.: 2002, Neuro-fuzzy approach to processing inputs with missing values in pattern recognition problems, *International Journal of Approximate Reasoning* **30**, 149–179.

Goh, C. and Law, R.: 2003, Incorporating the Rough Sets Theory into Travel Demand Analysis, *Tourism Management* **24**, 511–517.

Gold, C., Holub, A. and Sollich, P.: 2005, Bayesian approach to feature selection and parameter tuning for support vector machine classifiers, *Neural Networks* **18**, 693–701.

Goldberg, D.: 1989, *Genetic Algorithms in Search Optimization and Machine Learning*, Addison-Wesley, Reading-MA, USA.

Grzymala-Busse, J. W.: 1992, *LERS: A system for learning from examples based on rough sets*, Handbook of Applications and Advances of the Rough Sets Theory: Kluwer Academic Publishers.

Hajmeer, M. and Basheer, I.: 2003, Comparison of logistic regression and neural network-based classifiers for bacterial growth, *Food Microbiology* **20**(1), 43–55.

Han, J. and Kamber, M.: 2000, *Data Mining: Concepts and Techniques*, Vol. third edition, Morgan Kaufmann Publishers, San Mateo, CA.

Hand, D., Mannila, H. and Smith, P.: 2001, *Principles of Data Mining*, MIT Press, Cambridge-MA, USA.

Hassoun, M.: 1995, *Fundamentals of Artificial Neural Networks*, MIT Press.

Haupt, R. and Haupt, S.: 2004, *Practical Genetic Algorithms, 2ed*, Wiley and Sons.

Haykin, S.: 1994, *Neural Networks; A Comprehensive Foundation*, Macmillan.

HealthDept: 2005, South African Department of Health HIV Syphilis survey data 2001. `http://www.health.gov.za`.

Hines, J., Robert, E. and Wrest, D.: 1998, Use of Autoassociative Neural Networks for Signal Validation, *Journal of Intelligent and Robotic Systems* **21**(2), 143–154.

Holland, J.: 1975, *Adaptation in Natural and Artificial Systems*, University of Michigan Press, Ann Arbor, USA.

Honghai, F., Guoshun, C., Cheng, Y., Bingru, Y. and Yumei, C.: 2005, *A SVM regression based approach to filling in missing values: Lecture Notes in Computer Science*, Vol. 3683/2005, Springer-Verlag.

Hudson, D. and Cohen, M.: 2000, *Neural Networks and Artificial Intelligence for Biomedical Engineering*, NJ-IEEE Press, New Jersey, USA.

Jagannathan, G. and Wright, R.: 2008, Privacy-preserving imputation of missing data, *Data and Knowledge Engineering* **In Press**, Available online.

Jang, J.-S.: 1993, ANFIS: adaptive network based fuzzy inference systems, *IEEE Transactions on Systems, Man and Cybernetics* **23**(3), 665–685.

Jang, J.-S., Sun, C.-T. and Mizutani, E.: 2002, *Neuro-Fuzzy and Soft Computing*, Prentice Hall of India Private Limited, India.

Junninen, H., Niska, H., Tuppurainen, K., Ruuskanen, J. and Kolehmainen, M.: 2004, Methods for imputation of missing values in air quality data sets, *Atmospheric Environment* **38**, 2895–2907.

Khalil, M., Panu, U. and Lennox, W.: 2001, Groups and neural networks based streamflow data infilling procedures, *Journal of Hydrology* **241**, 153–176.

Knorr, A. and Srivastava, R.: 2005, Evaluation of HIV-1 kinetic models using quantitative discrimination analysis, *BioInformatics* **21**(8), 1668–1677.

Komorowski, J., Pawlak, Z., Polkowski, L. and Skowron, A.: 1999, *A Rough Set Perspective on Data and Knowledge*, The Handbook of Data Mining and Knowledge Discovery, Oxford Univesrity Press.

Kramer, M.: 1991, Nonlinear principal component analysis using autoassociative neural Networks, *AIChE Journal* **37**, 233–234.

Lakshminarayan, K., Harp, S., Samad, T. and Moor, B. D.: 1999, Imputation of missing data in industrial databases, *Applied Intelligence* **11**(3), 259–275.

Lantz, D.: 2007, *Introduction to One-Way Analysis of Variance*, University of Colgate, Department of Mathematics. `http://math.colgate.edu/math102/dlantz/`.

Laumann, E. and Youm, Y.: 1999, Racial/ethnic group differences in the prevalence of sexually transmitted diseases in the United States: a network explanation, *Sex Transm Dis* **26**, 250–261.

Lavrac, N.: 1999, Selected techniques for data mining in medicine, *Artificial Intelligence in Medicine* **16**, 3–23.

Lee, C. and Park, J.: 2001, Assessment of HIV/AIDS-related health performance using an artificial neural network, *Journal of Information and Management* **38**, 231–238.

Leke, B. and Marwala, T.: 2006, Ant Colony Optimisation for Missing Data Estimation, *Proceedings: Pattern Recognition Association of South Africa*, Parys, South Africa, pp. 183–188.

Leke, B., Marwala, T. and Tettey, T.: 2007a, Autoencoder Networks for HIV Classification, *Current Science* **91**(11), 1467–1473.

Leke, B., Marwala, T. and Tettey, T.: 2007b, Using Inverse Neural Networks for HIV Adaptive Control, *International Journal of Computational Intelligence Research* **3**(1), 11–15.

Leke, B., Marwala, T., Tim, T. and Lagazio, M.: 2006a, A comparative study between genetic algorithms and line search algorithm optimization for HIV predictions, *Proceedings of the 5th WSEAS International Conference on Artificial Intelligence, Knowledge Engineering and Data Bases*, Madrid, Spain, pp. 231–236.

Leke, B., Marwala, T., Tim, T. and Lagazio, M.: 2006b, Prediction of HIV status from demographic data using neural networks, *Proceedings of the 2006 IEEE International Conference of Systems, Man and Cybernetics*, Taipei, Taiwan, pp. 2339–2344.

Leke, B., Tim, T., Marwala, T. and Lagazio, M.: 2006, Using genetic algorithms versus line search algorithm optimization for HIV predictions, *WSEAS Transactions on Information Science and Applications* **4**(3), 684–690.

Liitiainen, E.: 2006, *Automatic Relevance Determination*, Time Series Prediction Group, Helsinki University of Technology, Finland. `http://www.cc.hut.fi/eliitiai/`.

Lint, J. V., Hoogendoorn, S. and Zuylen, H. V.: 2005, Accurate freeway travel time prediction with state-space neural networks under missing data, *Transportation Research Part C: Emerging Technologies* **13**(5–6), 347–369.

Lisboa, P.: 2002, A review of evidence of health benefit from artificial neural networks in medical intervention, *Neural Networks* **15**(1), 11–39.

Lopez, G., Batlles, F. and Trovar-Pescador, J.: 2005, Selection of input parameters to model direct solar irradiance by using artificial neural networks, *Energy* **30**, 1675–1684.

Lu, P. and Hsu, T.: 2002, Application of autoassociative neural network on gas-path sensor data validation, *Journal of Propulsion and Power* **18**(4), 879–888.

Lurie, P., Phillips, K., Avins, A., Kahn, J., Lowe, R., Franks, P. and Ciccarone, D.: 1992, Decision analysis models for HIV testing of health care workers and hospital-based patients, *Proceedings of 8th International AIDS Conference '92*, Amsterdam, Netherlands.

Mackay, D.: 1998, Introduction to Gaussian processes Neural networks and machine learning, *Computer and Systems Sciences* **68**, 135–165.

Mamdani, E.: 1977, Application of fuzzy logic to approximate reasoning using linguistic systems, *Fuzzy Sets and Systems* **26**, 1182–1191.

Manolios, P. and Fanelli, R.: 1994, First order recurrent neural networks and deterministic finite state automata, *IEEE Transactions on Neural Networks* **5**(2), 240–254.

Marwala, T.: 2001, Probabilistic fault identification using a committee of neural networks and vibration data, *Journal of Aircraft* **38**(1), 138–146.

Marwala, T. and Chakraverty, S.: 2006, Fault classification in structures with incomplete measured data using autoassociative neural networks and genetic algorithm, *Current Science* **90**(4), 542–549.

MATLAB: 2004, *Matlab and Simulink for Technical Computing*, Mathworks.

McMullen, P.: 2001, An ant colony optimization approach to addressing JIT sequencing problems with multiple objectives, *Artificial Intelligence in Engineering* **15**, 309–317.

Michalewicz, Z.: 1996, *Genetic Algorithms + Data Structures = Evolution Programs*, Springer, Berlin, Germany.

Monash: 2006, About ROC curves. `http://pops.csse.monash.edu.au/roccurves_doc.html`.

Nabney, I.: 2003, *NETLAB: Algorithms for Pattern Recognition*, Springer, London, UK.

Neal, R.: 1996, *Bayesian Learning for Neural Networks. Lecture notes in statistics*, Springer-Verlag, New York, USA.

Ohno-Machado, L.: 1996, Sequential use of neural networks for survival prediction in AIDS, *Proceedings: AMMA annual Fall Symposium '96*, pp. 170–174.

Olurotimi, O.: 1994, Recurrent neural network training with feedforward complexity, *IEEE Transactions on Neural Networks* **5**(2), 185–197.

Orr, J.: 2006, Neural Networks Slides. `http://www.willamette.edu/~gorr/classes/cs449/`.

Papadokonstantakis, S., Lygeros, A. and Jacobsson, S.: 2006, Comparison of recent methods for inference of variable influence in neural networks, *Neural Networks* **19**, 500–513.

Pawlak, Z.: 1991, *Rough sets: Theoretical Aspects of Reasoning About Data*, Kluwer Academic Publishers, Dordrecht.

Pelckmans, K., Brabanter, J. D., Suykens, J. and Moor, B. D.: 2005, Handling missing values in support vector machine classifiers, *Neural Networks* **18**(5–6), 684–692.

Pesonen, E., Eskelinen, M. and Juhola, M.: 1998, Treatment of missing data values in a neural network based decision support system for acute abdominal pain, *Artificial Intelligence in Medicine* **13**, 139–146.

Poundstone, K., Strathdee, S. and Celectano, D.: 2004, The social epidemiology of human immunodeficiency virus/acquired Immunodeficiency syndrome, *Epidemiologic Reviews* **26**, 22–35.

Quinlan, J.: 1993, *C4.5 Programs for machine learning*, Morgan Kaufmann Publishers, San Mateo, CA.

Rank, E.: 2003, Application of Bayesian trained RBF networks to nonlinear time-series modelling, *Journal of Signal Processing,* **83**, 1393–1410.

Rencher, A.: 1995, *Methods of Multivariate Analysis*, Wiley and Sons.

Root-Bernstein, R.: 1998, The evolving definition of AIDS, *Rethinking AIDS* . `http://www.virusmyth.net/aids/data/rrbdef.htm`.

Rowland, T., Ohno-Machado, L. and Ohrn, A.: 1998, Comparison of Multiple Prediction Models for Ambulation Following Spinal chord Injury, *In Chute* **31**, 528–532.

Sardari, S. and Sardari, D.: 2002, Applications of artificial neural network in AIDS research and therapy, *Current Pharmaceutical Design* **8**(8), 659–670.

Sawa, T. and Ohno-Machado, L.: 2003, A neural network-based similarity index for clustering DNA microarray data, *Computers in Biology and Medicine* **33**(1), 1–15.

Sivagaminathan, R. and Ramakrishnan, S.: 2007, A hybrid approach for feature subset selection using neural networks and ant colony optimization, *Expert Systems with Applications* **33**(1), 49–60.

Smauoi, N. and Al-Yakoob, S.: 2003, Analyzing the Dynamics of Cellular Flames Using Karhunen-Loeve Decomposition and Autoassociative Neural Networks, *Society for Industrial and Applied Mathematics* **24**(5), 1790–1808.

Smith, L.: 2003, An Introduction to Neural Networks, *University of Sterling* . `http://www.cs.stir.ac.uk/~lss/NNIntro/InvSlides.html`.

Ssali, G. and Marwala, T.: 2008, Estimation of missing data using computational intelligence and Decision Trees, **arXiv.0709.1640**.

Szpurek, D., Moszynski, R., Smolen, A. and Sajdak, S.: 2005, Artificial neural network computer prediction of ovarian malignancy in women with adnexal masses, *International Journal of Gynecology and Obstetrics* **89**(2), 108–113.

Takagi, T. and Sugeno, M.: 1985, Fuzzy identification of systems and its application to modeling and control, *IEEE Transactions on Systems, Man and Cybernetics* **15**(1), 116–132.

Tan, A.-H. and Pan, H.: 2005, Predictive neural network for gene expression data analysis, *Neural Networks* **18**(3), 297–306.

Tandon, R., Adak, S. and Kaye, J.: 2006, Neural Network for longitudinal studies in Alzheimer's disease, *Artificial Intelligence in Medicine* **36**(3), 245–255.

Tettey, T. and Marwala, T.: 2006, *Neuro-fuzzy modeling and fuzzy rule extraction applied to conflict management: Lecture Notes in Computer Science*, Vol. 4234, Springer-Berlin/Heidelberg.

Tettey, T., Nelwamondo, F. and Marwala, T.: 2007, HIV Data Analysis via Rule Extraction using Rough Sets, *International Joint Conference on Neural Networks* p. Submitted.

Tolle, K., Chen, H. and Chow, H.-H.: 2000, Estimating drug/plasma concentration levels by applying neural networks to pharmacokinetic data sets, *Decision Support Systems* pp. 139–151.

Twala, B., Jones, M. and Hand, D.: 2008, Good Methods for coping with missing data in decision trees, *Pattern Recognition Letters* **In Press, Accepted Manuscript**, Available online.

UNAIDS: 2006, Questions and Answers II: Basic facts about the HIV/AIDS epidemic and its impact. `http://www.unaids.org/en/resources/questions_answers.asp`.

USCensusBureau: 2004, HIV/AIDS Surveillance Data Base Installation. `http://www.census.gov/ipc/www/hsbhome.html`.

Vellido, A.: 2006, Missing data imputation through GTM as a mixture of t-distributions, *Neural Networks* **19**(10), 1624–1635.

Wang, D. and Lu, W.-Z.: 2006, Interval estimation of urban ozone level and selection of influential factors by employing automatic relevance determination model, *Chemosphere* **62**, 1600–1611.

WorldBank: 2002, Optimizing the allocation of resources among HIV prevention interventions in Honduras. `http://lnweb18.worldbank.org/external/lac/lac.nsf/FILE/OPTIMALALLOCATIONHIV-AIDSHonduras.pdf`.

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

Yang, Y. and John, R.: 2006, Roughness Bound in Set-oriented Rough Set Operations, *IEEE International Conference on Fuzzy Systems*, Vancouver, Canada, pp. 1461–1468.

Yi, S. and Chung, M.: 1993, Identification of fuzzy relational model and its application to control, *Fuzzy Sets and Systems* **59**, 25–33.

Zadeh, L.: 1973, Outline of a new approach to the analysis of complex systems and decision processes, *IEEE Transactions on Systems, Man and Cybernetics* **1**, 28–44.

Zhang, Y., Li, H., Hou, A. and Havel, J.: 2006, Artificial neural networks based on principal component analysis input selection for quantification in overlapped capillary electrophoresis peaks, *Chemometrics and Intelligent Laboratory Systems* **82**, 165–175.

Appendix

7.3 Structure of the Compact Disc

Main Folder: PHD Code

SubFolder: Autoencoder and MLP Classification

SubSubFolder: Autoencoder Prediction

SubSubFolder: MLP Prediction

Subfolder: Missing Data Code

Subfolder: Demographic Influence Code

Subfolder: Chapter 5 Code

SubSubFolder: Neuro-fuzzy code

SubSubFolder: Rough Sets code

Subfolder: ARD Code

Main Folder: PHD Thesis

Subfolder: Latex Source Codes

SubSubFolder: Figures

Subfolder: PhD PDF File

Subfolder: Journal and Conference Papers