

https://dspace.org/

Using optimisation techniques to granulise rough set partitions

Crossingham, Bodie

2009-01-26T08:58:40Z
http://hdl.handle.net/10539/5967

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

Using Optimisation Techniques to Granulise Rough Set Partitions

Bodie Crossingham

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

Johannesburg, 2008

Declaration

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

Signed this ____ day of _____ 20__

_____

Bodie Crossingham

Abstract

Rough set theory (RST) is concerned with the formal approximation of crisp sets and is a mathematical tool which deals with vagueness and uncertainty. RST can be integrated into machine learning and can be used to forecast predictions as well as to determine the causal interpretations for a particular data set. The work performed in this research is concerned with using various optimisation techniques to granulise the rough set input partitions in order to achieve the highest forecasting accuracy produced by the rough set. The forecasting accuracy is measured by using the area under the curve (AUC) of the receiver operating characteristic (ROC) curve. The four optimisation techniques used are genetic algorithm, particle swarm optimisation, hill climbing and simulated annealing. This newly proposed method is tested on two data sets, namely, the human immunodeficiency virus (HIV) data set and the militarised interstate dispute (MID) data set. The results obtained from this granulisation method are compared to two previous static granulisation methods, namely, equal-width-bin and equal-frequency-bin partitioning. The results conclude that all of the proposed optimised methods produce higher forecasting accuracies than that of the two static methods. In the case of the HIV data set, the hill climbing approach produced the highest accuracy, an accuracy of 69.02% is achieved in a time of 12624 minutes. For the MID data, the genetic algorithm approach produced the highest accuracy. The accuracy achieved is 95.82% in a time of 420 minutes. The rules generated from the rough set are linguistic and easy-to-interpret, but this does come at the expense of the accuracy lost in the discretisation process where the granularity of the variables are decreased.

To my family and friends...

Acknowledgements

I wish to thank my parents for all their support they have given me through out my studies. I thank them for instilling in me the important lesson of working hard. Most of all, thanks to my parents who provided me with the greatest item of worth - opportunity. I would also like to thank my siblings for all their encouragement and support. I also would like to thank my friends and girl friend for all their support through out my studies.

I would like to acknowledge my supervisor Prof. Tshilidzi Marwala. It was his determination to succeed and his wealth of knowledge that drives all of us in C-lab to achieve as much as we can. This determination of his is expressed in one of his favourite quotes, "The opportunity of a lifetime must be seized during the lifetime of the opportunity". Prof Marwala inspires all those around him, and it is this inspiration that drove me to pursue my Masters. I thank him for encouraging me to study further and for proving me with the financial backing to study as well as to attend conferences.

I thank all my colleagues working with me in the C-lab. It is great to be surrounded by such ambitious guys with whom I can share my ideas and get feedback from all your comments and criticisms on my work.

Contents

Declaration i

Abstract ii

Acknowledgements iv

Contents v

List of Figures ix

List of Tables xi

Nomenclature xiii

1 Introduction 1

1.1 Background 1

1.2 Research Focus and Approach 2

1.3 Literature Review 4

1.3.1 Rough Set Theory 4

1.3.2 Optimisation Techniques 4

1.3.3 Testing using HIV and MID data 7

1.4 Outline of Dissertation 8

2 The Theory of Rough Sets 10

2.1 Information Table 10

2.2 Information System 11

2.3 Indiscernibility Relation 12

2.4 Lower and Upper Approximations 13

2.5 Rough Membership Function 16

2.6 Rough Set Accuracy 17

2.7 Rough Sets Formulation 20

3 Using Optimisation Techniques for Discretisation 21

3.1 Introduction 21

3.2 Discretisation Criteria 22

3.3 Equal-Width-Bin and Equal-Frequency-Bin Partitioning 23

3.3.1 Equal-Width-Bin Partitioning 23

3.3.2 Equal-Frequency-Bin Partitioning 24

3.4 Genetic Algorithm 25

3.5 Particle Swarm Optimisation 28

3.6 Hill Climbing 31

3.7 Simulated Annealing 33

4 Experimental Investigation I : Modelling HIV **36**

4.1 Introduction 36

4.2 HIV Data 38

4.2.1 Data 38

4.2.2 Cleaning the Data 40

4.3 Results Obtained on HIV data 41

4.3.1 Results of Optimisation Techniques 42

4.3.2 Comparison of Methods on HIV data 44

5 Experimental Investigation II : Modelling MID **50**

5.1 Introduction 50

5.2 MID Data 52

5.2.1 Data 53

5.2.2 Cleaning the Data 54

5.3 Results Obtained on MID data 55

5.3.1 Results of Optimisation Techniques 55

5.3.2 Comparison of Methods on MID data 57

6 Conclusion **63**

6.1 Summary of Findings 63

6.2 Recommendations for Further Work 64

A Algorithms **66**

A.1 Rough Set Model Generation 66

A.2 Receiver Operating Characteristics (ROC) Curve 67

A.3 Area Under Curve (AUC) 68

B Flow Diagrams **69**

B.1 Genetic Algorithm 70

B.2 Particle Swarm Optimisation 71

B.3 Hill Climbing 72

B.4 Simulated Annealing 73

C Published Work **74**

References **75**

List of Figures

2.1 Approximating the set of HIV positive patients of Table 2.1 using all the conditional attributes. 15

3.1 Block diagram of the sequence of events in creating a rough set model 22

3.2 Figure illustrating a local maxima versus a global maxima 31

4.1 Receiver operating characteristic (ROC) curves for GA, EWB and EFB on HIV data 44

4.2 Receiver operating characteristic (ROC) curves for PSO, EWB and EFB on HIV data 45

4.3 Receiver operating characteristic (ROC) curves for HC, EWB and EFB on HIV data 45

4.4 Receiver operating characteristic (ROC) curves for SA, EWB and EFB on HIV data 46

4.5 Receiver operating characteristic (ROC) curves for GA, PSO, HC and SA on HIV data 47

4.6 Discretisation points for the various attributes using the GA method on HIV data 48

5.1 Receiver operating characteristic (ROC) curves for GA, EWB and EFB on MID data 57

5.2 Receiver operating characteristic (ROC) curves for PSO, EWB and EFB on MID data 58

5.3 Receiver operating characteristic (ROC) curves for HC, EWB and EFB on MID data 58

5.4 Receiver operating characteristic (ROC) curves for SA, EWB and EFB on MID data 59

5.5 Receiver operating characteristic (ROC) curves for GA, PSO, HC and SA on MID data 59

5.6 Discretisation points for the various attributes using the GA method on MID data 62

B.1 Flow diagram of genetic algorithm optimisation 70

B.2 Flow diagram of particle swarm optimisation 71

B.3 Flow diagram of hill climbing optimisation 72

B.4 Flow diagram of hill climbing optimisation 73

List of Tables

2.1 Information table of the HIV data. 11

2.2 Confusion matrix representing the dispositions of the set of instances 18

4.1 Summary of the HIV data set variables 39

4.2 Representation of a Confusion Matrix 41

4.3 Confusion matrix for EWB discretisation on HIV data 42

4.4 Confusion matrix for EFB discretisation on HIV data 42

4.5 Confusion matrix of the GA discretisation on HIV data 42

4.6 Confusion matrix of a the PSO discretisation on HIV data 43

4.7 Confusion matrix of a the HC discretisation on HIV data 43

4.8 Confusion matrix of a the SA discretisation on HIV data 44

4.9 Results obtained for the four optimisation and two static discretisation methods 47

5.1 Summary of the MID data set variables 54

5.2 Confusion matrix for EWB discretisation on MID data 55

5.3 Confusion matrix for EFB discretisation on MID data 55

5.4 Confusion matrix of the GA discretisation on MID data 56

5.5 Confusion matrix of a the PSO discretisation on MID data 56

5.6 Confusion matrix of a the HC discretisation on MID data 56

5.7 Confusion matrix of a the SA discretisation on MID data 56

5.8 Results obtained for the four optimisation and two static discretisation methods 60

Nomenclature

RST	Rough Set Theory
HIV	Human Immunodeficiency Virus
MID	Militarised Interstate Dispute
EWB	Equal Width Bin
EFB	Equal Frequency Bin
GA	Genetic Algorithm
PSO	Particle Swarm Optimisation
HC	Hill Climbing
SA	Simulated Annealing
RRHC	Random-Restart Hill Climbing
NN	Neural Network
AUC	Area Under Curve
ROC	Receiver Operating Characteristic
MLP	Multi-Layer Perceptron
SVM	Support Vector Machine
GSS	Golden Selection Search
COW	Correlates of War
PPS	Probability Proportion to Size
GDP	Gross Domestic Product
ARD	Automatic Relevance Detection
VFSR	Very Fast Simulated Re-annealing
ASA	Adaptive Simulated Annealing

Chapter 1

Introduction

1.1 Background

Rough set theory (RST) was introduced by Zdzislaw Pawlak in the early 1980s [1]. RST is a mathematical tool which deals with vagueness and uncertainty, it allows for the approximation of sets that are difficult to describe with the available information. It is of fundamental importance to artificial intelligence (AI) and cognitive science and is highly applicable to this study of performing the task of machine learning and decision analysis. RST can be seen as an extension of classical set theory by incorporating knowledge into its formalism, this allows sets to be approximated into a lower and an upper approximation of the original set. These approximations are the basic notion of rough set theory. Data can be acquired from measurement systems or human experts, and in principle, this data must be discrete [2]. There are methods which allow for continuous data to be processed, this involves discretisation, which converts the continuous data into discrete intervals. Several methods that have been previously, and that are currently used to perform this task are; boolean reasoning, Enc-MALIC, as well as the more popularly used equal-width-bin (EWB) partitioning and equal-frequency-bin (EFB) partitioning [3, 4].

For knowledge acquisition from data with numerical attributes, special techniques are applied, most frequently a step taken before the main step of rule induction or

decision tree generation, called discretisation, is used [5]. It is this step that this research focuses around. In this work, the use of combinatorial optimisation techniques to discretise the rough set partitions are explored. Combinatorial optimisation is related to algorithm and computational complexity theory, and is the intersection of the artificial intelligence, mathematical and software engineering fields. Combinatorial optimisation is the process of finding an optimal solution in a problem space. In the case of optimising the partition sizes of a rough set, the particular optimisation technique will search the entire search space, and create the partition sizes according to an objective criteria. The criteria used in this research is the maximisation of the classification accuracy of the rough set. Four optimisation techniques are used, namely; genetic algorithm (GA), particle swarm optimisation (PSO), hill climbing (HC) and simulated annealing (SA). The results produced from these four methods are compared to that of the more commonly used equal-width-bin and equal-frequency-bin partitioning methods. To test the above method, human immunodeficiency virus (HIV) data is used as well as militarised interstate dispute (MID) data.

RST's main advantage lies in its ability to deal with imprecise or conflicting data and it also produces easy-to-interpret and linguistic rules. Its downfall though is its classification or forecasting accuracy. Rough sets compromise accuracy over rule interpretability and this is brought about in the discretisation process where the granularity of the variables are decreased. RST is used for classification as it produces a balance between transparency of the rough set model and accuracy of data estimation, but it does come at the cost of high computational effort.

1.2 Research Focus and Approach

The approach taken to determine the effects of optimising rough sets using various techniques is as follows:

1. Create a rough set model to be used on the various data sets.

2. Using the HIV data set, discretise the partitions using EWB and EFB partitioning, this determines the classification accuracy produced of the well known, non-optimised methods.

3. Apply the GA, PSO, HC and SA optimisation techniques to the rough set for analysis on the HIV data, doing so generates the optimal partition sizes for each method.

4. Using these newly generated optimal values, compare them to the results produced using EWB and EFB partitioning.

5. Apply the above methodology to the conflict data set (MID) and investigate the results produced by the optimised and non-optimised methods.

The rough sets are discretised into four partitions for the purpose of comparison for this thesis. Four partitions are chosen as it provides a good balance between accuracy achieved and computational time.

The focus of the method described in this research is the classification accuracy of the rough set; this accuracy is based on the receiver operating characteristic (ROC) curves. The area under the curve (AUC) of the ROC curves is used as the performance criteria for each method and this will be explained fully in Chapter 2. It must be noted that, although there are many aspects involved in RST from the generation of reducts to the optimisation of the *number* of split/discretisation points, the focus of this research is how to optimally granulise the input data into a predefined four partitions. For the process of knowledge acquisition there are two discretisation approaches; the first approach is to discretise numerical attributes during the process of knowledge acquisition, the second and more commonly used approach is to use discretisation as a preprocessing step [5]. This research uses the later approach and once discretisation is performed, the attributes are then represented in a decision table from which rules are extracted.

1.3 Literature Review

1.3.1 Rough Set Theory

RST is a mathematical tool which deals with vagueness and uncertainty. It is of fundamental importance to artificial intelligence (AI) and cognitive science and is highly applicable to this study of performing the task of machine learning and decision analysis. The advantages of rough sets as with many other AI techniques are that they do not require rigid *a priori* assumptions on the mathematical nature of such complex relationships as do commonly used multivariate statistical techniques [6, 7, 8]. RST is based on the assumption that the information of interest is associated with *some information* of its universe of discourse [9, 10]. The main concept of rough set theory is an indiscernibility relation (indiscernibility meaning indistinguishable from one another). Rough set theory handles inconsistent information using two approximations, namely the upper and lower approximation. RST theory is primarily limited to binary-concept, providing either *yes/no* results in decision processes or *positive/negative* results in classification processes [11].

Rough sets have been used in many real-life applications, these include various biomedical applications [12, 13, 14], control algorithm acquisition [15], prediction of aircraft component failure, fault diagnosis and stock market analysis [16, 17, 18]. The use of rough sets have also been investigated in the analysis of conflict [19, 20]. Very limited work has been done on the investigation of applying RST to HIV. Tettey *et al* have looked at applying RST to HIV [21], and it is based on this work that an investigation on the maximisation of rough set classification is performed.

1.3.2 Optimisation Techniques

The genetic algorithm (GA) was developed extensively by John Holland in the mid 70's [22]. It is inspired by the principles of genetics and evolutionary biology, it uses techniques such as inheritance, mutation, selection and crossover. The GA employs

the principle of survival of the fittest in its search process, the fitness of every individual in the population is evaluated, multiple individuals are stochastically selected from the current population (based on their fitness), and modified (mutated) to form a new population. A new population is then used in the next iteration of the algorithm. The process terminates when either a set number of generations are executed or once a satisfactory fitness level is reached. It must be noted that if the algorithm terminates after reaching the maximum number of generations, the optimal solution may not have been reached. The fitness/evaluation function is the only part of the GA that has any knowledge of the problem.

Particle swarm optimisation (PSO) is a stochastic, population-based solving algorithm. It was invented by Kennedy and Eberhart in the mid 1990's whilst trying to simulate the swarm of birds as part of a sociocognitive study investigating the knowledge of "collective intelligence" in biological populations [23]. As mentioned, PSO is based on the analogy of flocks of birds, schools of fish, and herds of animals and how they maintain an "evolutionary advantage" by either adapting to their environment, avoiding predators or finding rich sources of food, all by the process of "information sharing" [23]. PSO initialises by creating a random population of solutions (particles), each solution's fitness is evaluated (according to a fitness function) and the fittest or best solution is noted (often known as *pbest*). All the solutions in the problem space have its own set of coordinates and these coordinates are associated with the fittest solution found (so far). Another value noted is the best solution found in the neighbourhood of the particular solutions or particles (often known as *lbest*). The PSO concept consists of accelerating the velocities of each particle towards the *pbest* and *lbest* locations, the acceleration for both *pbest* and *lbest* are separate randomly weighted values [24]. The next iteration takes place after all particles have been moved. It is this behaviour that mimics the swarm of birds after which the technique was developed.

Hill climbing (HC) is another optimisation technique used to partition the rough sets. Hill climbing works on the premise of obtaining the state of a current node and then moving towards a state which is better than the current one. Depending on the

variant of hill climbing implemented, either the closest node is chosen for its state to be evaluated, which is referred to as simple hill climbing. Another variant would be if all successors are compared and the closest solution is chosen, which is called steepest ascent hill climbing. For the purpose of this research, the later is chosen. The algorithm is called gradient ascent as the objective function is maximised, where as, if the function were to be minimised, the algorithm would be gradient descent.

The final optimisation technique investigated to discretise the rough set partitions is simulated annealing. Simulated annealing (SA) was invented in 1983 by Kirkpatrick, Gelatt and Vecchi [25]. SA is an iterative procedure that continuously updates one candidate solution until a termination condition is reached [25], it is a technique that mathematically mirrors the cooling of a set of atoms to a state of minimum energy. The SA algorithm replaces the current solution by a random nearby solution, chosen with a probability that depends on the difference between the corresponding function values and on a global parameter T (temperature). SA is based on the manner in which liquids freeze or metals recrystalise in the process of annealing. SA is a generalisation of a Monte Carlo method and relies on the Metropolis acceptance criterion [26]. SA is advantageous as its implementation allows the solution to move further away as well as closer to the solution, this prevents the solution getting stuck at a local minima but rather at the desired global minima. These techniques will be explained in more detail in Chapter 3.

There have been many uses of the four mentioned optimisation techniques, but no work has been done yet on the optimisation of rough set partition sizes. Limited work performed on RST using optimisation techniques include; feature extraction using rough set theory and genetic algorithms [11], finding minimal reducts using PSO [27]. Chen *et al* have used a GA to optimise the *number* of partitions. For a given number of partitions, the optimisation of the size of the cuts within each granule is yet to be investigated.

1.3.3 Testing using HIV and MID data

The proposed method is tested on two data sets. These two data sets are the HIV (human immunodeficiency virus) data set and the MID (militarised interstate dispute) data set. Previous uses of the particular data sets are listed below but a full description of the data used is given in Chapters 4 and 5.

Methods applied to HIV data:

Previously, computational intelligence techniques have been used extensively to analyse HIV. Leke *et al* have used autoencoder network classifiers, inverse neural networks, as well as conventional feedforward neural networks to analyse HIV [28, 29, 30], they used the inverse neural network for adaptive control of HIV status to understand how the demographic factors affect the risk of HIV infections [30].

Although an accuracy of 84% is achieved when using the autoencoder method [28], it is disadvantageous due to its "black box" nature, this disadvantage also applies to the other mentioned neural network techniques. The reason NNs are considered to be black boxes is because their connection weights and transfer functions are frozen upon completion of training of the neural network [31]. Chapter 4 will discuss previous work performed on HIV in full.

Methods applied to MID data:

One of the methods that have been applied successfully to MID's are neural networks (NN) [32, 33, 7], in particular the multi-layer perceptron (MLP) neural network. Neural networks are a mathematical technique especially suitable to the interactive, non-linear and contingent relations across the variables that may trigger militarised interstate disputes [33, 34]. Neural networks also provide a clear answer to the accuracy of predictions with numerical accuracies of the number of dyads involved in conflict as well as those dyads whose relationship is peace. If the prediction

accuracy is only what is required, it is argued that on the basis of these prediction accuracies, the interpretation of NN's results are unambiguous [32], this research is however interested in the causal interpretations as well as prediction accuracy of MIDs and thus uses RST. There are disadvantages associated with the use of NN to accurately forecast conflict; the first being, NN are not efficient classifiers of rare events, the conflict data is skewed which has the effect of biasing the result towards the modal value (i.e. the result with the most common events) and this is problematic as more than 96% of the results have the outcome of zero (peace); and the second disadvantage is that it is difficult to comprehend the causal model that the trained networks have internally constructed [32]. To overcome the first problem of rare events prediction, the cross validation technique is used as it allows for the maximisation of data [35], this is explained in Chapter 5.

Support vector machines (SVM) are another method that have been used to forecast and analyse conflict [8]. Habtemariam and Marwala compared the use of MLP and SVM to analyse conflict situations within a given set of data. It was found that SVM classifies both true positive and true negative results on the data with a higher degree of accuracy than that of the MLP method. SVM predicts correct results with an accuracy of 79% while an accuracy of 74% is achieved for the MLP [8]. Receiver operating characteristic (ROC) curves were used to compare the performance of the NN and the SVM classifiers. The area under curve (AUC) is used to measure the relative performances. The NN and SVM had AUC of 0.81 and 0.84 with errors of 0.00998 and 0.01022 respectively [8]. Marwala and Lagazio have conducted a study into modelling interstate conflict using Bayesian trained neural networks. A control theory approach which makes use of the golden selection search (GSS) optimisation which is then used to identify input variables that would give a peaceful outcome [36].

1.4 Outline of Dissertation

As mentioned, RST has been used previously to model interstate conflict as well as HIV. Amongst all the applications to which RST has been applied, it has yet

been investigated what the optimal partition sizes are for a particular data set when classifying using rough sets. A significant contribution is given in Chapters 4 and 5 where the results obtained using optimisation methods are compared to that of the non-optimised EWB and EFB techniques. The outline of this thesis is given below:

Chapter 2 describes the core concepts of rough set theory. It also describes how the rough sets are formulated and how the generated rules are extracted. The performance criteria or accuracy of the rough set is determined as the area under the curve (AUC) of the ROC curves, these concepts are also explained.

Chapter 3 gives a detailed description of the four optimisation techniques used and how they are applied to RST as performed in this work. The non-optimised methods of EWB and EFB partitioning are also explained.

Chapter 4 displays the results obtained using a neural network on the human immunodeficiency virus data set after which the results obtained for the application of rough sets to HIV is given. The HIV data set is given with a description of each variable. Also included in Chapter 4 is the comparison of the results obtained using the optimised rough set partitions against that of EWB and EFB partitioning.

Chapter 5 describes the militarised interstate dispute (MID) data set. The variables used are given and explained, and the data set used (correlates of war) is discussed. Similarly to Chapter 4, the classification accuracy obtained for the rough set using both the optimised and non-optimised methods are compared.

Chapter 6 summarises the findings of this work and presents recommendations for future work on this subject.

Appendix A gives the algorithms for various procedures as discussed in the thesis.

Appendix B illustrates the flow diagrams of the four optimisation techniques used.

Appendix C lists the papers published based on the work performed in this thesis

Chapter 2

The Theory of Rough Sets

The main goal of rough sets is to synthesise approximations of concepts from the acquired data. Unlike other methods used to handle uncertainty, rough set theory has its own unique advantages: it does not require any preliminary or additional information about the empirical training data such as probability distributions in statistics; basic probability assignment in Dempster Shafer theory of evidence or the value of possibility in fuzzy set theory [37].

Rough set theory deals with the approximation of sets that are difficult to describe with the available information [13]. It deals predominantly with the classification of imprecise, uncertain or incomplete information. Two approximations, namely the upper and lower approximation, are formed to deal with inconsistent information. These approximations along with other concepts that are fundamental to RST theory are given below.

2.1 Information Table

The data used is represented using an information table, an example of an information table can be shown using data from the HIV data set for the *ith* object. This is given in Table 2.1 below.

In the information table, each row represents a new case (or *object*). Besides *HIV*

Table 2.1: Information table of the HIV data.

	Race	Mothers Age	Education	Gravidity	Parity	Fathers Age	HIV Status
$Obj^{(1)}$	2	32	13	1	1	22	1
$Obj^{(2)}$	3	22	5	2	1	25	1
$Obj^{(3)}$	1	35	6	1	0	33	0
$Obj^{(4)}$	1	35	6	1	0	33	0
$Obj^{(5)}$	3	22	5	2	1	25	0
$Obj^{(6)}$	2	30	10	1	1	30	0
$Obj^{(7)}$	3	22	5	2	1	25	1
.	.	.	.	.	.	.	.
$Obj^{(i)}$	2	27	9	3	2	30	0

Status, each of the columns represent the respective case's variables (or *condition attributes*). The *HIV Status* is the outcome (also called the *concept* or *decision attribute*) of each object. The outcome contains either a 1 or 0, and this indicates whether the particular case is infected with HIV or not. For the application of this research, the outcome is known. This *a posteriori* knowledge is expressed as the *decision attribute* and this process is known as *supervised learning*. It can be seen from Table 2.1 that cases/objects 3 and 4 have the same condition attributes, and cases 5 and 7 have the same condition attributes. These cases are referred to as indiscernible cases; this concept is explained later. Taking into consideration that the process represented is supervised learning, of the indiscernible cases, cases 3 and 4 have the same decision attribute while cases 5 and 7 have conflicting/different decisions.

2.2 Information System

Once the information table is obtained, the data is discretised into four partitions as mentioned earlier. An information system can be understood by a pair Λ = (**U**,A), where **U** and A, are finite, non-empty sets called the universe, and the set of attributes, respectively [14].

For every attribute $a \in A$, we associate a set V_a, of its values, where V_a is called the

value set of a [1].

$$a : \mathbf{U} \rightarrow V_a \tag{2.1}$$

Any subset B of A determines a binary relation *I(B)* on $\mathbf{U}$, which is called an indiscernibility relation.

In the case as shown in Table 2.1, the process is one of supervised learning, i.e. the decision is known. Information systems of this kind are known as decision systems. Formally, a decision system is any information system of the form $\Lambda = (\mathbf{U}, A \cup \{d\})$, where $d \notin A$ is the *decision attribute* [9]. The elements of A are the *conditional attributes*.

Let $\Lambda = (\mathbf{U}, A \cup \{d\})$ be given and assume the set $V_d = \{\mathrm{v}_d^1, ..., v_d^{r(d)}\}$ to be the extension of the decision attribute. Denote the image $d(U) = \{k \mid d(x) = k, x \in U\}$, where $d(U) \subseteq V_d$. The cardinality of set $d(U)$ is called the *rank of d*, and is denoted by *r(d)* [38].

The decision d determines the partition $CLASS_\Lambda(d) = \{X_\Lambda^1, ..., X_\Lambda^{r(d)}\}$ of the universe U, where $X_\Lambda^k = \{x \in U \mid d(x) = v_d^k\}$ for $1 \leq k \leq r(d)$. $CLASS_\Lambda(d)$ is called the classification of objects in Λ determined by the decision d. The set X_Λ^i is called the *ith decision class of* Λ. The concept of an indiscernibility relation will be explained next.

2.3 Indiscernibility Relation

An information table expresses all the knowledge about a model. This table is generally unnecessarily large, this is in part because it is redundant in at least two ways; the same or indiscernible objects may be represented several times, and some of the attributes may be superfluous [2]. The concept of indiscernibility is one of the main concepts of rough set theory (indiscernibility meaning indistinguishable from

one another). Sets that are indiscernible are called elementary sets and these are considered the building blocks of RST's knowledge of reality. A union of elementary sets is called a crisp set, while any other sets are referred to as rough or vague.

More formally, for a given information system Λ, then for any subset $B \subseteq A$, there is an associated equivalence relation *I(B)* called the *B-indiscernibility relation* and is represented as shown in 2.2 below:

$$(x, y) \in I(B) \text{ iff } a(x) = a(y) \tag{2.2}$$

RST offers a tool to deal with indiscernibility, the way in which it works is for each concept/decision X, the greatest definable set containing X and the least definable set containing X are computed. These two sets are called the lower and upper approximation respectively.

2.4 Lower and Upper Approximations

The sets of cases/objects with the same outcome variable are assembled together. This is done by looking at the "purity" of the particular objects attributes in relation to its outcome. For example, with cases 5 and 7 in Table 2.1, it is not possible to induce a crisp (precise) description of such cases, it is from this that the notion of rough sets emerges. In most cases it is not possible to define cases into crisp sets, in such instances lower and upper approximation sets are defined.

The lower approximation is defined as *the collection of cases whose equivalence classes are fully contained in the set of cases we want to approximate* [13]. Let $x \subseteq \boldsymbol{U}$ be a target set that we wish to represent using attribute subset B. The lower approximation of set X is denoted $\underline{B}X$ and mathematically it is represented as:

$$\underline{B}X = \{x \in \mathbf{U} : B(x) \subseteq X\} \tag{2.3}$$

Where $B(x)$ The upper approximation is defined as *the collection of cases whose equivalence classes are at least partially contained in the set of cases we want to approximate* [13]. The upper approximation of set X is denoted $\overline{B}X$ and is mathematically represented as:

$$\overline{B}X = \{x \in \mathbf{U} : B(x) \cap X \neq \emptyset\} \tag{2.4}$$

It is through these lower and upper approximations that any rough set is defined. Lower and upper approximations are defined differently in literature, but it follows that a crisp set is only defined for $\overline{B}X = \underline{B}X$. Objects that cannot decisively be classified into X on the basis of knowledge in B, are grouped into the *B-boundary region of X*, this is represented as:

$$BN_B(X) = \overline{B}X - \underline{B}X \tag{2.5}$$

The *B-outside region of X* is the set $U - \overline{B}X$ which consists of objects that can be stated with certainty of not belonging to X (on the basis of knowledge of B) [2].

Using the example given in Table 2.1 we can synthesise definitions of the outcome in terms of the conditional attributes. If we consider the outcome of *HIV positive* i.e. Let $H = \{x \mid HIV(x) = 1\}$, then we obtain the following approximation regions:

- $\underline{B}H = Obj^{(1)}, Obj^{(2)}$;
- $\overline{B}H = Obj^{(1)}, Obj^{(2)}, Obj^{(5)}, Obj^{(7)}$;
- $BN_B(H) = Obj^{(5)}, Obj^{(7)}$
- $U - \overline{B}H = Obj^{(3)}, Obj^{(4)}, Obj^{(6)}$.

Thus it follows that the outcome *HIV Positive* is rough since the boundary region is not empty. Figure 2.1 illustrates this.

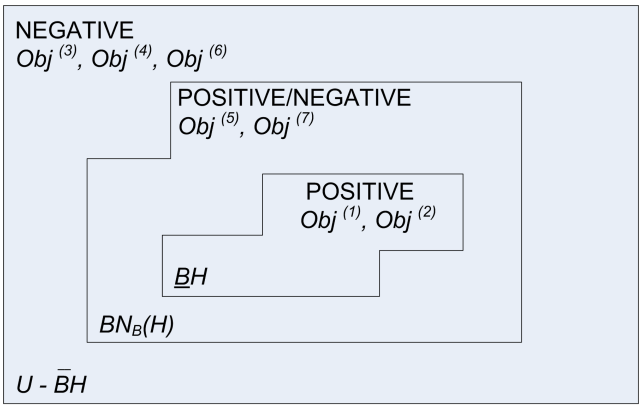

Figure 2.1: Approximating the set of HIV positive patients of Table 2.1 using all the conditional attributes.

The above state approximations have the following properties [2]:

1. $\underline{B}(X) \subseteq X \subseteq \overline{B}(X)$
2. $\underline{B}(\emptyset) = \overline{B}(\emptyset) = \emptyset; \underline{B}(U) = \overline{B}(U) = U$
3. $\overline{B}(X \cup Y) = \overline{B}(X) \cup \overline{B}(Y)$
4. $\underline{B}(X \cap Y) = \underline{B}(X) \cap \underline{B}(Y)$
5. $X \subseteq Y$ *implies* $\underline{B}(X) \subseteq \underline{B}(Y)$ *and* $\overline{B}(X) \subseteq \overline{B}(Y)$
6. $\underline{B}(X \cup Y) \supseteq \underline{B}(X) \cup \underline{B}(Y)$
7. $\overline{B}(X \cap Y) \subseteq \overline{B}(X) \cap \overline{B}(Y)$
8. $\underline{B}(U - X) = -\overline{B}(X)$
9. $\overline{B}(U - X) = -\underline{B}(X)$
10. $\underline{B}(\underline{B}(X)) = \overline{B}(\underline{B}(X)) = \underline{B}(X)$
11. $\overline{B}(\overline{B}(X)) = \underline{B}(\overline{B}(X)) = \overline{B}(X)$

It can be seen that the interior and the closure of the set in the topology generated by data are the lower and upper approximations respectively. Four basic classes of rough sets can be defined, i.e. four categories of vagueness:

1. X is *roughly B-definable*, iff $\underline{B}(X) \neq \emptyset$ and $\overline{B}(X) \neq U$
2. X is *internally B-undefinable*, iff $\underline{B}(X) = \emptyset$ and $\overline{B}(X) \neq U$
3. X is *externally B-undefinable*, iff $\underline{B}(X) \neq \emptyset$ and $\overline{B}(X) = U$
4. X is *totally B-undefinable*, iff $\underline{B}(X) = \emptyset$ and $\overline{B}(X) = U$

If X is *roughly B-definable*, the information provided by the attributes of B is sufficient to determine that some elements of U belongs to X, as well as that, some elements of U belong to $U - X$.

If X is *internally B-undefinable*, the information provided by the attributes of B is sufficient to determine that some elements of U belongs to $U - X$, but we are unable to determine whether any element of U belongs to X.

If X is *externally B-undefinable*, the information provided by the attributes of B is sufficient to determine that some elements of U belongs to X, but we are unable to determine whether any element of U belongs to $U - X$.

If X is *totally B-undefinable*, the information provided by the attributes of B is insufficient to determine whether any elements of U belongs to X or $U - X$.

It must be noted that for most cases in RST, reducts are generated to enable us to discard functionally redundant information [1]. Rules can be extracted from these reducts, alternatively rules can be extracted from the generated approximations, this work uses the later.

2.5 Rough Membership Function

A rough membership function is a function $\mu_X^B : U \rightarrow [0, 1]$ that, when applied to object x, quantifies the degree of relative overlap between the set X and the indiscernibility set $[x]$ to which x belongs. This membership function is a measure

of the plausibility of which an object x belongs to set X. The membership function is defined as:

$$\mu_A^X = \frac{|[X]_B \cap X|}{[X]_B} \tag{2.6}$$

The rough membership function can be treated as a fuzzification of rough approximation. It makes the translation from rough approximation into membership function. The fact that rough membership functions are derived from data, makes it stand out [39].

2.6 Rough Set Accuracy

Once the rules are extracted, they can be tested using a set of testing data. The classification output is expressed as a decision value which lies between 0 and 1. The accuracy of the rough set is determined using a technique called a receiver operating characteristic (ROC) curve. A ROC curve is a technique used for visualising, organising and selecting classifiers based on their performance. ROC curves depict the trade-off between hit rates and false alarm rates of classifiers, i.e. it depicts the trade-off between correctly predicted positives and falsely predicted positives. ROC curves have become increasingly important in the areas of cost-sensitive learning and learning in the presence of unbalanced data sets [40]. The rough set (classifier) will classify results based on a set of testing data that is given to it. The actual result or outcome of the testing data is known as an instance. Having obtained a classifier and an instance, there are four possible outcomes:

- If the instance is positive and it is classified as positive, then it is regarded as a *true positive*
- If the instance is positive and it is classified as negative, then it is regarded as a *false negative*

- If the instance is negative and it is classified as negative, then it is regarded as a *true negative*
- If the instance is negative and it is classified as positive, then it is regarded as a *false positive*

Given a classifier and a set of instances (test set), a two-by-two *confusion matrix* can be constructed. The confusion matrix will be in the form of that shown in Table 2.2. Correctly predicted positives are shown in the confusion matrix as actual true cases that are predicted as true cases. Falsely predicted positives are shown as actual true cases that are predicted as false cases.

Table 2.2: Confusion matrix representing the dispositions of the set of instances

	Actual True Cases	**Actual False Cases**	
Predicted True Cases	TP	FP	Pos
Predicted False Cases	FN	TN	Neg
	P	N	

where:

P = Positive

Pos = Positive

N = Negative

Neg = Negative

It is from the confusion matrix that several of the common metrics are derived, this will be illustrated below. True positive rate (*tp rate*), otherwise known as sensitivity of a classifier is estimated as:

$$tp\ rate \approx \frac{Positives\ correctly\ classified}{Total\ postives} \tag{2.7}$$

The false positive rate (*fp rate*), otherwise known as specificity of a classifier is:

$$fp\ rate \approx \frac{Negatives\ incorrectly\ classified}{Total\ negatives} \tag{2.8}$$

Formally these are represented as:

$$tp\ rate = \frac{TP}{P} \tag{2.9}$$

$$fp\ rate = \frac{FP}{N} \tag{2.10}$$

As previously mentioned, ROC accuracy is a measure of the trade-off between relative accuracy in correctly predicted positives against incorrectly predicted positives. Mathematically this is represented as:

$$accuracy = \frac{TP + TN}{P + N} \tag{2.11}$$

The ROC curve is a two-dimension depiction of a classifier performance. To compare two-or-more classifiers, we can reduce the ROC performance to a single scalar value representing expected performance [40]. A common method used and the method used in this research is, the area under curve (AUC). The AUC is deemed to be a better measure of classifier performance than accuracy [41, 42, 43].

The AUC is a portion of the area of the unit square on which the ROC curve is plotted and therefore, will have a value between 0 and 1.0. For a realistic classifier, the AUC will range between 0.5 (indicating that the test ranks no better than chance) to 1.0 (indicating perfect performance) [44]. The AUC of the ROC curve is computed using the trapezoidal method of integration, this can be shown to equal the Wilcoxon-Mann-Whitney statistic, or the probability that the classifier will assign a higher value to a positive case than to a negative case, assuming that the pair is randomly drawn from the population that the ROC curve is derived from [13].

2.7 Rough Sets Formulation

The process of modelling the rough set can be broken down into five stages;

The first stage would be to select the data. The two data sets to be used to test the method are obtained from the South African antenatal survey of 2001 [45], and the correlates of war (COW) project [46].

The second stage involves pre-processing the data to ensure it is ready for analysis, this stage involves discretising the data and removing unnecessary data (cleaning the data). This pre-processing step of removing outliers and cleaning the data is explained for the HIV and MID data in sections 4.2.2 and 5.2.2 respectively. Although the optimal selection of set sizes for the discretisation of attributes will not be known at first, an optimisation technique will be run on the set to ensure that the highest degree of accuracy is obtained when forecasting outcomes. This will be explained more clearly in Chapter 3.

If reducts were considered, the third stage would be to use the cleaned data to generate reducts. A reduct is the most concise way in which we can discern object classes [47]. In other words, *a reduct is the minimal subset of attributes that enables the same classification of elements of the universe as the whole set of attributes* [1]. To cope with inconsistencies, lower and upper approximations of decision classes are defined in this stage [1, 12, 14, 47, 48].

Stage four is where the rules are extracted or generated. The rules are normally determined based on condition attributes values [49]. Once the rules are extracted, they can be presented in an *if* CONDITION(S)-*then* DECISION format [50].

The fifth and final stage involves testing the newly created rules on a test set. The accuracy will be noted and sent back into the optimisation method used in step two and the process will continue until the optimum or highest accuracy is achieved. The algorithm for computing rough sets and extracting rules is given in Appendix A, Algorithm 1.

Chapter 3

Using Optimisation Techniques for Discretisation

3.1 Introduction

Chapter 1 explains that the data acquired from measurement systems or human experts must in principle be discrete [2]. The methods which allows continuous data to be processed involves discretisation. There are several methods available to perform discretisation but the two popularly used equal-width-bin (EWB) partitioning and equal-frequency-bin (EFB) partitioning are investigated [3, 4]. These methods are compared against the newly proposed method of using various optimisation techniques to discretise the continuous data.

The way in which these optimisation techniques work in conjunction with the rough set is explained in Appendix A, Algorithm 1, and it is also illustrated in Figure 3.1. The optimisation technique is run to create a set of four partitions from the given input data. Using these partitions, the rough set model is generated and the classification accuracy is determined using the AUC of the model produced against the unseen testing data. This result (AUC) is sent back to the optimiser and the partition sizes are changed accordingly to ensure the rough set produces a better model, i.e. a model with a higher classification accuracy.

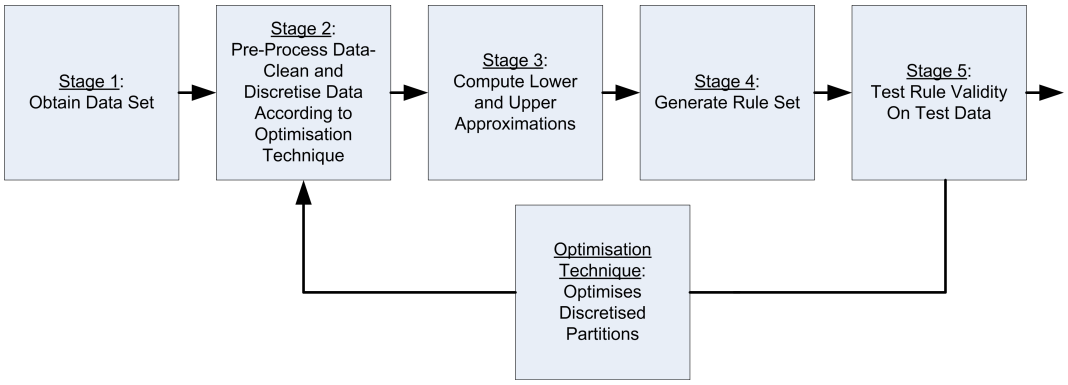

Figure 3.1: Block diagram of the sequence of events in creating a rough set model

3.2 Discretisation Criteria

Discretisation can be performed using different criteria. These criteria will be explained and the criteria used in this research will be stated.

- *Local versus Global methods*: *Global* methods work over an N-dimensional real space, where each attribute value set is partitioned into intervals independent of the other attributes. *Local* methods on the other hand produce partitions that are applied to localised regions of the object space [51]. In other words, methods that process the entire attribute value set are called *global*, global discretisation performs globally through the entire data set prior to proceeding with induction, an example of this method would be binning [5, 52, 53]. Methods that work on one attribute at a time are called *local*, local discretisation performs locally on partial regions during induction [5].

- *Static versus Dynamic methods*: Many discretisation methods require the maximum number intervals, k, to produce when discretising an attribute. *Static* methods such as binning, perform one discretisation pass of the data for each attribute and determine the value of k independent of the other attributes. In *static* discretisation the continuous features are discretised prior to the classification task. *Dynamic* methods conduct a search through the space of possible k values for all attributes simultaneously, thereby determining interdependencies in attribute discretisation [51, 53]. Dougherty *et al* state there is no clear

advantage of either method [53].

- *Supervised versus Unsupervised methods*: Methods that utilise the decision attribute of objects when performing discretisation are referred to as *supervised discretisation methods*. In contrast, methods that do not make use of decision attributes are called *unsupervised discretisation methods* [51].

The methods implemented in this research are global, dynamic and supervised. Binning as will be discussed below, uses global, static, and unsupervised learning methods. The two non-optimised, well known techniques of equal-width-bin, and equal-frequency-bin partitioning will be discussed, after which an explanation of the four optimisation techniques used will be given.

3.3 Equal-Width-Bin and Equal-Frequency-Bin Partitioning

While an investigation into the number of discretised partitions is a promising avenue of research, several studies have been conducted and for the purpose of this paper, the input data into the rough set is discretised into four partitions. Four partitions are chosen as they offer a good balance between classification accuracy produced and computational time taken to produce the optimal partition sizes. As mentioned in Chapter 1, there are several approaches available to discretise the continuous input data. Two approaches that are considered merely for the purpose of comparison against the newly proposed method in this research are, equal-width-bin (EWB) and equal-frequency-bin (EFB) partitioning.

3.3.1 Equal-Width-Bin Partitioning

EWB partitioning divides the range of observed values of an attribute into k equal sized bins. In this research, k is taken as four. One notable problem of such a method

is it is vulnerable to outliers that may drastically skew the data range. This problem was eliminated during the pre-processing step of cleaning the data as explained in Chapter 2. The way which the data was discretised using EWB partitioning is as follows:

- Determine the largest and smallest value for each attribute, these can be denoted as L and S.
- The width of each interval, W, can be calculated as follows:

$$W = \frac{L - S}{4} \tag{3.1}$$

- The interval boundaries can determined as: $S + W, S + 2W, S + 3W$. These boundaries can be determined for any amount of intervals k, up to the $S + (k - 1)W$ term.

3.3.2 Equal-Frequency-Bin Partitioning

EFB partitioning sorts the values of each attribute in ascending order and divides them into k bins where (given m instances) each bin contains $\frac{m}{k}$ adjacent values. In most instances there are most probably duplicated values. This is performed as follows:

- Sort the values of each attribute a, (i.e. $v_1^a, v_2^a, ..., v_m^a$) into four intervals (m being the number of instances)
- Each interval now contains:

$$\lambda = \frac{m}{4} \textit{sequential values.} \tag{3.2}$$

- The cut points are computed by $c_i = \frac{v_{i\lambda} + v_{i\lambda+1}}{2}$ for i = 1, 2, 3. The cut point for k intervals can be computed for i = 1, ..., $k - 1$.

The results obtained using the above two methods, as well as of those using the proposed optimisation methods will be given in Chapters 4 and 5.

3.4 Genetic Algorithm

Genetic algorithms (GAs) are population based search methods. GAs are popular and widely used due to their ease of implementation, intuitiveness and their ability to solve highly nonlinear optimisation problems. A GA is a stochastic search procedure for combinatorial optimisation problems based on the mechanism of natural selection [54]. Genetic algorithms are a particular class of evolutionary algorithms that use techniques inspired by evolutionary biology such as inheritance, mutation, selection and crossover. The fitness/evaluation function is the only part of the GA that has any knowledge about the problem. The fitness function tries to maximise the AUC. The GA represents the design variables of each individual design with binary strings of 0's and 1's, these are referred to as chromosomes. The variables are limited to the predefined upper and lower bound values. These limits are coded into the GA encoding feature. They perform the task of optimisation, the GA employs three main operators to propagate its population from one generation to the next.

The first operator is *selection*. During each successive generation, a proportion of the population is selected to breed a new generation. This is done on the basis of "survival of the fittest". The fitness of each solution is evaluated using a fitness function, and it is this function that maximises the AUC. After each generation, the AUC is evaluated and sent to the rough set for evaluation. This process continues until the termination criteria is reached. Several selection functions are available and in this paper, roulette wheel selection is used. Roulette wheel selection is a technique whereby members of the population of chromosomes are chosen in a way that is proportional to their fitness. The better the fitness of the chromosome, the greater the chance it will be selected, however it is not guaranteed that the fittest member goes to the next generation. Other selection functions that are available include tournament selection, normal geometric selection and ranking selection.

The second operator is *crossover* which mimics reproduction or mating in biological populations. The crossover technique used is uniform crossover; uniform crossover first generates a random crossover mask (a binary string with the same size of chromosomes) and then exchanges relative genes between parents according to the mask. The parity of each bit in the mask determines, for each corresponding bit in an offspring, which parent it will receive that bit from [55]. Other crossover functions that can be used include; single-point and two-point crossover as well as simple, arithmetic and heuristic crossover.

The third operator used is *mutation*, this is whereby an arbitrary bit in a generic sequence or string will be mutated or changed. This is analogous to biological mutation. The reason for implementing this operator is to promote diversity from one generation of chromosomes to another, this prevents the GA from getting stuck at a local optimum but rather a global optimum. Boundary mutation is chosen, this is whereby a variable is randomly selected and is set to either the upper or lower bound depending on a randomly generated uniform number. It must be noted that the chosen operators are case specific, dependent on the nature of the optimisation, different operators would have to be explored. Other mutation functions are uniform, non-uniform and multi-non-uniform. If the individuals (parents) were represented in binary format, binary mutation could have been used.

There was no significant difference between all the above mentioned functions for each operator, but alternatives were mentioned for the purpose of completeness. The pseudo-code algorithm for genetic algorithm is given on the next page

Genetic Algorithm Pseudo Code:

- *Initialise* a population P of chromosomes at random.
- *Evaluate* the *Fitness* of each chromosome (individual) h in the population P.
- While *Number Of Generations* $<$ *Termination Number of Generations (100)* do *Create a new population* P_s

  1. *Selection*: The probability of selecting a chromosome h_i from the population P is represented by $Pr(h_i)$. This probability is defined as:

  $$Pr(h_i) = \frac{Fitness_i}{\sum_{j=1}^{PopSize} Fitness_j} \tag{3.3}$$

  Add this chromosome from population P to population P_s.

  2. *Crossover*: Probabilistically obtain two parents $\langle h1, h2 \rangle$ from the population P according to $Pr(h_i)$. Create new offspring by copying the corresponding gene from either parent according to a randomly generated crossover mask. Add all new offspring to the population P_s.

  3. *Mutation*: Let $\overline{X}$ and $\overline{Y}$ be two *m-dimensional* row vectors denoting individuals. For real $\overline{X}$ and $\overline{Y}$, the lower and upper bounds for each variable i in population P_s are defined as a_i and b_i respectively. Boundary mutation randomly selects one variable j, and sets it to either its lower or upper bound, where $r = U(0, 1)$:

  $$x_i^{'} = \begin{cases} a_i, & \text{if } i = j, r < 0.5 \\ b_i, & \text{if } i = j, r \geq 0.5 \\ x_i, & \text{otherwise} \end{cases}$$

  4. Insert new members into the population P, i.e. update $P \leftarrow P_s$
  5. Evaluate the new members h in P, i.e. compute fitness

- Return the best chromosome as the solution

An initial population of 20 individuals is chosen. As mentioned in order to prevent premature convergence to a local minima, the mutation diversification mechanism is implemented. Other diversification mechanisms such as elitism can also be implemented in an attempt to improve the accuracy. One of the advantages of GAs is that they refine a single solution as they search for optima in a multi-dimensional landscape, genetic algorithms operate on entire populations of candidate solutions in parallel. Their parallel nature is their main strength, it is through this that GAs are more likely to get stuck at a global optimum rather than a local optimum. It is also due to its parallel nature that GAs are not that sensitive to initial conditions, which implies that the GAs time to convergence is rather predictable. The drawback of using GAs is that they are much slower than most traditional methods i.e. a good initial guess will allow traditional optimisation techniques to converge quickly towards the solution, whereas GAs will waste time testing the fitness of all suboptimal solutions. Although GAs will always approximate a solution but it must be noted, due to their stochastic nature, this approximation is only an estimate, whereas with traditional methods, if they can find an optimum, they will find it exactly [22]. The addition of an elitist strategy could further improve results. The flow chart of the genetic algorithm optimisation is shown in Appendix B, Figure B.1.

3.5 Particle Swarm Optimisation

Particle swarm optimisation (PSO) is not only a tool for optimisation, but also a tool for representing sociocognition of human and artificial agents, based on principles of social psychology. Eberhart and Kennedy developed PSO based on the analogy of flocks of birds and schools of fish [56].

Social optimisation can be thought of as the process that occurs when a person that is trying to solve a problem, interacts with another person. The information processed in the individuals thinking is referred to as cognition. If one person's cognition does not follow from the other or that one follows from the converse of the other, they are described as dissonant. Dissonant cognitions produce an aversive consequence that

the individuals will try to reduce. This can be done by reducing both cognitions [57, 58]. The particle swarm simulates this kind of social optimisation.

The way in which PSO works is as follows; PSO initialises by creating a random population of solutions (particles), these particles are placed in the parameter space of some problem or function. Each particle evaluates the fitness at its current location and the fittest or best solution is noted (often known as *pbest*). Each particle has its own set of coordinates, and these coordinates are associated with the fittest solution found so far. The movement of each particle is determined by combining some aspect of the history of its own fitness values with those of other particles in the neighbourhood of that particle. The best solution found in the neighbourhood of particular solutions or particles (often known as *lbest*) is also noted. This information is an analogy of knowledge of how the other particles around a particular particle have performed. Each particle tries to modify it position using the following information [59, 23]:

- its current position $p(x, y)$
- its current velocity $v(x, y)$
- the distance between its current position $p(x, y)$ and *pbest*.
- the distance between its current position $p(x, y)$ and *lbest*.

This modification can be represented by the concept of velocity. Velocity of a particle can be modified using the following equation [59]:

$$v_i^{k+1} = wv_i^k + c_1 rand_1 \times (pbest_i - s_i^k) + c_2 rand_2 \times (lbest - s_i^k) \tag{3.4}$$

where:

v_i^k = Velocity of particle i at iteration k

w = Weighting function

c_j = Weighting factor

$rand$ = Random number between 0 and 1

s_i^k = Current position of particle i at iteration k

$pbest_i$ = $pbest$ of particle i

$lbest$ = $lbest$ of the group

The following weighting function is used in Equation 3.4:

$$w = w_{max} - \frac{w_{max} - w_{min}}{iter_{max}} \times iter \tag{3.5}$$

where:

w_{max} = Initial weight

w_{min} = Final weight

$iter_{max}$ = Maximum iteration number

$iter$ = Current iteration number

Using the above equation, a velocity that gradually moves the current searching point close to *pbest* and *lbest* can be calculated. The current position of the searching point can be calculated as follows:

$$s_i^{k+1} = s_i^k + v_i^{k+1} \tag{3.6}$$

The next iteration takes place after all particles have been moved. PSO was implemented with the maximum number of generations of 30, and the initial number of particles also set to 30. PSO has several similarities with other optimisation techniques, these include a randomly generated initial population and a search for an optima through a series of generations, but unlike GA for example, PSO does not have evolutionary operators such as mutation and crossover, instead it has particles that travel (or swarm) through the solution space looking for an optima. The flow chart of PSO is given in Appendix B, Figure B.2.

3.6 Hill Climbing

Hill climbing is an optimisation technique that belongs to the group of local search algorithms, meaning the algorithm moves from solution to solution in the search space until an optimal solution is found. The algorithm tries to maximise the fitness function (accuracy) by iteratively comparing two solutions, it adopts the best solution and continues the comparison (i.e. move further up the hill). This iteration terminates when there are no better solutions on either side of the current solution (i.e. it has reached the peak). There are several variants or methods of hill climbing, the first and most basic form is simple hill climbing; in this method the first closest node is chosen for evaluation. A second variant is called steepest ascent hill climbing, in this method all successors are compared and the closest to the solution is chosen. Other variants that can be investigated include next-ascent hill climbing and zero-temperature Monte Carlo hill climbing [60]. In this paper, steepest ascent hill climbing is implemented, a major disadvantage of both simple and steepest ascent hill climbing is that it only finds the local optimum. This can be visualised as shown in Figure 3.2 [61].

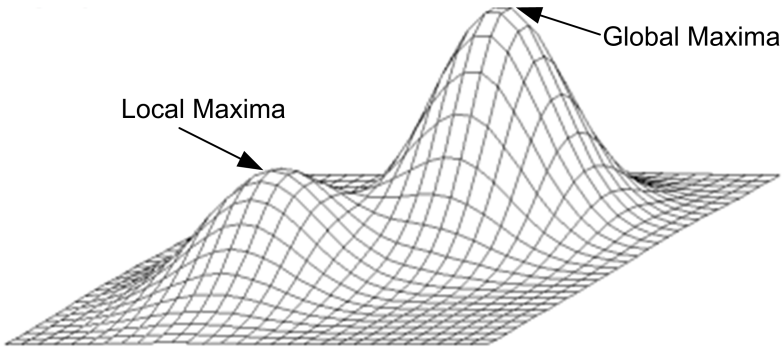

Figure 3.2: Figure illustrating a local maxima versus a global maxima

The algorithm may move to a point at the top of a local maxima, searching around it, the algorithm cannot see a better solution, yet, looking at the entire search space there is a better solution at the global maxima. This local maxima will be returned as the best solution and if this occurs a sub-optimal solution has been found. There are other local search algorithms that overcome this and these include: stochastic hill climbing, random walks and iterated hill-climbing. The advantages

of hill climbing are that it requires no memory as there is no backtracking and it is trivial to implement.

The pseudo code algorithm for hill climbing is given below.

Hill Climbing Pseudo Code:

- Set *initial* conditions
  - *currentNode* = *startNode* = *initial configuration* = S
  - Compute $S_{Fitness}$, i.e. *initial configuration fitness*
- Repeat:
  - Compute a neighbouring configuration S' by local transformation from the current point in the search space to the next point
  - Compute fitness of S' ($S'_{Fitness}$)
  - If $S'_{Fitness} > S_{Fitness}$ then $S = S'$
- Until *maxima is reached*

The flow chart illustrating the implementation of hill climbing is given in Appendix B, Figure B.3.

The hill climbing (HC) algorithm is chosen to have an initial 20 starting points. As with all the other optimisation techniques, hill climbing uses the AUC as the objective/fitness function, and because it is a "greedy" algorithm, other methods or variations of hill climbing should be investigated. A "greedy" algorithm is an algorithm that looks at the next solution, and if the next solution is a better solution (fitter), this algorithm will take on the position of the next solution. This is disadvantageous as it does not look at the entire search space, and if the next solution (as it is climbing the hill) is worse than the present one, it will return stating it has found the best solution, this may not be the case as explained above referring to local and global maxima.

A variation of steepest ascent hill climbing is random-restart hill climbing (RRHC). RRHC is an algorithm built on top of the hill climbing algorithm, what it does is that it runs an outer loop over HC, where each step chooses a random initial point (x_0) to start the HC. The best initial point is stored (x_{best}) and if a new run of HC produces a better initial point than the stored one (if x_0 is better than x_{best}), then the stored one gets replaced ($x_{best} := x_0$). There are several other variations of HC but these methods are not investigated in this research.

3.7 Simulated Annealing

Simulated annealing is an algorithm that locates a good approximation to the global optimum of a given function. It originated as a generalisation to the Monte Carlo method and relies on the Metropolis algorithm[1]. As is the case with GA, PSO and HC, SA continuously updates the solution until a termination criteria is reached. SA is a well established stochastic technique originally developed to model the natural process of crystallisation and later adopted as an optimisation technique [26]. The SA algorithm replaces a current solution with a "nearby" random solution with a probability that depends on the difference between the corresponding function values and the temperature (T). T decreases throughout the process, so as T starts approaching zero, there is less random changes in the solution. As with the case of greedy search methods, SA keeps moving towards the best solution, except that it has the advantage of reversal in fitness. That means it can move to a solution with worse fitness than it currently has, but the advantage of that is that it ensures the solution is not found at a local maxima, but rather a global maxima. This is the major advantage that SA has over most other methods, but once again its drawback is its computational time, the SA algorithm will find the global optimum if specified but it can approach infinite time in doing so. The probability of accepting the reversal is given by Botlzman's equation [62]:

$$P(\Delta E) \propto e^{-\frac{\Delta E}{T}} \tag{3.7}$$

[1] See [26] for more information regarding the Metropolis algorithm

Where ΔE is the difference in energy (fitness) between the old and new states, and T is the temperature of the system. The rate at which temperature decreases depends on the cooling schedule chosen. The following cooling model is used [62]:

$$T(k) = \frac{T(k-1)}{1+\sigma} \tag{3.8}$$

Where $T(k)$ is the current temperature, $T(k-1)$ is the previous temperature, and σ dictates the cooling rate. Other advantages of SA are that it can easily be tuned, and similarly like the GA, SA can deal with highly nonlinear models with many constraints. It must be noted, the precision of the numbers used in the implementation of SA can have a significant effect on the outcome. As with GAs, SA is also amenable to parallel implementation, and a method to improve the computational time is to implement either very fast simulated re-annealing (VFSR) or adaptive simulated annealing (ASA) [63]. The pseudo code algorithm for SA is given below:

Simulated Annealing Pseudo Code:

$//s$: current state; s' : state of neighbour of s

$//s_{best}$: current best state

$//E(s)$: is the function to be optimised (internal energy)

$//e = E(s)$: current energy

$//e_{best}$: current best energy

$//e' = E(s')$: energy of neighbour of s

$//k$: iteration number, initially set to 0

$//T$: temperature of the systems

$//P(e, e', T)$: function of the energies of e and e'

The algorithm of the simulated annealing pseudo code is continued on the next page.

- While $k < k_{max}$ and $e > e_{max}$ Do
  - $s' := s$
  - $e' := E(s)$
  - If $e' < e_{best}$ Then
    $s_{best} := s'$; $e_{best} := e'$
  - If $random() < P(e, e', T(\frac{k}{k_{max}}))$ Then
    $s := s'$; $e = e'$
  - $k := k + 1$
- Return s_{best}

This process can be visualised using the flow chart as shown in Appendix B, Figure B.4.

Using the optimisation techniques given in this chapter, and using the description of generating rough sets as given in Chapter 2, this newly proposed method is formed. The method as shown in Figure 3.1 and described in Appendix A, Algorithm 1 is generated, and in order to test this method, it is executed on two separate data sets, namely, the human immunodeficiency virus (HIV) data set and the militarised interstate dispute (MID) data set, from which rules are extracted, and classification accuracies obtained. An explanation of these data sets as well as the results of the experimental tests are given in Chapters 4 and 5.

Chapter 4

Experimental Investigation I : Modelling HIV

4.1 Introduction

In the last 20 years, over 60 million people have been infected with HIV (human immunodeficiency virus), and of those cases, 95% are in developing countries [64]. In 2006 alone, an estimated 39.5 million people around the world were living with HIV, with 27.5 million of those people living in Sub-Saharan Africa. During this year, AIDS (acquired immune deficiency syndrome) claimed an estimated 2.9 million lives [65]. HIV has been identified as the causative agent of AIDS. The effect of AIDS is not only detrimental to the individual infected but has a devastating effect on the economic, social, security and demographic levels of a country. Because AIDS is killing people in the prime of their working and parenting lives, it represents a grave threat to economic development. In the worst affected countries, the epidemic has already reversed many of the development achievements of the past generation [65]. AIDS has a large negative impact on the social and security levels of a country. Social levels drop as the health and educational development, that is supposed to benefit poor people, is impeded as well as the average life expectancy drops. It is estimated that by 2010 the number of orphans will be double that of 2006 [65].

AIDS and HIV has been the subject of extensive research, literature, and political debate. Although a wide array of viewpoints have stated on the subject, one consistent presupposition is the biological disease exists as an ontological reality [66]. Early studies on HIV/AIDS focused on the individual characteristics and behaviours in determining HIV risk, Fee and Krieger refer to this as *biomedical individualism* [67]. But it has been determined that the study of the distribution of health outcomes and their social determinants is just as important, this is referred to as *social epidemiology* [68]. Therefore risk factor epidemiology examines the individual demographic and social characteristics in an attempt to determine the factors that expose an individual to the risk of obtaining HIV/AIDS. This study uses individual characteristics as well as social and demographic factors in determining the risk of HIV.

It is thus evident from above that the analysis of HIV is of utmost importance. By correctly forecasting HIV, the causal interpretations of a patients being seropositive (infected by HIV) is made much easier. Previously, computational intelligence techniques have been used extensively to analyse HIV. Leke *et al* have used autoencoder network classifiers, inverse neural networks, as well as conventional feedforward neural networks to analyse HIV on the same data set used in this research [28, 29, 30]. They used the inverse neural network for adaptive control of HIV status to understand how the demographic factors affect the risk of HIV infections [30]. Lee *et al* used neural networks to assess HIV/AIDS related health performance [69]. Lumini and Nanni used machine learning for HIV-1 protease cleavage site prediction [70].

Although an accuracy of 84% is achieved when using the autoencoder method of Leke *et al* [28], it is disadvantageous due to its "black box" nature, this also applies to the other mentioned neural network techniques. Neural networks offer accuracy over analysis of data, but in the case of analysing HIV data, it can be argued that interpretability of the data is of more importance than just prediction. It is due to this fact that rough set theory (RST) is proposed to forecast and interpret the causal effects of HIV. RST produces linguistic rules that govern the predictions that they make and it is these rules that provide decision-makers and policy-makers with more accurate information to allow them to implement better health care systems

and HIV prevention schemes.

4.2 HIV Data

The data set used in this research was obtained from the South African antenatal sero-prevalence survey of 2001. The data was obtained through questionnaires completed by pregnant women attending selected public clinics and was conducted concurrently across all nine provinces in South Africa. The sentinel population for the study only included pregnant women attending an antenatal clinic for the first time during their current pregnancy. The choice of the first antenatal visit is made to minimise the chance for one woman attending two clinics and being included in the study more than once [45]. The probability proportion to size (PPS) sampling method was used to determine the sample size of the 2001 antenatal survey [45]. PPS is a sampling technique used (in surveys) to ensure the probability of selecting a sample unit (attribute) is proportional to the size of the population. The collected data used to study the conditions associated with HIV consists of six different attributes. The six demographic variables considered are: *race, age of mother, education, gravidity, parity* and, *age of father*. For any set of attributes considered, there is an associated outcome or decision. The outcome is either HIV positive or negative.

4.2.1 Data

As previously mentioned, HIV infections rate are described using six demographic variables. These variables are further described below.

Race: This variable is a nominal polytomous variable as there are four values associated with it but these values have no order. The four possible variables are *African*, *White*, *Coloured* and, *Asian*.

Age of Mother: This variable consists of continuous integers, as the values range

from 15 years old to 49 years old. The value taken for the age of mother was taken at the time of conducting the survey.

Education: This variable also contains continuous integers, with the values ranging from 0 to 13. The value of 0 indicates that no education has been undertaken by the mother, while an education level of 13 represents tertiary education.

Gravidity: Gravidity is given as continuous integers ranging from 0 to 11. Gravidity is the number of pregnancies, complete or incomplete, experienced by the mother.

Parity: Parity is also given as continuous integers with values ranging from 0 to 11. Parity is the number of times the mother has given birth. Multiple births are however counted as one. Both parity and gravidity are important factors as they show the reproductive activity as well as the reproductive health state of the women taking part in the test.

Age of Father: As with age of mother, this variable consists of continuous integers, the values range from 15 years old to 49 years old.

HIV Status: This variable is given as the outcome or decision value. This value is dichotomous as the variables or values are given in binary, with a 0 representing HIV negative while a 1 represents HIV positive.

A summary of the data is given in Table 4.1

Table 4.1: Summary of the HIV data set variables

Variable	**Description**	**Value Range**
x_1	Race	{*African, White, Coloured, Asian*}
x_2	Age of Mother	[15, 49]
x_3	Education	[0, 13]
x_4	Gravidity	[0, 11]
x_5	Parity	[0, 11]
x_6	Age of Father	[15, 49]
y_1	HIV Status	0 or 1

4.2.2 Cleaning the Data

As mentioned the data was collected from surveys, and like any data that is in raw form, there are several steps that need to be undertaken to ensure the data is transformed into a usable form. This usable form includes the removal of any outliers such as missing and/or incorrect data. The data needs to be cleaned and removed of irregularities before any processing can be performed on it. The first irregularity would be the case of missing data. This could be due to the fact that surveyees may have omitted certain information, it could also be attributed to the errors being made when the data was entered onto the computer. Such cases are removed from the data set. The second irregularity would be information that is false. Such an instance would be if gravidity was zero and parity was at least one. Gravidity is defined as the number of times that a woman has been pregnant, and parity is defined as the number of times that she has given birth. Therefore it is impossible for a woman to have given birth, given that she has not been pregnant, such cases are removed from the data set. As mentioned earlier, multiple births are still indicated with a parity of one, therefore if parity is greater than gravidity, that particular case is removed from the data set. Only 12945 cases remained from the initial total of 13087. Of the 12945 cases, the data sets were balanced and then split into training and testing data using the ratio of 70% to 30% respectively. The rough set model was created using the training data, while the performance of the rough set model was validated using the testing data.

Once the data had been pre-processed as described above, it was used to create a rough set model, and the model was validated using testing data. The results obtained by applying the newly proposed rough set approach to human immunodeficiency virus (HIV) data are given below.

4.3 Results Obtained on HIV data

A confusion matrix is a visualisation tool represented in the form of a table/matrix. Each column of the matrix represents the instances in a predicted class, while each row represents the instances in an actual class. A confusion matrix is used to see if the classification of a certain class is being confused with that of another, i.e. it illustrates the mislabelling of classes with another. The confusion matrix is represented in terms of actual positives and negatives against predicted positives and negatives. Table 4.2 illustrates how the actual and predicted classification is shown.

Table 4.2: Representation of a Confusion Matrix

	Predicted Positive	Predicted Negative
Actual Positive	a	b
Actual Negative	c	d

Where:

- a is the number of **correct** predictions that an instance is **positive**
- b is the number of **incorrect** of predictions that an instance **negative**
- c is the number of **incorrect** predictions that an instance is **positive**
- d is the number of **correct** predictions that an instance is **negative**

The four different optimisation techniques results will be plotted against each other using a ROC curve. These techniques will also be individually plotted against that of EWB and EFB partitioning to determine how the newly proposed method performs against that of the more well known static methods.

The confusion matrix for the EWB partitioning is given in Table 4.3 below. Table 4.4 gives the confusion matrix for EFB partitioning. The confusion matrices for the optimisation methods are given in section 4.3.1.

Table 4.3: Confusion matrix for EWB discretisation on HIV data

Equal-Width-Bin Partitioning		
	Predicted Positive	Predicted Negative
Actual Positive	800	43
Actual Negative	630	185
59.41%		

Table 4.4: Confusion matrix for EFB discretisation on HIV data

Equal-Frequency-Bin Partitioning		
	Predicted Positive	Predicted Negative
Actual Positive	485	378
Actual Negative	374	423
54.70%		

4.3.1 Results of Optimisation Techniques

The results obtained for the four optimisation techniques as described in Chapter 3 are given in this section. The results of equal-width-bin and equal-frequency-bin partitioning are compared to those of the various optimisation approaches. The confusion matrix of each method will be given in each respective section along with the ROC curve of the particular optimisation technique plotted against that of EWB and EFB. Table 4.9 summarises the findings.

Genetic Algorithm

The genetic algorithm was run with a roulette wheel selection function, a uniform crossover operator, a boundary mutation operator, an initial population of 20 individuals and the termination function of 100 generations. The confusion matrix for the GA is given in Table 4.5.

Table 4.5: Confusion matrix of the GA discretisation on HIV data

Genetic Algorithm		
	Predicted Positive	Predicted Negative
Actual Positive	677	135
Actual Negative	496	327
61.41%		

Particle Swarm Optimisation

Particle swarm optimisation was implemented with the maximum number of generations of 30, and the initial number of particles also set to 30. The confusion matrix of the results of PSO is given in Table 4.6.

Table 4.6: Confusion matrix of a the PSO discretisation on HIV data

Particle Swarm Optimisation		
	Predicted Positive	Predicted Negative
Actual Positive	578	239
Actual Negative	380	445
62.30%		

Hill Climbing

Steepest ascent hill climbing was implemented with 20 initial starting points. The theory of hill climbing (HC) as with the other methods is given in Chapter 3. The hill climbing confusion matrix is given in Table 4.7.

Table 4.7: Confusion matrix of a the HC discretisation on HIV data

Hill Climbing		
	Predicted Positive	Predicted Negative
Actual Positive	652	162
Actual Negative	438	377
63.17%		

Simulated Annealing

SA has the cooling model as given in Equation 3.8, it was run with a random generator with the bounds of the maximum and minimum input values, an initial temperature of 1, a stopping temperature of $1e^{-8}$, a maximum number of consecutive rejection of 200 and a maximum number of successes within one temperature set to 10. The confusion matrix for the results obtained using SA is given in Table 4.8.

Table 4.8: Confusion matrix of a the SA discretisation on HIV data

Simulated Annealing		
	Predicted Positive	Predicted Negative
Actual Positive	665	141
Actual Negative	466	350
62.58%		

4.3.2 Comparison of Methods on HIV data

The ROC curves of the various optimisation techniques plotted against EWB and EFB are given below. Figure 4.5 gives the ROC curve of the four optimised techniques plotted against each other. As stated in Chapter 2, the performance of the rough set is calculated as the AUC of the ROC curve. The AUC can be interpreted as the probability that when we randomly pick one positive and one negative example, the classifier will assign a higher score to the positive example than to the negative. The higher the AUC, the better the classification is.

Genetic Algorithm ROC Curve

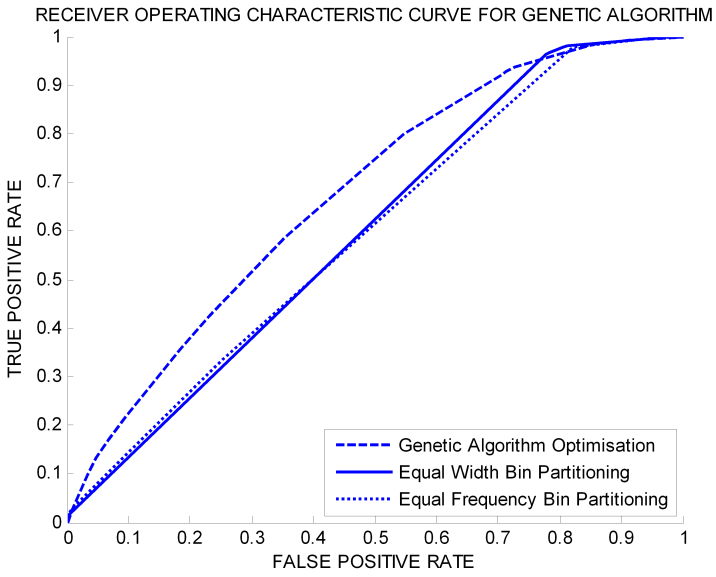

Figure 4.1: Receiver operating characteristic (ROC) curves for GA, EWB and EFB on HIV data

Particle Swarm Optimisation ROC Curve

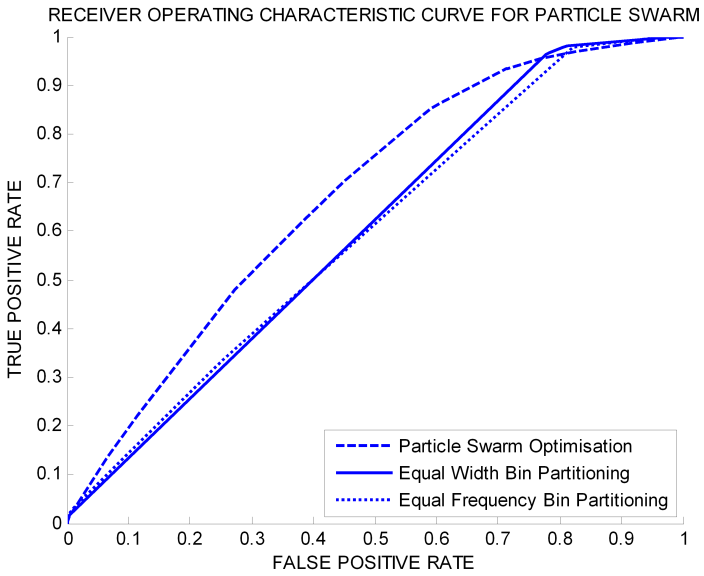

Figure 4.2: Receiver operating characteristic (ROC) curves for PSO, EWB and EFB on HIV data

Hill Climbing ROC Curve

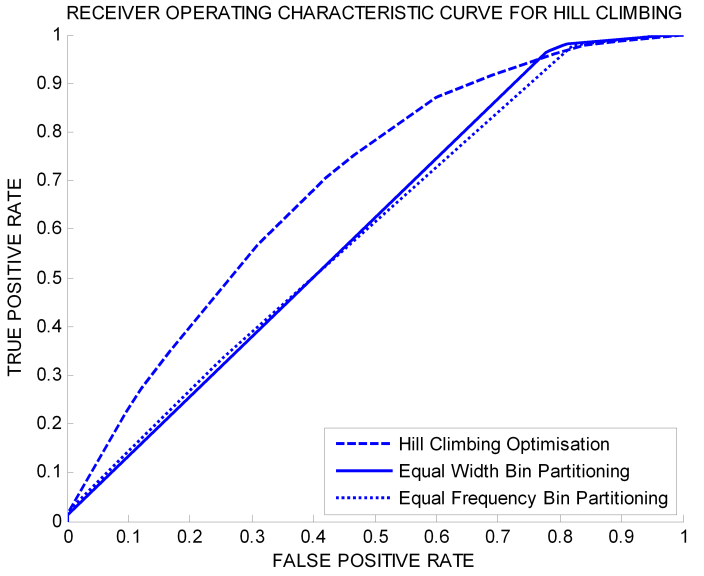

Figure 4.3: Receiver operating characteristic (ROC) curves for HC, EWB and EFB on HIV data

Simulated Annealing ROC Curve

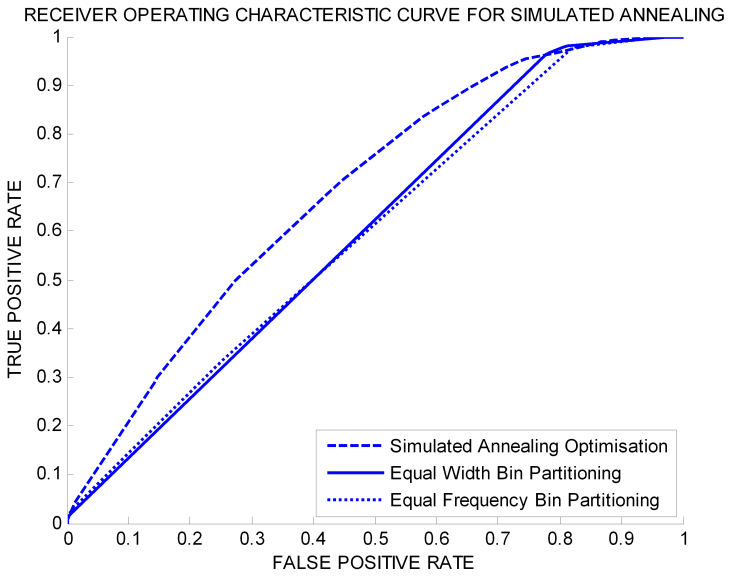

Figure 4.4: Receiver operating characteristic (ROC) curves for SA, EWB and EFB on HIV data

It is clearly visible from the previous four ROC curves that each optimisation technique has a better classification accuracy than both that of EWB and EFB partitioning. Figure 4.5 illustrates the performance of all four optimisation techniques against each other. It is clearly shown in Figure 4.5 that there is no significant difference between the various optimisation techniques. There is narrow band in which all the ROC curves fall. This narrow band indicates that no one particular optimisation technique is much better suited than the others. Table 4.9 tabulates the results from which it can be seen that, albeit marginal, hill climbing produces the highest classification accuracy when using the AUC measure. The table displays the number of rules produced, the computational time (minutes) and the AUC. As can be seen from Table 4.9 the simulated annealing optimisation has the lowest computational time with respect to the optimised methods. From the table, it is evident that the computational time required for the static EWB and EFB partitioning is much less than that for the optimised approaches. Having said that, the higher forecasting accuracy obtained using optimised methods will results in more concrete evidence from which policies can be generated. Less rules are also generated from

the optimised methods approach and it is from these extracted rules from which the causal interpretations are then formulated by a linguistic approximation.

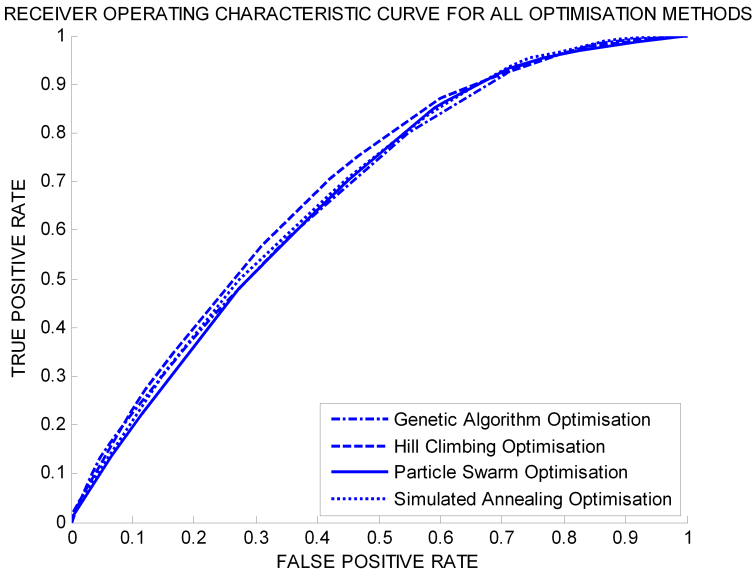

Figure 4.5: Receiver operating characteristic (ROC) curves for GA, PSO, HC and SA on HIV data

Table 4.9: Results obtained for the four optimisation and two static discretisation methods

	Number of Rules	Comp. Time	AUC
Genetic Algorithm	172	3726	0.6748
Particle Swarm Optimisation	146	1690	0.6718
Hill Climbing	209	12624	0.6902
Simulated Annealing	197	1159	0.6802
Equal Width Bin	231	2	0.5952
Equal Frequency Bin	307	2	0.5986

Once the highest classification accuracy was achieved for each method, the partition sizes were noted. Figure 4.6 shows the optimal cut points and partition sizes for the six variables for the GA method. The other methods discretisation points can be represented in the same way but is not shown. These cut points as shown in the figure indicate which values should fall into which range. As for EWB and EFB partitioning, the partitions are split as described in Chapter 3.

The results shown in this chapter indicate that the newly proposed method of using optimisation techniques to granulise/discretise rough set partitions is feasible and it

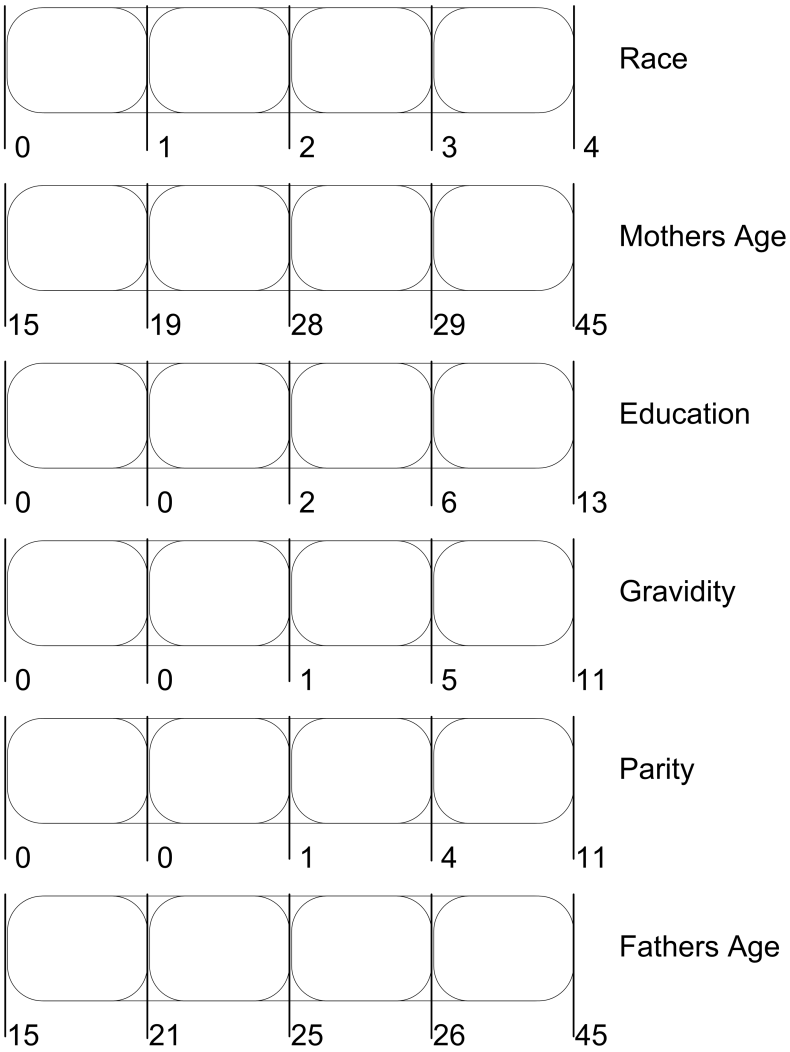

Figure 4.6: Discretisation points for the various attributes using the GA method on HIV data

also produces higher forecasting/classification accuracies than that of the previously used static methods of equal-width-bin and equal-frequency-bin partitioning. Of the four optimisation techniques, it can be stated that no one particular technique is superior to any other. There are marginal difference in the accuracies produced (using AUC), but in all cases easy-to-interpret, linguistic rules are generated. A case generated from the lower and upper approximation is represented as shown on the next page.

Lower Approximation Rules

- **If** Race = African **and** Mothers Age = 23 **and** Education = 4 **and** Gravidity = 2 **and** Parity = 1 **and** Fathers Age = 20 **Then** HIV = Most Probably Positive
- **If** Race = Asian **and** Mothers Age = 30 **and** Education = 13 **and** Gravidity = 1 **and** Parity = 1 **and** Fathers Age = 33 **Then** HIV = Most Probably Negative

Upper Approximation Rules

- **If** Race = Coloured **and** Mothers Age = 33 **and** Education = 7 **and** Gravidity = 1 **and** Parity = 1 **and** Fathers Age = 30 **Then** HIV = Positive with plausibility = 0.33333
- **If** Race = White **and** Mothers Age = 20 **and** Education = 5 **and** Gravidity = 2 **and** Parity = 1 **and** Fathers Age = 20 **Then** HIV = Positive with plausibility = 0.06666

The proposed method used in this research is further validated by applying it to another data set. The second data set used to test the method is the militarised interstate dispute (MID) data set. The results will be given in Chapter 5 in a similar form as that given in this chapter.

Chapter 5

Experimental Investigation II : Modelling MID

5.1 Introduction

Recent developments in the conflict literature has emphasised the importance of treating international conflicts as complex phenomena often displaying nonlinear and nonmonotonic patterns of interactions [71, 72]. Various methods have been implemented and there are still efforts underway to study interstate interactions or militarised interstate disputes. Militarised interstate disputes (MID) are defined as a set of interactions between or among states involving threats to use military force, displays of military force, or actual uses of military force [73]. In other words MID's are conflicts between states that do not involve a full scale war. The various states involved in the analysis of conflict are referred to dyads, therefore the variables associated with the analysis of conflicts are referred to as *dyadic variables*.

Beck *et al.* [33] state that no legitimate statistical model has managed to forecast an international conflict with a probability of greater than 0.5. Statistical models have also produced results that are conflicting; Thompson and Tucker state that for democratic dyads the absence of war and the low probability of serious disputes are restricted, in other words countries or states that are democratic have a reduced

chance of being involved in conflict [74]. However, Mansfield and Snyder argue that democratic states or countries may eventually become peaceful but in their early years they are prone to conflict or war involvement [75]. Lagazio *et al* state that the failure of statistical models is resultant of interstate variables that are related to MIDs are non-linear, highly interdependent and context dependent [32].

Thus the need for modelling and forecasting conflict using artificial techniques has become inevitable. Even though political scientists have long eschewed forecasting of conflicts in favour of causal interpretations, it is argued that causal theories are considered harder to verify than forecast, yet accurate forecasts can be used to verify claims about the causal structure, or at least in part [33]. Accurate forecasting will aid in the interpretation of the conflict.

Due to the shift from statistical methods to AI techniques, several AI methods have been applied to conflict management. Neural networks using Gaussian approximation have been used to model the complex interrelationships between the dyadic parameters and the MID's [7, 32, 33]. Other methods that have been implemented include neuro-fuzzy, regression, and support vector machines (SVM). Similarly, game theory has been used in conflict analysis [20]. Habtemariam and Marwala introduced support vector machines (SVM) to the study of conflict management [8], although SVM's offer high forecasting accuracy, they lack the ability to perform a causal analysis on the results achieved. Marwala and Lagazio proposed the use of Automatic relevance detection (ARD) to incorporate transparency into the prediction of neural networks. The results obtained allowed the ranking of importance of the 7 dyadic variables. In decreasing order the variables were ranked; *Democracy*, *Capability*, *Dependency*, *Allies*, *Contiguity*, *Distance* and *Major power* [36]. Offering a better balance between accuracy and interpretability, Tettey and Marwala proposed the use of neuro-fuzzy methods to control and analyze conflict [76].

5.2 MID Data

The MID data used in this study was obtained from the correlates of war project [46]. The correlates of war (COW) project is an academic study of the history of warfare. COW seeks to facilitate the collection, dissemination and use of accurate and reliable quantitative data in international relations [46]. It is argued however, that the COW project has tried to side-step the subjectivity problem by standardising their core concepts. Such standardisation by a research clique is often viewed and condemned by competitors as adding an ideological dimension that constrains or dogmatises research [77]. Although extensive data collection efforts have been made, a great deal of research is underway to ensure satisfactory and reliable conflict models are created. But by using databases such as those provided by COW, conflict causality is understood and the results obtained can then be specified as a tool that will contribute to decision making and policy formulation.

The collected data, referred to as *interstate variables* are used to study the interactions associated with militarised interstate disputes. The various *interstate* or *dyadic* variables of the data set consist of *kantian* and *realist* variables [72]. Kantian variables arise from the Kantian peace theory which was developed by Immanuel Kant [72, 78]. This theory states that by controlling variables such as democracy, interdependence and intergovernmental organizations, the risk of war can be reduced, such a theory is also known as the "liberal" theory [72]. The realist theory on the other hand encompasses a variety of theories and approaches, all of which share a belief that states are primarily motivated by the desire for military and economic power or security rather than ideals or ethics. Realists believe that overwhelming power will reduce the likelihood of conflict [72, 79].

The *realist* variables include *Allies*, *Contiguity*, *Major Powers*, *Distance* and *Capability*. The *kantian* variables include *Democracy* and *Dependency*. The above set of attributes have associated outcomes or concepts, the outcome is either peace or conflict. These variables will be explained in below.

5.2.1 Data

The seven dyadic variables as stated previously will be explained below and summarised in Table 5.1

Allies: This variable is dichotomous, consisting of a binary decision value. Allies refers to the measure of the military alliance between dyads. A value of 1 implies that there is an alliance of some kind, whether it be a mutual defence treaty or neutrality pact. A value of 0 implies that there is no alliance between the dyads.

Contiguity: This is also a dichotomous value, and it indicates whether the two countries or dyads share any common boundary or border. If they do share a border a value of 1 is assigned, and a value of 0 is assigned if they do not.

Major Powers: This dichotomous value is assigned to state if either of the two dyads are major powers or not. A country is considered a major power if it has substantial relative global destructive power based on a consensus of historians. If either dyad is a major power a value of 1 is assigned, or a 0 if not.

Distance: This continuous variable is simply the distance between the two countries. It is taken as the logarithmic distance (base 10) between the capital cities of the two countries. The values in the data set range from between 1.87 and 9.41.

Capability: Capability is a continuous variable and is the logarithm to base 10 ratio of the total population plus the number of people in urban areas plus industrial energy consumption plus iron and steel production plus number of military personnel in active duty plus military expenditure in dollars in the last 5 years measured on stronger country to weaker country.

Democracy: This variable consists of continuous values in the range [-10, 10] stating the democracy level of the relevant country. -10 would denote an extremely autocratic state whereas 10 denotes an extremely democratic state. The joint democracy level is then calculated as a minimum of both the states democracy scores as it is assumed the less democratic state plays a determinant role for an occurrence of a

conflict.

Dependency: Dependency is a continuous variable that measure the economic interdependence of the less economically dependent state. It is measured as the sum of the countries imports and exports with its partner divided by the gross domestic product (GDP) of the stronger country, in other words it is the minimum bilateral trade-to-GDP ratio between the two countries.

A summary of the described dyadic variables is given in Table 5.1.

Table 5.1: Summary of the MID data set variables

Variable	**Description**	**Value Range**
x_1	Allies	0 or 1
x_2	Contiguity	0 or 1
x_3	Major Powers	0 or 1
x_4	Distance	[1.84, 9.41]
x_5	Capability	[0, 9.49]
x_6	Democracy	[-10, 10]
x_7	Dependency	[0, 0.1719]
y_1	Outcome	0 or 1

5.2.2 Cleaning the Data

As is the case with the HIV data, any outliers if any, or missing attribute cases must be removed. The MID data used contains 27737 cases each consisting of seven dyadic attributes or variables and an associated dispute outcome. The data contains 26845 peace examples and 892 conflict examples therefore for accurate classification using AUC, the data set needs to be balanced. Once the data was balanced, the data set was used to generated two different sets, one training set and one testing set. This data was split on a ratio of 70% training data to 30% testing data. The training data was used to generate the rough set model while the testing data was used to assess out-of-sample accuracy.

5.3 Results Obtained on MID data

An important statement Beck *et al* make is, although political scientists are less likely to evaluate conflict models per se than, say economists, forecasting underlines all evaluations of such a model, and it is this information that will be of significant use [33]. The results using the four different optimisation techniques will be given. This includes the AUC along with their ROC curves as well as the respective confusion matrices. Each method will be plotted against that of EWB and EFB partitioning as well as all the methods will be plotted against each other.

5.3.1 Results of Optimisation Techniques

As is the case with HIV, the confusion matrices will be given for both EWB and EFB partitioning as well as for each optimisation technique. For the purpose of consistency, the chosen operators used for the optimisation techniques and the numerical values used to fine tune each method are maintained from the HIV experimental investigation. These values will not be stated again as they are given in the respective sections in Chapter 4. The confusion matrices are first given for the two static methods, and then for each optimisation method under their respective section.

Table 5.2: Confusion matrix for EWB discretisation on MID data

Equal-Width-Bin Partitioning		
	Predicted Positive	Predicted Negative
Actual Positive	181	79
Actual Negative	70	205
72.15%		

Table 5.3: Confusion matrix for EFB discretisation on MID data

Equal-Frequency-Bin Partitioning		
	Predicted Positive	Predicted Negative
Actual Positive	213	72
Actual Negative	79	171
71.77%		

Genetic Algorithm

Table 5.4: Confusion matrix of the GA discretisation on MID data

Genetic Algorithm		
	Predicted Positive	Predicted Negative
Actual Positive	256	25
Actual Negative	39	215
88.04%		

Particle Swarm Optimisation

Table 5.5: Confusion matrix of a the PSO discretisation on MID data

Particle Swarm Optimisation		
	Predicted Positive	Predicted Negative
Actual Positive	233	32
Actual Negative	73	197
80.37%		

Hill Climbing

Table 5.6: Confusion matrix of a the HC discretisation on MID data

Hill Climbing		
	Predicted Positive	Predicted Negative
Actual Positive	225	34
Actual Negative	70	206
80.56%		

Simulated Annealing

Table 5.7: Confusion matrix of a the SA discretisation on MID data

Simulated Annealing		
	Predicted Positive	Predicted Negative
Actual Positive	241	20
Actual Negative	67	207
83.74%		

5.3.2 Comparison of Methods on MID data

The optimised approaches are compared against EWB and EFB partitioning. The comparison is based on classification accuracy using AUC as the performance measure as is done with HIV. Figure 5.5 displays the four optimised approaches plotted against each other, this is performed to determine if one particular technique is better suited than the others for discretisation of input data into the rough set using the MID data set. Table 5.8 summarises the results of all the methods applied to the conflict (MID) data set, it states the number of rules produced, accuracies (AUC) and the computational time required (in minutes).

Genetic Algorithm ROC Curve

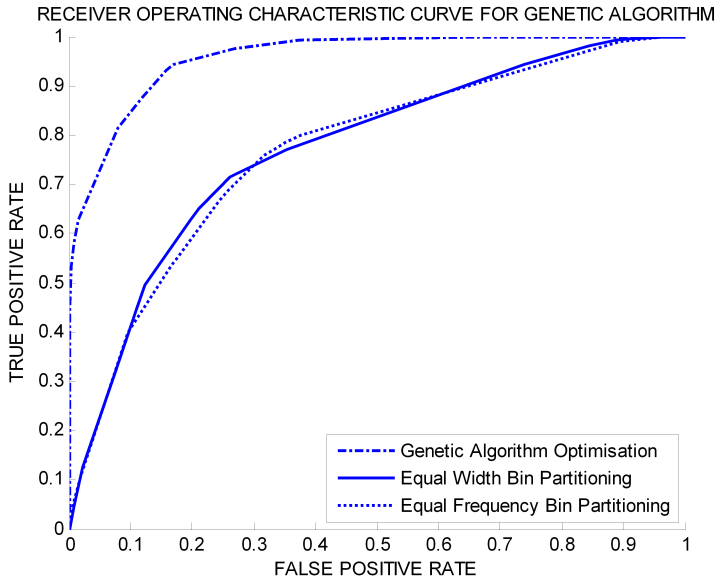

Figure 5.1: Receiver operating characteristic (ROC) curves for GA, EWB and EFB on MID data

Particle Swarm Optimisation ROC Curve

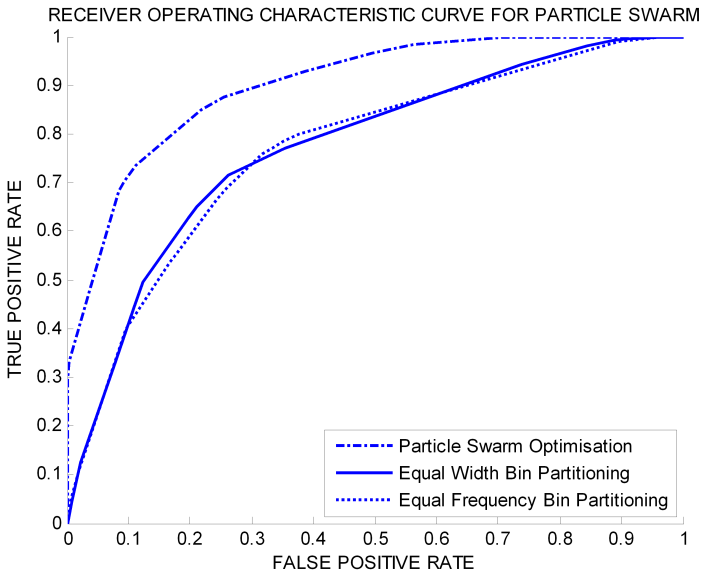

Figure 5.2: Receiver operating characteristic (ROC) curves for PSO, EWB and EFB on MID data

Hill Climbing ROC Curve

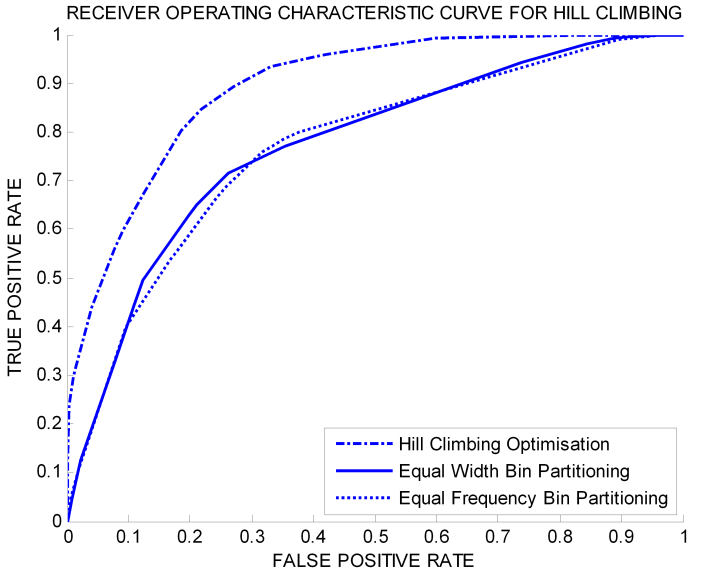

Figure 5.3: Receiver operating characteristic (ROC) curves for HC, EWB and EFB on MID data

Simulated Annealing ROC Curve

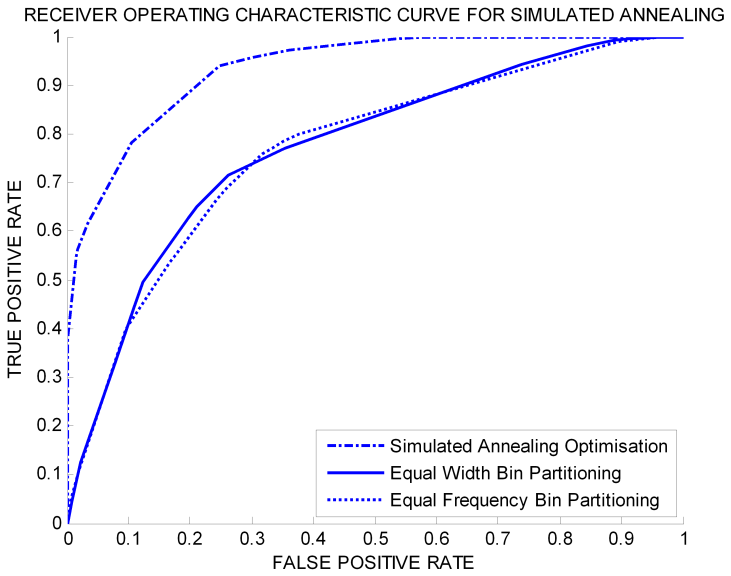

Figure 5.4: Receiver operating characteristic (ROC) curves for SA, EWB and EFB on MID data

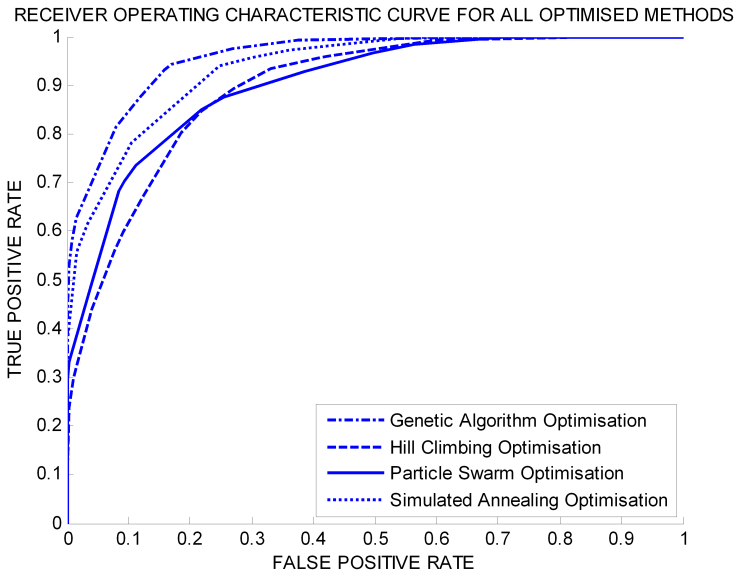

Figure 5.5: Receiver operating characteristic (ROC) curves for GA, PSO, HC and SA on MID data

Figure 5.5 illustrates the performance of each optimisation techniques against each other. As can be seen from the graph, the genetic algorithm approach produces the highest accuracy (AUC).

Table 5.8: Results obtained for the four optimisation and two static discretisation methods

	Number of Rules	Comp. Time	AUC
Genetic Algorithm	422	420	0.9582
Particle Swarm Optimisation	260	394	0.9281
Hill Climbing	187	988	0.8934
Simulated Annealing	396	100	0.9352
Equal Width Bin	123	1	0.7724
Equal Frequency Bin	84	1	0.7672

From the results given in Table 5.8, it can be seen when using the conflict data set, that more rules are produced when a higher accuracy is produced. Once again the optimised approaches produced higher accuracies than the previously more well known static methods of equal-width-bin and equal-frequency-bin partitioning. It must be stated that although the optimised approaches produce higher classification accuracies, it does come at the expense of computational time as can be seen when a comparison is made between the optimised and static methods. The genetic algorithm approach produced the greatest forecasting accuracy (AUC) of 95.82%, while considering only the optimisation methods, simulated annealing is the most time efficient, i.e. has the lowest computational time.

Using the results produced by each optimisation method, the optimal known discretisation points can be determined, and using these partition cuts, rough sets can be run on the particular data set and forecast with a certain plausibility based on the rough membership function. From these rough sets, rules can also be extracted in an *if* CONDITION(S)-*then* DECISION format. These rules are linguistic and easy to interpret. To illustrate this, a lower approximation case generated from the MID data set is represented as shown below. Before the input attributes are discretised, it will look in the form as it is shown below. The outcome or decision can only be determined once a rule set has been formed (from discretised attributes). Therefore, for a given set of attributes, an outcome associated with it can be determined.

- **If** Allies = 1 (True) **and** Contiguity = 0 (False) **and** Major Powers = 1 (True) **and** Distance = 8 (High) **and** Capability = 9 (High) **and** Democracy = 10

(True) **and** Dependency = 0.17 (High) **Then** Outcome = Most Probably Peace

The findings as illustrated in the generated rule shown above, are as what one would expect. Although an attribute sensitivity analysis does not make sense when applied to rough set theory, the rules produced do provide insight into the factors that may contribute to conflict. This can be done by varying a particular attribute or several attributes simultaneously and observe what factors result in what particular outcome. Another important observation is the rules can be generated without the use of *a priori* information. The optimal cut points and partition sizes for the seven dyadic variables for the GA method are shown in Figure 5.6.

From the findings of both experimental investigations (HIV and MID), the use of optimisation techniques produce higher classification/forecasting accuracies than that of the static EWB and EFB methods. These higher accuracies produce more accurate rules from which forecasting and causal interpretations can be performed. For the case of HIV, hill climbing produced the best results but this came at the huge expensive of computational time. For MID, the genetic algorithm approach produced the best classification accuracy, yet simulated annealing produced an accuracy nearly as high as that of GA but in the least computational time. As expected, there is no significant difference between the use of any of the four optimisation techniques. In most cases an optimum will be found. Using the method of hill climbing implemented, the risk of finding a local rather than a global optimum is increased. Genetic algorithms and simulated annealing are better optimisation techniques to be applied to the general problem of discretising rough sets due to their nature of finding a global optimum. Although the GA optimisation method produced good classification results, it must be noted that in some cases the GA may prematurely converge towards local optima.

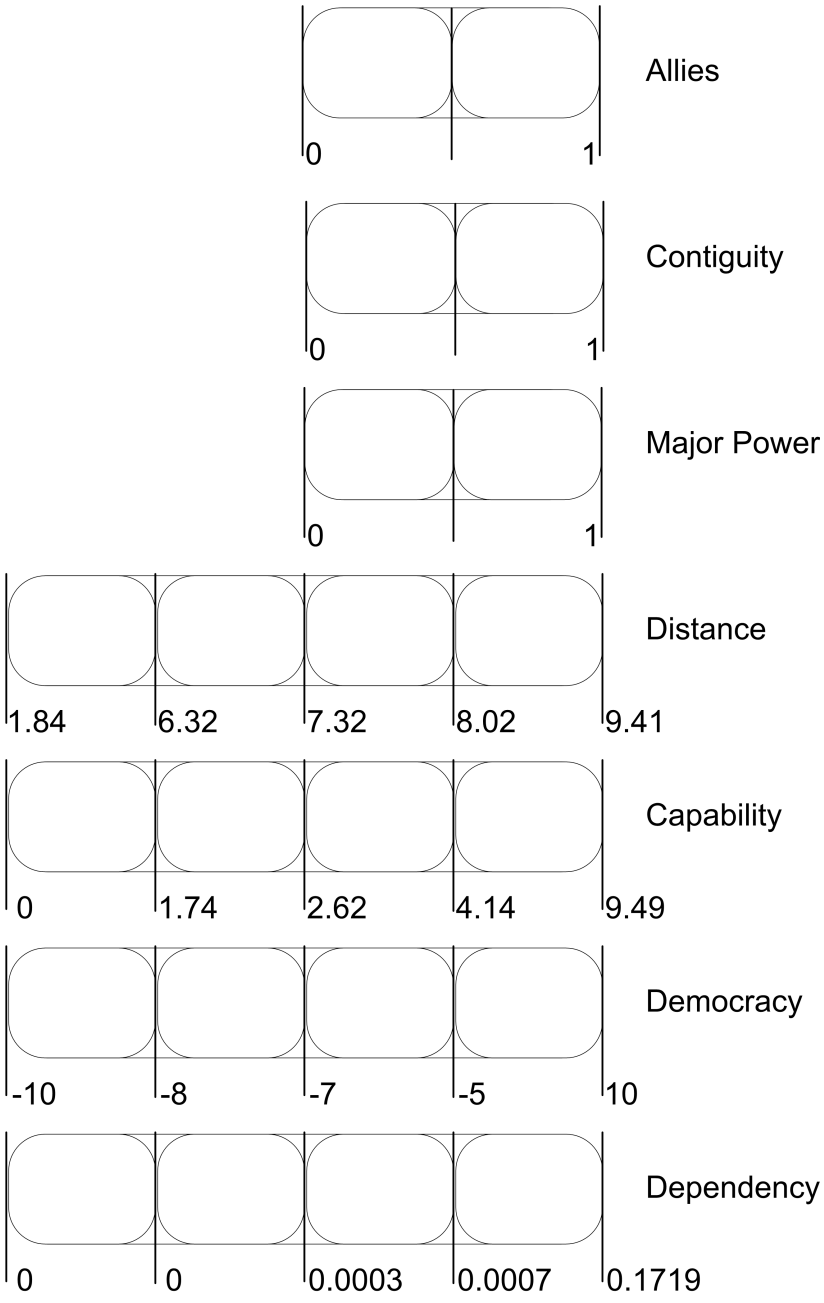

Figure 5.6: Discretisation points for the various attributes using the GA method on MID data

Chapter 6

Conclusion

6.1 Summary of Findings

The use of rough set theory is very useful when dealing with vague and uncertain data. It is also particularly useful for generating rules for the purpose of forecasting and determining causal interpretations of a particular data set. An important step in rough set model generation is the discretisation/granulisation of the input data into partitions. Four partitions are chosen in this research as they provide a good balance between accuracy achieved and computational time. Several methods have been used to perform this granulisation step, and in particular, two well known static methods called equal-width-bin (EWB) and equal-frequency-bin (EFB) partitioning. The work performed in this research is concerned with using various optimisation techniques to granulise the rough set input partitions to achieve the highest forecasting accuracy produced by the rough set. This forecasting accuracy is measured by using the area under the curve (AUC) of the receiver operating characteristic (ROC) curve. The four optimisation techniques used are genetic algorithm, particle swarm optimisation, hill climbing and simulated annealing. This newly proposed method is tested on two data sets, namely, the human immunodeficiency virus (HIV) data set and the militarised interstate dispute (MID) data set. The HIV data was obtained from the antenatal sero-prevalence survey taken in South Africa in 2001. The MID

data used is from the correlates of war (COW) project. The COW project facilitates the collection, dissemination and use of accurate and reliable quantitative data in international relations.

The results of both data sets indicate that the optimised methods produce higher forecasting/classification accuracies than that of the static EWB and EFB partitioning methods. When comparing the optimised approaches against each other, as is the case with both data sets, as expected there is no significant difference between the methods used. When looking at the MID results, the hill climbing accuracy is marginally lower than the other three methods. This could be attributed to the fact that the implementation of hill climbing used allows the optimum to be found at a local rather than a global point. It must be noted that depending on the linearity and complexity of the data set, the optimal technique to be used to discretise the partitions will vary. The rough sets produce a balance between transparency of the rough set model and accuracy of HIV estimation, but it does come at a cost of high computational effort. The data was balanced and split into a 70% to 30% training data to testing data ratio. The highest accuracy achieved for the HIV data set is obtained by using the hill climbing approach, and an accuracy of 69.02% is achieved in a time of 12624 minutes. For the MID data, the genetic algorithm approach produced the highest accuracy. This accuracy is 95.82% in a time of 420 minutes. As shown in Chapters 4 and 5, the rules extracted are linguistic, but this transparency of the model does come at the expense of classification accuracy. This loss in accuracy is brought about in the discretisation process where the granularity of the variables are decreased. A highly interpretable model is very useful in the study of HIV and conflict management studies as it allows for policy formulation and the testing of causal hypotheses.

6.2 Recommendations for Further Work

The work performed shows that the application of rough set theory to HIV and MID is feasible. The accuracies obtained for MID are very good while the results

produced for HIV are not as good. Altering the parameters and implementation of the optimisation techniques may lead to an increase in the rough set forecasting accuracy. For example, in the genetic algorithm, the introduction of other diversification mechanisms such as elitism can also be implemented. Other work that could be investigated includes using optimisation techniques to run in parallel with the existing framework to determine the *number* of partitions required. An investigation into the bounds of the optimisation techniques should also be performed to determine whether this has a large effect on the results or not. The effect of the rule generation from the inclusion of reducts should be tested, it can also be investigated whether optimisation techniques serve any purpose in discretising the input data given the fact that reducts are going to remove redundant information.

Current policies and theories of the relevant field can be tested against the obtained results. The model should also be tested in conjunction with experts in the particular field. These experts can indicate whether the results obtained are feasible and more importantly whether the results can actually be used to forecast future occurrences. If the data had been balanced the same way as that in the MID study performed by [80], and the HIV study performed by [50] the results from this research could be compared.

Appendix A

Algorithms

In this appendix, various algorithms that were discussed in the thesis are presented.

A.1 Rough Set Model Generation

Algorithm 1: Algorithm to generate rough set model

Input : Condition and Decision Attributes
Output: Certain and Possible Rules

Obtain data set to be used;
repeat
  for $conditionalattribute \leftarrow 1$ **to** $sizeoftrainingdata$ **do**
    Pre-process data to ensure its ready for analysis;
    Discretise data according to optimisation technique;
    Compute the lower approximation, such that:
    $\underline{B}X = \{x \in \ \mathbf{U}\text{: } B(x) \subseteq X\}$;
    Compute the upper approximation, such that:
    $\overline{B}X = \{x \in \ \mathbf{U}\text{: } B(x) \cap X \neq \ \emptyset\}$;
    From the generated rules, calculate plausibility measures of which an object x belongs to set X. This is defined by: $\mu_A^X = \ \frac{|[X]_B \ \cap X|}{[X]_B}$;
    Extract the certain rules from the lower approximation generated for each subset;
    Similarly, extract the possible rules from the upper approximation of each subset;
    Remove the rules generated for the purpose of testing on unseen data;
    Compute the classifier performance using AUC;
  end
until *Optimisation technique termination condition* ;

A.2 Receiver Operating Characteristics (ROC) Curve

The ROC curve implemented is the *efficient method* given by [40]. This is given by:

Algorithm 2: Algorithm to calculate ROC curve

Input : L, the test set of instances; $f(i)$, the probabilistic classifier's estimate that instance i is positive; P and N, the number of positive and negative examples.
Output : R, a list of ROC points increasing by *fp rate*.
Require: $P > 0$ and $N > 0$

```
1  begin
2      L_sorted ← L sorted decreasing by f scores;
3      FP ← TP ← 0;
4      R ← ⟨⟩;
5      f_prev ← −∞;
6      i ← 1;
7      while i ≤ |L_sorted| do
8          if f(i) ≠ f_prev then
9              push (FP/N, TP/P) onto R;
10             f_prev ← f(i);
11         end
12         if L_sorted[i] is a positive example then
13             TP ← TP + 1
14         else
15             FP ← FP + 1
16         end
17         i ← i + 1;
18     end
19     push (FP/N, TP/P) onto R;
20 end
```

There are various approaches to calculating ROC graphs, and although one possible method is given above, it must be noted that this is not the only method that can be used.

The area under curve (AUC) of the ROC curve is computed in a similar way as to that of just the ROC curve. This implementation will be given in Algorithm 3 on the next page.

A.3 Area Under Curve (AUC)

Algorithm 3: Algorithm to calculate the AUC

Input : L, the set of test examples; $f(i)$, the probabilistic classifier's estimate that instance i is positive; P and N, the number of positive and negative examples.
Output : A, the area under the ROC curve
Require: $P > 0$ and $N > 0$

begin
  $L_{sorted} \leftarrow L$ sorted decreasing by f scores;
  $FP \leftarrow TP \leftarrow 0$;
  $FP_{prev} \leftarrow TP_{prev} \leftarrow 0$;
  $A \leftarrow 0$;
  $f_{prev} \leftarrow -\infty$;
  $i \leftarrow 1$;
  while $i \leq |L_{sorted}|$ **do**
    if $f(i) \neq f_{prev}$ **then**
      $A \leftarrow A +$ TRAPEZOID AREA$(FP, FP_{prev}, TP, TP_{prev})$;
      $f_{prev} \leftarrow f(i)$;
      $FP_{prev} \leftarrow FP$;
      $TP_{prev} \leftarrow TP$;
    end
    if *i is a positive example* **then**
      $TP \leftarrow TP + 1$;
    else
      $FP \leftarrow FP + 1$;
    end
  end
  $A \leftarrow A +$ TRAPEZOID AREA$(1, FP_{prev}, 1, TP_{prev})$;
  $A \leftarrow \frac{A}{(P \times N)}$;
end

The algorithm describing the TRAPEZOID AREA function is given in Algorithm 4.

Algorithm 4: TRAPEZOID AREA function

Function: TRAPEZOID AREA (X1,X2,Y1,Y2)
  $Base \leftarrow |\text{X1} - \text{X2}|$;
  $Height_avg \leftarrow \frac{(Y1+Y2)}{2}$;
return : $Base \times Height_{avg}$

Appendix B

Flow Diagrams

In this appendix, the flow diagrams for the genetic algorithm, particle swarm optimisation, hill climbing and simulated annealing optimisation techniques are illustrated. Each flow chart will be given on a new page starting from the next page.

B.1 Genetic Algorithm

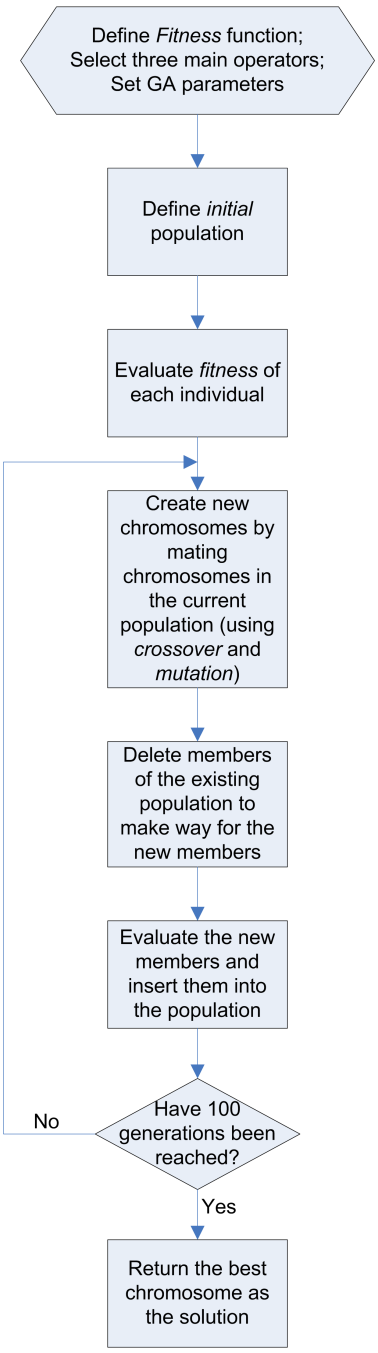

Figure B.1: Flow diagram of genetic algorithm optimisation

B.2 Particle Swarm Optimisation

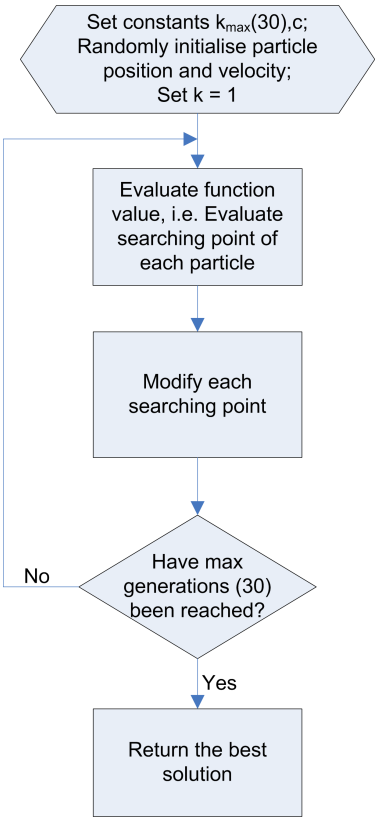

Figure B.2: Flow diagram of particle swarm optimisation

B.3 Hill Climbing

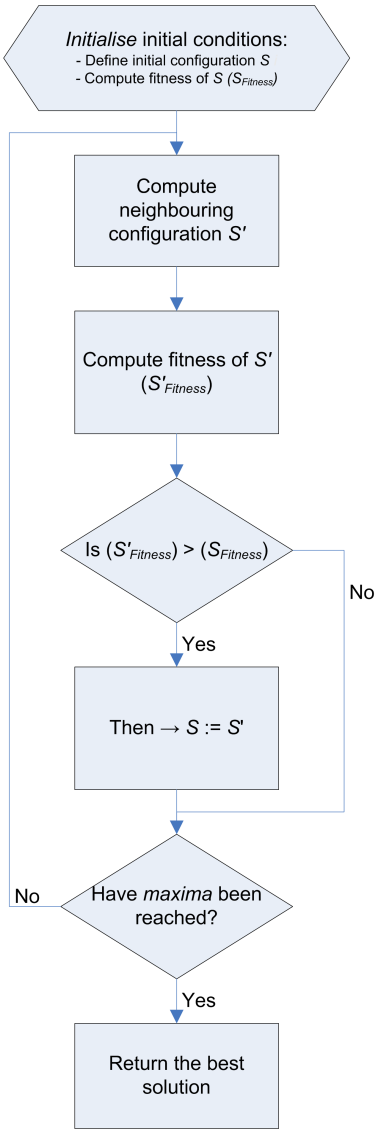

Figure B.3: Flow diagram of hill climbing optimisation

B.4 Simulated Annealing

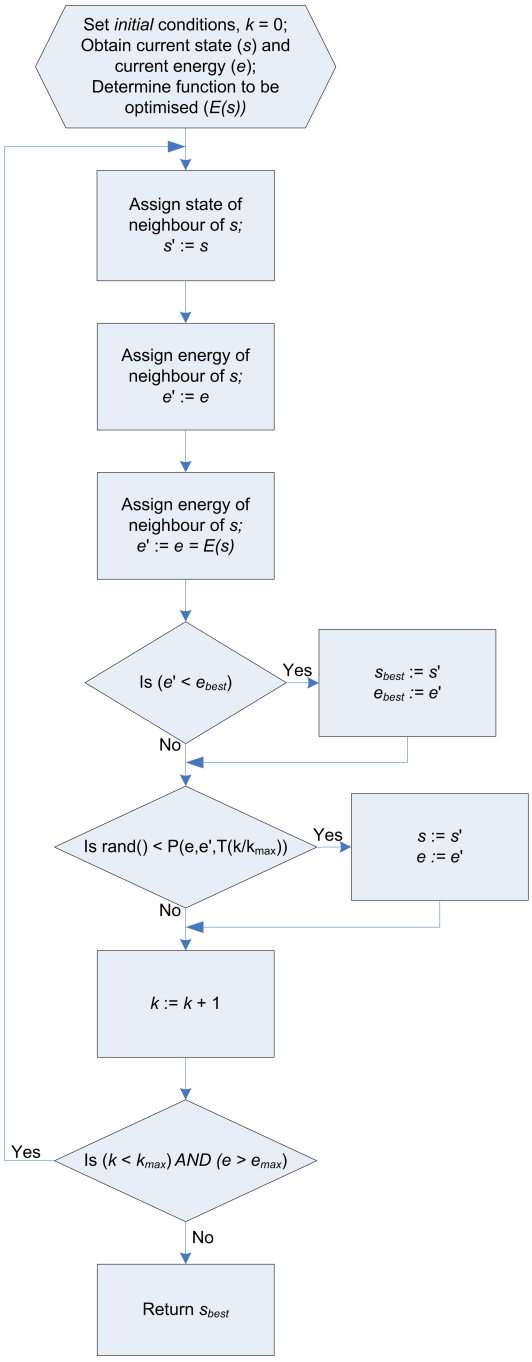

Figure B.4: Flow diagram of hill climbing optimisation

Appendix C

Published Work

In this appendix the papers published based on the work performed in this thesis are listed.

- B. Crossingham and T. Marwala. "Using Optimisation Techniques to Granulise Rough Set Partitions". Mathematical and Statistical Physics, In Proceedings of International Symposium on Computational Models of Life Sciences, American Institute of Physics, Volume 1, 2007, pp. 248-258.
- B. Crossingham and T. Marwala. "Using Genetic Algorithms to Optimise Rough Set Partition Sizes for HIV Data Analysis". Studies in Computational Intelligence, In Advances in Intelligent and Distributed Computing, Springer, Volume 78, 2008, pp. 245 - 250.

These papers are attached to the back of this dissertation.

References

[1] Z. Pawlak, *Rough Sets, Theoretical Aspects of Reasoning about Data*, ch. 3, p. 33. Kluwer Academic Publishers, 1991.

[2] J. Komorowski, L. Polkowski, and A. Skowron, "Rough sets: a tutorial. in s.k. pal and a. skowron, editors, rough-fuzzy hybridization: A new method for decision making," 1998.

[3] A. F. B. Jaafar, J. Jais, M. H. B. H. A. Hamid, Z. B. A. Rahman, and D. Benaouda, "Using rough set as a tool for knowledge discovery in dss," in *Proceedings of the 4th International Conference on Multimedia and Information and Communication Technologies in Education*, Seville, Spain, November 2006.

[4] U. Fayyad and K. Irani, "Multi-interval discretization of continuous-valued attributes for classification learning," in *Proceedings of the 13th International Joint Conference on Artificial Intelligence*, p. 10221027, Los Alamos, CA 1993.

[5] J. W. Grzymala-Busse, "Mining numerical data - a rough set approach," in *Proceedings of the Rough Sets and Emerging Intelligent Systems Paradigms*, pp. 12–21, June 2007.

[6] G. D. Garson, "A comparison of neural network and expert systems algorithms with common multivariate procedures for analysis of social science data," *Social Science Computer Review*, vol. 9, no. 3, pp. 399–433, 1991.

[7] L. Zeng, "Prediction and classification with neural network models," *Sociological Methods and Research*, pp. 499–524, 27 1999.

[8] E. A. Habtemariam and T. Marwala, "Artificial intelligence for conflict management," in *Proceedings of the IEEE International Joint Conference on Neural Networks (Montreal, Canada)*, pp. 2583–2588, 2005.

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

[10] Y. Yang and R. John, "Roughness bounds in set-oriented rough set operations," in *2006 IEEE International Conference on Fuzzy Systems*, pp. 1461–1468, Vancouver, Canada 2006.

[11] L. Zhai, L. Khoo, and S. Fok, "Feature extraction using rough set theory and genetic algorithms-an application for the simplification of product quality evaluation," *Computers and Industrial Engineering*, vol. 43, pp. 661–676, 2002.

[12] A. Ohrn, "Discernibility and rough sets in medicine: Tools and applications." PhD Thesis, Department of Computer and Information Science Norwegian University of Science and Technology, 1999.

[13] A. Ohrn and T. Rowland, "Rough sets: A knowledge discovery technique for multifactorial medical outcomes," *American Journal of Physical Medicine and Rehabilitation*, vol. 79, pp. 100–108, 2000.

[14] A. Deja and P. Peszek, "Applying rough set theory to multi stage medical diagnosing," *Fundamenta Informaticae*, pp. 387–408, 54 2003.

[15] A. Mrozek, *Rough sets in computer implementation of rule-based control of industrial process. In R. Sowinski (Ed.), Intelligent decision supportHandbook of applications and advances of the rough sets theory*, pp. 19–32. Kluwer Academic Publishers, 1992.

[16] J. Pe-a, S. Ltourneau, and A. Famili, "Application of rough sets algorithms to prediction of aircraft component failure," in *Proceedings of the Third International Symposium on Intelligent Data Analysis*, Amsterdam 1999.

[17] F. E. H. Tay and L. Shen, "Fault diagnosis based on rough set theory," *Engineering Applications of Artificial Intelligence*, vol. 16, p. 3943, 2003.

[18] R. H. Golan and W. Ziarko, "A methodology for stock market analysis utilizing rough set theory," in *Proceedings of Computational Intelligence for Financial Engineering*, pp. 32–40, New York, USA 1995.

[19] R. Deja and D. Slezak, "Rough set theory in conflict analysis," in *New Frontiers in Artificial Intelligence : Joint JSAI 2001 Workshop Post-Proceedings*, pp. 349–353, 2001.

[20] Z. Pawlak, "Some remarks on conflict analysis," *European Journal of Operational Research*, vol. 166, p. 649654, 2005.

[21] T. Tettey, F. V. Nelwamondo, and T. Marwala, "HIV data analysis via rule extraction using rough sets," in *Proceedings of the 11th International Conference on Intelligent Engineering Systems*, June 2007.

[22] D. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning*, pp. 1–25. Addison-Wesley, Reading, MA, 1989.

[23] R. Hassan, B. Cohanim, and O. de Weck, "A comparison of particle swarm optimisation and the genetic algorithm," in *Proceedings of the 46th American Institute of Aeronautics and Astronautics*, Austin, Texas, April 2005.

[24] R. Poli, W. B. Langdon, and O. Holland, "Extending particle swarm optimisation via genetic programming," in *Proceedings of the 8th European Conference on Genetic Programming*, vol. 3447 of *Lecture Notes in Computer Science*, (Lausanne, Switzerland), Springer, 2005.

[25] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, "Optimization by simulated annealing," *Science, Number 4598, 13 May 1983*, vol. 220, no. 4598, pp. 671–680, 1983.

[26] N. Metropolis, A. Rosenbluth, M. Rosenbluth, A. Teller, and E. Teller, "Equation of state calculations by fast computing machines," *The Journal of Chemical Physics*, vol. 21, no. 6, pp. 1087–1092, 1953.

[27] N. P. X. Wang, J. Yang and X. Teng, *Finding Minimal Rough Set Reducts with Particle Swarm Optimization*, ch. Rough Sets, Fuzzy Sets, Data Mining, and Granular Computing, pp. 451–460. Lecture Notes in Computer Science, 3641 ed., 2005.

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

[29] B. B. Leke, T. Marwala, T. Tim, and M. Lagazio, "Prediction of HIV status from demographic data using neural networks," in *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics*, pp. 2339–2344, Taiwan 2006.

[30] B. B. Leke, T. Marwala, and T. Tettey, "Using inverse neural network for HIV adaptive control," *International Journal of Computational Intelligence Research*, vol. 3, pp. 11–15, 2007.

[31] J. Cannady, "Artificial neural networks for misuse detection," in *Proceedings of the 1998 National Information Systems Security Conference*, Arlington, VA 1998.

[32] M. Lagazio and B. Russet, *Temporal Stability and Causal Complexity, ch. A Neural Network Analysis of Militarized Disputes, 1885-1992: Temporal Stability and Causal Complexity*, pp. 28–62. New Jersey: University of Michigan Press, 2003.

[33] N. Beck, G. King, and L. Zeng, "Improving quantitative studies of international conflict: A conjecture," *American Political Science Review*, pp. 21–33, 94 2000.

[34] P. Schrodt, "Prediction of interstate conflict outcomes using a neural network," *Social Science Computer Review*, vol. 9, no. 3, pp. 359–380, 1991.

[35] S. Theodolite and K. Outnumbers, *Pattern Recognition*. New York: Academic Press, first ed., 1999.

[36] T. Marwala and M. Lagazio, "Modelling and controlling interstate conflict," in *Proceedings of the IEEE International Joint Conference on Neural Networks*, pp. 1233–1238, Budapest, Hungary 2004.

[37] Z. Pawlak and T. Munakata, "Rough control application of rough set theory to control," in *Proceedings of the 4th European Congress on Intelligent Techniques and Soft Computing*, vol. 1, pp. 209–218, September 1996.

[38] A. Pagnoni, S. Parisi, and S. Lombardo, "Analysis of patient flows via data mining," *Medininfo*, vol. 10, pp. 1379–1383, 2001.

[39] N. S. Hoa and N. G. Son, "Rough set approach to approximation of concepts from taxonomy," in *Proceedings of Knowledge Discovery and Ontologies Workshop*, (Pisa, Italy), September 2004.

[40] T. Fawcett, "ROC graphs: Notes and practical considerations for researchers." http://home.comcast.net/ tom.fawcett/public_html/papers/ROC101.pdf, 2004.

[41] A. Bradley, "The use of the area under the ROC curve in the evaluation of machine learning algorithms," *Pattern Recognition*, vol. 30, pp. 1145–1159, 1997.

[42] S. Rosset, "Model selection via the AUC," in *In Proceedings of the 21st International Conference on Machine Learning*, July 2004.

[43] C. X. Ling, J. Huang, and H. Zhang, "A statistically consistent and more discriminating measure than accuracy," in *Proceedings of the 18th International Joint Conferences on Artificial Intelligence*, August 2003.

[44] D. Mossman and E. Somoza, "ROC curves, test accuracy, and the description of diagnostic tests," *Neuropsychiatry and Clinical Neurosciences*, vol. 3, pp. 330–333, 1991.

[45] R. Department of Health, "National HIV and syphilis sero-prevalence survey of women attending public antenatal clinics in south africa." http://www.info.gov.za/otherdocs/2002/hivsurvey01.pdf, 2001.

[46] "Correlates of war." www.correlatesofwar.org/. Last accessed: 22/2/2007.

[47] F. Witlox and H. Tindemans, "The application of rough sets analysis in activity-based modelling. opportunities and constraints," *Expert Systems with Applications*, vol. 27, p. 585592, 2004.

[48] M. Inuiguchi and T. Miyajima, "Rough set based rule induction from two decision tables," *European Journal of Operational Research*, vol. In Press, Corrected Proof, 2006.

[49] C. Goh and R. Law, "Incorporating the rough sets theory into travel demand analysis," *Tourism Management*, vol. 24, p. 511517, 2003.

[50] B. B. Leke, *Computational Intelligence for Modelling HIV*. PhD thesis, University of the Witwatersrand, School of Electrical and Information Engineering, 2007.

[51] N. H. Son, "Discretization of real value attributes: A boolean reasoning approach." PhD Thesis, Department of Mathematics, Computer Science and Mechanics, Warsaw University, 1997.

[52] M. R. Chmielewski and J. W. Grzymala-Busse, "Global discretization of continuous attributes as preprocessing for machine learning," in *Proceedings of the 3rd International Workshop on Rough Sets and Soft Computing*, pp. 294–301, 1994.

[53] J. Dougherty, R. Kohavi, and M. Sahami, "Supervised and unsupervised discretization of continuous features," in *Proceedings of the 12th International Conference in Machine Learning*, Los Altos, CA 1995.

[54] S. Malve and R. Uzsoy, "A genetic algorithm for minimizing maximum lateness on parallel identical batch processing machines with dynamic job arrivals and incompatible job families," *Computers and Operations Research*, vol. 34, pp. 3016–3028, 2007.

[55] C. H. Lim, Y. S. Yoon, and J. H. Kim, "Genetic algorithm in mix proportioning of high-performance concrete," *Cement and Concrete Research*, vol. 34, p. 409420, 2004.

[56] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of IEEE International Conference on Neural Networks*, vol. 4, pp. 1942–1948, Perth, Australia 1995.

[57] J. Cooper, "Dissonance and the return of the self-concept," *Psychological Inquiry*, vol. 3, no. 4, p. 320323, 1992.

[58] J. H. Wingfield, *A Goal Systemic Analysis of Cognitive Dissonance Phenomena*. PhD thesis, University of Maryland, College Park, 2005.

[59] S. Naka, T. Genji, T. Yura, and Y. Fukuyama, "Hybrid particle swarm optimization for distribution state estimation," *IEEE Transactions on Power Systems*, vol. 18, pp. 60–68, 2003.

[60] M. Mitchell, J. Holland, and S. Forest, "When will a genetic algorithm outperform hill climbing," *Advances in Neural Information Processing Systems 6, J. Cowan, G. Tesauro, and J. Alspector eds.*, pp. 51–58, San Mateo, Calif.: Morgan Kaufman 1994.

[61] t. f. e. Wikipedia, "Hill climbing." http://en.wikipedia.org/wiki/Image:Local_maximum.png, 2007.

[62] K. Bryan, P. Cunningham, and N. Bolshkova, "Application of simulated annealing to the biclustering of gene expression data," *IEEE Transactions on Information Technology in Biomedicine*, vol. 10, no. 3, pp. 519–525, 2006.

[63] R. Salazar and R. Toral, "Simulated Annealing using Hybrid Monte Carlo," *arXiv:cond-mat/9706051*, 2006.

[64] A. Lasry, G. S. Zaric, and M. W. Carter, "Multi-level resource allocation for HIV prevention: A model for developing countries," *European Journal of Operational Research*, vol. 180, p. 786799, 2007.

[65] "UNAIDS." www.unaids.org/en/HIV_data/2006GlobalReport/default.asp/. Last accessed: 20/3/2007, 2006.

[66] D. Reznik, "A critique of the AIDS metanarrative." Website: http://www.clas.ufl.edu/users/dreznik/Aarrative.pdf.

[67] E. Fee and N. Krieger, "Understanding AIDS: historical interpretations and the limits of biomedical individualism," *American Journal of Public Health*, vol. 83, pp. 1477–1486, 1993.

[68] K. E. Poundstone, S. A. Strathdee, and D. D. Celentano, "The social epidemiology of human immunodeficiency virus/acquired immunodeficiency syndrome," *Epidemiol Reviews*, vol. 26, pp. 22–35, 2004.

[69] C. W. Lee and J. A. Park, "Assessment of HIV/AIDS-related health performance using an artificial neural network," *Information & Management*, vol. 38, pp. 231–238, 2001.

[70] A. Lumini and L. Nanni, "Machine learning for HIV-1 protease cleavage site prediction," *Pattern Recognition Letters*, vol. 27, pp. 1537–1544, 2006.

[71] M. Lagazio and T. Marwala, "Assessing Different Bayesian Neural Network Models for Militarized Interstate Dispute," *Social Science Computer Review*, vol. 24, pp. 119–131, 2006.

[72] J. R. Oneal and B. Russet, "Causes of peace: Democracy, interdependence, and international organizations, 1885-1992," in *2001 Annual Meeting of the American Political Science Association*, San Francisco, CA, USA 2001.

[73] C. Gochman and Z. Maoz, "Militarized interstate disputes 1816-1976," *Journal of Conflict Resolution*, vol. 28, no. 4, pp. 585–615, 1984.

[74] W. R. Thompson; and R. Tucker, "A tale of two democratic peace critiques," *The Journal of Conflict Resolution*, vol. 41, pp. 428–454, 1997.

[75] E. Mansfield and J. Snyder, "Democratization and the danger of war," *International Security*, vol. 20, pp. 5–38, 1995.

[76] T. Tettey and T. Marwala, "Neuro-fuzzy modelling and fuzzy rule extraction applied to conflict management," *Lecture Notes in Computer Science, In Neural Information Processing, Springer Berlin / Heidelberg*, vol. 4234, pp. 1087–1094, 2006.

[77] M. G. Marshall, "The scientific study of international conflict processes: Post-cards at the edge of the millennia," *Centre for Systematic Peace, Website: http://www.cidcm.umd.edu/inscr/papers/icpmgm.pdf*.

[78] L. P. Pojman, "Kants perpetual peace and cosmopolitanism." Website: http://www.blackwell-synergy.com/doi/pdf/10.1111/j.1467-9833.2005.00258.x.

[79] G. Rose, "Neoclassical realism and theories of foreign policy," *World Politics*, pp. 144–172, 51 1998.

[80] M. Lagazio, T. Marwala, and T. Tettey, "An Integrated Human-Computer System for Controlling Interstate Disputes," *arXiv:0704.3862*, 2006.