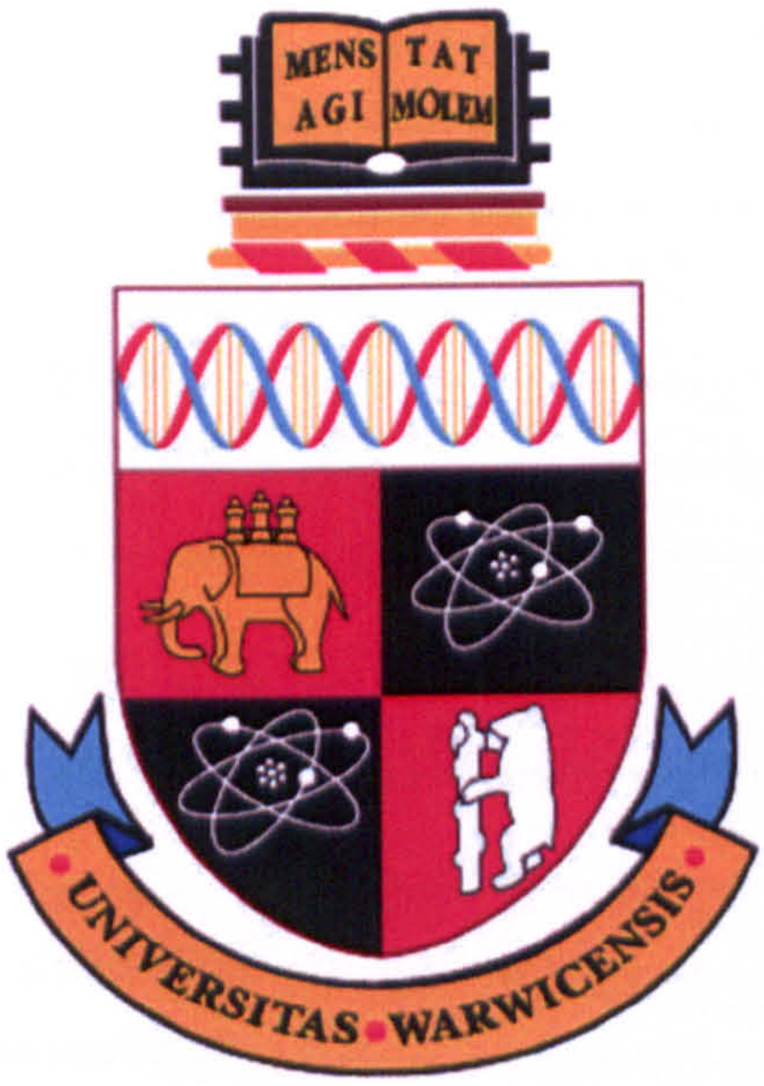
On the Development of Intelligent Medical Systems for Pre-operative Anaesthesia Assessment



By

Ross Simon Folland

A thesis submitted in partial fulfilment of the requirements for the degree of Doctor of Philosophy in Engineering

University of Warwick, School of Engineering,
January 2005

Contents

List of Figures

List of Tables

Acknowledgements

I would like to thank the following people from the School of Engineering, University of Warwick for their assistance during my period of research. I would like to thank my supervisor, Dr. Evor. L. Hines for his support and encouragement over the years. Secondly, I would like to extend my sincere gratitude (in no particular order) to Drs. Reza Ali, Pascal Boilot, James Covington, Ritaban Dutta, Neil Evans, Mario Gongora, Alex Jouvray, James Yates, Ms. Aruneema Das and Ms. Elizabeth Erickson.

From Birmingham Heartlands Hospital, I would like to thank Mr. David Morgan, consultant otolaryngologist & skull-base surgeon, for his time and assistance. Furthermore, I would like to thank Dr. F. Leyva (Good Hope Hospital, West Midlands), Dr. S. OHickey (Worcestershire Royal Hospital) and the ENT outpatient clinic staff of Birmingham Heartlands Hospital, Good Hope Hospital and Solihull Hospital (West Midlands) for their assistance in the gathering of data.

I would also like to gratefully acknowledge the financial support of Medicdirect.co.uk Ltd.

Finally I would like to thank my parents and sister, whose never ending support and encouragement made this work possible.

Declaration

This thesis is presented in accordance with the regulations for the degree of doctor of philosophy. All work reported has been carried out by the author unless otherwise stated, including the production of this document.

Glossary of Abbreviations

ANFIS	Adaptive Neuro Fuzzy Inference System
ANN	Artificial Neural Network
ANS	Autonomic Nervous System
BP	Back Propagation (*see* BPGDM)
BP	Blood Pressure
BPGDM	Back Propagation, Gradient Descent with Momentum
bpm	beats per minute
bps	bits per second
BPT	Bayesian Probability Theory
CHD	Coronary Heart Disease
CPNN	Constructive Probabilistic Neural Network (*see* PNN)
CXR	Chest X-Ray
DDA	Dynamic Decay Adjustment
DFT	Discrete Fourier Transform (*see* FFT)
DSA	Decision Support Aid
DST	Dempster-Shafer Theory
ECG	Electrocardiogram
eFMM	Evolutionary Fuzzy Mixture Model
EFuNN	Evolving Fuzzy Neural Network
EHR	Electronic Health Record
ENT	Ear, Nose & Throat
EPROM	Erasable Programmable Read-Only Memory
FEV	Forced Expiratory Volume
FFT	Fast Fourier Transform
FIS	Fuzzy Inference System
FL	Fuzzy Logic

GA	Genetic Algorithm
GAv	Genetic Algorithm variant (*see* GA)
Gb	Gigabyte
GMM	Gaussian Mixture Model
GOR	Gastro-Oesophageal Reflux
GP	Genetic Program / Genetic Programming
HL7	Health Level 7
HR	Heart Rate
HRV	Heart Rate Variability
Hz	Hertz
KB	Knowledge Base
kHz	Kilohertz (*see* Hz)
LDA	Linear Discriminant Analysis
MA	Motion Artefact
Mb	Megabyte
MLP	Multi-Layer Perceptron
MRI	Magnetic Resonance Imaging
MSE	Mean Squared Error
NFS	Neural Fuzzy System
NHS	National Health Service (UK)
PC	Personal Computer
PCA	Principal Component Analysis
PEFR	Peak Expiratory Flow Rate
PID	Patient Identification
PNN	Probabilistic Neural Network (*see* CPNN)
P-RCE	Probabilistic Restricted Coulomb Energy (*see* RCE)
RBFN	Radial Basis Function Network
RCE	Restricted Coulomb Energy
RMSE	Root Mean Squared Error
ROC	Receiver Operating Characteristic
SD	Standard Deviation
SSE	Sum Squared Error
UML	Unified Modelling Language

List of Author's Publications

Journal publications:

Folland, R., Hines, E.L., Boilot, P. and Morgan, D.W, Classifying coronary dysfunction using neural networks through cardiovascular auscultation, *Medical & Biological Engineering & Computing*, **40**(3), 2002, pp.339-343.

Folland, R., Hines, E.L., Dutta, R., Boilot, P. and Morgan, D.W., Comparison of neural network predictors in the classification of tracheal-bronchial breath sounds by respiratory auscultation, *Artificial Intelligence in Medicine*, **31**(3), 2004, pp.211-220.

Boilot, P., Hines, E.L., Gongora, M.A. and Folland, R., Electronic noses inter-comparison, data fusion and sensor selection in discrimination of standard fruit solutions, *Sensors and Actuators B: Chemical*, **88**(1), 2003, pp.80-88.

Conference publications:

Folland, R., Dutta, R., Hines, E.L. and Morgan, D.W, Parallel combinations of intelligent systems in classifying tracheal-bronchial breath sounds using a novel cross-validation technique, *Proceedings of the 1st IASTED International Conference on Biomedical Engineering*, (Ed. Hamza, M.H.), 2003, pp.17-22.

Folland, R., Das, A., Dutta, R., Hines, E.L., Stocks, N.G. and Morgan, D.W., Detecting motion artefacts in pulse waveforms using a stochastic resonance technique, *Proceedings of the International Congress on Computational Bioengineering*, (Eds. Doblaré, M., Cerrolaza, M. and Rodrigues, H.), 2003, pp.291-296.

Folland, R., Das, A., Dutta, R., Hines, E.L., Stocks, N.G. and Morgan, D.W., Pulse waveform classification using neural networks with cross-validation techniques: some signal processing considerations, *Proceedings of the 2nd IASTED International Conference on Biomedical Engineering*, (Ed. Tilg, B.), 2004, pp.337-342.

Folland, R., Hines, E.L. and Gongora, M.A., Evolving fuzzy rules: a mixture model approach for medical diagnosis, *Proceedings of the 5th International Conference on Recent Advances in Soft Computing*, (Ed. Lotfi, A.), 2004, pp.312-317.

Folland, R., Gongora, M.A., Hines, E.L. and Morgan, D.W., Evolutionary fuzzy mixture models: applications in anaesthesia, *Proceedings of the 3rd IASTED International Conference on Biomedical Engineering*, (Ed. Hamza, M.H.), 2005, pp.596-601.

Summary

This thesis describes the research and development of a decision support tool for determining a medical patient's suitability for surgical anaesthesia. At present, there is a change in the way that patients are clinically assessed prior to surgery. The pre-operative assessment, usually conducted by a qualified anaesthetist, is being more frequently performed by nursing grade staff. The pre-operative assessment exists to minimise the risk of surgical complications for the patient. Nursing grade staff are often not as experienced as qualified anaesthetists, and thus are not as well suited to the role of performing the pre-operative assessment. This research project used data collected during pre-operative assessments to develop a decision support tool that would assist the nurse (or anaesthetist) in determining whether a patient is suitable for surgical anaesthesia. The three main objectives are: firstly, to research and develop an automated intelligent systems technique for classifying heart and lung sounds and hence identifying cardio-respiratory pathology. Secondly, to research and develop an automated intelligent systems technique for assessing the patient's blood oxygen level and pulse waveform. Finally, to develop a decision support tool that would combine the assessments above in forming a decision as to whether the patient is suitable for surgical anaesthesia.

Clinical data were collected from hospital outpatient departments and recorded alongside the diagnoses made by a qualified anaesthetist. Heart and lung sounds were collected using an electronic stethoscope. Using this data two ensembles of artificial neural networks were trained to classify the different heart and lung sounds into different pathology groups. Classification accuracies up to 99.77% for the heart sounds, and 100% for the lung sounds has been obtained. Oxygen saturation and pulse waveform measurements were recorded using a pulse

oximeter. Using this data an artificial neural network was trained to discriminate between normal and abnormal pulse waveforms. A discrimination accuracy of 98% has been obtained from the system. A fuzzy inference system was generated to classify the patient's blood oxygen level as being either an inhibiting or non-inhibiting factor in their suitability for surgical anaesthesia. When tested the system successfully classified 100% of the test dataset. A decision support tool, applying the genetic programming evolutionary technique to a fuzzy classification system was created. The decision support tool combined the results from the heart sound, lung sound and pulse oximetry classifiers in determining whether a patient was suitable for surgical anaesthesia. The evolved fuzzy system attained a classification accuracy of 91.79%.

The principal conclusion from this thesis is that intelligent systems, such as artificial neural networks, genetic programming, and fuzzy inference systems, can be successfully applied to the creation of medical decision support tools.

CHAPTER 1

Introduction

This thesis presents the problem of providing a decision support tool for the anaesthetist, and ergo to model the knowledge of the anaesthetist in evaluating patients for pre-operative assessment. This chapter introduces the problem, and offers the motivation for this work. The structure of this thesis is outlined, and the proposed contribution of this work is stated.

1.1 Introduction

In 1798 a Clifton doctor by the name of Thomas Beddoes appointed a new superintendent of experiments at his Medical Pneumatics Institution in Bristol, England. The appointee was a Cornish chemist named Humphry Davy who set to work to investigate newly-discovered gases and their properties in curing and preventing disease [1,2]. Davy experimented with nitrous oxide (*laughing gas*) which lead him to describe the effects of breathing the compound as "*a sensation*

Figure 1.1. Gillray's 1802 caricature of Davy's lecture on nitrous oxide at the Royal Institution. Davy is pictured holding the bellows [4].

analogous to gentle pressure on all the muscles, attended by a highly pleasurable thrilling, particularly in the chest and extremities" [3]. Davy became famous for these observations and in 1799 was appointed to the Royal Institution in London by its founder Count Rumford (Benjamin Thompson)[1]. Following these experiments Davy later went on to describe the effects of nitrous oxide on the clinical manifestations of pain and noted that:

> "*In cutting one of the unlucky teeth called dentes sapientiae, I experienced an extensive inflammation of the gum, accompanied with great pain, which equally destroyed the power of repose, and of consistent action. On the day when the inflammation was most troublesome, I breathed three large doses of nitrous oxide. The pain always diminished after the first four or five inspirations; the thrilling*

[1] It may be of interest to note that Davy later went on to offer a young Michael Faraday his first scientific appointment at the Royal Institution in 1813.

> *came on as usual, and uneasiness was for a few minutes, swallowed up in pleasure. As the former state of mind however returned, the state of organ returned with it; and I once imagined that the pain was more severe after the experiment than before.*" [3]

It was with this observation that nitrous oxide became the first anaesthetic of modern medicine.

1.2 The role of the anaesthetist

The science of anaesthesia has progressed considerably since Davy's first experiments to the point where the administration of such drugs requires specialist training. The anaesthetist is responsible for deciding whether a patient is fit for anaesthesia [5]. The modern anaesthetist plays a critical and varied role in healthcare delivery.

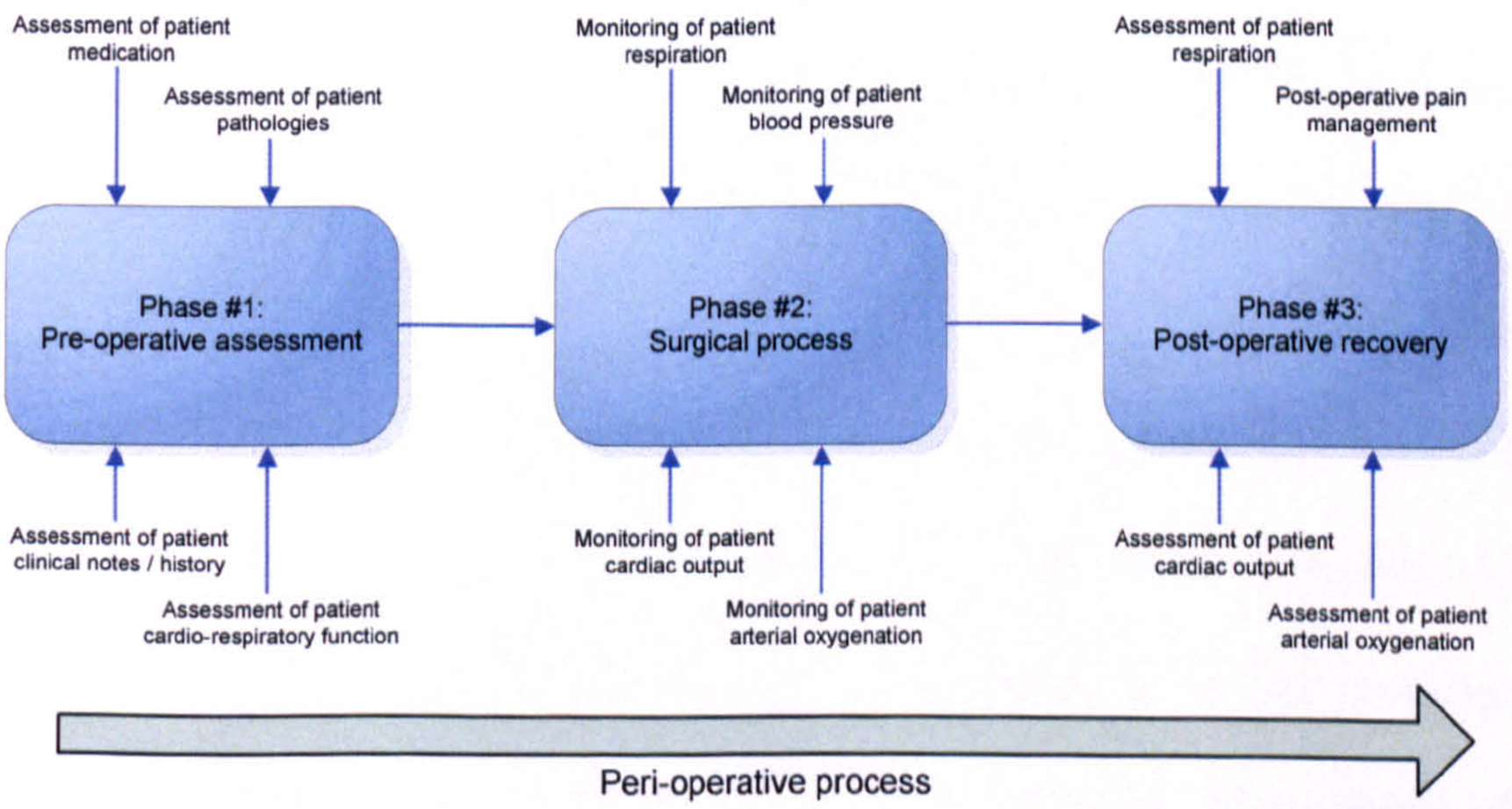


Figure 1.2. The main stages of the peri-operative process, including the role of the anaesthetist during each stage.

The anaesthetist 'follows' the patient through the entire peri-operative process: that is attending the patient when they first enter hospital; through the surgical process; and during post-operative recovery. This, however, was not always the case and the role of the anaesthetist has evolved over recent years. As Notcutt [6] observed, the traditional function of the anaesthetist was once clear: it was their job to render the patient insensible for the surgeon to operate. With the complexity of surgery increasing and the average patient becoming older due to increasing life expectancies [7], the traditional function of the anaesthetist is becoming displaced in favour of a more varied responsibility [6]. Figure 1.2 illustrates the peri-operative process and the typical involvement of the anaesthetist.

1.2.1 The pre-operative process

In the pre-operative process (*i.e.*, prior to the operation) the anaesthetist is responsible for assessing the patient's suitability for anaesthesia [5,8]. The anaesthetist decides whether the general physical state of the patient is robust enough for the stresses, *etc.*, which are a natural part of surgery, by applying a series of tests and clinical scoring tools. Such tools include blood pressure measurements, pulse oximetry measurements, lung function tests, body weight measurements, *etc.*, (these tests, and more, are discussed in Section 4.1.2). Frequently the anaesthetist may have to refer the patient for further investigative procedures before a decision can be made. A complete assessment of the patient's cardiovascular, respiratory and circulatory systems must be made as it is these systems in which the anaesthetist will be intervening during surgery.

1.2.2 The surgical process

It is during the surgical process that the anaesthetist fulfils their more traditional role. The anaesthetist renders the patient insensible by placing them in a reversible coma using an appropriate mixture of drugs. The drugs and tools have changed substantially over recent years [6] and our understanding of human physiology has advanced at a steady pace thereby increasing dependency on the anaesthetist to care for the patient during surgery. Vohra comments that "*its not just one big syringe and one small syringe followed by a cup of coffee and* The Times *crossword*," [9]. They liken the anaesthetist to the patient's 'replacement brain' in that they regulate and control various critical bodily functions of the patient that the patient's brain would normally be responsible for. The anaesthetist's surgical role can basically be summarised in the context of the following main points [9]:

- Assessing airway and respiration competence: determining the best type of airway (*e.g.*, laryngeal mask, tracheal, bronchial tube, *etc.*) for the patient; determining the best mode of breathing (*e.g.*, spontaneous, assisted, *etc.*); and determining blood gas composition.
- Continuous appraisal of circulatory and cardiac competence: determining the appropriate blood pressure and heart rate parameters; and deciding on the adequacy of blood volume, haematocrit, acid base balance, clotting function, ionic composition, *etc.*

The anaesthetist must be furnished with an adequate understanding of the patient's physiological condition prior to surgical intervention, in order to realise

optimal systemic control during surgery. It is this requirement that makes the pre-operative assessment a paramount consideration for the anaesthetist. In this context it should be noted that The Association of Anaesthetists of Great Britain and Ireland states: "*The aim in assessing patients before anaesthesia and surgery is to improve outcome*" [5]. Given the role of the anaesthetist throughout the peri-operative process, attention is now given to the pre-operative assessment.

1.2.3 The importance of pre-operative assessment

The UK National Confidential Enquiry into Patient Outcome and Deaths (NCEPOD)[2] scoring system categorises patients into four groups based on the type of surgery they require.

Category	Description	Surgery timing
NCEPOD 1: Emergency	Immediate life saving operation, resuscitation simultaneous with surgical treatment (*e.g.*, trauma, ruptured aortic aneurysm).	Within one hour.
NCEPOD 2: Urgent	Operation as soon as possible after resuscitation (*e.g.*, irreducible hernia, intussusception, oesophageal atresia, intestinal obstruction, major fractures).	Within 24 hours.
NCEPOD 3: Scheduled	An early operation, but not immediately life-saving (*e.g.*, malignancy).	Within 3 weeks.
NCEPOD 4: Elective	Operation at a time to suit both patient and surgeon (*e.g.*, cholecystectomy, joint replacement).	At time to suit patient & surgeon.

Table 1.1. NCEPOD grades and their accepted definitions [8,10].

Table 1.1 illustrates these grades and includes the accepted definitions of each classification [8,10]. Each NCEPOD group given in Table 1.1 requires different pre-operative investigative procedures to be ordered by the anaesthetist. These

[2] Prior to April 2002, this was called the *National Confidential Enquiry into Peri-Operative Deaths*.

classifications serve to ensure that the patient is thoroughly assessed during the pre-operative assessment phase, as any oversight can have serious repercussions during surgery. For example, failure to pre-operatively identify aortic stenosis in the patient, who is usually asymptomatic, can incur potentially fatal consequences. For example this can result in the anaesthetist monitoring the patient by non-invasive means during surgery thereby overlooking potentially severe cardiac incompetence. NCEPOD has published several case studies in which patients have died post-operatively due to the failure to adequately assess cardiac competence at the start of the peri-operative process [11]. Given this, it becomes imperative to the safety of the patient that the anaesthetist is sufficiently well trained and unburdened such that rigorous pre-operative assessments can be realised.

1.3 Introduction to the problem

In UK National Health Service (NHS) hospitals in 2003 the total number of operations in which a consultant anaesthetist was present was 53% [12]. This is a decrease from 2001 in which 77% of cases had a consultant anaesthetist present [11]. In addition to this the 2003 NCEPOD annual report expressed concern at the "*low level of consultant supervision of trainee anaesthetists*" [12].

The operating theatre is not the only place which has seen a shift in the delivery and management of anaesthesia. In 2001 the US Health Care Financing Administration announced its intention to remove the requirement that nurse anaesthetists be supervised by a qualified physician during the administration of anaesthetics to a patient [13]. The announcement generated debate about the

ability of nurses to perform this function which is normally discharged by the anaesthetist. The main concern is whether nursing grade staff possess the experience and clinical knowledge to adequately perform in this role.

With the expanding role of the anaesthetist bringing with it new pressures and the trend towards assigning additional roles to nursing staff, a pertinent question is whether nursing grade staff are appropriate for making decisions which were once the responsibility of the anaesthetist. Anaesthetists are perceived to be overburdened, and in an attempt to reduce their workload healthcare providers are seeking ways of reallocating some of their responsibilities. Given the concerns expressed by NCEPOD over the level of supervision provided for existing anaesthetists, the risks associated with employing less experienced staff to discharge the anaesthetist's duty can only logically by implication be transferred to the patient.

1.4 Research objectives

The primary objective of this work is to investigate the potential application of intelligent systems to support the anaesthetist in pre-operative assessments. Additional sub-objectives will be outlined later in this section.

Let us begin by considering the term 'intelligent systems'. Kasabov describes intelligent systems as comprising "*methods, tools, and systems for solving problems that normally require the intelligence of humans*" [14]. A more detailed exposition of intelligent systems is provided in Chapter 3. For the purposes of this section it is sufficient to describe intelligent systems as being

systems which can learn; can emulate human inductive reasoning; and which can perform pattern recognition and classification tasks.

Given the overarching objective here; to investigate the potential of intelligent systems for assisting the anaesthetist in making pre-operative assessment decisions, let us now refine this into several sub-objectives in a modular sense. The key ones are:

1. to research and develop new techniques for assessing the competence of the cardiac and respiratory systems of the patient by listening to heart and lung sounds;
2. to research and develop new techniques for assessing the patient's arterial oxygen saturation which can account for artefacts in the observed data; and
3. to research and develop new techniques for determining the patient's suitability for surgical anaesthesia which accords with the anaesthetist's decisions.

Upon first inspection the three points above may appear to be divorced from the primary objective. In clarifying these points we must firstly consider the anaesthetist's decision-making strategy. When presented with a range of information concerning the physiological state of a patient (*e.g.*, blood pressure, blood oxygenation, *etc.*) the anaesthetist assesses each of these *metrics* individually prior to forming a decision. It is these intermediate decisions – and how they are made – to which the aforementioned objectives pertain.

Objective 1, which pertains to the heart and lung sounds, is a complex diagnostic manoeuvre for the anaesthetist. For example, Wipf *et al.*, [15] identified that experienced physicians were capable of identifying respiratory pathology from the lung sounds with a sensitivity of 60-70%. The objective of this work is therefore to develop tools to help improve this performance in terms of sensitivity of cardio-respiratory pathology classification. Objective 2 pertains to adequate continuous pulse oximetry assessment (*i.e.*, blood oxygenation). A detailed account of the motivation towards the research into this objective is given in Section 6.2. Suffice it to say at this juncture that assessing arterial oxygen saturation is integral to the accurate appraisal of a patient's suitability for surgical anaesthesia [5]. Objective 3 pertains to the research and development of novel classification techniques to assist the anaesthetist. The system is referred to as a Decision Support Aid (DSA). The purpose of the DSA is to classify patients into one of the following three categories depending upon their clinical circumstances [5]:

i. *Suitable for surgical anaesthesia.* Although suitability for anaesthesia partially depends upon factors such as the type of anaesthetic (local, general, sedative, *etc.*), type and duration of surgery; whether a patient is suitable primarily depends upon whether their biometric parameters are within an 'acceptable range' [5]. Acceptable ranges are discussed in Section 7.3.

ii. *Possibly suitable for surgical anaesthesia conditional on further investigative tests.* This category stipulates that a patient is deemed provisionally suitable for surgical anaesthesia. It is usually applied if one

or more tests reveal abnormalities (*e.g.*, aortic stenosis) which require further investigation but may not preclude surgery. The anaesthetist's 'tests' will be discussed in Section 4.1.2.

iii. *Unsuitable for surgical anaesthesia.* Patients in this category typically fail one or more of the anaesthetist's tests, and whereby the risk of surgical mortality outweighs the benefits of the operation for which they are scheduled. While this does not necessarily preclude them from undergoing their operation, they may require prior surgical intervention or treatment for the identified abnormality. This will be discussed further in Chapters 5 and 6. Examples of such abnormalities would include excessively obese patients with respiratory disorder, or those with malignant hypertension (*i.e.*, dangerously high blood pressure).

Given the above three classifications, the requirement is that the DSA must make determinations as to the patient's suitability for surgical anaesthesia when furnished with input data pertaining to their physiological condition. Linked to this is the requirement that the DSA must be able to linguistically express to the anaesthetist its classifications and how it arrived at them. As we will show in Chapters 5, 6 and 7, certain standard intelligent system techniques do not have this ability and so new techniques must be researched and developed.

Having presented an overview of the key objectives of this research work, Section 1.6 later presents an overview of the subsequent chapters and how they pertain to the realisation of the objectives.

1.5 Contributions to knowledge

In the context of the objectives outlined in Section 1.4, this thesis aims to provide a contribution to scientific knowledge via the demonstration of intelligent systems in the provision of healthcare. In particular, it will show that the complex task of cardio-respiratory pathology detection can be effectively performed using intelligent system techniques, and that these techniques can afford superior levels of accuracy and sensitivity than the average physician. It is however not the objective of this work to replace the physician, but rather to provide the physician with high-quality information to help in the decision-making process. Furthermore it will be demonstrated that the information pertaining to the anaesthetist's domain can be encapsulated using intelligent systems. It is the opinion of the author that the work reported here is a 'significant and original contribution to knowledge'.

1.6 Thesis outline

In brief, this thesis contains a review of selected clinical applications of intelligent systems. This is supplemented with an overview of the theory underpinning intelligent systems which is intended to clarify their operation. Methods of data acquisition and analysis are discussed, alongside the application of intelligent systems in disseminating information and knowledge interpretation. New applications of intelligent systems are presented, as are novel techniques for the analysis of arterial oxygenation. The thesis finishes with the creation of a new, hybridised classification mechanism to perform as the DSA. In conclusion the problems facing decision support are discussed and further areas of

investigation are recommended. The contents of each chapter is summarised below.

In Chapter 2 a review of intelligent systems in healthcare is provided. This is foremost intended to assess the capabilities of various intelligent system paradigms in a medical context. Methods of delivering improved healthcare using these techniques are discussed, as are the potential pit-falls associated with each technique.

Chapter 3 gives a more detailed description of each intelligent system paradigm selected for consideration in this research. This includes a technical description of each paradigm. The chapter illuminates the merits and limitations of each approach and is intended to provide the background for the work presented in subsequent chapters.

Given detailed reviews of the theory of intelligent systems and their applications in healthcare in previous chapters, Chapter 4 begins by describing the purpose of our data collection, followed by the selected data acquisition methods. Each biometric collected is discussed and its relevance to the research is expounded. The equipment utilised throughout the experiments is presented as is any additional development work undertaken to acquire relevant information from the equipment.

Chapter 5 presents the work undertaken to develop a novel means of detecting cardio-respiratory disease processes using intelligent systems with heart and lung

sounds. Here strategies for audio data processing are discussed as well as methods for extracting sufficient, appropriate information from the data on which to base classifications as to the patient's cardio-respiratory state. Various classification techniques are applied and a combined classification approach is finally implemented.

Chapter 6 addresses issues regarding the acquisition of reliable pulse oximetry data. Motivation is presented for the application of intelligent systems to identifying erroneous pulse waveforms. The chapter presents two novel techniques for extracting artefact-corrupted pulse waveforms thus improving confidence in pulse oximetry measurements.

Given that the two previous chapters addressed pertinent issues concerned with the acquisition and reliable processing of cardio-respiratory sounds and pulse oximetry measurements, Chapter 7 discusses the principal means by which these data could be fused with other biometric values to form a DSA. The main aspects of fuzzy intelligent systems are discussed and an application of these systems is presented in the context of clinical data classification. The DSA is subsequently evaluated against other techniques. The chapter concludes by considering the applicability of this technique to the classification of clinical data in order to support the anaesthetist.

Chapter 8 summarises the results of the research presented in Chapters 5, 6 and 7. The relative merits and limitations of the approaches are discussed and suggestions for improvement are provided. The chapter provides the principal

conclusions of the research and suggests how the system may be expanded to take on more varied roles in the delivery of healthcare.

1.7 Summary

To summarise this chapter, the clinical function of the anaesthetist has been introduced to be a varied and complex one, usually requiring the specialist to perform critical tasks across the peri-operative process. The literature suggests that as healthcare develops, pre-operative assessment may be adversely affected by the substitution of anaesthetists with potentially less experienced staff. So an implicit part of this work is to ameliorate the possible consequences this may have for the good of the patient and to the healthcare system as a whole. In order to do this, the work presented here researches and develops various intelligent system solutions. The thesis outline has been discussed and new contributions of this research presented herein identified.

References

[1] Cropper, W.H., *Great physicists: the life and times of leading physicists from Galileo to Hawking*, Oxford University Press, Oxford, 2001.

[2] Sprigge, J.S., Sir Humphry Davy; his researches in respiratory physiology and his debt to Antoine Lavoisier, *Anaesthesia*, **57**, 2002, pp.357-364.

[3] Davy, H., *Researches, chemical and philosophical; chiefly concerning nitrous oxide, or dephlogisticated nitrous air, and its respiration*, J. Johnson, London, 1800.

[4] Gillray, J., *New discoveries in pneumaticks* (sic)*: or, an experimental lecture on the power of air*, 1802; reprinted in Golinsky, J., *Science as public culture: chemistry and enlightenment in Britain, 1780-1820*, Cambridge University Press, Cambridge, 1992.

[5] *Pre-operative assessment: the role of the anaesthetist*, The Association of Anaesthetists of Great Britain and Ireland, London, 2001.

[6] Notcutt, W., Anaesthesia, what's in a name?, *Anaesthesia*, **56**, 2001, pp.273-274.

[7] *Then and now: the 2000 report of the National Confidential Enquiry into Peri-Operative Deaths*, National Confidential Enquiry into Peri-Operative Deaths, London, 2000.

[8] *National good practice guidance on pre-operative assessment for inpatient surgery: operating theatre & pre-operative assessment programme*, NHS Modernisation Agency, London, 2003.

[9] Vohra, A., The role of the anaesthetist: replacement brain, *Anaesthesia*, **56**, 2001, pp.272-273.

[10] *The 2002 report of the National Confidential Enquiry into Peri-Operative Deaths*, National Confidential Enquiry into Peri-Operative Deaths, London, 2002.

[11] *The 2001 report of the National Confidential Enquiry into Peri-Operative Deaths*, National Confidential Enquiry into Peri-Operative Deaths, London, 2001.

[12] *The 2003 report of the National Confidential Enquiry into Peri-Operative Deaths*, National Confidential Enquiry into Peri-Operative Deaths, London, 2003.

[13] Josefson, D., Unsupervised nurses may soon give anaesthetics in United States, *The British Medical Journal*, **320**, 2000, pp.959.

[14] Kasabov, N.K., *Foundations of neural networks, fuzzy systems, and knowledge engineering*, MIT Press, Cambridge, MA, 1996.

[15] Wipf, J.E., Lipsky, B.A., Hirschmann, J.V., Boyko, E.J., Takasugi, J., Peugeot, R.L. and Davis, C.L., Diagnosing pneumonia by physical examination: relevant or relic?, *Archives of Internal Medicine*, **159**(10), 1999, pp.1082-1087.

CHAPTER 2

Intelligent Systems in Healthcare

This chapter presents an overview of some of the more common types of intelligent systems which have been applied in healthcare. In each case, a review of how these techniques have been applied to healthcare to solve particular problems is outlined.

2.1 Introduction

Whilst there is no formal definition, it can be said that an intelligent system is a system which can be considered to be able to learn from and adapt to input stimuli [1]. In particular they can solve problems by applying knowledge gained from past experiences. Intelligent systems are varied and encompass Artificial Neural Networks (ANNs), Fuzzy Logic (FL), Genetic Algorithms (GAs) and hybrid approaches such as Neural Fuzzy Systems (NFS), some of which are discussed here. A detailed account of ANNs can be found in Chapter 3; FL in

Chapters 6 and 7; and GAs in Chapter 7. Figure 2.1 presents the three main intelligent system paradigms and shows possible areas of hybridisation.

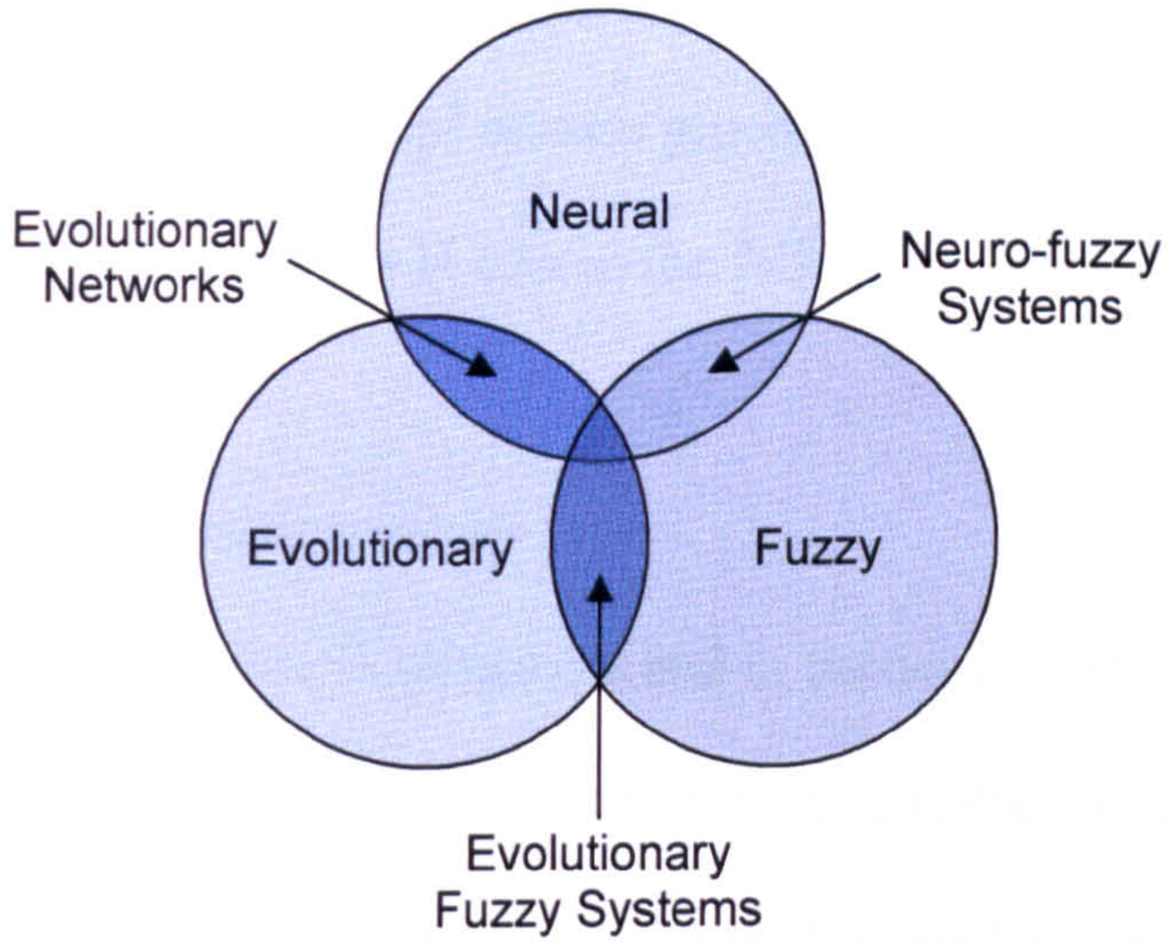


Figure 2.1. The neural, evolutionary, and fuzzy paradigms with areas of possible hybridisation.

The following sections focus on clinical applications of ANNs. In particular, in Section 2.2, we discuss applications of Multi-Layer Perceptrons (MLPs); Radial Basis Function Networks (RBFNs) in Section 2.3; and Probabilistic Neural Networks (PNNs) in Section 2.4. Some clinical applications of NFS are given in Section 2.6.

2.2 Clinical applications of the Multi-Layer Perceptron

MLPs are a particular type of ANN favoured in classification problems as they are relatively simple to use, and have been shown to operate well on linear and non-linear problems [1]. An in-depth discussion of MLPs and linear separability is presented in Section 3.2.

There are numerous publications which report on the application of MLPs to medical problems; for example see reviews by Baxt [2], Itchhaporia *et al.*, [3] and Lisboa [4]. These largely concentrate on prognosis and prediction, Coronary Heart Disease (CHD) and general disease process classification. We will discuss some of these application areas in turn in the following sections.

2.2.1 MLPs in medical prognosis

A major area of application is prognosis and in particular, surgical risk analysis: Lisboa *et al.*, [5] utilised the MLP to determine prognostic risk groups of patients undergoing breast cancer surgery in order to best target post-operative resources. They found that the MLP was more specific in allocating patients to prognostic risk groups than their clinically-accepted benchmark proportional hazards model [5]. Santos-García *et al.*, [6] applied an ensemble of MLPs to the prediction of post-operative morbidity after pulmonary resection for lung cancer *bronchogenic carcinoma*. The ensemble consisted of 100 different MLPs trained on patient data (*i.e.*, age, gender, body mass index, *etc.*) and operative information, (*i.e.*, extent of resection, peri-operative blood transfusion, *etc.*). They found that the ensemble gave 98% accuracy for the prediction of cardio-respiratory morbidity. The authors concluded that patient criteria as well as operative criteria have a direct impact on the predicted post-operative morbidity and mortality, a point which is supported by Bernard *et al.*, [7,8].

Prognostic applications range from surgical risk analysis to diabetes prediction and CHD prediction [9,10,11]. CHD prediction using MLPs has generated considerable interest; in particular examination of the prognostic survival curve

based on patient lifestyle variables [11] and risk prediction based on biomedical indicators [12]. Ohno-Machado *et al.*, [11] applied MLPs to predict the probability of CHD development over a period of time. They applied MLPs sequentially such that the prediction of CHD development at one instant in time was used to bias a subsequent MLP to the probability of CHD development at a later instant. They incorporated patient criteria (*i.e.*, age, gender, *etc.*) as well as patient lifestyle variables, such as tobacco use, into their prediction model and concluded that the MLP ensemble constituted a good alternative to existing survival analysis models.

In a different study Azuaje *et al.*, [12] explored the prediction of CHD by assessing Heart Rate Variability (HRV) with an MLP. HRV dynamics were assessed from the R-R interval of the cardiac QRS wave [12]. They approached the prediction of CHD development from the biosignal direction, as opposed to the patient criterion basis adopted by Ohno-Machado *et al.*, [11]. Azuaje *et al.*, [12] found that an MLP could predict the onset of CHD from small HRV datasets with a sensitivity of 90%.

2.2.2 MLPs in disease management

As well as detection and prediction of CHD, MLPs have also been applied to disease management, such as determining the suitability of medical procedures and treatments. For example, Mobley *et al.*, [13] employed the MLP to try to determine the suitability of coronary angiography – which is an invasive diagnostic manoeuvre. They found that when tested with 100 historical cases the MLP identified 81 patients who had stenosis and required coronary angiography.

Of the remaining 19 patients asymptomatic of stenosis, the MLP identified 9 who did not require the procedure. Mobley *et al.*, concluded that "*The results demonstrated that, given a multiplicity of input data or risk factors for coronary artery disease, artificial neural networks could nonetheless become valuable tools in identifying patients who do not need coronary angiography*" [13].

2.2.3 MLPs in general disease process classification

There is quite a large number of publications that report on the success of MLPs in prognosis and prediction. This is rivalled by the number of publications reporting similar successes in disease process classification. Topical subjects such as cardio-respiratory analysis and cancer detection have attracted the majority of research in recent years and this is reflected in the available literature [2,3,4]. For example, cardiac disease process detection has been considered by authors such as Cathers [14] and Turkoglu *et al.*, [15] who have applied MLPs to the analysis of auscultated heart sounds. Cathers [14] applied the MLP to the analysis of time-domain heart sounds. Cathers found that by increasing the size of the MLP, the classification rate was improved to the extent that 95% of the training dataset was successfully separated. Cathers proposed that time-frequency analysis in the form of wavelet transforms may afford a more comprehensive approach to heart sound analysis. Turkoglu *et al.*, [15] utilised wavelet decomposition of auscultated heart sounds as a processing technique prior to MLP analysis in order to develop a sensitive classification system. They trained their MLP on wavelet decompositions of Doppler heart sounds. Their study was motivated by the effectiveness with which they perceived that wavelet analysis could capture heart sound information. The authors attained a

classification accuracy of 94.5% with abnormal subjects, and 94% accuracy with normal subjects.

Whilst Cathers [14] and Turkoglu *et al.*, [15] attained classification accuracies in excess of 90%, Ölmez *et al.*, [16] commented that the MLP suffered from problems concerning local minima in the available datasets (we will discuss this in more detail in Section 3.2.1), is slow at learning the 'data features' and suffers from the requirement that the MLP architecture be predefined at the outset. These observed problems, whilst they are technically true, can be overcome, or at least ameliorated, by for example: (i) the appropriate selection of MLP parameters; (ii) increasing the computational power of the processing platform; and (iii) the appropriate selection of the architecture prior to training using techniques such as Cascade Correlation [17] (this is discussed in more detail in Section 5.4.2).

2.2.4 MLPs in cardio-respiratory disease process classification

Consideration has also been given to the analysis of respiratory abnormalities and disease processes. This has resulted in MLP-based classifiers that are trained to operate by analysing auscultated lung sounds, for example. Both Rietveld *et al.*, [18] and Leonhardt *et al.*, [19] applied the MLP to discriminate between respiratory disease processes. The authors assessed different biosignals, namely respiratory auscultation and Tidal Breathing Flow Volume (TBFV), respectively. Both applied the MLP in the same configuration with the same training algorithm, and although tackling the problem of classification using different signals, they found that the MLP could significantly outperform human

experts. Rietveld *et al.*, found that using the Fourier power spectrum of lung sounds to identify airway obstruction resulted in 43% of their test data being correctly classified by the MLP. Leonhardt *et al.*, found that TBFV analysis resulted in the MLP correctly classifying 70% of the test data. Both of these findings suggest that respiratory disease processes can be identified in a variety of ways and that the appropriate selection of the 'indicating biosignals' is imperative for accurate diagnosis.

MLP-based analysis of the respiratory system is not confined to disease process detection but is also applicable to other abnormalities. Wilks *et al.*, [20] approached the subject of Sudden Infant Death Syndrome (SIDS) by identifying a correlation between changes in the pattern of respiration with the onset of hypoxaemic episodes. Such correlations were achieved by simultaneous MLP analysis of the respiration pressure and breathing pattern of the patient. In a similar way, Oud [21] showed that it was possible to establish a relationship between the obstruction of the airways (through FEV_1 measurements) and the sounds produced by the lungs. In their study the airway obstruction was allergen provoked and they observed that the MLP could successfully identify the obstruction characteristics through assessment of the consequent lung sounds. Wilks *et al.*, concluded that it is possible to predict changes in oxygen saturation earlier with an MLP than with existing techniques [20].

2.2.5 MLPs in disease identification

Whilst cardio-respiratory abnormalities and disease processes account for a considerable amount of interest in the application of MLPs to medical

classification and prognostic determination, other applications are also presented in the available literature. Two other areas are cancer detection and classification, and ECG analysis which have both been of significant interest over recent years. Two publications representative of this are by Antal *et al.*, [22] and Papaloukas *et al.*, [23]. In the former, Antal *et al.*, discriminated between benign and malignant tumours by incorporating *prior beliefs* into the MLP to form an informative MLP model. Their prior beliefs were probability statements concerning the differences between benign and malignant tumours. The authors concurred with the general notion that selection and coding of the relevant training data features is of central importance when developing an MLP for medical analysis in order to attain not only highly accurate but also highly sensitive systems. This notion was also expounded by Papaloukas *et al.*, who applied MLPs to the detection of *myocardial ischemia* (defined by insufficient blood supply to the heart muscle [23]) in long-duration ECGs. Their results showed a 90% sensitivity of the system towards ischemic episodes and they attribute this achievement to numerous factors, including for example the reduction in size of the dataset whilst retaining its informative features (*i.e.*, components that contain the most information).

2.2.6 Discussion

A recurring theme throughout the presentation of the above literature is that of appropriate data coding, *i.e.*, an appropriate data representation scheme. Whilst it is true of most ANN architectures, the MLP will not perform well with seemingly homogeneous datasets. These types of problems usually require different ANN solutions. When considering highly non-linear separable datasets

a popular ANN architecture used is the RBFN. Many medical imaging problems tend to be non-linear in ways which are better suited to the use of RBFNs. Whilst the RBFN architecture is discussed in Section 3.3 some of its reported applications are presented in the following section.

2.3 Clinical applications of the Radial Basis Function Network

It can be seen from the previous sections that the MLP has found numerous applications within the field of clinical data prediction, prognosis and classification. An alternative approach is discussed presently. As we will discuss in Chapter 3, the neuronal activation functions of the hidden layer of a RBFN are locally active, or locally tuned. This term is used to describe the behaviour of a Gaussian function in response to an input datum which is similar to the function's local maximum. Input data which are substantially different from the function's centre will prompt a low (*i.e.*, near zero) response and it is this property that makes the RBFN an attractive neural network for some medical applications.

2.3.1 RBFNs in medical imaging

One particular area in which RBFNs have been found to be applicable is that of medical imaging. In particular, frequent applications of the RBFN involve medical images such as Magnetic Resonance Imaging (MRI) scans or X-ray images of tumours and other 'growths'. For example, Fornefett *et al.*, [24] extended the Gaussian function to incorporate compact support (*i.e.*, spatial limitation) and found that networks of these radial basis functions can perform well with local changes (*e.g.*, scanner-induced deformations, differences in

patient anatomy, *etc.*) in medical images. The authors of the work noted that these localised differences in medical images were particularly difficult for conventional image recognition paradigms to solve. They applied RBFNs to the localisation and identification of brain tumour images and attained higher degrees of accuracy than with existing automated systems.

Not only are RBFNs used in classification of images, but technical issues arising from medical equipment have also been addressed using these ANN implementations. For example, Lai *et al.*, [25] applied RBFNs to determine the re-mapping of MRI images for display on a computer screen. The objective was to reduce the colour depth of the image whilst retaining the features by appropriate adjustment of the colours and contrast. The RBFNs were used in an ensemble and the authors concluded that the processed images could be successfully re-mapped and analysed by clinicians.

Identification and classification has also been of substantial interest and RBFNs have been applied to a range of areas from the detection of cancers to the isolation of particular image features. For example Behloul *et al.*, [26] applied RBFNs to the identification of the myocardium in noisy Positron Emission Tomography (PET) images. The authors trained the RBFN to generate fuzziness measures of the image and to minimise this fuzziness such that a high contrast image for the myocardium could be obtained with respect to the noisy background. They concluded that the RBFN could segment the image of the myocardium to a degree of sensitivity comparable to an expert human examiner,

thereby allowing further computerised methods of image analysis to be performed.

2.3.2 RBFNs in medical prediction

Non-linear modelling of biological processes is also ideally suited to the RBFN due to the nature of Gaussian functions. One particular area of interest has been in the assessment and prediction of Heart Rate Variability (HRV). HRV is an important factor in assessing cardiac activity yet it remains difficult to predict due to the complexity of the controlling mechanisms such as the Autonomic Nervous System (ANS). Investigation into the non-linearity of heart rate dynamics and the prediction of HRV from these dynamics has resulted in mixed reports, in which authors identify partial success. In a study by Bezerianos *et al.*, [27] normal heart rate dynamics were modelled by RBFNs and compared to the same subjects after ANS blockade. The authors found that normal ANS control over HRV dynamics resulted in the RBFN attaining high prediction accuracies (0.009 mean square prediction error). Dysfunctional ANS yielded poorer results (0.08 mean square prediction error) [27]. Total ANS blockade resulted in the inability of the RBFN to predict the heart rate. The authors concluded that the results obtained could support the hypothesis that HRV dynamics are non-linear and chaotic in nature. These claims are partially supported by Haque *et al.*, [28] in that the authors determined the HRV controlling characteristics to be non-linear. The authors used the RBFN to predict HRV under ANS control and compared their system with a non-linear autoregressive model. They found that the RBFN outperformed the autoregressive model in 2 out of 5 cases of heart rate prediction.

2.3.3 Discussion

It can be concluded from the presented literature that the RBFN is ideally suited to situations requiring locally receptive functions such as medical image analysis; and scenarios such as non-linear predictions of complex biological processes. The advantages of the RBFN over the MLP in these areas seem to arise from the Euclidean distance manner in which the network responds to new data. The Gaussian functions used by the RBFN respond to data based on its similarity to the function's centroid. RBFNs are also naturally disposed towards probability density estimation which makes them ideal in classification scenarios [1]. A natural extension to this is an ANN paradigm that is firmly rooted in probability density estimation; one such example is the Probabilistic Neural Network (PNN). At this juncture it is important to note that the PNN, (which will be discussed in more detail in Section 3.4), can only be employed as a classifier and it cannot function as a prediction mechanism [29].

2.4 Clinical applications of the Probabilistic Neural Network

The PNN was introduced in 1990 [29] and is a relatively new neural network architecture compared to the MLP and RBFN and as such few authors have yet reported on the clinical applicability of this architecture. The PNN operates by constructing a plurality of Gaussian functions describing the distribution of the training data features. This principle is similar to that for the RBFN, but whereas the RBFN attempts to learn the partitions between classes, the PNN forms class probability densities.

In the paper by Shan *et al.*, [30] the PNN was applied directly in the role of a classifier to detect cancers in patients based on nucleosides in their urine samples. In comparison with methods such as Linear Discriminant Analysis (LDA) [1] the authors found that the PNN could differentiate between healthy patients and those with cancer with an accuracy of 85% (LDA achieved 80%). The PNN was presented with a binary (*i.e.*, 2-class) partially homogenous dataset representing patients with and without cancer. Shan *et al.*, further concluded that the PNN demonstrated high sensitivity and specificity (80% and 90% respectively) towards the classification of cancers [30].

The range of classification problems that the PNN could be applied to is large and varied. A study by Piliouras *et al.*, [31] concluded that the PNN could be a good patient management tool for the assessment of stroke risk resulting from atherosclerotic carotid plaque. The PNN was trained to classify patients into prognostic risk groups based on ultrasound images. The trained PNN successfully identified 23 of the 24 high-risk cases and 29 of the 32 low-risk cases. Whilst Piliouras *et al.*, advocated the necessity for extensive and thorough testing of their approach, they concluded that it had the potential to be a valuable second opinion in the classification of risk groups [31].

Given that the PNN alongside other ANNs such as the MLP and RBFN can be reformulated in the Bayesian framework [1], the classifications or predictions resulting from these systems can be argued to be probabilistic assertions. It is possible to assess the ANN responses in terms of basic belief assignments, *i.e.*, the level of belief that is attributed to classifying some quantity x as class k. By

considering the responses of multiple neural network implementations to a given input, it is possible to improve the classification (or predictive) accuracy, of the ensemble by combining these responses at the decision-theoretic level. Piliouras *et al.*, [31] considered this in their work as a means for increasing the accuracy of their classification system. Methods of evidence combination are considered later in more detail in Section 3.5. The PNN and RBFN are generally more efficient than the MLP at handling homogeneity within datasets due to their ability to mix Gaussian functions of all classes to obtain optimal discrimination.

Hybridised intelligent systems have also found applications in biomedical engineering, in particular evolutionary systems such as GAs, Genetic Programs (GPs) and Neural Fuzzy Systems (NFS). The following sections present an overview of a few of these pertinent applications; in the context of this work.

2.5 Clinical applications of Evolutionary Systems

This section discusses clinical applications of GAs and GPs in the context of healthcare delivery. The theory underpinning these techniques is given in Section 7.5.2 and Section 7.5.3. The available literature is replete with reports on the application of GAs to medical classification scenarios; primarily concerning the evolution of ANN-based classifiers to diagnosis [32,33]. In a paper of particular relevance, Vinterbo *et. al.*, [34] discussed the problem of multi-disorder diagnosis *i.e.*, that expert systems frequently assume that multiple-disorders do not co-occur in a single patient. The focus of the paper was on a means of evolving a classifier that did not attempt to explain all of the symptoms pertaining to a disorder but to a plurality of input-output ANN mappings that

could account for multiple disorders. The authors concluded that the GA approach afforded a good framework for managing the ANN in performing these multi-diagnoses. For an in-depth discourse on the subject of evolutionary techniques in medicine the reader is referred to [35].

In their paper Tsakonas *et. al.*, [36] applied the GP technique to the evolution of fuzzy classifiers. They examined two classification problems: diagnosis of the subtype of aphasia (impairment of the power to use or pronounce words); and pap-smear examinations. In the former case the authors applied GP to the evolution of a fuzzy rule-base and concluded that the GP-evolved fuzzy system could attain classification accuracies of 88.5% on the training dataset, and 79.1% on the test dataset. When applied to pap-smear examinations, the GP-evolved fuzzy system that attained an average classification accuracy of 90.3% on the test dataset. The authors concluded that the GP approach to fuzzy system evolution offered a more comprehensible rule-base than competitive expert systems like C4.5 [36]. Other methods of rule-base construction and fuzzy systems learning, in the medical context, is given in the following section.

2.6 Clinical applications of Neural Fuzzy Systems

This section focuses on clinical applications of NFS. The fuzziness in NFS is derived from the fact that real-world classes and concepts are fuzzy rather than crisp, thus potentially resulting in imprecision and ambiguity in the classification process. A complete account of Fuzzy Set Theory (FST) and ensuing NFS can be found in texts such as [37,38]. In their review, Mahfouf *et al.*, [39] noted that fuzzy systems provide an excellent approach to defining inexact medical entities

and affords the user a means for approximate reasoning with medical data. It is these qualities that have drawn researchers to the field of NFS in medicine, and some of the results of these endeavours are presented here.

Fuzzy logic [37], and more recently NFS, has found numerous applications in the field of medicine and biomedical engineering due to the inherent fuzziness in medical data; uncertainty in the co-occurrence of medical events; and incompleteness of medical theories [40]. ANNs can be used to attempt to resolve these issues, but when combined with fuzzy representations of medical variables and data, these hybridised approaches can offer potentially superior levels of performance in certain scenarios. In one such approach, Allen *et al.*, [41] trained an MLP to attempt to identify latency in delivering anaesthesia and the response from the patient. The MLP was trained to generate fuzzy outputs that a fuzzy controller could use in administering the anaesthetic. The authors concluded that the MLP and fuzzy controller could adequately maintain the depth of anaesthesia of a patient.

Control systems based on real-time biosignals is an area in which NFS have found considerable applications. One type of NFS which is popular in this area is the Adaptive Neuro-Fuzzy Inference System (ANFIS) [38,42]. Kwok *et al.*, [43] applied ANFIS to the control of intensive care ventilators. The authors examined 71 clinical scenarios in which patients undergoing ventilator assisted respiration experience changes in respiration pattern. This was achieved by examining the relationships between the inspired fraction of oxygen (FiO_2), the arterial oxygen tension (PaO_2) and the positive end-expiratory pressure. For each

of the scenarios, the responses of eight clinical experts were recorded and used to train both ANFIS and MLPs. They found that the performance of the systems in controlling the ventilator was comparable to the expert clinicians and concluded that adaptive neuro-fuzzy systems can facilitate the modelling of clinicians' knowledge. ANFIS is discussed in more detail in Chapter 7.

Aside from control systems, NFS have also been applied to signal identification. For example the diagnosis of Gastro-Oesophageal Reflux (GOR) is of great interest to paediatric gastroenterologists as it is considered to be one possible reason for sudden infant death syndrome [44]. In their paper Trachterna *et al.*, [45] used intraluminal electrical impedance measurements to allow long-term, detailed, pH-independent registration of GOR in 50 paediatric patients. They trained an ANFIS to recognise GOR and compared this method with existing intraluminal measurement techniques. They found that the ANFIS reduced the time taken to register GOR by 96% and did not suffer any loss in precision. The authors concluded that ANFIS offers a robust and suitable method for the detection of GOR in infants.

2.7 Summary

Intelligent systems have found wide and varied application in healthcare. Different types of intelligent systems confer different advantages and as such have been found applicable in scenarios ranging from prognosis to hardware control of medical imaging. This chapter has sought to present an overview of the medical applications of ANNs and NFS, whilst the following chapters discuss

the technical aspects of these intelligent systems. Fuzzy Logic and evolutionary techniques are discussed in Chapters 6 and 7.

References

[1] Bishop, C.M., *Neural networks for pattern recognition*, Oxford University Press, Oxford, 1995.

[2] Baxt, W.G., Application of artificial neural networks to clinical medicine, *The Lancet*, **346**, 1995, pp.1135-1138.

[3] Itchhaporia, D., Snow, P.B., Almassy, R.J. and Oetgen, W.J., Artificial neural networks: current status in cardiovascular medicine, *Journal of the American College of Cardiology*, **28**, 1996, pp.515-521.

[4] Lisboa, P.J.G., A review of evidence of health benefit from artificial neural networks in medical intervention, *Neural Networks*, **15**(1), 2002, pp.11-39.

[5] Lisboa, P.J.G., Wong, H., Harris, P. and Swindell, R., A Bayesian neural network approach for modelling censored data with an application to prognosis after surgery for breast cancer, *Artificial Intelligence in Medicine*, **28**(1), 2003, pp.1-25.

[6] Santos-García, G., Varela, G., Novoa, N. and Jiménez, M.F., Prediction of postoperative morbidity after lung resection using an artificial neural network ensemble, *Artificial Intelligence in Medicine*, **30**(1), 2004, pp.61-69.

[7] Bernard, A., Ferrand, L., Hagry, O., Benoit, L., Cheynel, N. and Favre, J.-P., Identification of prognostic factors determining risk groups for lung resection, *Annals of Thoracic Surgery*, **70**(4), 2000, pp.1161-1167.

[8] Bernard, A., Deschamps, C., Allen, M., Miller, D., Trastek, V., Jenkins, G. and Pairolero, P., Pneumonectomy for malignant disease: factors affecting early morbidity and mortality, *Journal of Thoracic and Cardiovascular Surgery*, **121**(6), 2001, pp.1076-1082.

[9] Parvis, M., Gulotta, C. and Torchio, R., Evaluation of surgical risks by means of neural networks in the presence of uncertainties, *Measurement*, **23**(3), 1998, pp.171-178.

[10] Park, J. and Edington, D.W., A sequential neural network model for diabetes prediction, *Artificial Intelligence in Medicine*, **23**(3), 2001, pp.277-293.

[11] Ohno-Machado, L. and Musen, M.A., Sequential versus standard neural networks for pattern recognition: an example using the domain of coronary heart disease, *Computers in Biology and Medicine*, **27**(4), 1997, pp.267-281.

[12] Azuaje, F., Dubitzky, W., Lopes, P., Black, N., Adamson, K., Wu, X. and White, J.A., Predicting coronary disease risk based on short-term RR interval measurements: a neural network approach, *Artificial Intelligence in Medicine*, **15**(3), 1999, pp.275-297.

[13] Mobley, B.A., Schechter, E., Moore, W.E., McKee, P.A. and Eichner, J.E., Predictions of coronary artery stenosis by artificial neural network, *Artificial Intelligence in Medicine*, **18**(3), 2000, pp.187-203.

[14] Cathers, I., Neural network assisted cardiac auscultation, *Artificial Intelligence in Medicine*, **7**(1), 1995, pp.53-66.

[15] Turkoglu, I., Arslan, A. and Ilkay, E., An intelligent system for diagnosis of heart valve diseases with wavelet packet neural networks, *Computers in Biology and Medicine*, **33**(4), 2003, pp.319-331.

[16] Ölmez, T. and Dokur, Z., Classification of heart sounds using an artificial neural network, *Pattern Recognition Letters*, **24**(1-3), 2003, pp.617-629.

[17] Fahlman, S.E. and Lebiere, C., The cascade-correlation learning architecture, *Advances in Neural Information Processing Systems*, **2**, (Ed. Touretzky, D.S.), Morgan Kaufmann Publishers Inc., San Francisco, CA, 1990, pp.524-532.

[18] Rietveld, S., Oud, M. and Dooijes, E.H., Classification of asthmatic breath sounds: preliminary results of the classifying capacity of human examiners versus artificial neural networks, *Computers in Biomedical Research*, **32**(5), 1999, pp.440-448.

[19] Leonhardt, S., Ahrens, P., Hofmann, D. and Isermann, R., A decision support system for lung function diagnosis in infants, *Control Engineering Practice*, **5**(10), 1997, pp.1355-1361.

[20] Wilks, P.A.D. and English, M.J., A system for rapid identification of respiratory abnormalities using a neural network, *Medical Engineering & Physics*, **17**(7), 1995, pp.551-555.

[21] Oud, M., Lung function interpolation by means of neural-network-supported analysis of respiration sounds, *Medical Engineering & Physics*, **25**(4), 2003, pp.309-316.

[22] Antal, P., Fannes, G., Timmerman, D., Moreau, Y. and Moor, B.D., Bayesian applications of belief networks and multilayer perceptrons for ovarian tumor classification with rejection, *Artificial Intelligence in Medicine*, **29**(1-2), 2003, pp.39-60.

[23] Papaloukas, C., Fotiadis, D.I., Likas, A. and Michalis, L.K., An ischemia detection method based on artificial neural networks, *Artificial Intelligence in Medicine*, **24**(2), 2002, pp.167-178.

[24] Fornefett, M., Rohr, K. and Stiehl, H.S., Radial basis functions with compact support for elastic registration of medical images, *Image and Vision Computing*, **19**(1-2), 2001, pp.87-96.

[25] Lai, S.-H. and Fang, M., A hierarchical neural network algorithm for robust and automatic windowing of MR images, *Artificial Intelligence in Medicine*, **19**(2), 2000, pp.97-119.

[26] Behloul, F., Boudraa, A., Lelieveldt, B.P.F., Janier, M. and Reiber, J.H.C., Myocardium extraction in positron emission tomography based on soft computing, *Computerized Medical Imaging and Graphics*, **25**(3), 2001, pp.277-286.

[27] Bezerianos, A., Papadimitriou, S. and Alexopoulos, D., Radial basis function neural networks for the characterization of heart rate variability dynamics, *Artificial Intelligence in Medicine*, **15**(3), 1999, pp.215-234.

[28] Haque, M.A., Hasan, M.K. and Tazawa, H., Investigation of the nonlinearity of heart rate dynamics, *Medical Engineering & Physics*, **23**(2), 2001, pp.111-115.

[29] Specht, D.F., Probabilistic Neural Networks, *Neural Networks*, **3**(1), 1990, pp.109-118.

[30] Shan, Y., Zhao, R., Xu, G., Liebich, H.M. and Zhang, Y., Application of probabilistic neural network in the clinical diagnosis of cancers based on clinical chemistry data, *Analytica Chimica Acta*, **471**(1), 2002, pp.77-86.

[31] Piliouras, N., Kalatzis, I., Theocharakis, P., Dimitriopoulos, N. and Cavouras, D., Development of the probabilistic neural network – cubic least squares mapping (PNN-LSM3) classifier to assess carotid plaque's risk, *Pattern Recognition Letters*, **25**(2), 2004, pp.249-258.

[32] Su, F.-C. and Wu, W.-L., Design and testing of a genetic algorithm neural network in the assessment of gait patterns, *Medical Engineering & Physics*, **22**, 2000, pp.67-74.

[33] Liang, H., Lin, Z. and McCallum, R.W., Application of combined genetic algorithms with cascade correlation to diagnosis of delayed gastric emptying from electrogastrograms, *Medical Engineering & Physics*, **22**, 2000, pp.229-234.

[34] Vinterbo, S. and Ohno-Machado, L., A genetic algorithm approach to multi-order diagnosis, *Artificial Intelligence in Medicine*, **18**, 2000, pp.117-132.

[35] Peña-Reyes, C.A. and Sipper, A., Evolutionary computation in medicine: an overview, *Artificial Intelligence in Medicine*, **19**, 2000, pp.1-23.

[36] Tsakonas, A., Dounias, G., Jantzen, J., Axer, H., Bjerregaard, B. and von Keyserlingk, D.G., Evolving rule-based systems in two medical domains using genetic programming, *Artificial Intelligence in Medicine*, **32**, 2004, pp.195-216.

[37] Zadeh, L.A., Fuzzy sets, *Information and Control*, **8**(3), 1965, pp.338-353.

[38] Jang, J.-S.R., Sun, C.-T. and Mizutani, E., *Neuro-fuzzy and soft computing: a computational approach to learning and machine intelligence*, Prentice Hall Press, Upper Saddle River, NJ, 1997.

[39] Mahfouf, M., Abbod, M.A. and Linkens, D.A., A survey of fuzzy logic monitoring and control utilisation in medicine, *Artificial Intelligence in Medicine*, **21**(1-3), 2001, pp.27-42.

[40] Adlassnig, K.-P., The Section on Medical Expert and Knowledge-based systems at the Department of Medical Computer Sciences of the University of Vienna Medical School, *Artificial Intelligence in Medicine*, **21**(1-3), 2001, pp.136-146.

[41] Allen, R. and Smith, D., Neuro-fuzzy closed-loop control of depth of anaesthesia, *Artificial Intelligence in Medicine*, **21**(1-3), 2001, pp.185-191.

[42] Jang, J.-S. R., ANFIS: adaptive-network-based fuzzy inference system, *IEEE Transactions on Systems, Man and Cybernetics: Part B Cybernetics*, **23**(3), 1993, pp.665-685.

[43] Kwok, H.F., Linkens, D.A., Mahfouf, M. and Mills, G.H., Rule-based derivation for intensive care ventilator controls using ANFIS, *Artificial Intelligence in Medicine*, **29**(3), 2003, pp.185-201.

[44] Jolley, S.G., Halpern, L.M., Tunell, W.P., Johnson, D.G. and Sterling, C.E., The risk of sudden infant death from gastroesophageal reflux, *Journal of Pediatric Surgery*, **26**(6), 1991, pp.691-696.

[45] Trachterna, M., Wenzl, T.G., Silny, J., Rau, G. and Heimann, G., Procedure for the semi-automatic detection of gastro-oesophageal reflux patterns in intraluminal impedance measurements in infants, *Medical Engineering & Physics*, **21**(3), 1999, pp.195-201.

CHAPTER 3

Introduction to Artificial Neural Networks

This chapter presents some of the main theory underpinning the principles of ANNs. In particular the MLP, RBFN and PNN architectures are discussed. The purpose of this chapter is to furnish the reader with an understanding of how these ANNs function. The chapter concludes with an overview of Dempster-Shafer Theory (DST).

3.1 Introduction

This section introduces ANNs and begins with an overview of Bayesian Probability Theory. Section 3.1.2 introduces the artificial neurone – the most basic component of an ANN. The subsequent sections present an overview of the MLP, RBFN, PNN and Constructive PNN (CPNN).

ANNs are biologically-inspired statistical methods *i.e.*, they 'learn' probability distributions. An ANN is a collection of neurones that are inter-connected with synapses. The biological inspiration is derived from the fact that the ANN exhibits what is seen to be a crude resemblance to the mammalian brain. The statistical nature of ANNs is derived from their ability to classify or predict data based upon past experience. ANNs can be examined in the context of Bayesian Probability Theory (BPT) which affords a mathematical description of how an ANN operates. The following subsection provides an overview of BPT whilst a more in-depth account can be found in texts such as [1,2].

3.1.1 Bayesian Probability Theory

The Bayesian approach to probability theory offers a robust method to model the statistical distribution of data; to construct prior probabilities about the occurrence of data within the distribution; and to estimate posterior probabilities of occurrence based on previous knowledge [1]. As will be demonstrated in this chapter, ANNs estimate these probabilities during learning.

Let us consider some value x drawn at random from a statistical distribution. We may possess knowledge of the mechanism that generates the distribution. Furthermore, this knowledge may enable us to form statements about the probability of x being some value; for example $P(x>6)=0.7$ *i.e.*, there is a probability of 70% that the value of x will be greater than 6. The equality $x>6$ can be considered a class of values C_k and therefore the probability statement can be written as $P(C_k)=0.7$. These statements are referred to as the *prior probabilities* as they encapsulate our knowledge of the distribution prior to any

observation being made. Let us now consider a scenario in which a classification is made based on the observation x. The objective is to correctly classify some event indicated by x and to minimise the probability of misclassification. The *conditional probability* $P(x|C_k)$ is the probability that the observation x indicates the event given that the event belongs to class C_k. Bayes Theorem expresses the *posterior probability* $P(C_k|x)$ as:

$$P(C_k|x) = \frac{P(x|C_k)P(C_k)}{P(x)}, \tag{3.1}$$

which is defined as the probability of the event belonging to class C_k given the observation x. In (3.1) $P(x)$ represents the probability of occurrence of datum x irrespective of class assignment. It serves as a normalisation factor ensuring that the posterior probabilities sum to unity. Posterior probabilities are of central importance in classification systems insofar that given K posterior probabilities resulting from the measurement of x, a classifier can assign the event to class C_k entailing the highest posterior probability [2].

It will be shown in Section 3.2.2 that ANNs estimate Bayesian posterior probabilities during their learning process [2]. These estimations are determined from the available data in an iterative manner *i.e.*, the posterior probabilities are updated in response to the observed data. These estimations arise through the iterative adjustments made to the ANN neurones during the learning process. The following section introduces the artificial neurone model employed in ANNs.

3.1.2 The artificial neurone model

An ANN is a collection of inter-connected artificial neurones. Generally, input data is presented to these neurones which as a collective perform a mathematical transform, generating output data as a result of these calculations. The artificial neurone was first proposed by McCulloch and Pitts in 1943 [3]. It comprises of three components: the m inputs $\mathbf{x}_m$ which are weighted with some arbitrary factor w_{ij} ($1 \leq j \leq m$, for neurone i); the processing element in which the weighted inputs are summed and translated through some function $a(f)$; and the output response y_i of the function.

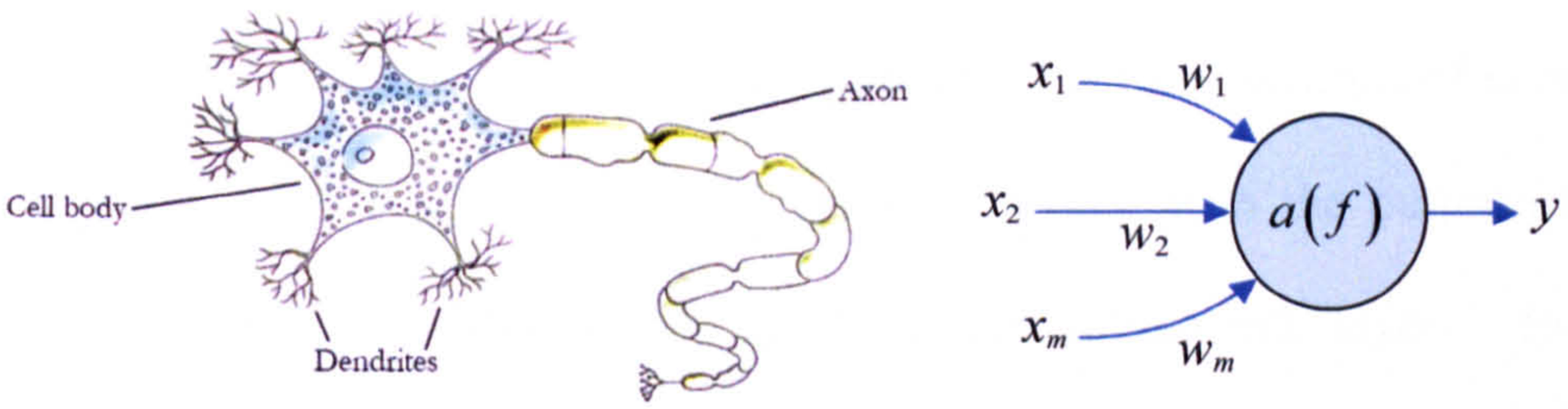


Figure 3.1. Representations of (a) the biological neurone; and (b) the artificial neurone.

An illustration is given in Figure 3.1. The function $a(f)$ determines the behaviour of the McCulloch and Pitts neurone. The sigmoid function is the most popular due to it being continuously differentiable – an important factor during the learning process (discussed in Section 3.2.1). The two sigmoid functions are the unipolar (3.2) and bipolar (3.3) functions:

$$a(f) = \frac{1}{1+e^{-\lambda f}}, \tag{3.2}$$

$$a(f) = \frac{2}{1+e^{-\lambda f}} - 1, \tag{3.3}$$

where λ defines the slope of the function. The output of the neurone is the transform of the weighted sum of inputs such that:

$$y_i = a\left(\sum_{j=1}^{m} w_{ij}x_j - \theta_i\right), \tag{3.4}$$

where θ_i is an arbitrary offset. As can be seen from (3.2), (3.3) and (3.4) the neurone has a number of adjustable parameters such as the weights w_{ij}, offset θ_i and function slope λ which affect how the neurone responds to input data.

When neurones are inter-connected with synapses the output of an individual neurone is passed to the inputs of its neighbours. These collections of neurones are typically structured such that they exist in layers, with the output of one neuronal layer connecting to the input of the subsequent layer. In this configuration, the collection is referred to as a Multi-Layer Perceptron (MLP). The following section introduces the MLP and discusses how it employs the neuron parameters of (3.2), (3.3) and (3.4) in its learning process.

3.2 The Multi-Layer Perceptron architecture

The MLP architecture is comprised of layers of neurones inter-connected in such a way that the outputs of one layer are propagated to the subsequent layer. Figure 3.2 shows the architecture of the MLP and illustrates the relationships between neurones, layers and inter-connections. The illustration is of a three layer MLP. In Figure 3.2 some data vector $\mathbf{x}_m$ is presented to the input layer. At this layer, the values of $\mathbf{x}_m$ are not weighted and are transformed through the neurones (depicted by circles in Figure 3.2). The values are propagated to the

hidden layer neurones through the inter-connecting synapses (denoted in Figure 3.2 by the unbroken and dashed lines). Here they are weighted and transformed by the neurones as given by (3.4). This process is replicated as the transformed outputs are propagated to the subsequent layer(s). It can be seen from Figure 3.2 that the output vector of the MLP is some transform of the input vector $\mathbf{x}_m$. It follows that the MLP's response to an input vector $\mathbf{x}_m$ is governed by the parameters of the individual neurones. Rumelhart *et al.*, [4] elegantly said of the MLP, "*In this case the information coming into the input units [neurones] is recoded into an internal representation and the outputs are generated by the internal representation rather than the original pattern*" [4].

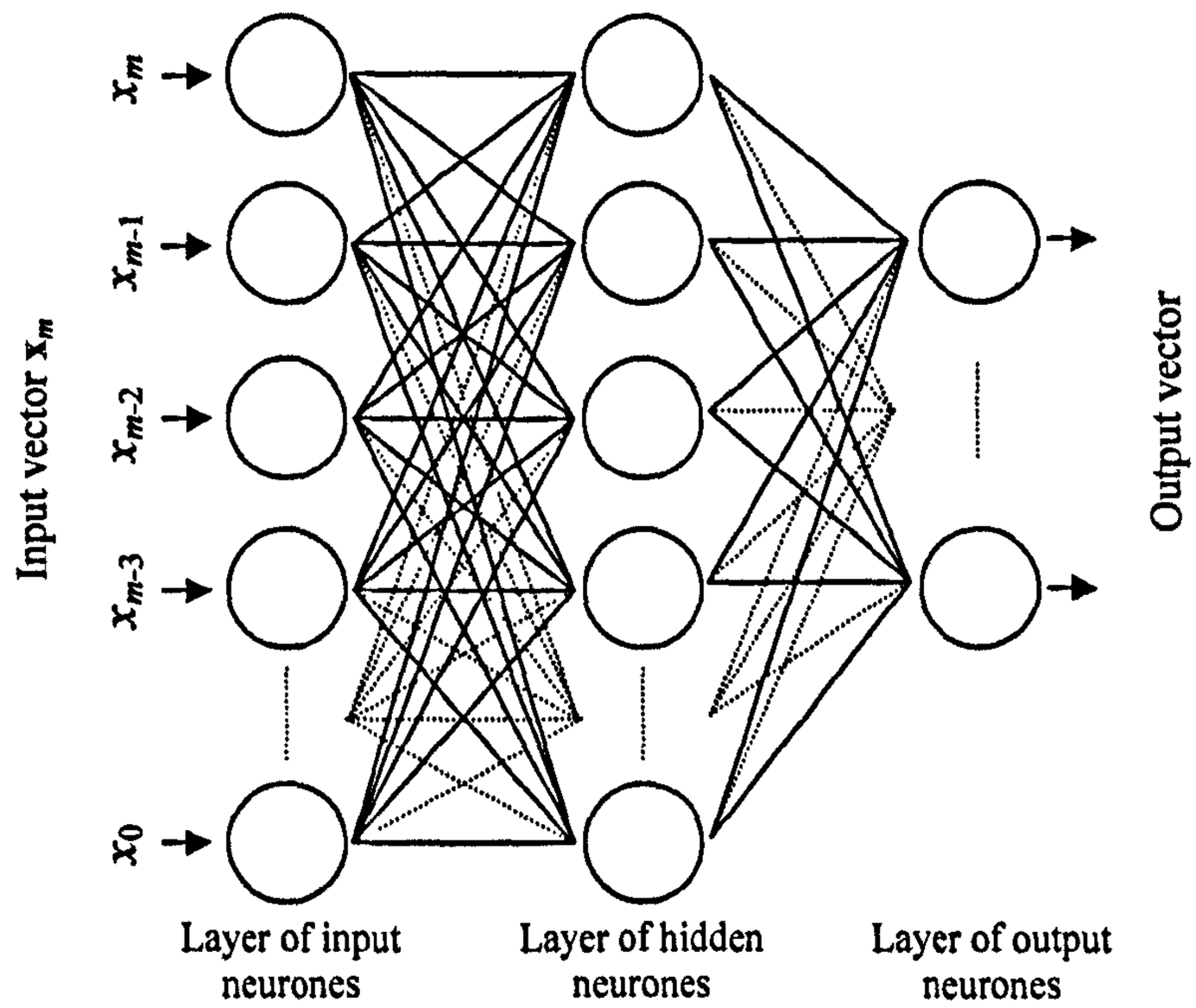


Figure 3.2. Architecture of the MLP.

In classification scenarios the MLP adjusts the neuronal parameters to discriminate between input patterns. Presented with an input pattern $\mathbf{x}$ linked to some class C_k (*i.e.*, $\mathbf{x} \in C_k$) the neurone parameters will be adjusted such that the

MLP output signifying C_k is amplified and that all other-class output patterns are attenuated. A process referred to as a *training algorithm* is responsible for this. We will discuss this in the following section.

3.2.1 Training the Multi-Layer Perceptron

Many different training algorithms exist that can be applied to the MLP, by far the most common being the error Back-Propagation (BP) algorithm [4,5]. The purpose of the error BP algorithm is to minimise the difference between the generated network output and the desired output, termed the error. As was shown by Rumelhart *et al.*, [4,5] this minimisation cannot be solved analytically but is an iterative process employing the derivatives of the error, neuronal activation functions and weights. The error between the j-dimensional generated network output o and the desired output t for pattern p is expressed in the form of the Sum-Squared Error (SSE) E_p as:

$$E_p = \frac{1}{2}\sum_j \left(t_{pj} - o_{pj}\right)^2. \tag{3.5}$$

Many implementations of the MLP with error BP algorithm use the Mean Squared Error (MSE) which is merely the SSE averaged over the number of output neurones j. The error BP algorithm follows a method known as *gradient descent* [4,5]. That is the algorithm follows the largest negative error gradient in minimising the error. This is shown in the derivation of the error BP algorithm formalised in Appendix I. Suffice to say that the change in synapse weight $\Delta_p w_{ji}$ between neurones j and i in response to pattern p is:

$$\Delta_p w_{ji} = \eta \delta_{pj} o_{pi}, \tag{3.6}$$

where δ_{pj} is the total neuronal error and o_{pi} is the network response to pattern p. η is the learning rate [4] and is used to modify the weight change. The MLP when trained with the error BP algorithm attempts to minimise the SSE E_p through iterative adjustments to the network weights (henceforth referred to as the *weight matrix*) but the success of this approach cannot be guaranteed. Considering that a q-dimensional error surface can be constructed representing the problem at hand (where q is the number of neuronal weights in the ANN), the objective becomes that of finding the global minimum on this surface. The global minimum represents the lowest error for a given weight matrix. It is possible for the error surface to present a number of local minima and as a consequence of the algorithm's gradient descent method, it becomes possible for the process to terminate prematurely upon finding one of these minima. Rumelhart *et al.*, [4] introduced the concept of momentum (denoted by α) to ensure that the change in weight matrix during an iteration is influenced by the change made during the previous iteration. They give the modified learning rule as being:

$$\Delta_p w_{ji}(n+1) = \eta\left(\delta_{pj} o_{pi}\right) + \alpha \Delta_p w_{ji}(n), \tag{3.7}$$

where n is the iteration number. This modified algorithm is popularly termed error Back Propagation with Gradient Descent and Momentum (BPGDM). By iteratively updating the weight matrix, the MLP begins to discriminate between different classes of patterns based on their indicative features. As is evident from

(3.7) and Appendix I, in order for the BPGDM algorithm to be successfully applied, a set of training input patterns accompanied by their associated target outputs need to be provided. Target outputs are required in order to calculate δ_{pj}. The process of providing target outputs alongside input patterns is referred to as *supervised learning*. For methods of unsupervised learning the reader is referred to [6] for a non-exhaustive description of some of the more popular algorithms. Supervised learning requires a dataset of input patterns with associated target patterns. This dataset is referred to as the *training dataset*. The MLP is trained to recognise indicative features from this dataset with the BPGDM algorithm. There are no formalised rules governing the size of the training dataset in terms of the patterns it contains. The number of patterns required for successful training (*i.e.*, the number of patterns that will allow the error to be reduced to its lowest possible value) largely depends upon the type of data used *e.g.*, the number of features present in the patterns (typically corresponding to the number of input neurones in the MLP). Methods for estimating the number of required training patterns for a given problem exist in the form of qualitative guidelines and these are discussed in Section 5.4.1.

3.2.2 Bayesian formulation of the Multi-Layer Perceptron

The MLP can be reformulated in the Bayesian framework. The network attempts to determine the posterior probabilities $P(C_k \mid \mathbf{x})$ for each class *i.e.*, the probabilities of some input pattern $\mathbf{x}$ belonging to class C_k. Estimation of the posterior probabilities occurs as the error BP algorithm minimises the error between network output and the desired target output during the learning phase. Let us consider an MLP in some dichotomous classification scenario with binary

targets. Given an MLP $F(\mathbf{x},\mathbf{W})$ of weight matrix $\mathbf{W}$, the response of the network to an m-dimensional pattern $\mathbf{x}$ in Euclidean space (*i.e.*, $\mathbf{x}\in\mathbb{R}^m$) can be expressed as:

$$E_{MSE}(\mathbf{W})=\sum_{\mathbf{x}\in C_1}\left(F(\mathbf{x},\mathbf{W})-1\right)^2+\sum_{\mathbf{x}\in C_2}F(\mathbf{x},\mathbf{W})^2, \tag{3.8}$$

where C_i is class i (i = 1,2) [7]. Assuming that there are an infinite number of learning samples in the training dataset, the MSE generalises to:

$$\begin{aligned}E_{MSE}(\mathbf{W})=&P(C_1)\int_{\mathbb{R}^m}\left(F(\mathbf{x},\mathbf{W})-1\right)^2P(\mathbf{x}|C_1)d\mathbf{x}\\&\times P(C_2)\int_{\mathbb{R}^m}F(\mathbf{x},\mathbf{W})^2P(\mathbf{x}|C_2)d\mathbf{x},\end{aligned} \tag{3.9}$$

where the P probability distributions have their usual meanings. Using (3.1) Ruck *et al.*, [8] obtained the following expression:

$$E_{MSE}(\mathbf{W})=e^2(\mathbf{W})+\int_{\mathbb{R}^m}P(C_1|\mathbf{x})\left(1-P(C_1|\mathbf{x})\right)P(\mathbf{x})d\mathbf{x}, \tag{3.10}$$

where:

$$e^2(\mathbf{W})=\int_{\mathbb{R}^m}\left[F(\mathbf{x},\mathbf{W})-P(C_1|\mathbf{x})\right]^2P(\mathbf{x})d\mathbf{x}. \tag{3.11}$$

As is evident from (3.10) and (3.11) the MSE is proportional to $e^2(\mathbf{W})$. As the MSE decreases during MLP learning the difference between the MSE and posterior probability densities decreases *i.e.*, the MLP tends towards the posterior

probability densities $P(C_k \mid \mathbf{x})$ [7]. Further treatment of this subject can be found in [2,9].

Chapter 5 demonstrates the abilities of the MLP as a classification mechanism by showing that it can differentiate between different cardio-respiratory disease processes based on the associated heart and lung sounds. Furthermore, it is demonstrated in Chapter 6 that the MLP can be applied to predict arterial pulsatile waveforms in forming classification systems.

The standard MLP is only one type of ANN. It is a fully inter-connected network insofar that the neurones in one layer are fully inter-connected with neurones in the subsequent layer. This level of connectivity affords the MLP a large set of parameters which can be adjusted during the training process and it follows that learning can be a computationally intensive task. Some other ANN architectures are computationally less intensive, such as the RBFN, which we discuss in the next section.

3.3 The Radial Basis Function Network

Similar to the MLP, the RBFN is an ANN consisting of artificial neurones. Recalling Figure 3.2, the RBFN has a similar architecture to the standard MLP, exhibiting fully interconnected layers. Whilst the output layer neurones follow the McCulloch and Pitts model, the hidden layer employs a different type of neurone – the Radial Basis (RB) neurone.

The RBFN, suggested by Moody *et al.*, [10] employs RB neurones which have activation functions based on dissimilarity measures. An RB neurone's activation function has an *m*-dimensional Gaussian kernel φ centred around some *prototype* vector μ. Considering an *m*-dimensional input pattern **x**, the response of the activation function is a measure of the similarity of the input pattern and prototype vector. The activation function $\varphi_n(\mathbf{x})$ is expressed as:

$$\varphi_n(\mathbf{x}) = \exp\left\{\frac{\|\mathbf{x}-\boldsymbol{\mu}_n\|^2}{2\sigma_n^2}\right\},\ \mathbf{x}\in\mathbb{R}^m,\ n=1,...,N, \tag{3.12}$$

where the variance of the n^{th} Gaussian function is σ_n. The Gaussian functions are typically normalised such that:

$$g_n(\mathbf{x}) = \frac{\varphi_n(\mathbf{x})}{\sum_{n=1}^{N}\varphi_n(\mathbf{x})}. \tag{3.13}$$

It is evident from (3.12) that the response of a Gaussian activation function is partially governed by the Euclidean distance between the input pattern and the prototype vector. The smoothness of the function is defined by σ_n.

The output layer of the RBFN consists of M-P neurones with linear activation functions. Linear activation functions perform linear transforms of the input pattern as opposed to the non-linear transforms performed by (3.2) and (3.3). The output neurones sum the weighted outputs of the hidden-layer RB neurones and transform the result through the activation function so the output o_i becomes:

$$o_i = a_i\left(\sum_{k=1}^{N} w_{ik}g_k(\mathbf{x}) - \theta_i\right),\ i=1,...,L, \tag{3.14}$$

where o_i is the i^{th} final-layer neurone output, L is the number of neurones in the output layer, N is the number of neurones in the hidden layer and θ_i is the i^{th} neurone offset (*i.e.*, bias).

3.3.1 Training the Radial Basis Function Network

Training an RBFN is a two-stage process. Similar to the MLP, the RBFN is trained in a supervised manner, being presented with the input patterns and associated target patterns simultaneously. During the first stage of the training process, the RBFN establishes RB neurones for each pattern in the training dataset. Each RB neurone prototype vector is set to a training pattern from the dataset. The second stage performs the weight w_{ik} adaptations required by the output layer. It is evident from (3.14) that normalisation of the Gaussian functions as given in (3.13) is not strictly required as the normalisation effect can be incorporated into the w_{ik} weights [2]. The weights in the output layer are updated using the expression:

$$\Delta w_{ik} = \eta (t_i - o_i) g_k (\mathbf{x}), \tag{3.15}$$

where t_i is the desired target value for the i^{th} output neurone. It is axiomatic from the above expressions that the RBFN can be trained with relatively lower computational overheads than would be required by the MLP with the error BP algorithm. Firstly, the establishment of the RB neurones φ_n is a relatively trivial exercise as the prototype vectors μ_n are the training dataset patterns. Secondly the output neurones' weights are updated without the lengthy iterations required by the error BP algorithm.

From the above account it follows that as the size of the training dataset increases so does the computational overheads required by the RBFN. A technique for estimating the number of RB neurone Gaussian functions (or [*radial*] *basis functions*) that is optimally required for the training dataset is the *K-means algorithm* [6,11,12]. This algorithm is discussed in the following section.

3.3.2 RBFN optimisation methods

Establishing one RB neurone per pattern is not an optimal method of constructing an RBFN. For the method to be optimal, it demands that each training pattern presents unique information about the problem – which is often not the case [2,13]. Optimisation becomes the process of selecting only the training patterns that are useful for solving the problem. Hartman *et al.*, [14] considered an artificial (*i.e.*, non real world) dataset containing patterns of 20 dimensions (*i.e.*, elements). 19 of the dimensions consisted of noise and contained no useful information whilst the remaining element contained information about the classification scenario. The RBFN when trained with the dataset exhibited poor performance in discriminating against the irrelevant features and found that the error could only be minimised by increasing the number of RB neurones in the hidden layer [14]. This non-optimal method motivates the need for a prior selection of the number of RB neurones and their prototype vectors.

Optimisation through selection falls into two distinct categories: *non-informative* and *informative*. Non-informative methods typically involve randomly selecting subsets of training patterns to be utilised as prototype vectors [2]. These methods

often result in sub-optimal RBFNs as the chosen patterns can be non-representative of the overall statistical distribution. Informative methods often work by sequentially adding RB neurones to the network and assessing the impact these additions have on the error. Let us now consider two key methods.

- One particular method which follows this strategy is the *orthogonal least squares* algorithm proposed by Chen *et al.*, [15]. The method involves sequentially adding those RB neurones to the RBFN which prompt the largest decrease in error. Without termination criteria, the algorithm will create as many RB neurones as there are training patterns thereby undoing any optimisation that may have occurred. The algorithm offers an account of how the error changes with the addition of each prototype vector and leaves it to the user to determine how many prototypes to add.

- Another method for analysing the training data to determine the optimal number of RB neurones is the K-means clustering algorithm (also referred to as *hard C-means clustering*).

The K-means clustering algorithm operates by partitioning the training dataset into c clusters C_i ($i = 1,...,c$). The clusters are chosen such that the dissimilarity (or distance) between each datum in the cluster and the cluster centre is minimised. This dissimilarity measure is sometimes referred to as a *cost function* or *objective function*. Typically a Euclidean distance measure is used such that the total cost J_i incurred for cluster i can be expressed as:

$$J_i = \sum_{k,\mathbf{x}_k \in C_i} \left\| \mathbf{x}_k - \mathbf{c}_i \right\|^2, \tag{3.16}$$

where $\mathbf{x}_k$ is the k^{th} pattern vector in the cluster and $\mathbf{c}_i$ is the i^{th} cluster centre vector. The expression in (3.16) can be rewritten to give the total cumulative cost J for all clusters:

$$J = \sum_{i=1}^{c} \sum_{k,\mathbf{x}_k \in C_i} \left\| \mathbf{x}_k - \mathbf{c}_i \right\|^2. \tag{3.17}$$

The partitioned training dataset (comprising of n vectors) is represented by the binary matrix $\mathbf{U} = \{u_{ij}\}$, $i = 1,\ldots,c$ and $j = 1,\ldots,n$. The element u_{ij} is 1 if the j^{th} vector belongs to cluster i, and is 0 otherwise. Once all cluster centres are established, **U** can be populated as follows:

$$u_{ij} = \begin{cases} 1 & \text{if } \left\| \mathbf{x}_j - \mathbf{c}_i \right\|^2 \leq \left\| \mathbf{x}_j - \mathbf{c}_k \right\|^2, \forall k \neq i \\ 0 & \text{otherwise} \end{cases}. \tag{3.18}$$

Therefore $\mathbf{x}_j$ is assigned to cluster i if $\mathbf{c}_i$ is the closest cluster centre to $\mathbf{x}_j$. Given a populated partition matrix **U** the cluster centre $\mathbf{c}_i$ that will minimise (3.17) can be computed from the mean vector of all pattern vectors in cluster i such that:

$$\mathbf{c}_i = \frac{1}{|C_i|} \sum_{k,\mathbf{x}_k \in C_i} \mathbf{x}_k, \tag{3.19}$$

where $|C_i|$ is the number of patterns assigned to cluster C_i, *i.e.*:

$$|C_i| = \sum_{j=1}^{n} u_{ij}. \tag{3.20}$$

As can be seen from (3.16) and (3.20) optimisation of the membership matrix **U** is an iterative problem requiring the successive recalculation of cluster centres and their impact upon the total cumulative cost. The algorithm terminates when the change in cluster centres between two successive iterations falls below some predefined threshold. The algorithm – formalised in [11,16] – does not provide a means to ascertain the optimal number of RB neurones. The user selects the number of clusters, executes the algorithm, and then examines the suitability of the clusters based on the incurred cumulative error. The process of determining the number of clusters is best performed via an inspection of the available training data. From inspection of the dataset it could be argued that there should be more clusters than there are classes in some scenarios. Such a requirement is plausible and indeed advisable with heterogeneous datasets. The final cluster centres obtained using the algorithm can be directly employed in the RBFN architecture by setting one hidden RB neurone prototype vector μ_n to a cluster centre vector $\mathbf{c}_n$. Using this technique it proves possible to reduce the size of the RBFN architecture without compromising its classification ability.

These methods of RBFN optimisation provide a means to reduce the complexity of the architecture. Random selection of training data subsets can frequently result in sub-optimal solutions whilst the orthogonal least squares method provides a more informative account of how the network error is related to the training patterns. The orthogonal least squares method is limited insofar that it exclusively uses the training patterns and cannot deviate from these prescribed vectors. The K-means algorithm however is free to adjust its cluster centres

away from the initial training vectors if this provides a general decrease in error. The K-means algorithm is therefore an attractive heuristic in RBFN optimisation.

RBFNs are attractive architectures when attempting to solve certain types of non-linear problems. Classification and prediction tasks are frequently solvable using these ANNs but they can suffer from problems. In particular the work of Hartman *et al.*, [14] demonstrated that the RBFN cannot easily discriminate between relevant and irrelevant dataset features. The authors found that the MLP can outperform the RBFN in this respect. Chapter 5 of this thesis shows an example of the abilities of the RBFN in classifying cardio-respiratory disease processes by analysis of the associated heart and lung sounds. The RBFN is compared to the MLP in order to assess its generalisation capabilities. Due to the nature of the hidden layer RB neurones the RBFN is particularly adept at probability density estimation. Each RB neurone can represent the probability of occurrence of its prototype vector. Given that the final layer aggregates the responses of the RB neurones into the final output, the RBFN can be considered to be a form of Gaussian Mixture Model (GMM) [2]. An ANN which follows the GMM and is firmly rooted in probability density estimation is the PNN. This architecture is discussed further in the next section.

3.4 The Probabilistic Neural Network

The PNN – introduced by Specht in 1990 [17] and later refined in 1992 [18] – is architecturally similar to the MLP and RBFN discussed in Sections 3.2 and 3.3 respectively. Recalling Figure 3.2, the PNN is similar to the 3-layer MLP in that it is comprised of an input layer, hidden layer and output layer. The PNN

possesses a fully inter-connected input and hidden layer, ensuring that all pattern vectors presented to the network are available to all hidden neurones. The hidden neurones utilise an exponential activation function in their neuronal model and there exist as many neurones as there are patterns presented to the network – similar to an un-optimised RBFN. The second synaptic layer of inter-connections exists to aggregate the responses of the hidden neurones into class neurones and is only partially connected (Figure 3.2). The existence of these partial inter-connects indicate that the true class assignments of the training data are known during the training phase *i.e.*, that the PNN employs a supervised training mechanism. The final layer represents the probability of class assignment for each class [19]. As discussed in Section 3.1.1 the function of any classifier is to estimate the statistical distribution of the training data and use this estimation to define posterior probability densities $P(x|C_k)$ for classification of unlabelled data. One of the most straightforward approaches to this statistical density estimation is to represent the probability density $P(x)$ in a functional form. The PNN represents these densities as multivariate normal distributions which can be expressed as:

$$P(\mathbf{x}) = \frac{1}{(2\pi)^{m/2}|\boldsymbol{\Sigma}|^{1/2}} \exp\left\{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^{T}\boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})\right\}, \tag{3.21}$$

where m is the dimensionality of $\mathbf{x}$ and $\boldsymbol{\mu}$, $\boldsymbol{\Sigma}$ is an $m \times m$ covariance matrix, and $|\boldsymbol{\Sigma}|$ is the determinant of $\boldsymbol{\Sigma}$. The factor on the left hand side of the exponent in (3.21) ensures that the integral of $P(\mathbf{x})$ is unity. Furthermore the quantity $(\mathbf{x}-\boldsymbol{\mu})^{T}\boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})$ is the distance measure between the pattern vector $\mathbf{x}$ and the

mean vector μ, and is termed the Mahalnobis distance. Given (3.21) it can be seen that the expressed probability contours of the distribution are hyperellipsoidal with the principal axes being given as the eigenvectors $\mathbf{u}_i$ $(i = 1,...,m)$ of Σ such that:

$$\Sigma \mathbf{u}_j = \lambda_i \mathbf{u}_j \tag{3.22}$$

where λ_i is the variance along the i^{th} axis. As is evident from the description of the hidden neurone's activation functions, the PNN behaves in a manner not dissimilar to a GMM [2]. Considered in the GMM framework the individual hidden neurones can be argued to be independent probability distributions of the training data. The PNN models the class densities φ_i of the input data by aggregating the probability densities weighted by the prior probabilities of occurrence:

$$\varphi_i(\mathbf{x}) = \sum_{j=1}^{M} P(j) P_j(\mathbf{x}), \tag{3.23}$$

where M is the number of prototype vectors (*i.e.*, number of patterns in the training dataset) belonging to class *i*, $P(j)$ is the prior probability of occurrence of the response from the j^{th} component and $P_j(\mathbf{x})$ is the probability density given in (3.21). The PNN architecture can be extended to incorporate a fourth layer that serves to attribute *decision risks* to classes. If an erroneous classification results in $\mathbf{x} \in C_k$ instead of $\mathbf{x} \in C_i$ then a penalty v_i^k can be imposed. The risk function $\rho_k(\mathbf{x})$ of class *k* can then be expressed as:

$$\rho_k(\mathbf{x}) = \sum_{i=1}^{c} v_i^k P(C_i)\varphi_i(\mathbf{x}), \tag{3.24}$$

where c is the number of classes and $P(C_i)$ is the prior probability of class i. Used in a risk-based framework, the PNN would simply classify input patterns based on minimum risk, selecting the class that incurs the lowest cost *i.e.*:

$$C_k = \arg\min_{1 \le k \le c} \{\rho_k(\mathbf{x})\}. \tag{3.25}$$

The fourth layer can be omitted if there are no cost functions utilised in the classification scheme, reverting the architecture to a three-layer network that directly employs GMM. The PNN suffers from the same drawback as the RBFN insofar that a hidden-layer neurone is created for each pattern in the training dataset. Numerous training algorithms have been proposed that can optimise the neuronal parameters of the PNN. These parameters include the covariance matrix Σ and mean vector μ in (3.21) which respectively affect the multidimensional smoothness and centre point of the activation function. Specht [17] proposed the method of generating one hidden neurone per training pattern and further proposed restricting Σ to a scalar quantity σ in (3.21). Substitution of Σ for σ results in homoscedastic activation functions under which even small changes can affect the density function along all principal axes. Berthold *et al.*, [19] commented that minimal adjustments of σ can impact heavily on network performance. It is axiomatic that large datasets will yield large PNN architectures which can detrimentally increase computational complexity. Another proposal by Specht [18] involved replacing σ with a diagonal matrix Σ and iteratively adjusting the diagonal entries in Σ depending upon the network

error. This method of adjustment mimics the error BP algorithm in Section 3.2.1 which is computationally intensive and does not address the issues surrounding large datasets. Streit *et al.*, [20] proposed fixing the number of hidden neurones in each class followed by an iterative adjustment of the parameters in (3.21) using a maximum likelihood training method [1,2]. This approach becomes attractive when training a PNN with large datasets as it minimises the size of the PNN and hence computational complexity. These methods of training or optimising the PNN either require homoscedastic activation functions (which limits the network's adaptability) or fixed-size architectures. A more attractive method is one in which the PNN architecture is constructed by sequentially adding neurones in a similar manner to the orthogonal least squares algorithm discussed in Section 3.3.2. Such networks are referred to as Constructive Probabilistic Neural Networks (CPNNs). CPNNs ameliorate the problems encountered by PNNs by dynamically 'growing' the architecture whilst retaining heteroscedastic activation functions.

3.4.1 The Constructive Probabilistic Neural Network

The CPNN is essentially a PNN that is grown by the sequential addition of neurones in the hidden and output layers. The neurones are added in response to patterns presented in the training dataset. Prior to adding a neurone, an assessment is firstly made as to whether existing neurones can perform the same function. If they can, then they are adjusted to encompass the new training pattern, otherwise a new neurone is added. This method of constructing the CPNN often results in highly compact and efficient architectures requiring low

computational overheads. The algorithm responsible for constructing the PNN is now discussed.

The Dynamic Decay Adjustment (DDA) algorithm [19] is an extension to the Restricted Coulomb Energy (RCE) [21] and Probabilistic Restricted Coulomb Energy (P-RCE) [22] algorithms. The DDA algorithm, introduced by Berthold *et al.*, [19] is a constructive algorithm in that it iteratively adds neurones to the network and adjusts the parameters of the neuronal activation functions in (3.21). Parameters are adjusted by the algorithm to prevent conflict between neurones *i.e.*, when two or more neurones linked to different classes respond to an input pattern hence resulting in an ambiguous classification. Adjustment to the parameters include modification of the amplitude of the function and the covariance matrix Σ. Berthold *et al.*, [19] define two terms:

- θ^+ the *minimum correct-class probability.* Given an input pattern **x** linked to class k presented to the network, at least one neurone linked to the same class must respond with an activation greater than θ^+.
- θ^- the *maximum incorrect-class probability.* Given an input pattern **x** linked to class k presented to the network, any neurones linked to any other class are permitted to respond provided that their activations are no greater than θ^-.

Presented now is an annotated formal description of the algorithm as given by Berthold *et al.*, [19] followed by a qualitative description. For clarity $\varphi_i^k(\mathbf{x})$ is the activation function for neurone i linked to class k as given in (3.21).

Step 1. Present a training pattern – target class pair $(\mathbf{x}, k)$ to the network.

Step 2. If $\exists \varphi_i^k(\mathbf{x}) : \varphi_i^k(\mathbf{x}) \geq \theta^+$ then increase the amplitude A_i^k of function $\varphi_i^k(\mathbf{x})$ by 1 and proceed to step 4.

Step 3. Introduce a new neurone with activation function $\varphi_{l+1}^k(\mathbf{x})$, amplitude $A_{l+1}^k = 1$ and mean vector $\boldsymbol{\mu}_{l+1}^k = \mathbf{x}$. Calculate the standard deviation using:

$$\sigma_{l+1}^k = \min_{l \neq k, 1 \leq j \leq q_l} \left\{ \sqrt{-\frac{\left\| \boldsymbol{\mu}_j^l - \boldsymbol{\mu}_{l+1}^k \right\|^2}{\ln \theta^+}} \right\}, \tag{3.26}$$

where $l \neq k, 1 \leq j \leq q_l$ is equivalent to all neurones connected to all classes other than class k (the correct class), and q_l is the number of neurones in class l.

Step 4. Adjust the standard deviation of all conflicting neurones using:

$$\forall l \neq k, 1 \leq j \leq q_l : \sigma_j^l = \min \left\{ \sigma_j^l, \sqrt{-\frac{\left\| \mathbf{x} - \boldsymbol{\mu}_j^l \right\|^2}{\ln \theta^-}} \right\}, \tag{3.27}$$

and then proceed to step 1.

In qualitative terms, when a pattern-class pair $(\mathbf{x}, k)$ is presented to the network during training, if any neurone i linked to class k responds with an activation $\geq \theta^+$ then the neurone's activation function amplitude is increased by 1. All neurones linked to classes $l \neq k$ have their standard deviations σ reduced using (3.27). The reduction is such that their activation responses to pattern $\mathbf{x}$ will be

less than θ^-. If no neurone of class k responds to the pattern $\mathbf{x}$ then a new neurone is added to the architecture, linked to class k and centred around $\mathbf{x}$ *i.e.*, $\boldsymbol{\mu}_{j+1}^k = \mathbf{x}$. The new neurone is added to the architecture to account for future occurrences of the pattern.

Once all training patterns have been presented to the network and the algorithm has created the activation functions, it can be noted that the activation functions may have differing amplitudes; as a consequence of step 2 of the algorithm. These amplitudes relate to the prevalence of the training patterns that these neurones respond to and hence can be seen to represent the prior probabilities $P(j)$. The prior probabilities can be computed by normalising the activation function's amplitudes against their same-class counterparts such that:

$$P^k(j) = \frac{A_j^k}{\sum_{i=1}^{q_k} A_i^k}, \tag{3.28}$$

where hidden neurone j is linked to class k and q_k is the number of hidden neurones linked to class k. As can be seen from the aforementioned steps the purpose of the DDA algorithm is to iteratively determine an appropriate mixture of Gaussian density functions to represent the training dataset drawn from m-dimensional input space $\mathbb{R}^m$. Berthold *et al.*, [19] found that this algorithm can offer substantial advantages in terms of the final PNN network size and generalisation capabilities. Indeed it can be argued that the CPNN affords the user a less computationally complex architecture and can remain a network capable of generalising well with unseen data because the algorithm creates only

the number of elements (*i.e.*, neurones) required to represent the underlying statistical distribution.

As can be seen from the above PNN and CPNN descriptions, these ANN architectures can only be used in classification scenarios. Function or time series prediction is beyond the capabilities of these architectures whereas it has been shown in Chapter 2 that the MLP and RBFN have both found numerous applications in this field. These mainstream ANN architectures can all be used as classifiers, being capable of effectively resolving some linear and non-linear problems. Whilst the PNN and CPNN are inherently computationally lightweight ANNs, the RBFN requires an optimisation heuristic to impart the same level of flexibility. In comparison, whilst the MLP demands higher computational overheads than the PNN and CPNN, the ability of the BPGDM training algorithm allows the MLP to reject aspects of the training data that are not pertinent to the classification problem. From this it can be seen that selection of the appropriate ANN architecture for solving a problem can heavily impact on the solution. An inappropriate ANN selection may yield an ineffective solution, or a solution prone to erroneous classifications.

Given that the main purpose of statistical pattern recognition is to formulate *decision rules* whereby data can be assigned to classes, application of a singular ANN may be insufficient in solving the problem [23]. One practical problem is the scarcity of training data which can introduce uncertainty into the estimation of posterior probability densities. Binaghi *et al.*, [24] noted that "*Classification problems may present different kinds of uncertainty that render classification*

statements vague and doubtful, membership in classes a matter of grades, and probability theory requirements either too restrictive or all together inadequate" [24]. A plurality of different ANN architectures (or *topologies*) may be required to render the problem solvable – or at least improve on single-classifier approaches. Uncertainty, the requirement of multiple ANN topologies, and near-homogeneous data all conspire to produce a scenario which Bayesian theory cannot effectively solve [23]. A robust method of combining the results of multiple classifiers whilst providing a framework for evaluating uncertainty is Dempster-Shafer Theory (DST) which falls under the umbrella of Evidence Combination Theory (ECT). These concepts are discussed in the next section.

3.5 The Dempster-Shafer Theory of evidence combination

The Dempster-Shafer Theory of evidence offers a framework for reasoning between imprecise, ambiguous and conflicting hypotheses (*i.e.*, classifications or probabilistic assertions). This section presents an overview of DST whilst a complete account can be found in [25]. DST extends traditional Bayesian classification to incorporate *belief* and *plausibility* terms. In order to define these concepts and examine how they can assist in combined classification, we consider the following. Let us consider each possible classification $\mathbf{x} \in C_k$ as being one of a number of mutually exclusive propositions A_i where $i = 1,...,k$. These propositions form the *frame of discernment* $\Theta = \{A_1,...,A_k\}$. The classification from an ANN in an ensemble can be considered to be a basic belief assignment m. The value $m(A)$ can be considered to be the mass of belief that is committed to $\mathbf{x} \in A_k$ (and to none of its subsets) [23,25]. Consequently for all

subsets (*i.e.*, classifications) $B \subseteq A$ the belief measure $Bel(A)$ can be expressed as:

$$Bel(A) = \sum_{B \subseteq A} m(B). \tag{3.29}$$

The quantity $Bel(A)$ is the extent to which one believes in A either directly attributable to $m(A)$ or indirectly attributable to $m(B)$ for all $B \subseteq A$ [23]. In extension to this, plausibility is defined as the body of evidence committed to $\overline{A}$ or its subsets *i.e.*, the extent of doubt of $\mathbf{x} \in A_k$. Plausibility is therefore expressed as:

$$Pl(A) = \sum_{A \cap B \neq \varnothing} m(B) = 1 - Bel(\overline{A}). \tag{3.30}$$

Belief and plausibility measures afford the user with a framework for assessing the strength of classifications from ANNs. An important aspect of DST is its ability to combine classifications (*i.e.*, evidence) provided by multiple sources (*e.g.*, ANNs in an ensemble). Given two basic belief assignments m_1 and m_2 from two independent sources, Dempster's rule of evidence combination follows the form:

$$m_E(A) = \frac{\sum_{B \cap C = A} m_1(B) m_2(C)}{1 - \sum_{B \cap C = \varnothing} m_1(B) m_2(C)}, \ \forall A \neq \varnothing, \tag{3.31}$$

where $m_1(B) > 0$ and $m_2(C) > 0$ for some non-disjoint subsets B and C of Θ and $m_E(A)$ is the orthogonal sum of m_1 and m_2. As can be seen from (3.31), Dempster's rule of evidence combination permits the aggregation of independent ANN classifications at the output-neurone level. It facilitates the creation of a combined ANN classifier that accounts for conflicting and ambiguous classifications. The following sections summarises the discussions presented in this chapter.

3.6 Summary

This chapter has introduced some of the most prevalent ANN architectures in the field of biomedical engineering. It has introduced the field of Bayesian Probability Theory and demonstrated how these ANNs perform classifications based on these probabilities. It has considered the relative merits of each type of ANN and has highlighted their shortcomings, emphasising the necessity for appropriate selection of architectures based on the problem at hand. This chapter has further examined situations where single ANN approaches cannot effectively solve classification problems, and has introduced an effective means by which to aggregate ANNs by way of DST. The following chapters discuss how ANNs were applied to the development of *biosignal classification* systems as part of this work.

References

[1] Congdon, P., *Bayesian statistical modelling*, John Wiley & Sons Ltd., Chichester, 2001.

[2] Bishop, C.M., *Neural networks for pattern recognition*, Oxford University Press, Oxford, 1995.

[3] McCulloch, W.S. and Pitts, W., A logical calculus of ideas immanent in nervous activity, *Bulletin of Mathematical Biophysics*, **5**, 1943, pp.115-133.

[4] Rumelhart, D.E., Hinton, G.E. and Williams, R.J., Learning internal representations by error propagation, *Parallel distributed computing: explorations in the microstructure of cognition*, (Eds. Rumelhart, D.E. and McClelland, J.L.), MIT Press, Cambridge, MA, 1986, pp.318-362.

[5] Rumelhart, D.E., Hinton, G.E. and Williams, R.J., Learning representations by back-propagating errors, *Nature*, **323**, 1986, pp.533-536.

[6] Lin, C.-T. and Lee, C.S.G., *Neural fuzzy systems: a neuro-fuzzy synergism to intelligent systems*, Prentice Hall PTR, Upper Saddle River, NJ, 1996.

[7] Funahashi, K., Multilayer neural networks and Bayes decision theory, *Neural Networks*, **11**(2), 1998, pp.209-213.

[8] Ruck, D.W., Rogers, S., Kabrisky, M., Oxley, H. and Suter, B., The multilayer perceptron as an approximation to a Bayes optimal discriminant function, *IEEE Transactions on Neural Networks*, **1**(4), 1990, pp.296-298.

[9] Likas, A., Probability density estimation using artificial neural networks, *Computer Physics Communications*, **135**(2), 2001, pp.167-175.

[10] Moody, J. and Darken, C., Fast learning in networks of locally-tuned processing units, *Neural Computing*, **1**(2), 1989, pp.281-294.

[11] Bezdek, J.C., *Pattern recognition with fuzzy objective function algorithms*, Plenum Press, New York, 1981.

[12] De Gruijter, J.J. and McBratney, A.B., A fuzzy K-means method for predictive classification, *Classification and related methods of data analysis: proceedings of the first conference of the International Federation of Classification Societies*, (Ed. Bock, H.H.), North-Holland, Amsterdam, 1988, pp.97-104.

[13] Duda, R.O., Hart, P.E. and Stork, D.G., *Pattern classification*, 2nd Ed., John Wiley & Sons Inc., New York, 2000.

[14] Hartman, E.J, Keeler, J.J. and Kowalski, J.M., Layered neural networks with Gaussian hidden units as universal approximators, *Neural Computation*, **2**(2), 1990, pp.210-215.

[15] Chen, S, Cowan, C.F.N. and Grant, P.M., Orthogonal least squares learning algorithm for radial basis function networks, *IEEE Transactions on Neural Networks*, **2**(2), 1991, pp.302-309.

[16] Jang, J.-S.R., Sun, C.-T. and Muzutani, E., *Neuro-fuzzy and soft computing: a computational approach to learning and machine intelligence*, Prentice Hall PTR, Upper Saddle River, NJ, 1997.

[17] Specht, D.F., Probabilistic neural networks, *Neural Networks*, **3**(1), 1990, pp.109-118.

[18] Specht, D.F., Enhancements to probabilistic neural networks, *Proceedings of the IEEE International Joint Conference on Neural Networks*, IEEE, New York, 1992.

[19] Berthold, M.J. and Diamond, J., Constructive training of probabilistic neural networks, *Neurocomputing*, **19**(1-3), 1998, pp.167-183.

[20] Streit, R.L. and Luginbuhl, T.E., Maximum likelihood training of probabilistic neural networks, *IEEE Transactions on Neural Networks*, **5**(5), 1994, pp.764-783.

[21] Hudak, M.J., RCE classifiers: theory and practice, *Cybernetics and Systems*, **23**, 1992, pp.483-515.

[22] Reilly, D.L., Cooper, L.N. and Elbaum, C., A neural model for category learning, *Biological Cybernetics*, **45**(1), 1982, pp.35-41.

[23] Denœux, T., Analysis of evidence-theoretic decision rules for pattern classification, *Pattern Recognition*, **30**(7), 1997, pp.1095-1107.

[24] Binaghi, E., Gallo, I. and Madella, P., A neural model for fuzzy Dempster-Shafer classifiers, *International Journal of Approximate Reasoning*, **25**(2), 2000, pp.89-121.

[25] Shafer, G., *A mathematical theory of evidence*, Princeton University Press, Princeton, 1976.

CHAPTER 4

Experimental Method and Data Collection

This chapter discusses the manner in which the data for this research was acquired. It describes the hardware used and the rationale behind the experimental methods. The chapter concludes by discussing the construction of the final datasets that are used for the subsequent analyses.

4.1 Rationale behind data collection

In order to develop intelligent systems which can clinically assess patients, the systems must firstly have been 'trained' with appropriate medical data. As was discussed in Chapter 3, the BPGDM training algorithm, RBFN, and CPNN require datasets from which to learn. Motivation for our data collection is derived from this fact. In order to model clinical processes using these systems, 'patient data' must be at hand. The type of data considered as part of this research, and motivation towards its collection, is presented in Section 4.1.2.

The following section discusses how the data collected as part of this research is used to generate information, and how this information is used elicit knowledge.

4.1.1 From data-mining to knowledge extraction

Section 4.1 gave a brief account of the motivation for data collection. In Chapter 1 we introduced one of our objectives which was to attempt to model the anaesthetist's knowledge in order to provide an automated means of decision support. One of the most important questions therefore, is how one progresses from data (*e.g.*, raw, unprocessed numbers relating to biomedical parameters) to knowledge, *i.e.*, an understanding of the underlying relationships between biological processes and how these can be affected by medical intervention. The process one typically follows is illustrated in Figure 4.1, in which information is elicited from data, and in turn, knowledge is extracted from the information. As one follows the process it is usually the case that the 'data' becomes more and more useful in the context within which it is being utilised.

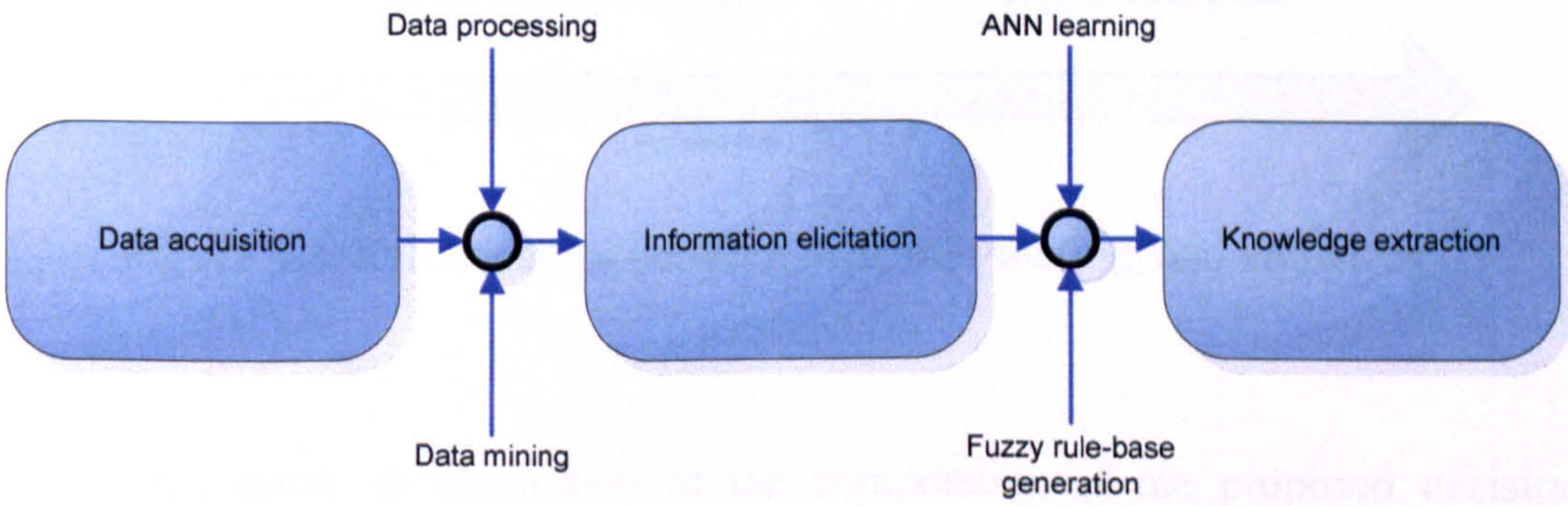


Figure 4.1. Process flow, from data to knowledge.

In its raw, unprocessed form, data imparts little information concerning the process or event that it describes. Typically it is correlated, noisy, incomplete, *etc.*, so a method of refining the data must be applied to elicit useful information.

Throughout the remainder of this thesis – and in particular Chapters 5, 6 and 7 – we will observe that the first process to be performed on the acquired dataset is to process the data to elicit information. In the context of cardio-respiratory auscultation, information could mean that: a certain range of coefficient values is indicative of a certain pathology; or in the context of arterial pulse waveforms, that an aperiodic (and hence) abnormal waveform has a low cross-correlation with a periodic version. In the context of this work we define information to be some quantity, refined – or *mined* – from raw data. Figure 4.2 illustrates the relationships between data and information in the context of this work.

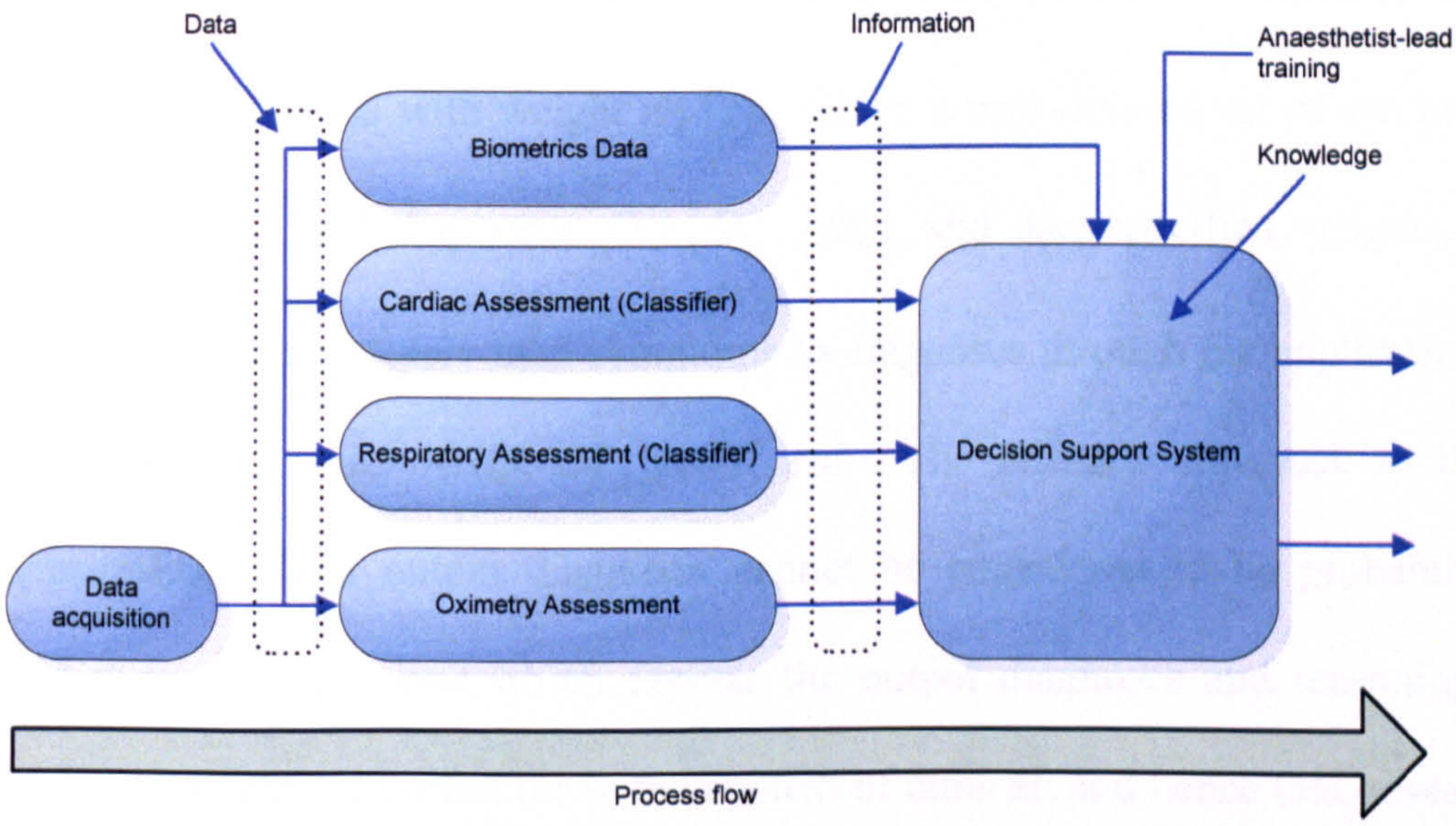


Figure 4.2. Relationships between data, information, and knowledge.

Figure 4.2 gives an illustration of the components of the proposed decision support tool, discussed in Section 4.1.2. Importantly it shows how information can be elicited from data, and how this information can be used to form knowledge. In this work we define knowledge to be the application of information in fulfilling a mandated task, *e.g.*, the classification of patients for surgical anaesthesia; and to be an understanding of the relationships between the

'tokens' of information. The decision support system extracts knowledge from the information provided by the previous components in conjunction with information provided by the anaesthetist during 'learning'. In essence a trained system will encapsulate the knowledge of the anaesthetist to this particular problem. By extracting knowledge and encapsulating it within the decision support system, the objective of an expert medical assessment system can be realised. One of the first rule-based expert medical assessment systems is MYCIN [1] which provides a framework for constructing tree-based decision systems. MYCIN-style systems are rule-based knowledge systems with weighted rules of the form $E \Rightarrow H(w)$ (*i.e.*, if the premise E holds, then the consequent H holds with weight w) [2]. Given a user-defined set of symptoms $S = \{S_1,...,S_n\}$ and diagnoses $D = \{D_1,...,D_m\}$ and user-specified weights, the objective is to accurately map symptoms to diagnoses through the application of combinatorial logic: AND, OR, NOT, *etc.* The primary drawback to these approaches is that output diagnoses cannot be considered to be probabilistic assertions [3]. Systems which express the output diagnoses and reasoning as belief measures are preferable in the context of clinical, and hence this, research. For examples of some expert rule-base systems and their application in clinical diagnosis the reader is referred to [4,5].

Throughout subsequent chapters we will refer back to this section in detailing how the presented research and development adheres to this objective. Discussed now are the *information sources* considered as part of this research.

4.1.2 Data types and information sources

This section briefly outlines the different types of data collected as part of this research. Motivation for its inclusion is presented, alongside the information that each data type provides. The data types are defined as:

- **Cardio-respiratory auscultation.** Since the advent of Laënnec's stethoscope in 1816 [6] the process of auscultating (*i.e.*, listening to) the chest has been of central importance in the assessment of patients. Auscultation provides the anaesthetist with useful data from which they derive information as to the presence of cardio-respiratory disease processes. Anaesthetists consider that it is important to perform this assessment. This is because knowledge about the presence or absence of cardio-respiratory disease processes can ultimately help them prescribe the appropriate type and duration of surgical anaesthesia; or this may prevent the patient from undergoing surgery altogether [7].

- **Blood pressure sphygmomanometry.** The sphygmomanometer (or blood pressure (BP) monitor) provides data about the state of the patient's cardiovascular system. In particular cardiac output, arterial distension, and the volume and viscosity of the blood. The sphygmomanometer measures systolic BP and diastolic BP in millimetres of mercury (mmHg), and also heart rate (HR) in beats per minute (bpm). These values inform the anaesthetist as to whether the patient is suffering from *e.g.*, hypertension, hypotension, cardiac fibrillation, *etc.* This assessment affords the anaesthetist a view as to how the cardiovascular system will respond to surgical

anaesthesia and is considered to be an important factor as to whether the patient can undergo surgery or not [8].

- **Pulse plethysmography.** Pulse oximetry data provides a means by which to quantify the extent of arterial haemoglobin oxygen saturation (SpO_2) *i.e.*, the amount of oxygen being transported by the blood. This is typically measured by a pulse oximeter. Data pertaining to the patient's level of oxyhaemoglobin may provide important clues about cardio-respiratory competence and of blood chemistry. The anaesthetist must assess this metric in order to safely prescribe the type and duration of surgical anaesthesia [8].

- **Lung spirometry.** Respiratory competence can be assessed using data relating to the amount of air exhaled by the patient during respiration. Respiratory function is typically measured as Peak Expiratory Flow Rate (PEFR), measured in L/min, and Forced Expiratory Volume in 1 second (FEV_1), measured in L. Respiratory competence may be diminished by existing disease processes, which may also result in complications during surgery. In Section 1.2.2 it was shown that the anaesthetist is responsible for the patient's respiration during surgery [8,9]. It is therefore important to the patient, and to the anaesthetist, that the respiratory system can convey sufficient oxygen during respiration and that the process is not impeded by an abnormality, *e.g.*, asthma, *etc.*

- **Clinical history.** Data pertaining to the clinical history of a patient can be shown to possibly be an indispensable source of information, regarding the

patient's state of health [10]. For example, anaesthetists rely upon the clinical history to provide indications about the patient's current state of disease processes and pathologies, *e.g.*, asthma, blood pressure, diabetes (affecting blood chemistry), *etc.* Used in conjunction with cardio-respiratory auscultation, blood pressure, pulse oximetry and lung function measurements, the clinical history can provide an informative account of the state of health of a patient. It can assist the anaesthetist in prescribing the most appropriate types of anaesthesia and any appropriate monitoring required during surgery. An exhaustive list of the clinical history indicators used throughout this research is presented in Section 4.4.

The types of data described above are essential to the accurate appraisal of a patient undergoing surgical anaesthesia [8]. As will be demonstrated through subsequent chapters, these metrics can be 'processed' using intelligent systems to produce accurate patient pre-operative assessments.

4.1.3 Dataset sources

Data was acquired from 194 patients between June 2003 and August 2004. The patients were scheduled for outpatient appointments at the Ear, Nose & Throat (ENT) clinics at Good Hope Hospital (Sutton Coldfield, U.K.), Birmingham Heartlands Hospital (Birmingham, U.K.) and Solihull Hospital (Solihull, U.K.). These hospitals are operated by NHS Trusts. These patients are henceforth referred to as the sample group.

In Section 4.2, an account of the cardio-respiratory data acquisition process is presented. This is followed in Section 4.3 by a description of how BP, HR, PEFR, FEV_1 and SpO_2 measurements were made. Section 4.4 presents the indicators that were taken from the patient's anonymous clinical notes. Section 4.5 gives a brief introduction to the ways in which this data is used in subsequent chapters.

4.2 Cardio-respiratory data acquisition

Cardio-respiratory sounds were collected from patients at the aforementioned hospitals. The author developed appropriate data collection protocols and worked with qualified physicians in order to collect 'sound data'. The purpose of these experiments was to acquire auscultated cardiac and respiratory sounds indicative of different disease processes and pathologies.

4.2.1 Experimental procedure for cardiac auscultation

Cardiac auscultation was performed on four areas of the precordium: the aortic area, pulmonic area, ventricular area and tricuspid area; in accordance with existing clinical practice. Certain cardiac pathologies manifest themselves at different locations in the heart. For example, aortic stenosis will manifest itself in the aortic region whilst ventricular impairment will be best heard when auscultating the ventricular region. It is this disparity between the audible location of disease processes and pathologies that motives the necessity for auscultating the various regions of the heart.

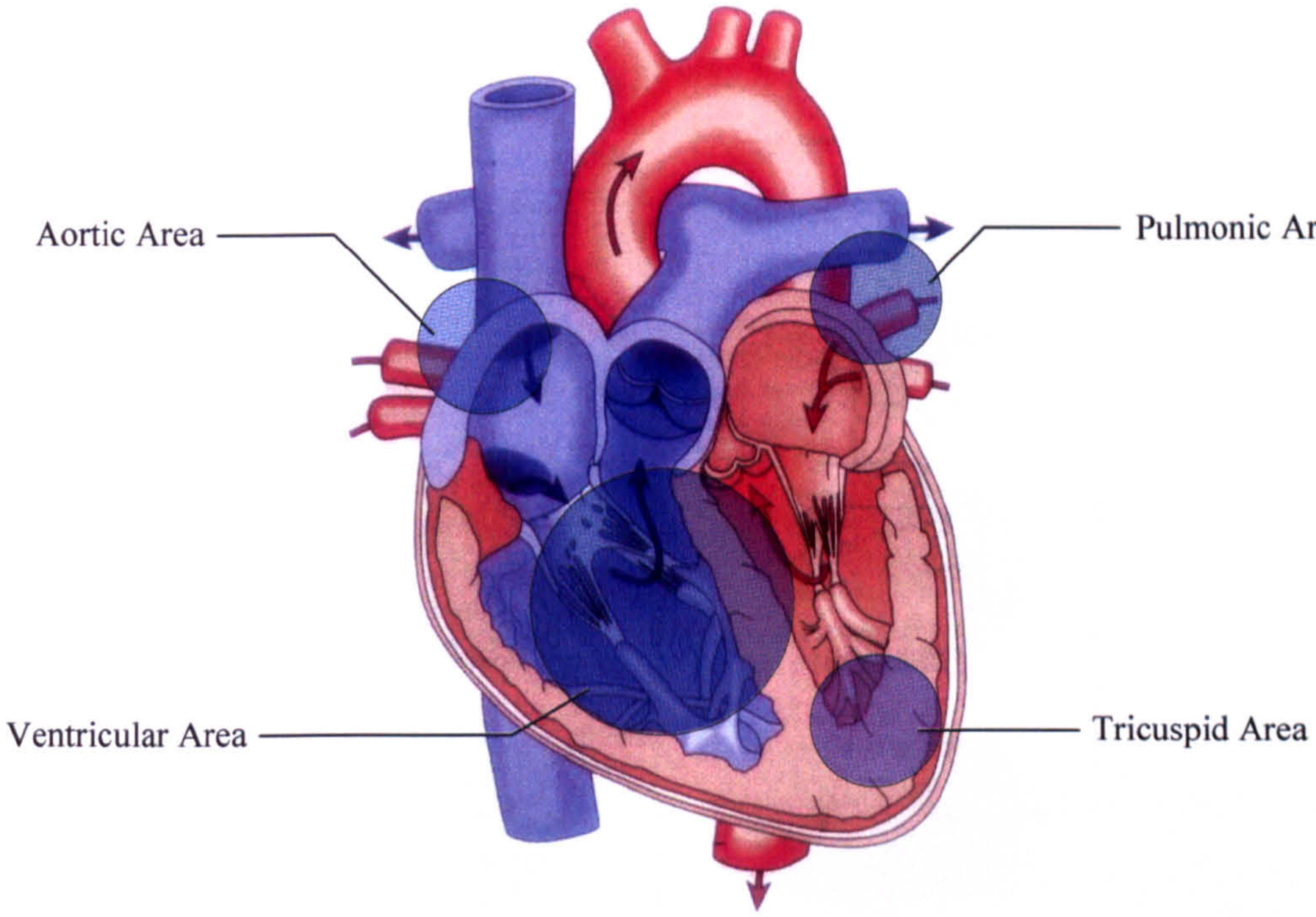


Figure 4.3. Cross section of the heart with the regions of auscultation [10].

Figure 4.3 shows the areas of auscultation in relation to the heart. Each patient in the sample group was assessed by a qualified physician and the presence of any cardiovascular disease process or pathology was recorded.

4.2.2 Data collection protocol for cardiac auscultation

Each heart sound and associated diagnosis was noted against their unique patient ID (PID) number and digitally recorded using the methods described in the following section. The PID was used as an anonymous identifier such that in the event of uncertainty an echocardiogram could be performed to ascertain the disease process. The echocardiogram is the standard and accepted tool for non-invasive cardiac valvular assessment [11]. When used in conjunction with the auscultated diagnostic assessment, the echocardiogram can present a sufficiently

informative record from which to make an accurate diagnosis. A further factor that was noted was whether the patient was fitted with prosthetic heart valves.

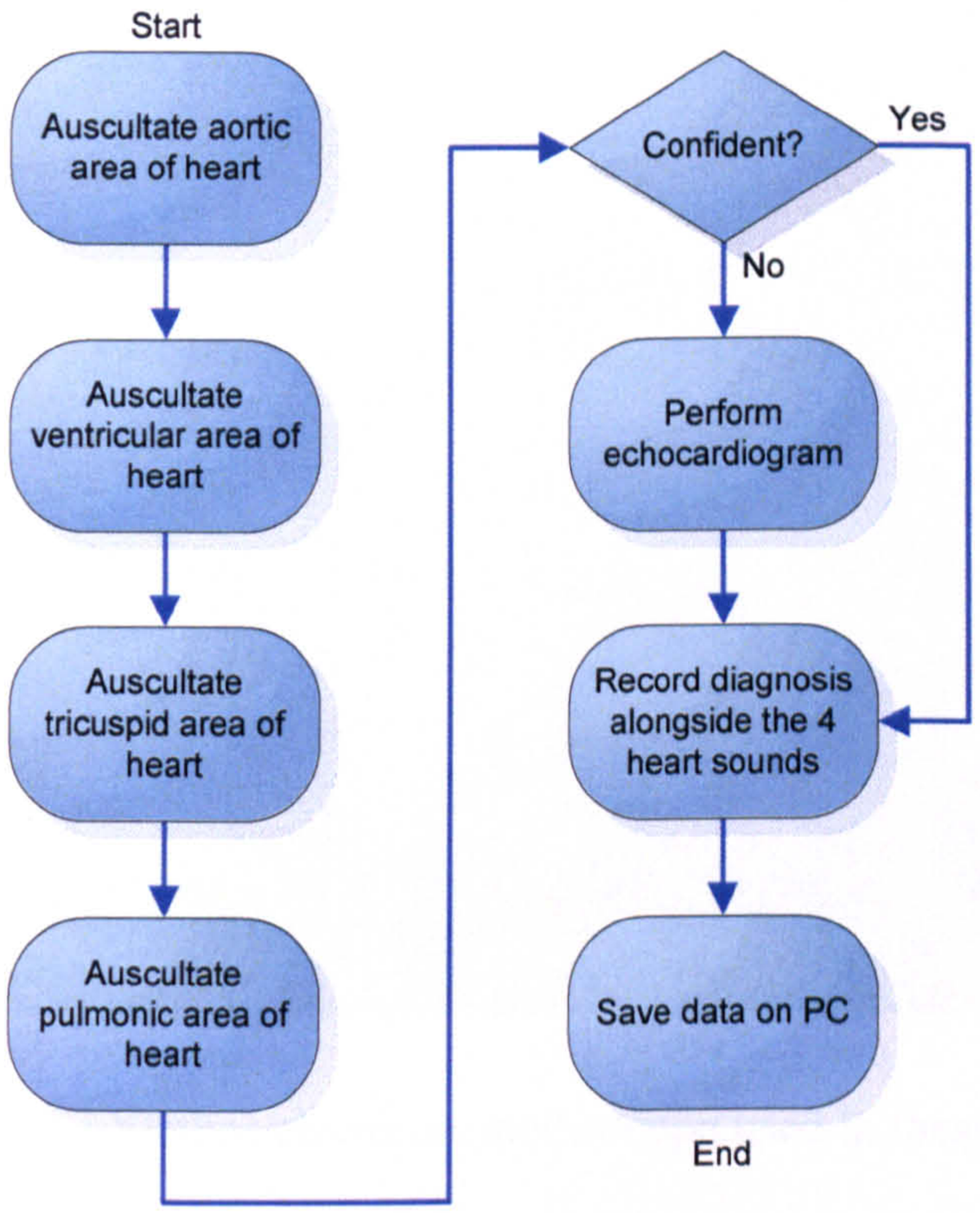


Figure 4.4. Flow diagram of the experimental procedure defined for cardiovascular auscultation.

Prosthetic valves can present audible clicks during the cardiac cycle which could affect subsequent audio signal processing. The experimental procedure is summarised in Figure 4.4. The data collection sheet is presented in Appendix II.

4.2.3 Equipment configuration for cardiac auscultation

The electronic stethoscope used throughout these experiments was the i-Stethos (Andromed Inc., Quebec, Canada), given in Figure 4.5. The electronic stethoscope is ergonomically similar to its acoustic counterpart and possesses an electret microphone and basic signal 'conditioning' electronics. For example, the

stethoscope employs a band-pass filter that attenuates extraneous noise from the signal during auscultation.

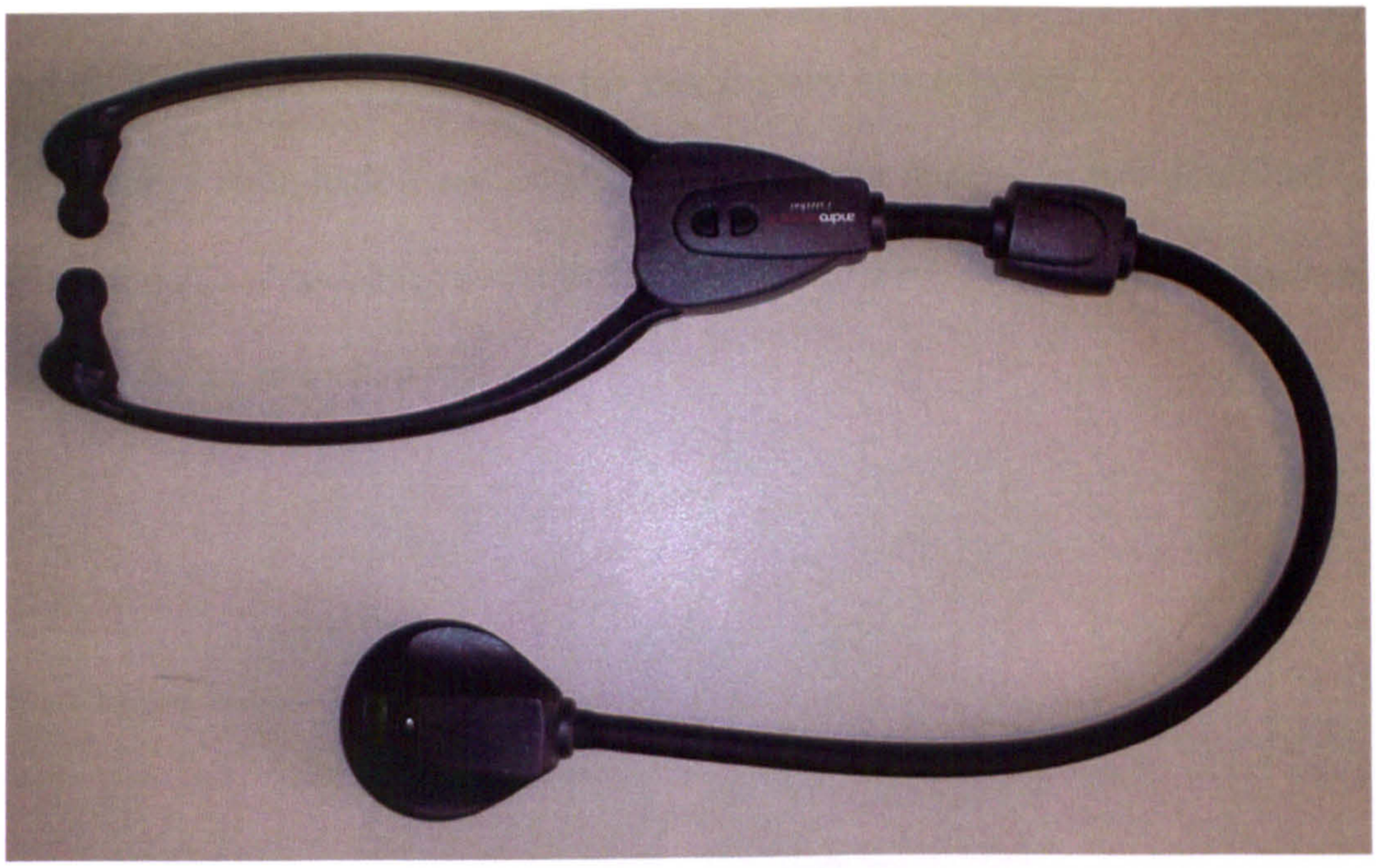


Figure 4.5. The i-Stethos electronic stethoscope used in these experiments.

When configured to auscultate the heart, the stethoscope exhibits a pass-band frequency range of 20Hz – 200Hz. This is in agreement with the audible frequencies of normal heart sounds and cardiac murmurs. By filtering out extraneous non-cardiac sounds the electronic stethoscope affords the physician a less obstructed account of cardiac competence[1]. The electronic stethoscope was connected to a Personal Computer (PC) via a 3.5mm jack plug and audio cable. The electronic stethoscope signal was sampled by the PC at a frequency of 44kHz using 16-bit data resolution. The duration for each auscultated sound was approximately 30 seconds. The cardiac sounds recorded by the electronic stethoscope were subsequently stored by the PC in an uncompressed form for

[1] The physicians who used the electronic stethoscope commented favourably on the clarity with which cardiac auscultated sounds could be heard.

subsequent analysis. A list of the cardiac abnormalities considered in this research is presented in Section 5.2.

4.2.4 Experimental procedure for respiratory auscultation

Respiratory auscultation was conducted in a similar manner to that described in Section 4.2.2. Auscultation was performed on two areas of the chest: the tracheal region and the bronchial region.

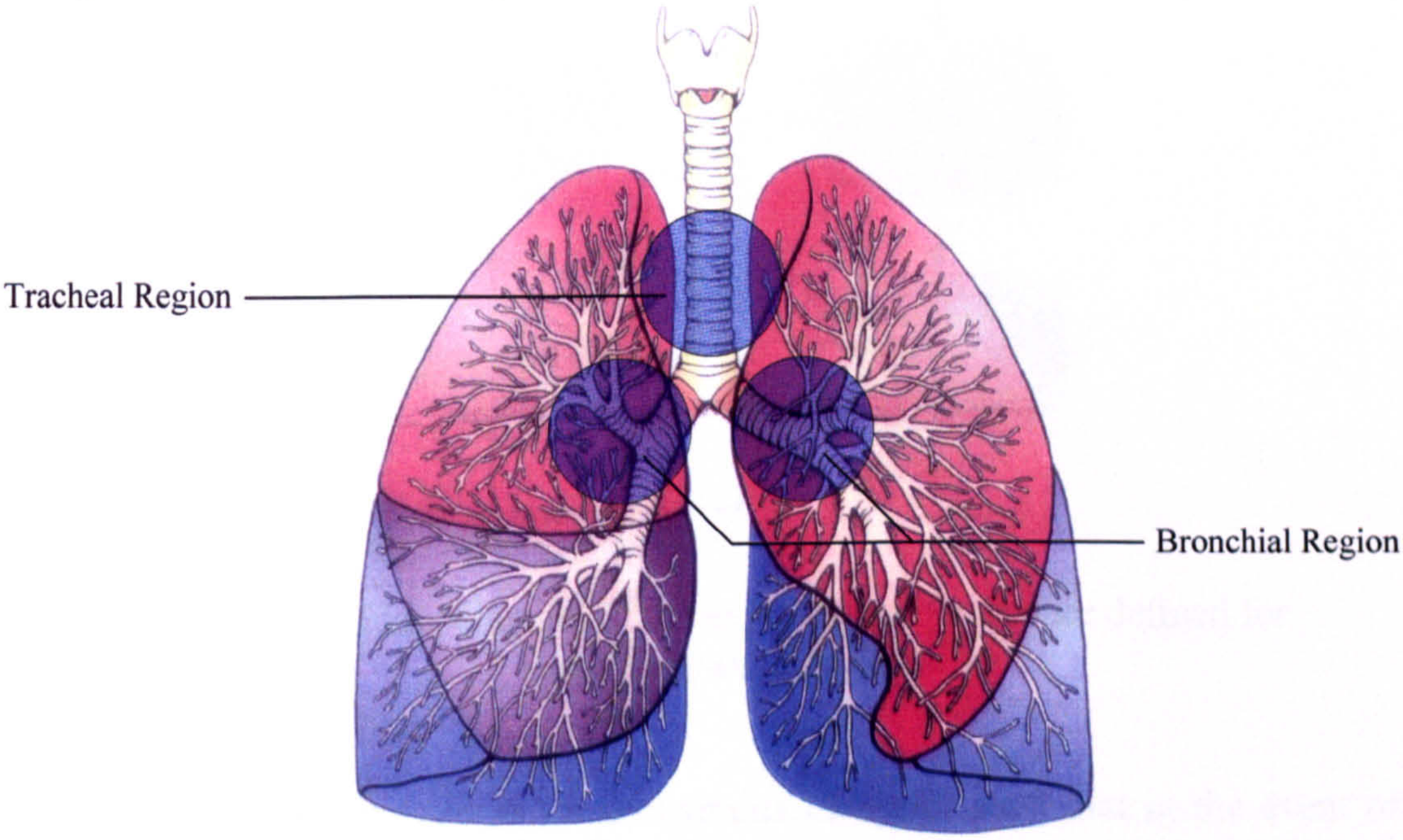


Figure 4.6. Cross section of the lungs showing the regions of auscultation [10].

Most of the breath sounds extrinsic to regular lung function manifest themselves in these two regions, particularly in the bronchial region [10]. Figure 4.6 shows a diagram of the lungs overlaid with the regions of auscultation. Respiratory auscultation was performed by a qualified physician with the i-Stethos electronic stethoscope (Figure 4.5). The filtering mechanism was configured for lung assessment, in which the frequency response band is 100Hz – 500Hz. Patients were auscultated in both bronchial and tracheal regions for approximately 30

seconds during which time a representative account of respiratory competence could be established. The physician identified whether the patient was presenting crackles, wheezes, *etc.*, which were possibly indicative of a disease process. These indicators are presented in Section 5.2.

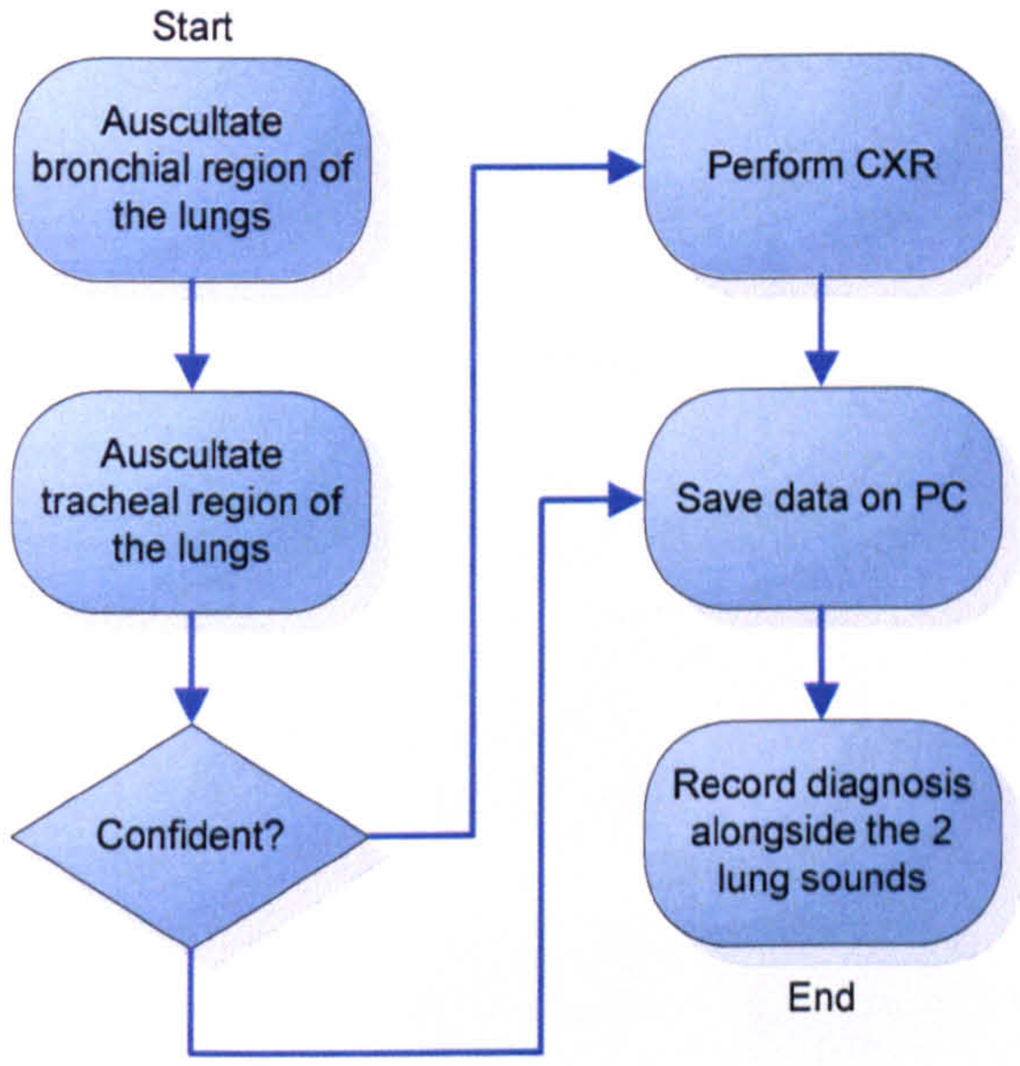


Figure 4.7. Flow diagram of the experimental procedure defined for respiratory auscultation.

The PID was again used as an anonymous identifier such that in the event of uncertainty a chest x-ray (CXR) could be performed to ascertain the abnormality. The experimental procedure for respiratory auscultation is summarised in the flow diagram presented in Figure 4.7. Section 5.2 gives the respiratory abnormalities considered as part of this research.

4.3 Biometric data acquisition

Biometric data comprising BP, HR, PEFR, FEV_1 and SpO_2 was collected from the sample group during their visit to the ENT outpatient clinics. Data was collected by the author under the partial supervision of a clinical nurse.

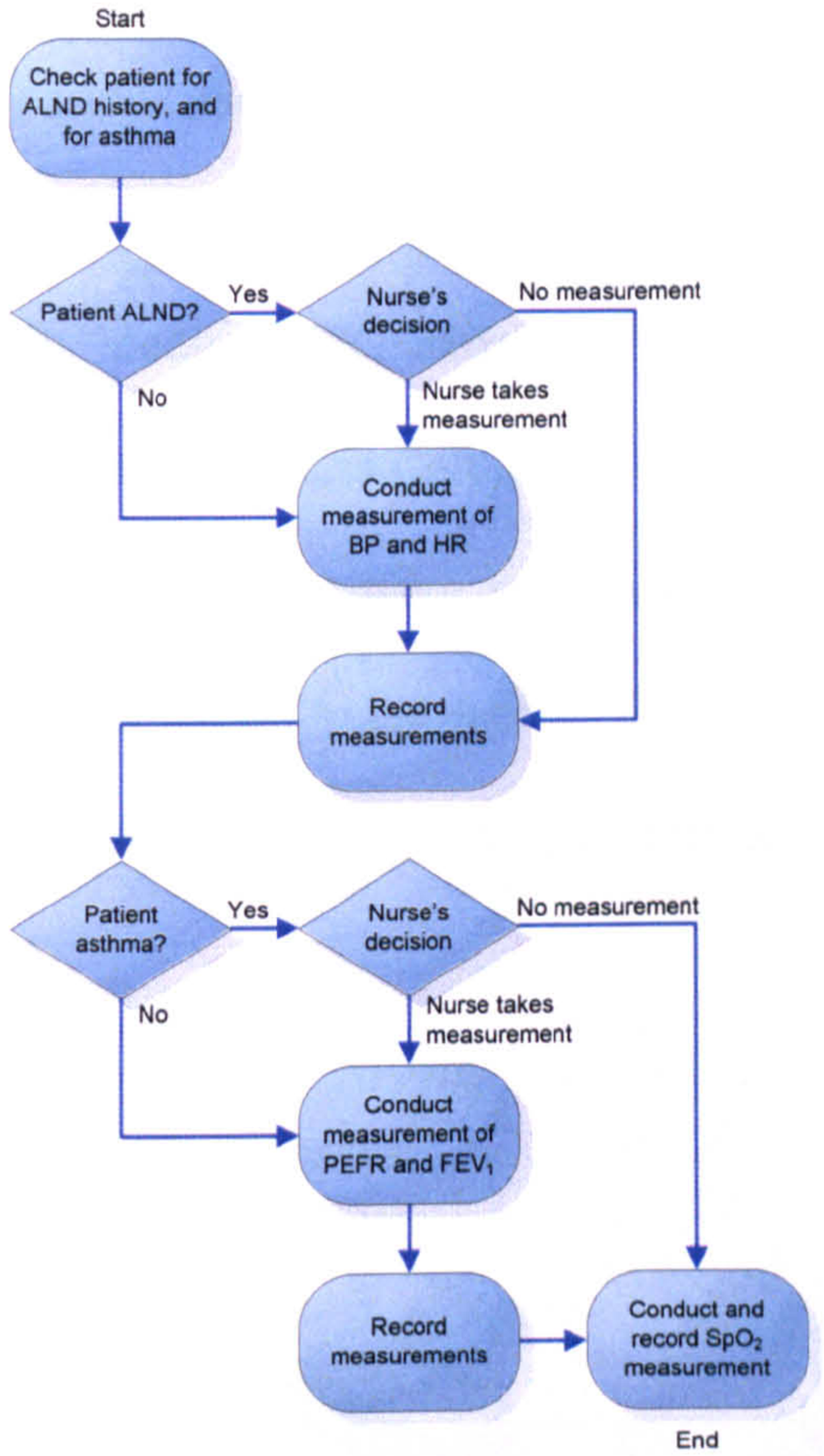


Figure 4.8. Biometric acquisition process accounting for patient risk.

Under certain circumstances it was required that BP, HR, PEFR and FEV_1 measurements be acquired by a qualified nurse. For example, in the instance of the patient having had axillary lymph node dissection (ALND), *i.e.*, removal of the axillary lymph nodes for cancer staging (detection), it is inadvisable to measure blood pressure on the side of the resection. Furthermore, if the patient is asthmatic then PEFR and FEV_1 measurements can aggravate the patient's condition. As such, the supervising nurse was required to decide whether it was appropriate to measure BP, HR, PEFR and FEV_1. The data acquisition process is formalised in Figure 4.8.

4.3.1 Hardware connectivity and data storage

The three items of hardware used during these experiments were the digital BP monitor (Boso Medicus: Bosch & Sohn, Germany), pulse oximeter (Nonin OEM2 monitor: Nonin Medical Inc., Minneapolis, USA), and lung function spirometer (KoKo Peak Flow: Ferraris CardioRespiratory, Colorado, USA). The digital BP monitor and spirometer were previously used by the supervising physicians and were recommended to the author. The Nonin OEM2 monitor was chosen as it was designed for rapid prototyping as well as clinical accuracy.

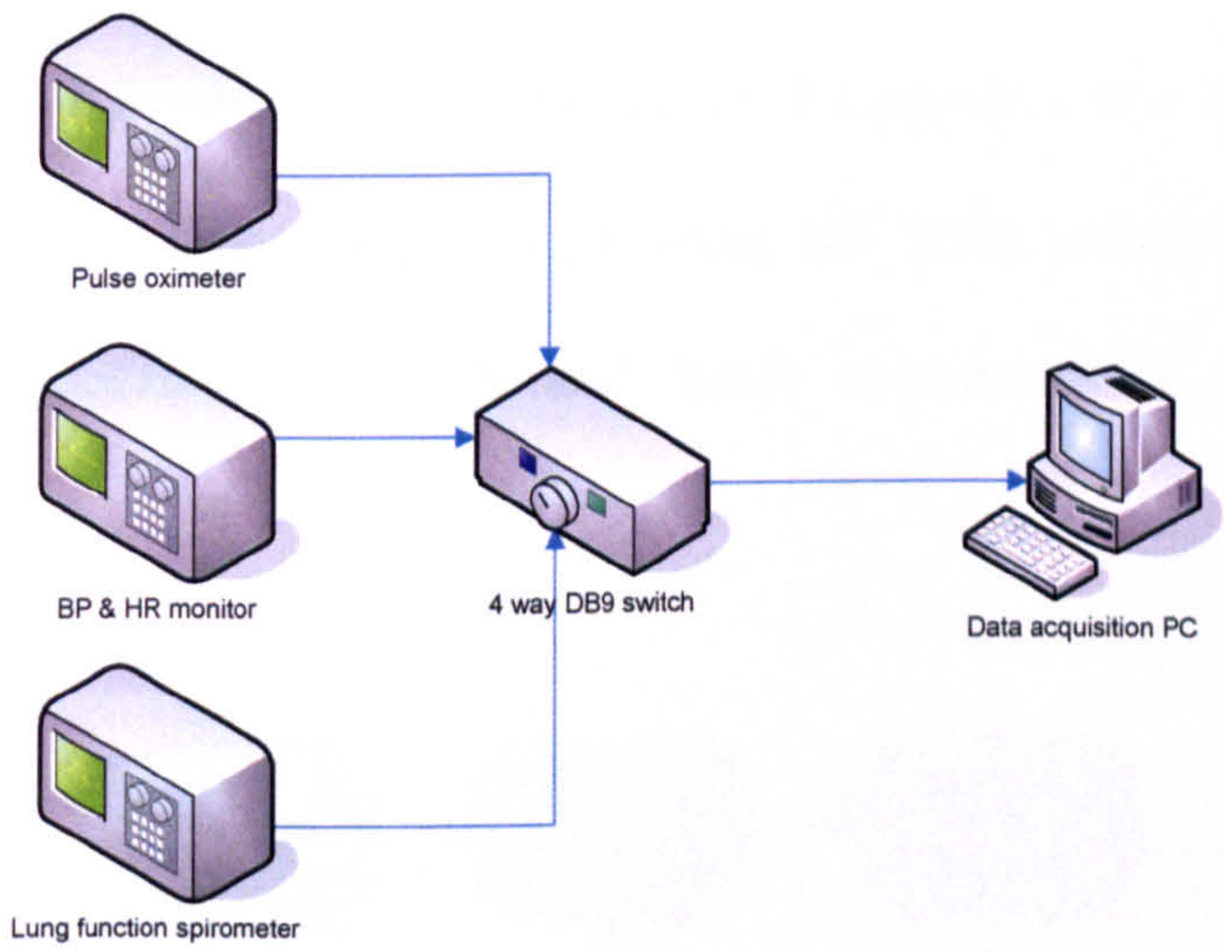


Figure 4.9. Diagram of the experimental setup, including the pulse oximeter, BP monitor and lung function spirometer.

These devices were connected to a PC via an RS232 serial port which enabled the devices to transmit their recorded values directly to the computer. Custom software to pre-process and store the data is discussed in Section 4.3.2, whilst data communication protocols are discussed in Section 4.3.3. A diagram showing the hardware setup is given in Figure 4.9.

Technically it was permissible to run the three tests in any order. The tests were not all run concurrently for the reason that they could interfere with one another. For example, the lung function test is a physically demanding exercise for the patient, requiring them to perform a forced respiratory expiration. The pulse oximeter is sensitive to physical movement and hence could collect a corrupted pulse waveform, *i.e.*, a waveform with *motion artefacts*. Motion artefacts are discussed at length in Section 6.2.2. However, BP, HR and SpO_2 data could be collected simultaneously if – and only if – measurements were taken from opposing sides of the patient, *i.e.*, left and right side. This is because, when in use, the BP monitor cuff exerts a pressure on the patient's arm and can restrict arterial blood flow, ergo potentially affecting the SpO_2 measurement. In the interests of expediency BP, HR and SpO_2 measurements were acquired simultaneously.

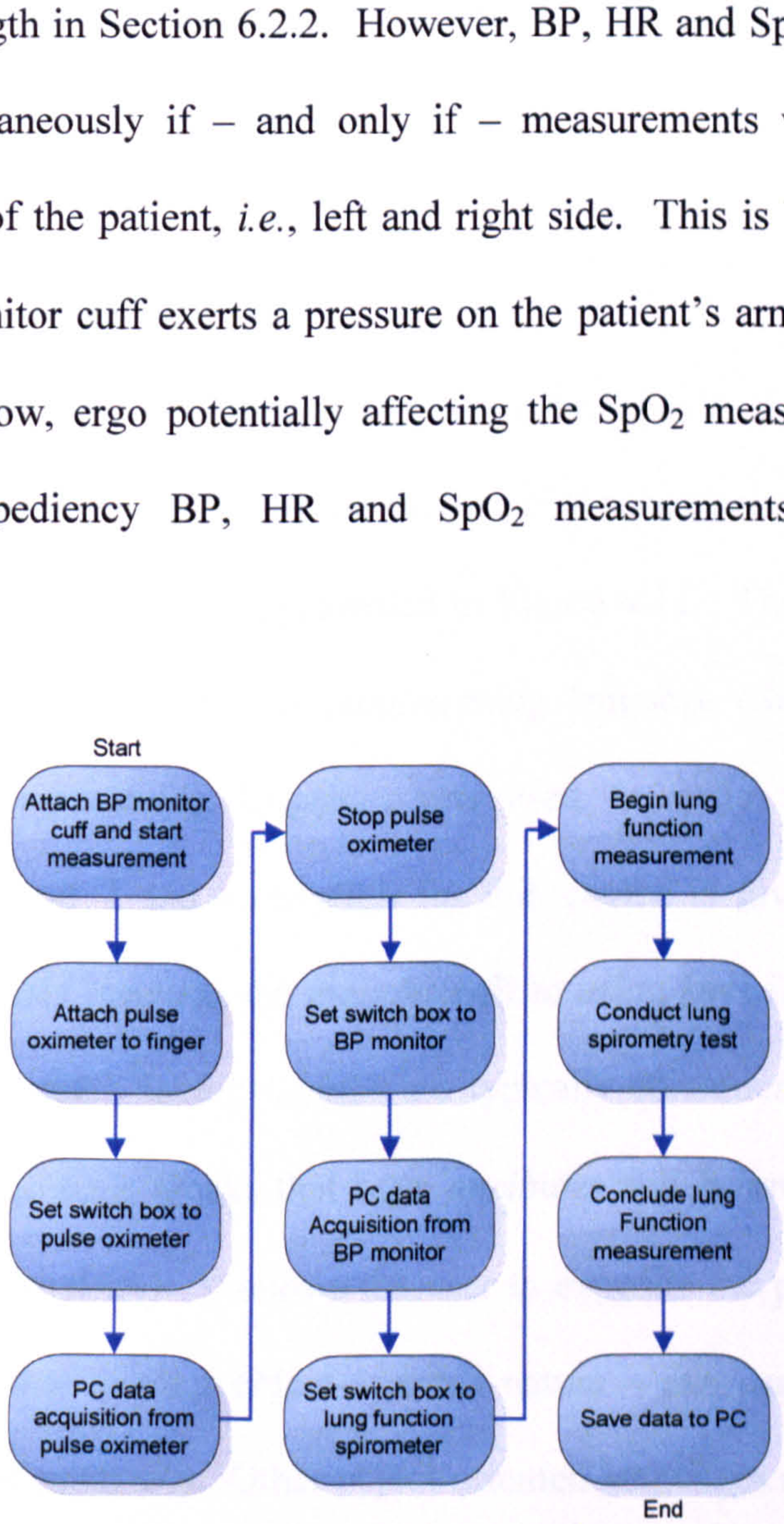


Figure 4.10. Flow diagram of the adopted experimental process.

The BP monitor was attached to the patient's (usually) right arm and the cuff was allowed to inflate in preparation for the BP measurement. During this time the oximeter sensor was attached to the index finger on the left hand and the instantaneous SpO_2 and pulse waveform measurements were transmitted to the PC via the switch box (Figure 4.9). After the BP monitor had completed its measurement the switch box was reset allowing the BP monitor to transmit its measurements to the PC. Subsequently the PEFR and FEV_1 measurements were taken. The process is summarised in Figure 4.10.

4.3.2 Data acquisition software

A computer program was written by the author to expedite the process of data collection and enable recording of the biometric measurements on the PC. An example of the user interface is presented in Figure 4.11. The data acquisition software was written in the Java programming language (Sun Microsystems, USA). This programming language was used to write further programs discussed in Chapter 7 and motivation for this choice is given in Chapter 8. Suffice to say at this juncture, the main benefit to using Java is derived from its *object-oriented* nature. Java programs are typically constructed out of objects (*i.e.*, programmatic code blocks that have *attributes* and *behaviours* rather than variables and subroutines). It allows the user to expedite the process of writing programs that – due to their object-oriented nature – can easily integrate into other object-based programs. Other object-oriented languages such as C++ offer similar benefits, but Java has potential benefits such as platform independence, improved networking (and *worldwide web*) capabilities, *etc.* We believe that Java will provide substantial benefits as the results of the work presented here are

integrated, and to what we might call the *universal diagnostic machine* (discussed in Section 8.3).

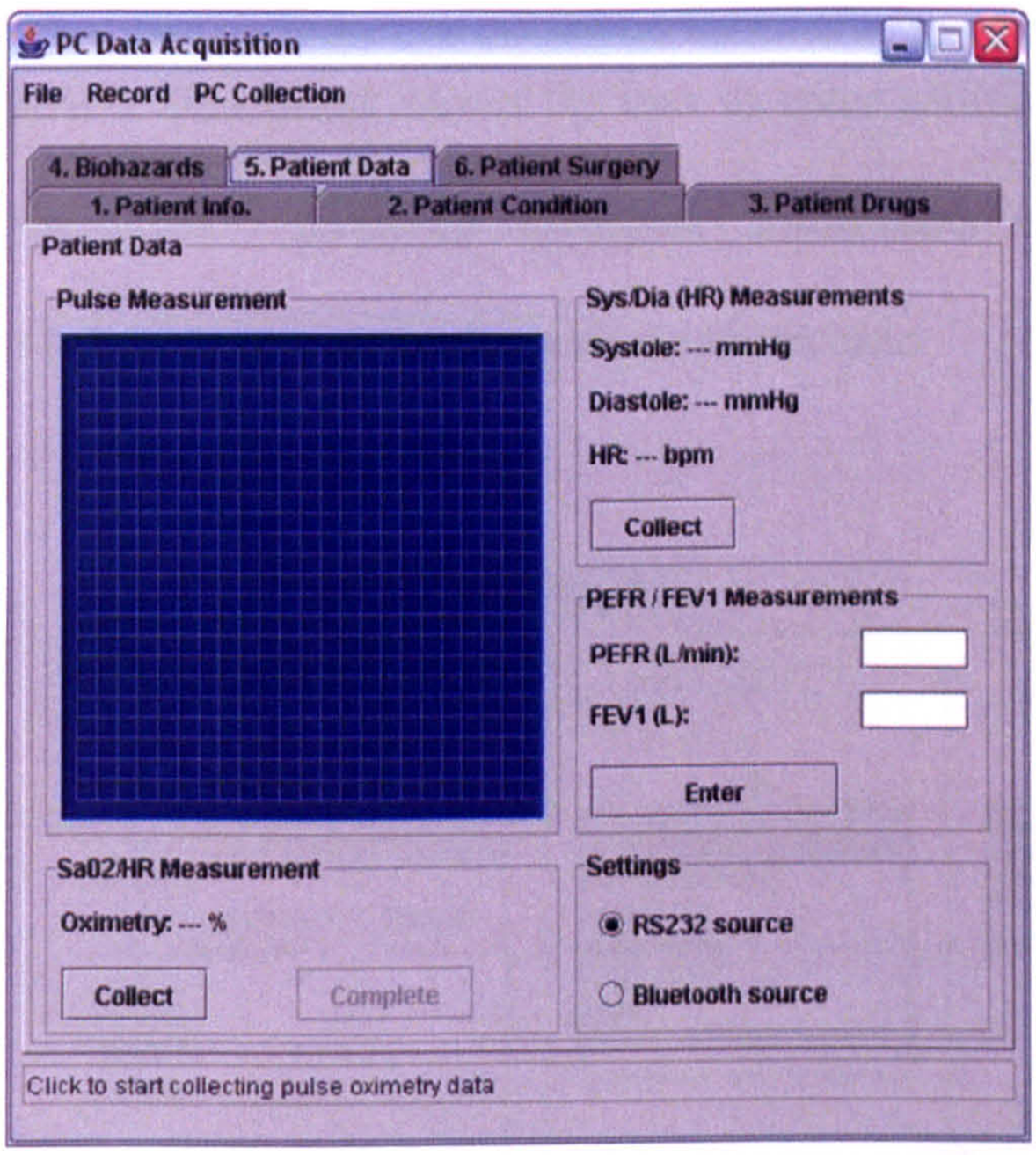


Figure 4.11. Graphical user interface for the data acquisition software.

The data acquisition program was structured into 6 tabbed panels:

- 'Patient Info.' panel. This panel allowed for the entry of basic patient information such as PID, age, gender, *etc.*
- 'Patient Condition' panel. This panel allowed for the entry of clinical history indicators (discussed in Section 4.4).
- 'Patient Drugs' panel. This panel recorded the medication the patient was taking that was relevant to the anaesthetist (discussed in Section 4.4).
- 'Biohazards' panel. This panel allowed for the entry of biohazard indicators such as MRSA infection, active Tuberculosis (TB), *etc.* These biohazard indicators are discussed in Section 4.4.

- 'Patient Data' panel. This panel (given in Figure 4.11) allows the user to start and stop acquiring data from the PC-connected equipment such as pulse oximeter, BP monitor, *etc.*
- 'Patient surgery'. This panel allows the user to enter additional information concerning the patient's proposed operation. Information entered into this panel is stored directly alongside the patient information.

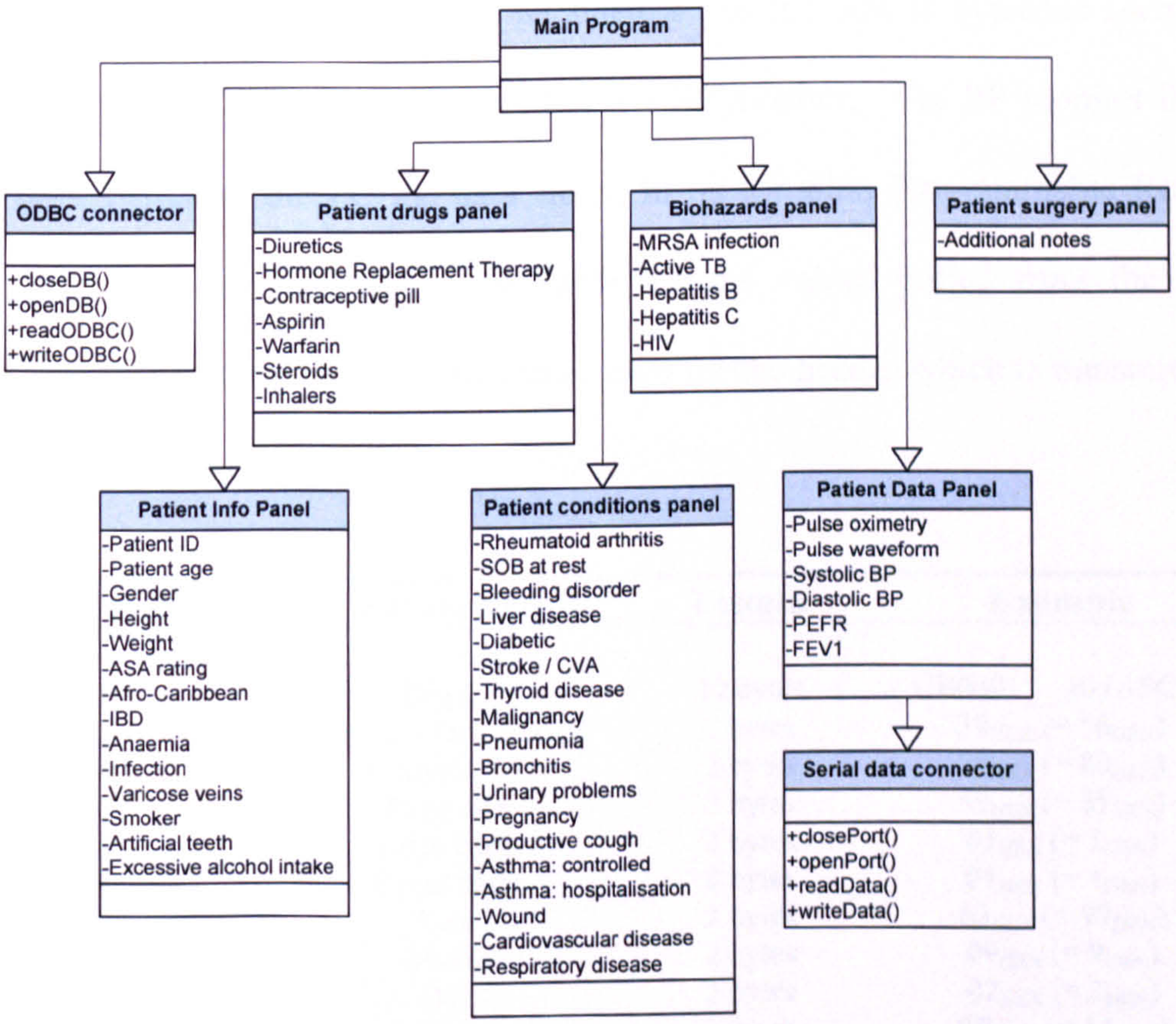


Figure 4.12. Structure of the data acquisition program and the information each panel requests.

4.3.3 Data transmission protocols

The data transmission protocol is defined as the way in which the PC communicates with the acquisition hardware and the manner in which the data is transferred from the device to the computer. This section gives a brief overview

of the data transmission protocols for the BP monitor, pulse oximeter and lung spirometer. The protocols are given as follows.

- **The BP monitor** was interfaced to the PC via an RS232 interface with a DB-9 terminated cable. The BP monitor used serial data transmission with the communication speed set at 4800 bits per second (bps), 8 data bits per frame, 1 start bit, 2 stop bits and no parity bits. The connection was opened to the device using the above parameters and the ASCII bytecode (denary equivalent) '240' was transmitted to the BP monitor. The BP monitor then polled back to the PC the data stored in its Erasable Programmable Read-Only Memory (EPROM). In Table 4.1 the values polled from the BP monitor are encoded in hexadecimal, save for the header which is transmitted in ASCII plain text.

Sequence No.	Parameter	Length	Example
1	Header	10 bytes	UB0401_ _20 (ASCII)
2	Systolic – Diastolic BP	2 bytes	38_{HEX} (= 56_{DEN})
3	Diastolic BP	2 bytes	50_{HEX} (= 80_{DEN})
4	Pulse rate	2 bytes	59_{HEX} (= 89_{DEN})
5	Error code	2 bytes	01_{HEX} (= 1_{DEN})
6	Event code	2 bytes	01_{HEX} (= 1_{DEN})
7	Year	2 bytes	63_{HEX} (= 99_{DEN})
8	Month	2 bytes	09_{HEX} (= 9_{DEN})
9	Day	2 bytes	02_{HEX} (= 2_{DEN})
10	Hour	2 bytes	$0E_{HEX}$ (= 14_{DEN})
11	Minute	2 bytes	$2D_{HEX}$ (= 45_{DEN})
12	Checksum	2 bytes	$8C_{HEX}$ (= 140_{DEN})

Table 4.1. Data transmission values for the BP monitor.

Datum sequence 2 in Table 4.1 is defined as the difference between systolic and diastolic BP in mmHg. Diastolic BP (sequence 3) is expressed in mmHg and pulse rate (sequence 4) in bpm. The error codes and event codes

(sequences 5 and 6 respectively) were inconsequential with regards to these experiments. Measurements were only accepted by the software if these codes were zero (*i.e.*, no error). The error and event codes did not relate to issues with patient physiology, rather to mechanical failings of the BP monitor which impeded the collection of reliable measurements. The year (sequence 7) represented the last two digits of the year whilst the checksum (sequence 12) was the last two digits of the hexadecimal sum of sequence 2 – 11. Upon transferring the string of values, the BP monitor closed the connection and awaited another connection attempt.

- **The pulse oximeter** was interfaced to the PC via an RS232 interface with a DB-9 terminated cable. When turned on, the oximeter attempted to make a connection to the attached fingerclip sensor and when connected it began calculating the SpO_2 values. The PC established a serial data connection at a speed of 9600 bps with 8 data bits per frame, 1 stop bit and no parity bits. The oximeter began polling the measurements at a rate of 75 times per second. 5 bytes were transmitted on every poll, which are detailed in Table 4.2. In Table 4.2, the first byte is the synchronisation character and was always '01'. The second byte is the status byte and was 128 during normal operation. In the event of errors (sensor misalignment, *etc.*), the status byte takes any value between 129 and 255. During this time the PC ignored all data from the oximeter until the error has been resolved and the status byte returned to the value 128.

Hz 1/75	Byte 1	2	3	4	5	Hz 1/75	Byte 1	2	3	4	5	Hz 1/75	Byte 1	2	3	4	5
1	01	ST	PLE	HR1	CS	**26**	01	ST	PLE	HR1	CS	**51**	01	ST	PLE	HR1	CS
2	01	ST	PLE	HR2	CS	**27**	01	ST	PLE	HR2	CS	**52**	01	ST	PLE	HR2	CS
3	01	ST	PLE	SpO_2	CS	**28**	01	ST	PLE	SpO_2	CS	**53**	01	ST	PLE	SpO_2	CS
4	01	ST	PLE	REV	CS	**29**	01	ST	PLE	REV	CS	**54**	01	ST	PLE	REV	CS
5	01	ST	PLE	*	CS	**30**	01	ST	PLE	*	CS	**55**	01	ST	PLE	*	CS
6	01	ST	PLE	*	CS	**31**	01	ST	PLE	*	CS	**56**	01	ST	PLE	*	CS
7	01	ST	PLE	*	CS	**32**	01	ST	PLE	*	CS	**57**	01	ST	PLE	*	CS
8	01	ST	PLE	*	CS	**33**	01	ST	PLE	*	CS	**58**	01	ST	PLE	*	CS
9	01	ST	PLE	*	CS	**34**	01	ST	PLE	*	CS	**59**	01	ST	PLE	*	CS
10	01	ST	PLE	*	CS	**35**	01	ST	PLE	*	CS	**60**	01	ST	PLE	*	CS
11	01	ST	PLE	*	CS	**36**	01	ST	PLE	*	CS	**61**	01	ST	PLE	*	CS
12	01	ST	PLE	*	CS	**37**	01	ST	PLE	*	CS	**62**	01	ST	PLE	*	CS
13	01	ST	PLE	*	CS	**38**	01	ST	PLE	*	CS	**63**	01	ST	PLE	*	CS
14	01	ST	PLE	*	CS	**39**	01	ST	PLE	*	CS	**64**	01	ST	PLE	*	CS
15	01	ST	PLE	*	CS	**40**	01	ST	PLE	*	CS	**65**	01	ST	PLE	*	CS
16	01	ST	PLE	*	CS	**41**	01	ST	PLE	*	CS	**66**	01	ST	PLE	*	CS
17	01	ST	PLE	*	CS	**42**	01	ST	PLE	*	CS	**67**	01	ST	PLE	*	CS
18	01	ST	PLE	*	CS	**43**	01	ST	PLE	*	CS	**68**	01	ST	PLE	*	CS
19	01	ST	PLE	*	CS	**44**	01	ST	PLE	*	CS	**69**	01	ST	PLE	*	CS
20	01	ST	PLE	*	CS	**45**	01	ST	PLE	*	CS	**70**	01	ST	PLE	*	CS
21	01	ST	PLE	*	CS	**46**	01	ST	PLE	*	CS	**71**	01	ST	PLE	*	CS
22	01	ST	PLE	*	CS	**47**	01	ST	PLE	*	CS	**72**	01	ST	PLE	*	CS
23	01	ST	PLE	*	CS	**48**	01	ST	PLE	*	CS	**73**	01	ST	PLE	*	CS
24	01	ST	PLE	*	CS	**49**	01	ST	PLE	*	CS	**74**	01	ST	PLE	*	CS
25	01	ST	PLE	*	CS	**50**	01	ST	PLE	*	CS	**75**	01	ST	PLE	*	CS

* Undefined

Table 4.2. Data transmission values for the pulse oximeter [12].

The third byte is the instantaneous arterial pulse waveform value. The fourth byte changed on every poll and was used to transmit HR1 and HR2 (heart rate most significant and least significant bits respectively), the SpO_2 measurement and hardware version number. The fourth byte is a checksum value calculated by summing the preceding 4 bytes. The undefined values given in Table 4.2 were typically zero. This transmission process continued until the device was switched off.

- **The lung function spirometer** was interfaced to the PC via an RS232 interface with a DB-9 terminated cable. When a measurement was taken by the spirometer, the device attempted to instantiate a serial data connection at a speed of 9600 bps with 8 data bits per frame, 1 stop bit and no parity bits.

The PEFR and FEV_1 measurements were transmitted as ASCII plain text which the data acquisition program parsed into numeric values.

This section has presented a summary of the biometric data that was acquired as part of this research. It has presented the methods by which the data was acquired and stored on the PC. The following section details the data acquired from the patient's anonymous clinical notes.

4.4 Clinical history indicators

As was discussed in Chapter 1, and further in Section 4.1.2, the patient's clinical history provides indications as to their physiological state. These indicators are grouped here into 5 categories determined by the author, *i.e.*, the categorisation is merely performed for clarification and does not affect the information content of the data.

- **Medication indicators.** These indicators relate to the medication that the patient is currently taking. These drugs may affect hormone production or other chemicals that are present in the patient. The drugs considered as part of the relevant medication indicators are given in Table 4.3. Patients taking any of these medicines may require further investigative tests prior to being deemed suitable for surgical anaesthesia.

- **Biohazard indicators.** These indicators relate to potentially dangerous infections that the patient may have. Common biohazard indicators are

given in Table 4.3. Testing positive for any of these indicators may require the patient to be referred for further investigative tests.

- **Haemodynamic indicators.** Haemodynamic abnormalities can affect blood chemistry which may have a detrimental effect during anaesthesia. For example, patients of an Afro-Caribbean origin are particularly susceptible to certain blood disorders such as sickle cell disease which can affect blood circulation and hence arterial oxygen transportation. The haemodynamic indicators (given in Table 4.3) must be carefully studied by the anaesthetist when determining the circulatory competence of the patient.

- **Cardiovascular indicators.** These indicators pertain to any patient activity or physiological abnormality that may impede normal cardiovascular function. For example, whether the patient is a 'heavy smoker' (*i.e.*, smokes more than 40 cigarettes (or equivalent) per day) is an important indicator.

- **Respiratory indicators.** These indicators can affect respiratory competence which can detrimentally affect the patient's risk of mortality during surgical anaesthesia. As discussed in Section 1.2.2 and Section 4.1.2, adequate respiratory competence is essential for minimising patient risk during surgical anaesthesia. Testing positive for any of the respiratory indicators could motivate the anaesthetist to refer

the patient for further investigative tests prior to deeming them suitable for surgical anaesthesia.

The clinical history indicators were taken from the patient's notes during the consultation period by the physician, using protocols devised by the author. Table 4.3 presents each category with the associated indicators.

Medication indicators	**Biohazard indicators**	**Haemodynamic indicators**	**Cardiovascular indicators**	**Respiratory indicators**
Warfarin	Hepatitis B	Afro-Caribbean	Smoker (>40 per day)	Smoker (>40 per day)
Inhalers	Hepatitis C	Excessive alcohol intake	Cardiovascular disease	Pneumonia
HRT	MRSA	Infection	Varicose veins	Chronic bronchitis
Steroids	Active TB	Blood disorders	Diabetes	Asthma: controlled
Aspirin	HIV	Anaemia		Asthma: hospitalisation
Diuretics		Diabetes		Shortness of breath at rest
Contraceptive pill		Kidney problems		Respiratory disease history
		Urinary problems		Productive cough
		Bleeding disorders		
		Thyroid disease		
		Malignancy		
		Open wound		

Table 4.3. Clinical history indicators considered as part of this research.

The indicators in Table 4.3 were recorded alongside PID on the information sheet prior to the biometric data acquisition presented in Section 4.3. Following the consultation, the data on the information sheet was recorded alongside the biometric measurements. A blank information sheet is included in Appendix II, whilst the final format of the acquired patient data is discussed in the following section.

Figure 4.13. Sequence of data acquisition events.

For clarity, the processes of data acquisition given in Section 4.2 through Section 4.4 are summarised in the diagram presented in Figure 4.13.

4.5 Dataset construction

The result of the process given in Figure 4.13 was a database of 194 anonymous patients. The database contained BP, HR, SpO_2, PEFR and FEV_1 measurements; digital audio recordings of the heart and lung sounds; and categorised clinical history indicators. It was proposed by the author that the dataset contained sufficient information on which to base a decision as to whether a patient was suitable for surgical anaesthesia or not. This proposition is validated in Chapter 7 with the results obtained from the decision support system. Chapters 5, 6 and 7 discuss the various means by which intelligent systems were trained to recognise physiological abnormalities from this dataset, and hence identify patients who are unsuitable for surgical anaesthesia.

As was discussed in Section 1.4, the problem of identifying whether a patient is suitable for surgical anaesthesia, based on their physiological assessment, is broken down into a series of manageable sub-problems, *e.g.*, listening to the heart sounds; then listening to the lung sounds; then checking BP; and so forth. Individual intelligent systems are required to 'learn solutions' to these sub-problems. As such the patient dataset was separated into three components: cardio-respiratory sounds (used in Chapter 5); pulse oximetry and arterial pulse waveforms (used in Chapter 6); and biometrics and clinical indicators (used in Chapter 7). In each instance, individual intelligent systems were applied to the

analysis of these components. Chapters 5, 6 and 7 respectively address the three problems that need to be solved in order to derive a DSA:

1. diagnosing cardio-respiratory abnormalities via auscultation;
2. identifying valid arterial pulse waveforms, and hence the SpO_2 measurement; and
3. forming a generalised classifier for the accurate appraisal of patients for surgical anaesthesia.

4.6 Summary

This chapter has described the types of data considered as part of this research, and outlines the motivation for their analysis. The types of hardware employed in the data collection process have also been discussed alongside the experimental procedure used. The data transmission protocols – which give an appreciation of how the hardware reports these biometric measurements – are also described. The chapter finishes by identifying that a compartmentalised approach to creating a DSA presents two requirements. These requirements are: that there exists a pre-DSA classifier that can diagnose cardio-respiratory abnormalities; and that there exists a pre-DSA mechanism that can remove corrupt arterial pulse waveforms and hence accurately measure SpO_2. These two requirements are respectively addressed in Chapter 5 and Chapter 6. Referring to Figure 4.2, this chapter has discussed the protocols used in acquiring the medical datasets. In Chapters 5, 6 and (partially) 7, this data is processed to elicit information regarding the biological processes such as cardio-respiratory pathologies, indicative features of motion-corrupted arterial pulse waveforms,

etc. The information extracted from this data is subsequently used in Chapter 7 to form the DSA, and hence model, and extract the knowledge of the anaesthetist for performing the task of assessing patient suitability for surgical anaesthesia.

References

[1] Buchanan, B.G. and Shortliffe, E.H., *MYCIN experiments of the Stanford heuristic programming project*, Addison Wesley, Reading, MA, 1984.

[2] Daniel, M., Hájek, P. and Nguyen, P.H., CADIAG-2 and MYCIN-like systems, *Artificial Intelligence in Medicine*, 9(3), 1997, pp.241-259.

[3] Cruz, G.P.A. and Beliakov, On the interpretation of certainty factors in expert systems, *Artificial Intelligence in Medicine*, 8(1), 1996, pp.1-14.

[4] Rabelo Jr., A., Rocha, A.R., Oliveira, K., Souza, A., Ximenes, A., Andrade, C., Onnis, D., Olivaes, I., Lobo, N., Ferreira, N. and Werneck, V., An expert system for diagnosis of acute myocardial infarction with ECG analysis, *Artificial Intelligence in Medicine*, **10**(1), 1997, pp.75-92.

[5] Robnik-Šikonja, M., Cukjati, D. and Kononenko, I., Comprehensible evaluation of prognostic factors and prediction of wound healing, *Artificial Intelligence in Medicine*, **29**(1), 2003, pp.25-38.

[6] Weitz, H.H. and Mangione, S., In defence of the stethoscope and the bedside, *The American Journal of Medicine*, **108**(8), 2000, pp.669-671.

[7] *The 2001 report of the National Confidential Enquiry into Peri-Operative Deaths*, National Confidential Enquiry into Peri-Operative Deaths, London, 2001.

[8] *Preoperative tests: the use of routine preoperative tests for elective surgery*, National Institute for Clinical Excellence, Oaktree Press: London, 2003.

[9] Vohra, A., The role of the anaesthetist: replacement brain, *Anaesthesia*, **56**, 2001, pp.272-273.

[10] *Brunner and Suddarth's textbook of medical-surgical nursing*, 9th Ed., (Eds., Smeltzer, S.G. and Bare, B.G.), Lippincott Williams & Wilkins: Philadelphia, PA, 2000.

[11] Weissman, N.J., Panza, J.A., Tighe, J.F., Perras, S.T., Kushner, H. and Gottdiener, J.S., Specificity of Doppler echocardiography for the assessment of changes in valvular regurgitation: comparison of side-by-side versus serial interpretation, *Journal of the American College of Cardiology*, **37**(6), 2001, pp.1614-1621.

[12] *Nonin OEM II Module Specifications*, Nonin Medical Inc.: Minneapolis, MN, 2000.

CHAPTER 5

Cardio-respiratory Data Analysis

The aim of this chapter is to provide an account of how an ANN ensemble was developed to determine cardio-respiratory abnormalities based on auscultated sounds. An account of the data pre-processing methods and subsequent ANN analysis is presented. The chapter concludes with a statement of how the final ANN ensemble can be applied to patient diagnosis.

5.1 Introduction

In Chapter 1 we discussed the importance of cardio-respiratory examination in the assessment of a patient's suitability for surgical anaesthesia. We also discussed the requirement that an intelligent systems-based technique is developed to identify cardio-respiratory abnormalities. In this chapter we discuss a novel means for the automated classification of abnormal heart and lung sounds using ANNs and demonstrate that these methods can be used to successfully

diagnose cardio-respiratory pathologies and other abnormalities. The chapter begins with a description of the cardio-respiratory abnormalities under consideration, followed in Section 5.3 by an account of how these sounds were investigated. Section 5.4 details ANN classification of these sounds whilst Section 5.6 describes a method which can potentially improve classification performance.

Figure 4.2 presented the relationships between data, information and knowledge. This chapter is concerned with the processing of cardio-respiratory 'sound data' and the subsequent elicitation of information. The information derived from this work exists in the form of probability statements relating heart and lung sounds to their respective pathologies. These probability statements are learned by the ANN implementations as part of their learning processes. We now extend Figure 4.2 and show the sequence of data processing events, presented in Figure 5.1.

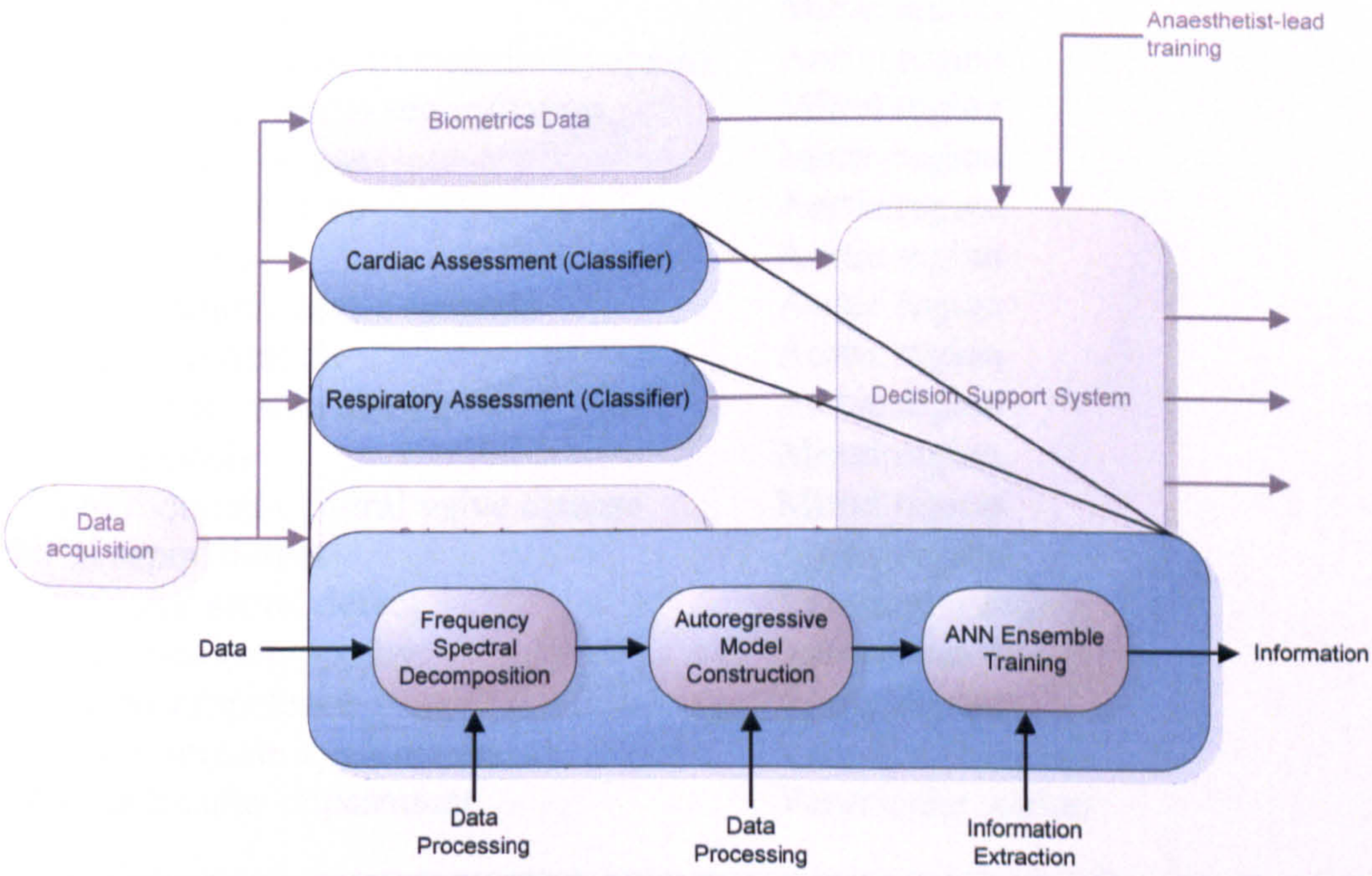


Figure 5.1. Data processing and information extraction.

Figure 5.1 shows the sequence of data processing events as part of this chapter, and how they relate to the overall decision support system. The following section discusses the cardiac and respiratory pathologies that are considered as part of this work.

5.2 Cardiac and respiratory pathologies

The audible sounds heard during auscultation are the consequence of either normal or abnormal cardio-respiratory function. Abnormal sounds are frequently the result of some pathology or defect. The different cardiac and respiratory sounds that the ANNs were trained to recognise are presented in Table 5.1 and Table 5.2 respectively. The regions of auscultation are also indicated in both cases.

Cardiac abnormality	Region of auscultation
Normal heart sound	Mitral region
Hypertrophic obstructive cardiomyopathy	Aortic region
Non-rheumatic mitral regurgitation	Mitral region
Mitral valve prolapse	Mitral region
Bicuspid aortic valve	Aortic region
Aortic stenosis	Aortic region
Severe calciphic aortic stenosis	Aortic region
Aortic regurgitation	Aortic region
Mixed aortic valve disease	Aortic region
Mitral stenosis	Mitral region
Mixed rheumatic mitral valve disease	Mitral region
Atrial septal defect	Aortic region
Ventricular septal defect	Ventricular region
Dilated cardiomyopothy	Ventricular region
Aortic incompetence	Aortic region
Papillary muscle dysfunction	Ventricular region
Left ventricular impairment	Ventricular region

Table 5.1. Cardiac abnormalities the ANNs were trained to recognise.

Respiratory abnormality	**Region of auscultation**
Normal tracheal-bronchial sounds	Tracheal region
Normal broncho-vesicular sounds	Bronchial region
Normal vesicular sounds	Bronchial region
Diminished sounds	Bronchial region
Tubular breath sounds	Bronchial region
Fine inspiratory crackles	Bronchial region
Medium, coarse inspiratory and expiratory crackles	Bronchial region
Fine, late inspiratory crackles	Bronchial region
Mild expiratory wheeze	Bronchial region
Medium inspiratory crackles with moderate and severe expiratory wheeze	Bronchial region
Pleural friction rub	Bronchial region
Inspiratory and expiratory stridor	Tracheal region
Bone crepitus	Bronchial region
Subcutaneous emphysema	Bronchial region

Table 5.2. Respiratory abnormalities the ANNs were trained to recognise.

These sounds are the most commonly occurring and are deemed important by anaesthetists as to whether the patient is suitable for surgical anaesthesia [1].

Two separate ANN ensembles were trained: one which classified cardiac auscultated sounds into the classes given in Table 5.1; and one which classified respiratory auscultated sounds into the classes given in Table 5.2. Prior to ANN training, the acquired samples were *pre-processed* using methods which are discussed in the next section.

5.3 Cardio-respiratory data pre-processing

Data pre-processing is defined as the processes that are undertaken prior to data analysis (*e.g.*, ANN classification). These processes are used primarily to address several key issues when analysing data, the most pertinent being

dimensionality reduction and *feature extraction*. Real-world data is frequently unsuitable for direct analysis because the information it contains is inefficiently represented [2]. For example, given a dataset pertaining to some process we wish to model, each datum within the set is N dimensional (*i.e.*, N variables that describe the data). It is often the case that a subset of these dimensions is irrelevant as it contains no apparently useful information (*e.g.*, the dimensions are correlated with each other). Dimensionality reduction is the process by which one identifies and removes redundant dimensions from the dataset. By removing apparently unimportant dimensions we retain the informative *features* of the dataset which may be used in a subsequent analysis. The importance of feature extraction is exemplified by Hartman *et al.*, [3] and was discussed in Chapter 3. Figure 5.2 summarises the data processing stages performed in this chapter. The methods employed for pre-processing the cardiac and respiratory sounds are presented in the following sections alongside the rationale behind their use.

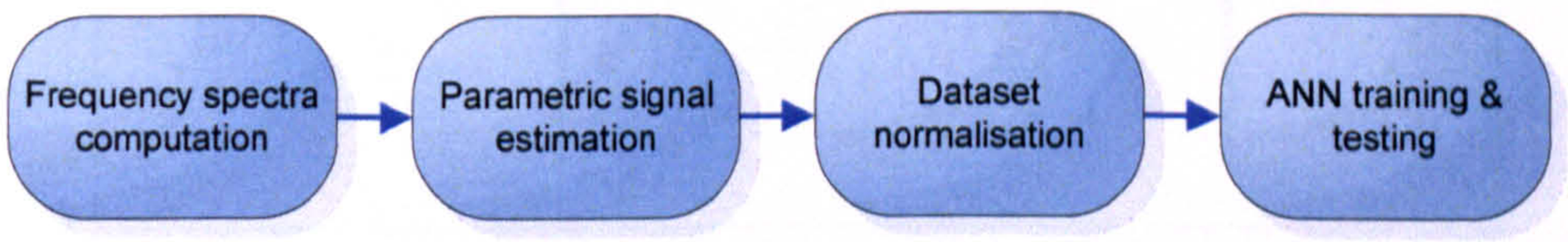


Figure 5.2. Data processing stages used for information extraction.

5.3.1 Fourier analysis

The Fourier Transform $\mathcal{F}$ is based on the principle that periodic, symmetric, real-world signals can be decomposed into a series of N sinusoidal waves of frequency ω_n and phase ϕ_n $(1 \leq n \leq N)$.

Recalling the methods of data acquisition in Section 4.2, the auscultated sounds were stored on a PC in uncompressed form. The sounds were sampled at a rate of 44kHz and were approximately 30 seconds in duration. The following processes were performed separately for the cardiac and respiratory sounds.

1. Each sound was imported into MATLAB 6.5 (Mathworks Inc., USA) which represented the sample as a column vector of approximately 1.32×10^6 elements (*i.e.*, ~30 seconds × 44 kHz). Unless otherwise stated all computations were performed in MATLAB 6.5.
2. A matrix was created by importing the auscultated sounds and transposing the vectors (so they became row vectors). Considering that each vector was of a different length they were cropped to the size of the smallest vector. The resultant matrix was $\mathbf{X} = \{x_{ij}\}$ where x_{ij} was the jth element of sample i.

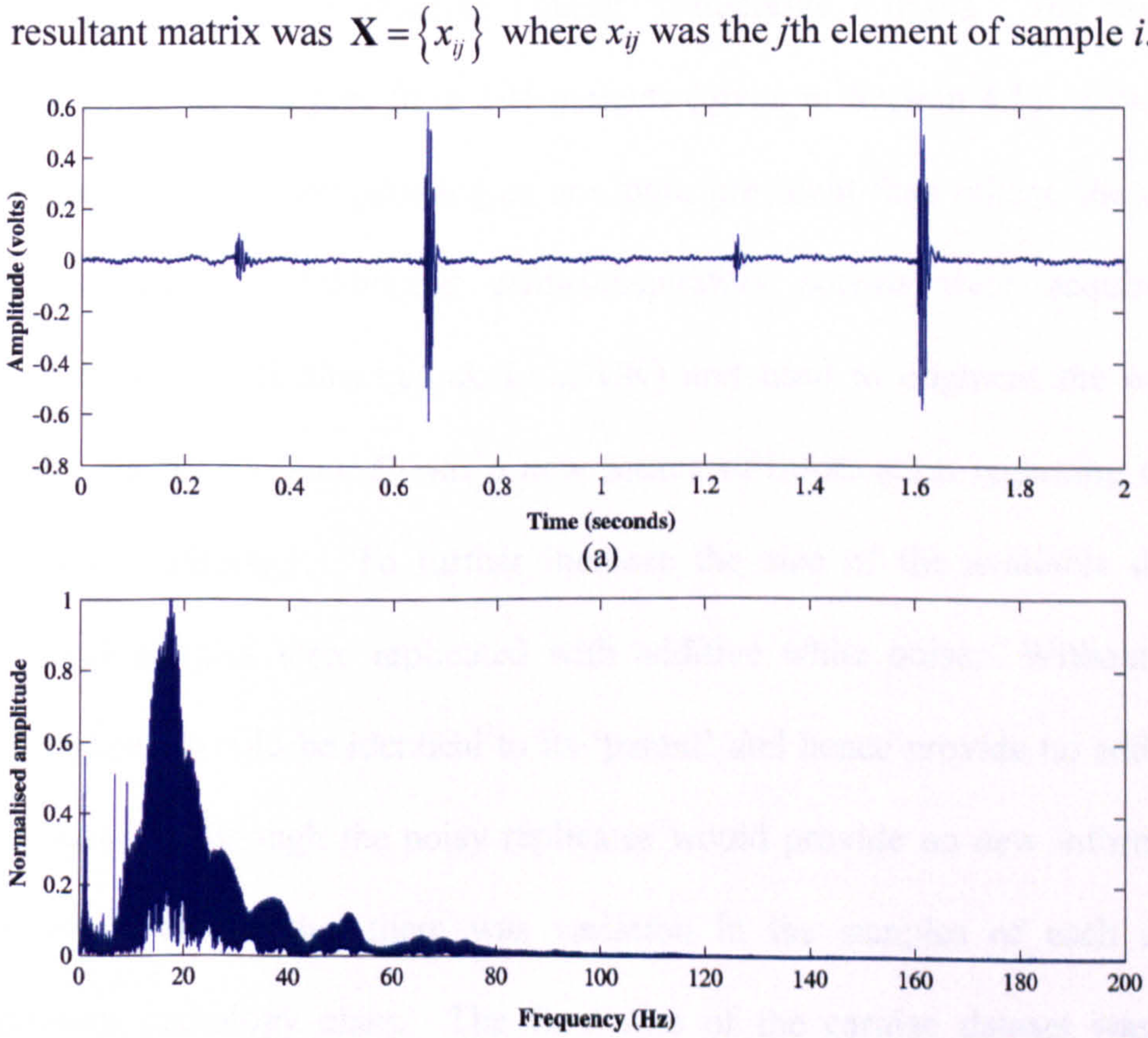


Figure 5.3. Example of (a) 2 normal cardiac sound cycles; and (b) the associated frequency spectrum.

3. The frequency spectra of the acquired sounds were computed using the Fast Fourier Transform (FFT) [4] which afforded a frequency domain representation of the sounds. The frequency domain representation was chosen over the temporal domain for two reasons; 1) that it is resilient to variations in the cardio-respiratory cycle *i.e.*, heart-rate or respiratory-rate variations; and 2) it readily contains information that is difficult to detect in the temporal domain [5].

We recall that the observed frequencies for cardiac auscultation cut off at 200Hz, and 500Hz for respiratory auscultation. Figure 5.3 shows (a) a normal heart sound over 2 seconds auscultated in the aortic area, and (b) the associated normalised frequency spectrum. Matrices were constructed containing the absolute values of the Discrete Fourier Transforms (DFTs). The available dataset contained samples from 194 patients (given in Section 4.5). Given that some cardio-respiratory pathologies are more prevalent than others, the dataset was unbalanced. Additional cardio-respiratory sounds were acquired on compact-disc (Medicdirect.co.uk Ltd., UK) and used to augment the existing dataset. Each new 'sound' was a new source of information regarding cardio-respiratory pathology. To further increase the size of the available dataset, individual samples were replicated with additive white noise. Without noise each replicate would be identical to its 'parent' and hence provide no additional information. Although the noisy replicates would provide no new information, they would ensure that there was variation in the samples of each cardio-respiratory pathology class. The final size of the cardiac dataset was 1700 samples, and the respiratory dataset was 1400 samples.

Given that the cardiac DFTs presented 200 dimensions (corresponding to the 200Hz range) and the respiratory DFTs had 500 dimensions (corresponding to the 500Hz range), visual inspection of the dataset heterogeneity was not possible. Principal Component Analysis (PCA) affords a means by which to reduce the dimensionality of the dataset whilst retaining as much information as possible [2]. This method was applied in order to facilitate an appraisal of dataset heterogeneity.

5.3.2 Principal Component Analysis

Principal Component Analysis – otherwise knows as the *Karhunen-Loéve Transform* – affords a method with which to remap n-dimensional datasets to m-dimensional space where $m < n$ (typically $m << n$) [2].

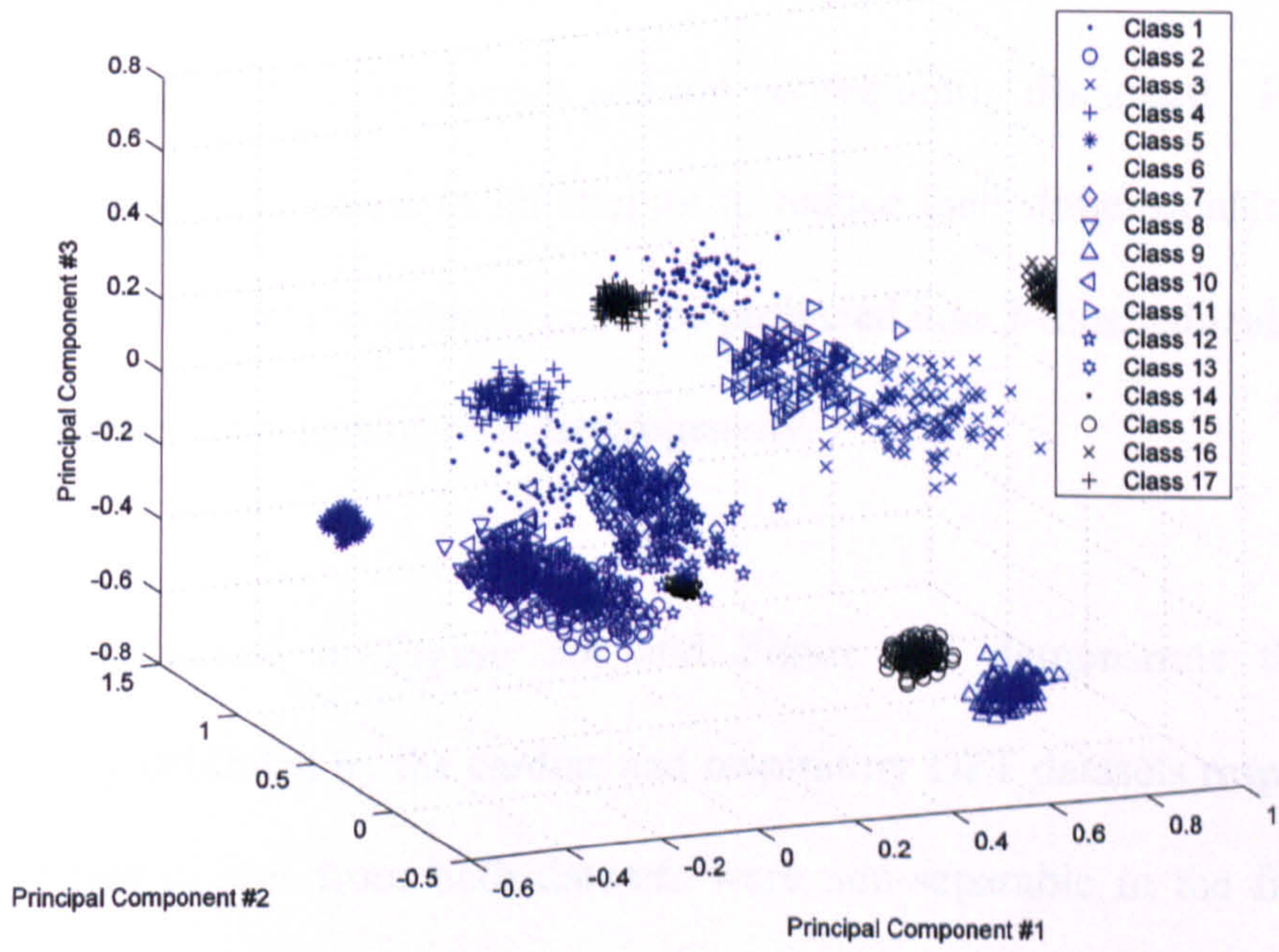


Figure 5.4. Plot of the first 3 principal components of the cardiac DFTs.

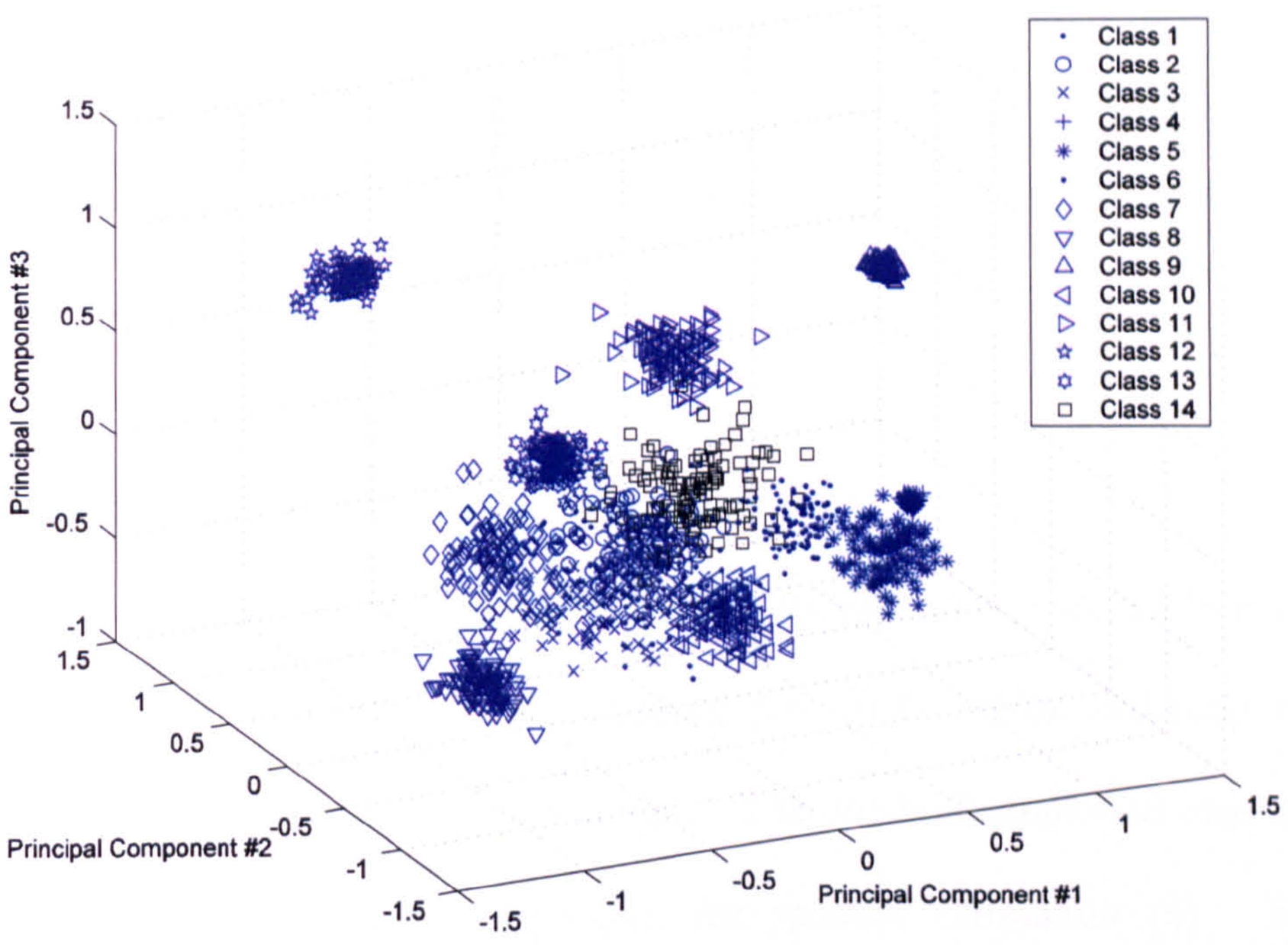


Figure 5.5. Plot of the first 3 principal components of the respiratory DFTs.

This is achieved by firstly evaluating the eigenvalues and eigenvectors of the dataset's covariance matrix. The smallest $n-m$ eigenvalues of the matrix afford the least information in the dataset and are consequently discarded. PCA was applied to the DFT datasets in an attempt to reduce their dimensionality. This was performed so that the datasets could be projected into 3-dimensional space to facilitate visual inspection of class heterogeneity.

The plots presented in Figure 5.4 and Figure 5.5 demonstrate the class heterogeneity exhibited by the cardiac and respiratory DFT datasets respectively. Whilst certain classes from both datasets were non-separable in the first three dimensions, a degree of confidence in overall heterogeneity could be ascribed to both datasets. A statistical measure as to the degree of separability cannot easily be given as such a value is conditional on the number of dimensions considered

in the PCA. Given a visual inspection of Figures 5.4 and 5.5, analysis of the cardiac and respiratory sounds by ANNs could be undertaken with a degree of confidence that the ANNs could separate the different classes and hence classify future (*i.e.*, novel) sounds.

5.3.3 Spectral estimation with autoregression

Direct analysis of these spectra by ANNs was not feasible due to the DFTs being highly-dimensional. A method of encoding the DFTs whilst reducing the dimensionality of the dataset was required. The method employed was the *Levinson-Durbin* autoregression algorithm for spectral estimation [6]. The method is a computationally light-weight approach to parametric signal estimation, constructing a *P*-order data model that describes the frequency spectra [6]. Pardey *et al.*, [7] found the method to be attractive in spectral matching problems. Given a data model $\hat{s}_x(f)$ which is a *P*-order estimate of signal *s*, information from *s* can be encoded into the estimation model parameters. We consider each DFT to be a frequency spectral composition $s_x(f)$ of the discrete sound sample $x(n)$ such that:

$$s_x(f) = \sum_{m=-\infty}^{\infty} R(m) e^{-j2\pi mf\Delta t}, \tag{5.1}$$

where $R(m)$ is the autocorrelation function of $x(n)$ and the unit time Δt is 1.

The normalised maximum entropy spectral estimation $\hat{s}_x(f)$ is expressed as:

$$\hat{s}_x(f) = \frac{1}{\left|1 + \sum_{k=1}^{P} a_k e^{-j2\pi kf}\right|^2}. \tag{5.2}$$

The set of values $\{a_k \mid k=1,...,P\}$ in (5.2) are coefficients of the signal estimation model $\hat{s}_x(f)$. Appropriate selection of P and a_k is required to form an adequate model such that $\hat{s}_x(f) \to s_x(f)$. The coefficients must satisfy:

$$\sum_{k=0}^{P} a_k R(m-k) = \begin{cases} 1, & m=0 \\ 0, & m=1,2,...,P \end{cases}, \tag{5.3}$$

which can be expanded in the form of the *Yule-Walker* matrix [6]:

$$\begin{pmatrix} R(0) & R(-1) & \cdots & R(-P) \\ R(1) & R(0) & \cdots & R(1-P) \\ \vdots & \vdots & \ddots & \vdots \\ R(P) & R(P-1) & \cdots & R(0) \end{pmatrix} \begin{pmatrix} 1 \\ a_1 \\ a_2 \\ \vdots \\ a_P \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix}. \tag{5.4}$$

The Levinson-Durbin algorithm estimates the frequency spectrum by firstly estimating the $P+1$ autocorrelation values $\hat{R}(k)$ from the initial sample $x(n)$ where $\{x(n) \mid n=1,...,N-1\}$ with:

$$\hat{R}(k) = \frac{1}{N} \sum_{n=0}^{N-k-1} x^*(n+k)x(n), \quad k=0,1,...,P. \tag{5.5}$$

The algorithm then constructs the Yule-Walker matrix given in (5.4) and calculates the estimated model parameters $\hat{a}_k$ with (5.3) by substituting $R(m)$ for $\hat{R}(m)$ and a_k for $\hat{a}_k$.

The Levinson-Durbin method of spectral estimation is not without problems. Luo *et al.*, [6] noted that the computational complexity of the exercise prevents

real-time spectral estimation. In this case the datasets were intended to be analysed offline and hence real-time estimation was not a consideration. This also fulfils the requirement that this form of parametric signal estimation be only used in stationary processes [6,7]. Christini *et al.*, [8] noted that sharp spikes in the original signal can also impede estimation accuracy; a consideration not required as the DFTs were smooth over their frequency ranges. A final consideration is from de Hoon *et al.*, [9] who found that nearly-periodic signals can result in incorrect parameter estimations; again a consideration not required as the spectral signals were aperiodic. Given that the Levinson-Durbin algorithm was capable of computing the optimal values of the autoregressive model, the most important criterion was the appropriate selection of the number of parameters to use in the estimation.

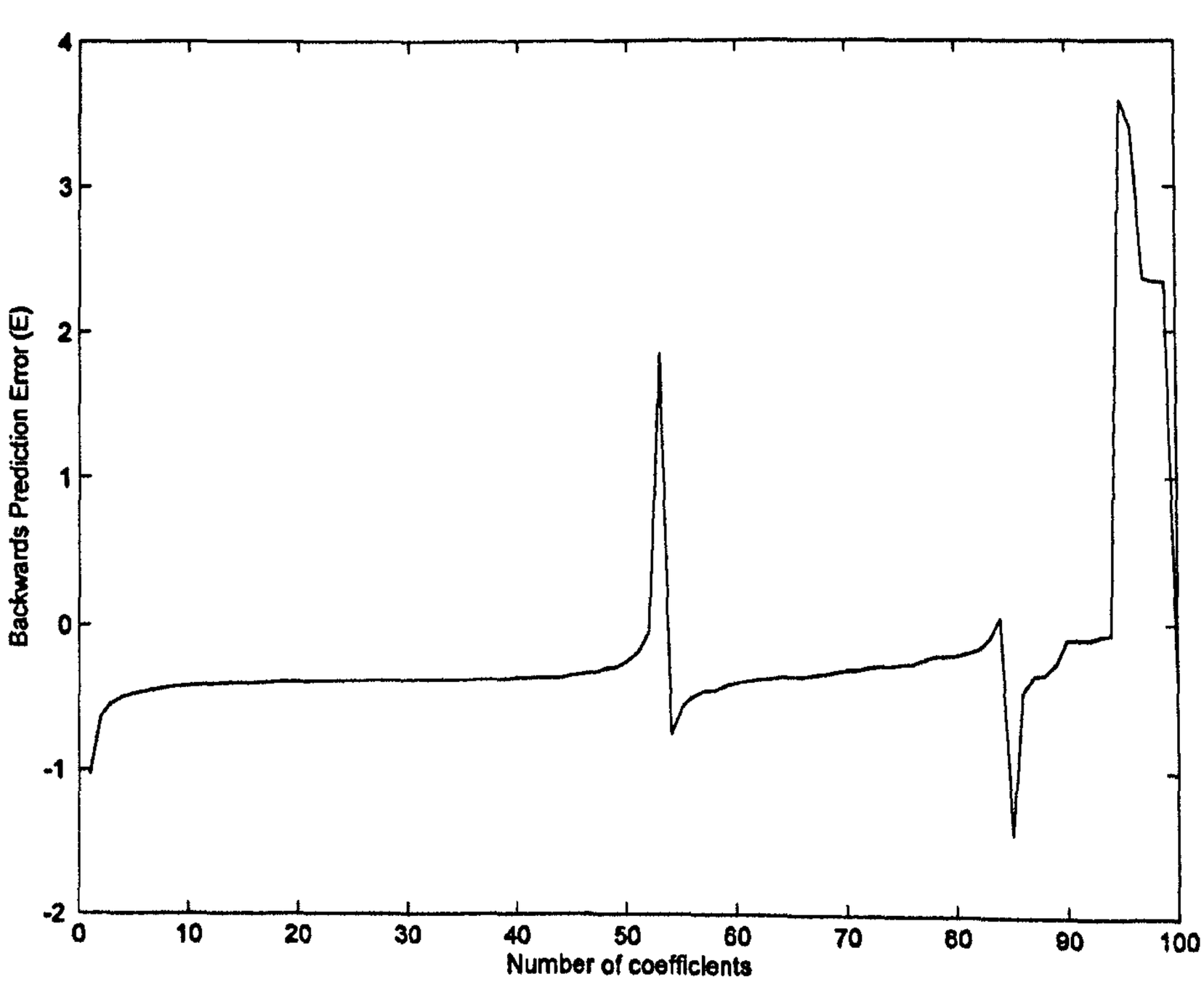


Figure 5.6. Backwards prediction error (SSE) for normal cardiac sound spectral estimation vs. autoregressive model parameters.

From (5.2) and (5.5) it follows that the number of parameters P influences the error between $s_x(f)$ and $\hat{s}_x(f)$ – called the *residual error*. The spectral estimation of a normal cardiac DFT was computed with models of order P = 1,2,...,100 which can be seen in Figure 5.6. From visual inspection of Figure 5.3b, models of order $P<10$ were insufficiently complex enough to capture the information content of the DFT and were therefore discounted. However, at P = 15 the SSE curve became asymptotic to the horizontal (Figure 5.6) indicating that further increases in the number of coefficients did not significantly change the error. Consequently a model order of 15 coefficients was selected. Given that each frequency spectrum was unique to a class of cardiac and respiratory abnormalities, the associated estimated spectral model would also be unique with this being reflected in the model coefficients. The Levinson-Durbin algorithm was applied to all of the frequency spectra for both the cardiac and respiratory sounds. This resulted in two matrices, one containing cardiac DFT estimation coefficients and the other containing respiratory DFT estimation coefficients, both containing 15 columns. The coefficients for each sample were linearly normalised such that $\{a_k \mid k=1,2,...,15\} \rightarrow [0,1]$ with:

$$\bar{a}_k = \frac{a_{\max} - a_k}{a_{\max} - a_{\min}},\ k=1,2,...,15, \tag{5.6}$$

where:

$$a_{\max} = \arg\max_{1\le n\le 15}\{a_n\} \text{ and } a_{\min} = \arg\min_{1\le n\le 15}\{a_n\}. \tag{5.7}$$

ANNs were trained to discriminate between the different cardiac and respiratory sounds based on these model coefficients of their estimated frequency spectra.

The following sections introduce the procedures underpinning the training and testing of the ANNs alongside the results obtained. Section 5.4 follows the processes undertaken to develop an ANN-based classification system in conjunction with the auscultated cardiac and respiratory sounds.

5.4 Classification by Artificial Neural Network

This section details the ANN approaches that were used in classifying the cardiac and respiratory sounds. We recall the result of the data pre-processing stages as being a matrix of spectral estimation model coefficients which were representative of the frequency spectra of the sounds. We recall the operation of an ANN from Chapter 3, in particular supervised learning methods. We now discuss the application of the three ANN architectures, notably the MLP, RBFN and CPNN.

5.4.1 Analysis by Multi-Layer Perceptron

The MLP was applied in the discrimination between the different classes of cardiac and respiratory sounds. The 3-layer topology (*i.e.*, input layer, hidden layer, and output layer) MLP was employed. The size of the MLP topology was firstly estimated using the geometric mean approach suggested by Widrow *et al.*, [10], and was later validated against Bayesian Regularisation after training (discussed later in this section). Firstly discussed is the development of an MLP

intended for the classification of the cardiac dataset. This is followed by a similar description concerning the respiratory dataset.

5.4.2 MLP analysis of the cardiac dataset

The cardiac dataset contained 1700 samples which were used in the training and subsequent testing of the MLP. Given that the dataset contained 15 dimensions (features), the input layer to the MLP contained 15 input neurones. Furthermore, the output layer contained one neurone per target class resulting in 17 output neurones. Provided with these two layers, a single hidden layer comprising 15 neurones was added to the ANN (the geometric mean of inputs and outputs). The resulting topology is referred to a *15-15-17* MLP. There are alternative methods of estimating the number of hidden layer neurones. For example, one of these methods is the *Cascade-Correlation* architecture proposed by Fahlman *et al.*, [11,12]. The method involves the continual addition of hidden layer neurones and the successive recalculation of the network weight matrix in growing the MLP. The method constitutes a significant departure from the error backpropagation algorithm discussed in Chapter 3 and is a time-consuming process. This coupled with the existing ability to assess MLP capacity utilisation with Bayesian Regularisation motivated the exclusion of this method from consideration. The MLP (consisting of 512 neuronal weights and biases) was trained with 1000 samples for 2500 epochs after which the network training SSE differential became negligible. Given that there were 1000 training samples, and 512 network parameters, upon first inspection it would appear that there were too few parameters available for effective learning. However, we recall that approximately 40% of the available samples were noisy replicates, and hence

contributed no additional information. The MLP was tested with the remaining 700 data samples, from which a results matrix was formed (of dimensions 700 × 17). Given that each MLP response y took some value $y \rightarrow [1,0]$ a threshold θ was used to convert these results to a binary 1-in-N equivalent such that:

$$y_{\text{bin}} = \begin{cases} 1: & y \geq \theta \\ 0: & y < \theta \end{cases} \tag{5.8}$$

With the threshold θ initialised to zero and incremented by 0.01, the optimal θ was found to be 0.30, yielding an MLP classification accuracy of 99.20% with sensitivity of 88.24% and specificity of 99.88%. On the training dataset the MLP correctly classified 98.56% of the samples (84.10% sensitivity, 99.47% specificity, $\theta = 0.33$).

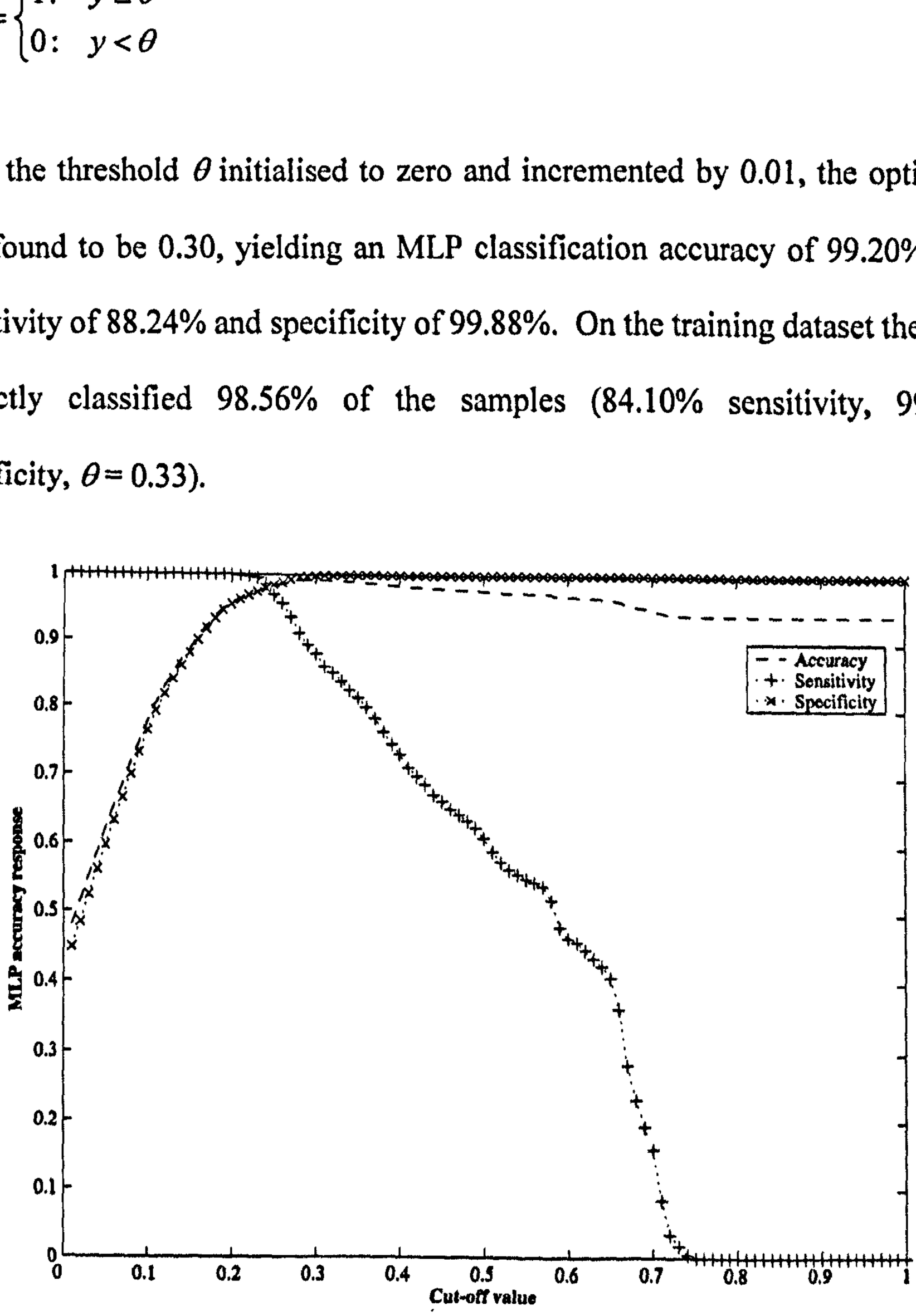


Figure 5.7. MLP accuracy, sensitivity and specificity in classifying the cardiac auscultated test dataset.

The classification accuracy curve alongside sensitivity and specificity is presented in Figure 5.7.

Prior to accepting these accuracy, sensitivity and specificity values as being true indicators of MLP performance, an assessment of MLP capacity utilisation was performed. The ability of an MLP to classify input data inherently depends on the ANN's neuronal weights and biases. Bayesian Regularisation [2] affords a means of estimating these parameters in a MLP through application of the Bayes Theorem (discussed in Section 3.1.1). Prior to training a MLP, the optimal network weights are unknown and can be represented as a broad prior probability distribution $P(\mathbf{W})$ over weight space. When input and target data D is presented to the MLP, the posterior probability distribution $P(\mathbf{W}|D)$ (*i.e.*, the probability of the weight matrix $\mathbf{W}$ given the data D) using (3.1) can be computed. Through successive calculations of $P(\mathbf{W}|D)$ the weight matrix can be estimated using the Bayesian technique. A more in depth account of this technique can be found in [2]. The method is called Bayesian Regularisation and gives the number of weights and biases being used by the MLP (called the *effective parameters*). If the number of effective parameters is smaller than the total number of parameters of the MLP then it can be concluded that the size of the architecture is not constraining its learning and classification abilities.

The MLP was re-initialised and trained using the above technique with the aim of identifying the number of effective parameters used in the learning process and ergo whether the MLP was being constrained by its architecture. The number of effective parameters used by the MLP was 253 (out of a total of 512

weights and biases) and it was concluded that the size of the MLP topology did not constrain its learning abilities. The results from these experiments are discussed in more detail in Section 5.5, whilst attention is now given to the MLP's abilities in discriminating between different classes of respiratory sounds.

5.4.3 MLP analysis of the respiratory dataset

The processes used to train an MLP in identifying different classes of respiratory abnormalities were similar to those used in the previous cardiac sounds experiments. We recall that the respiratory dataset contained 1400 samples. Recalling Section 5.3.3 the respiratory dataset was a matrix consisting of 15 columns (corresponding to 15 autoregressive coefficients). The resulting MLP topology was a 15-15-14 architecture. The MLP was trained for 2500 epochs on a dataset of size 1000 × 15 and was later tested against a dataset of size 400 × 15.

When presented with the test dataset the MLP gave responses which were not binary and hence required thresholding with (5.8). The optimal threshold θ was found to be 0.10 resulting in a classification accuracy of 67.11% (77.00% sensitivity, 66.35% specificity). Bayesian regularisation indicated that the MLP was using 448 of its parameters (out of 464 weights and biases). Discussion of these results and their statistical significance in relation to other cases is presented in Section 5.5.

5.4.4 Analysis by Radial Basis Function Network

A Radial Basis Function Network was trained to classify the cardiac and respiratory datasets in an attempt to compare with – and improve upon – the

MLP-based classifier. The RBFN's architectural considerations – discussed in Section 3.3 – are such that an appropriate number of hidden radial basis neurones should be selected prior to training. K-means clustering analysis was performed on the cardiac dataset in order to determine the optimal number of clusters that could partition the data and hence the optimal number of neurones in the RBFN hidden layer. Figure 5.8 shows the relationship between the number of clusters utilised by the K-means algorithm and the resulting sum of distances between data points and cluster centroids.

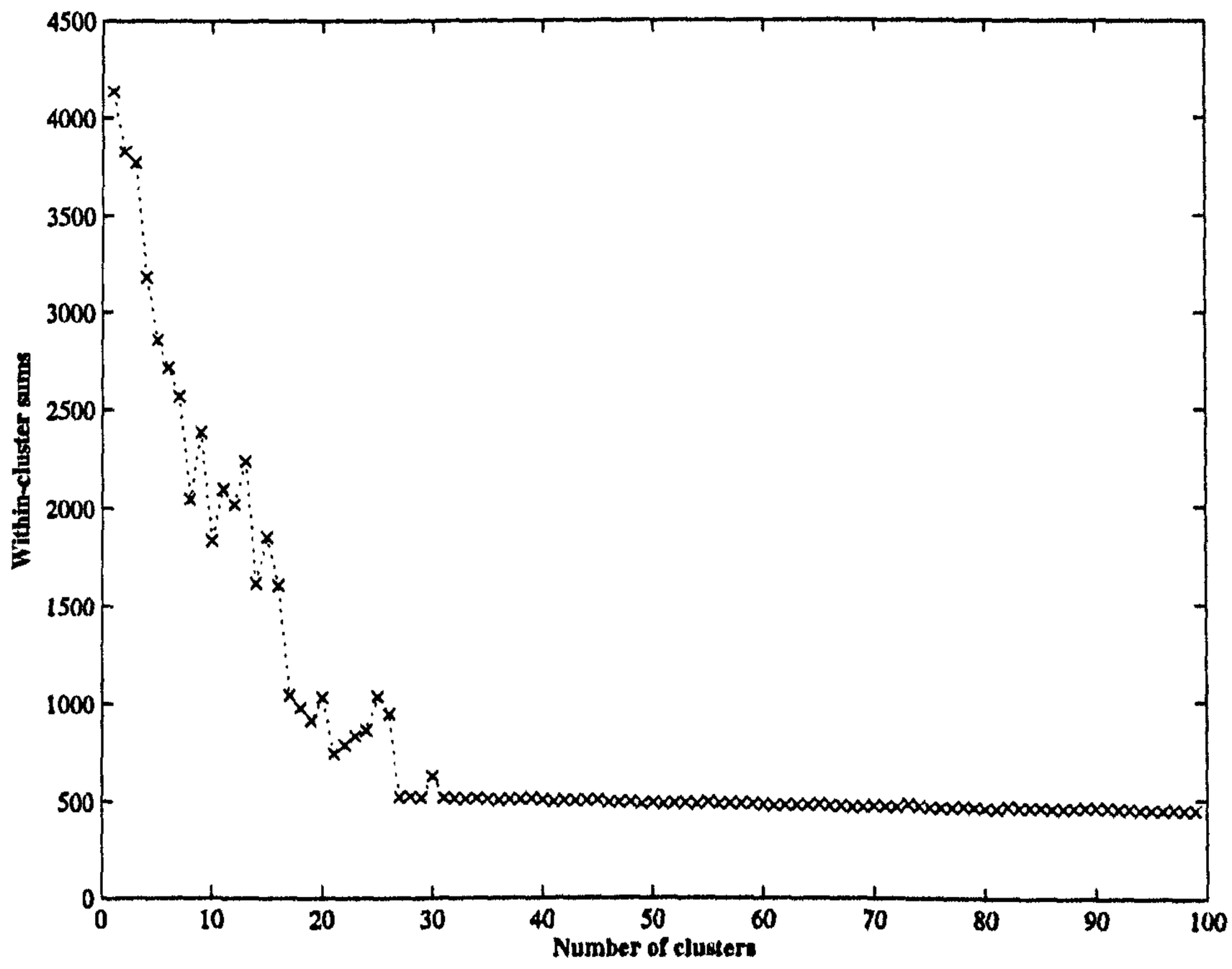


Figure 5.8. The number of clusters used by the K-means algorithm vs. the within-cluster sums.

As can be seen from Figure 5.8, the optimal number of clusters was 27 and further clusters did little to optimise data partitioning.

5.4.5 RBFN analysis of the cardiac dataset

A 15-27-17 RBFN was created and trained on the 1000-sample training dataset with the resultant SSE being 92.30. Given the 700-sample novel test dataset, the RBFN attained a classification accuracy of 99.88% (sensitivity 99.06%, specificity 99.93%). The optimal threshold cut-off θ yielding these results was found to be at 0.45. A discussion of these results and how they compare with other ANNs' is given in Section 5.5. Attention is now given to the application of the RBFN in discriminating between the different respiratory sounds.

5.4.6 RBFN analysis of the respiratory dataset

Training and subsequent testing of the RBFN to discriminate between the different respiratory sounds followed a similar procedure to that discussed in Section 5.4.5. The K-means clustering technique identified that 24 clusters provided the optimal coverage of feature space and hence the optimal number of RBFN hidden layer neurones. A 15-24-14 RBFN was created and trained on 1000 samples of the available respiratory dataset. The ANN was then tested on the remaining 400 samples and attained a classification accuracy of 95.79% (100% sensitivity, 95.46% specificity). The optimal θ was found to be 0.02. These results are discussed in Section 5.5, whilst the following section details the application of the CPNN to the discrimination between different cardio-respiratory abnormalities.

5.4.7 Analysis by Constructive Probabilistic Neural Network

Prior to creating, training and testing a CPNN with the cardiac data the threshold levels θ^+ and θ^- were defined (Section 3.4.1). These two values fundamentally

govern the resulting sensitivity and specificity of the CPNN [13]. Berthold *et al.*, [13] stated that the respective values of 0.4 and 0.2 provided excellent results in practical situations.

5.4.8 CPNN analysis of the cardiac dataset

The CPNN was trained using 1000 of the cardiac data samples, with a fixed input layer size of 15 neurones, and fixed output layer size of 17 neurones. The number of training epochs was set to 5 and the algorithm grew the architecture entirely within the first training pass. Adjustments to the hidden neurones' density functions were made during each training epoch. After training the CPNN hidden layer contained 286 neurones and when tested with the remaining 400 test patterns, attained a classification accuracy of 89.17% (89.17% sensitivity, 99.36% specificity).

5.4.9 CPNN analysis of the respiratory dataset

A CPNN was also created and trained to discriminate between the different abnormalities presented in the respiratory dataset. Similar to the previous CPNN implementation, the classifier was trained for 5 epochs on 1000 samples from the respiratory dataset. The CPNN added all 390 neurones in the first training pass. When tested with the remaining 400 samples the CPNN attained a classification accuracy of 97.80% (97.80% sensitivity, 89.60% specificity).

5.5 Statistical Significances with McNemar's Test

Given the results obtained from the MLP, RBFN and CPNN implementations, this section investigates their statistical significance. A direct comparison

between different ANNs is insufficient to determine whichever architecture is best in terms of classification accuracy and sensitivity. Whilst sensitivity and specificity measures afford the user with an account of the ability of the ANN to detect the presence or absence of events (*e.g.*, cardio-respiratory abnormalities), it does so without reference to a benchmark case. Statistical significance testing offers a means by which to determine whether the two classification accuracies could occur by chance rather than by direct association [14].

McNemar's Test provides a means by which to assess the statistical significance in the difference between two different ANNs' responses to the same stimuli [14]. The following notation is applied: n_{10} is (in the context of this scenario) the number of samples misclassified by *e.g.*, the RBFN but not by the MLP; and n_{01} is the number of samples misclassified by the MLP but not by the RBFN [15]. McNemar's value χ^2 (based on the chi-squared test) can be expressed as:

$$\chi^2 = \frac{\left(|n_{01} - n_{10}| - 1\right)^2}{n_{01} + n_{10}}, \tag{5.9}$$

where the -1 term is Yates' continuity correction which is applied due to the data being discrete whilst the χ^2 is continuous. The Null Hypothesis was defined as being that the error rates between two ANNs were the same. With the level of significance being $P = 0.05$ (or 5%), expressed as $\chi^2_{(1,0.95)}$ or 3.84, for the Null Hypothesis to be true (*i.e.*, the difference between the two ANNs' error *not* to be statistically significant), χ^2 would have to be below 3.84. Conversely for the difference to be considered statistically significant χ^2 would have to be above

this value. For example, when analysing the cardiac dataset the statistical significance χ^2 of the difference between the MLP's and RBFN's error rates was 316.87. This indicated that there was a significant difference between the ANN's abilities to classify the dataset. This was not readily apparent from the classification accuracies attained (99.20% from the MLP and 99.88% from the RBFN). In analysing the respiratory dataset the statistical significance between the error rates of the MLP and RBFN was 16.46 which suggested that there was a difference in the classification abilities. Such significance could arise from one ANN demonstrating a poor ability to classify one class of sounds which the other ANN could, and vice-versa. Tables 5.3 and 5.4 give the statistical significance of the error rates between the MLP, RBFN and CPNN classification mechanisms for both the cardiac and respiratory datasets respectively.

Method (Accuracy)	**MLP (99.20%)**	**RBFN (99.88%)**	**CPNN (89.17%)**
MLP	-	316.87	207.66
RBFN	-	-	113.41
CPNN	-	-	-

Table 5.3. Statistical significance of the error rates between ANNs on the heart sounds dataset.

Method (Accuracy)	**MLP (67.11%)**	**RBFN (95.79%)**	**CPNN (97.80%)**
MLP	-	16.46	14.75
RBFN	-	-	7.17
CPNN	-	-	-

Table 5.4. Statistical significance of the error rates between ANNs on the lung sounds dataset.

Section 5.6 presents a method of combining the results of these different ANN implementations in forming a more accurate and sensitive classification mechanism.

5.6 Combined Classification

This section reports on the efforts to improve upon the MLP, RBFN and CPNN implementations in classifying cardiac and respiratory sounds. In both cases the ANNs occasionally gave conflicting or ambiguous classifications when analysing the test datasets. By combining classifiers to form a single system it becomes possible to improve classification accuracy and sensitivity. We recall from Section 3.5 the Dempster-Shafer Theory (DST) method of aggregating the outputs of multiple ANNs to facilitate this type of improvement.

The results of the MLP, RBFN and CPNN classifiers for both the cardiac and respiratory experiments were linearly normalised such that for any given 1-in-N binary classification, all output neurones summed to unity. The results of the ANNs could then be directly considered to be probabilistic assertions to class assignment and could be employed in the DST framework. For clarity, given a class C_k the opposing assertion $\neg C_k$ was

$$\neg C_k = \sum_{l \neq k} C_l. \tag{5.10}$$

Using Dempster's rule of evidence combination presented in (3.31) the combined classifications for both datasets were established. Table 5.5 and Table 5.6

present the individual ANN classifications and their combined results for both the cardiac and respiratory datasets respectively.

Classifier	Accuracy	Sensitivity	Specificity
MLP	99.20%	88.24%	99.88%
RBFN	99.88%	99.06%	99.93%
CPNN	89.17%	89.17%	99.36%
Combined via DST	*99.77%*	*98.14%*	*99.88%*

Table 5.5. Comparison of accuracy, sensitivity and specificity of the MLP, RBFN, CPNN and DST approaches to the cardiac sounds classification.

Classifier	Accuracy	Sensitivity	Specificity
MLP	67.11%	77.00%	66.35%
RBFN	95.79%	100.0%	95.46%
CPNN	97.80%	97.80%	89.60%
Combined via DST	*100.0%*	*100.0%*	*100.0%*

Table 5.6. Comparison of accuracy, sensitivity and specificity of the MLP, RBFN, CPNN, and DST approaches to the respiratory sounds classification.

Table 5.5 and Table 5.6 demonstrate that given a plurality of ANN classifiers, an aggregated approach can improve classification accuracy. Whilst the accuracy of the combined classification mechanism for analysing cardiac auscultated sounds appeared lower than the RBFN, the difference was negligible (0.11%). This coupled with the desire to increase generalisation capability of the cardiac classifiers motivated the ensemble's inclusion in the overall classification schema discussed in Chapter 7. Table 5.6 shows that the combined classification scheme attained 100% classification of the test dataset, with 100% sensitivity and specificity. This suggests that a DST approach can aggregate multiple classifications and improve upon general classification accuracy. In both cases

of cardiac and respiratory dataset classification, a threshold θ of 0.11 was shown to be the optimal binary 1-in-N encoding parameter.

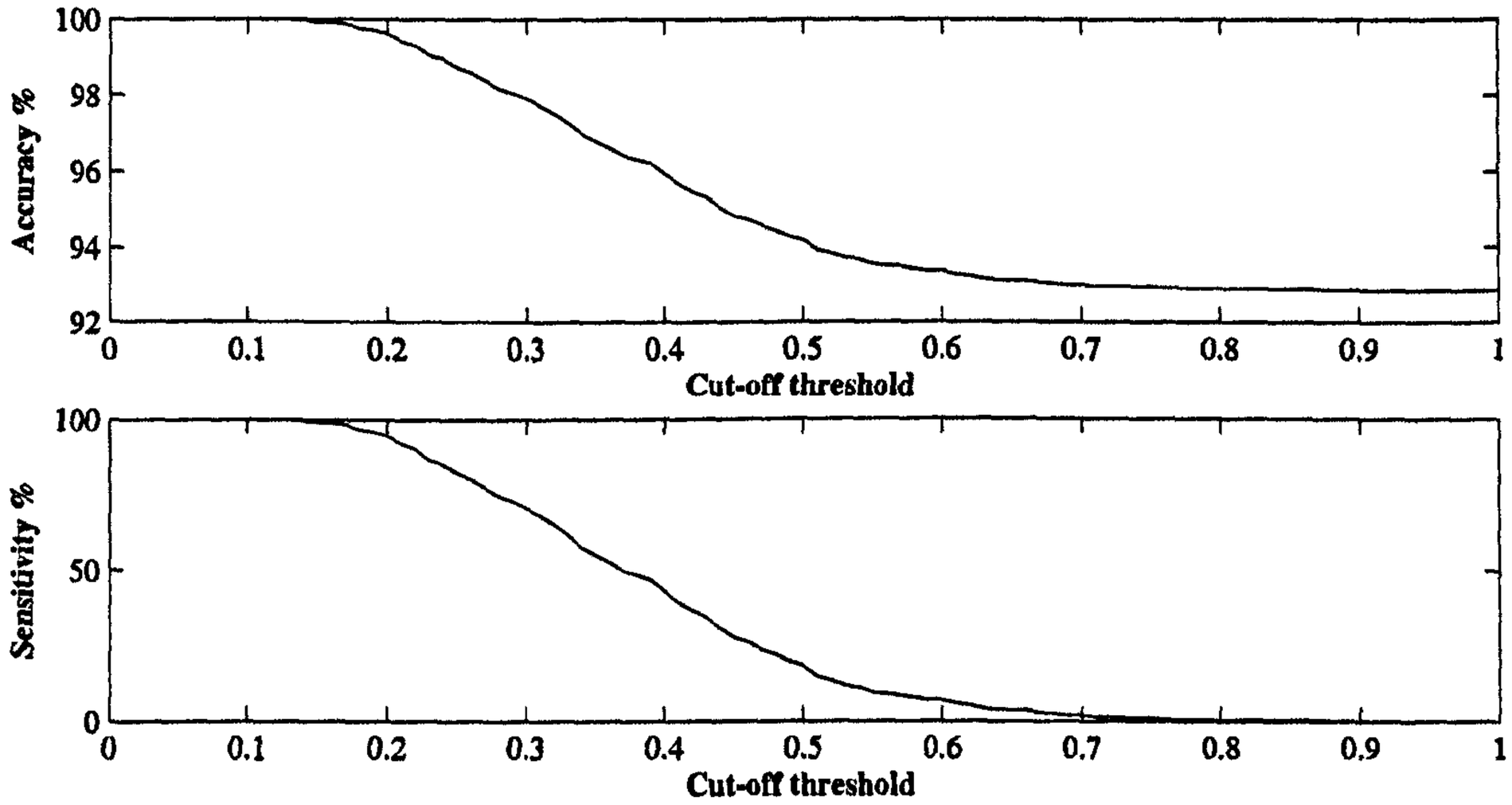


Figure 5.9. Variation in accuracy and sensitivity for the DST-based classifier on the respiratory dataset.

Figure 5.9 gives the relationship between the cut-off threshold and the attained accuracy and sensitivity of the combined respiratory classifiers.

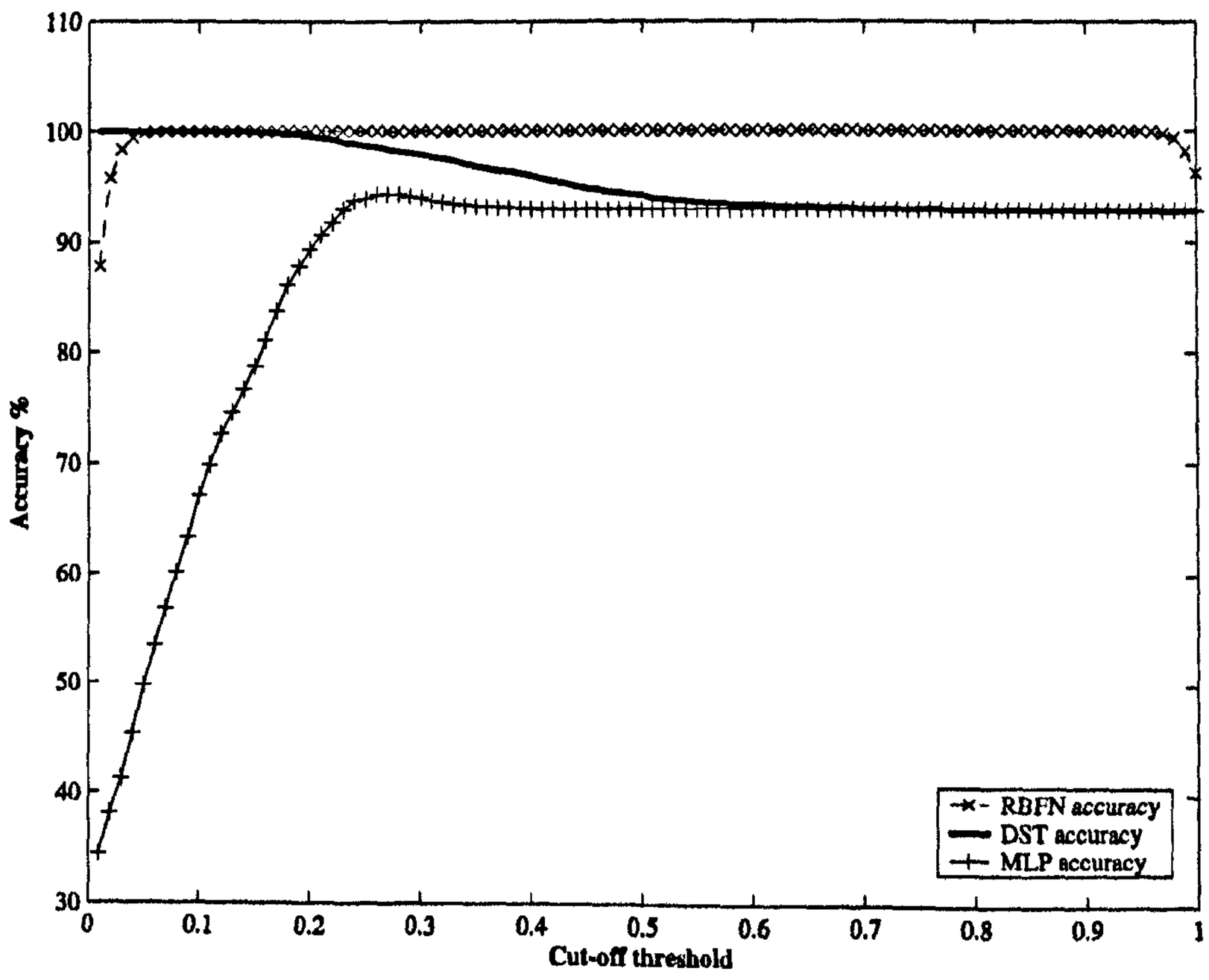


Figure 5.10. Classification accuracies of the RBFN, DST and MLP classifiers on the respiratory dataset.

Figure 5.10 gives the combined classifier's accuracy against those attained from the RBFN and MLP. It is interesting to note that for low (< 0.3) values of θ the DST system behaves like the RBFN classifier in terms of its accuracy, whereas above this θ value it tends towards the relatively poorer abilities of the MLP. At $\theta > 0.3$ the MLP sensitivity approached 2% so a trade-off between classification accuracy and sensitivity was found at $\theta = 0.10$. Given the results in Table 5.5 and Table 5.6, the combined classification mechanism does not inherit the relatively poor sensitivities attained by the weakest classifier in the ensemble.

5.7 Conclusions

This chapter was concerned with classifying different cardio-respiratory pathologies and abnormalities based on the audible sounds heard during chest auscultation. In terms of ANN architecture, Widrow's method of computing MLP topology enabled an efficient and compact architecture with over 50% of the available network parameters being effectively used in classification. The presented solution classifies the sounds using the associated frequency spectra which are unique to each class of sound. The two principal means of discriminating between the different classes of cardio-respiratory sound are the RBFN and CPNN which independently attained classification accuracies in excess of 90%. It has been identified that these methods of classification are sensitive to different classes of cardio-respiratory sound and a means of combining their results has been explored. Evidence combination via DST enabled a higher classification rate for respiratory sounds that would not otherwise have been possible.

In conclusion it can be observed that the problem of discriminating between different classes of cardiac and respiratory abnormalities by assessment of their associated frequency spectra is solvable. In particular, it is suggested that the problem can be solved to a degree of accuracy that would be clinically acceptable by individual ANNs or an ensemble of such techniques. The methods presented in this chapter are further used in Chapter 7 where a generalised classification schema is developed. Prior to this Chapter 6 discusses a further important clinical measurement and the effort taken to form a suitable classification mechanism.

References

[1] Leyva, F., Personal correspondence, 2002.

[2] Bishop, C.M., *Neural networks for pattern recognition*, Oxford University Press, Oxford, 1995.

[3] Hartman, E.J., Keeler, J.J. and Kowalski, J.M., Layered neural networks with Gaussian hidden units as universal approximators, *Neural Computation*, **2**(2), 1990, pp.210-215.

[4] Strum, R.D. and Kirk, D.E., *Contemporary linear systems using MATLAB 4.0*, PWS Publishing Co., London, 1996.

[5] Messer, S.R., Agzarian, J. and Abbott, D., Optimal wavelet denoising for phonocardiograms, *Microelectronics Journal*, **32**(12), 2001, pp.931-941.

[6] Luo, F.-L. and Unbehauen, R., *Applied neural networks for signal processing*, Cambridge University Press, Cambridge, 1997.

[7] Pardey, J., Roberts, S. and Tarassenko, L., A review of parametric modelling techniques for EEG analysis, *Medical Engineering & Physics*, **18**(1), 1996, pp.2-11.

[8] Christini, D., Kulkarni, A., Rao, S., Stutman, E., Bennett, F., Hausdorff, J., Oriol, N. and Lutchen, K., Influence of autoregressive model parameter uncertainty on spectral estimates of heart rate dynamics, *Annals of Biomedical Engineering*, **23**, 1995, pp.127-134.

[9] de Hoon, M.J.L., van der Hagen, T.H.J.J., Schoonewelle, H. and van Dam, H., Why Yule-Walker should not be used for autoregressive modelling, *Annals of Nuclear Energy*, **23**(15), 1996, pp.1219-1228.

[10] Widrow, B. and Lehr, M., 30 years of adaptive neural networks: perceptron, madaline and backpropagation, *Proceedings of the IEEE*, **78**, 1990, pp.1415-1442.

[11] Fahlman, S.E. and Lebiere, C., The cascade-correlation learning architecture, *Advances in Neural Information Processing Systems*, **2**, (Ed. Touretzky, D.S.), 1990, Morgan Kaufmann Publishers Inc., San Francisco, CA, pp.524-532.

[12] Fahlman, S.E., The recurrent cascade-correlation architecture, *Advances in Neural Information Processing Systems*, **3**, (Eds. Lippman, R.P., Moody, J.E. and Touretzky, D.S), 1991, Morgan Kaufmann Publishers Inc., Los Altos, CA, pp.190-196.

[13] Berthold, M.J. and Diamond, J., Constructive training of probabilistic neural networks, *Neurocomputing*, **19**(1-3), 1998, pp.167-183.

[14] Campbell, M.J. and Machin, D., *Medical statistics: a commonsense approach*, 3rd Ed., John Wiley & Sons Ltd., Chichester, 1999.

[15] Roggo, Y., Duponchel, L. and Huvenne, J.-P., Comparison of supervised pattern recognition methods with McNemar's statistical test: application to qualitative analysis of sugar beet by near-infrared spectroscopy, *Analytica Chimica Acta*, **447**, 2003, pp.187-200.

CHAPTER 6

Pulse Oximetry Data Analysis

This chapter presents an analysis of the arterial pulse waveform and arterial oxygen saturation measurements. It describes the steps taken towards an automated means of identifying motion artefacts in the pulse waveform, and hence validating the associated SpO_2 measurements. The chapter also presents the development of a system for classifying arterial oxygen saturation measurements into different patient suitability groups for surgical anaesthesia. It is shown that an MLP can successfully classify 98.9% of the example normal and abnormal arterial pulse waveforms. Furthermore, the final system successfully classified 100% of the example SpO_2 measurements, in complete agreement with the expert anaesthetist. The chapter concludes with a statement of how the final system can be applied to patient diagnosis.

6.1 Introduction

We recall from Section 4.1.2 the importance of oxygen saturation measurements in facilitating an accurate appraisal of cardio-respiratory function. Here we examine the arterial pulse waveform and its effects on pulse oximetry. Also discussed is the manner in which the arterial oxygen saturation measurements were analysed and the information they provide. We demonstrate that the arterial oxygen saturation measurement is too important a criterion to act upon without validating the accompanying pulse waveform.

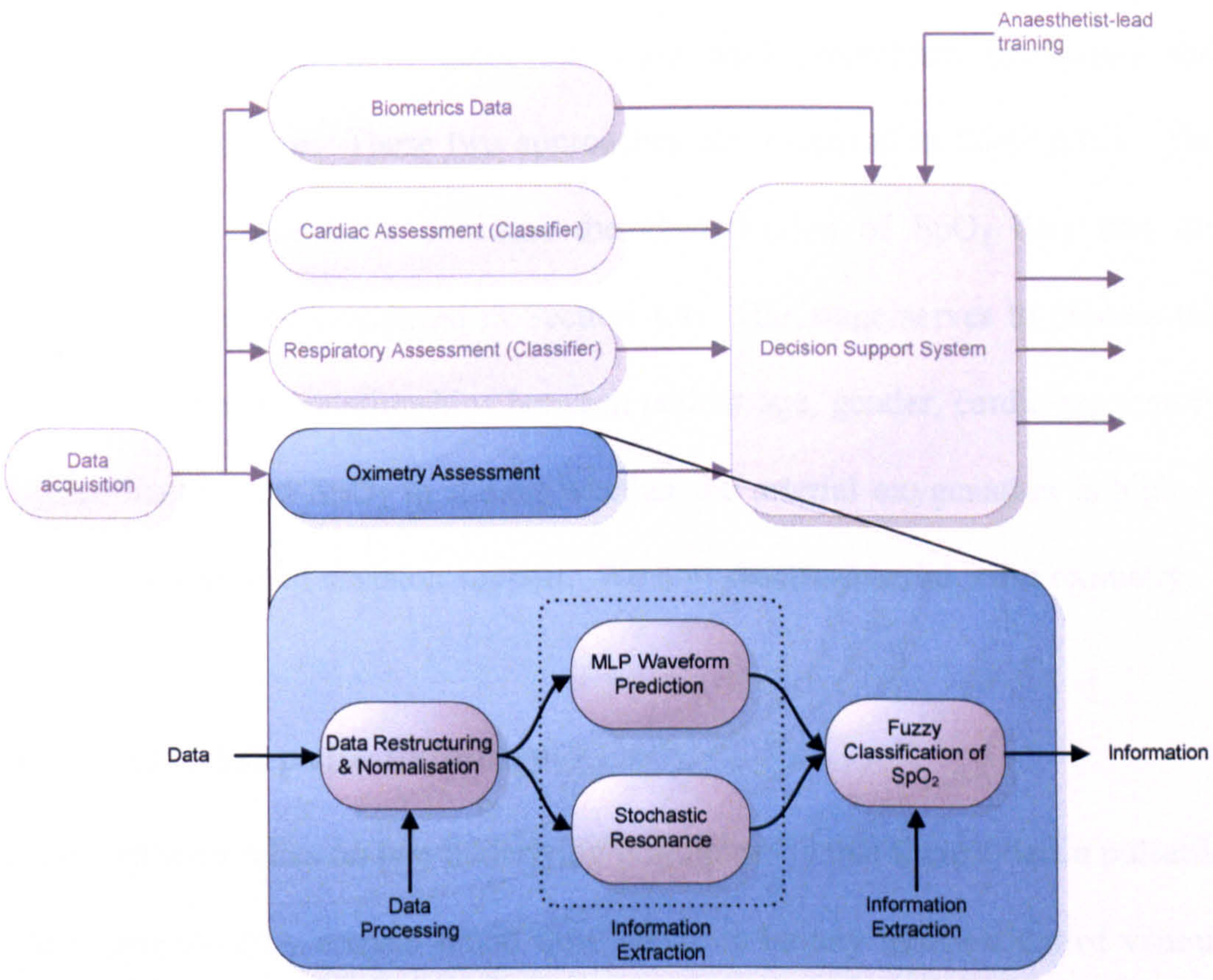


Figure 6.1. Data processing and information extraction performed in this chapter.

In this chapter we present three novel techniques. Firstly we demonstrate that the arterial pulse waveform can be predicted, and hence classified by a MLP. Secondly we apply a statistical method to extract information from normal and

abnormal arterial pulse waveforms in forming a gating mechanism which identifies bad (*i.e.*, distorted, corrupt, *etc.*) waveforms. Thirdly we apply a fuzzy systems technique to classify SpO_2 measurements into the 'anaesthesia suitability groups' presented in Section 1.4. In the following section we introduce the theory underpinning pulse oximetry and demonstrate why this technology is prone to errors which must be addressed.

In Figure 6.1 we show the stages of data processing and subsequent information extraction that is performed in this chapter. We observe the two approaches of corrupt arterial waveform detection being MLP waveform prediction and stochastic resonance. These two approaches are discussed in Section 6.3. The final stage of processing concerns the classification of SpO_2 data into the different categories introduced in Section 1.4. This stage serves to process the SpO_2 and learn the relationships between patient age, gender, cardio-respiratory disease history and SpO_2 in stating whether the arterial oxygenation is high or low in the context of decision support. We now discuss arterial pulse oximetry.

6.2 Arterial pulse oximetry

Pulse oximetry relies on two underlying principles: 1) that there exists a pulsatile signal generated by arterial blood flow which is largely independent of venous blood flow (*i.e.*, blood flow in the veins); and 2) that *oxyhaemoglobin* O_2Hb (oxygen-saturated red blood cells) and *reduced haemoglobin* Hb (desaturated red blood cells) have different light-absorption spectra [1]. We present a definition of the oxyhaemoglobin measurement and how it is calculated, followed by an

account of motion artefacts; how they are introduced into the pulsatile waveform; and how they impact upon the accuracy of oxygenation measurements.

6.2.1 Oxyhaemoglobin measurements

The oxygen-carrying red blood cells are comprised of haemoglobin (Hb) molecules which due to their iron content have an affinity to oxygen. When bound to oxygen, the haemoglobin molecule is referred to as *oxyhaemoglobin*. Conversely when devoid of oxygen molecules, the haemoglobin is referred to as being *reduced*. Oxygen saturation (SaO_2, expressed as a percentage) is derived from the ratio of oxyhaemoglobin to total haemoglobin in the blood stream [1] using the expression:

$$SaO_2 = \frac{O_2Hb}{Hb + O_2Hb} \times 100. \tag{6.1}$$

Due to pulse oximetry being a non-invasive technique, SaO_2 is indirectly estimated and the result is expressed as the SpO_2 measurement. An alternative albeit invasive technique is blood gas analysis which affords a directly measured SaO_2 value. This technique is not frequently employed as it is time consuming, invasive and inappropriate for certain patients, *e.g.*, those with comorbidity from cardiovascular disease [2]. In order to estimate SaO_2 an indirect measure of oxyhaemoglobin and reduced haemoglobin is made. This is achieved by transmitting light waves through the artery and measuring the level of absorption by the haemoglobin. Two wavelengths are used in the measurement: red light (wavelength λ = 660nm) and near-infrared light (λ = 940nm). The former is the peak absorption wavelength of reduced haemoglobin whilst the latter is that of

oxyhaemoglobin. A receiver (typically a photodiode) collects the transmitted light, and the received luminous intensity is directly proportional to the potential difference. The oximeter used in these experiments calculated SpO_2 by monitoring light attenuation at the photodiode as a result of blood flow. This is based on the Beer-Lambert law:

$$V_n = I_0 e^{-\alpha d}, \tag{6.2}$$

where V_n is the received luminous intensity (at the photodiode), I_0 is the transmitted luminous intensity and α is the extinction coefficient of the medium (*i.e.*, tissue, arterial walls, blood, *etc.*) of width d. Given that the artery expands and contracts by Δd during a cardiac cycle, maximal absorption (V_{max}) and minimal absorption (V_{min}) [3] can be derived from (6.2) and expressed as:

$$V_{\text{R-max}} = I_{\text{R0}} e^{-(\alpha d + \alpha \Delta d)},\ V_{\text{R-min}} = I_{\text{R0}} e^{-\alpha d}, \tag{6.3}$$

and:

$$V_{\text{IR-max}} = I_{\text{IR0}} e^{-(\alpha d + \alpha \Delta d)}, V_{\text{IR-min}} = I_{IR0} e^{-\alpha d}, \tag{6.4}$$

where the subscript R denotes red light and subscript IR denotes near-infrared light. SpO_2 can be estimated from the ratio of these values using:

$$\text{SpO}_2 = \frac{\ln\left(V_{\text{R-max}}/V_{\text{R-min}}\right)}{\ln\left(V_{\text{IR-max}}/V_{\text{IR-min}}\right)} \times 100. \tag{6.5}$$

Given the estimated oxygenation measurement in (6.5), *hypoxaemia* is defined as being the condition when the $SpO_2 < 90\%$ [4]. Considering the means by which the oximeter determines the level of oxyhaemoglobin it can be seen that the technology is prone to errors. For example, ambient light can affect the oximeter [5,6,7] as well as physical disturbances of the patient which introduces motion artefacts (MA) into the pulsatile waveform. It follows that SpO_2 measurements cannot not be relied upon unless a valid pulse waveform trace accompanies the reading [8]. This critical fact provides the motivation for developing a means of automatically and reliably assessing the validity of the pulse waveform. In the following section we discuss the impact that MA has on the pulsatile waveform.

6.2.2 The pulse waveform and motion artefacts

A motion artefact is some disturbance of the pulsatile waveform which corrupts the observed signal, *e.g.*, by suppressing or distorting the waveform. Whilst MA can in many circumstances be attributed exclusively to physical disturbances (*e.g.*, patient movement) these artefacts can also be exaggerated by sensor misalignments. Hamber *et al.*, [4] found in their study that the placement of the sensor impacts the time taken for the pulse oximeter to respond to changes in oxygen saturation. In their study of 13 healthy patients (aged 18 to 44 years old) they examined the responsiveness of a pulse oximeter (Nellcor N-200, Nellcor Inc., Pleasanton, CA, USA) to hypoxaemia when attached to the patient's toe, ear and finger. They found that there were statistically significant differences between the simultaneous ear and hand monitoring (6 seconds mean lag, *i.e.*, 6 seconds between the onset of hypoxaemia and the oximeter reporting the episode), hand and foot monitoring (57 seconds mean lag), and ear and foot

monitoring (63 seconds mean lag). Whilst it may appear clear from the above account that optimal placement of the pulse oximeter sensor is on the patient's hand (finger) or ear to rapidly detect hypoxaemic episodes, these sites frequently generate false-positive alarms. In their observational study Tsien *et al.*, [9] monitored a paediatric bedside pulse oximeter for 298 hours and found that the device generated 2,942 alarms during this period. They found that 86% of these alarms were false-positive and that 18% were the result of patient intervention, *e.g.*, motion artefacts and sensor misalignments. Furthermore Poets *et al.*, [10] reported a case of the oximeter sensor becoming completely detached from an infant patient without triggering an alarm. These instances highlight the necessity for continual appraisal of the arterial pulse waveform as a means for identifying clinical and non-clinical events.

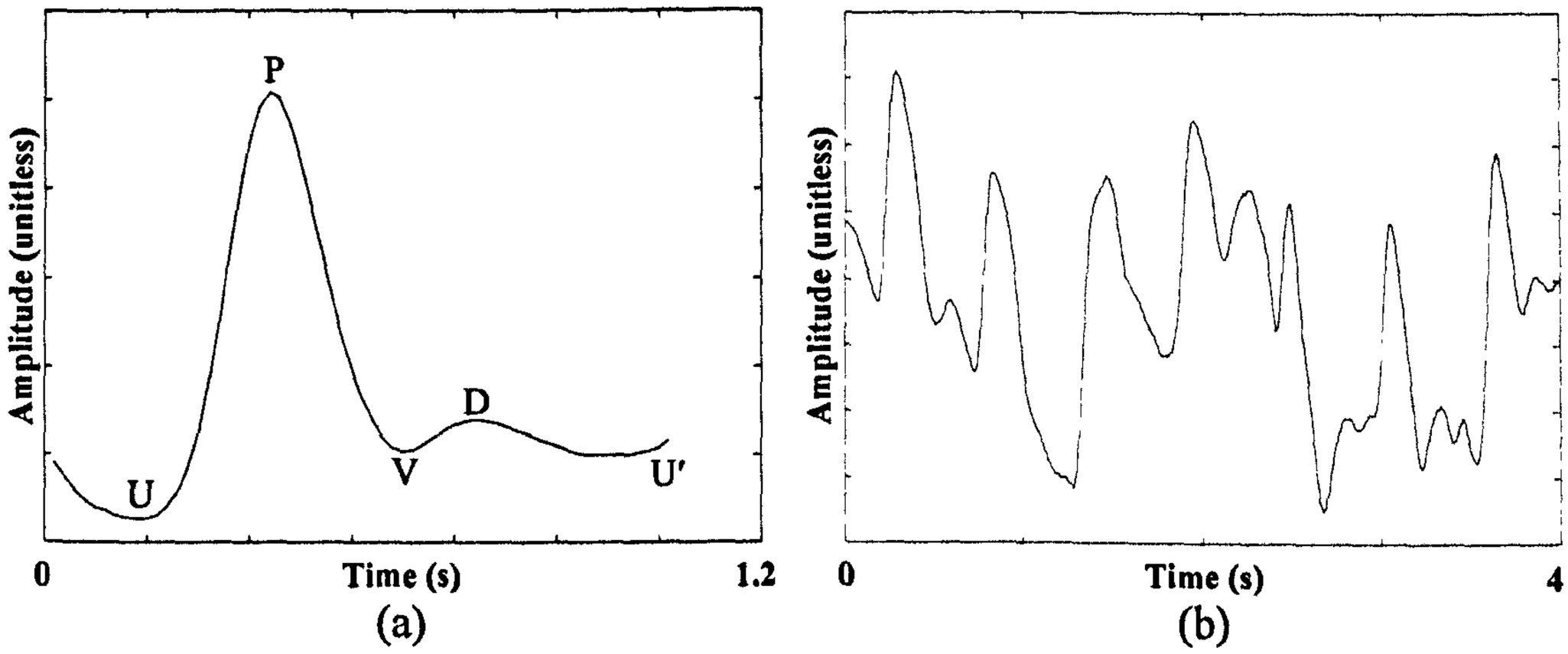


Figure 6.2. (a) A normal pulse waveform for one cardiac cycle; and (b) a MA-corrupted waveform.

Considered now is the structure of the arterial pulse waveform given a normal cardiac cycle. Figure 6.2(a) shows a normal cycle whilst Figure 6.2(b) gives an example of a waveform corrupted due to some motion artefact. The waveform in Figure 6.2(a) comprises the P (percussion) wave, D (dichotic) wave, U (up

stroke) point, and V notch. In particular, the region between U and U′ represents one cardiac cycle, whilst the segment between U and P represents the systolic ejection. The segment between V and D represents the diastolic shut whilst the region between P and V represents the ejection slow time [11]. These waveform characteristics correspond to the mechanical operation of the heart. Cardiac pathologies can on occasion alter these characteristics. We now present a means of automatically identifying and flagging MA in the arterial pulse waveform.

6.3 Motion artefact rejection

This section presents the work undertaken to develop a means by which continual appraisal of the arterial pulse waveform could be performed with the aim of automatically identifying and rejecting MA and other non-standard waveform components. The aim of this was to develop a *gating mechanism*, *i.e.*, one which will state whether the SpO_2 is being measured under normal or abnormal waveform conditions.

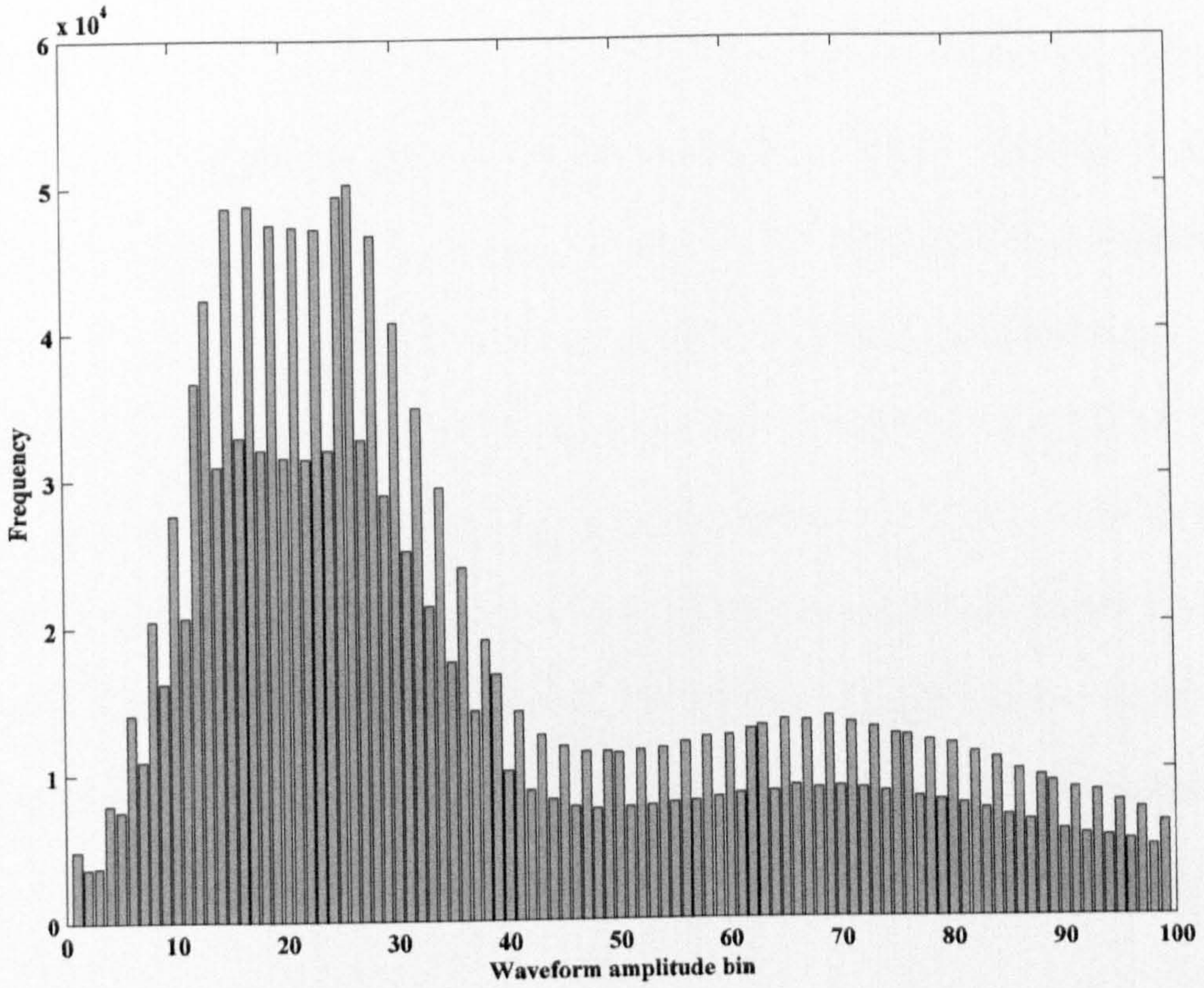
6.3.1 Dataset acquisition

The arterial pulse waveform dataset acquired from the patient sample group (Section 4.1.3) was insufficient for this task. Several hours worth of data was required in order to form the MA-identification mechanism, as will be seen in subsequent sections. We recall from Section 4.3.3 that the pulse oximeter sampled the waveform at a rate of 75Hz. For the purposes of system development, data was acquired independently from the normal data gathering procedure discussed in Section 4.3. Pulse waveforms were collected from 10

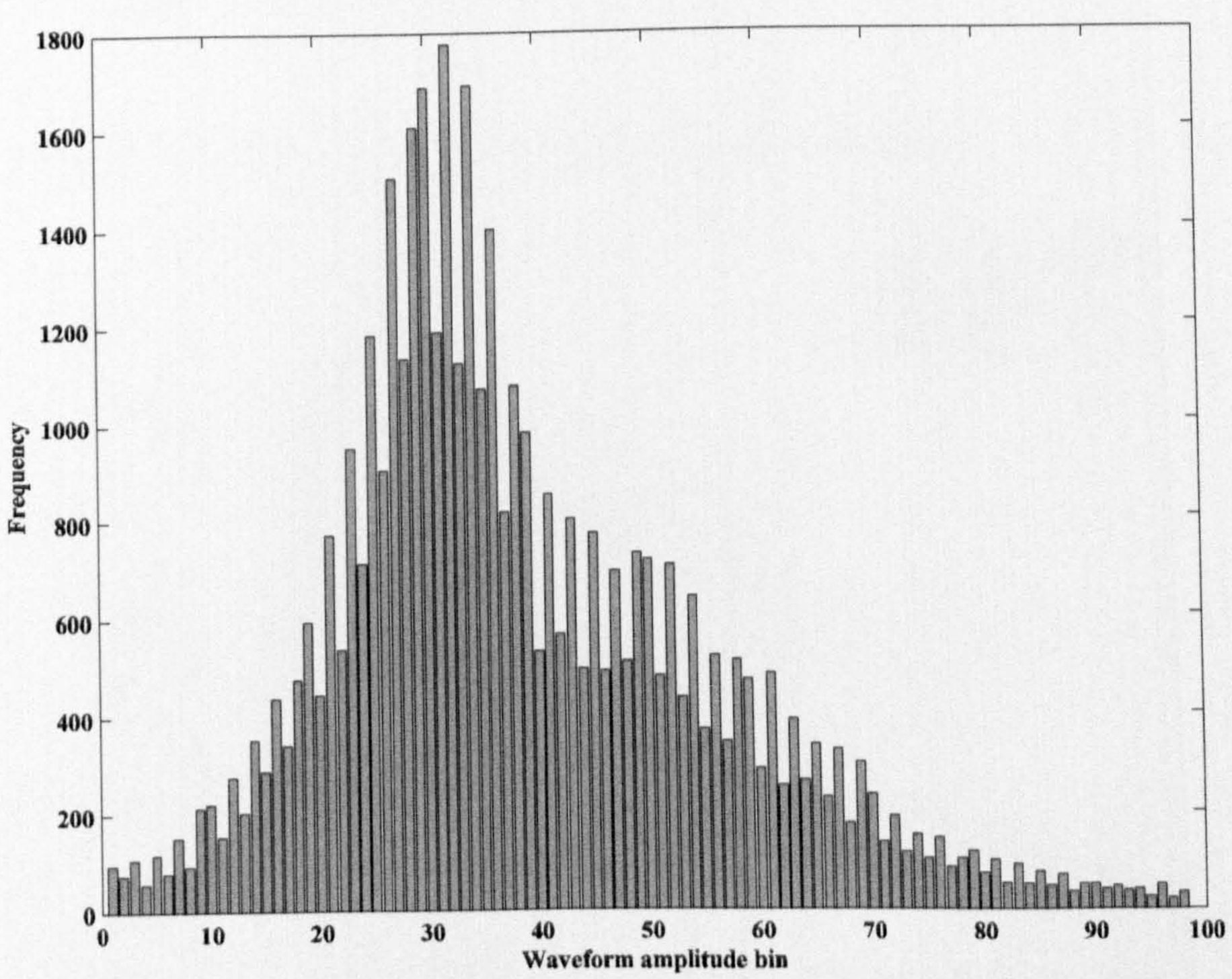
adult patients (mean age 46 years) during overnight studies. There was an equal distribution of male and female patients and all had no history of cardiovascular disease. The period of collection was 7 hours resulting in a waveform dataset of approximately 1.8Gb per patient. After collection, regions of MA and poor perfusion were identified by visual inspection by a physician and extracted. The result was two datasets, one containing samples of normal arterial pulse waveforms (58.36 hours), the other containing examples of MA and poor perfusion (11.64 hours). In order to develop a better understanding of the differences between the datasets, statistical waveform analyses were performed which are presented in the following section.

6.3.2 Statistical models of the pulsatile waveform

As will be shown in the following discussion, MA and episodes of poor perfusion (characterised by and henceforth referred to as *abnormal waveforms*) exhibit different statistical characteristics than normal arterial pulse waveforms. Given these heterogeneous characteristics, a measure of the differentiability between normal and abnormal waveforms could be established. The typical distribution of data points within the *normal waveform* and *abnormal waveform* datasets is shown in Figure 6.3. The data points were segmented into 100-bin histograms cross-sectioning the amplitude of the pulse waveforms. As can be seen from the histograms, the two datasets exhibited different data distributions. In a point of similarity, it can be seen that the majority of the data points occurred at around 20% - 30% of the overall waveform amplitudes.



(a)



(b)

Figure 6.3. (a) Normal, and (b) abnormal, arterial waveform amplitude distributions.

Whilst the *abnormal waveform* data distribution past this peak decayed at a seemingly exponential rate, the *normal waveform* dataset exhibited a large (relatively uniform) density of data towards the peak of the waveform amplitude. This is suggestive of the waveform frequently reaching its amplitude peak, and although the histogram cannot afford a measure of waveform periodicity, it is symptomatic of what one would expect of a periodic waveform. It can be seen from the two histograms that there are no sufficiently unique distributions within normal and abnormal waveforms that can facilitate discrimination. Figure 6.4 gives a superposition of both distributions alongside fitted Gaussian curves for clarity. An N order Gaussian curve fit can be expressed in the following form:

$$\varphi_N(x) = \sum_{n=1}^{N} a_n \exp\left\{-\left(\frac{x-b_n}{c_n}\right)^2\right\}, \tag{6.6}$$

where a_n, b_n and c_n are the coefficients of the nth Gaussian.

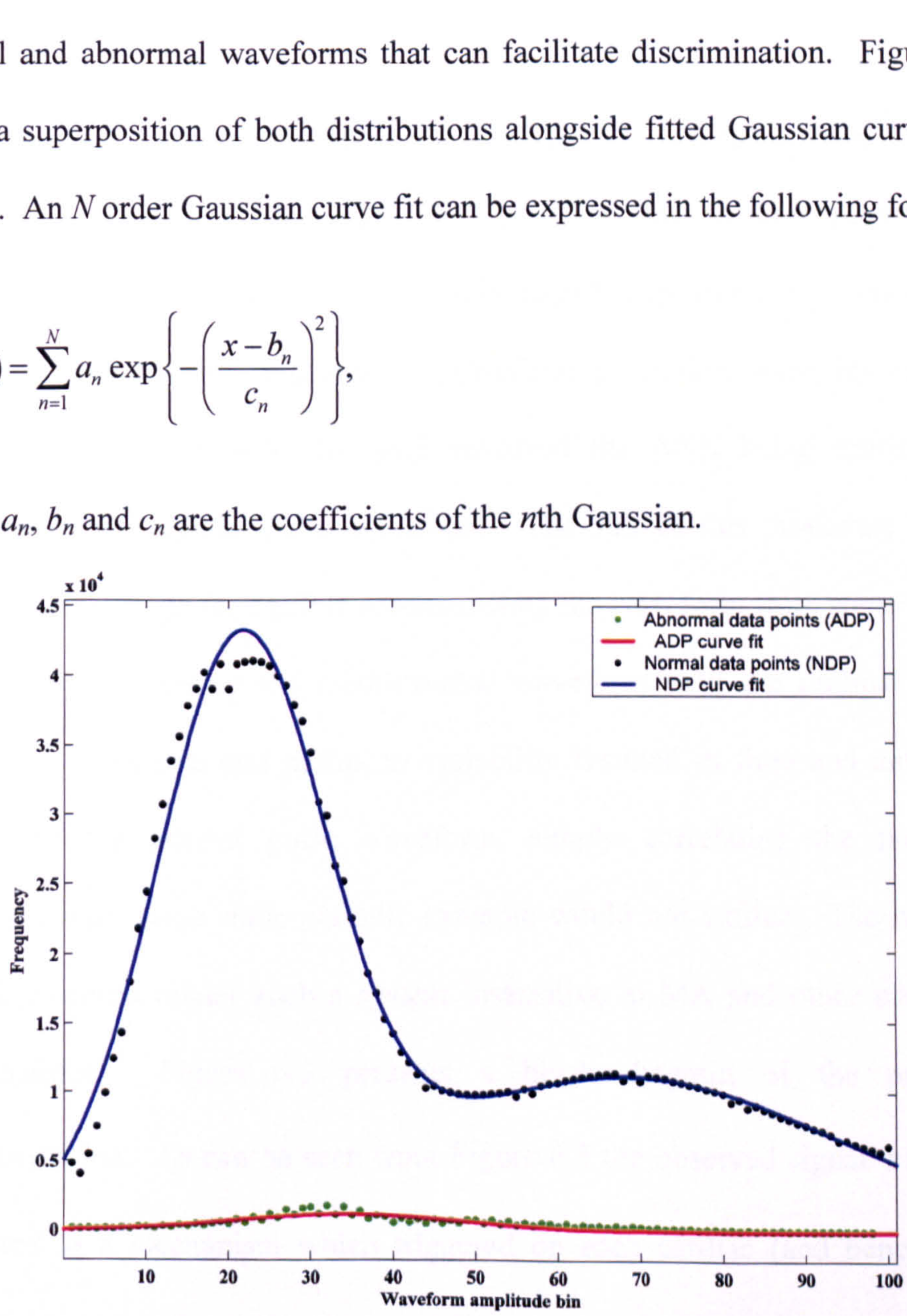


Figure 6.4. Superimposed normal and abnormal waveform amplitude distributions with Gaussian fitted curves.

Two curve fits were established for the normal and abnormal datasets, and are given in Figure 6.4. From a visual inspection of Figure 6.3, a second order Gaussian fit was created for the normal dataset (SSE: 1.37×10^{8}) and a first order Gaussian fit for the abnormal dataset (SSE: 3.81×10^{6}). What is evident from Figure 6.4 is that there were no defining characteristics of abnormal waveforms from the distribution curves, which altogether mandated a different method of differentiating between normal and abnormal instances.

6.3.3 Pulse waveform prediction with the Multi-Layer Perceptron

Given that any means of non-parametric (*e.g.*, ANN) classification of normal and abnormal arterial pulse waveforms would be impeded by seemingly homogenous waveform distributions, methods of waveform prediction were investigated. Waveform prediction with the MLP involved the ANN being trained with examples of the normal pulse waveform. The aim of this procedure was to subsequently engage the MLP in reconstructing the waveform from some trigger, followed by correlating the reconstructed waveform with the original signal. Given that heart rate and perfusion variability resulted in time and amplitude scaling of the arterial pulse waveform, simply correlating the measured waveform with some static periodic example would not suffice. The resultant high SSE would render such a system insensitive to MA and other waveform abnormalities. Figure 6.5 presents a block diagram of the proposed implementation. As can be seen from Figure 6.5 the observed signal $x(t)$ was presented to a mechanism which triggered on each cardiac (and hence pulse waveform) cycle. From the trigger information $z(t)$ the MLP constructed the

waveform $y(t)$ which was cross correlated with the observed waveform $x(t)$. The resulting cross correlation measure afforded a metric with which to assess the similarity of the observed waveform to the known standard.

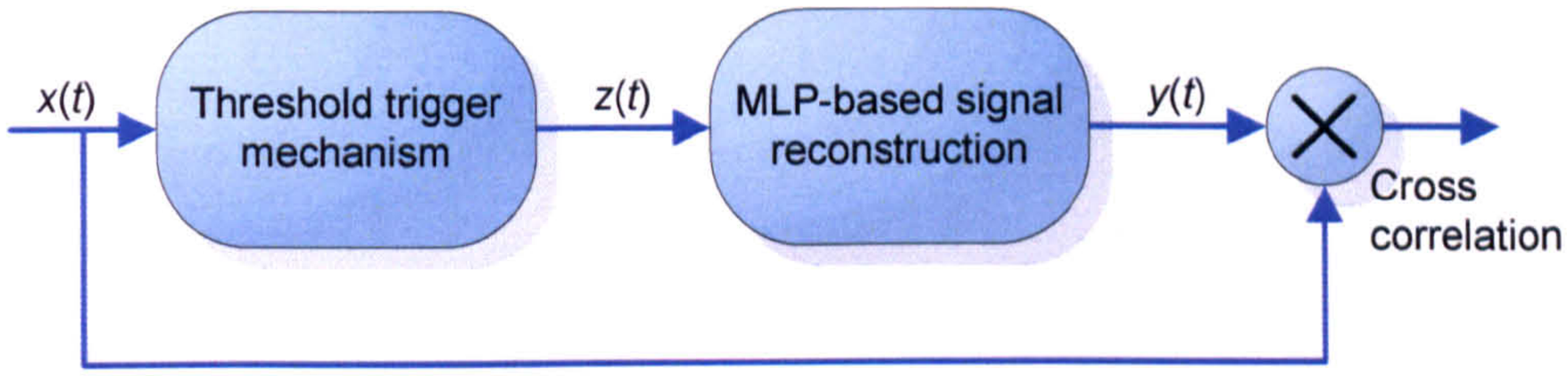


Figure 6.5. Schematic representation of the MLP-based MA rejection mechanism.

The trigger used throughout these experiments was a single-level threshold detector which generated a spike when the waveform $x(t)$ exceeded some threshold with a positive gradient. The result was a *spike train* $z(t) \in \{0,1\}$ where the threshold was the waveform mean over a 1 second window (*i.e.*, 75 samples). Spike trains are of considerable interest to researchers as a means of efficiently coding information [12] since the work of Hodgkin *et al.*, [13] who in 1952 reported the biological process in the giant axon of *Loligo* (squid). The functional form of the threshold detector is $z(t) = f\left(x(t)\right)$ where:

$$f\left(x(t+1)\right) = \begin{cases} 1 & \text{if } x(t+1) \geq m \text{ and } x(t) < m \\ 0 & \text{otherwise} \end{cases}, \tag{6.7}$$

where m is the mean of that 75-sample segment of the waveform. The continuously-valued waveforms from the normal and abnormal datasets were passed through the threshold detector generating associated spike trains.

For the MLP to be successfully trained with the normal arterial pulse waveform data and therefore be able to accurately flag subsequent abnormal instances, a training dataset was constructed from the available data. To successfully predict the pulse waveform, every training pass had to encompass information concerning the current and previous states of the waveform. The motivation for this is derived from the normal waveform being continuous and it is therefore the case that the instantaneous value of the waveform at time t partially depends upon the previous value at time $t-1$.

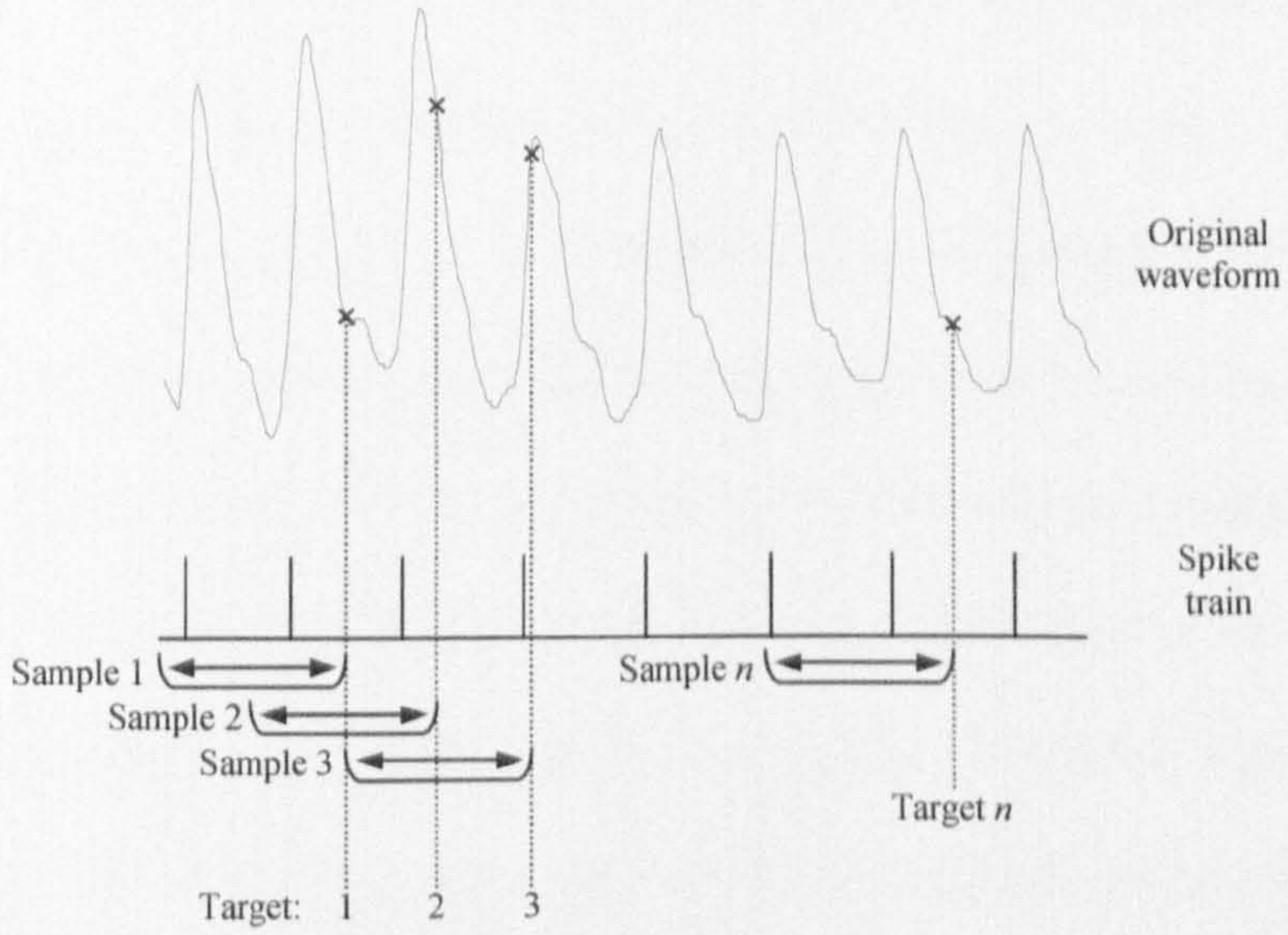


Figure 6.6. Relationship between the continuous arterial pulse waveform, spike train samples and target values.

Each training vector was composed of 100 elements corresponding to samples of the waveform spike train (1.25 seconds given a sampling frequency of 75Hz). It was required that the vector contained at least one spike therefore a window of time length greater than that of a cardiac cycle was used. The target value for this window was the corresponding 100^{th} element of the continuous waveform. To maintain the requirement that past as well as present states of the waveform were presented to the MLP in each sample, the window was time-shifted by 1

sample. This is emphasised in Figure 6.6 which shows the relationship between the training samples and target values derived from the waveform. As can be seen from Figure 6.6, each training pattern provided a record of the past 1.25 seconds of the arterial pulse waveform. The spike train provided no further information than the start of the systolic ejection of the cycle and in some cases the diastolic shut (we recall the potentially positive gradient between points V and D in Figure 6.2(a)).

The training dataset of normal arterial pulse waveforms was created by taking 5000 individual samples (of length 100 elements) from the data – which were generated using the window time-shifting method described above. The individual samples were evenly distributed amongst the 10 patients. An MLP was created possessing 100 neurones in the input layer (corresponding to the 100 elements of the sample), and a single output neurone (corresponding to the continuous waveform). Linear neuronal activation functions were employed throughout. The alternative tangential sigmoid activation functions (discussed in Section 3.1.2) could introduce non-linear biasing effects which precludes their use here. To determine the number of MLP parameters that were effectively used during training, Bayesian Regularisation [14] was used in a precursory test. The first implementation used one neurone in the hidden layer resulting in 103 effective parameters (101 neuronal weights and 2 neuronal biases). The regularisation technique reported that 102 of the parameters were being used during the learning process. This process was conducted again with a 100-2-1 MLP and 101 of the available 205 parameters (202 neuronal weights and 3 biases) were effectively used.

Given that the addition of the second hidden layer neurone had little impact on the computational complexity of the MLP, the 100-2-1 MLP architecture was retained. Furthermore, in the 100-1-1 configuration the MLP became a direct memory device in which the state of the target waveform was derived from the 100 neuronal weights between the input and hidden layers. The addition of a second neurone did not impact upon the number of parameters being used because it was largely redundant; neuronal weights to and from this neurone were not required to represent the waveform. However the diastolic shut time could also trigger a spike, so it was possible that two input neurones could fire at any given time. The initial 100 input-hidden weights were not sufficient for encoding this eventuality and therefore a second neurone was added to accommodate these instances.

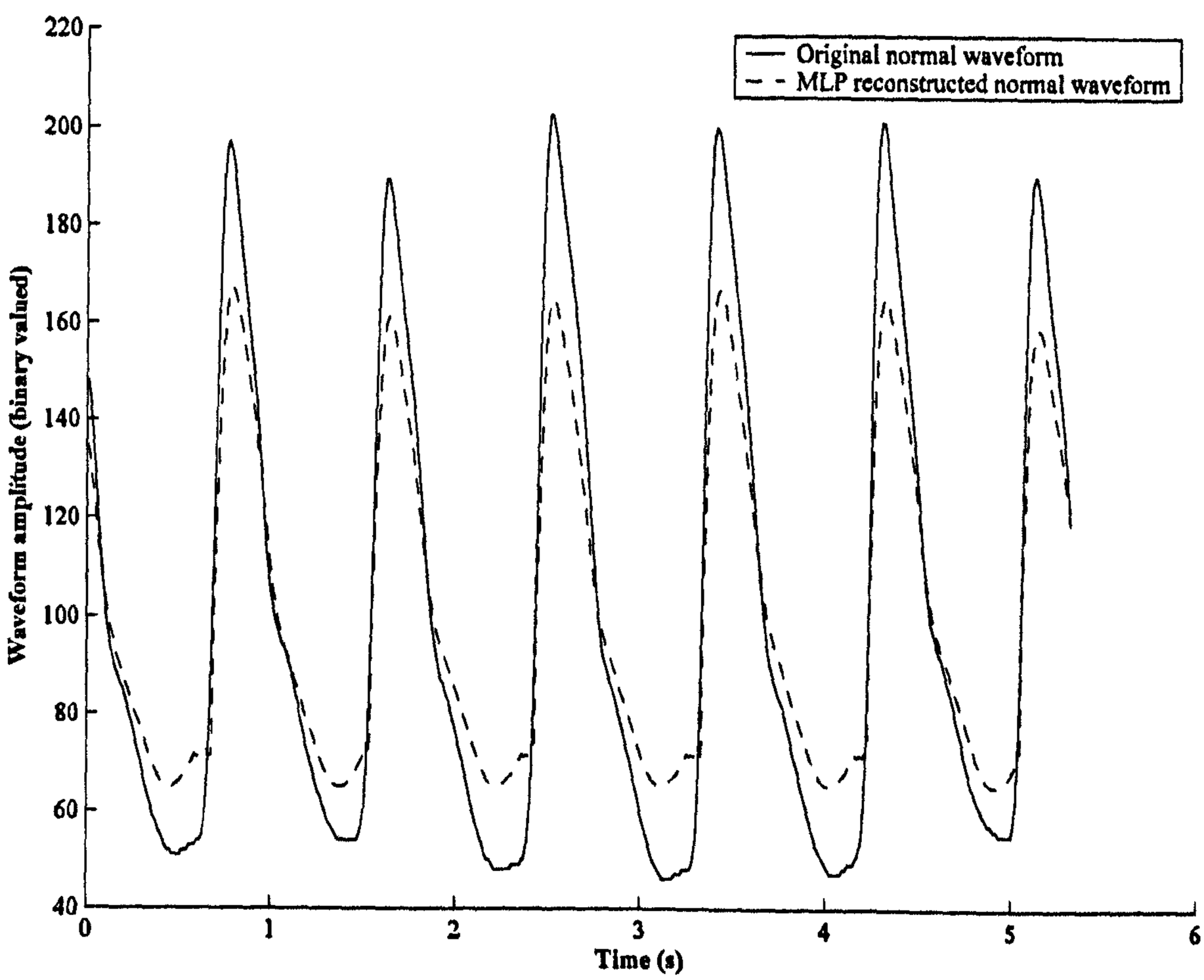


Figure 6.7. Reconstruction of a normal arterial pulse waveform by a spike-triggered 100-2-1 MLP.

During training it was observed that the overall SSE of the MLP stabilised at 200 epochs. The SSE was 4.58×10^5 (MSE: 1.13×10^3) which from observation of the predicted waveform was attributable to waveform amplitude difference and not to periodicity. This is evident from Figure 6.7 which shows a novel (*i.e.*, previously unseen) pulse waveform and its reconstructed counterpart. A measure of the agreement between the original and reconstructed waveforms was evaluated using the cross correlation [15]. For example, cross correlation of the two waveforms presented in Figure 6.7 yielded a coefficient of 0.99. The statistical significance (*i.e.*, the probability that this correlation could occur by chance rather than by direct association) could not be established due to neither waveform exhibiting a normal distribution (Figure 6.3). The MLP was tested with the available dataset of MA-corrupted waveforms and an example is presented in Figure 6.8.

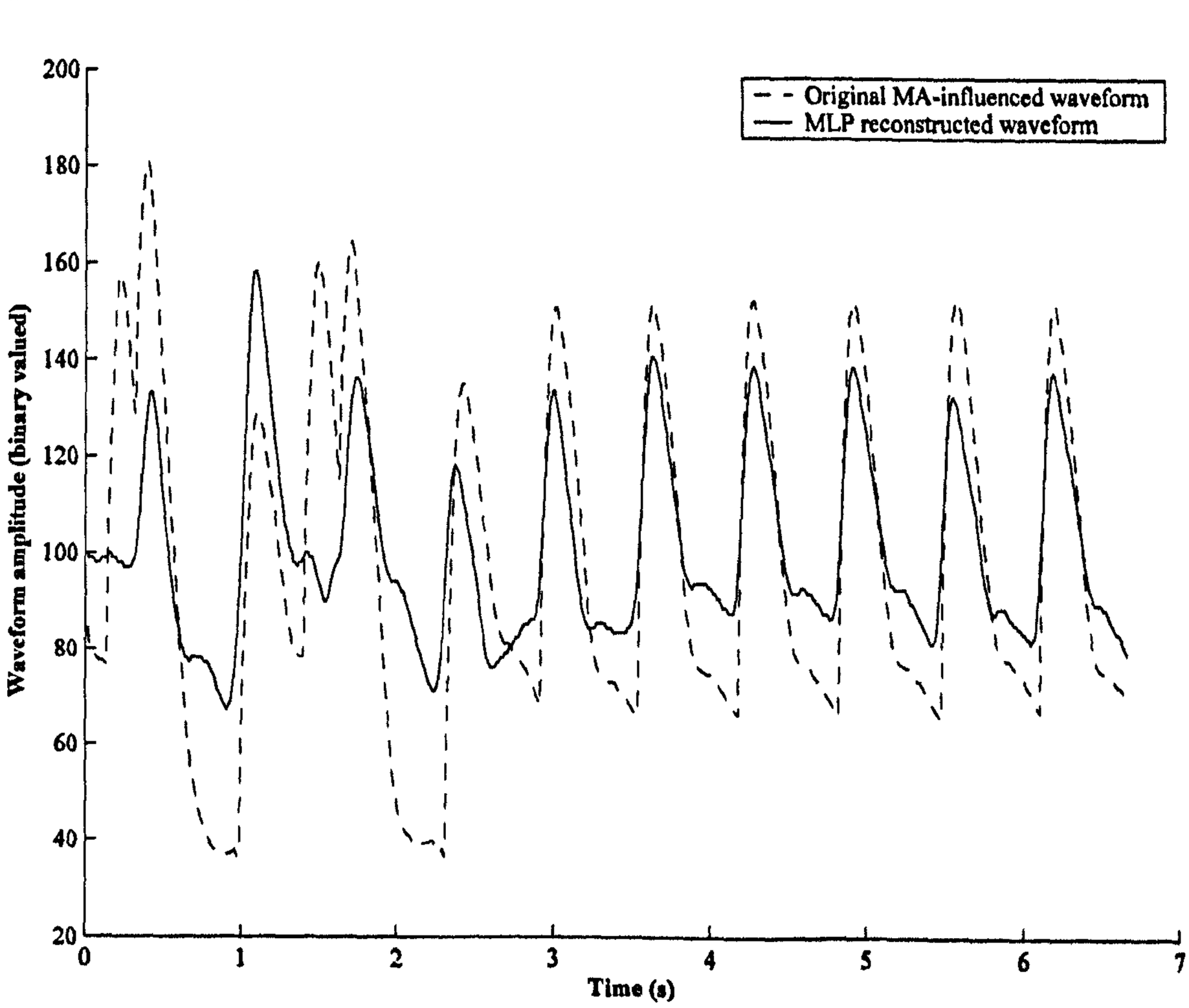


Figure 6.8. Reconstruction of an MA-influenced arterial pulse waveform by a spike-triggered 100-2-1 MLP.

Whilst the episode of MA was relatively short lived (1-2 seconds duration), the effect on the MLP was significant enough to generate a highly-erroneous reconstructed waveform. The correlation between these two waveforms was 0.79. From the available dataset, 1000 examples of MA and 5000 examples of normal waveforms were randomly selected and tested by the MLP. The distribution of cross correlations was examined in order to determine the ability of the MLP to adequately filter normal waveforms from abnormal ones (Figure 6.9).

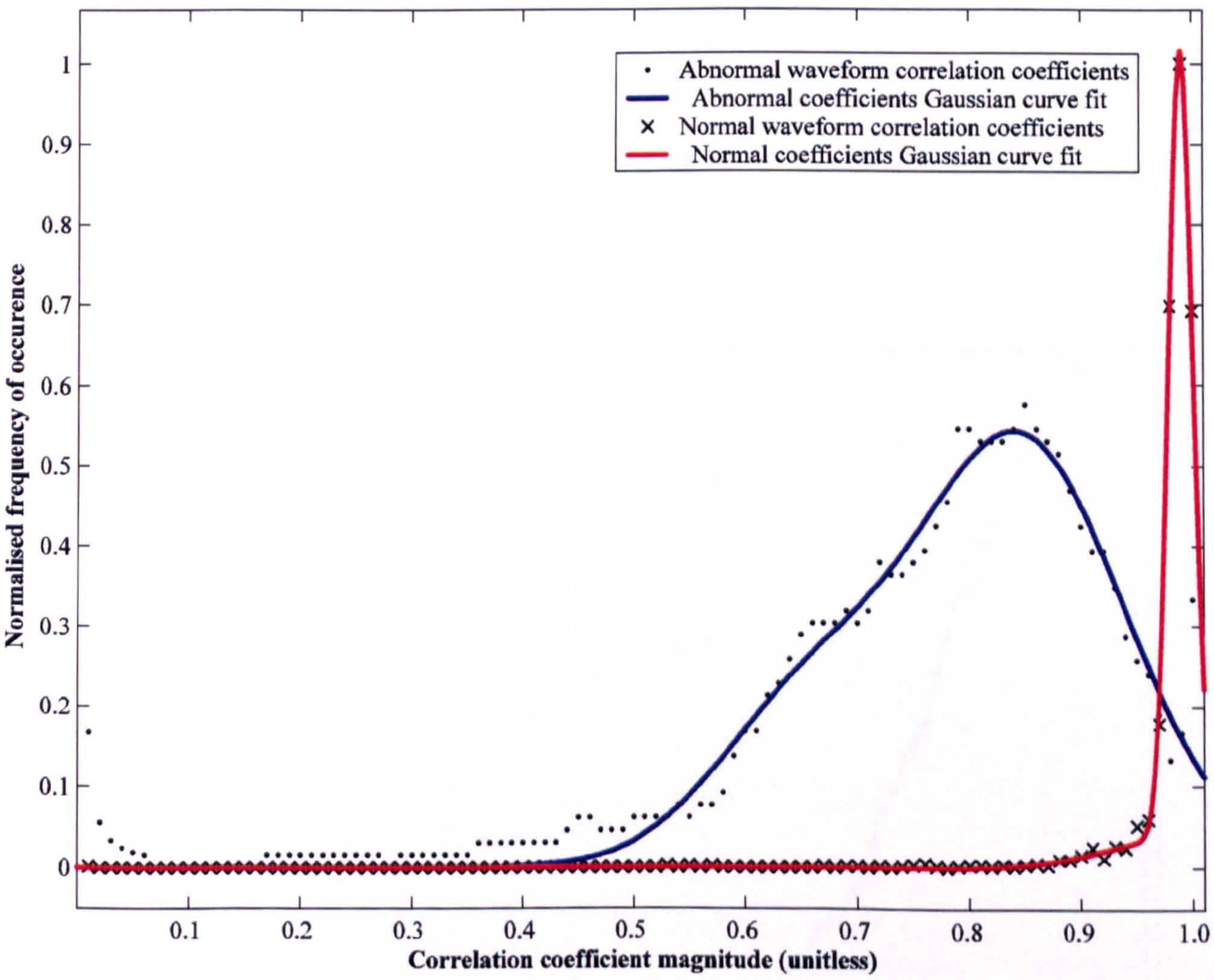


Figure 6.9. Distributions of the correlation coefficients of normal and abnormal arterial pulse waveforms and their MLP-reconstructed counterparts.

As is evident from Figure 6.9, the two distributions exhibited markedly different characteristics in their centre-point and spread. The normal waveform

distribution (fitted with a second order Gaussian, SSE: 2.8×10^{-3}) was mainly contained in the region above the 0.9 correlation coefficient mark whereas the abnormal waveform distribution (fitted with a second order Gaussian, SSE: 1.2×10^{-1}) was mainly spread over the region between 0.5 and 0.95.

Appropriate selection of the correlation coefficient cut-off α was required such that any novel waveform generating a coefficient above α would be deemed normal whilst any coefficient below α would be deemed abnormal. To aid this decision the associated *Relative Operating Characteristic* (ROC) curve [15] was examined. The ROC curve is a plot of the sensitivity vs. 1–specificity of the classifier. The knee-point of the curve affords the best sensitivity trade-off available to the classifier for the given problem.

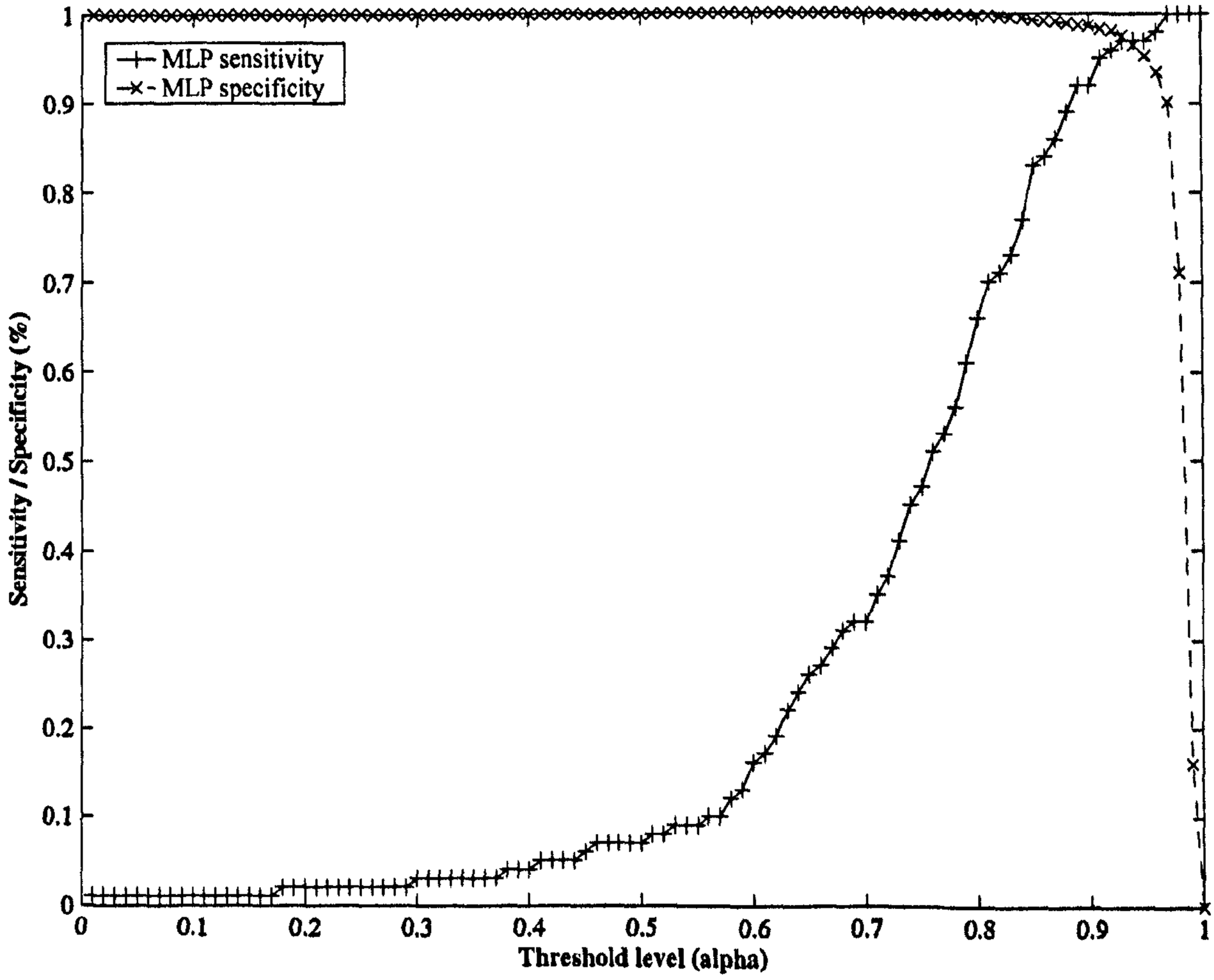


Figure 6.10. Sensitivity and specificity of the MA filtering mechanism.

Examination of the ROC curve indicated that a cross correlation coefficient cut-off of $\alpha = 0.96$ would provide the best available discrimination between normal and abnormal arterial pulse waveforms. Given this cut-off the sensitivity of the classifier became 98.0% with specificity 93.5%. Total predictive accuracy was found to be 98.9%. Figure 6.10 gives the plot of sensitivity and specificity. It can be seen from this figure that the α point is where the two plots intersect, and this is in agreement with a visual inspection of Figure 6.9.

The MLP filtering mechanism presented in this section was capable of discriminating between normal and abnormal arterial pulse waveforms with an accuracy of 98.9%. The MLP required samples of 1.25 seconds duration in assessing whether the waveform was normal or not – a significant improvement on the time scales for existing techniques referred to in Section 6.2.2. These results are discussed in Section 6.3.5 alongside the results of another technique, given in the following section.

6.3.4 Information extraction through stochastic resonance

In the previous section we discussed a method of evaluating the correctness of the arterial pulse waveform by correlating it with a reconstructed version. Whilst the method yielded what appears to be an acceptable degree of predictive accuracy and sensitivity it nonetheless relied upon the MLP being trained with a example waveform from which the correlation was derived. The threshold detector given in (6.7) controlled the amount of information provided by the original waveform during spike train generation. Information concerning the state of the original waveform is not encoded into the spike train if it does not

cross the threshold with a positive gradient. The threshold detector can therefore be viewed as an information transmission channel [16,17].

We briefly discuss pulse waveform additive noise, how this affected the observed waveform information content, and how a classifier could be constructed in the context of the resulting phenomenon of *Stochastic Resonance*. Stochastic Resonance (SR) [18] is a phenomenon in which certain non-linear systems perform optimally with non-zero levels of noise η. The resonance effect is observed when there is a match between the system dynamics and noise statistics, in particular at a certain ratio of signal standard deviation (SD) σ_s and additive noise SD σ_n. A single-level threshold detector similar to that presented in Figure 6.5 was used. The waveform with additive white noise was passed through the threshold detector and the resulting spike train was cross correlated with the original signal. The threshold detector expression in (6.7) was rewritten incorporating the noise term $\eta(t)$ as:

$$f(x(t+1)) = \begin{cases} 1 & \text{if } x(t+1)+\eta(t+1) \geq \sigma_s \text{ and } x(t)+\eta(t) < \sigma_s \\ 0 & \text{otherwise} \end{cases}. \tag{6.8}$$

In each experiment, the normal and abnormal pulse waveform datasets were added with noise and thresholded using (6.8). The experiments were conducted repeatedly with increasing noise:waveform SD ratio. Disparities between the cross correlations of the two datasets were observed. The resonance curves for the abnormal and normal waveform datasets are shown in Figure 6.11. The maximum cross correlation coefficient for the abnormal waveform dataset was 0.29 (SD: 1.0×10^{-2}), and 0.45 (SD: 1.2×10^{-2}) for the normal waveform dataset.

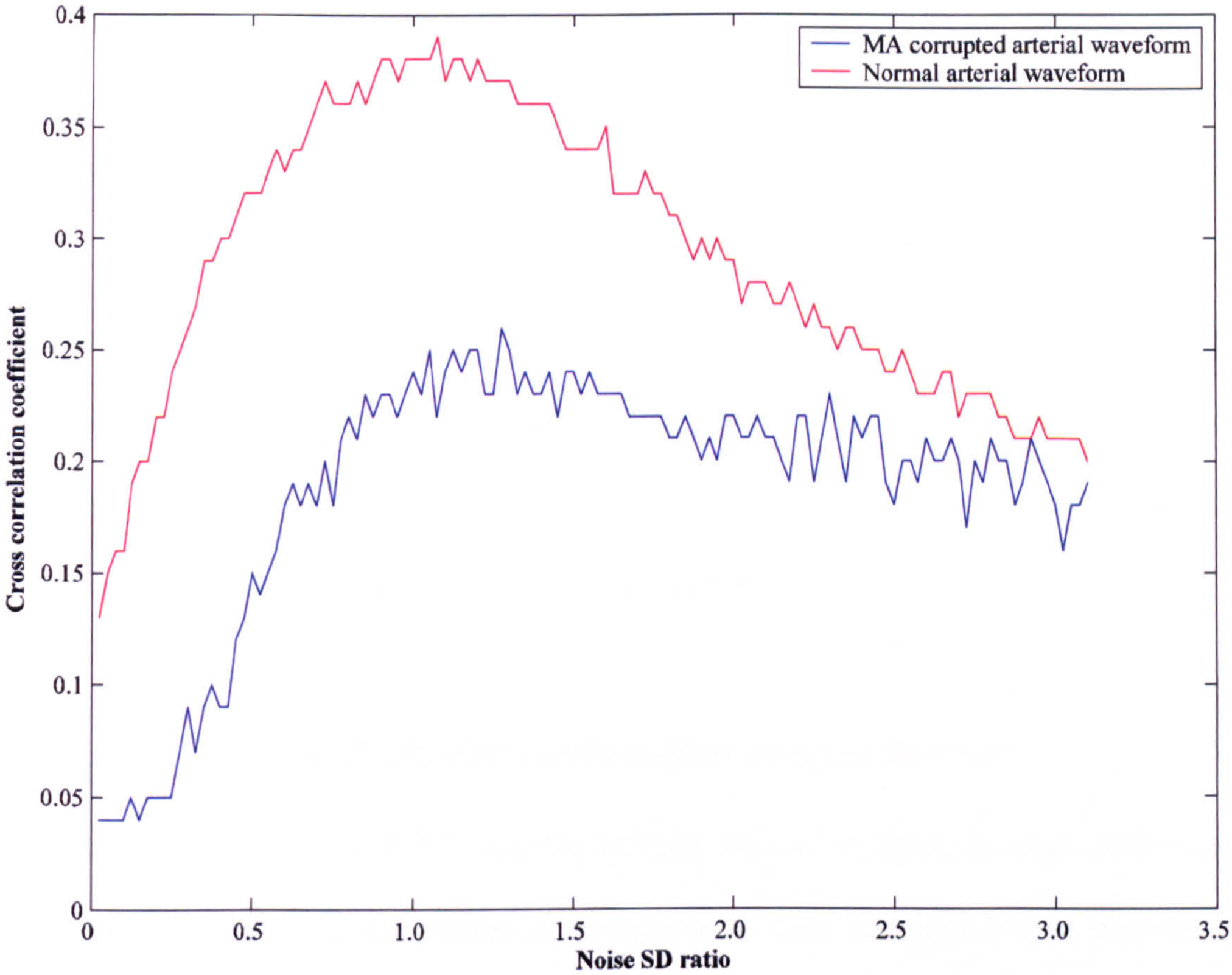


Figure 6.11. Cross correlations for a normal and abnormal arterial pulse waveform at different noise SD ratios (σ_s/σ_n).

Given the heterogeneous nature of the cross correlation coefficients obtained from the normal and abnormal waveform resonance curves, the construction of a classifier became trivial. A simple 1-1-1 MLP was used to map the cross correlations to binary targets ('1' indicating normal waveform and '0' otherwise) which was sufficient to correctly classify 100% of the normal and abnormal dataset patterns.

6.3.5 Discussion

The previous sections have introduced two possible means of identifying MA in arterial pulse waveforms through the continuous appraisal of the waveform structure. The gating mechanism fulfils a requirement that the pulse waveform

be free from non-physiological abnormalities when evaluating the accompanying SpO_2 measurement. Both mechanisms (MLP reconstruction, and waveform resonance) allow a resolution of down to 2 seconds for accurate appraisal of the pulse waveform validity (98.0% and 100% respectively). The underlying statistical heterogeneity between normal and abnormal waveforms that is responsible for these performances has been demonstrated by the different resonance curves in Section 6.3.4. Now we will consider the arterial oxygenation measurements and their interpretation.

6.4 Analysis of arterial oxygenation measurements

As stated in Section 6.2.1, arterial oxygen saturation SpO_2 is expressed as a percentage ratio of oxygenated haemoglobin to total haemoglobin in the blood stream. Measurements below 90% are indicative of cardio-respiratory (or circulatory) pathology, episodic MA, or some combination of these. Whilst Section 6.3 addressed the identification of MA and a means of validating pulse waveforms, this section focuses on the detection of clinical hypoxaemia (insufficient oxygen in the blood) [4,19] through assessment of the SpO_2 measurement.

In order to define a range of acceptable oxygenation values, an experiment was conducted in which SpO_2 measurements were acquired and classified by a consultant anaesthetist. The distributions of acceptable and unacceptably low SpO_2 were then formalised for subsequent use in a classification system. The purpose of this exercise was to couple a classification mechanism with the MA-rejection system described in Section 6.3 to form a complete oxygen saturation

classifier for subsequent use in Chapter 7. We define the system as being complete due to the fact that it encompasses both classification and validation components.

The SpO_2 measurements were acquired from the sample group discussed in Section 4.1.3. Of the 194 patient SpO_2 samples, 150 (mean age 51.78 years) were used of which 52% were from male patients, and 48% from female. Figure 6.12 shows the distribution of SpO_2 measurements.

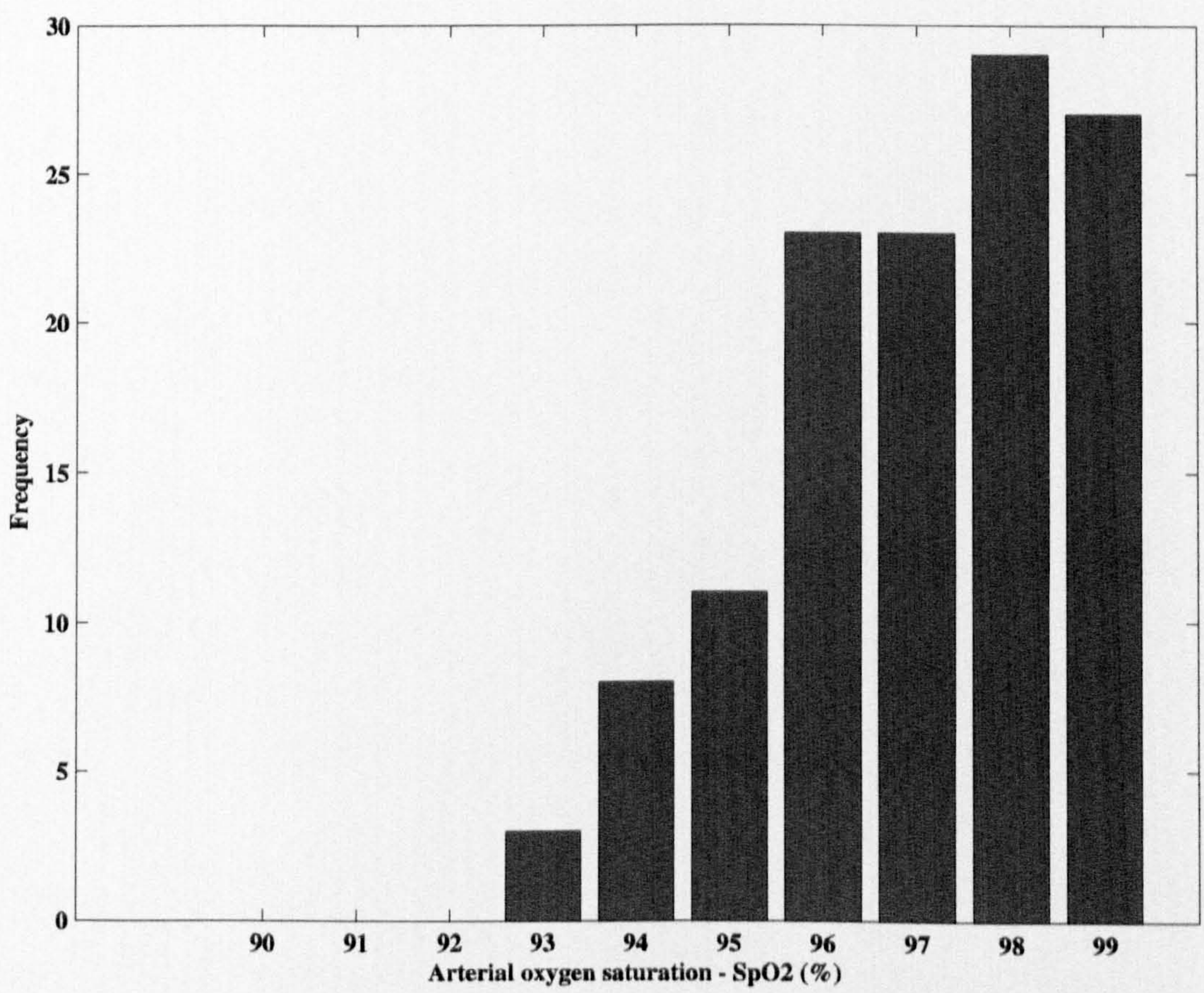


Figure 6.12. Distribution of SpO_2 measurements acquired from patients.

As can be seen from Figure 6.12, the acquired arterial oxygen saturation measurements were consistently over the 90% hypoxaemic threshold, with the

majority of samples being over 95%. When considering a patient's suitability for surgical anaesthesia, to state that a particular SpO_2 measurement is either acceptable or unacceptable neglects the underlying causes that affect the measurement. For example, variables such as age, gender, haemodynamic properties and the presence (or history) of cardio-respiratory disease are contributing factors to a measured SpO_2 value and these must be taken into consideration when directly assessing the arterial oxygen saturation. In determining the prevalence of these variables in an anaesthetist's decision, the acquired data was presented to a hospital anaesthetist for consideration. The aforementioned variables were presented alongside the acquired SpO_2 measurements and the anaesthetist was commissioned to group the patients into the three categories given in Section 1.4: 1) suitable for surgical general anaesthesia; 2) unsuitable for surgical anaesthesia; and 3) possibly suitable for surgical anaesthesia after referral for other tests. It is self evident that an anaesthetist would require more information than the SpO_2 measurement when making this type of decision so an assumption that all other biometrics 'indicated suitable' was applied, *i.e.* a decision was made exclusively on the merit of the SpO_2 measurement.

In order to determine the influence of age, gender and cardio-respiratory history on the decision-making process, the anaesthetist's classifications were grouped according to each variable and a statistical significance test was performed. The null hypothesis stated that the classification was made independent of these variables and was exclusively attributable to the SpO_2 measurement. Paired contingency tables were computed for the influence of gender, age, and cardio-

respiratory history on the anaesthetist's decision and a McNemar's statistical test was performed for each variable. Table 6.1 presents the results of the statistical significance test for each of the aforementioned variables.

Variable	Applied *P* level	Significance
Age	$\chi^2 = 3.84$, $P = 0.05$ (5%)	$P = 0.013$ (*i.e.*, $P << 0.05$)
Gender	$\chi^2 = 3.84$, $P = 0.05$ (5%)	$P = 0.001$ (*i.e.*, $P << 0.05$)
Cardio-respiratory history	$\chi^2 = 3.84$, $P = 0.05$ (5%)	$P = 0.001$ (*i.e.*, $P << 0.05$)

Table 6.1. The statistical significance of the effects of age, gender and cardio-respiratory history upon the patient's suitability for surgical anaesthesia.

Table 6.1 demonstrates that the level of statistical significance $P << 0.05$, suggesting that the null hypothesis could be rejected, *i.e.*, that the age, gender, and cardio-respiratory history variables impact upon the patient's suitability for surgical anaesthesia. These variables were used to create a classification scheme that would assign subsequent patients to the three anaesthesia groups. We required that the classification mechanism be capable of expressing its class assignments in such a way that a human expert examiner could interpret its decision-making strategy. ANNs are considered 'black box' solutions insofar that classifications are derived from transfer functions and not from qualitative analysis of the input variables [20]. Various techniques have been developed to extract information (in the form of rules) from ANNs such as the MLP [21,22,23,24]; but here an alternative approach to classification called the Fuzzy Inference System (FIS) is used. In the following section, we introduce the concept of the FIS and further describe how it was employed in deriving a suitable classification mechanism.

6.4.1 The Fuzzy Inference System

In this section we present a brief account of the FIS and how it was employed in the context of classifying SpO_2 measurements. A more detailed exposition of the fuzzy paradigm is presented in Section 7.4.

The FIS is derived from the calculus of fuzzy set theory introduced by Zadeh in 1965 [25]. We consider that some continuously-valued variable x can be ascribed to a number of linguistic terms *e.g.*, '*very low*', '*low*', '*slightly low*', '*medium*', '*slightly high*', '*high*', and '*very high*' [26]. Given that x is on the universe of discourse Φ (*i.e.*, $x \in \Phi$), x can belong to any of the linguistic terms with a degree of membership μ_n where n is the linguistic term. For example, in the functional form $\mu_n(x)$ describes the degree of membership of x to the nth (*e.g.*, '*medium*') linguistic class, for any value of x on Φ. Fuzzy rules can be derived from these membership functions μ_n.

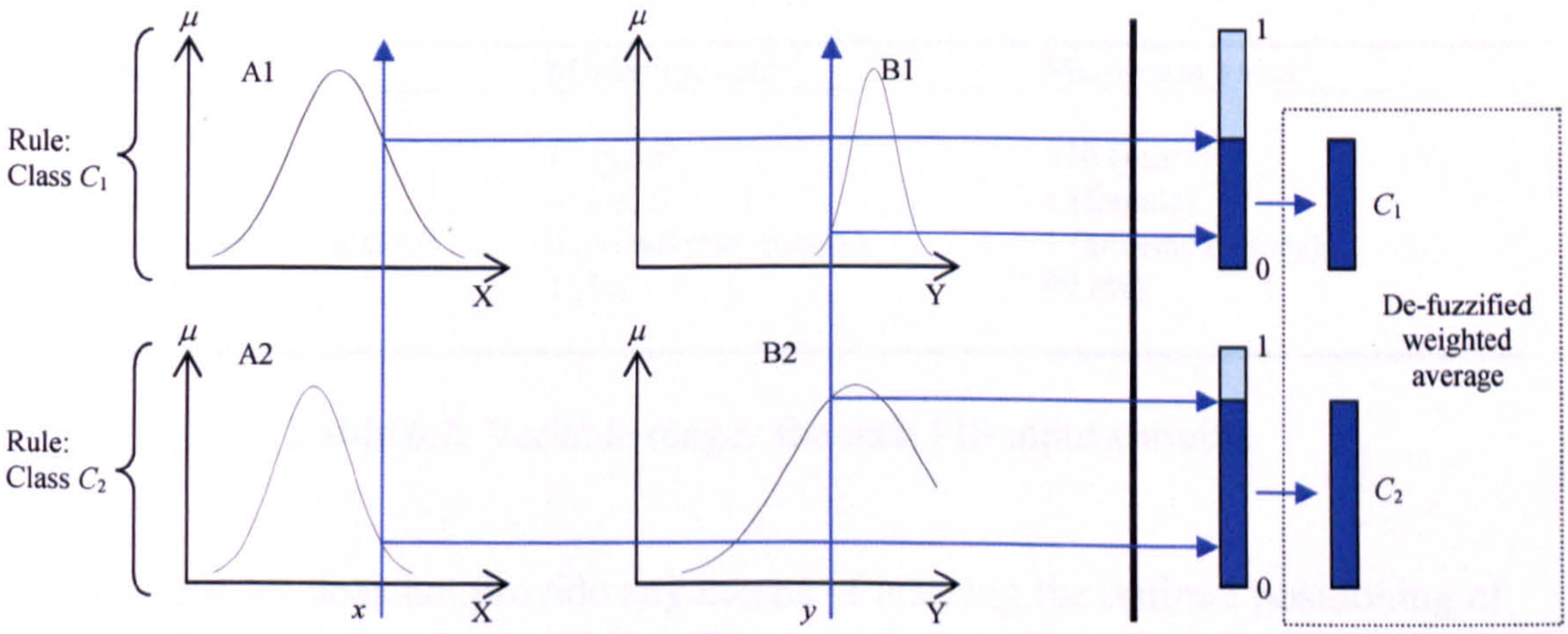


Figure 6.13. An example two-input Sugeno FIS on domains X and Y with two fuzzy rules.

A comprehensive treatment of fuzzy logic and fuzzy rules can be found in texts on soft computing, such as that by Jang *et al.*, [26]. A FIS is an assemblage of

fuzzy rules which ascribes variables to classes (*i.e.*, classify variables) through the logical flow of its rules. A desirable feature of the FIS is that the user may analyse the rule-base to determine how the system proceeds in classifying variables, which is an unintuitive task in ANNs. In the instance of classifying patients into suitability groups based on biometric variables such as age, gender, cardio-respiratory history and arterial oxygen saturation, the FIS becomes an attractive mechanism.

A FIS was created with 4 input variables corresponding to age, gender, cardio-respiratory history and arterial oxygen saturation. The FIS provided 3 output terms corresponding to 'suitable for general anaesthesia', 'possibly suitable for general anaesthesia (refer patient)', and 'unsuitable for general anaesthesia'; binary 1-in-*N* encoded in accordance with Section 1.4. The FIS input variable range is given in Table 6.2.

Variable	Minimum value	Maximum value
Age	18 (year)	110 (years)
Gender	0 (male)	1 (female)
Cardio-respiratory history	0 (no adverse history)	1 (adverse history)
SpO_2 value	1 (%)	99 (%)

Table 6.2. Variable ranges for each FIS input variable.

The FIS itself does not provide any means of learning the optimal positioning of membership functions or fuzzy rules. The FIS was trained using the Adaptive Neuro-Fuzzy Inference System (ANFIS) [20,26] in which the variable membership functions, fuzzy rule-base and interconnecting weights were optimised.

Of the available 150 patient records, 125 were used to train the FIS whilst the remaining 25 were used to test the system with novel patterns. 59 samples of the available 150 were classified by the anaesthetist as 'suitable for anaesthesia', 37 as 'suitable after referral', and the remainder as being 'unsuitable for anaesthesia' (henceforth named *suitable*, *refer*, and *unsuitable* respectively). Test training identified that 30 training epochs allowed the FIS to be trained sufficiently without over fitting occurring. The FIS was trained via the ANFIS method on the available 125-sample dataset and the results obtained from a simulation with the 25-sample dataset. The FIS successfully classified all samples of the test dataset (100% accuracy). The relationships between the fuzzy membership functions for each variable are given in Figure 6.14.

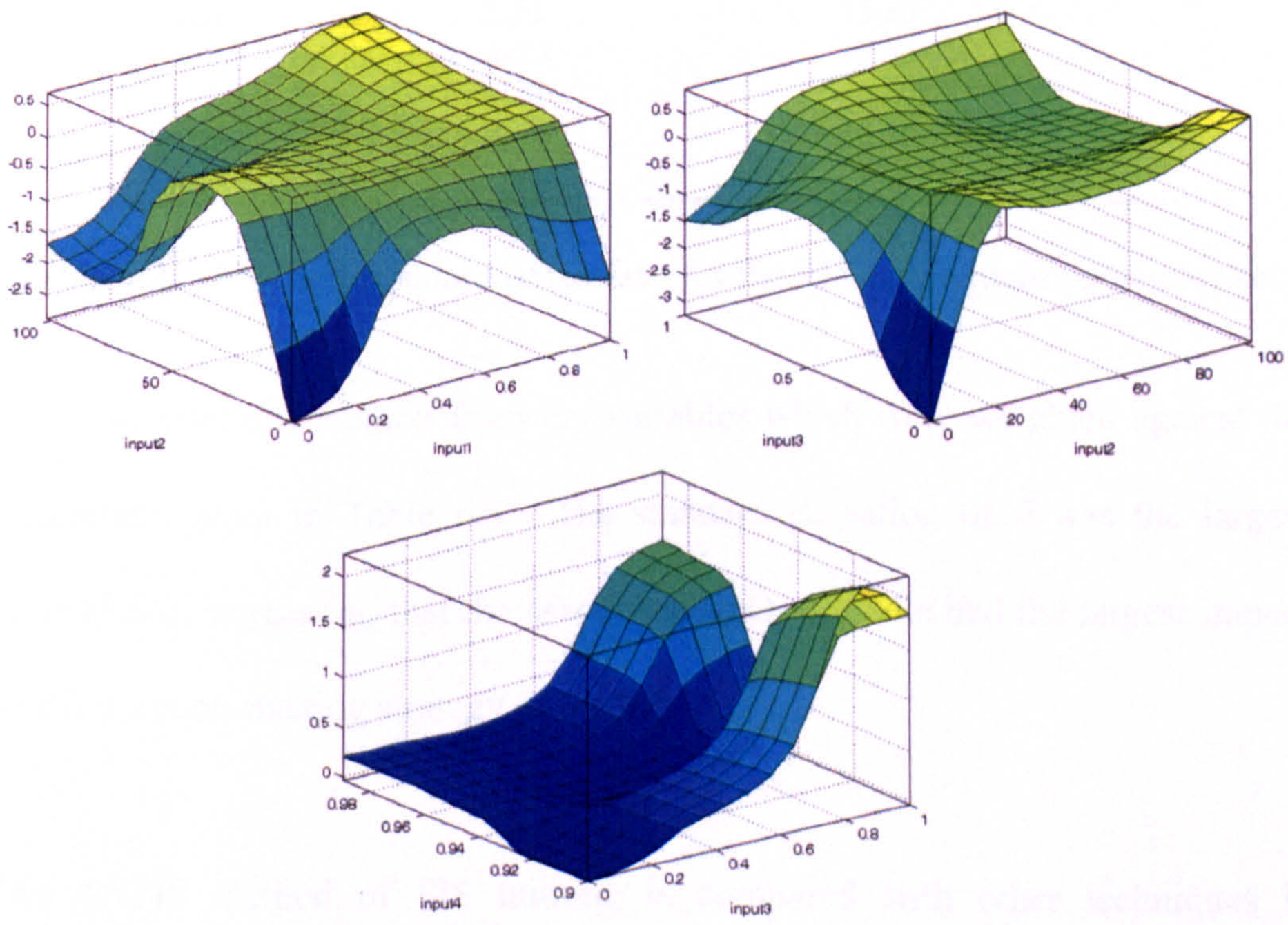


Figure 6.14. Fuzzy relationship between the 4 FIS input variables.

Figure 6.14 demonstrates the relationship between input variable #1 (gender), variable #2 (age), variable #3 (history of adverse cardio-respiratory function),

and variable #4 (SpO_2 measurement). As can be seen from the figure, a patient's history of cardio-respiratory disease presents the largest impact on patient suitability for anaesthesia. The FIS generated by the ANFIS training method contained a total of 81 rules which were used to classify patients into the three groups. These rules are too numerous to present here and are given in Appendix III. ANFIS generated an average of 6.75 rules per variable per class. Each rule in the rule-base has five parameters which define the impact of the four variables described above (the fifth parameter is an offset). Table 6.3 gives the mean and standard deviation of these parameters.

Parameter	Mean	Standard deviation
α (gender)	-1.97	11.02
β (age)	2.30	13.60
χ (cardio-respiratory)	-0.08	13.55
δ (SpO_2)	-7.50	35.64
ε (offset)	-0.46	14.30

Table 6.3. Mean and standard deviations for the FIS consequent parameter-set.

Each rule took information from the variables which were weighted against the parameters given in Table 6.3. The standard deviation of δ was the largest (SD: 35.64), suggesting that the associated SpO_2 variable had the largest impact on the decision-making strategy of the FIS.

The ANFIS method of FIS training is compared with other techniques in Chapter 7 where it is demonstrated that *evolutionary* techniques to FIS training can potentially generate smaller, more compact rule-bases.

6.5 Conclusions

This chapter has presented an analysis of arterial oxygen saturation in the context of validation and classification. In terms of validation, it has been shown that the arterial pulsatile waveform can be corrupted by physiological and extraneous environmental events, and importantly, that these events can be identified in the waveform. Detection of these artefacts allows rapid (< 1.25s) assessments to be performed on the validity of the arterial pulse waveform (98.0-100% accuracy), and an appropriate degree of confidence ascribed to the accompanying SpO_2 measurement. It has further been shown that the oxygenation measurement should be considered in the context of patient cardio-respiratory history when deciding upon their suitability for surgical anaesthesia. Whilst this may be considered intuitive it is highlighted by the fuzzy relationships between the variables: age, gender, cardio-respiratory disease history, and SpO_2.

In the context of Figure 6.1 it has been shown how the arterial pulse waveform data can be used to evaluate the validity of the associated SpO_2 measurement and subsequently how a FIS can extract information from the acquired data. In conclusion, it can be stated that sufficiently rapid (*i.e.*, < 2 s) identification of MA-corrupted arterial pulse waveforms is possible using both ANN-based signal reconstruction methods (98.0% accuracy), and stochastic resonance techniques (100% accuracy). It can be further concluded that SpO_2 measurements can be used in conjunction with rudimentary clinical histories in determining a patient's suitability for surgical anaesthesia. The following chapter extends this work to incorporate the cardio-respiratory data analysis systems (presented in Chapter 5)

with analysis of further biometric signals in the creation of an overall classification system for decision support.

References

[1] Jubran, A., Pulse oximetry, *Critical Care*, **3**(2), 1999, pp.11-17.

[2] National Institute for Clinical Excellence, *Preoperative tests: the use of preoperative tests for elective surgery*, Oaktree Press, London, 2003.

[3] Lee, J., Jung, W., Kang, I., Kim, Y. and Lee, G., Design of a filter to reject motion artifact of pulse oximetry, *Computer Standards & Interfaces*, **26**(3), 2004, pp.241-249.

[4] Hamber, E.A., Bailey, P.L., James, S.W., Wells, D.T., Lu, J.K. and Pace, N.L., Delays in the detection of hypoxemia due to site of pulse oximetry probe placement, *Journal of Clinical Monitoring and Computing*, **11**, 1999, pp.113-118.

[5] Trivedi, N.S., Ghouri, A.F., Shah, N.K., Lai, E. and Barker, S.J., Effects of motion, ambient light, and hypoperfusion on pulse oximeter function, *Journal of Clinical Anaesthesia*, **9**, 1997, pp.179-183.

[6] Amar, D., Neidzwski, J., Wald, A. and Finck, D., Fluorescent light interferes with pulse oximetry, *Journal of Clinical Monitoring*, **5**, 1989, pp.135-136.

[7] Brooks, T.D., Paulus, D.A. and Winkle, W.E., Infrared heat lamps interfere with pulse oximeters, *Anaesthesiology*, **61**, 1984, pp.630.

[8] Fearnly, S.J., Pulse oximetry beyond critical care, *2nd Annual Symposium for Clinical and Biomedical Engineers*, Birmingham, UK, 2003.

[9] Tsien, C.L. and Fackler, J.C., Poor prognosis for existing monitors in the intensive care unit, *Critical Care Medicine*, **25**(4), 1997, pp.614-619.

[10] Poets, C.F., Seidenberg, J. and von der Hardt, H., Failure of pulse oximeter to detect sensor detachment, *The Lancet*, **341**, 1993, pp.244.

[11] Chiu, C.-C., Yeh, S.-J. and Chen, C.-H., Self-organising arterial pressure pulse classification using neural networks: theoretical considerations and clinical applicability, *Computers in Biology and Medicine*, **30**(2), 2000, pp.71-88.

[12] Amigó, J.M., Szczepański, J., Wajnryb, E. and Sanchez-Vives, M.V., On the number of states of the neuronal sources, *Biosystems*, **68**, 2003, pp.57-66.

[13] Hodgkin, A.L. and Huxley, A.F., Currents carried by sodium and potassium ions through the membrane of the giant axon of Loligo, *Journal of Physiology*, **116**, 1952, pp.449-472.

[14] Bishop, C.M., *Neural networks for pattern recognition*, Oxford University Press, Oxford, 1995.

[15] Campbell, M.J. and Machin, D., *Medical statistics: a commonsense approach*, 3rd Ed., John Wiley & Sons Ltd., Chichester, 1999.

[16] Stocks, N.G., Suprathreshold stochastic resonance: an exact result for uniformly distributed signal and noise, *Physics Letters A*, **279**, 2001, pp.308-312.

[17] McDonnell, M.D., Abbott, D. and Pearce, C.E.M., An analysis of noise enhanced information transmission in an array of comparators, *Microelectronics Journal*, **33**, 2002, pp.1079-1089.

[18] Gammaitoni, L., Stochastic resonance in multi-threshold systems, *Physics Letters A*, **208**, 1995, pp.315-322.

[19] Bowton, D.L., Scuderi, P.E. and Haponik, E.F., The incidence and effect on outcome of hypoxemia in hospitalized medical patients, *The American Journal of Medicine*, **97**, 1994, pp.38-46.

[20] Lin, C.-T. and Lee, C.S.G., *Neural fuzzy systems: a neuro-fuzzy synergism to intelligent systems*, Prentice Hall PTR, Upper Saddle River, NJ, 1996.

[21] Krishnan, R., Sivakumar, G. and Bhattacharya, P., A search technique for rule extraction from trained neural networks, *Pattern Recognition Letters*, **20**(3), 1999, pp.273-80.

[22] Setiono, R., Generating concise and accurate classification rules for breast cancer diagnosis, *Artificial Intelligence in Medicine*, **18**(3), 2000, pp.205-19.

[23] Taha, I.A. and Ghosh, J., Symbolic interpretation of artificial neural networks, *IEEE Transactions on Knowledge and Data Engineering*, **11**(3), 1999, pp.448-63.

[24] Towell, G. and Shavlik, J., The extraction of refined rules from knowledge based neural networks, *Machine Learning*, **13**(1), 1993, pp.71–101.

[25] Zadeh, L.A., Fuzzy sets, *Information & Control*, **8**, 1965, pp.338-353.

[26] Jang, J.-S.R, Sun, C.-T. and Mizutani, E., *Neuro-fuzzy and soft computing: a computational approach to learning and machine intelligence*, Prentice Hall Inc., New Jersey, 1997.

CHAPTER 7

Evolutionary Classifiers for Decision Support

This chapter presents the decision support aid which was researched and developed to assist the anaesthetist/nurse in determining a patient's suitability for surgical anaesthesia. Discussed here are the methods by which information is extracted from the classifications performed in Chapters 5 and 6, and fused with other biometric signals in forming the decision support aid. Evolutionary approaches are discussed and applied to fuzzy classifiers and this chapter presents the design of the subsequent hybrid *evolutionary fuzzy* classifier for decision support. The chapter concludes by quantitatively assessing the patient diagnostic capabilities of this integrated decision support tool and demonstrating possible deployment modalities in the context of clinical decision-making.

7.1 Introduction

In forming the Decision Support Aid (DSA) this chapter draws upon the work presented in Chapters 5 and 6 alongside other biometrics discussed in this chapter. As stated in Section 1.4 the purpose of this work was to develop a DSA for the anaesthetist (or less experienced physician). Forming part of this objective, the work presented in Chapters 5 and 6 address issues pertaining to abnormal physiological manifestations in the cardiac, respiratory and circulatory systems. In Section 1.4 it was stated that the decision-making strategy is broken down into, *e.g.*, cardiac assessment, followed by respiratory assessment, *etc.* The systems developed in Chapters 5 and 6 performed these tasks and here they are combined to form the DSA, see Figure 7.1. They are augmented with biometric signals such as blood pressure, heart rate values and lung function responses. In the following sections the DSA is designed to realise the objectives stated in Chapter 1.

7.2 Biometric signals within the decision support aid

In Chapter 1 the biometrics considered in this work were introduced as being cardio-respiratory competence, blood pressure sphygmomanometry, pulse plethysmography, and lung spirometry. Motivation for their inclusion was subsequently given in Section 4.1. Whilst Chapters 5 and 6 presented the work undertaken to develop analysis mechanisms for cardio-respiratory auscultation and pulse oximetry measurements respectively, this chapter discusses the remaining biometrics: blood pressure (incorporating heart rate) and lung function; and the patients' clinical notes which detail, for example the patient's age, gender, history of cardio-respiratory disease, present medication, *etc.*

Considering that the purpose of a learning algorithm is to search input domain space for an optimal solution to the given problem, the domain space must be bounded [1]. Therefore each input variable x must be constrained between a minimum and maximum value (*e.g.*, a and b) *i.e.*, $x \in \mathbb{R} : a \leq x \leq b$. Prior to discussing the development of the DSA, consideration is now given to the form of the inputs and outputs, and any constraints imposed.

7.3 System inputs, outputs and constraints

In forming decisions with which to aid the anaesthetist, the DSA is loaded with information from the preceding-stage classifiers (*i.e.*, cardio-respiratory auscultation and pulse oximetry classifiers) and the available biometric values. Figure 7.1 shows the relationship between the *information sources*, classifiers and the DSA. Given that the work presented in Chapters 5 and 6 addressed the data processing and information extraction aspects of cardio-respiratory pathology identification and pulse oximetry classification, we can now consider these systems to be information sources rather than data sources.

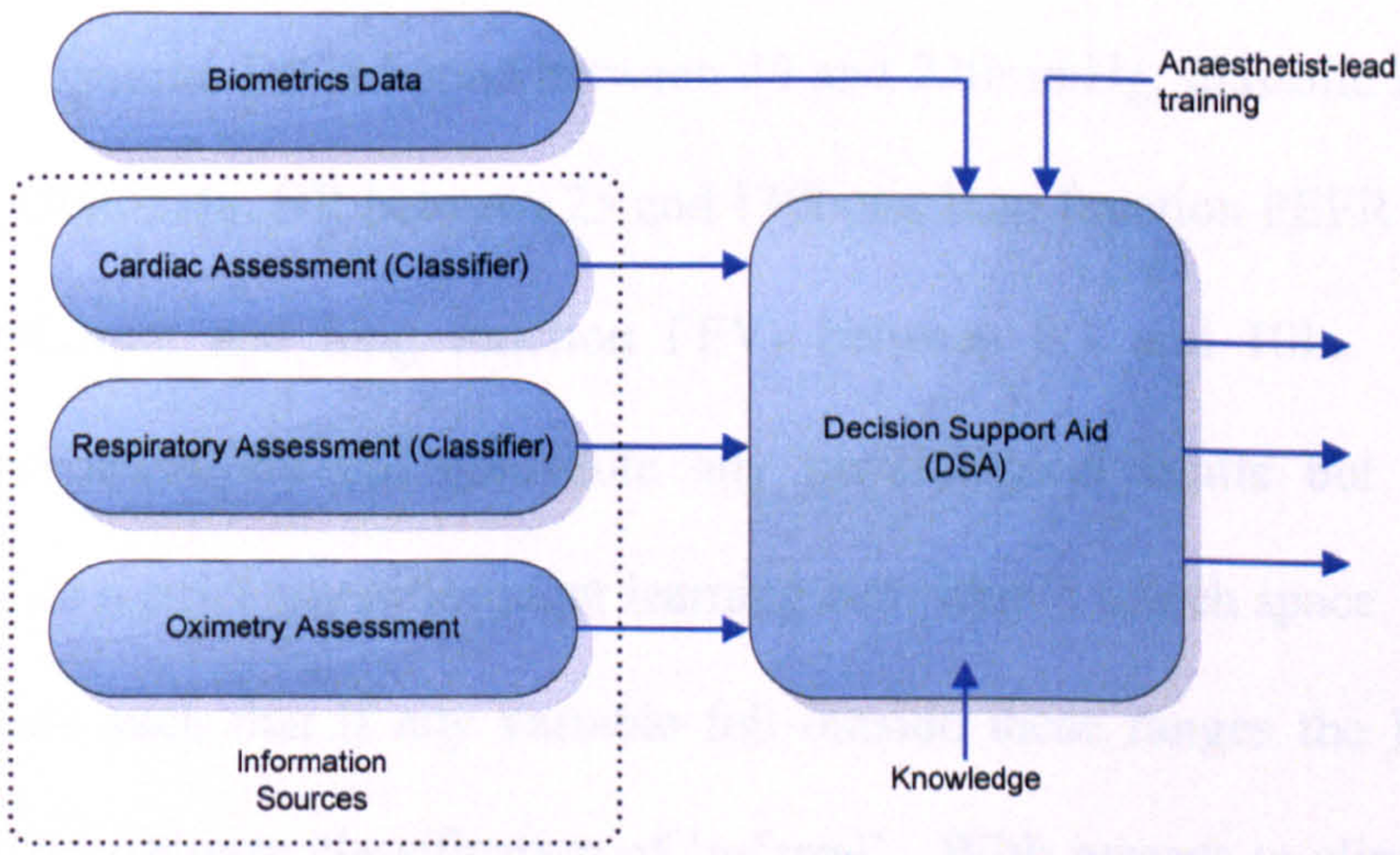


Figure 7.1. Relationship between information sources, classifiers and the DSA.

In Figure 7.1 the DSA is furnished with input from the cardiac and respiratory classifiers. The cardiac classifier provides 17 inputs – one input per pathology (or abnormality). Recalling Chapter 5, the outputs of the cardio-respiratory ANNs were 1-in-*N* encoded and normalised such that the Dempster-Shafer method could be applied to aggregate responses in forming more accurate classifications. The outputs from the DST-based system were similarly normalised thereby representing the classifications as probabilistic assertions as to the cardio-respiratory abnormality. Each DSA input from both the cardiac and respiratory classifiers is bound between 0 and 1 representing classification probabilities. A further DSA input is the patient's arterial oxygenation measured via pulse oximetry. The value presented to the DSA is a percentage and therefore bound between 1 and 100. The value 0 (zero) is retained for validity purposes, *i.e.*, if the DSA encounters a zero from the pulse oximetry classifier, this is due to the oximetry value being rejected by the gating mechanism on the grounds of poor waveform validity. This gating mechanism was the MA-rejection system (MLP) developed in Chapter 6. The final DSA inputs are a plurality of biometric values comprising of BP, HR, lung function and clinical history. Systolic BP is bound between 40 and 220mmHg; diastolic BP between 30 and 150mmHg; HR between 25 and 130bpm; lung function PEFR between 50 and 500L/min; and lung function FEV_1 between 0.1 and 10L. The values presented above do not constitute any physiological limits but are merely imposed to restrict any subsequent learning algorithm's search space. The values are chosen such that if any variable fell outside these ranges the DSA would impose the arbitrary classification of 'referral'. With regards to clinical history,

the patient's age is bound between 18 and 110 years. The remaining variables are binary encoded and are presented in the summary Table 7.1.

Metric	Unit	Minimum value	Maximum value
Cardiac sound classification	N/A	0	1
Respiratory sound classification	N/A	0	1
Blood Pressure (systolic)	mmHg	40	220
Blood Pressure (diastolic)	mmHg	30	150
Heart Rate	bpm	25	130
Peak Expiratory Flow Rate	L/min	50	500
Forced Expiratory Volume (1 second)	L	0.1	10
SpO_2	%	1	100
Patient age	years	18	110
Patient gender	N/A	0 (female)	1 (male)
Medication indicators			
Warfarin, Inhalers, HRT	N/A		
Steroids, Aspirin, Diuretics	N/A	Binary encoded:	
Female Contraceptive Pill	N/A		
Biohazard indicators		1: true / positive presence or history	
Hepatitis B, Hepatitis C	N/A	0: false / negative presence or history	
MRSA, Active TB, HIV	N/A		
Haemodynamic indicators			
Afro-Caribbean origin	N/A		
Excessive alcohol intake	N/A		
Infection	N/A		
Blood disorders / anaemia	N/A	Binary encoded:	
Diabetic	N/A		
Kidney / urinary problems	N/A	1: true / positive presence or history	
Bleeding disorders	N/A	0: false / negative presence or history	
Thyroid disease	N/A		
Malignancy	N/A		
Open wound	N/A		
Cardiovascular indicators			
Smoker (> 40 per day)	N/A		
Cardiovascular disease history	N/A	Binary encoded:	
Varicose veins	N/A	1: true / positive presence or history	
Diabetic	N/A	0: false / negative presence or history	
Respiratory indicators			
Smoker (> 40 per day)	N/A		
Pneumonia / chronic bronchitis	N/A	Binary encoded:	
Asthma (well controlled)	N/A		
Asthma (hospitalised)	N/A	1: true / positive presence or history	
Shortness of breath at rest	N/A	0: false / negative presence or history	
Respiratory disease history	N/A		
Productive cough	N/A		

Table 7.1. Values provided to the DSA during normal operation.

The encoding of the patient's clinical history and medication as presented in Table 7.1 merits particular attention. Relevant patient information can be grouped into five categories: haemodynamic, cardiac (incorporating circulatory), respiratory, biohazard and medication (see Section 4.4). Table 7.1 further expands each category into the relevant conditions which are noted in the patient's history. Each item (*e.g.*, cardiovascular disease history) is a binary-encoded value with 1 and 0 representing the presence or absence of the item respectively. These values are masked such that the category value is set to 1 if any of its 'child' values are 1, *e.g.*, '*Respiratory indicators* IS *smoker* OR *pneumonia* OR...' in Boolean logic form. Given the input variables, attention is now given to the structure of the DSA and how it aids decision support for the physician.

7.4 Fuzzy classification techniques in decision support

The purpose of the DSA was for it to be a classifier that would ascribe different levels of anaesthesia suitability to the patient. As such the DSA could either exist in the form of an ANN-type classifier; or as a fuzzy rule-based classifier (introduced in Section 6.4.1). Whilst ANNs can generalise solutions to domain classification problems, they cannot easily produce a linguistic interpretation of their decisions. The response of a conventional ANN is the weighted sum of successive linear and non-linear translations usually devoid of any inductive reasoning that could explain its decision-making. Because of this ANNs are infrequently employed in medical applications which require classifications based on this inductive reasoning. Fuzzy classifiers have taken this mantel as they possess the ability to linguistically demonstrate the chain of reasoning

which results in their classifications. Rule-based expert systems can also be used for this type of decision-making, however fuzzy systems are frequently used due to their ability to handle imprecision in their reasoning scheme [2]. For this compelling reason the DSA employed a form of fuzzy classification in its decision-making strategy. This section introduces aspects of fuzzy systems that will furnish the reader with sufficient information for the purposes of this chapter. For a more in depth treatment of the subject the reader is referred to [2,3]. Chapter 6 provided a brief account of FIS as a classification framework. In order for the DSA to classify input data into different categories the necessary fuzzy rules must firstly be developed. The manner in which these rules are developed is the subject of Section 7.5, whilst the structure of fuzzy rules is firstly discussed in Section 7.4.1.

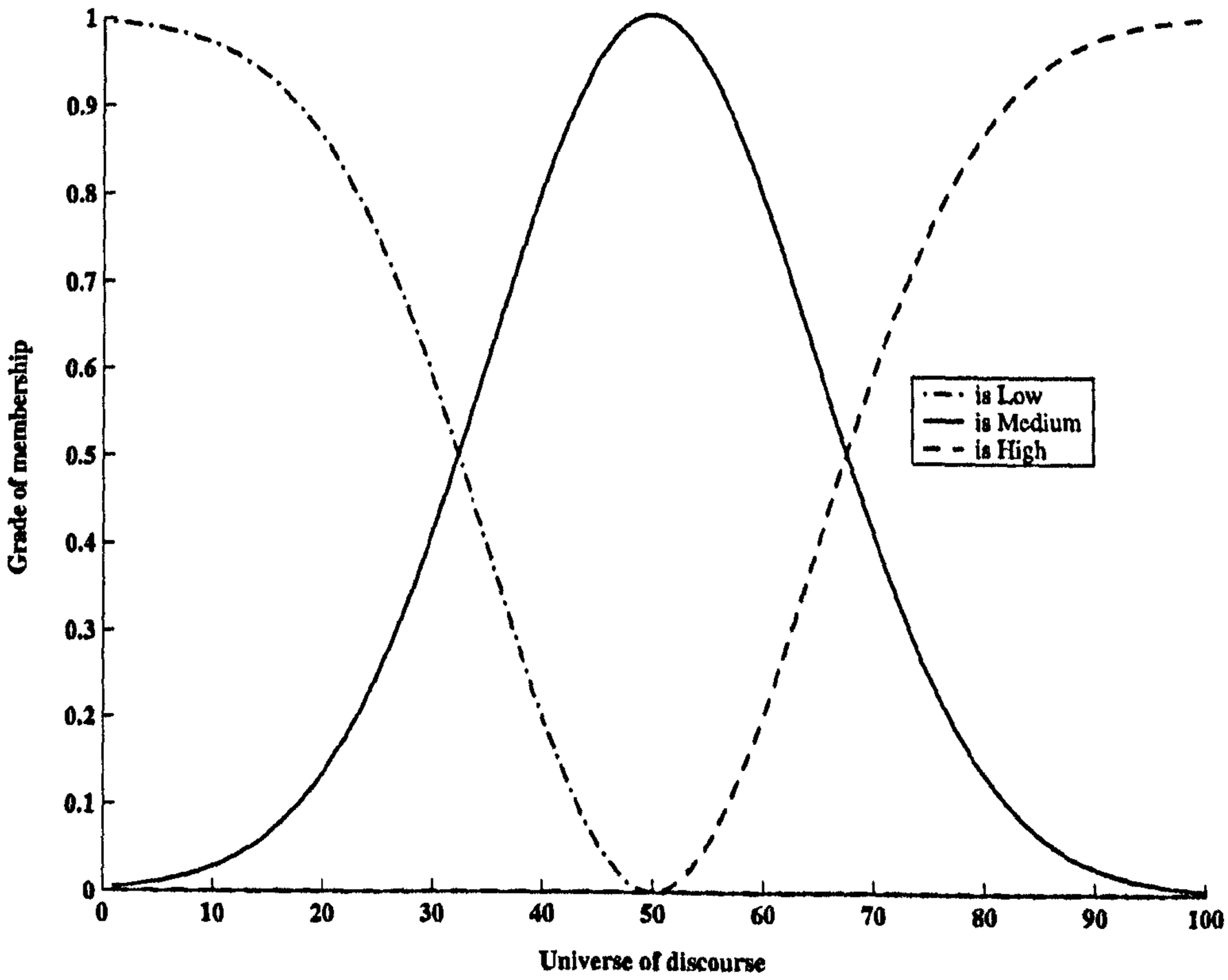


Figure 7.2. Three example membership functions denoting the grade of membership of some value *x* to the fuzzy sets *low*, *medium* and *high*.

7.4.1 Fuzzy rules in classification

In recalling Section 6.4.1 we consider some variable x to be continuously valued over the universe of discourse Φ (*i.e.*, $x \in \Phi$). Given an example in which x is bounded such that $\{x \in \mathbb{R} : 1 \leq x \leq 100\}$, three fuzzy sets A, B and C can be established describing whether x is low, medium or high respectively. The grades of membership μ define x's membership to each of the aforementioned fuzzy sets. Figure 7.2 gives three example membership functions: $\mu_A(x)$, $\mu_B(x)$ and $\mu_C(x)$ which are represented by the graphs '*is Low*', '*is Medium*' and '*is High*' respectively. These membership functions can take the functional form of a Gaussian μ_g (depicted in Figure 7.2), triangular function μ_{tr}, trapezoidal function μ_t and bell-shaped (or *Cauchy*) function μ_b [2]. There are other functions but these are the ones which are most commonly used. In their algebraic form the functions are:

$$\mu_t(x;a,b,c,d) = \max\left(\min\left(\frac{x-a}{b-a}, 1, \frac{d-x}{d-c}\right), 0\right), \tag{7.1}$$

where a, b, c and d define the position on the x-axis of the four corners of the trapezoidal function;

$$\mu_{tr}(x;a,b,c) = \max\left(\min\left(\frac{x-a}{b-a}, \frac{c-x}{c-b}\right), 0\right), \tag{7.2}$$

where a, b and c define the position on the x-axis of the three corners of the triangular function;

$$\mu_g(x;c,\sigma) = \mathrm{e}^{-\frac{1}{2}\left(\frac{x-c}{\sigma}\right)^2}, \tag{7.3}$$

where c defines the centre-point of the Gaussian function and σ defines the spread;

$$\mu_b(x;a,b,c) = \left(1+\left|\frac{x-c}{a}\right|^{2b}\right)^{-1}. \tag{7.4}$$

where a defines the spread, b defines the gradient of the spread, and c defines the centre-point of the bell function.

Returning to Figure 7.2 the response of the membership function μ depends upon the value of x and is bound such that $0 \le \mu_c(x) \le 1$ where c can be any of the fuzzy sets A, B or C on Φ. In this form the fuzzy membership function becomes a powerful tool in representing uncertainty and ambiguity in classification problems [2]. These membership functions can be used in fuzzy rules which follow the general form

if x is A then y is B, (7.5)

where x and y are some variables on respective universes of discourse X and Y, and A and B are linguistic values defined by fuzzy sets on the same universes [2]. The term 'x is A' in (7.5) is commonly referred to as the antecedent whilst the term 'y is B' is called the consequent. Given a multiple-antecedent fuzzy rule, the associated consequent term can be computed from the degree of membership

each variable is ascribed to its fuzzy (*e.g.*, the membership that x is ascribed to A in (7.5)).

Fuzzy techniques based on the principles discussed above often surpass conventional techniques for system analysis due to their intrinsic ability to deal with scenarios in which vagueness and ambiguity are prevalent. Whilst being a method suited to forming classifications (*e.g.*, medical diagnosis) fuzzy reasoning nonetheless requires a means of training for the construction of the underlying fuzzy rules (termed the *rule-base*). As was discussed in Section 6.4.1 the assemblage of fuzzy rules and membership functions forming a classification mechanism is frequently referred to as a FIS. There are two main 'flavours' of FIS: the Mamdani model, and Sugeno model [2]. The Sugeno model is computationally less complex than the Mamdani type, mainly due to its different defuzzification method (see Figure 6.13). Throughout the remainder of this chapter we will be using the Sugeno FIS model. Methods of training the FIS are considered in the following section.

7.5 Training of Fuzzy Inference Systems

This section introduces two principal means of training FIS implementations towards successfully classifying data according to some supervised learning mechanism.

7.5.1 Adaptive Neuro Fuzzy Inference System

A common approach to FIS training is via ANNs in which the creation of fuzzy rule-base antecedents and the weight attached to each rule is determined using a

combination of least-squares and error gradient descent learning algorithms. The method, known as Adaptive Neuro Fuzzy Inference System (ANFIS) [5] was briefly introduced in Section 6.4.1 as a means of identifying the influencing factors on the patient's arterial oxygenation measurements. Here the technique is evaluated as a means of training a FIS to classify data by adjusting the fuzzy rule-base parameters. For completeness, the architecture of an ANFIS is discussed and is foremost intended to illustrate the hybridisation of fuzzy rule-base with the ANN paradigm. Figure 2.1 shows some of the possible areas of intelligent system hybridisation.

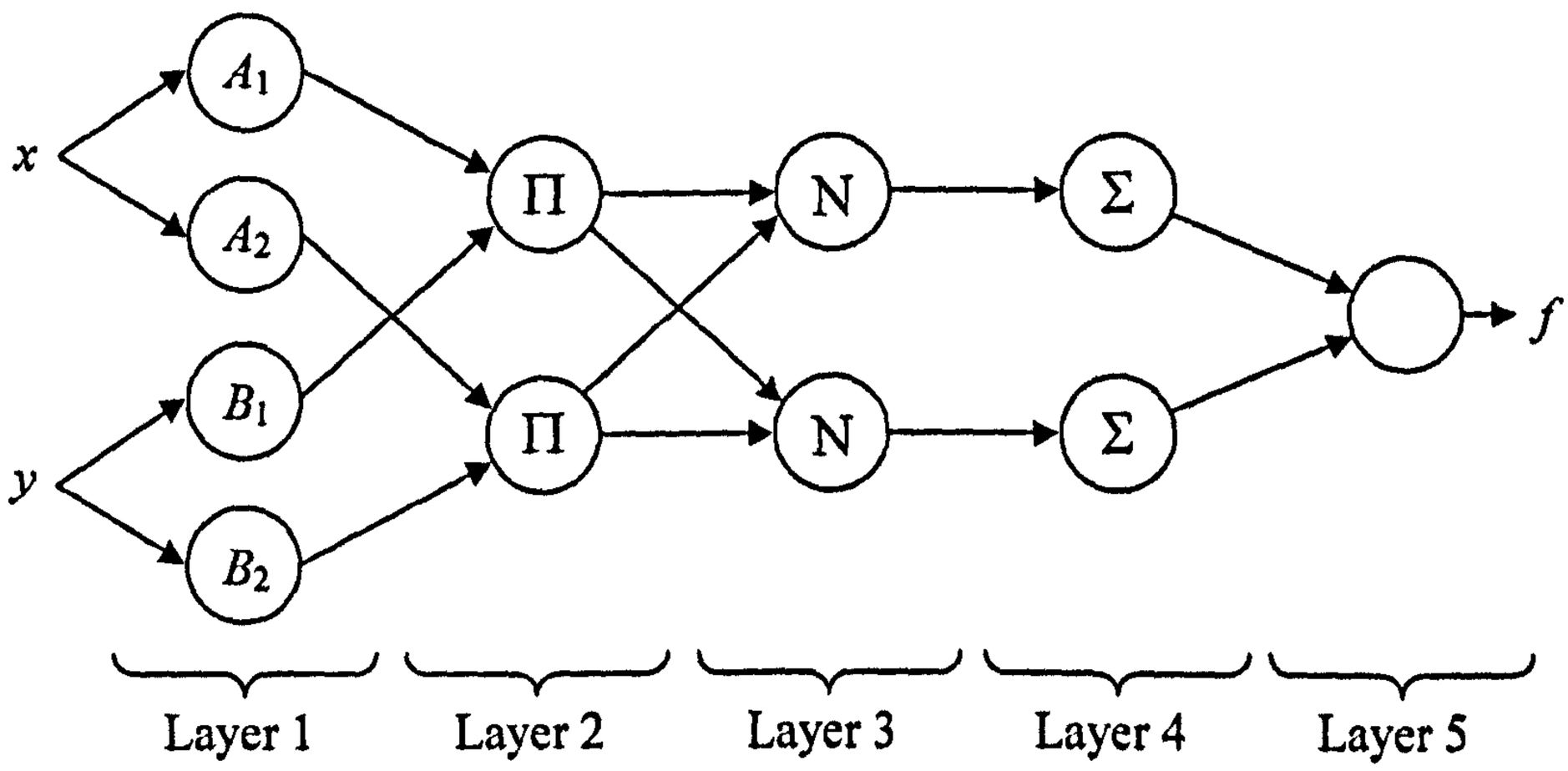


Figure 7.3. Example two-input first-order ANFIS architecture with two rules [2].

Figure 7.3 gives an example ANFIS architecture comprising of two input variables utilised in two fuzzy classification rules:

Rule 1: **If** x is A_1 and y is B_1 **then** $f_1 = p_1x + q_1y + r_1$,

Rule 2: **If** x is A_2 and y is B_2 **then** $f_2 = p_2x + q_2y + r_2$.

The purpose of the ANFIS hybridised training method is to adjust the parameters of fuzzy set membership functions and the output function parameter set $\{p_i, q_i, r_i\}$ such that subsequent classifications of x and y can be correctly determined [2]. In Figure 7.3, the nodes in layer 1 apply the fuzzy membership functions $\mu_{A_1}(x)$, $\mu_{A_2}(x)$, $\mu_{B_1}(y)$ and $\mu_{B_2}(y)$ to variables x and y in determining the grade of membership of each variable to fuzzy sets in their respective universes. Layer 2 implements the rules given above by aggregating the responses $o_{j,i}$ of the first layer (where $o_{j,i}$ is the ith node in layer j). From Figure 7.3 the response of node $o_{2,i}$ becomes:

$$o_{2,i} = w_i = \mu_{A_i}(x) \tilde{\times} \mu_{B_i}(y), \; i = 1,2, \tag{7.6}$$

where $\tilde{\times}$ is the T-norm operator, in this case derived from the union of two fuzzy sets and used as a logical AND operator for two membership functions [6]. The nodes in layer 3 normalise the outputs (referred to as rule *firing strengths*) of the second layer nodes using:

$$o_{3,i} = \bar{w}_i = \frac{w_i}{w_i + w_2}, \; i = 1,2, \tag{7.7}$$

such that $\bar{w}_i$ becomes the *normalised* firing strength of w_i. Layer 4 nodes weight the normalised firing strength of the antecedents against the class-specific output function f_i using:

$$o_{4,i} = \bar{w}_i f_i = \bar{w}_i (p_i x + q_i y + r_i), \tag{7.8}$$

where $\{p_i, q_i, r_i\}$ are the consequent parameter set. The fifth layer performs a summation of all the weighted-rule firing strengths in generating the overall FIS output via the expression:

$$o_{5,i} = \sum_i \bar{w}_i f_i = \frac{\sum_i w_i f_i}{w_i}. \tag{7.9}$$

The ANFIS hybridised training algorithm involves adjusting the FIS parameter sets during the forward-pass phase (*i.e.*, when data is being presented to the FIS during training), and via error backpropagation. In the former, the *premise parameters* (*i.e.*, the membership function parameters) remain fixed and the consequent parameter sets are adjusted using a least-squares algorithm. In the latter phase the consequent parameter sets remain fixed whilst the premise parameter sets are adjusted using an error gradient descent algorithm.

The ANFIS method of FIS generation and learning has found numerous applications in the field of biomedical engineering [7,8,9]. A recurring theme throughout the literature is that of speed and computational complexity. As is discussed later in Section 7.6.9 the ANFIS method is a computationally complex one, and typically generates rules in the order of n^n where n is the number of input variables. Therefore adaptive training via ANFIS is difficult to achieve in near real-time applications. The next section details two methods of FIS learning based on the *evolutionary* paradigm. We introduced some of medical applications of evolutionary techniques in Section 2.5.

7.5.2 Genetic Algorithm-based FIS learning

Genetic Algorithms (GAs) can be defined as search procedures based on the mechanics of natural selection and genetics [10]. To better discuss GAs – and hence their application in this problem domain – we consider the following four points made by Koza [11]:

1. There exists an environment possessing some entity which has the ability to reproduce;
2. The environment contains a population of such self-reproducing entities;
3. There is some variety between the self-reproducing entities;
4. This variety affects the entity's ability to survive in the environment.

Given a natural or artificial system adhering to these four points it becomes possible to observe an evolutionary process by which successive generations of the self-reproducing population change in response to their environment and peers (usually to the betterment of the population). Theorised by Darwin in 1859 [12] the process of evolution remained until 1962 when Holland [13,14] demonstrated how it could be applied to artificial systems. One such modality is the GA which provides a framework for codifying the problem domain and evolving solutions via a mixture of 'breeding' functions and random mutation operations. For a detailed exposition of the theory of GAs the reader is referred to [10,14,15]. For simplicity the GA is recounted in algorithmic form in Figure 7.4. In Figure 7.4, the '++' operator is programmatic shorthand for 'increment by one'. Each entity is defined by a series of problem-specific parameters which fundamentally govern its fitness and it is these parameter sets that evolve over successive generations.

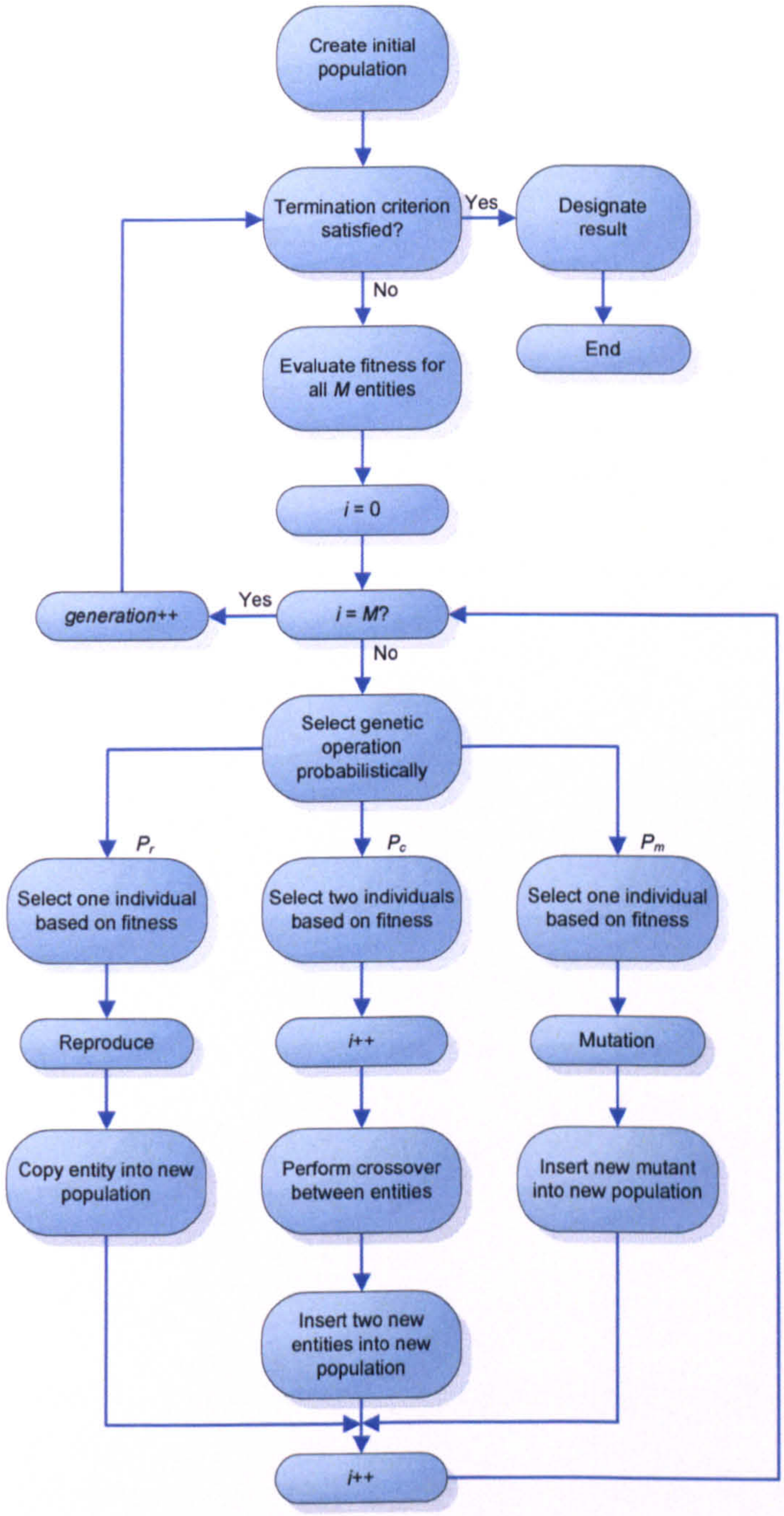


Figure 7.4. Flowchart of the conventional GA [11].

In Figure 7.4 three probabilities P_r, P_c, and P_m are defined representing the probabilities of reproduction, crossover (breeding) and mutation respectively:

- During reproduction a single entity (sometimes referred to as *chromosome* or *solution*) is copied (unchanged) into the new population.

- During crossover two chromosomes are chosen and – in the simplest case – parameter information is taken from both in equal measure to form a 'child' chromosome. The second child is created using the remaining parameter information.
- During mutation a single chromosome is randomly modified (*i.e.*, one or more chromosome parameters are randomly adjusted).

The GA is typically allowed to run over a predefined number of generations. When executing a GA – which is inherently a recurrent process – one must specify a termination criterion by which the GA can establish whether the population of chromosomes can solve the given problem. Typically the fitness of each chromosome is used such that when the best chromosome fitness exceeds a predefined value, evolution terminates. In his book Goldberg [10] recounts a variety of applications of GA to control, optimisation and classification problems which all employ different termination and execution criteria. Here it is proposed that a GA variant (GAv) can be applied to the construction and evolution of a FIS by adjusting both the fuzzy rule-base and the membership function parameter sets. The variant aspect of the proposed solution is discussed later in Section 7.6.1.

The theory of GAs as developed by Holland [13,14] and subsequent researchers was based primarily on the underlying principle that the GA operates on chromosome parameter sets of fixed size. Koza argues that this principle limits the effectiveness of GAs on numerous fronts [11]. Given that one generally lacks sufficient information concerning the optimisation problem, it is ill advised

to impose a fixed-size parameter set (or *character string*) on the GA as this can have the undesired effect of constraining the optimisation process. Koza noted that "*The initial selection of string length limits in advance the number of internal states of the system and limits what the system can learn*" [11]. In response to the lack of *dynamic variability* of GAs the *Genetic Programming* (*GP*) paradigm has been lent considerable credence [11].

7.5.3 Genetic Programming-based FIS learning

The GP and GA paradigms share a common vein in that both of these approaches implement the evolutionary schema formalised by Holland. It is with the mode of implementation where these two techniques differ. The process of machine learning (*e.g.*, the training of an ANN, or the learning of a FIS, *etc.*) can be considered to be the attempt to discover a computer program that gives a desired output for particular inputs. Koza [11] claims that the solution to a given classification problem can be reformulated as the search for a highly fit computer program in the domain space of all possible computer programs. This is analogous to the GA approach which searches parameter set space for the fittest parameter set for a given problem. For GP, the chromosomes (or *solutions*) are referred to as *programs*.

In the GP paradigm a genetic program consists of *functions* and *terminals* specific to the problem under consideration. Functions can by arithmetic operations, mathematical functions, logical operations, *etc.*; whilst terminals can be variables, parameters, internal system states, *etc.* These functions and terminals respectively form the function set $\mathcal{F}$ and terminal set $\mathcal{T}$.

As an illustration let us now consider the GP approach in the context of the problem of FIS rule-base generation and training. For simplicity, consider a function set consisting of the fuzzy operator $\tilde{\times}$ and the membership functions μ_{A_1}, μ_{A_2}, μ_{B_1} and μ_{B_2} (used in Figure 7.3). Similarly the terminal set consists of variables x and y. The sets can be formulated as:

$$\begin{aligned} \mathcal{F} &= \left\{\tilde{\times},\ \mu_{A_1},\ \mu_{A_2},\ \mu_{B_1},\ \mu_{B_2}\right\}, \\ \mathcal{T} &= \left\{x,\ y\right\}. \end{aligned} \tag{7.10}$$

If one were to express the antecedents of the two fuzzy rules given in Section 7.5.1 using a *program tree* in conjunction with (7.10) the representation would be similar to that given in Figure 7.5. As is demonstrated in Figure 7.5 the program tree decomposes the fuzzy rule antecedent terms into functional blocks that can be represented in a computer program. The GP paradigm employs the same genetic operators as the GA namely reproduction, crossover and mutation. In reproduction, program tree fragments are copied back into the program.

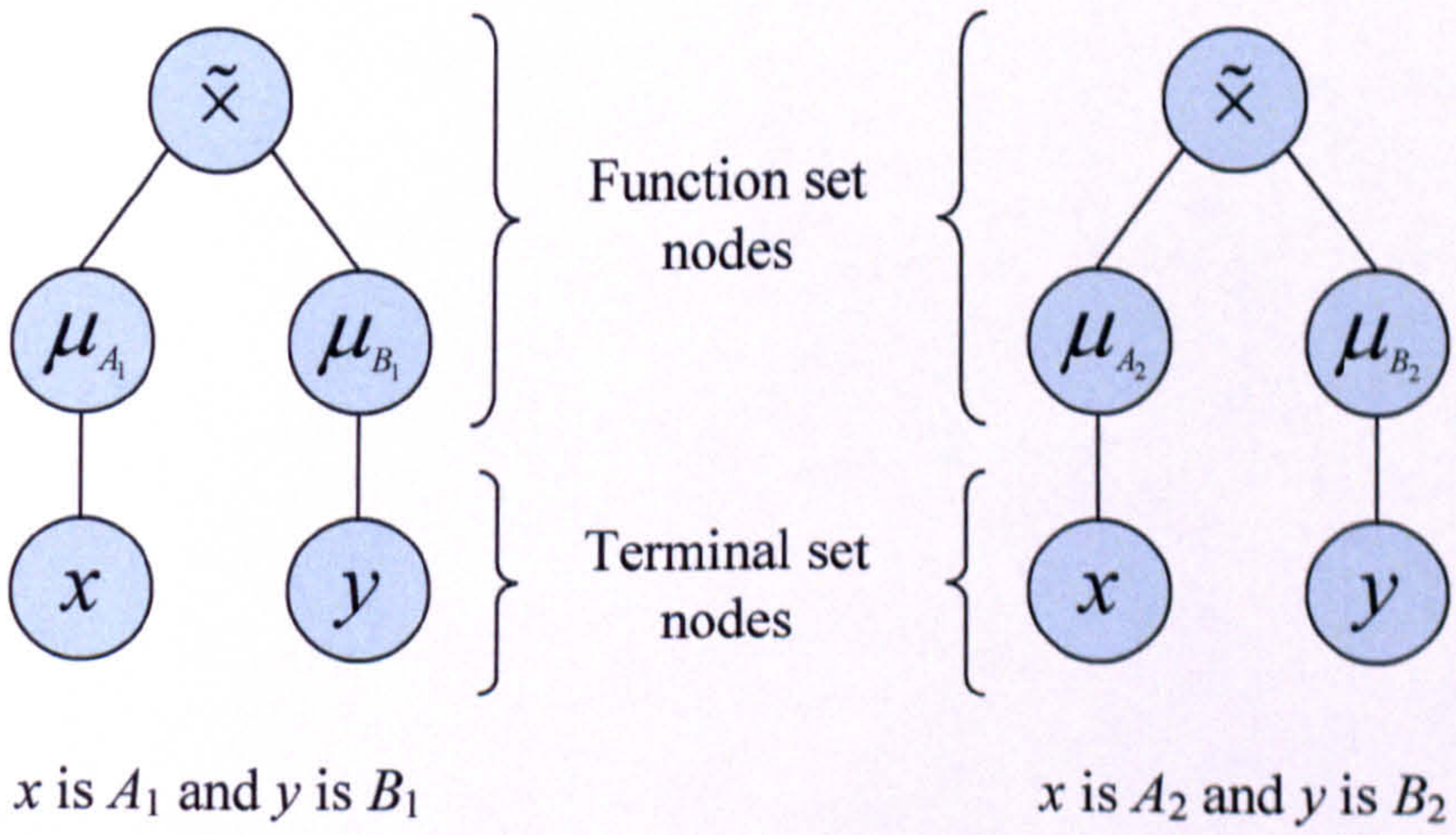


Figure 7.5. Functional program tree of fuzzy rule antecedents in the GP framework.

Similarly the crossover operation swaps two program tree fragments between two parent programs thereby generating two offspring programs. Finally mutation involves the random adjustment of either a function node, terminal node or both.

The GP approach is relatively new in relation to GA and other evolutionary strategies, and as such there are few cited examples of its application in the medical literature. In their paper Tsakonas *et. al.*, [16] apply the GP technique to the evolution of fuzzy classifiers in two medical domains: the diagnosis of the subtype of aphasia (impairment of the power to use or pronounce words), and the classification of pap-smear examinations. The authors applied a correlation measure as the fitness of each program which provided an accurate account of program performance [17]. The correlation measure C was defined as:

$$C = \frac{(tp \times tn) - (fn \times fp)}{\sqrt{(tn + fn) \times (tn + fp) \times (tp + fn) \times (tp + fp)}}, \tag{7.11}$$

where tp, tn, fp and fn are true positive, true negative, false positive and false negative classifications respectively. A further observation that Tsakonas *et. al.*, made was program tree complexity. They employed a simplicity measure that would effectively reward simple programs whilst penalising complex ones. They give the simplicity measure D as:

$$D = \frac{M - RN - (1 - R)}{M - 1}, \tag{7.12}$$

where M is the maximum allowable tree size (expressed in nodes) and N is the tree size of the program under consideration. They further define $R=(100-H)/100$ and $H=1.005(100-100/T)$ where T is the size of the training set. The simplicity measure is implemented via multiplication with the fitness measure. Given that the fitness measure is a normalised form of the correlation function C, the overall individual program fitness F can be expressed as [16]:

$$F=\frac{1}{2}(D(1+C)), \tag{7.13}$$

such that $F\rightarrow[0,1]$. This technique of evaluating the fitness of each program whilst incorporating the complexity of the program tree was implemented in the DSA, which is discussed in the following section. The process flow of a genetic program is similar to that of the conventional GA given in Figure 7.4. The GP process flow is shown in Figure 7.6.

7.6 On the development of the DSA

Two implementations of the DSA were formulated: one implementing the GAv approach to FIS evolution; and one utilising the GP paradigm. These two approaches were benchmarked against the ANFIS method of FIS training. Section 7.6.1 and Section 7.6.2 detail the development of the GAv-based DSA whilst Section 7.6.3 and Section 7.6.4 gives an account of the GP-based approach.

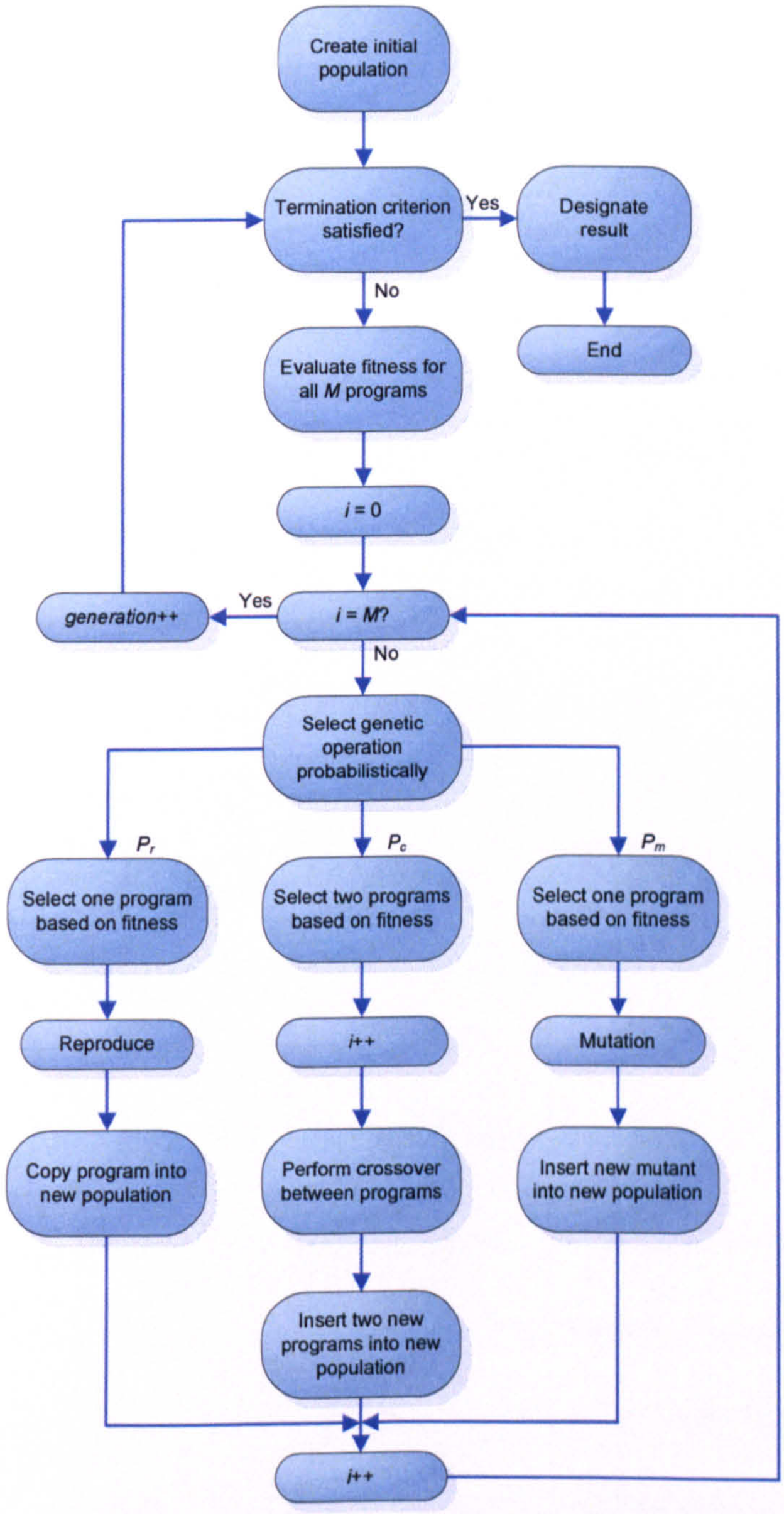


Figure 7.6. Flowchart of the conventional GP [11].

7.6.1 The Genetic Algorithm variant-based DSA

The GAv method was applied to the FIS in evolving a classifier that would provide decision support to the anaesthetist. There was a departure from the conventional GA approach insofar that this implementation could handle chromosomes of varying lengths thereby precluding Koza's statement of

dynamic variability [11]. Given the nature by which the FIS was trained the system was referred to as an *evolutionary Fuzzy Mixture Model* (*eFMM*) [18] which was the kernel of the DSA, and is described thus.

The eFMM is notionally derived from the Gaussian Mixture Model (GMM) introduced in Section 3.3.2. In many real-world scenarios, data does not generally follow set distributions *e.g.*, Gaussian, *etc.* For example, the arterial oxygen saturation readings presented in Section 6.4 follow a skewed-normal distribution. When analysed with models employing fuzzy membership functions such as those in (7.1 – 7.4) the analysing mechanism is committed to increasing rule-base complexity to account for these distributions, sometimes to the detriment of classification accuracy and sensitivity. The eFMM attempts to ameliorate this by providing a framework for the mixing of membership functions. Given the set of membership functions $\psi = \{\mu_{tr}, \mu_g, \mu_b, \mu_t\}$ one can identify a set of common function parameters: the function's centre point c and spread value σ. Other membership function parameters (such as b in (7.4)) can be derived from some arbitrarily-defined function of σ. The fuzzy membership functions are mixed by simple addition of the weighted functions of ψ such that

$$\mu(x) = w_{tr}\mu_{tr}(x) + w_g\mu_g(x) + w_b\mu_b(x) + w_t\mu_t(x), \tag{7.14}$$

where w_{tr}, w_g, w_b and w_t are weight terms which sum to unity. The weights themselves are defined by a series of triangular membership functions which are governed by a mixture coefficient m $(0 \le m \le 100)$. The membership function

$\mu(x)$ will be a linear mixture of at most two functions in ψ. Between the range $0 \le m \le 33$ the membership function will be a mixture of triangular and Gaussian components (at $m = 33$ the function is purely Gaussian). Similarly between the range $33 \le m \le 66$ the membership is a mixture of Gaussian and bell-shaped components (at $m = 66$ the function is purely bell-shaped), and so forth. The fuzzy rules are aggregated using the AND $\tilde{\times}$, OR $\tilde{+}$ and NOT $\neg$ fuzzy operators.

Whilst the membership function parameter set was real-encoded, the fuzzy operator parameter set was integer-valued.

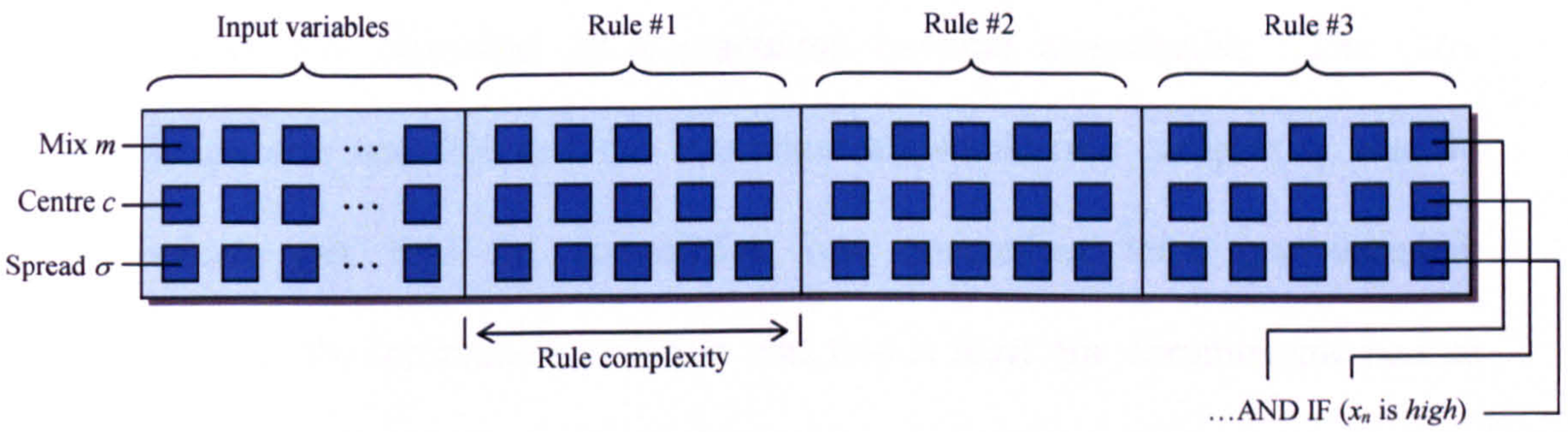


Figure 7.7. Structure of the eFMM chromosome.

The consequent terms employed by the eFMM were those introduced in Section 1.4 and subsequently used in Chapter 6, namely 'suitable for general anaesthesia', 'possibly suitable for general anaesthesia (refer patient)', and 'unsuitable for general anaesthesia'. Two constraints were imposed on the system:

1. mutation involved changing a membership function parameter AND a consequent term parameter; and
2. the maximum rule complexity was limited by some arbitrarily value (discussed below).

The structure of the chromosome is given in Figure 7.7. Recalling Table 7.1, there were 44 input variables available to the eFMM: 17 cardiovascular classifications, 14 respiratory classifications, 6 biometrics, 2 patient details, and 5 indicators. It is important to note at this juncture that the eFMM was not obliged to select all of these variables during its evolutionary learning.

Of the available dataset (discussed in Chapter 4), 150 patient samples were made available to the eFMM for learning whilst 44 samples were retained for testing purposes. The system was trained for 100 generations on a dataset which had been previously classified by a practicing hospital anaesthetist. The GAv population size was 500 and the maximum allowable rule complexity was 44 antecedents per rule *i.e.*, permitting one antecedent term per variable. Furthermore, the termination criterion was that at least one chromosome had an MSE ≤ 0.1, which was chosen arbitrarily to assess whether the population could evolve to an optimal solution. The following section details the results obtained from these two tests.

7.6.2 Results obtained from the GAv-based DSA

System evolution was terminated after 15 generations because a visual inspection of the MSE showed that the error gradient was approximately zero. Termination criteria did not include 'minimum error gradient'. Figure 7.8 shows the MSE of the best and worst performing chromosomes throughout the generations. As can be seen from Figure 7.8 the FIS converged to a solution after 5 generations, with the fittest chromosome MSE being 0.26.

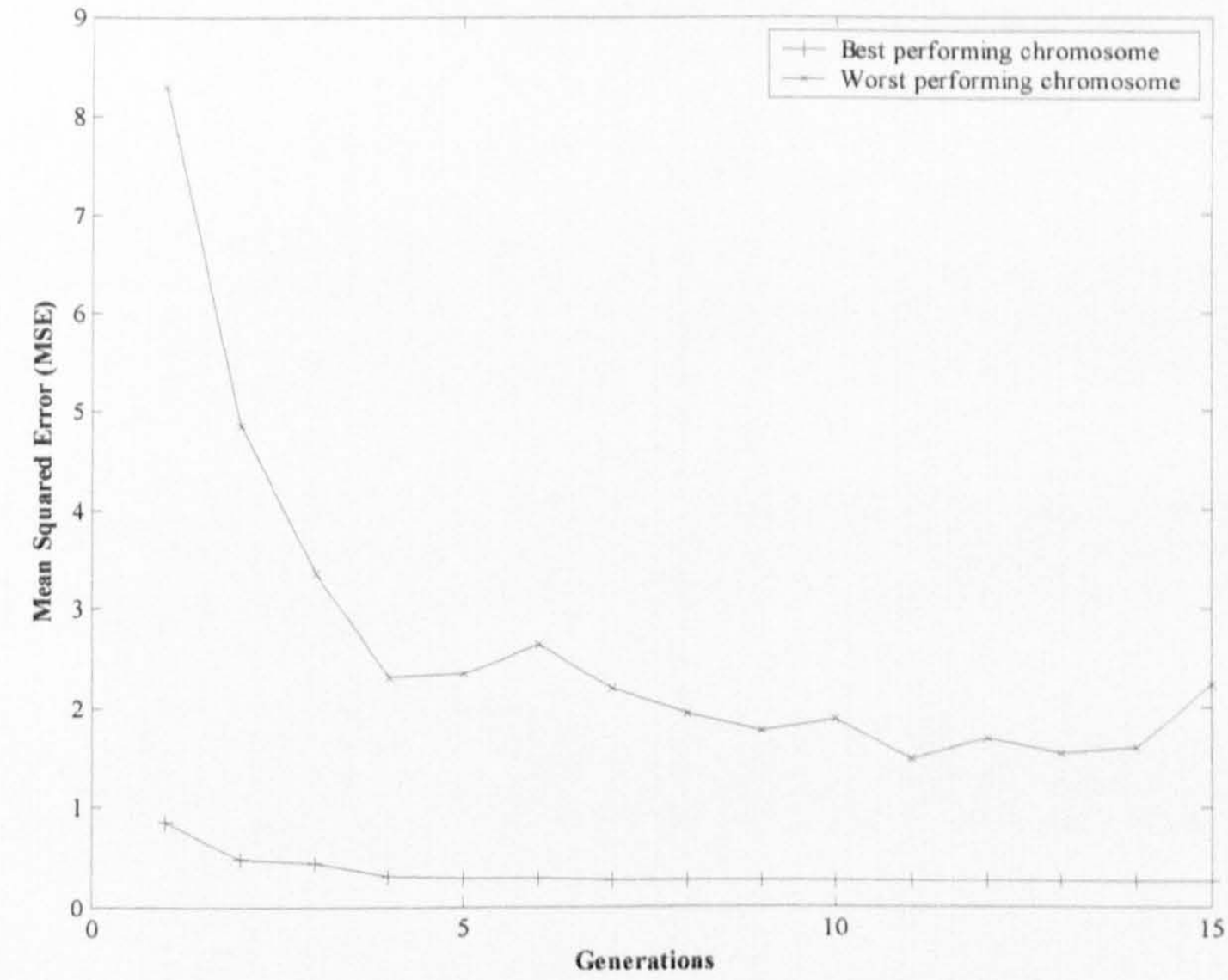


Figure 7.8. Error curves for the best and worst performing eFMM chromosomes over the generations.

The worst performing chromosome MSE appeared to stabilise at approximately 1.75. The final population was sorted according to fitness, and as can be seen from Figure 7.9, the majority of the population exhibited approximately the same MSE of 0.26.

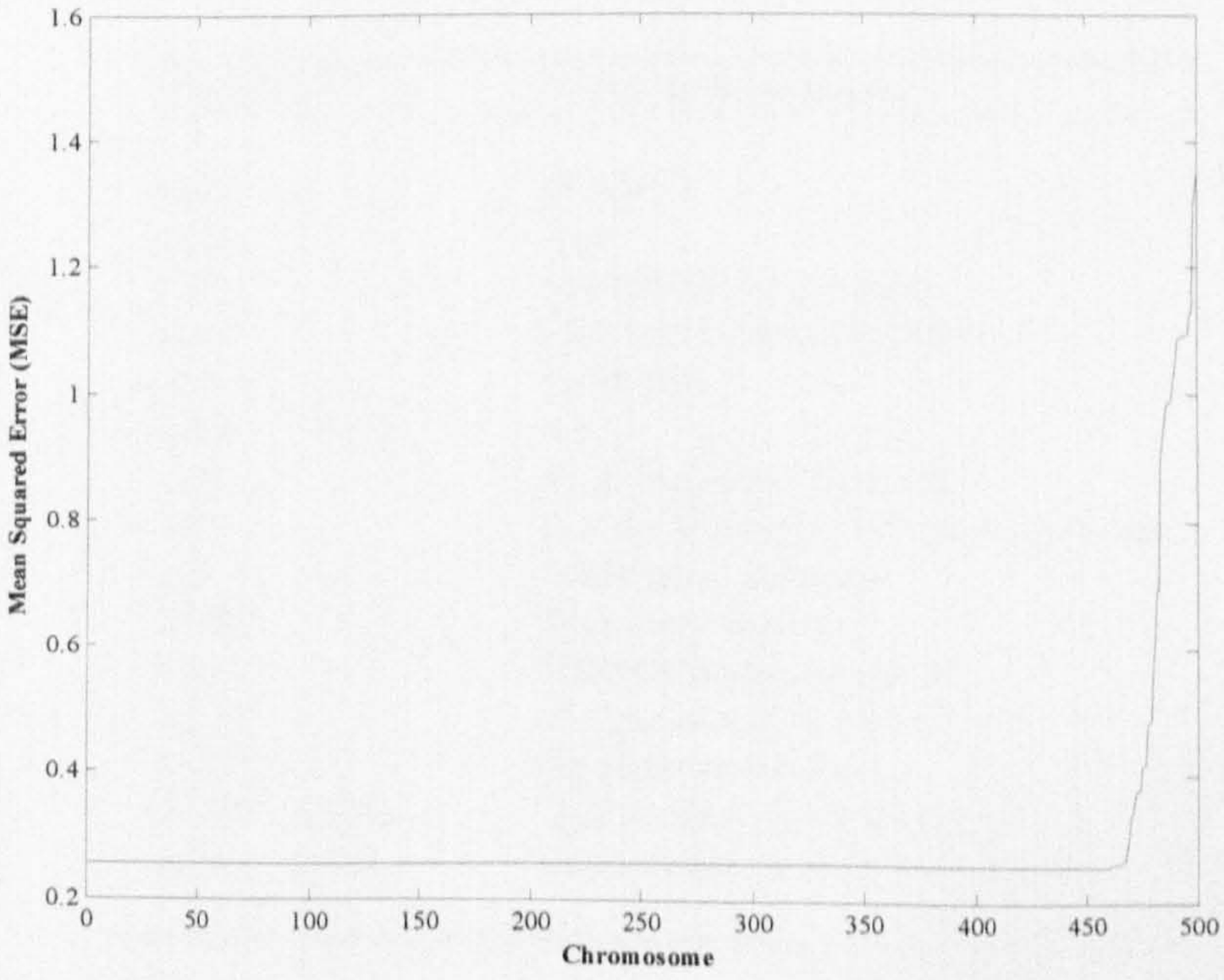


Figure 7.9. Change in error between the final eFMM chromosomes.

The rules obtained from the best performing chromosome of the final population are given as:

Rule #1: (Suitable for surgical anaesthesia)
IF (X(7) IS high) AND (X(12) IS high) AND (X(12) IS low) OR (X(4) IS NOT high) AND (X(34) IS NOT high) OR (X(2) IS low) AND (X(13) IS NOT high) OR (X(5) IS low) AND (X(8) IS high) OR (X(11) IS NOT high) AND (X(14) IS low) AND (X(4) IS NOT low) AND (X(13) IS high) AND (X(10) IS high) AND (X(8) IS low) THEN (Y IS class_1)

Rule #2: (Referral)
IF (X(10) IS low) AND (X(11) IS high) AND (X(3) IS high) OR (X(9) IS NOT low) AND (X(3) IS NOT high) AND (X(9) IS NOT medium) AND (X(5) IS low) AND (X(11) IS NOT medium) AND (X(11) IS high) OR (X(2) IS NOT medium) AND (X(5) IS low) AND (X(12) IS NOT low) OR (X(4) IS high) THEN (Y IS class_2)

Rule #3: (Unsuitable for surgical anaesthesia)
IF (X(7) IS high) AND (X(8) IS NOT high) AND (X(2) IS low) AND (X(30) IS high) AND (X(2) IS low) OR (X(10) IS NOT low) AND (X(13) IS low) AND (X(9) IS NOT high) AND (X(3) IS high) AND (X(4) IS NOT low) AND (X(5) IS low) AND (X(10) IS low) AND (X(2) IS NOT low) AND (X(13) IS NOT low) THEN (Y IS class_3)

These rules can be logically reduced by removing contradicting terms which cannot coexist. Variable descriptions are shown in Table 7.2. The reduced rulebase can be expressed as:

Rule #1: (Suitable for surgical anaesthesia)
IF (X(4) IS NOT high) THEN (Y IS class_1)

Rule #2: (Referral)
IF (X(11) IS high) AND (X(5) IS low) AND (X(3) IS NOT high) AND (X(9) IS high) OR (X(10) IS low) AND (X(11) IS high) AND (X(3) IS high) OR (X(5) IS low) AND (X(12) IS NOT low) AND (X(2) IS NOT medium) OR (X(4) IS high) THEN (Y IS class_2)

Rule #3: (Unsuitable for surgical anaesthesia)
IF(X(2) IS low) AND (X(7) IS high) AND (X(8) IS NOT high) THEN (Y IS class_3)

Variable	Variable description
$x(1)$	Gender
$x(2)$	Age
$x(3)$	Systolic blood pressure
$x(4)$	Diastolic blood pressure
$x(5)$	Heart rate
$x(6)$	SpO_2
$x(7)$	Peak expiratory flow rate
$x(8)$	Forced Expiratory volume (1 second)
$x(9)$	Medication indicator
$x(10)$	Biohazard indicator
$x(11)$	Haemodynamic indicator
$x(12)$	Cardiovascular indicator
$x(13)$	Respiratory indicator
$x(14) - x(30)$	Heart sound classification (1-17)
$x(31) - x(44)$	Lung sound classification (1-14)

Table 7.2. Variable descriptions for the evolved rulebase.

The rule 'pruning' was conducted manually after evolution. Upon inspection the above rules appear implausible; for example suitability for anaesthesia cannot exclusively be based on diastolic BP. Furthermore one would expect the rules to reference variables $x(14) - x(44)$ more frequently. It was possible to implement a means of penalising chromosomes that were 'too simple', *i.e.*, rule #1 above, but this could also have a detrimental affect by constraining the evolutionary system (recall that we also stipulated a maximum rule complexity). Given the training dataset of 150 samples, the GAv eFMM correctly classified 87.56% with a sensitivity of 81.33% and specificity of 90.67%. The eFMM was tested with the 44 remaining samples and correctly classified 90.91% with a sensitivity of 86.36% and specificity 93.18%. A discussion of these results alongside the results obtained from other eFMM implementations is given in Section 7.6.9. We now discuss the GP-based DSA.

7.6.3 The Genetic Programming-based DSA

This section provides an account of the GP-based DSA development. This approach was implemented in the Java programming language, as an object-based language was deemed the most appropriate for coding. As will be seen in subsequent diagrams, the optimal structure of the eFMM is object-based. As was discussed in Section 4.3.2, Java is a suitable language for these types of programming problems. The Java programming language is more portable *i.e.*, platform-independent than C++ thereby allowing easier deployment of the DSA and parallelisation in distributed computing environments (see Section 8.3).

The kernel of the DSA was the eFMM, as given in Section 7.6.1. The structure of the eFMM and the program environment is described in Unified Modelling Language (UML) form [19] in Figure 7.10 and Figure 7.11 respectively. The eFMM was implemented in two distinct fashions: object-based and thread-based, and the rationale behind both implementation strategies is discussed in Section 7.6.4 and Section 7.6.5 respectively.

7.6.4 The thread-based GP-DSA

With reference to the UML diagram in Figure 7.10, the `ThreadManager` class is responsible for creating and maintaining the universe of genetic programs involved in the solution to the FIS training problem. The terms *thread*, *solution* and *program* are used somewhat interchangeably, suffice to say that a *solution* is a *thread*-based genetic *program*. `ThreadManager` creates an initial population of `Solution` objects (which extend the `Thread` class). Making `Solution` an extension of `Thread` gives one the ability to execute solutions concurrently instead of sequentially.

Each `Solution` object is created by `ThreadManager` firstly writing new source code (in accordance with the GP paradigm), compiling the code, executing, and then auditing. Each solution is therefore different in its behaviour towards the classification of data, as implemented by its unique rule-base and membership function parameter set (as defined by the `ParameterSet` object). Furthermore the `ReferenceLedger` object is responsible for maintaining the audit of active GP thread-based solutions. `ReferenceLedger` audits and periodically polls the solutions in the search for the best and worst performing programs. Periodic

polling of each program's performance is governed by the `GlobalTimer` thread. Another function that `GlobalTimer` performs is the periodic spooling of data from `ThreadManager` to each solution being audited by `ReferenceLedger`.

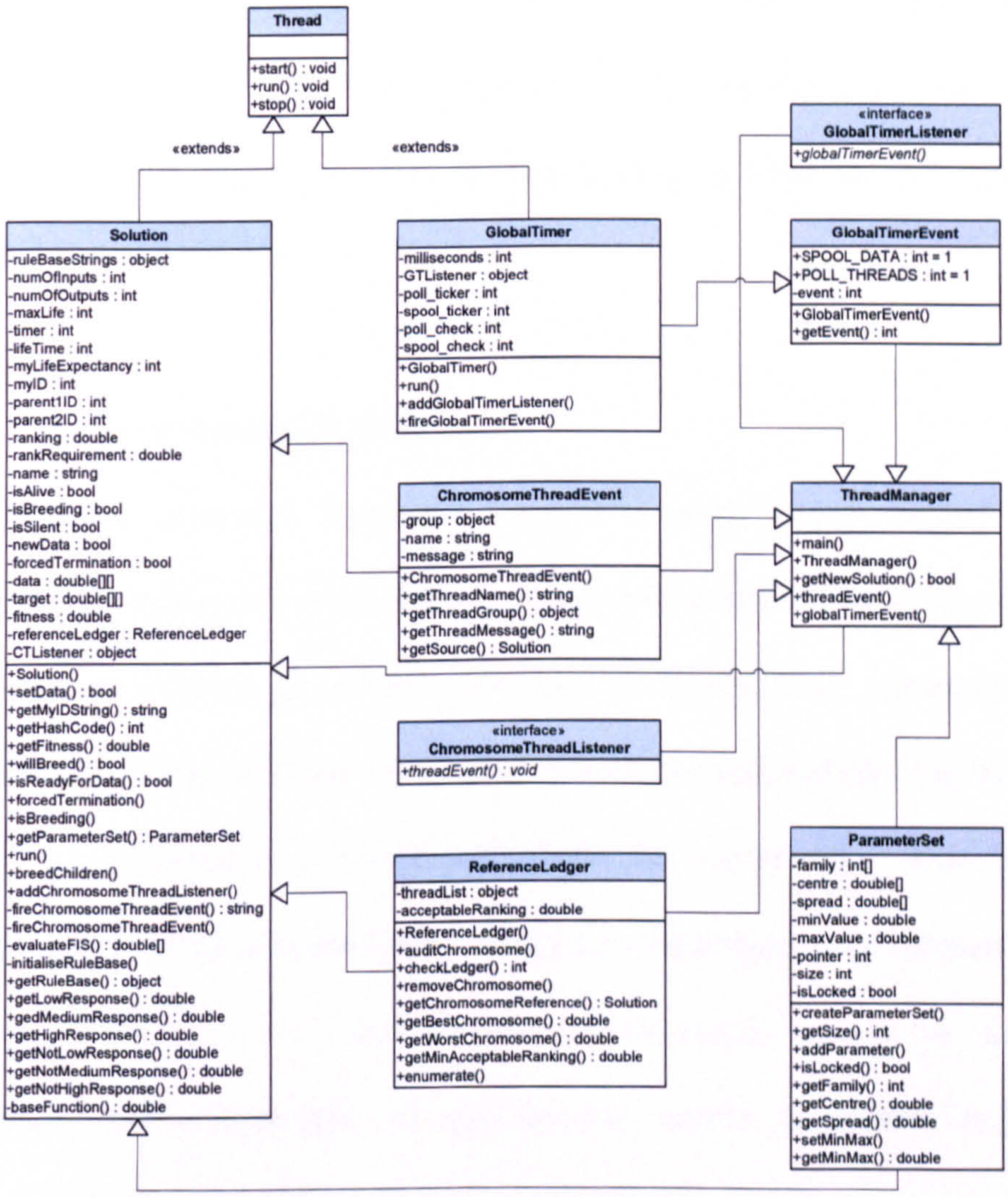


Figure 7.10. UML diagram of the thread-based GP approach to FIS training.

In the Java programming framework threads are computationally intensive constructs, so it was desirable that a configurable means of clocking the GP solution universe was in place. In the interests of computational overheads, each thread terminated at a different time. Given that each thread would need to observe all of the available training data in order to determine its fitness, under

normal circumstances every thread would terminate at the same time after being presented with the last data point. A proportion of threads would breed or reproduce thereby placing additional pressure on the simulation platform. Allowing threads to execute for longer coupled with the introduction of randomised *life expectancies* would spread out the computational load thereby improving system efficiency. The Java programming code for this system is presented in Appendix IV.

7.6.5 The object-based GP-DSA

The alternative approach investigated was the object-based method which realised each solution as a sequentially-accessed object rather than a thread. This presented two distinct advantages over the aforementioned implementation: 1) the overall system architecture complexity could be reduced (see Figure 7.11); and 2) the computational overheads could be minimised. Due to this implementation being executed sequentially the requirement for listeners (such as `GlobalTimerListener` and `ChromosomeThreadListener`) is negated. Furthermore the complexities of synchronous access to objects (*e.g.*, the reference ledger) ceases to be a design consideration.

A preliminary trial with both approaches identified that the object-based design methodology surpassed its thread-based counterpart in terms of the maximum allowable population size.

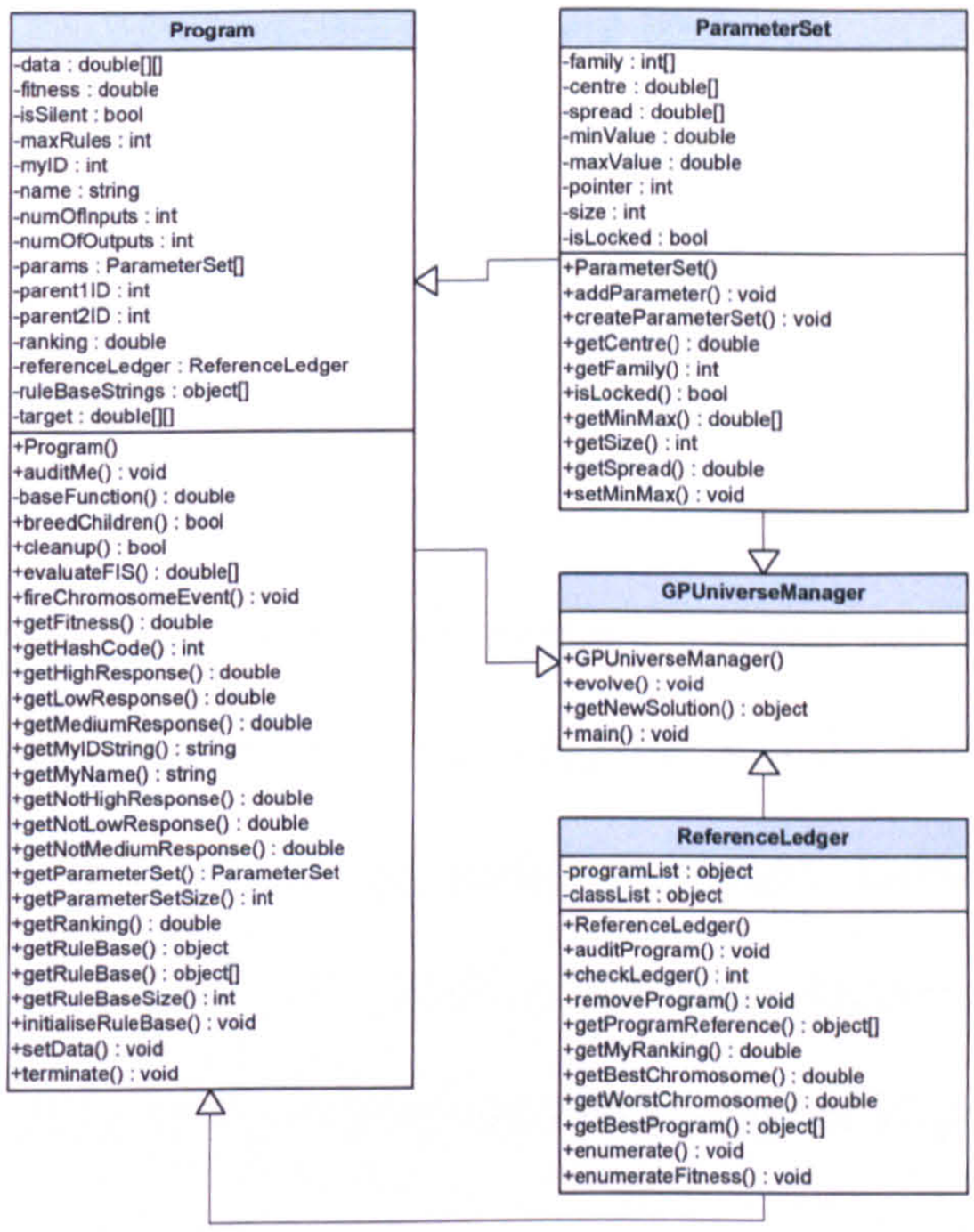


Figure 7.11. UML diagram of the object-based GP approach to FIS training.

The thread-based method could not support more than 25 parallel threads before system memory limitations (~350Mb) mandated that the program terminate. In contrast to this, the object-based approach supported 2500 concurrent genetic programs before realising the same memory limitations. In addition the object-based approach afforded significant improvements in system speed thereby motivating its use as the preferred training method for the eFMM kernel. The program code for this implementation is given in Appendix IV. Given the design overview of the GP-based DSA, the results obtained from simulations on the object-based approach are presented in the following section.

7.6.6 Results obtained from the GP-based DSA

The GP-based DSA incorporating the eFMM kernel was executed with an initial population size of 50 programs. This was performed as a preliminary test to assess the performance of the system. The poorest performing 20% of the population was arbitrarily terminated between generations, thereby freeing space for new programs. Furthermore a probabilistic measure was applied to program crossover such that the *highest fitting* programs had the highest probability of breeding, in accordance with the genetic paradigm formalised by Holland [13,14]. The probability of mutation and reproduction were both 10%. Figure 7.12 shows the error curves for each population of 20 generations.

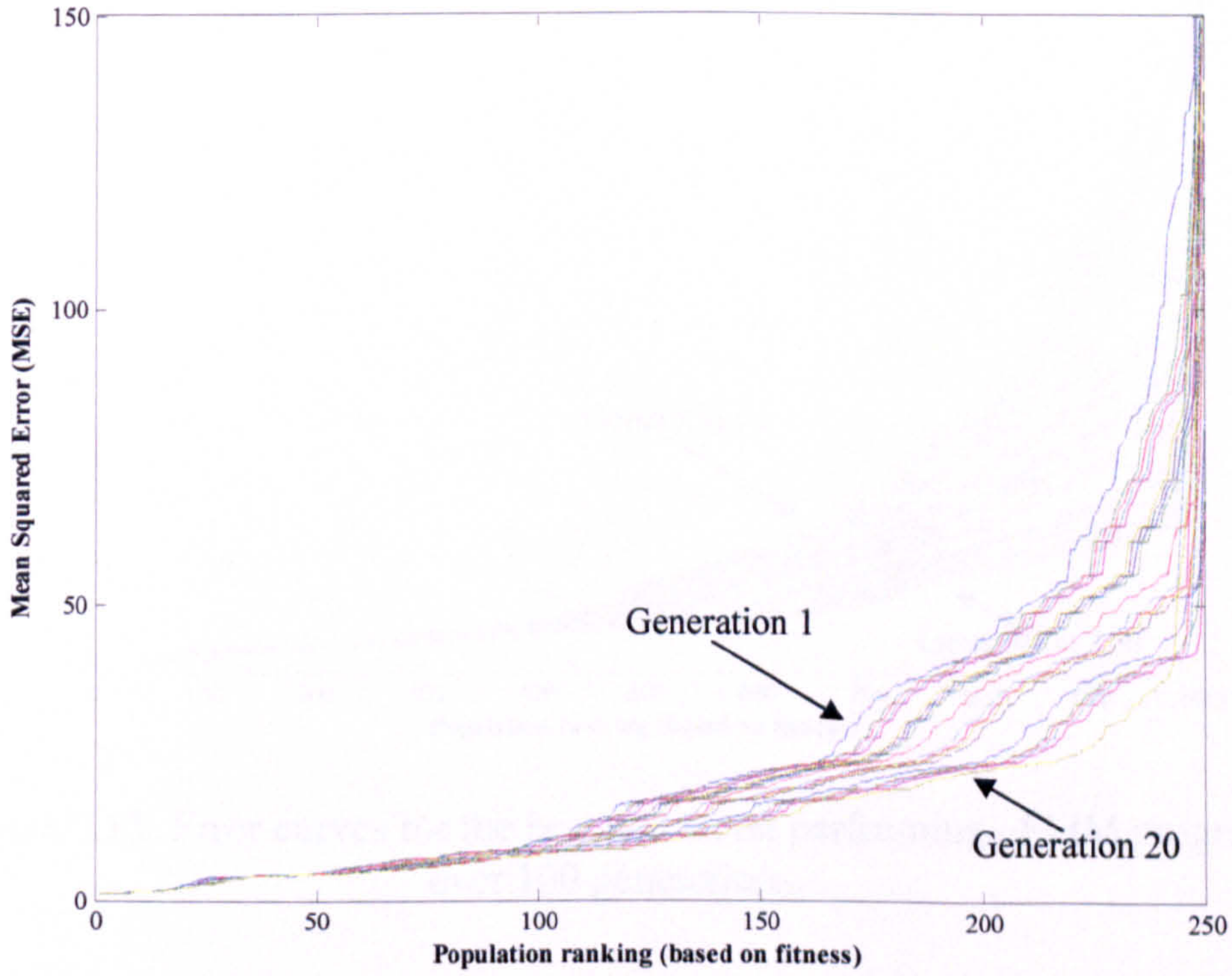


Figure 7.12. Error curves for the best and worst performing eFMM programs over 20 generations.

Figure 7.12 shows that as the population changed from generation to generation, the average program MSE became smaller. After ascertaining that the GP-based eFMM did converge and that population MSE became smaller over the

generations, the experiment was conducted again with a larger population size (1000 programs) and over 100 generations. The early stopping criterion was that at least one program MSE ≤ 0.25. Figure 7.13 shows the population MSE (sorted according to fitness) for the 100th generation. The fittest program classified the training dataset (*i.e.*, successfully predicted the surgical anaesthesia suitability group) with an accuracy of 89.86%, sensitivity of 88.37% and specificity of 92.11%. When presented with the test dataset, the fittest program correctly classified 91.79% (88.21% sensitivity, 95.18% specificity).

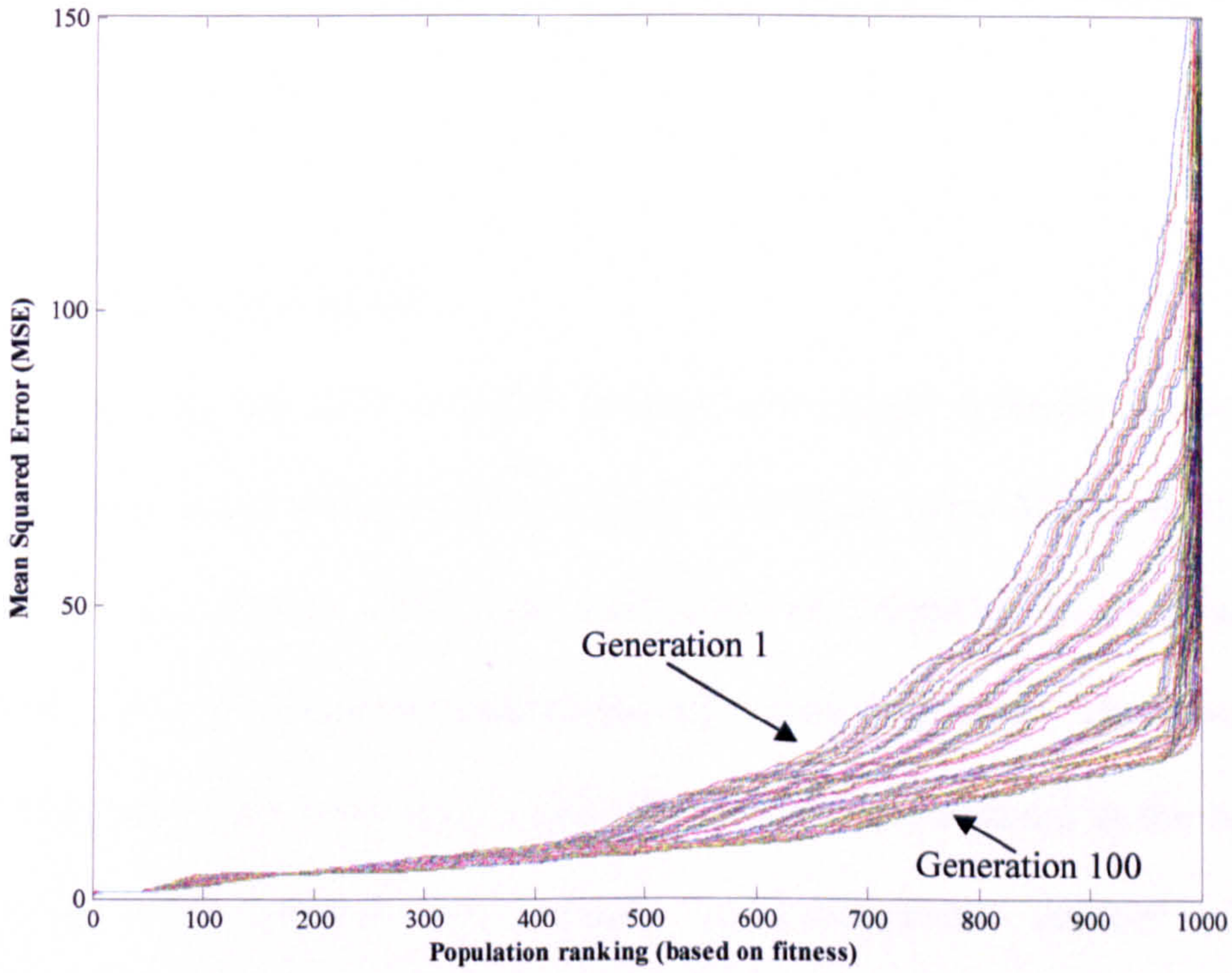


Figure 7.13. Error curves for the best and worst performing eFMM programs over 100 generations.

The optimal fuzzy rule-base, as derived by the fittest program on the final generation can be formulated as:

Rule #1: (Suitable for surgical anaesthesia)
IF (X(6) IS high) AND (X(9) IS NOT high) AND (X(30) IS medium) AND (X(4) IS low) OR (X(4) IS NOT high) AND (X(41) IS NOT high) OR (X(38) IS low) AND (X(5) IS NOT high) OR (X(8) IS low) AND (X(5) IS low) OR (X(41) IS NOT high) AND (X(3) IS low) AND (X(13) IS low) AND (X(7) IS NOT medium) AND (X(7) IS high) AND (X(10) IS low) AND (X(17) NOT high) AND (X(1) IS NOT medium) OR (X(7) IS medium) AND (X(9) IS low) THEN (Y IS class_1)

Rule #2: (Referral)
IF (X(40) IS low) AND (X(5) IS high) AND (X(6) IS high) OR (X(5) IS NOT high) AND (X(8) IS NOT high) AND (X(17) IS NOT medium) AND (X(24) IS low) OR (X(22) IS NOT medium) AND (X(4) IS high) OR (X(4) IS NOT medium) AND (X(1) IS NOT high) OR (X(13) IS low) OR (X(5) IS high) OR (X(9) IS high) AND (X(3) IS medium) AND (X(10) IS low) OR (X(19) IS NOT low) AND (X(4) IS low) THEN (Y IS class_2)

Rule #3: (Unsuitable for surgical anaesthesia)
IF (X(5) IS low) AND (X(13) IS high) AND (X(8) IS low) AND (X(11) IS high) AND (X(6) IS low) OR (X(4) IS NOT medium) AND (X(17) IS high) AND (X(12) IS NOT low) AND (X(3) IS high) AND (X(7) IS NOT low) AND (X(6) IS low) OR (X(28) IS high) OR (X(37) IS high) AND (X(2) IS high) AND (X(1) IS medium) THEN (Y IS class_3)

These results were pruned manually. These results alongside the results of a benchmark case are discussed in Section 7.6.9. Section 7.6.8 discusses the relative performance obtained from the *Evolutionary Fuzzy Neural Network* [20,21]. The following section discusses the application of ANFIS to FIS generation and training.

7.6.7 Application of ANFIS

In benchmarking the GAv and GP systems, one would typically do so with reference to an established technique such as ANFIS. The ANFIS method of constructing and training a Sugeno FIS could not be undertaken in the context of this work due to the computational complexity of the technique. There were 44 input variables to the DSA which ultimately would have resulted in the ANFIS prescribing 44^{45} decision rules; clearly an unacceptable number of rule permutations. It is plausible to suggest that a FIS with this number of rules, for a task such as this, is an inefficient implementation and as such a method of dynamic construction and pruning of the FIS connectionist architecture is desirable. In the following section we discuss the application of the Evolving Fuzzy Neural Network in the context of this problem.

7.6.8 Comparison with the Evolutionary Fuzzy Neural Network

The Evolving Fuzzy Neural Network (EFuNN), introduced by Kasabov in 2001 [20], is a hybridised intelligent systems paradigm. The system is a fuzzy ANN which allows for the construction of fuzzy rules from the network weights, and hence knowledge extraction. Furthermore the system implements a strategy of dynamically growing and pruning the connectionist (*i.e.*, ANN) architecture. For a detailed treatment of the EFuNN architecture, the reader is referred to [20,21].

The EFuNN system was trained on the 150 sample training dataset, and subsequently tested on the 44 sample test dataset. During training, the EFuNN system generated 64 rule nodes. The training algorithm did not prune or aggregate any rule nodes during the training phase. The final training RMSE was 0.48. The system correctly predicted the surgical anaesthesia suitability class with an accuracy of 96.44% (94.67% sensitivity and 97.33% specificity). When tested with the remaining 44 samples the system attained a predictive accuracy of 90.91% (86.36% sensitivity and 93.18% specificity). The resulting EFuNN architecture alongside the attained predictive accuracies is given in the following section.

7.6.9 Discussion

When comparing the performances of the GAv and GP approaches, two criteria must be considered: system accuracy and architectural size. The former relates to the ability of the system to correctly predict the 'surgical anaesthesia suitability' group, in accordance with the anaesthetist's decision. The second pertains to the 'efficiency' of the decision *i.e.*, how many antecedent rule-base

terms the system required in order to make the decision. As was evident from the rule-base evolved by the GAv eFMM, there were insufficient terms by which the system could make an accurate appraisal; yet the system attained an accuracy of 87.56%. This is suggestive of the GAv eFMM using the 'suitable for anaesthesia' category as a default classification: a potentially dangerous mode of operation. In comparison, the GP-based approach resulted in a more complex rule-base which accounted for more input variables. This would suggest that the system offered 'competition' between the rules, rather than the default-state case of the GAv-based system. Importantly however, it can be suggested that the problem is solvable in that the EFuNN technique succeeded in evolving a connectionist system in solution to the problem. A summary of the results obtained using the different approaches is given in Table 7.3. The main findings of this work are presented in the following section.

DSA Approach	Classification Accuracy
Genetic Algorithm variant	90.91%
Genetic Programming	91.79%
Evolving Fuzzy Neural Network	90.91%

Table 7.3. Summary of main results obtained from the GAv, GP and EFuNN approaches.

7.7 Conclusions

The work presented in the chapter has attempted to address the issue of categorising patients into the three 'suitability for surgical anaesthesia' groups as given in Section 1.4. Two main approaches were investigated: a GAv-based approach, and a GP-based approach. It can be demonstrated that the process of knowledge discovery (*i.e.*, the encapsulation of the anaesthetist's knowledge into

a fuzzy rule-based system) is possible, which is highlighted by the predictive accuracies obtained from both implementations. The GAv approach attained 90.91% accuracy on the unseen, test dataset, whilst the GP approach attained 91.79%. In terms of the complexity of the evolved systems, the GP implementation afforded a compact and robust rule-base from which to derive an understanding of the anaesthetist's decision-making strategy. These results, in combination with the work presented in Chapters 5 and 6, can be aggregated to form a DSA, as will be discussed in the next chapter.

References

[1] Bishop, C.M., *Neural networks for pattern recognition*, Oxford University Press, Oxford, 1995.

[2] Jang, J.-S.R., Sun, C.-T. and Mizutani, E., *Neuro-fuzzy and soft computing: a computational approach to learning and machine intelligence*, Prentice Hall Press, Upper Saddle River, NJ, 1997.

[3] Kasabov, N., *Neuro-fuzzy techniques for intelligent information systems*, (Ed., Kozma, R.), Physica-Verlag, New York, NY, 1999.

[4] Lin, C.T. and Lee, C.S.G., *Neural fuzzy systems: a neuro-fuzzy synergism for intelligent systems*, Prentice Hall Press, Upper Saddle River, NJ, 1996.

[5] Jang, J.-S. R., ANFIS: adaptive-network-based fuzzy inference system, *IEEE Transactions on Systems, Man and Cybernetics: Part B Cybernetics*, **23**(3), 1993, pp.665-685.

[6] Zadeh, L.A., Fuzzy sets, *Information & Control*, **8**, 1965, pp.338-353.

[7] Kwok, H.F., Linkens, D.A., Mahfouf, M. and Mills, G.H., Rule-base derivation for intensive care ventilator control using ANFIS, *Artificial Intelligence in Medicine*, **29**, 2003, pp.185-201.

[8] Kwok, H.F., Linkens, D.A., Mahfouf, M. and Mills, G.H., Adaptive ventilator FiO_2 advisor: use of non-invasive estimations of shunt, *Artificial Intelligence in Medicine*, **32**, 2004, pp.157-169.

[9] Trachterna, M., Wenzl, T.G., Silny, J., Rau, G. and Heimann, G., Procedure for the semi-automatic detection of gastro-oesophageal reflux patterns in intraluminal impedance measurements in infants, *Medical Engineering & Physics*, **21**, 1999, pp.195-201.

[10] Goldberg, D.E., *The design of innovation: lessons from and for competent genetic algorithms*, Kluwer Academic Publishers, MA, 2002.

[11] Koza, J.R, *Genetic programming: on the programming of computers by natural selection*, MIT Press, Cambridge, MA, 1992.

[12] Darwin, C.R., *On the origin of species by means of natural selection: or the preservation of favoured races in the struggle for life*, John Murray, London, 1859.

[13] Holland, J.H., Outline for a logical theory of adaptive systems, *Journal of the Association for Computing Machinery*, **9**(3), 1962, pp.297-314.

[14] Holland, J.H., *Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control and artificial intelligence*, University of Michigan Press, Ann Arbor, MI, 1975.

[15] Goldberg, D.E., *Genetic algorithms in search, optimisation, and machine learning*, Addison-Wesley Publishing Co., Reading, MA, 1989.

[16] Tsakonas, A., Dounias, G., Jantzen, J., Axer, H., Bjerregaard, B. and von Keyserlingk, D.G., Evolving rule-based systems in two medical domains using genetic programming, *Artificial Intelligence in Medicine*, **32**, 2004, pp.195-216.

[17] Koza, J.R., Bennett III, F.H., Andre, D. and Keane, M.A., *Genetic programming III: Darwinian invention and problem solving*, Morgan Kaufmann Publishers, San Francisco, CA, 1999.

[18] Folland, R.S., Hines, E.L. and Gongora, M.A., Evolving fuzzy rules: a mixture model approach to medical diagnosis, *Proceedings of the 5th International Conference on Recent Advances in Soft Computing*, Nottingham, UK, 2004.

[19] Holt, J., *UML for systems engineering: watching the wheels*, Antony Rowe Ltd., Chippenham, Wiltshire, 2001.

[20] Kasabov, N., Evolving fuzzy neural networks for supervised / unsupervised on-line, knowledge-based learning, *IEEE Transactions on Systems, Man and Cybernetics: Part B Cybernetics*, **31**(6), 2001, pp.902-918.

[21] Futschik, M.E., Reeve, A. and Kasabov, N., Evolving connectionist systems for knowledge discovery from gene expression data of cancer tissue, *Artificial Intelligence in Medicine*, **28**(2), 2003, pp.165-189.

CHAPTER 8

Conclusions and Further Work

This chapter summarises the main findings of this research and presents the conclusions which are formed. The chapter discusses further work that could be undertaken using this research and to deliver more generic clinical decision support tools; thus contributing to what may one day become the perfect healthcare assistant: the *virtual doctor*. This can provide the healthcare professional with all the information, *etc.* which they need so the professional can make the correct decision more effectively using more of the latest information as it becomes available over the health information network, worldwide web, *etc.* Section 8.1 summarises the main findings of this research, whilst Section 8.2 presents the main conclusions. Section 8.3 discusses further work and future potential avenues of research and development.

8.1 Summary of main findings

The main findings of this research are presented as follows:

- **Data collection.** In Chapter 4 it was proposed by the author that the collected data types constituted a sufficiently informative body of evidence on which to base a clinical decision as to the patient's suitability for anaesthesia. Given the ability of the DSA to learn the anaesthetist's decision-making strategy, it was found that the proposition was merited in that sufficient information could be conveyed during DSA learning.
- **Cardiovascular data processing.** In Chapter 5 it was demonstrated that the frequency spectra of an auscultated heart sound contained sufficient information on which to diagnose a cardiovascular abnormality. It was shown that the MLP could diagnose the abnormality with an accuracy of 99.20%; the RBFN with an accuracy of 99.88%; and the CPNN with an accuracy of 89.17%.
- **Respiratory data processing.** In Chapter 5 it was demonstrated that the Levinson-Durbin autoregression algorithm could reduce the dimensionality of the lung sounds frequency spectra whilst retaining enough information to enable the sounds to be diagnosed by the ANNs. It was shown that the MLP could diagnose the lung sounds with an accuracy of 67.11%; the RBFN with an accuracy of 95.79%; and the CPNN with an accuracy of 97.80%.
- **Arterial pulse waveform analysis.** In Chapter 6 it was demonstrated that a MLP could adequately reconstruct the arterial pulse waveform and hence enable an automated means of detecting MA and other forms of 'noise' with 98.0% accuracy. It was shown that the normal pulse waveform exhibited a

unique resonance peak that could be similarly used in identifying MA-corrupted waveforms with 100% accuracy (on the test dataset).

- **GP evolution of the DSA kernel.** In Chapter 7 it was shown that the GP paradigm could be applied to the design of a decision support rule-base tool. It was further shown that the process of evolutionary DSA learning formed knowledge from the information content of the DSA inputs. It was subsequently demonstrated that the GP approach was superior to the GA-variant approach in this context in terms of computational complexity.

The points above summarise the main findings of this research, presented in Chapters 4, 5, 6 and 7. The following section draws upon these findings in forming the main conclusions to this thesis.

8.2 Conclusions

The objective of this work was to attempt to capture and model the expert knowledge of the anaesthetist, and hence develop a decision support aid that would assist the physician in determining a patient's suitability for surgical anaesthesia. This thesis has tried to address this objective by decomposing the anaesthetist's perceived decision-making strategy into a plurality of (sub)problems. The three main problems were:

1. *How does the physician diagnose cardio-respiratory disease processes from the associated heart and lung sounds?*
2. *How can the physician determine a normal arterial pulse waveform from a corrupted waveform, and hence validate the SpO_2 measurement?*

3. *How can the anaesthetist combine the above diagnoses with additional patient information in determining their suitability for surgical anaesthesia?*

From the summaries presented in Section 8.1, it can be concluded that an automated means of cardio-respiratory disease process classification is possible using ANNs (individually and collectively). In particular, the frequency spectra of the heart and lung sounds provided sufficient information with which to form these classifications, and the parametric signal estimation coefficients could provide appropriate data for the ANNs to learn from. It can be concluded that Widrow's geometric mean approach to MLP topology estimation [1] provided a near-optimal MLP architecture with over 50% of the ANN effective parameters being used during learning.

In the context of arterial pulse waveform identification, it is concluded that the neural spike-train method of data encoding allows for sufficient information to be assimilated by the MLP during learning. The MLP is shown to be able to predict the pulse waveform with a clinically acceptable degree of accuracy. Furthermore, it can be concluded that whilst stochastic resonance can provide a sufficiently informative classification of the arterial pulse waveform, the method's computational complexity may preclude its use in (near) real-time applications.

The final conclusions relate to the function of the DSA; to its design; and to its performance in the context of the thesis objectives. In the context of this work, the main function of the DSA was to encapsulate the knowledge of the

anaesthetist, and do so in such a way that an understanding of the decision-making strategy could be imparted in the form of a rule-base. It is concluded that the evolutionary approach to rule-base generation is attractive, due to it being computationally efficient. This approach can confer the additional advantage of not generating overly-complex rules, thereby facilitating an easier understanding of the decisions it makes.

A caveat to the provision of clinical decision support (such as the DSA) is given by Dreiseitl *et al.*, [2]. The important question Dreiseitl *et al.*, posed was whether a physician would accept the diagnosis of an automated expert system or change their diagnosis if contradicted by the system. In their study they assessed the diagnostic opinion of 52 dermatologists on 25 skin legion images. They found that when contradicted by the expert system 24% of the dermatologists were willing to change their decision. Dreiseitl *et al.*, [2] also make two observations from their study that may potentially be applicable to this work:

- the number of times a decision was reversed correlates negatively with the experience level of the physician using the system; and
- physicians are more willing to accept a decision support system recommendation when they are not confident of their diagnosis.

In conclusion, it is possible to encapsulate the knowledge of the anaesthetist in an expert decision support system, in determining whether a patient is suitable for surgical anaesthesia, and that this tool can – in theory – be adopted by the medical community. The contribution of this thesis is in the demonstration that

intelligent systems, such as ANNs, fuzzy systems and evolutionary techniques, can be applied to the healthcare domain in eliciting knowledge of the anaesthetist in the context of pre-operative anaesthesia assessment.

8.3 Further work

The work presented in this thesis can be extended on numerous fronts, the most potentially advantageous being *generic clinical diagnosis*. A generic model for clinical diagnosis can be formalised to be a combined process of pattern recognition and inductive reasoning. Pattern recognition is performed by the physician identifying patient symptoms with past experience; and inductive reasoning is derived from an understanding of the relationships between clinical biometrics, *etc.* An artificial implementation of this process could potentially exist in the form shown in Figure 8.1.

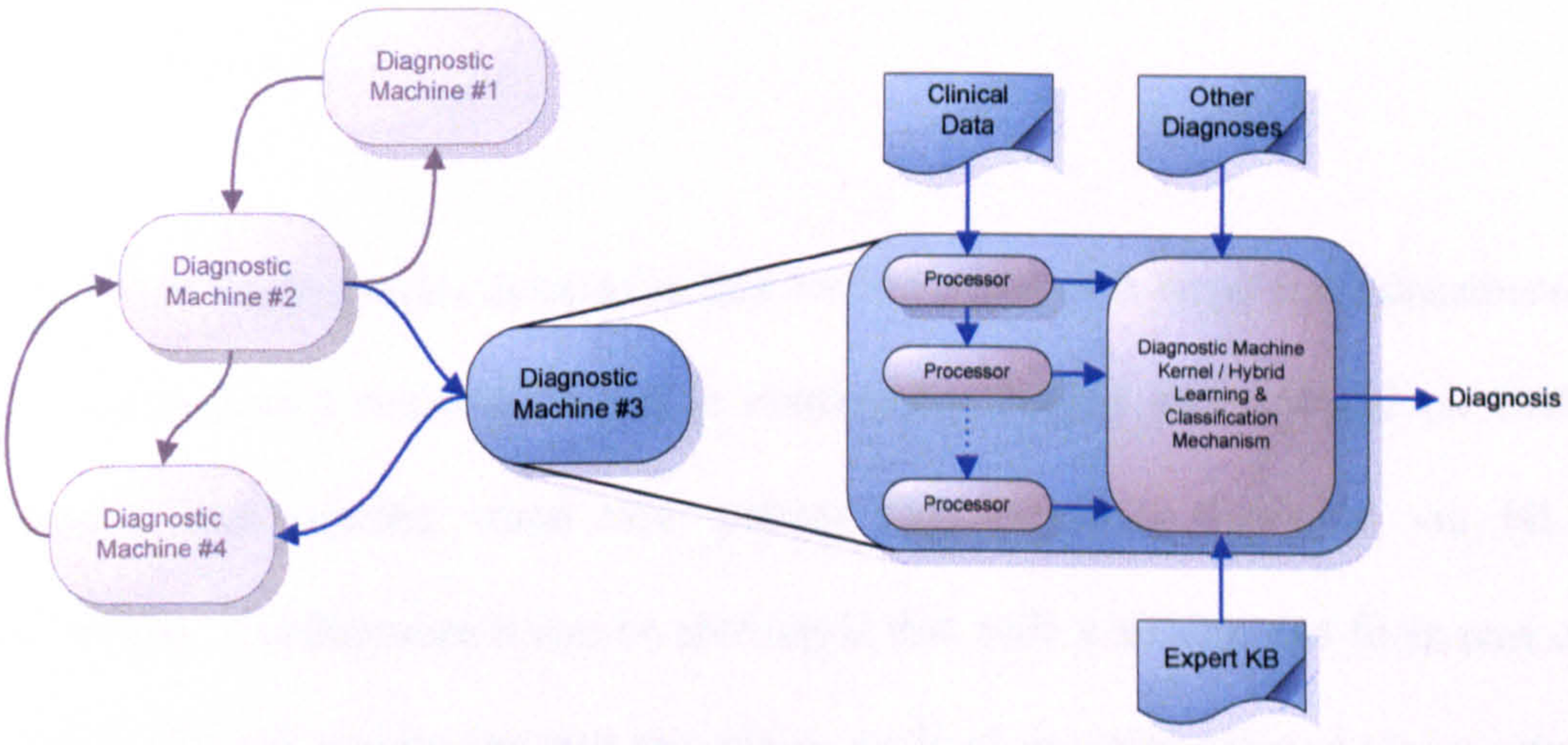


Figure 8.1. A network of clinical diagnostic machines.

In Figure 8.1 it is proposed that there exists a network of 'diagnostic machines', *i.e.*, a plurality of DSAs. From Figure 8.1, the diagnostic machine follows a structured approach of data processing (information elicitation) followed by

some expert knowledge-base (KB) system. Each diagnostic machine can be created and updated in response to new data using for example a GP-based approach.

For systems to exist like that presented in Figure 8.1, a large data repository containing biometric data and clinical diagnoses needs to be in place. With the emergence of electronic health records (EHRs), such a diagnostic system becomes possible. The Health Level 7 (HL7) international standard provides a technological framework for clinical information exchange between healthcare providers [3,4,5]. In particular HL7 standardises name-value paired messages in its framework (*e.g.*, 'name=*haematocrit*, value=*44.1%*' [5]) and defines trigger events and responses [4]. This framework would work well with the proposed network of clinical diagnostic machines as these systems would be trained on the large datasets available through EHRs and implemented through the triggering mechanisms of HL7.

A result of the proposals detailed in this section could be a large-scale distributed clinical diagnosis network, which is continually trained and updated on EHR datasets, and enacted when new patient data becomes available via HL7 messaging. Furthermore it can be envisaged that such a system can form part of a larger clinical monitoring and pro-active medical decision support system that can operate transparently. By dynamically creating, growing and pruning these clinical diagnostic machines in conjunction with continuous patient monitoring, greater understanding of medical abnormalities and their early detection may be possible.

References

[1] Widrow, B. and Lehr, M., 30 years of adaptive neural networks: perceptron, madaline and backpropagation, *Proceedings of the IEEE*, **78**, 1990, pp.1415-1442.

[2] Dreiseitl, S. and Binder, M., Do physicians value decision support? A look at the effect of decision support systems on physician opinion, *Artificial Intelligence in Medicine*, **33**(1), 2005, pp.25-30.

[3] *HL7 Reference Information Model*, Health Level Seven, Ann Arbor, MI, 2004.

[4] Beeler, G.W., HL7 version 3 – an object-oriented methodology for collaborative standards development, *International Journal of Medical Informatics*, **48**, 1998, pp.151-161.

[5] Coyle, J.F., Mori, A.R. and Huff, S.M., Standard for detailed clinical models as the basis for medical data exchange and decision support, *International Journal of Medical Informatics*, **69**, 2003, pp.157-174.

APPENDIX I

Derivation of the error BP algorithm

This appendix describes the derivation of the error BP algorithm, as given by Rumelhart *et al.*, [1]. This derivation is cited in Section 3.1.4, and demonstrates that the algorithm minimises the MLP training error by following the largest negative error gradient. The derivation is presented as follows.

We start by defining the MLP output values as o_{pj} (where p is the MLP input pattern and j is the neurone under consideration). The difference between desired output values t_{pj} and actual output values o_{pj} can be defined as $\delta_{pj} = t_{pj} - o_{pj}$. In order for the error BP algorithm to follow the largest negative error gradient, it should be shown that:

$$-\frac{\partial E_p}{\partial w_{ji}} = \delta_{pj} i_{pi} \propto \Delta_p w_{ji}, \tag{A1.1}$$

where i_{pi} is the ith element of input pattern p and $\Delta_p w_{ji}$ is the change to be made to the weight from the ith to the jth neurone. Using the chain rule and considering output neurones only, the first derivative of the overall error with respect to the neuronal weight can be expressed in terms of the derivative of the overall error with respect to the neuronal output multiplied by the derivative of the neuronal output with respect to the weights:

$$\frac{\partial E_p}{\partial w_{ji}} = \frac{\partial E_p}{\partial o_{pj}} \frac{\partial o_{pj}}{\partial w_{ji}}. \tag{A1.2}$$

From (A1.2) and considering that the neuronal output is a linear combination of its inputs and weights such that:

$$o_{pj} = \sum_i w_{ji} i_{pi}, \tag{A1.3}$$

it can be shown that the two derivatives expressed in (A1.2) can be reduced to $-\delta_{pj}$ and i_{pi} like so:

$$\begin{aligned} \frac{\partial E_p}{\partial o_{pj}} &= -\left(t_{pj} - o_{pj}\right) = -\delta_{pj}, \\ \frac{\partial o_{pj}}{\partial w_{ji}} &= i_{pi}. \end{aligned} \tag{A1.4}$$

Combining the two expressions given in (A1.4) the resultant expression can be seen to be identical to that given in (A1.1)

$$-\frac{\partial E_p}{\partial w_{ji}} = \delta_{pj} i_{pi}. \tag{A1.5}$$

The same can be seen to apply to MLP network configurations containing hidden layers and is discussed in [1]. Given a two layer network consisting of input and output layers, the calculation of δ_{pj} is a trivial exercise. However, when considering an MLP consisting of hidden layers of neurones (such as that depicted in Figure 3.2), calculation of each neurone's contribution to the overall error becomes a more complex task [2]. Indeed, calculation of the hidden layer neuronal errors and the subsequent *back propagation* of these errors in the weight update procedure is how the algorithm is given its name. Derivation of the hidden layer errors and the output layer errors is now given. Generalising the notation to accomodate input and hidden neurones gives:

$$net_{pj} = \sum_i w_{ji} o_{pi}, \tag{A1.6}$$

where $o_i = i_i$ if the neurone is in the hidden layer. A semi-linear activation function f_j can be defined as being partially linear within a certain range and when applied to (3.3) and (A1.6) yields:

$$o_{pj} = f_j\left(net_{pj}\right). \tag{A1.7}$$

Recalling that $\Delta_p w_{ji} \propto \delta_{pj} i_{pi}$ and subsituting o_{pj} for net_{pj} in (A1.2), the following expression can be obtained:

$$\begin{aligned}\frac{\partial E_p}{\partial w_{ji}} &= \frac{\partial E_p}{\partial net_{pj}} \frac{\partial net_{pj}}{\partial w_{ji}} \\ &= \frac{\partial E_p}{\partial net_{pj}} \frac{\partial}{\partial w_{ji}} \sum_k w_{jk} o_{pk} \\ &= \frac{\partial E_p}{\partial net_{pj}} o_{pi}.\end{aligned} \tag{A1.8}$$

Recalling (A1.4), it can be shown that:

$$\frac{\partial E_p}{\partial net_{pj}} = -\delta_{pj}, \tag{A1.9}$$

and therefore by substitution (A1.8) can be reduced to:

$$\begin{aligned}\frac{\partial E_p}{\partial w_{ji}} &= \frac{\partial E_p}{\partial net_{pj}} \frac{\partial net_{pj}}{\partial w_{ji}} \\ &= -\delta_{pj} o_{pi}.\end{aligned} \tag{A1.10}$$

The objective becomes to compute the value of δ_{pj} for all neurones. Therefore given (A1.9), the following can be determined:

$$\delta_{pj} = -\frac{\partial E_p}{\partial net_{pj}} = -\frac{\partial E_p}{\partial o_{pj}} \frac{\partial o_{pj}}{\partial net_{pj}} \tag{A1.11}$$

The second factor can be related to (A1.7) and it can be seen that:

$$\frac{\partial o_{pj}}{\partial net_{pj}} = f_j'\left(net_{pj}\right). \tag{A1.12}$$

Evaluation of the first factor of (A1.11) depends upon whether the neurone under consideration is an output layer neurone. If so, the expression given in (A1.4) may be applied such that the output neurone error can be expressed as:

$$\delta_{pj} = \left(t_{pj} - o_{pj}\right) f_j'\left(net_{pj}\right). \tag{A1.13}$$

If the first factor of (A1.11) is not an output neurone (*i.e.*, it is a hidden neurone), then the factor can to be expanded using the chain rule such that:

$$\sum_k \frac{\partial E_p}{\partial net_{pk}} \frac{\partial net_{pk}}{\partial o_{pj}} = \sum_k \frac{\partial E_p}{\partial net_{pk}} \frac{\partial}{\partial o_{pj}} \sum_i w_{ki} o_{pi},$$

$$= \sum_k \frac{\partial E_p}{\partial net_{pk}} w_{kj},$$

$$= -\sum_k \delta_{pk} w_{kj}. \tag{A1.14}$$

By substituting back into (A1.11), the expression for calculating the value δ_{pj} becomes:

$$\delta_{pj} = f_j'(net_{pj}) \sum_k \delta_{pk} w_{kj}, \tag{A1.15}$$

which is a recursive expression. The expression given in (A1.15) can be solved iteratively during the weight updating process. Using (A1.13) and (A1.15) the neuronal weights of the MLP can be updated during the learning process, when input patterns are being presented to the network alongside the desired targets. Recalling (A1.1), the weight update expression can be given by:

$$\Delta_p w_{ji} = \eta \delta_{pj} o_{pi} \tag{A1.16}$$

where the constant scalar η is the learning rate [1]. The larger this constant, the larger the changes in the weights. Termination of this algorithm occurs when the error gradient, calculated between iterations, falls below a certain threshold.

References

[1] Rumelhart, D.E., Hinton, G.E. and Williams, R.J., Learning internal representations by error propagation, *Parallel distributed computing: explorations in the microstructure of cognition*, (Eds. Rumelhart, D.E. and McClelland, J.L.), MIT Press, Cambridge, MA, 1986, pp.318-362.

[2] Bishop, C.M., *Neural networks for pattern recognition*, Oxford University Press, Oxford, 1995.

APPENDIX II

Patient Information Sheet

The following page gives the patient information sheet that was used to record the patient's clinical history. The information sheet is based on the design recommended by The Association of Anaesthetists of Great Britain and Ireland [1]. The sheet records information, and is split into the five categories: medication indicators, biohazard indicators, haemodynamic indicators, cardiovascular disease indicators, and respiratory disease indicators (see Section 4.4). The patient information sheet is presented overleaf.

References

[1] *Pre-operative assessment: the role of the anaesthetist*, The Association of Anaesthetists of Great Britain and Ireland, London, 2001.

PATIENT INFORMATION FORM

PID ..... DOB ..... Gender: M / F

Afro-Caribbean origin	Blood disorders / anaemia
Varicose veins	Have crowns / false teeth
Excessive alcohol intake	IBD
Infection	Smoker > 40 per day
Rheumatoid arthritis	Bleeding disorder
Diabetic	Thyroid disease
Pneumonia / Chronic bronchitis	Pregnancy
Asthma (well controlled)	Asthma (req. hospital admission)
SOB at rest	Has liver disease
Stroke / CVA	Malignancy
Kidney / urinary problems	Productive cough
Epilepsy	Wound
Cardiovascular disease	Respiratory disease
Diuretics	Contraceptive pill
Warfarin	Inhalers
HRT	Aspirin
Steroids	
MRSA Infection	HIV
Hepatitis B	Hepatitis C
Active TB	

APPENDIX III

SpO_2 Fuzzy Inference System Rule-Base

This appendix gives the FIS rule-base and membership function parameters after training with the ANFIS technique. The FIS was generated in Section 6.4.1, and contained 81 rules in its rule-base. Each rule followed the form:

$$\text{If } (x_1 \text{ is } A_p) \text{ and } (x_2 \text{ is } B_q) \text{ and } (x_3 \text{ is } C_r) \text{ and } (x_4 \text{ is } D_s) \text{ Then } y = C, \qquad \text{(A3.1)}$$

for all combinations of p, q, r and s which correspond to *high*, *medium* and *low* (*i.e.*, $3^4 = 81$ rules), where x_n are the input variables. (A3.1) can be reformulated:

$$\begin{aligned} f &= \mu_{A_p}(x_1) \tilde{\times} \mu_{B_q}(x_2) \tilde{\times} \mu_{C_r}(x_3) \tilde{\times} \mu_{D_s}(x_4), \\ f_i &= \alpha_i x_1 + \beta_i x_2 + \chi_i x_3 + \delta_i x_4 + \varepsilon_i, \ 0 < i \le 81, \\ C &= \sum_i \bar{w}_i f_i. \end{aligned} \qquad \text{(A3.2)}$$

The values α_i, β_i, χ_i, δ_i and ε_i for all rules are given in Table A3.1.

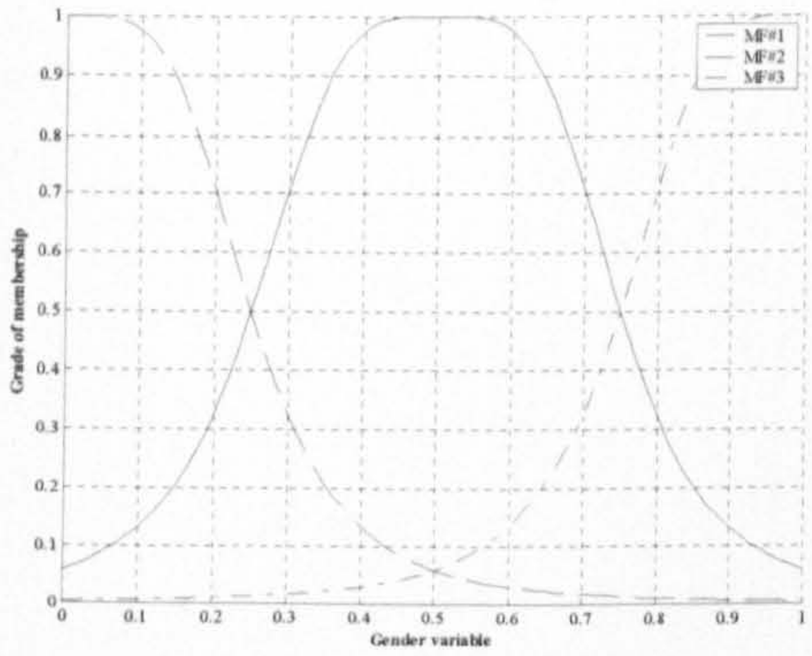
Consequent Rule	Consequent parameters for rule *i*				
	α_i (Gender)	β_i (Age)	χ_i (History)	δ_i (SpO_2)	ε_i (Offset)
1	-9.805×10^{-5}	-10.72	0.00927	-0.3561	-0.388
2	-0.001016	2.176	0.003064	-16.59	-18.03
3	0.002381	0.01703	0.03277	-2.771	3.651
4	-0.0001622	3.423	0.1393	0.1118	0.116
5	-0.001188	0.07244	0.04603	-0.8499	-1.01
6	-0.0005959	-0.07223	0.4924	0.01073	0.7043
7	-0.002689	69.55	2.383	2.277	2.382
8	-0.01939	-0.9372	0.7879	2.03	0.7177
9	-0.01263	-1.258	8.427	2.962	8.441
10	-0.001376	0.1589	0.04785	3.06	2.659
11	0.002216	0.1633	-0.0134	22.76	-30.28
12	0.00211	0.00293	0.05943	-0.1392	1.223
13	-0.01447	-0.1824	0.7188	0.8697	0.8717
14	0.0003271	0.00613	-0.2013	2.149	-1.974
15	-0.005852	-0.004401	0.8928	-1.339	0.9609
16	-0.2473	-3.294	12.3	11.88	12.31
17	0.003398	-0.05837	-3.445	14.11	-3.563
18	-0.1027	-0.07856	15.28	-22.87	15.28
19	-0.001009	0.09002	0.05675	-0.5454	-1.434
20	4.35×10^{-5}	0.239	0.2205	-37.82	19.27
21	-0.006448	0.02006	0.1078	3.04	-3.003
22	-0.01519	-0.009026	0.8525	0.9347	0.7652
23	-0.0006701	0.01171	3.312	-8.818	4.428
24	0.00565	0.01228	1.62	-3.836	1.437
25	-0.26	-0.2453	14.59	16.61	14.59
26	-0.01156	-0.03836	56.68	-113.6	56.76
27	0.1035	0.1909	27.72	-68.95	27.71
28	-0.001475	-0.6356	0.0003853	-0.02318	-0.02416
29	-0.01529	0.1318	-0.0009508	-0.9878	-1.07
30	0.03583	0.002047	0.00118	-0.1823	0.2491
31	-0.002441	0.5397	0.005788	-0.01012	0.00435
32	-0.01787	0.007951	-0.01428	-0.07566	-0.07688
33	-0.008967	-0.005864	0.01772	0.06501	0.03225
34	-0.04046	9.91	0.09906	-0.1507	0.09896
35	-0.2917	0.004457	-0.2445	-0.3097	-0.2486
36	-0.19	-0.1028	0.3033	1.3	0.3043
37	-0.0207	0.01313	-0.01162	0.1192	0.1349
38	0.03335	0.009801	-0.0005858	1.313	-1.737
39	0.03175	0.0005921	-0.002511	-0.03051	0.1031
40	-0.2177	0.008493	-0.1745	-0.7521	-0.1659
41	0.004921	0.0005427	-0.008801	0.1124	-0.1105
42	-0.08806	-3.429×10^{-5}	-0.03772	0.2018	-0.03153
43	-3.721	0.1328	-2.987	-13.04	-2.986
44	0.05113	-0.000501	-0.1506	0.6141	-0.1574
45	-1.545	-0.001183	-0.6455	3.498	-0.6451
46	-0.01518	0.0004206	-0.01184	-0.05687	-0.09895
47	0.0006545	0.01493	0.01222	-2.216	1.127
48	-0.09702	0.008345	0.01234	-0.008907	-0.2722
49	-0.2285	0.01078	-0.1778	-0.3205	-0.1829
50	-0.01008	0.001003	0.1835	-0.4979	0.2488
51	0.08501	0.001899	0.1854	-0.5643	0.1687
52	-3.912	0.1849	-3.043	-5.449	-3.044
53	-0.1739	0.002276	3.141	-6.333	3.146
54	1.558	0.02427	3.173	-9.687	3.172

Continued...

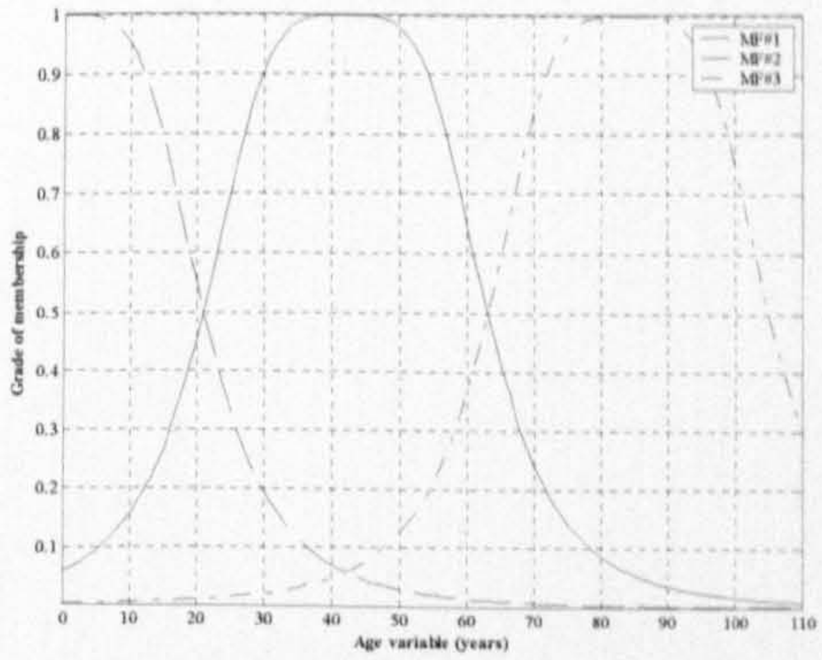
Consequent Rule	Consequent parameters for rule *i*				
	α_i (Gender)	β_i (Age)	χ_i (History)	δ_i (SpO_2)	ε_i (Offset)
55	-0.0252	-0.1964	-0.002652	-0.04176	-0.02671
56	-0.2612	0.08688	-0.01936	-0.3667	-0.3314
57	0.612	0.0181	-0.0125	-0.3588	0.6262
58	-0.0417	5.836	-0.03984	-0.2852	-0.04125
59	-0.3053	0.064	-0.2909	-0.4485	-0.3092
60	-0.1532	-0.02842	-0.1878	1.104	-0.1504
61	-0.6912	100.5	-0.6818	-4.858	-0.6819
62	-4.983	1.013	-4.979	-7.338	-4.98
63	-3.245	-0.5064	-3.215	19.33	-3.212
64	-0.3536	0.06646	-0.247	-1.013	-0.3432
65	0.5696	0.00499	0.003336	-0.2072	0.4518
66	0.5424	0.007228	-0.1024	-0.3841	0.5471
67	-3.72	0.3278	-3.71	-13.77	-3.716
68	0.08406	0.003184	0.05012	-0.2195	0.07639
69	-1.504	0.003808	-1.538	4.798	-1.5
70	-63.55	5.566	-63.5	-235.5	-63.5
71	0.8733	0.0497	0.8577	-3.563	0.8595
72	-26.39	0.05818	-26.33	82.83	-26.33
73	-0.2594	-0.0827	-0.2596	-0.4306	-0.2649
74	0.01118	0.01727	-0.01069	-0.2338	0.08614
75	-1.657	0.1231	0.1039	-3.19	-1.669
76	-3.904	0.1939	-3.901	-6.429	-3.901
77	-0.1722	0.005509	-0.1607	0.2695	-0.155
78	1.452	0.02031	1.562	-5.846	1.458
79	-66.82	3.415	-66.76	-110	-66.76
80	-2.97	0.07734	-2.75	4.864	-2.75
81	26.61	0.2255	26.73	-97.25	26.72

Table A3.1. Consequent parameters for the SpO_2 FIS rule-base.

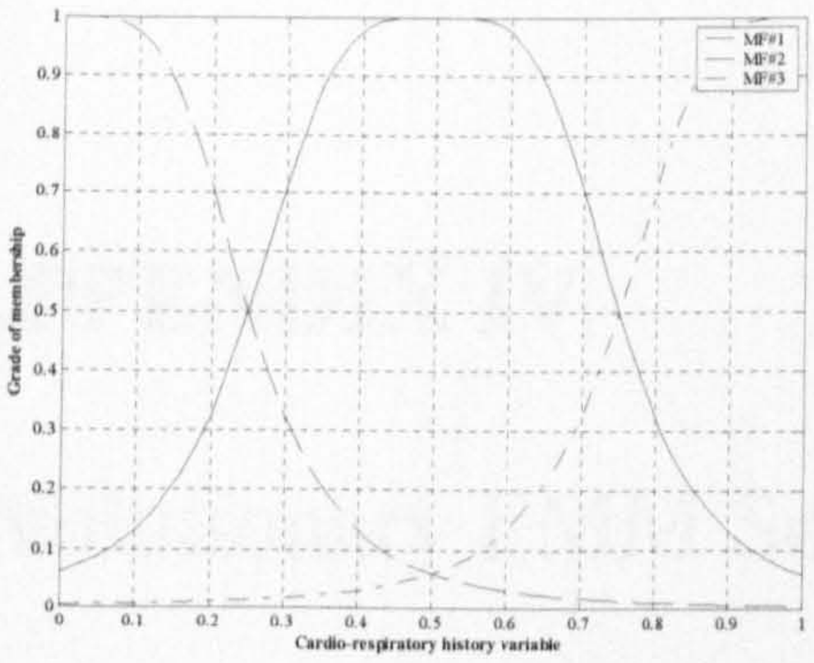
Analysis of the values presented in Table A3.1 is given in Section 6.4.1. The input variable membership functions (3 membership functions per variable) are given in Figure A3.1.



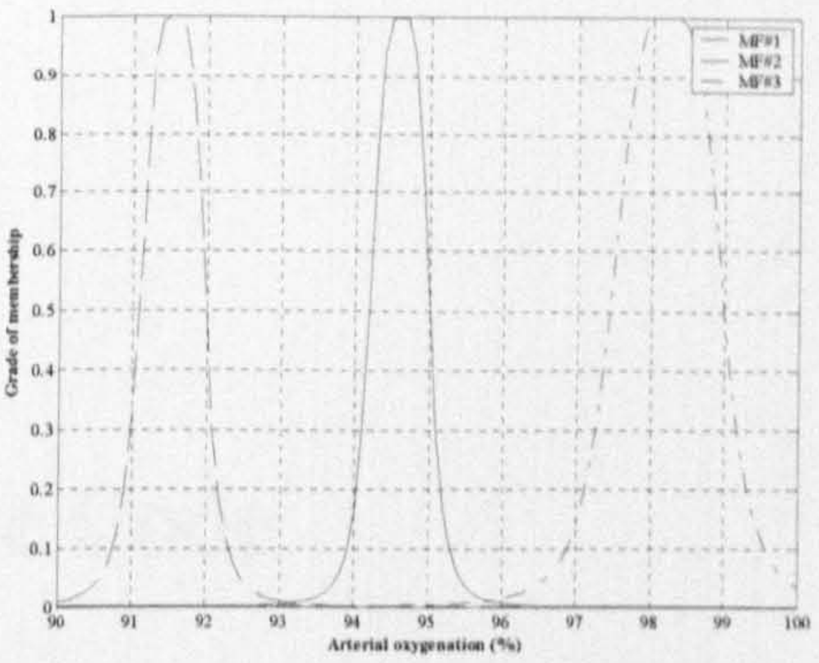
(a)



(b)



(c)



(d)

Figure A3.1. Membership functions for the four FIS input variables

APPENDIX IV

Evolutionary FMM Source Code

The source code for the three eFMM implementations may be found on the compact disc accompanying this thesis (back page). There are three directories on the disc: `eFMM_matlab`, `GP_thread_java`, and `GP_object_java`. The Java implementations will require compiling, and for optimal results, the author recommends the free Java J2SE compiler (part of the standard software development kit) available for download from http://java.sun.com.