Chapter 2

ARTIFICIAL NEURAL NETWORKS

Artificial Neural Networks (more commonly, and hereafter, referred to as neural networks) are a recently developed and fundamentally new approach to computing and artificial intelligence, which is inspired by the functioning of neurons in the brain. The neural network approach to artificial intelligence differs radically from the "Expert System" approach outlined at the end of the last chapter. The expert system approach is based upon the elucidation (from a clinician or other expert) and application of an appropriate set of rules to the solution of a problem, such as clinical diagnosis. Neural networks on the other hand are based on the observation that animals with a nervous system (i.e. a network of interconnected neurons) can make behavioural adaptations to their local environment by learning responses to stimuli.

Some responses to stimuli, such as knee jerk reflex in humans (where the leg moves forward in response to a small hammer tap at the base of the knee) are clearly hardwired into the Central Nervous System (CNS) prior to birth. The knee jerk reflex occurs when a tap stimulates receptor neurons located in a leg muscle, above the knee. These receptor neurons synapse onto sensory neurons, which connect from the leg to the spinal cord. In the spinal cord these sensory neurons synapse onto motor neurons, which inturn synapse onto muscle cells in the leg, below the knee. An appropriate level of stimulation of receptors at one end of this neural circuit leads directly to contraction of muscles at the other end. As a result of these neural connections a knee jerk response exists.

However many animals with a CNS demonstrate responses to stimuli which cannot be explained as a result of a set of “hardwired at birth” neural connections such as those which underlie the knee jerk reflex response. They demonstrate specific responses to specific stimuli, which could only have been learned in the context of their environment and experiences. They can develop new responses to stimuli, as demonstrated in the famous classical conditioning experiment of Pavlov, in which a dog learned to salivate in response to a bell. The existence of learning of this kind implies that connectivity within the nervous systems of such animals is to some extent plastic, in the sense that new sets of connections, which create new stimulus-response circuits, can somehow be created. The neural network approach to artificial intelligence is based upon the attempt to harness the learning properties of networks of interconnected neurons, to develop solutions to practical problems such as medical diagnosis, amongst other things.

An artificial neural network is a network made up of simulated "artificial neurons", called *units*, that are multiply interconnected with one another (see figure 2.1). Each unit exhibits behaviour that is similar to the behaviour of a biological neuron. A unit can have both excitatory and inhibitory inputs. Units sum their inputs and if this sum exceeds a given threshold, the unit fires an output. If the sum of the inputs fails to exceed the threshold value then the unit does not fire. This phenomenon in biological neurons has been termed the "all or none principle". Many connection configurations (topologies) of these artificial neurons into networks are possible. One configuration, the Multi-Layer Perceptron, which is described below is the one most commonly applied to Clinical Decision Making problems [Cross, Harrison & Kennedy, 1995; Price et al, 2000].

2.1 The Multi-Layer Perceptron

A Multi-Layer Perceptron (MLP) is a Neural Network configured by connecting layers of units, such that the outputs of units in one layer fully interconnect with the inputs of units in the next layer. The most common number of layers in a MLP is three (see Figure 2.1). The first layer, which is called the *Input Layer*, fires its units according to a set of inputs external to the network, which is called the *Input Pattern*. In the second layer, called the *Hidden Layer*, each hidden layer unit receives an input connection from each input layer unit. The last layer, is called the *Output Layer*, each output layer unit receives an input connection from each hidden layer unit. In neural networks that have more than 3 layers, the additional layers are nominated as hidden layers.

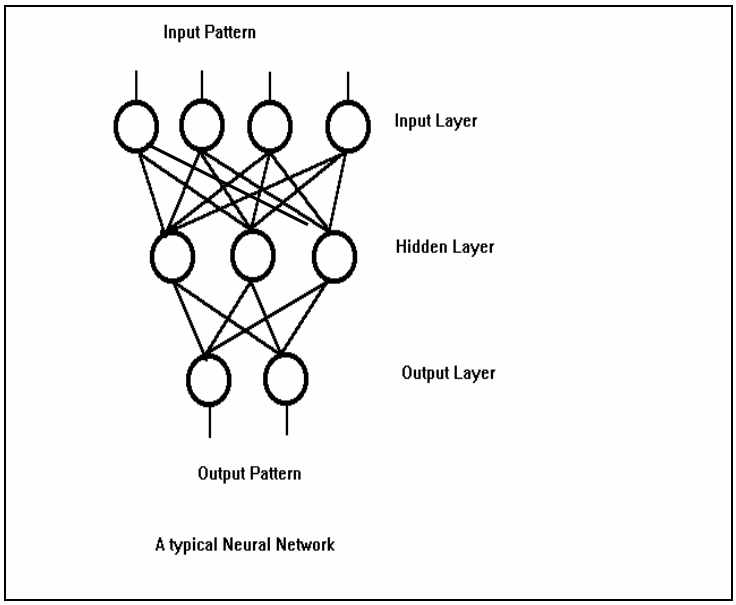


Figure 2.1 A Multi-Layer Perceptron Artificial Neural Network

For units in hidden and output layers, each input connection (the value of which is always either 1 or 0, since these are the only possible values of the output of another

unit which has either fired [1] or not fired [0]) is multiplied by a specific connection *weight* to give a weighted input. If the sum of these weighted inputs exceeds a threshold (called the firing threshold) then that unit fires (generates an output). If the weighted sum does not exceed the threshold then that unit does not fire (does not generate an output). Thus the firing pattern of the MLP as a whole depends only on two things: the input pattern and; the set of weights (across the entire MLP) used by units to weight their input connections.

In the Central Nervous Systems of animals, networks of biological neurons encode learning by modifications to their firing patterns in response to stimuli. These changes in firing patterns that mediate learning are in turn mediated by changes in the connection strength of biological synapses, the points at which information is transmitted from one neuron to another. This form of learning has been termed Hebbian Learning, after the person who first proposed it, Donald O. Hebb in his 1949 book *The Organisation of Behavior: A Neuropsychological Theory* [Hebb 1949]. In an artificial neural network, such as an MLP, each unit assigns a mathematical weight to each of its inputs. This mathematical weight is the equivalent of the connection strength of a synapse. Just as learning in brains is a function of alterations in the connection strength of synapses, learning in artificial neural networks is a function of alterations in the mathematical weights each unit gives to each of its inputs. Similarly these alterations in input connection weights can lead to changes in the firing patterns exhibited by the units and by the network as a whole. A full technical exposition of learning by MLPs is contained in Appendix 2. A briefer less technical discussion takes place in the next section.

2.2 Back-Error Propagation

For an artificial neural network, such as an MLP, to have the ability to encode learning, it requires some mechanism whereby the connection weights (hereafter referred to as only as weights) assigned by a unit to each of its inputs can be appropriately modified in response to some external stimulus. The Back-Error Propagation algorithm (Werbos, 1974) is the most commonly used process by which the weights are altered so as to encode learning [Dayhoff 1990, Bishop 1995, Reed & Marks, 1999].

During the training phase (i.e. the successive trials during which learning occurs) a neural network is presented with a large set of input patterns (the training set), one at a time. Each time it is presented an input pattern the neural network generates an output pattern according to how the input pattern causes units to fire in the hidden layer and how these in turn cause units to fire in the output layer. The neural network is then also presented with the correct answer (i.e. what its output pattern should have been). If the network's output pattern and the correct output pattern do not match then Back-Error Propagation begins. Commencing with the final output layer and then proceeding backwards through the network layer by layer, the weights of each unit are adjusted by a small amount (called the delta value), so that next time the same input pattern is presented it will be more likely to produce the correct output pattern. The cases in the training set are presented many times (often thousands of times) with all the possible combinations of inputs and correct outputs being presented many times. The training phase continues until the neural network achieves a predetermined rate of accuracy called *criterion* (e.g. 99% correct responses) with the training set. Once criterion has been achieved, the training phase is terminated and the neural network is switched into

production mode. That is, the weights are no longer adjusted, they are frozen at the values that produced the criterion accuracy, and the network can be put to use in the real world classifying cases where the answer is unknown. Such Back-Error Propagation Multi-layer Perceptron neural networks have been applied to a wide variety of pattern recognition problems, such as handwriting recognition, speech recognition, text to speech translation, image analysis, and medical diagnosis [Dayhoff, 1990].

2.3 Pattern Recognition by Multi-Layer Perceptron Neural Networks

Traditional computer applications such as word processors, spreadsheets, accounting packages or hospital patient databases are based upon the ability of the traditional computer system to manipulate and store data. Essentially, all traditional computing applications involve only the manipulation and storage of data. The advent of neural networks makes possible a new kind of application, those that involve pattern mapping. What a Multi-Layer Perceptron neural network does is take one pattern (an input pattern) and from that produce another pattern (the output pattern). After successful training it does this reliably, and is able to discriminate between many different input patterns, producing the correct output pattern for each one.

From a practical standpoint, a Multi-layer Perceptron can be conceptualised as a pattern mapping black box. The exact details of what goes on inside the box do not matter, as much as the fact that a particular input pattern will always elicit a particular output pattern. The sorts of problems, which could be considered as applications that neural networks can help to solve, are those that can be conceptualised as a pattern recognition or pattern-mapping problem. The “black box” conceptualisation of neural networks has

caused much consternation[1] amongst those more used to statistical and/or other classification techniques, which take an explicit modelling approach. However as we shall see later in this thesis, a black box approach may produce reasonably good results in some problems, but a more general framework, which looks inside the black box and reconciles MLPs with statistical and other approaches is required for their intelligent application [Cheng & Titterington 1994, Ripley 1994, Sarle 1994, Bishop 1995, Reed & Marks 1999, Hastie et al 2001].

Dayhoff [1990] lists some of the areas in which MLPs have and can be applied. On this list she includes such things as weather forecasting, financial analysis (e.g. which loan applicants should be approved), image analysis (i.e. computer vision, a machine being able to identify objects and actions in context, for example spotting tanks in satellite images or recognising human emotions by facial expression in video images), fault diagnosis in machines and industrial processes, automated control, intelligent robots that can be taught physical tasks (e.g. welding a car body, moving in an unfamiliar terrain or space), speech recognition (i.e. deciphering meaning from the spoken word), text recognition (i.e. deciphering meaning from written text), handwriting recognition (e.g. handwritten postcode recognition on mail), artificial speech (text to phoneme translation), and medical diagnosis.

[1] For Example: a number of letters criticising the "black box" nature of neural networks appeared in the journal Lancet as letters in issues following the publication of a paper by Cross et al [1995], which described the use neural networks for clinical decision making in medicine

2.4 The Use of Neural Networks in Diagnostic Medicine

One of the first diagnostic neural network applications developed in medicine is that described by Baxt [1990]. This study examined the performance of a neural network for the differential diagnosis of patients with acute myocardial infarction from those without, amongst patients presenting to an emergency department with chest pain. The neural network was a four-layer back-error propagation MLP. The input layer had 20 units, then two hidden layers of 10 units each and 1 unit in the final output layer.

The 20 variable input data set is described, in Table 2.1 below.

History	Past History	Examination	Electrocardiogram
Age	Past Acute MI	Jugular venous distension	2 mm ST elevation
Sex	Angina	Rales	1 mm ST elevation
Location of Pain	Diabetes		ST depression
Response to nitroglycerine	Hypertension		T wave inversion
Nausea & vomiting			
Diaphoresis			
Syncope			
Shortness of Breath			
Palpitations			

Table 2.1: Input variables used by Baxt [1990] to diagnose acute myocardial infarction

The data used to train and test the network consisted of 356 patients, of whom 236 did not have acute myocardial infarction and 120 did have infarction. Half this dataset was randomly chosen as the training set (N = 178, 118 without acute myocardial infarction, 60 with). After training, the other half of the dataset was used for cross validation. In the cases that the network had not previously seen, the network performed with a sensitivity of 92% and a specificity of 96%.

This study is a good example of how neural networks can be applied to a clinical decision making problem. In 1990, when the study was published, it was one of only several that had been published up to that date. Now, in 2002, there is a published literature of hundreds of studies of neural networks applied to clinical decision-making in medicine. For the purposes of exposition, some of the studies published in the medical literature, during the past decade, are presented in Table 2.2 below.

Area	Study
Diagnosis of Myocardial Infarction	Baxt [1996] In a replication of his earlier study Baxt [1990], a neural network was trained on 351 patients hospitalised for suspected myocardial infarction. It was then prospectively tested on 331 consecutive patients presenting to an emergency department with anterior chest pain. The network was directly compared to the diagnoses of emergency department physicians. The network achieved a sensitivity of 97% and a specificity of 96%. Physicians achieved a sensitivity of 78% and a specificity of 85%.
	Furlong et al [1991] trained a neural network to predict acute myocardial infarction using data on cardiac enzymes as inputs. Compared to a pathologist's interpretation of the same data, the network correctly classified 100% of cases (n=24) and 93% of non-cases (n=29). Compared to cardiologists' diagnoses made from echocardiograms, the network correctly classified 86% of cases (n=14) and 33% (n=3) of non-cases. Compared with diagnosis made on autopsy the network correctly classified 92% of cases (n=26) and 67% of non-cases (n=6).
	Baxt et al. [2002] studied 2076 who had MI ruled out and 128 who had sustained MI, who were consecutive patients presenting to an emergency department with anterior chest pain over an 18 month period. Using the neural network previously developed by Baxt [1990], 121 of the 128 were correctly identified (95% sensitivity) with a specificity of 96%.
Diagnosis of Breast Cancer	Astion & Wilding [1992] trained a network to differentially diagnose patients with malignant breast cancer from those with benign conditions on the basis of patient's age and nine biochemical variables. The network attained 80% accuracy during training on a set of 57 patients, 23 with malignant cancer and 34 with benign breast conditions. On a cross-validation with another 20 patients it correctly classified 84%. A Discriminant function derived from the original 57 cases correctly diagnosed only 75% of patients.

Diagnosis of Alzheimer's Disease	Kippenhan et al [1992] trained a network to diagnose Alzheimer's disease from PET scans. The network achieved an area under the ROC curve of 0.85 compared to that of a clinical expert 0.89. The neural network also greatly outperformed the statistical method of Discriminant Analysis.
Prediction of allograft rejection in Liver Transplantation	Hughes et al [2001] studied rejection of a transplanted liver in the period 3 months post-transplant in 124 consecutive transplants. The predictor set consisted of pre-transplant clinical and biochemical data. The neural network obtained an Area under the ROC Curve of .902 and had a sensitivity of 80% and a specificity of 90%. The neural network outperformed clinical judgement based upon the same set of input variables.
Diagnosis of microcalcifications in mammograms	Markopoulos et al. [2001] studied 108 malignant and 132 benign cases. A neural network was trained on several physical parameters of the microcalcifications to classify mammograms into malignant and benign categories. The neural network achieved an Area under the ROC Curve of .937 compared to .810 for physicians. This difference was statistically significant.
Prediction of Stage in Prostate Cancer	Han et al [2001] studied 5744 men treated for cancer of the Prostrate. Trained a neural network to predict organ confinement and lymph node involvement status using clinical and biochemical parameters as inputs. The neural network performed better than the widely used standard practice of using a nomogram based upon a logistic regression. .

Table 2.2: Some Neural Network Applications in Medicine

The studies cited in Table 2.2 indicate that neural networks are being widely considered as aids for clinical decision-making and diagnostics.

2.5 The place of Neural Networks in Clinical Decision Making

Though clinical decision making by clinicians can be dichotomously categorised as clinical judgement or statistical, to do so denies the reality that there are a range of decision-making practices. A more realistic schema is to see clinical and statistical as the extreme poles of a dimension of decision-making tasks. At one end is pure clinical decision making whereby both the information gathering and the decision-making are relatively unstructured. At the other end is pure statistical decision making whereby both information gathering and decision-making are mechanical, and there is the important proviso that the decision-making component is directly derived from an empirically derived relationship. In between there is a continuum of practices whereby there is increasing mechanisation of both the information gathering and decision making component. Such mechanisation by and of itself will increase the reliability of decision making by clinicians, but it will not necessarily increase the validity or accuracy of their decisions. Only the addition of an empirically derived decision rule ensures that the decision is valid.

To this schema we can now also add neurocomputational decision-making. This is the practice of making a clinical decision on the basis of a neural network. Neurocomputational decision-making is as yet an unknown quantity. We know from a large database of studies that statistical decision-making is the best method overall (in terms of prediction), and that structured and automated decision-making is not as good, but superior to clinical decision-making. However we do not know the relative ranking of neurocomputational decision-making amongst these alternatives. Table 2.3, below, sets out a classification of clinical decision-making practices.

Type of Decision Making	Definition	Example(s)	Reliability & Validity
Clinical Judgement	The clinician makes decisions using only their own judgement. Information gathering is relatively unstructured	Deciding that a patient has schizophrenia on the basis of interview, presentation and background information	Relatively Low
Structured Clinical Decision Making	The clinician makes decisions using a structured technique. The rules for decision-making are not empirically derived. Information gathering is unstructured or in some cases semi-structured	Deciding that a patient has schizophrenia on the basis that they fit DSM-IV criteria	Higher
Automated Clinical Decision Making	The clinician uses structured (or computerised) interview and/or information gathering, that in turn elicits a diagnosis and/or recommendations based on a structured decision making rule that has not been empirically derived.	Using the Computerised version of the Composite International Diagnostic Interview (CIDI) [Andrews 1991] Structured interviews that elicit DSM-IV or ICD-10 diagnoses. Expert Systems e.g. MYCIN [Shortliffe 1976]	Higher Still
Statistical Decision Making	The clinician uses structured information gathering and passes on the information to an empirically derived formula or rule, which generates a diagnosis or recommendation.	IQ testing and classification, Parker & Hadzi-Pavlovic's [1993] Sign based index for Melancholia, Einfeld and Tonge's [1993] Developmental Behaviour Checklist (DBC) cutoff for presence of psychiatric problems in intellectually disabled children and adolescents.	Highest
Neurocomputational Decision Making	The clinician uses structured information gathering and passes on the information to a neural network trained to make diagnoses and/or recommendations	Baxt's neural network for diagnosis of acute myocardial infarction in casualty ward patients Baxt [1990,1996,2002]	Unknown

Table 2.3: Decision Making Practices by Clinicians

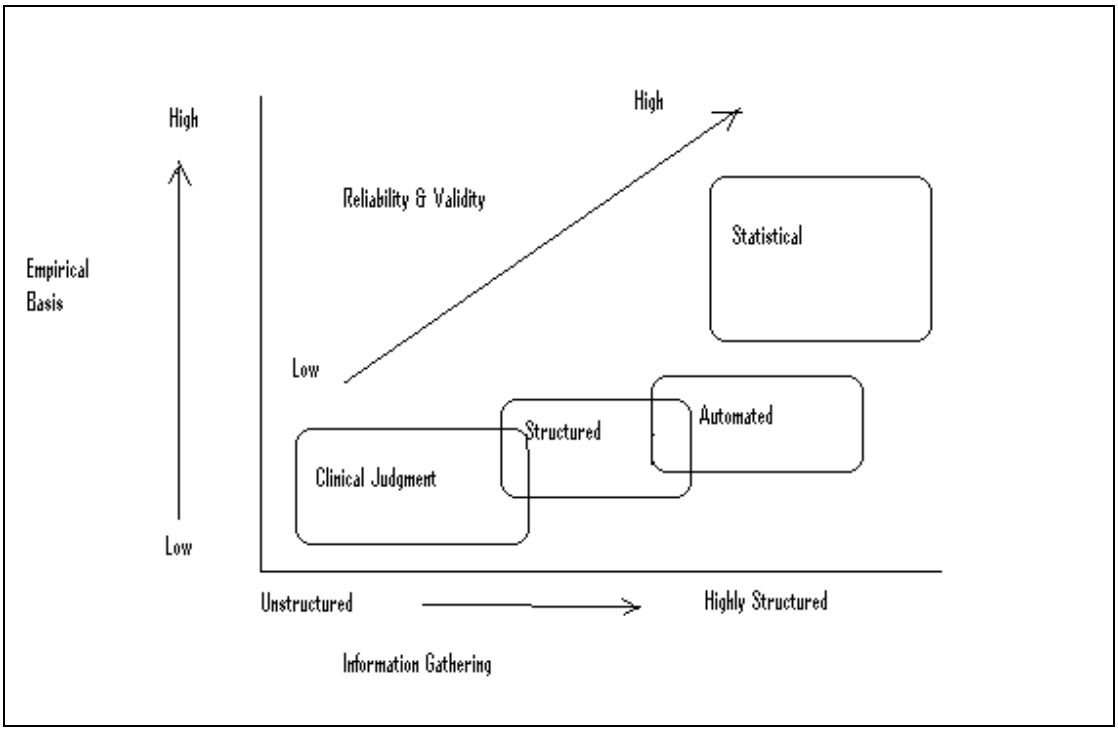


Figure 2.2 Conceptual Map of Clinical Decision Making Practices

The Conceptual Map in Figure 2.2 shows where the different practices, defined in Table 2.3, lay in a conceptual space defined by the degree of structure in Information Gathering, the degree to which the practice has an empirical basis and the overall Reliability and Validity of the practice. One of the objectives of this thesis is to determine the place of Neural Networks on this map.

2.6 Neural Networks from a Statistical Perspective

Psychologists and computer scientists initially developed neural networks, but statisticians have now become interested in them as well. A number of statistical writers have pointed out that neural networks can be readily interpreted within a statistical framework [White 1989, Ripley 1994, Sarle 1994, Weiss & Kulikowski 1991, Florio et al 1994, Bishop 1995, Reed & Marks 1999, Hastie et al 2001] and that they are very similar to statistical pattern recognition techniques such as Projection Pursuit Regression and Multivariate Adaptive Regression Splines [Ripley 1994, Sarle 1994].

Linear Classifiers

The most commonly used statistical approach to developing solutions for clinical decision-making problems is to use a linear classification technique [Dawes & Corrigan 1974]. In its simplest form, when the clinical decision is binary and it can be made on the basis of a single score on some variable, this involves finding the optimal cutoff value on this variable for classifying those with values at or above the cutoff into one group and those with values below into another.

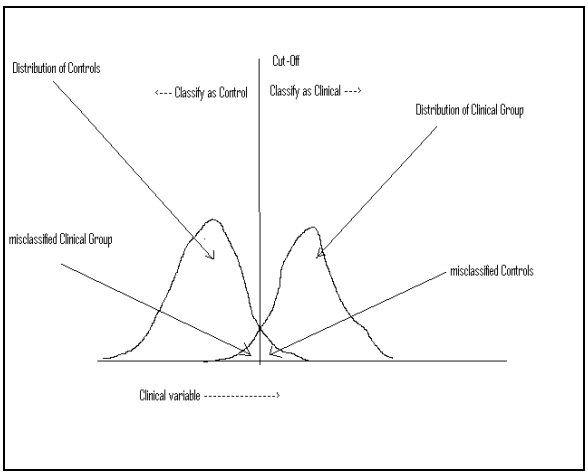


Figure 2.3 Classification using a cut-off on a single variable

When there is more than one discriminating variable, a Linear Discriminant Function Analysis (LDFA) or a Logistic Regression (LR)[2] can be used to combine the variables, using a weighted linear combination, into a single scale and then determine a cutoff on this new composite scale to classify cases into one group or another [McLachlan 1990]. LDFA is a parametric technique that uses the data to estimate parameters of the underlying distributions and then apply Bayes theorem to delineate a decision boundary, whereas LR directly estimates conditional class membership probabilities. Both techniques are optimal in the case of classes which have multivariate normal distributions with equal covariance matrices. If the population being sampled is known to have such distributions then LDFA is more efficient, otherwise LR should be preferred to LDFA for Linear classification [Hand et al 2001, Kiernan et al, 2001].

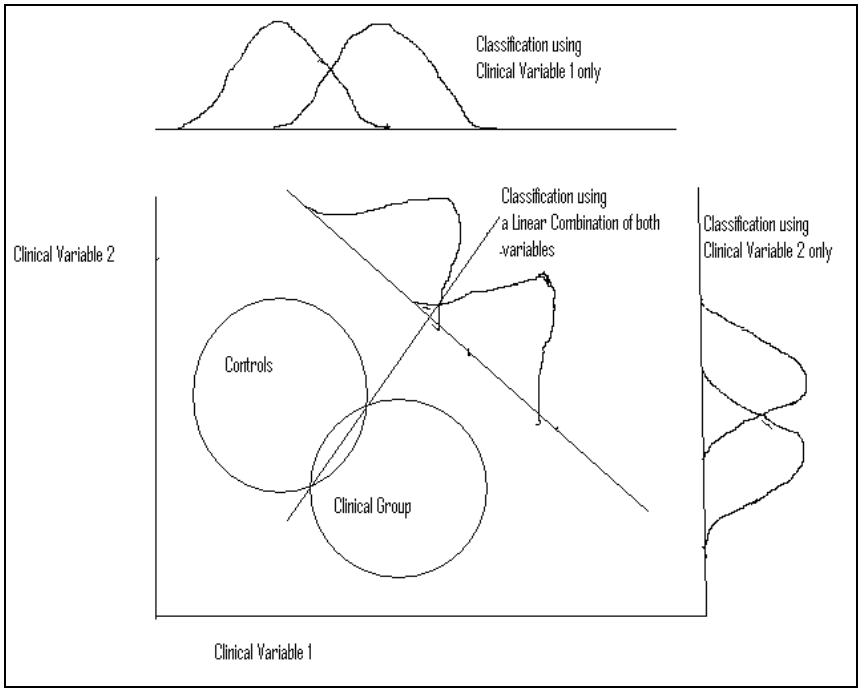


Figure 2.4 Classification using a linear combination of two variables.

[2] A common use of Logistic Regression is to determine the relative contribution of individual input variables to group membership. However a Logistic Regression can also be used to derive an equation that can be used to predict group membership on the basis of the input variables. That is, it is used as a Discriminant Function and as a method to derive a classification rule.

Figure 2.4 above demonstrates how a linear combination of two variables can produce better classification (less overlap between the two distributions and therefore fewer cases are misclassified) than either of the single clinical variables alone. The principle displayed in Figure 2.4 can be generalised to any number of variables greater than two. The basic aim of both LDFA and LR is to find a linear combination of variables which maximises classification, by minimising the overlap between the two groups.

The Bayesian Classification Decision Boundary

Every classification problem has a theoretically optimal solution known as the *Bayesian Classification Decision Boundary* [3][McLachlan 1990]. Implicit in making use of a linear approach is an assumption that the best approximation to the Bayesian Classification Decision Boundary, for a given problem, is a linear function (a straight line in 2 dimensions or linear hyperplane in higher dimensional spaces). When there is only one variable on which to make the clinical decision this assumption is always necessarily true (see Figure 2.3). However when there are two or more variables the assumption may be true (as in Figure 2.4), but is not always necessarily true (as in Figure 2.6). For such multivariate classification problems the Bayesian Classification Decision Boundary can, in theory be any function, a linear function or a non-linear function.

[3] If we know the exact distributions of the classes being classified then we can calculate the boundary, as the set of point where the probability of belonging to one or another class changes. However we do not normally know these distributions and therefore cannot know the Bayesian Classification Decision Boundary in most practical problems. We only know it exists and that it represents the absolute best possible classification decision boundary for a particular problem.

For all classification problems, and therefore all Clinical Decision-Making problems that are classification problems, all empirical classification techniques (such as LR or MLP type Neural Network) are attempts to approximate the Bayesian Classification Decision Boundary with a mathematical function that has been derived from a dataset. How accurately a classifier performs, depends upon how accurately the Classification Decision Boundary produced by the classifier is able to approximate the Bayesian Classification Decision Boundary [Ripley 1994, Sarle 1994, Bishop 1995, Reed & Marks 1999].

Piece-Wise Linear Approximation of Non-Linear Functions

When viewed from a statistical perspective, MLP-type Neural Networks are a non-linear function approximation technique. When applied to a classification problem, which has a non-linear Bayesian Classification Boundary, they can be used to approximate the non-linear classification boundary and provide a basis for classification. The approach they take to non-linear function approximation has been called piece-wise linear approximation [Ripley 1994, Sarle 1994, Weiss & Kulikowski 1991, Florio et al 1994, Bishop 1995, Reed & Marks 1999]. That is the non-linear function is approximated, by the fitting of a number of linear functions that shadow the form of the non-linear function (see Figure 2.5. below).

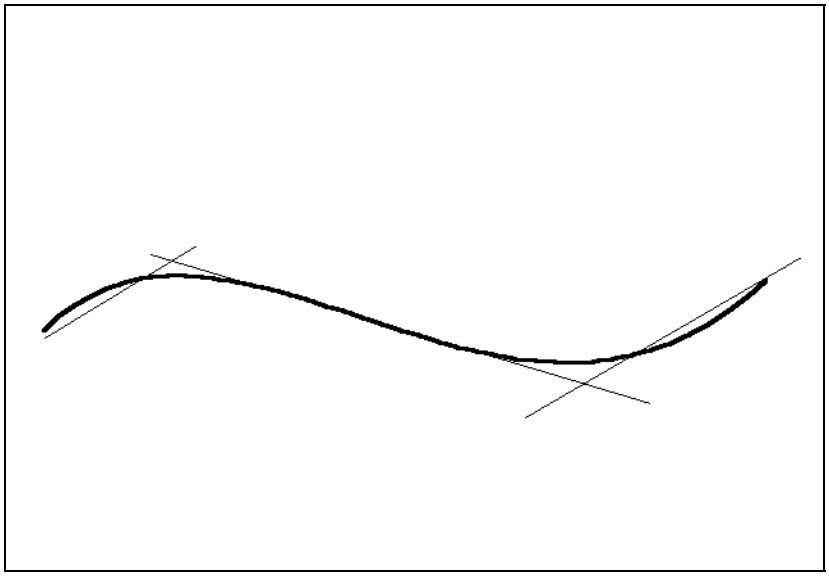
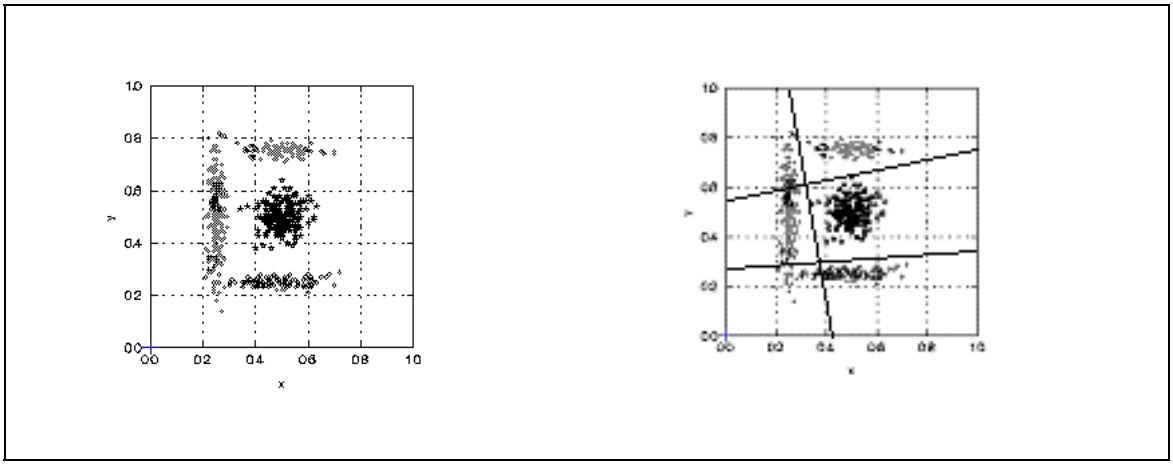


Figure 2.5 Piece-wise linear approximation (straight lines in grey) of a non-linear function (curve in black)



a) **b)**

Figure 2.6 Artificial Classification Problem:

a) Distribution plots for two classes: Diseased cases (n=300) are denoted by diamond shapes (top, bottom and left of graph), Non-Diseased cases (n=300) denoted by stars (centre of graph). The x and y axes represent scores on two symptom scales,

b) The same distribution plot with the neural network piece-wise linear decision boundary superimposed

Figure 2.6, shows how piece-wise linear approximation by a neural network can be used to solve classification problems, which have a non-linear Bayesian decision boundary. The clinical decision-making problem is to correctly classify cases using only individual case scores on the two symptoms x and y. The three straight lines in Figure 2.6 .b) were generated by the hidden units of an MLP-type neural network after it was trained on the cases in Figure 2.6. The lines clearly segment the cases into the diseased and non-diseased groups. For the data in Figure 2.6, there is no single linear classification decision boundary (single straight line), which could solve the classification problem as well as the MLP has. The Bayesian Decision Boundary for this problem is a curve. The three straight lines, in conjunction, give a good approximation to this curved boundary.

2.7 Bias – Variance Tradeoff

The goal of training a neural network for use as a clinical decision making tool in psychiatry, is not to learn to optimally classify all the cases in a training dataset (which in theory is possible [Bishop, 1995]), but rather to build a statistical model of the process which generated the dataset, and so be able to optimally classify cases from the population from which the training dataset was drawn (Bishop 1995). Accomplishment of this latter goal, with neural networks, and numerous other modeling techniques, has been the subject of much research and much theory development, over the past two decades.

A seminal contribution, to the area of modeling with neural networks, was made by Geman et al [1992], in a paper which examined the application of a decomposition of

Mean Square Error (MSE, a measure of regression fit) into two components bias and variance, as it applies to neural networks.

MSE, bias and variance are quantitative attributes of specific statistical models.

MSE, is the average squared error (difference between predicted and actual value) of a model, measured on a large independent test dataset.

Bias is the averaged difference between a model and the target function, inherent in a population, which the model is attempting to approximate. A small bias indicates the model approximates the target function well (across the measurement range of the predictors) and produces on average (across many derivation samples) good predictions of the value of the target output variable. A large bias indicates the opposite. Geman et al [1992] measured bias (see formula 2.1 below) by calculating the average error on a large test set (of size N = 600), of a number of versions (M = 50) of the same model (derived by training the model on 50 (M) training datasets (of 200 cases each) sampled randomly from a pool of 600 training cases).

Variance is the averaged difference between different versions of the same model which arise due to training on different training datasets. A small variance indicates that the model will tend to produce the same or similar value(s) in response to the same set of values of the input variables, regardless of which training dataset (sampled from a population) is used to derive values for the parameters of that model. In essence all training datasets will result in the same or very similar set of model parameter values for

that model. A large variance indicates the opposite. Geman et al [1992] measured variance (see formula 2.1 below) by calculating the average difference on a large test set (of size N = 600), between a number of versions (M = 50) of the same model (derived by training the model on the 50 training sets, sampled randomly from a pool of 600 training cases), and the average response of all these models (see below for how this is calculated). Equation 2.1 below presents the equation used by Geman et al [1992] to calculate MSE, bias and variance and also shows the interrelationships between these terms.

$$\frac{1}{MN}\sum_{i=1}^{N}\sum_{j=1}^{M}(t_i - y_{ij})^2 = \frac{1}{N}\sum_{i=1}^{N}(t_i - \bar{y}_i)^2 + \frac{1}{NM}\sum_{i=1}^{N}\sum_{j=1}^{M}(\bar{y}_i - y_{ij})^2 \qquad (2.1)$$

$$\underbrace{\qquad\qquad}_{\text{MSE}} = \underbrace{\qquad\qquad}_{\text{bias}^2} + \underbrace{\qquad\qquad}_{\text{variance}}$$

Where :

N is the number of cases in the test dataset

i is an index for cases in the training dataset, i ranges from 1 to N

M is the number of training datasets used

j is an index for training data sets, j ranges from 1 to M

t_i is the true or target value of the ith case in the test dataset

y_{ij} is the output of the model trained on the jth training dataset to the ith case in the test dataset

$\bar{y}_i = \frac{1}{M}\sum_{j=1}^{M} y_{ij}$ is the average output of the M models derived from M training datasets for the ith case in the test dataset.

Note: Equation 2.1 above has been adapted from Geman et al [1992], by Keijzer 2002

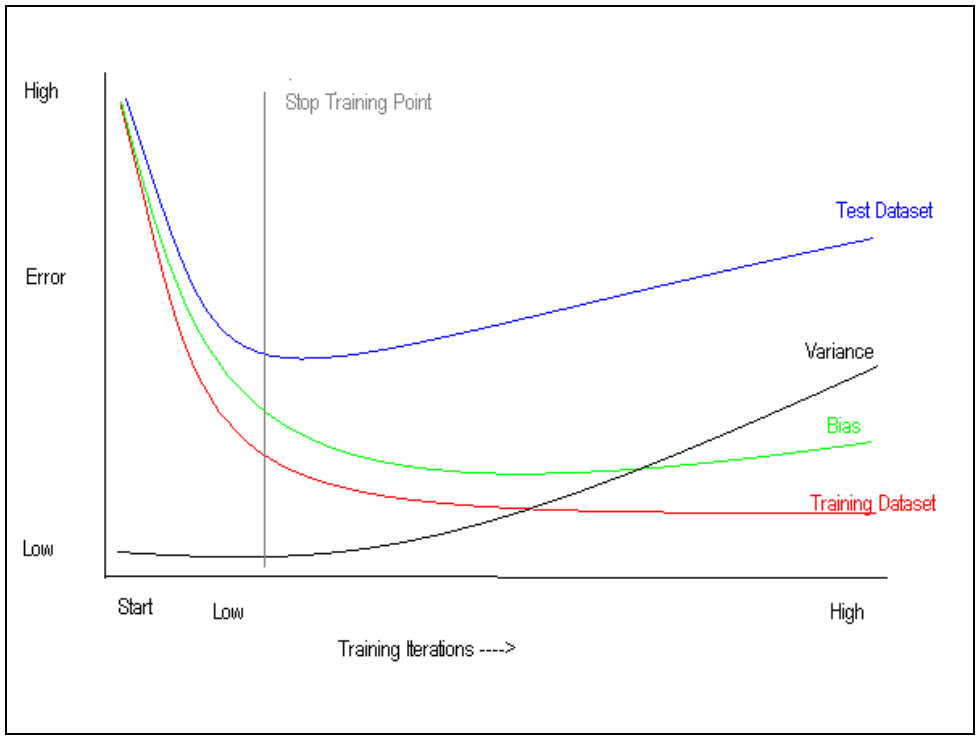


Figure 2.7 The relationships between Mean Square Error measured on a Training Dataset, Mean Square Error measured on a Test Dataset, Bias and Variance, as a function of the progress of training through successive iterations of a training algorithm such as Backprop. (Adapted from Gemen et al [1992]).

Using the above formula, and applying it to a dataset of 1200 handwritten digits (600 for training datasets, 600 for the test dataset), Geman et al [1992] demonstrated firstly that the test dataset Mean Square Error (MSE) varies as the sum of bias squared and variance (according to equation 2.1) at any particular point in training, and secondly that as training progresses bias decreases and variance increases in such a way that Test Dataset MSE at first decreases and then begins to rise. As a consequence, there is point in training where Test Dataset MSE is at minimum. Up until this point, decreases in bias

have outweighed increases in variance, so the value of Test Dataset MSE has progressively decreased. Beyond this point, increases in variance outweigh further decreases in bias, and as a result, Test Dataset MSE increases. The point designated "Stop Training Point" in Figure 2.7, is the point at which training should be stopped, in order to obtain a model which generalises the best (Geman et al [1992]). If one stops training at any point either to the left or to the right of this point then the associated models will all generalise less well than the model associated with the point of minimum Test Dataset error.

Geman et al [1992] also demonstrate that bias and variance (and therefore Test Dataset MSE) also vary as a function of model complexity. That is, as complexity increases from low to high, bias and variance behave similarly as they do in response to training proceeding from few to many iterations. As a consequence, Test Dataset MSE also behaves similarly, that is it has a minimum value at some point on the complexity continuum. In the case of MLP type neural networks, the common way in which to adjust model complexity is to vary the number of hidden units. MLPs with fewer hidden units have a lower complexity than those with more hidden units. These relationships are presented in Figure 2.8 below.

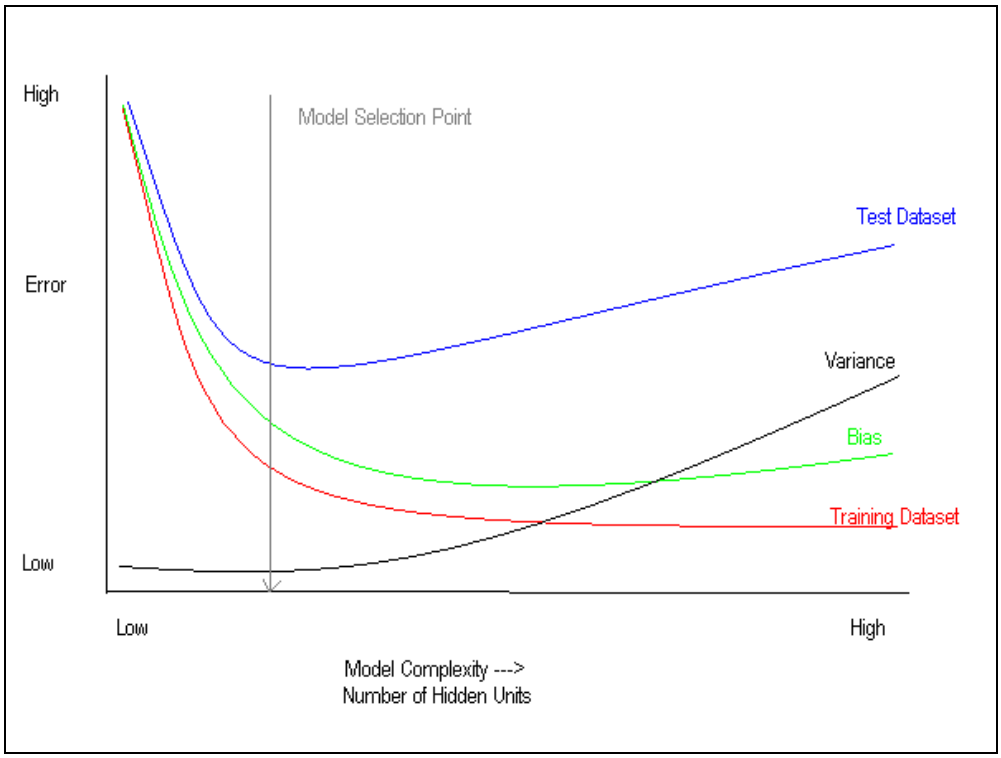


Figure 2.8 The relationships between Mean Square Error measured on a Training Dataset, Mean Square Error measured on a Test Dataset, Bias and Variance, as a function of model complexity, which is adjusted by varying the number of hidden units. (Adapted from Gemen et al [1992] and Hastie et al [2001]).

The relationships and the tradeoff depicted in Figure 2.8 above are very similar to those depicted in Figure 2.8. The main difference is that we have substituted complexity for training iterations as the x-axis of the graph. Complexity is directly related to the number of adjustable parameters in a model. For MLPs, an easy way to vary complexity is to add or remove hidden units. This adds and removes the weights of the connections made by those hidden units, and it is these weights which are the adjustable parameters of the model embodied in an MLP type neural network. The reason for the similarity of

effects between training and complexity is that training is, in a sense, a complexity realisation process. At the start of training, the weights of an MLP are initialised to a set of small (near zero), random values. A zero weight is effectively a non-weight or in the case of a model a non-parameter. In this sense, the MLP at the start of training has a low effective complexity. As training progresses, The MLP weights are updated to minimise error in respect to a Training Dataset. That is they develop values, which depart from zero, and raise the effective complexity of the MLP model. The ceiling on complexity, that a particular MLP model can achieve, in training, is determined by the number of parameters it contains. So MLPs with more parameters (usually determined by having more hidden units), can implement more complex models. As a consequence, we observe similar relationships between bias, variance, and Test Dataset error in relation to both training and complexity. The model selection point, in Figure 2.8, is the minimum point of Test Dataset error. The model at this point on the complexity continuum represents the best model, in terms of generalisation to future cases drawn from the same population.

Figure 2.8 is misleading in one respect: It suggests that the minimum point of Test Dataset Error, the best model, is always at a level of complexity beyond the lowest possible. In the case of neural networks, the least complex model is a linear logistic regression model (i.e. an MLP with no hidden units, where inputs connect directly to an output unit). It is of course possible, in fact common, that the linear model is the best model. In which case progressing to models of any higher complexity causes the increase in variance to exceed the decrease in bias obtained, and hence result in models with inferior performance, which have a greater Test Dataset Error. In such cases the

minimum lies at the beginning of the Test Dataset error curve, at the minimum level of model complexity (a linear model) and this is the also the Model Selection Choice Point.

Figure 2.7, on the other hand is not misleading in the same respect. The same situation would almost never occur in respect of training (unless, the data was generated by a random process), a newly initialised model would have a very high bias, which would drop rapidly in the initial few iterations of training. At the same time, in these initial few iterations, the variance would begin to rise, but the rate of rise would at first be slow. As a result, Test Dataset Error is likely to initially fall, reach a minimum value and then rise once the decrease in bias slows and the increase in variance quickens.

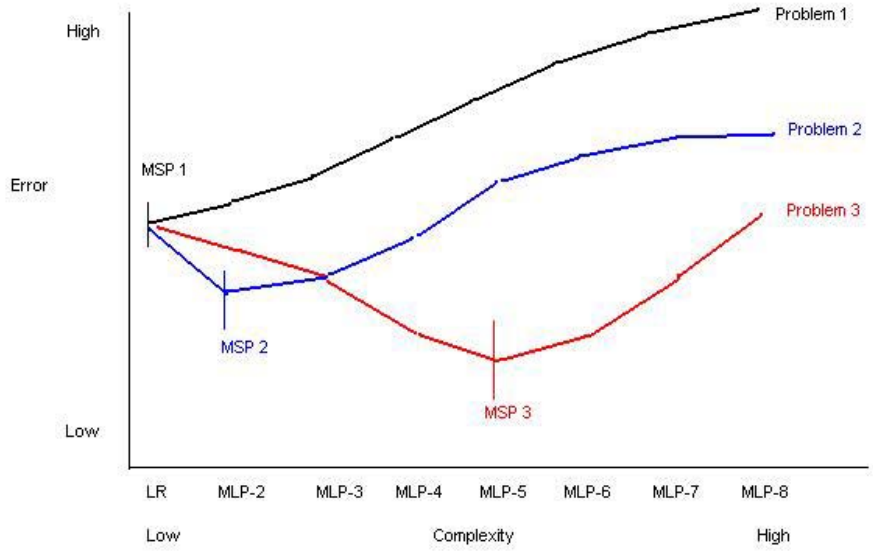


Figure 2.9 Test dataset MSE by Complexity curves for 3 hypothetical classification problem explored using an LR model and seven MLP models of varying complexity (varied by adjusting the number of hidden units in the MLP model)

Figure 2.9 above presents 3 hypothetical classification problems explored using an LR model and several MLP models of varying complexity. The model selection point (MSP1) for Problem 1 is an LR model, which is the least complex of the models explored. For Problem 1 the amount of decrease in bias, which results from the increased complexity of an MLP-2 (MLP with 2 hidden units) model, is outweighed by the associated increase in variance. For Problem 2, the MLP-2 model has a lower MSE than the LR model indicating that in this problem the reduction in bias outweighs the increase in variance associated with increasing the model complexity by moving from an LR model to an MLP-2 model. However, increasing the complexity of the MLP model beyond 2 hidden units results in increases in variance which outweigh their associated reductions in bias. For Problem 3, decreases in bias which outweigh increases in variance occur for increasing levels of model complexity up until an MLP-5 model, with the converse applying after that.

In practice, we can use our knowledge of the possible situations depicted in Figure 2.9 to search for a good model amongst a range alternatives. We do this by measuring the performance of each model on a large independent test dataset, arranging these results on a complexity continuum and then selecting a model which seems lies at a turning point of the performance curve. This will be a minimum or a maximum depending upon how performance is measured, in terms of error or accuracy.

Implications for the modeling of clinical decision making problems

The general aim in developing a good classification model, for a clinical decision making problem, is to obtain a model, which has the lowest possible test dataset error. Since test dataset error is a simple additive function of bias and variance, then ideally what we need to do is to attempt to arrive at minimal values for both. However, as shown in Figures 2.8 and 2.9, it often the case that an action which decreases the value of one, will also increase the value of the other. Geman et al [1992] liken this to the *uncertainty principle* in quantum physics. Thus as iterative training progresses bias decreases and variance increases. Similarly, if we increase the complexity of an MLP type neural network model by increasing the number of hidden units, then again there will be a decrease in bias accompanied by an increase in variance.

Importantly though, the growth and decay curves of variance and bias with respect to training and complexity are problem and training dataset dependent. The crucial factors, which determine the location of the minimum on the test dataset error curve, are the rates of change of bias and variance. The test dataset error minima, is located at the point where the rate of increase in variance begins to exceed the rate of decrease in bias.

One strategy for attempting to reduce test dataset error is to increase the sample size of the training dataset. Doing this will shift the variance growth curve to the right with respect to both training and complexity. This is because each trained model is now more likely to be similar to the average of that model (the central limit theorem predicts this behaviour). A consequence of this is that a lower bias value may be reached before the cross-over of rates of change of the curves is encountered and also the cross-over point

may correspond to lower values of both bias and variance. Added to this more data points will better represent finer details of the features of the target function (if they exist), allowing a sufficiently complex model to shadow the population target function better, that is reduce bias. Thus, test dataset error (the sum of bias and variance) is likely to be lower for a training dataset of increased sample size.

In figure 2.9, the model complexity of an MLP is varied by adjusting the number of hidden units, which in turn adjusts the number of model parameters (MLP Weights). However, there are other schemes for adjusting the effective complexity of the model embodied in an MLP type neural network. We have already pointed out that stopping training early, at a point where generalisation appears to be maximised, is a way of reducing the effective complexity of the model. At this point, the weights have been adjusted more under the influence of bias, than under the influence of variance. Some of the weights are still relatively close to their near zero initialisation values, and are, in some sense, non-parameters in terms of the model. This reduces the effective complexity of the model to that of a model with fewer parameters.

Another scheme is to implement regularisation as part of the algorithm, used in training, to update the weights. This is commonly known as *Weight Decay*. In such a procedure, a proportionate amount of the value of each weight is subtracted from the value of the weight, after each weight update. This introduces a tendency into the training algorithm for weights to reduce in magnitude, as training progresses. The end result is that weights which do not grow in magnitude, in response to the Training Dataset, as training progresses, will tend towards zero. Thus, unneeded weights are systematically

eliminated (zeroed) by training and the effective number of model parameters (the effective complexity) is lowered.

There several important conclusions about the application of neural networks (and other statistical classifiers) that can be drawn from consideration of the bias–variance tradeoff. Firstly the general nature of the relationship between bias and variance in statistical models for classification based upon MLP type neural networks is one in which both contribute to generalisation error. At the level of an individual MLP model, trained on a fixed size training dataset, a decrease in one will result in an increase in the other. In practice this means that generalisation error can never reach the absolute Bayesian minimum (which in itself introduces a irreducible level of error), for a particular problem, as it will be added to by an amount of error, dependent upon the amounts of bias and variance. In some cases, the size of this additional error created by bias and variance may be relatively large.

Secondly, improvement in the generalisation accuracy of a classifier, under development, may be obtained by:

- Increasing the training dataset sample size.
- Using the minimum turning point value obtained by measuring error on a test dataset as a criteria to stop training and/or choosing a complexity level
- Use a regularisation scheme, such as weigh decay which reduces the effective complexity of the model.

Thirdly, the training dataset size, the point at which we stop training, model complexity and input dimensionality are, in some sense, an interrelated set of hyperparameters that, in combination, define a manifold surface with respect to generalisation error, in a solution space, not dissimilar to the way in which the weights of an MLP define an error surface. As such, this surface is likely to have regions of minima which define 'good solutions'. Underlying this error surface in this hyperparameter defined solution space, there would be bias and variance surfaces, which directly determine the shape of the overlying error surface. Because all these hyperparameters trade-off against each other, then the 'good solutions' regions will tend to lie around the origin of the space. As such, one would not consider a combination of small training dataset size, high dimensionality, high complexity and training continuing until the training error reached a minimum, as a path to a good solution. The hyperparameters need to traded-off in some fashion as part of the model search strategy.

Finally, the differences between models being compared can be in kind as well as in complexity. Different kinds of models may exhibit different levels of bias at the same level of complexity. Variance on the other hand is more directly related to a model's complexity (or the number of parameters of a model to be more exact). Thus one kind of model might perform better than another, on a particular dataset, though both are equally complex, because in relation to the problem at hand it exhibits less bias and therefore less error. It has been a common finding of reviews (e.g. Ripley 1994, 1996, Michie et al 1994) which have applied a number classification models to a panel of datasets that some methods do better on some datasets but no method comes out best. Ripley [1996] describes this phenomena as *"every method has its day"*.

2.8 Studies comparing Neural Networks with Statistical Methods in Medical Decision Making Problems

There are now a large and increasing number of studies in the medical literature, which use clinical datasets, and which have compared the performance, as classifiers, of a neural network(s) to logistic and other regression models [Grant et al 2001, Sargent 2001]. In some studies, a neural network model has been found to classify better than a regression model, in some other studies the two are found to be equivalent, and in a small number of studies a regression model classifies better than a neural network model. On face value this would seem to agree with the conclusion of our consideration of the bias-variance tradeoff: that whether a particular model classifies better, the same or worse on a dataset than another model depends upon how both these models tradeoff bias and variance on the dataset. In such a situation we might conclude that all the studies which found in favour of a neural network are examples of datasets where the bias-variance tradeoff was better for the neural network model than for the regression model. However this is not necessarily the case. Publication bias might account for some or all these studies. Therefore a more detailed review of this literature, which takes publication bias into account, is warranted to elicit a clearer understanding of what has led to the observed pattern of results.

In reviewing this or any similar literature we need to take account of the possible effects of publication biases in determining the availability of studies for inclusion in a review. A simplistic review would assume we have full access to all studies (or at least a representative sample) which have comparatively investigated neural networks and logistic regression models. In such a situation, when multiple studies investigate the same general hypothesis, we would tend to conclude that the hypothesis is true if the

number of studies finding in favour of the hypothesis, far exceeds the number expected from the application of the type I error rates. On the other hand, we would tend to conclude the hypothesis is false if only a small number of studies, as predicted by the effects of type I errors, find in favour of the hypothesis.

However, in the real world, there are filters which prevent the publication of some studies and boost the publication of others. Firstly, journals, their editors and reviewers, are more likely to reject papers with negative findings because, they don't highlight something new and the readership is less likely to be interested in reading about negative findings. Secondly investigators, pre-empting this bias of journal editorial decisions, might decide to conserve their effort and not write up and submit studies which have a negative finding. Both these biases also work in reverse. That is investigators are more likely (in fact almost certain) to write up and submit studies with a positive finding and journals are more likely to publish these studies.

Studies with a positive finding, in favour of a hypothesis, can arise in two ways. Firstly they can occur, in very large proportion, when that hypothesis is true. Secondly they can occur by chance and in very small proportion, when that hypothesis is not true (type I error). As such it is possible that when there is strong publication bias, most of the spuriously significant studies are published, and most of the studies involving the same hypothesis, but which have a null finding, are not published. This raises the apparent proportion of the studies that support the hypothesis. This inturn would give an appearance in the literature that the hypothesis is true, at least sometimes, when fact it is not ever. *Publication bias*, of this kind, is a well recognised problem in the medical literature [Begg & Berlin, 1988]

Sargent [2001] examines the hypothesis that MLP type neural networks can in some circumstances classify better than a logistic regression, by analysing a set of 28 selected studies from the medical literature. The inclusion criteria for studies in his review were that: the study compared a neural network with a Logistic Regression or Cox Regression in a clinical application; the sample size was greater than 200; and the comparison was made on the basis of independent test dataset error or an equivalent technique (that assesses model performance on data that was not used in training or in model development). He found that the neural network outperformed the regression in 10 studies (36%), that regression outperformed the neural network in 4 studies (14%) and, that they had equivalent performance in 14 studies (50%). The sample sizes of the studies varied from 226 to 80,600, with a median around 1,000. However, all the studies which found in favour of the neural network had sample sizes that were below the median. Of the 14 with an 'equivalent' finding, 11 were above median. Of the 4 with a finding for regression 2 were above median and 2 below. Sargent [2001] interprets this set of results as suggesting the possible operation of a publication bias in favour of neural networks, and concludes that "*Both methods should continue to be used and explored in a complementary manner. However, based on the available data, ANN should not replace standard statistical approaches as the method of choice for the classification of medical data"* (p. 1636)

A set of 28 studies is small, and does not allow conclusions any firmer than those reached by Sargent [2001]. A larger set of studies may provide a firmer basis for evaluating the strength and pervasiveness of a publication bias as well as the potential role, if any, of neural networks for classification problems in medicine.

Using the same Medline (Ovid) search strategy and the same inclusion criteria as Sargent [2001], another 21 studies, published subsequently, were located. These are listed in Table 2.4, in a similar format as that used by Sargent [2001] in his Table 1. The difference being that in our Table 2.4, we have listed the actual size of the training dataset sample and the Validation dataset sample, to facilitate a later analysis of these data, whereas Sargent [2001] lists the size of total sample and the % split used derive training and validation datasets. Of course, these formats are interchangeable with simple calculations.

In Table 2.4 below, there are 5 studies, with training datasets above the median in size, which have a finding in favour of the neural network. There are also another 6 studies, with training dataset samples sizes above 1000 which have an 'Equivalent' or 'Regression' finding as well as ten studies with training dataset sample sizes below 1000, which find mostly (except for one 'Equivalent') in favour of the neural network.

Citation	Regression	ANN	Training Sample N	Validation Sample N	Result
Snow et al [2001]	LR	BP	28,125	9,375	NN
Colombet et al [2000]	LR	BP	10,296	5,148	EQUIV
Li et al [2000]	LR	BP	9,480	3,160	NN
Di Russo et al [2000]	LR	BP	5,768	4,841	NN
Han et al [2001]	LR	BP	4,308	1,436	NN
Resnic et al [2001]	LR	BP	2,804	1,460	EQUIV
Kalra et al [2003]	LR	BP	1960	1308	NN
Freeman et al [2000]	LR	BP	1,554	1,465	REGR
Wang et al [2001]	LR	BP	1,253	500	EQUIV
Clermont et al [2002]	LR	BP	1,200	447	EQUIV
Finne et al [2004]	LR	BP	1183	592	REGR
Finne et al [2000]	LR	BP	656	Leave one out	NN
Veltri et al [2000]	LR	BP	636	120	NN
Ioannidis et al [2003]	LR	BP	504	284	NN
Orr [2001]	LR	BP	490	798	NN
Kim et al [2000]	LR	BP	409	183	NN
Verive et al [2000]	MR	BP	394	69	NN
Samli & Dogan [2004]	LR	BP	230	73	NN
Mello et al [2001]	LR	BP	187	116	EQUIV
Eldar et al [2002]	LR	BP	180	45	NN
Zlotta et al [2003]	LR	BP	140	60	NN

Table 2.4 Summary information for 21 additional articles published since Sargent [2001] which meet, his inclusion criteria.

Figure 2.10, below, plots findings from all the studies on a graph similar to that presented by Sargent [2001]. Twenty one new studies have been added to Sargent's [2001] original plot of 28 studies. For clarity Sargent's [2001] original 28 studies are represented as open circles and the 21 new studies added in the current review are represented as closed circles. Also for clarity the vertical axis on each graph, which quantifies dataset sample size, is on a logarithmic scale. Graph a) displays the distributions of training dataset sample size for the three types of outcomes found by the

49 studies, and graph b) similarly displays distributions of validation dataset sample size by study outcome.

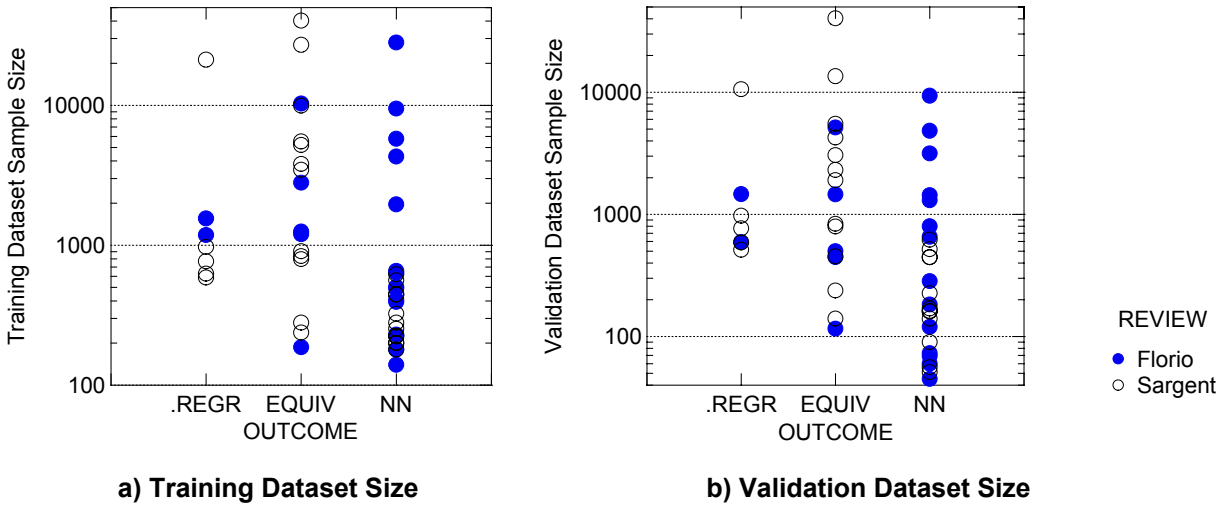


Figure 2.10 Distributions of study dataset sample sizes, according to study outcome for the combined set (Sargent's [2001] review of 28, plus 21 new studies), broken down by review source

Looking only at Sargent's [2001] set (open circles) in Figure 2.10, all the studies which found a neural network to classify better than a regression are below the median and have relatively small training dataset sample sizes. By contrast the distributions of training dataset sample sizes of studies in the "Equivalent" and "Regression better" outcome categories are more centred on the median and for the "Equivalent" category they straddle most of the range. This "funnel" appearance, in this plot, suggests the operation of a publication bias in favour of neural networks [Sargent 2001].

However, looking at full set of 49 studies (combining Sargent's 28 studies with the 21 new studies identified in this review), a different picture emerges. The "funnel" appearance of the plot, noted by Sargent [2001], is no longer as apparent. This is because there are now 5 newly added studies, which have relatively large training dataset and validation dataset sample sizes (over 1000) and which have a finding in favour of a neural network model in comparison with a regression model. The training dataset sample size distributions of the three outcome categories, using all 49 studies, now suggest a mixed picture. There is a preponderance of studies with small training dataset sample size with an outcome in favour of a neural network. But, clearly in some clinical applications (more specifically in some clinical datasets), with relatively large training dataset sample sizes a neural network based model was able to classify better than a regression model.

If we restrict ourselves to looking only at studies with a training dataset size of over 1,000 cases (19 studies). Four studies found in favour of an LR model, 5 found in favour of an MLP model and 10 found an "equivalent" result. Given the associated large test datasets, upon which differences between models were significance tested, it is unlikely that this distribution of findings is due to a publication bias. The alternate explanation is that the distribution of findings is due to differences in bias-variance tradeoffs between models when applied to the various datasets. In other words, the MLP model classified better than the LR model on a proportion of these large datasets.

The findings of our extension of Sargent's [2001] review are consistent with a hypothesis that in some cases (datasets) a neural network model is able to classify better

than a regression model, but that there is also a superimposed publication bias operating, in this literature, in favour of studies which find a neural network model to classify better than a regression model and this leads to an overabundance of studies with relatively small sample sizes and an outcome favouring neural networks. Some or many of these studies may have a spuriously significant finding.

Sargent's [2001] review, and our extension of it, narrowly selected only comparative studies and only those with a relatively good methodology. The empirical literature on the application of Neural Networks to Clinical Decision-Making, in general, is of much poorer average quality. It is highly disjointed. Many studies are one-offs which do not refer to or build upon other studies. The overwhelmingly typical template for studies involves the application of a neural network to a relatively small dataset, sometimes with a comparison to a traditional statistical technique (such as Logistic Regression), and only in a small proportion is some form of cross-validation used. The choice of predictors and criterion variables is often idiosyncratic, so that even in the same clinical decision making problem domain there is a great deal of variety. This makes it hard to compare results between studies or to perform any kind of systematic review or meta-analysis on a problem-wide basis. There are few threads in this literature. Most studies fail to consider the larger literatures which exist on Clinical Decision-Making and on Discrimination and Classification in Statistics. The generally poor quality of this literature is highlighted by Schwarzer et al [2000], who provide an informative review on misuses of the neural networks in oncology

Following Sargent's [2001] example, we have been able sidestep the poor quality of the bulk of the literature by selecting only studies with a good comparative methodology and by focusing our review only on the single issue of a direct comparison between MLPs and LR (mostly) on medical datasets. Our conclusion, based on this review, is that despite the apparent operation of a publication bias in favour of neural networks, there is also evidence that in some clinical decision making problems (datasets), a particular neural network model offered better classification than a particular logistic regression model. Our earlier consideration of the bias-variance tradeoff, pre-empts a finding of this nature because it predicts that an MLP model may in some datasets offer a better bias-variance tradeoff than does an LR model.

2.9 Conclusions

Neural networks are a new type of computer system, inspired by the functioning of neurons in the brain and CNS. They are particularly suited for the development of applications that rely upon pattern recognition or pattern categorisation. These are the kinds of problems that traditional techniques have been unable to satisfactorily address. Neural Networks have been successfully applied to range of applications, such as speech recognition and handwritten postcode digit recognition. There is a growing interest in applications using neural networks in clinical decision-making problems in medicine, with some systems such as PAPNET becoming widely used.

Psychiatry contains many clinical decision making problems which have not been satisfactorily solved and which are good candidate applications for using neural

networks. The use of clinical judgement has been found to have significant limitations. Structured decision-making overcomes some of these, but we also know that it is not as effective as statistical decision-making. Adoption of the latter by clinicians has been very slow.

The advent of neurocomputational decision-making provides a new alternative that has most of the features of statistical decision-making. Both statistical decision-making and neurocomputational decision-making are empirically based. The key difference is that neural networks are able to exploit non-linear relationships in data, which traditional linear statistical techniques do not. In terms of the classification schema outlined in Table 2.3 (Decision Making Practices by Clinicians), Neurocomputational should conceptually be considered to be a type of statistical decision-making.

The bias-variance tradeoff, is a theoretical framework that can be broadly applied to classification and regression models, such as neural networks and logistic regression. This framework can be used to derive a set of conditions for neural network models to classify better than a logistic regression models, but it also suggests that in practice these conditions may be difficult to obtain.

A review of studies comparing neural networks with the statistical technique of logistic regression for classification of medical datasets found that even though there seems to be a pervasive publication bias in favour of neural networks, there also seems to be some cases where the neural network classified better.

Therefore, neural networks have potential to better address certain classes of clinical decision making problems in psychiatry. However this has yet to be demonstrated empirically. Furthermore, due to a lack of experimentation with, and application of, neural networks to psychiatric clinical decision-making, little is known about issues of practical application of neural networks to psychiatric clinical decision-making.