CHAPTER IV

APPLICATION OF ARTIFICIAL NEURAL NETWORK IN DISCRIMINANT AND REGRESSION ANALYSIS

4.1 INTRODUCTION

In previous two Chapters, we developed and discussed the theory associated with single layer and multilayer ANN models. We also discussed how these models are useful in certain types of pattern recognition and classification problems. Now, the purpose of the present Chapter is to discuss how based on the theory presented in Chapter II and Chapter III, ANN models offer an entirely novel Statistical data analysis tools.

First, in Section two, we are concerned with an application of ANN in Discriminant analysis. Since ANN models are extremely powerful in pattern recognition problems, it is but natural to exploit their use in Non-linear Regression problems. In Section three, we devote our discussion to this aspect. We compare the performance of regression models- linear as well as non-linear models with that of neural networks with the help of simulated and live examples. We conclude this Chapter by discussing, when it is advantageous to use this type of model in place of regression model.

4.2 ANN AND DISCRIMINANT ANALYSIS

The problem of discriminant analysis is as follows : Suppose there are two distinct p-variate populations Π_1 and Π_2. The problem of discrimination is to decide whether an individual belongs to Π_1 or Π_2 based on the basis of a p-component vector of variables $\underline{x}$. The process of obtaining a solution to this problem proceeds on the following lines :

Let R denote the entire p-dimensional space in which an observation $\underline{x}$ falls. Divide this region R into two distinct regions say R_1 and R_2 ($R = R_1 \cup R_2$).

Then a rule or procedure such as

If $\underline{x} \in R_1$, assign an individual to Π_1

and

If $\underline{x} \in R_2$, assign an individual to Π_2

is called a 'discrimination' or 'classification' rule.
In order to obtain an optimal discrimination rule, it is assumed that (Kshirsagar, 1972) X and Y are p X n_1 and p X n_2 matrices of the sample observations from Π_1 which is $N_p(\underline{\mu}_1, \Sigma)$ population, and Π_2, which is $N_p(\underline{\mu}_2, \Sigma)$ population respectively. Then the Fisher's best linear discriminant function is given by $\underline{l}'\underline{x}$ and the classification rule is :

assign the individual with measurements $\underline{x}$ to Π_1 or Π_2 according as

$$\underline{l}'\underline{x} - \frac{1}{2}\underline{l}'(\bar{\underline{x}} + \bar{\underline{y}}) \geq 0 \quad \text{or} \quad < 0 \qquad (4.2.1)$$

where $\underline{l} = f\, s^{-1}\underline{d}$, $\underline{d} = \bar{\underline{x}} - \bar{\underline{y}}$ and s = SSSP matrix based on f degrees of freedom ($f = n_1 + n_2 - 2$).

At this stage, we draw the similarity between $\underline{l}'\underline{x}$ and ANN modelling. For this, note that $\underline{l}'\underline{x}$ is a 'linear classifier'. Further, one can immediately observe the similarity between (4.2.1) and the rule given in (3.2.2) namely

$$\underline{w}'\underline{x}_i = \begin{cases} > 0, & \underline{x}_i \in \Pi_1 \\ < 0, & \underline{x}_i \in \Pi_2 \end{cases} \qquad (4.2.2)$$

We recall that the theorem which we have proved in Section (3.2) with such a rule, states that a single layer ANN with the learning procedure given in Section (2.3) always correctly classifies the pattern $\underline{x}$ to appropriate class.

This fact indicates that a single layer ANN can therefore be used for discrimination purpose as an alternative tool to Fisher's discriminant function. Below we demonstrate the use of ANN in such situations.

Example 4.1 : Consider the data collected by Fisher (1936) (reported in Kendall (1980)) in his classical paper on discrimination. Data contains two different species of flowers namely: Iris Setosa and Iris Versicolor, distinguishable on the four variables (petal length, petal width, sepal length and sepal

width).

Let $\underline{x}$ denote the vector of variables then the discriminant function for these variables is $\underline{l}'\underline{x}$ where

$$\underline{l} = (-3.0692, -18.006, 21.7641, 30.7549)'$$

Then using the above rule we observe that all the 50 observed patterns $\underline{x}$ are correctly classified into appropriate classes. Hence, the observed error of misclassification is zero.

Now, consider the single layer ANN model with sigmoid activation function (2.3.4) presented in Fig. 4.1

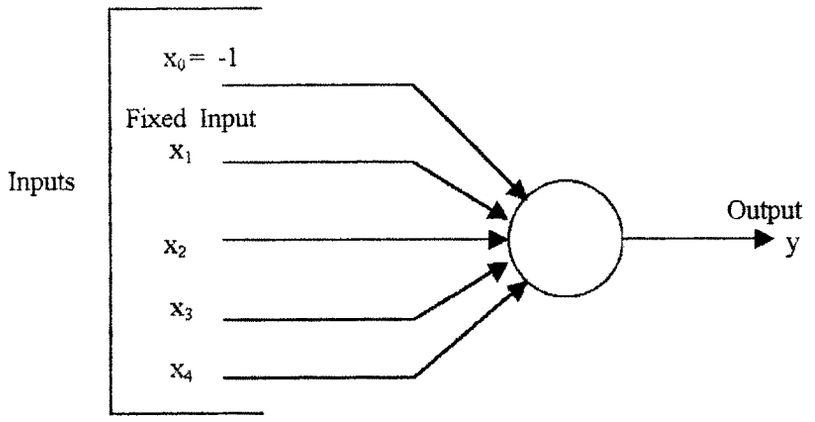

Figure 4.1 Single Layer ANN Model for Fisher's Data.

Here, Iris Setosa and Iris Versicolor are two populations corresponding to class Π_1 and Π_2 respectively. For training the ANN, we have selected 25 input vectors from Π_1 and 25 from Π_2. Further, the vectors are normalized, since it is a requirement

for implementing the ANN effectively. With this set, the network is trained using the training rule given in Section (2.3). The weights are obtained (by using program 'SANN' enclosed in Appendix B) as follows :

$$\hat{\underline{w}} = \left[-2.431,\ -4.85,\ 3.18,\ 5.56,-7.251 \right]'$$

Thus, we have ANN discriminatory rule as

$$\text{If } \hat{\underline{w}}'\underline{x} > 0.5 \text{ then } \underline{x} \in \Pi_1$$

$$\text{If } \hat{\underline{w}}'\underline{x} < 0.5 \text{ then } \underline{x} \in \Pi_2$$

Now, to check the validity of trained network, we have selected remaining 25 observations from Π_1 and 25 from Π_2. For this set, we observed that, these observations were also correctly classified into appropriate classes.

For this particular example, we see that, by using both neural network and discriminant analysis, the observations are correctly classified. However, we note that for using ANN models, we do not need any assumptions such as Normal distribution, equality of two covariance matrices etc. And, this is an important advantage over the traditional method.

4.2.1 Use of ANN As a Linear Classifier In More Than Two Population Problems

Below, we demonstrate how ANN model can be used for a classification problem, when there are several populations. As

is well-known, when there are more than two classes, an elegant solution like Fisher's discriminant function in the case of two populations, is not possible (Kshirsagar, 1972, pp 354). However, several statistical procedures for this problem exist (Anderson, 1958). Usually, while deriving optimal classification rule, one needs to assume the underlying distributions as multivariate normal for each population and with the same variance-covariance matrix. And, hence alternative methods such as ANN models which do not need such assumptions are preferred. Here, we discuss how a simple single layer ANN can be used in a classification problem, when there are more than two groups.

For the classification problem when there are R populations, we have developed the following modified form of ANN model and it is shown in Fig. 4.2

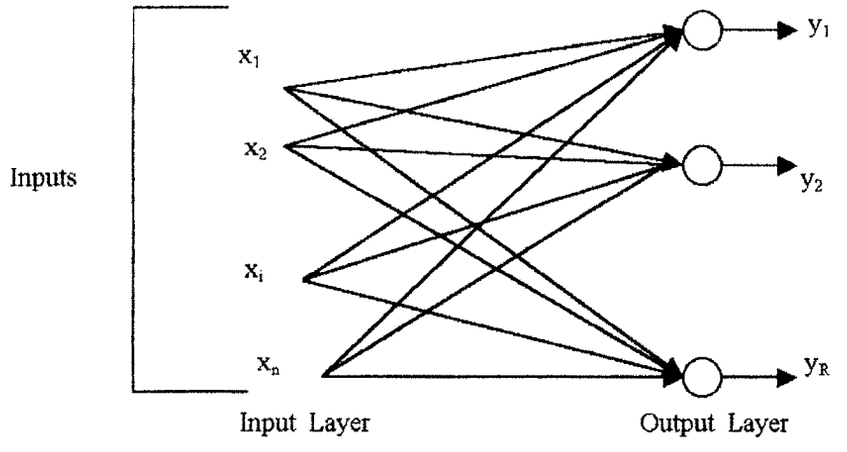

Figure 4.2 Single Layer ANN with R-output Units

Here, $\underline{x} = (x_1, x_2, \ldots, x_n)'$ denotes an n-dimensional input (observation) vector and upon receiving an input $\underline{x}$, the network produces R outputs as shown in the output layer. The above network is trained with the rule given earlier, by using a given training set $\{ (\underline{x}_i, \underline{d}_i), i=1,2,\ldots,P \}$ of P patterns where new $\underline{d}_i = (0,0,\ldots 1,0,\ldots,0)'$ is a desired or target vector containing elements 0 or 1 (1 occurs at j-th position when an observation vector $\underline{x}$ comes from Π_j, j =1,2,...,R, and 0 otherwise).

As an illustration of the above model, we present the following example :

Example 4.2 : Consider Fisher's data on three species (Iris setosa, Iris versicolor and, Iris virginica) with four variables (two species are reported earlier).

Now, consider the single layer ANN model useful for three population classification problem as shown in Fig. 4.3.

For this problem, the desired output $\underline{d}$ for a network in a modified form is taken as follows :

The outputs are represented as (1 0 0), (0 1 0), and (0 0 1) if observation $\underline{x}$ comes from class Π_1, Π_2, and Π_3 respectively. In this example, three populations Π_1, Π_2, and Π_3 are three species of Iris setosa, Iris versicolor, and Iris virginica respectively.

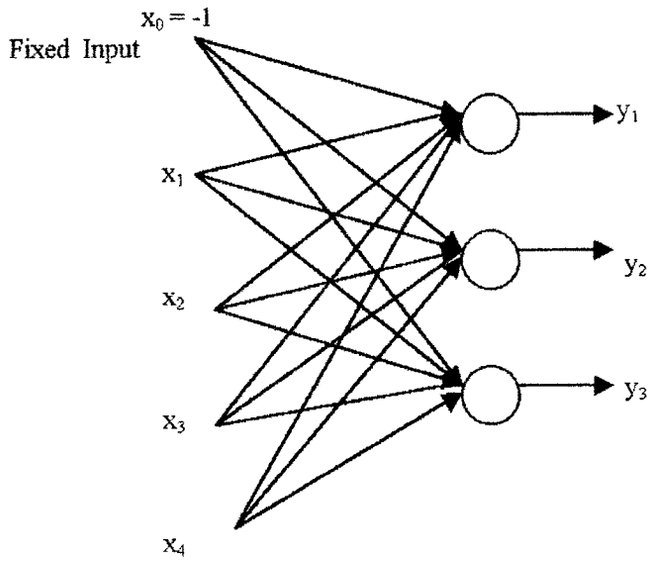

Figure 4.3 ANN Model For Three Population Problem

For training the ANN model in Fig. 4.3, the selected training set contains 75 observations (Data are available in Kendall, 1980) from three populations with their corresponding desired output as given above. With these observations, the above network is trained.

For checking validity of trained network, we used the data set of 50 observations from these three populations. We observed that, observed error of misclassification is almost equal to zero for this data set. This implies that, all selected set of observations are correctly classified into appropriate classes.

NOTE: We obtained results of Example 4.1 and 4.2 after executing program in 'C' language enclosed in Appendix B. Once again we note that the ANN approach works 'efficiently' in the absence of any assumptions.

4.3 ANN AND REGRESSION ANALYSIS

Regression is used to model a relationship between response and stimulus variables. The stimulus (or independent variables) are denoted by x_i. The response (also called outcome or dependent variable) variable is denoted by y. One of the main objectives of regression analysis is to predict response y from the variables x_i.

The general form of regression model is

$$y = f(\underline{x}'\underline{\beta}) \qquad (4.3.1)$$

where

$$\underline{x} = (x_1, x_2, \ldots, x_n)'$$

and

$$\underline{\beta} = (\beta_0, \beta_1, \beta_2, \ldots, \beta_n)'$$

are the vectors of independent variables and parameters (or coefficients) associated with the model respectively, and f(.) is a function which relates $\underline{x}$ to y (to be consistent with our earlier notations, we are using the symbol n for n-covariates rather than the traditional 'k').

As a starting point, in many situations, f(.) is taken to be a linear function of $\underline{x}$. Moreover, we note that if the underlying distribution is Normal, then such a choice of linear function is quite adequate. Under the assumption of linear relationship, the model (4.3.1) becomes

$$\underline{Y} = \underline{x}'\underline{\beta} + \underline{\varepsilon}, \qquad (4.3.2)$$

where $\underline{\varepsilon}$ is a random error component. Eq (4.3.2) is called the 'multiple linear regression' model with n regressors and the parameters β_i's $(i=0,1,2,\ldots,n)$ are called 'regression coefficients'.

Once the model is proposed, the next objective is to estimate the coefficients β_i's. To find the coefficients, we must have a dataset that includes the independent variables and associated known values of the response variables. For finding or estimating these coefficients, different methods are available. One such well-known method is 'Least-Squares method' of estimation. Using this, we estimate the coefficients as follows :

Least-Squares Method of Estimation

Suppose that p > n observations are available and let y_j denote the j^{th} observed response. Let x_{ji} denote the i^{th} observation on regressor x_j. We assume that the error term ε in

the model has mean 0 and constant variance σ^2, and the errors are uncorrelated.

We may write the model corresponding to (4.3.2) as

$$y_j = \beta_0 + \beta_1 x_{j1} + \beta_2 x_{j2} + \ldots + \beta_n x_{jn} + \varepsilon_j$$

$$= \beta_0 + \sum_{i=1}^{n} \beta_i x_{ji} + \varepsilon_j \ , \qquad \text{for } j = 1,2,\ldots,P \tag{4.3.3}$$

The above model can be written in matrix notation as

$$\underline{y} = X\underline{\beta} + \underline{\varepsilon}, \tag{4.3.4}$$

where $\underline{y}$ is a (P X 1) vector of observations, X is (P X m) (where m=n+1) matrix of observations on regressor variables, $\underline{\beta}$ is (mX1) vector of regression coefficients, and $\underline{\varepsilon}$ is a (P X 1) vector of random errors.

Consider the following residual sum of squares E

$$E = S(\beta_0, \beta_1, \beta_2, \ldots, \beta_n) = \sum_{j=1}^{p} \varepsilon_j^2$$

$$= \sum_{j=1}^{p} (y_j - \beta_0 - \sum_{i=1}^{n} \beta_i x_{ji})^2 \ , \tag{4.3.5}$$

Now, we wish to obtain β_i's which minimizes (4.3.5). Equivalently, (4.3.5) can be written as

$$E = (\underline{y} - X\underline{\beta})' (\underline{y} - X\underline{\beta}) \qquad (4.3.6)$$

Then, the problem is to obtain

$$\min_{\beta} (\underline{y} - X\underline{\beta})' (\underline{y} - \underline{X}\beta) \qquad (4.3.7)$$

A solution $\hat{\beta}$ which minimizes (4.3.6) is given by

$$(X'X)\ \hat{\underline{\beta}} = X'\underline{y}$$

A vector $\hat{\beta}$ is called 'Least-Square(LS) estimator' of $\underline{\beta}$. In particular, if $(X'X)^{-1}$ exists, then

$$\hat{\underline{\beta}} = (X'X)^{-1}\ X'\underline{y} \qquad (4.3.8)$$

Using the above LS estimator, the fitted regression model is given by

$$\hat{\underline{y}} = X'\hat{\underline{\beta}}$$

For checking the adequacy of fitted model, several measures are available. There is vast literature available on this topic (Draper, C., and Smith, 1981; Cook and Weisberg, 1982).

4.3.1 Problems Associated With Regression Models

We first note that, in regression models, a functional form is imposed on the data. For instance, in the case of multiple linear regression model, this assumption is that the response is related to a linear combination of the independent variables. Naturally, if this assumption does not hold, it will lead to an

error in prediction. Also, there are some more assumptions like, the error term ε has mean zero and constant variance σ^2, errors are uncorrelated and are normally distributed and so on. The assumption of constant variance is a basic requirement of regression analysis. If error variance is nonconstant, the regression coefficients will have larger standard errors than necessary. Unless these problems are overcome regression models will not be effectively useful.

So quite obviously, if there exists an alternative tool which assumes less but serves the purpose will definitely be preferred over regression model and precisely here for the reason mentioned earlier the ANN models will do this job (Warner and Misra, 1996). Further, as discussed in Chapter III, a two layer feedforward network with sigmoid activation function is a best approximator, for it can approximate any function to any degree of accuracy (Cybenko, 1989). Thus, a neural network is useful when we do not have any idea about the functional relationship between the dependent and independent variables. Below, we discuss how ANN can be used to model the relationship between response and independent variables.

4.3.2 ANN Model For Regression Problem

A regression model with n independent variables discussed

above is similar to a single layer feedforward neural network as shown in Fig. 4.4

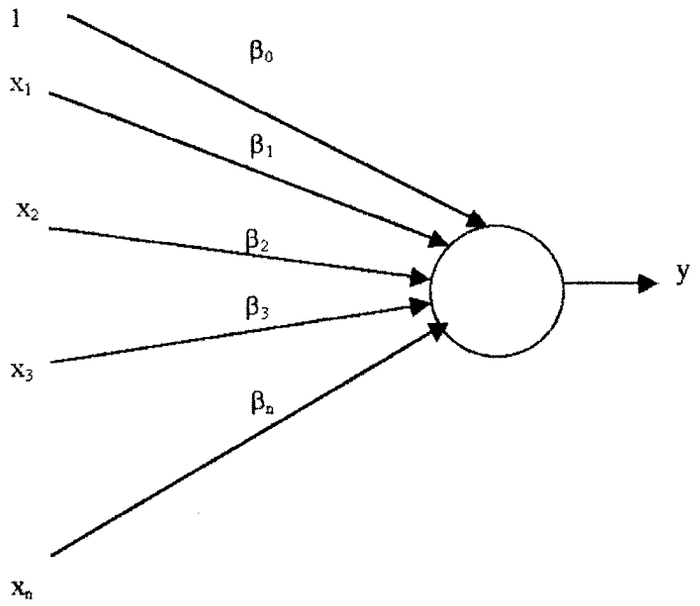

Figure 4.4 ANN model for Regression Analysis

Here, the independent variables x_i's ($i = 1,2,\ldots,n$) correspond to the input of the neural network and y acts as the desired output. The regression coefficients β_i's correspond to weight's w_i's in the neural network. And finding estimates of regression coefficients is similar to estimating weights of ANN. In general, from the theory presented in Chapter II, it follows that any linear regression model can be mapped into an equivalent single layer neural network of the type discussed in Section (2.3) (Warner and Misra, 1996).

In the following, we present some examples to demonstrate the use of ANN as an alternative approach to regression analysis.

Example 4.3 : To compare linear regression model and ANN model, we have generated a data set on (y,x) where y and x are related as follows:

$$y = 10 + 20x \qquad (4.3.9)$$

Fifty random samples were obtained by generating 50 random error terms from a normal distribution with mean 0 and variance 30 and adding these to y values of data set. Here, the values of x were randomly selected from the U(10,100).

First we will fit the linear regression model to the generated data. The model is

$$y = \beta_0 + \beta_1 x + \varepsilon$$

and estimated β_i's (i=0,1) are $\hat{\beta}_0 = -0.5$ and $\hat{\beta}_1 = 20.3$. The fitted line, $\hat{y} = \hat{\beta}_0 + \hat{\beta}_1 x$ alongwith the scatter plot of $\hat{y}$ vs x is shown in Fig. 4.5. As expected the scatter plot, it can be seen clearly that y and x are linearly related and the proposed model is a good fit ($R^2 = 0.98$).

Now, consider a single layer ANN structure as an alternative way of modelling the data. This is done since we know that there is a linear relationship between y and x . To this end, consider the ANN model as given in Fig. 4.4.

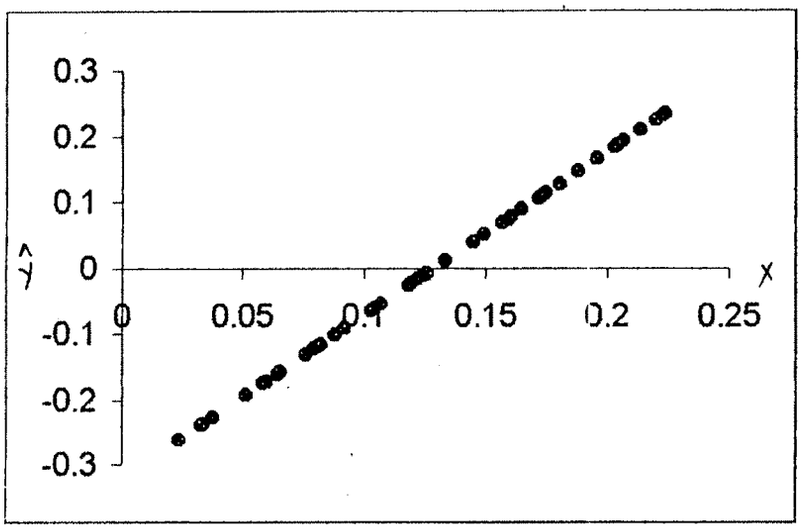

Figure 4.5 Regression Curve

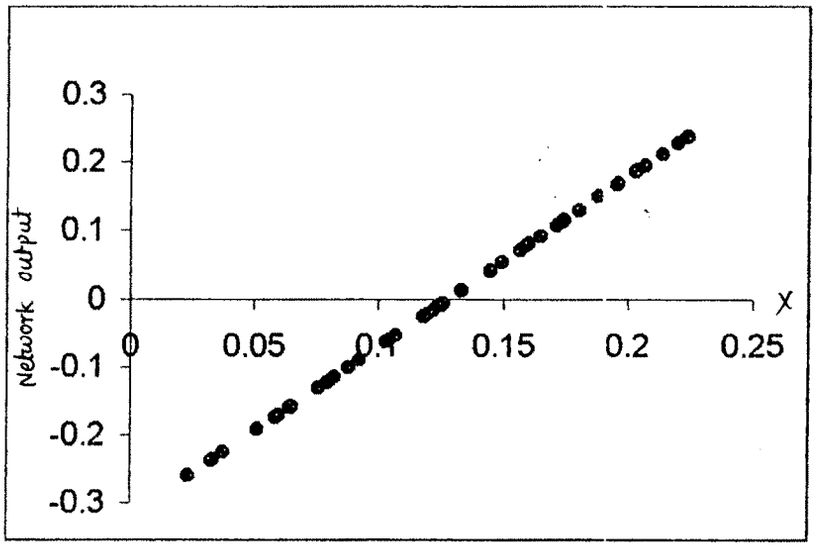

Figure 4.6 Neural Network Output versus x

The above generated data set is used to train the network. For this, the vectors in data set are normalized. For training the above network, we have used training rule discussed in Section 3 of Chapter II, and the software which is enclosed in Appendix B.

After training the network, estimated weights are obtained as follows :

$$\hat{\underline{w}} = \begin{bmatrix} -0.0027, & 2.0251 \end{bmatrix}$$

The Fig. 4.6 shows the results from implementing the neural network (that is, the plot of output of neural network vs x). In Fig. 4.7, the dots represent the regression curve and solid line shows the results from the neural network. This example clearly indicates that, the neural network adequately approximates the linear relationship between y and x.

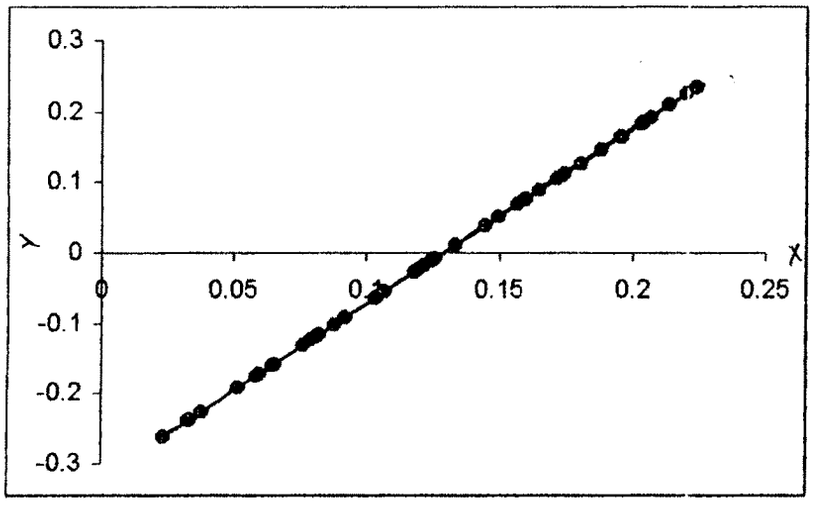

Figure 4.7 Comparison Between Regression and ANN Output

Further, for checking adequacy of the model, a separate set of 100 values on x were generated and applied to the regression model and the trained neural network model. To measure the predictive performance of ANN model, the sum of squared errors $\sum (y_i - \hat{y}_i)^2$ (where y_i's are observed and $\hat{y}_i$'s are predicted values) was computed and these are 0.041 and 0.035 for the regression and neural network model respectively. So the predictive performance of both the models was approximately equal.

NOTE: We note that, in the absence of a priori knowledge of relationship between y and x the above problem of extracting the relationship has to be dealt with a two layer feedforward neural network with more than one hidden units and sigmoid activation function. We have used 4 hidden units and the network was trained using back-propagation training method (discussed in Chapter III). However, since the above problem is linearly separable, a single layer ANN is just sufficient and there is no need to use complex ANN structures.

Non-linear Modelling

The real power of ANN models lies in capturing the non-linear relationship among the variables and as discussed in

Chapter III, the multilayer neural network models are necessary for this type of relationship. Below, we illustrate the same.

Example 4.4: To illustrate how ANN can be used in extracting nonlinear relationship between y and x, we consider the following function

$$y = x^2 \tag{4.3.10}$$

Here, fifty random x values from U(-1,1) were used to generate data. The true curve for above function is shown in Fig. 4.9.

For the above case, two-layer feedforward neural network with five hidden units and sigmoid activation function is used and it is shown in Fig. 4.8

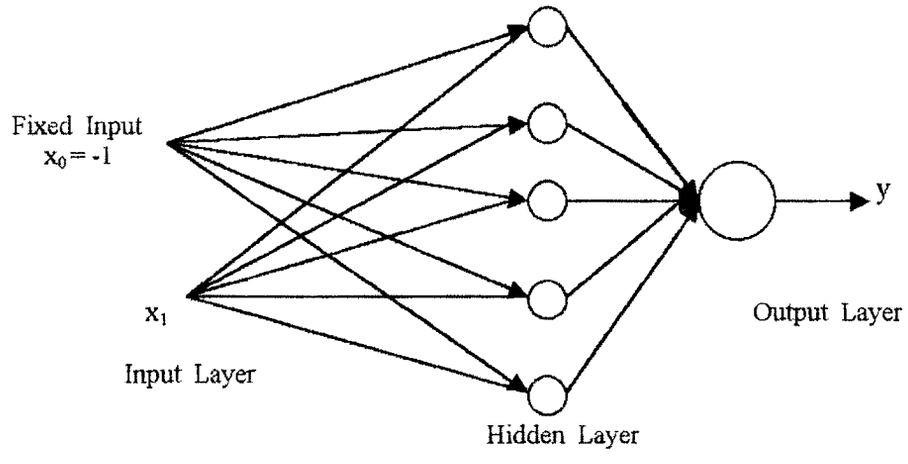

Figure 4.8 Two-Layer Feedforward ANN With Five Hidden Units.

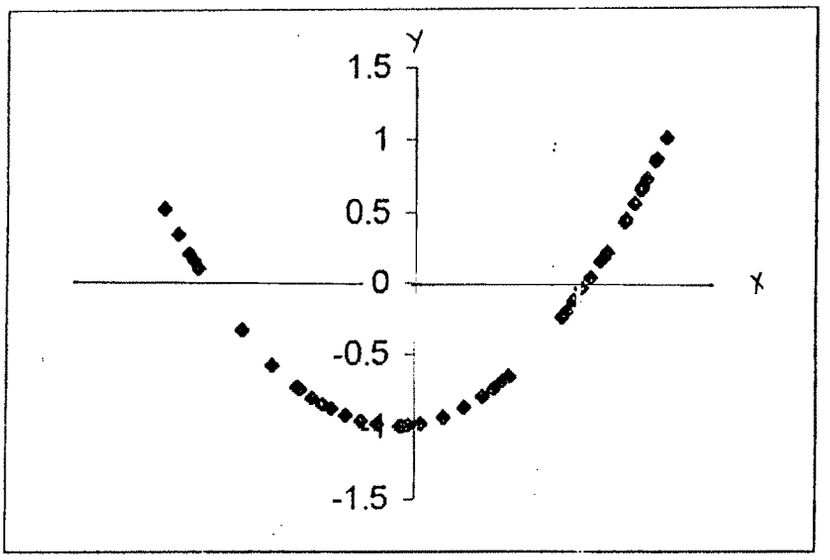

Figure 4.9 True Curve

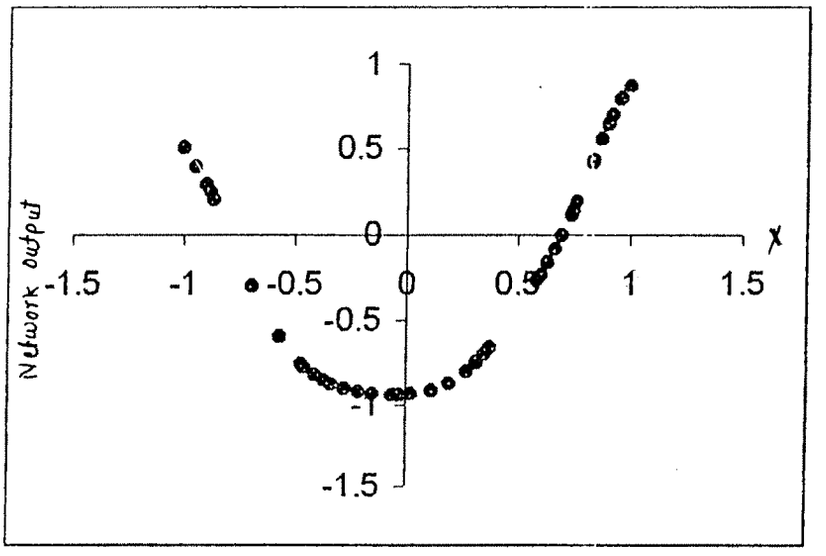

Figure 4.10 Network Output versus x

After training the ANN by using the software given in Appendix C(2), we obtain following weights :

For the sake of simplicity, we present the weights in matrix form. The matrix V denote the weights connecting from input to hidden units are :

$$\hat{v} = \begin{bmatrix} -6.129 & -4.65 & -4.95 & 3.09 & -3.64 & 0.752 \\ 1.986 & 4.812 & -5.45 & 0.22 & -0.183 & 0.861 \end{bmatrix}$$

and the matrix W denote the weights which are connected from hidden units to output units are:

$$\hat{w} = \begin{bmatrix} 2.864 & 2.925 & -3.772 & 1.61 & -1.99 & -5.807 \end{bmatrix}$$

Fig. 4.10 shows the results from implementing the neural network (i.e. plot of actual output of network vs x). From Figs. 4.9 and 4.10, it is observed that the neural network curve and true curve are 'closer' to each other.

Example 4.5: Consider a more complex nonlinear relationship (Warner and Misra, 1996)

$$y = 20 \exp(-8.5\, x)\, [\log(0.9\, x + 0.2) + 1.5] \qquad (4.3.11)$$

Fifty random x values between U(0,1) were used to generate data. A random component consisting of normally distributed error terms with mean 0 and standard deviation of 0.5 were added to each y value. The Fig. 4.12 shows the true curve of above function.

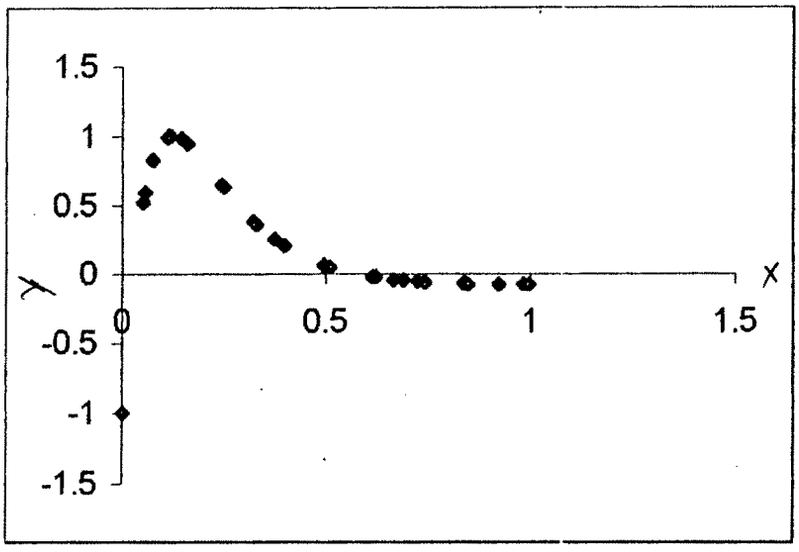

Figure 4.12 True Curve

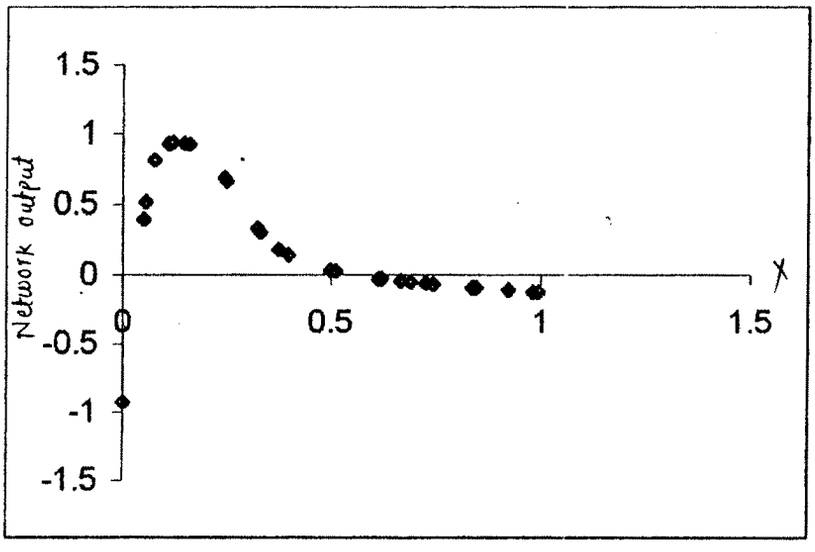

Figure 4.13 Network Output versus x

Clearly, this problem would be difficult to model using regression technique, and would require some techniques of variable transformations. Most of these transformations assume power or logarithmic transformation. But it is easy to understand the functional relationship between independent and dependent variable with the help of ANN, without making any transformation.

Consider the neural network model with eight hidden units presented in Fig. 4.11 and was trained on the above data.

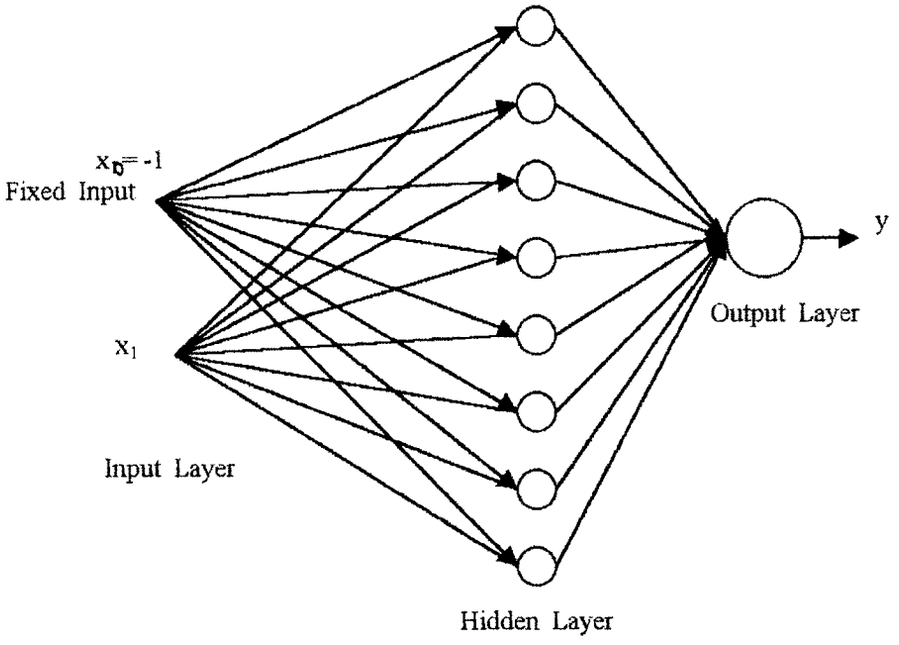

Figure 4.11 MFN With Eight Hidden Units

The result plotted in Fig. 4.13 (i.e. the plot of actual output of network vs x) and it is observed that this curve is 'closer' to curve shown in Fig. 4.12. Thus the ANN model captures the relationship adequately.

In the following we present a live example.

Example 4.6 : Consider the data collected by Rayan, Joiner and Rayan (1976) reported in Cook and Weisberg (1982, pp. 66). The data consists of measurements on the volume V, height H, and diameter D at 4.5 ft. above ground level for a sample of 31 black cherry trees in the Allegheny National Forest, Pennsylvania. The data were collected to provide a basis for determining an easy way of estimating the volume of a tree using its height and diameter. Since the volume of cone or cylinder is not a linear function of diameter, a transformation of volume is likely to result in a fit superior to that provided by the untransformed data.

On the other hand, without making such type of transformation, using the ANNs we obtained similar results (using the original data). For this, we have used two-layer feedforward network with eight hidden units and sigmoid activation function (1.3.7), and network is trained using software enclosed in Appendix C(2).

NOTE : 1. Generally, the knowledge of functional form is unknown a priori and hence, for a given problem, one should always start with a Two-layer Feedforward **ANN** with more than two hidden nodes. Therefore by trial and error one can determine an optimal choice of hidden nodes.

2. In Examples 4.4, 4.5, and 4.6 though we have used ANN with five, eight, and eight hidden nodes respectively, one can always use more than two hidden units.

4.4 CONCLUDING REMARKS

From the above discussion, it can be observed that neural network is indeed useful when one does not have any idea of functional relationship between the dependent and independent variables. Also, this approach requires no distributional assumptions, (precisely for this reason, ANN approach is called a 'Model-free' or 'Distribution-free' approach). If functional relationship between dependent and independent variables is known a priori, obviously better way is to use regression model. Further, we observe that, there are some difficulties associated with neural networks such as choosing the number of hidden units, the learning parameter η, the initial starting weights, the choice of objective function and deciding when to stop training etc. The process of determining appropriate values for

these parameters is often an experimental process where the different values are used and evaluated. And the problem in this process is that it is very time consuming, especially when neural networks are known to have slow convergence rates.

Before concluding this Chapter, we would like to mention that a lot of research is required in this field regarding how to measure the performance of ANN models, what are the effects of 'outliers' on the weights, how many independent variables should be included in the structure and so on, and we have plans to work on these problems.

■