LETTER RECOGNITION USING BACKPROPAGATION ALGORITHM

AZRUL HAFIZ BIN AWANG

A report submitted in partial fulfillment of the
Requirements for the award of the degree of
Bachelor of Computer Science (Software Engineering)

Faculty of Computer Systems & Software Engineering
University Malaysia Pahang

MAY 2010

ABSTRACT

Letter Recognition using Backpropagation Algorithm is an application to analyze the letter data and the performance of Backpropagation Neural Network in letter recognition. The objective of this project is to implement Backpropagation Neural Network for letter recognition task. This application will be developed using C++ Builder. Letter Recognition using Backpropagation Algorithm will make analysis and show the accuracy percentage, errors and the result of the letter training. The expected output will be compared with the actual output. Software Development Life Cycle (SDLC) methodology will be used in this project development to make sure all the development process is running smoothly.

ABSTRAK

Pengecaman Huruf menggunakan algoritma Backpropagation ialah satu aplikasi untuk menganalisis data-data huruf dan mengenal pasti prestasi Backpropagation Neural Network di dalam process pengecaman huruf.Objektif dan tujuan aplikasi ini dibangunkan adalah untuk melaksanakan dan mengaplikasikan Backpropagation Neural Network untuk tugas pengecaman huruf. Aplikasi ini dibangunkan dengan menggunakan perisian C++ Builder. Pengecaman Huruf menggunakan algoritma Backpropagation akan membuat analisis dan menunjukkan peratusan ketepatan, kesilapan dan keputusan dari latihan huruf. Hasil keluaran yang dijangka akan dibandingkan dengan nilai keluaran yang sebenar.Metodologi Software Development Life Cycle (SDLC). Metodologi Software Development Life Cycle (SDLC) digunakan di dalam proses pembangunan projek ini bagi memastikan semua proses pembangunan berjalan dengn lancar.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF APPENDIX

CHAPTER 1

INTRODUCTION

1.0 Introduction

Nowadays, amongst the vast array of topics in intelligent systems is computer vision and comprehension. A particularly important area of computer vision is that of character recognition. Character recognition allows computers go beyond matrices of ones and zeros by gleaning useful information from scanned images of text.

Character recognition systems have enormous potential to solve real problems in a wide range of practical applications such as mail sorting, data compression, and many more. Letter recognition, a sub-topic of character recognition, is developed to recognize the alphabet for training and test, and make analysis and view result and percentage accuracy.

1.1 Problem Statement

One of the most common problems in pattern recognition is identifying letter writing which involves 26 alphabets in English language. Currently, there are many techniques have been implemented in pattern recognition especially Artificial Neural Network.

The performance of Neural Network technique is depends on the selected features and the imputation of values of these features that could help to create the best Neural Network model.

Therefore, in this study, we try to investigate one of the Neural Network's techniques, which is the Backpropagation Neural Network in letter recognition problem.

1.2 Objective

The objectives of this system are to:

i. Investigate the important features in Backpropagation Neural Network algorithm.
ii. Implement Backpropagation Neural Network for letter recognition task.
iii. Develop an application to analyze the letter data and the performance of Backpropagation Neural Network in letter recognition.

1.3 Scope

The scopes of the system are:-

i. Data taken from University of California, Irvine (UCI) machine learning database

Website : http://archive.ics.uci.edu/ml/.

The data include for training data and test data for Letter Recognition System. There have 500 data, which 50 data for testing and 450 data for training.

ii. Backpropagation Neural Network will be used as a technique to recognize the classification of the letter data.

iii. Expected Result: Target Output (Letter) compare with Actual Output (Letter), Analysis of the Result Resting, and Accuracy of Error.

1.4 Organization of the Thesis

This thesis consists of four (4) chapters. Each chapter describe as below:

First chapter is to provide a brief overview of the entire project include objective of the project, scope and problem statement.

Second chapter is to presents the background of the system including the overview of technique that will be used, the existing systems and proposed system.

Third chapter is to discuss of methodology used for the application development. It includes and depict on the system development life cycle including the software as well as the hardware specification for the implementation and development.

Fourth chapter is to conclude the chapter one, chapter two and chapter three of the system that has been discussed before.

CHAPTER 2

LITERATURE REVIEW

2.0 Introduction

This section discusses the introduction to neural network, types of neural network, such as feed forward neural network algorithm, back propagation neural network algorithm. The comparison of the feed forward algorithm and back propagation algorithm are also included together with the explanation of the technique that will be used in this project.

2.1 Introduction To Neural Network

Human brain consists of billions of neurons which is the basic information processing units. These neurons are connected to each other by trillions of connections called synapse between them. Through this complicated and sophisticated network, brain can process information received and sending control to execute reaction to the information given.

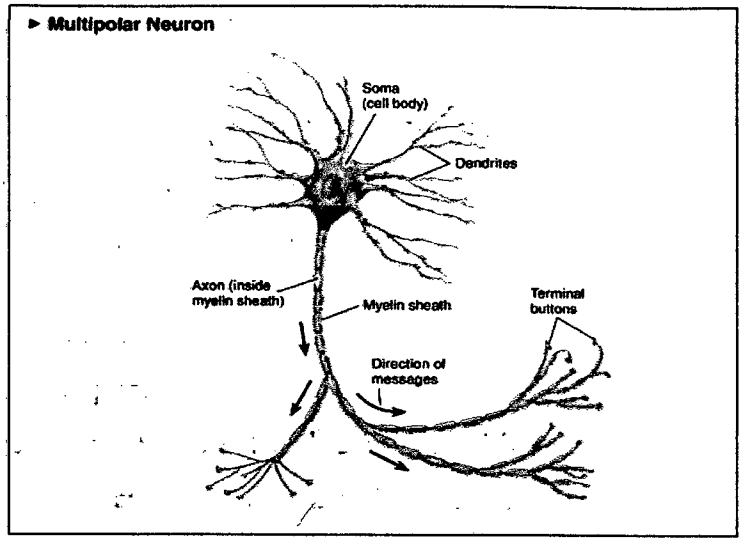

Figure 2.1: Biological Neuron

Figure 2.1 shows the biological neural network in the human brain. (Negnevitsky, 2005)Human brain consists of billions of neurons which is the basic information processing units. These neurons are connected to each other by trillions of connections called synapse between them. Through this complicated and sophisticated network, brain can process information received and sending control to execute reaction to the information given.

There are cell body, soma, a few fibres called dendrites and a single long fibre called the axon. Dendrites make a link with axon to connect somas from other neurons to form a network. Synapses allow electric signal to jump across from neuron to neuron. These electrical signals are then passed across to the soma which performs some operation and sends out its own electrical signal to the axon. The axon then distributes this signal to dendrites. Dendrites carry the signals out to the various synapses, and the cycle repeats. This process is called learning in human brain. (Negnevitsky, 2005)

2.2 Artificial Neural Network

Artificial neural network (ANN) is an information processing paradigm based on human brain. Human information processing structure consists of many processing elements (neuron), communicate to each other to solve certain problems.

Like human, artificial neural network learn through experience. ANN is developed to certain applications such as pattern recognition, or data classification through learning process. Neural network can be used in pattern extraction or sense complicated patterns that cannot be traced by human or any other ordinary computer. Trained neural network will be given knowledge in certain areas to be analyzed. Expertise in these areas will be used to solve a new problem arise in future. (Negnevitsky, 2005)

There are reasons why neural network is important:

i) **Adaptive learning.** Ability to learn and perform a task based on given data or previous experiences.

ii) **Self-organization.** Neural network can build information presentation during learning process.

iii) **Real-time operation.** Operations in neural network are performed simultaneously.

iv) **Fault tolerance via redundant information coding.** Any damage that occurs at certain level of the neural network won't affect the whole operation of the net.

2.3 Types of Neural Network

There are a few types of neural network built by earlier researchers. The selection of which kind of neural network to be used depends on several aspects including learning algorithm selected, and what kind of data used to the network. A

few models of neural network have been introduced since research on neural network had captured the interest of researchers. Regarding on this, modification on the models has been done based on the application needed to be built. In this section, commonly used learning algorithms and models will be discussed. Aside from that, explanation on types of data used also is included. (Negnevitsky, 2005)

2.3.1 Learning Algorithm

Learning implies that processing unit is capable of changing the behavior of input/output as a result of changes that happens in its environment. Since activation algorithm usually determined during development of the neural network, plus input/output cannot be changed, we have to adjust the value of the weights associated with the inputs in order to change their behavior. One method is needed at least during training so that weight can be adjusted based on the process executed to the input/output. A number of learning algorithm designed for the neural network model. In neural network, learning can be either be supervised which is correct output is given during training or unsupervised where no help given. (Negnevitsky, 2005)

Types of learning algorithm:

i) **Supervised Learning.** A specific type of learning algorithm which supervised learning consists of input/output pattern given to the network. The output pattern of the network is compared with a target output pattern. Depending on the difference between these patterns, the net error is computed.

ii) **Unsupervised Learning.** Unsupervised learning algorithms require data set that includes typical input variable values. Target output values are not required. If output values are present in the data set, they are simply ignored.

iii) **Reinforcement Learning.** Algorithm between supervised and unsupervised learning are input and reward signal as output. The learner is not told which actions to take, as in most forms of machine learning, but instead must

discover which actions yield the most reward by trying them. Reinforcement learning less used because it takes long time to learn.

2.3.2 Types of Data

Neural network is different from the types of data used to the network. There are two types of data discussed here: (Negnevitsky, 2005)

i) **Category.** Categorical variables take only a finite (countable) number of possible values, and there are usually several or more cases falling into each category. Categorical variables may have symbolic values that must be encoded into numbers before being given to the network.
Example: male, female, red, white color.

ii) **Quantitative.** Quantitative variables are numerical measurements of some attribute, in such as a way that at least some arithmetic relations among the measurements reflect analogous relations among the attributes of the objects that are measured.
Example: distance in meter.

2.4 Neural Network Model

i) Feedforward Model

Connection between units does not generate a cycle. The journey begins from input to output layer. This model usually produces reaction to the input quickly. Most of the models can be trained using multiple numerical methods. Forward propagation happen when input pattern is put into the network. Every input unit matches with every input pattern vector and take the value of

the inputs. After activation function for the first layer executed, propagation for the next layer will done.

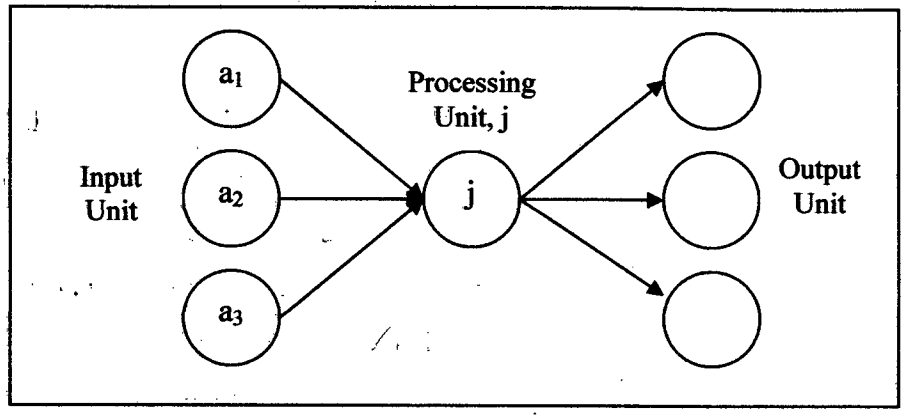

Figure 2.2: Feedforward Neural Network Operation

Figure 2.2 show the Feedforward Neural Network Operation done.Inputs are put into the network to the processing unit j from the input layer which is at the left side. Output from unit I propagated to unit j and generate:

$$S_j = \sum_{i=1}^{n} a_i w_{ij} \quad , j = 1, 2, .., m \qquad (2.1)$$

with a_i is activation function for unit i, m is number of output and w_{ij} as weight from unit i to unit j. nonlinear function is used to calculate $f(s_j)$, and f in this case is sigmoid function. After function f has been calculated, result obtained is going to be the activation function for unit j and output for the network.

ii) **Backpropagation Model**

This model involving two phases, feedforward phase and backprop phase. In the first phase (feedforward), inputs are propagated to the output layer to obtain output. And then, it will calculate error and propagate the error back to the input layer and the weight values will be adjusted. The process is repeated and only stopped when the error value satisfied the selected error criteria. (Negnevitsky, 2005)

2.5 Basic Structure of Neural Network

Neural network consist of processors (neurons) that similar to human neuron. These neurons are interconnected through signal links from one neuron to another. Each neuron receive input signal but only produce one output signal. The signal is transmitted through outgoing link similar to axon. Neuron outgoing link branch into a few branches that carries the same signal. The link ends at the front entrance of other neuron in the same network.

Neurons are connected through links. Each link has a weight value. Weight is a basic feature for memory in network. Neural network learns through adjustment to these weights. Weight gives the neuron capability. Neural network learn through iterative weight adjustment.

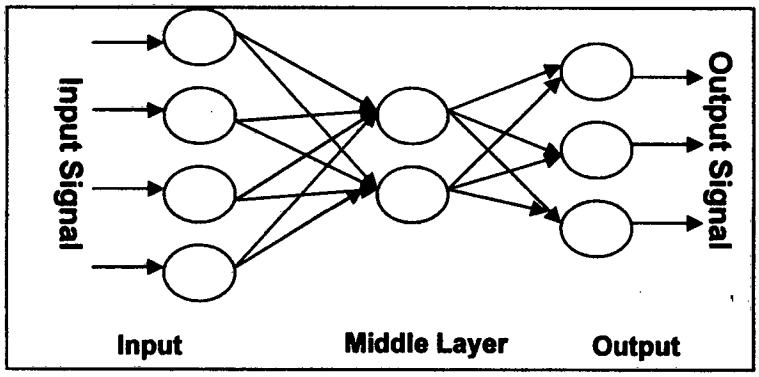

Figure 2.3: Architecture of a basic neural network

As shown in Figure 2.3, neural network consist of layers which there are neurons at every layer. Neurons connected with the inputs from outside and output layer. Weight is adjusted to ensure that the value of input and output are correct. Each neuron is basic information processing unit. It performs activation function by using provided inputs and weights. To build a neural network, number of neurons needed and how the network will be connected to form a network are determined. In other word, we have to select architecture of the network, and then determine the learning algorithm that we want to use. Finally, the network must be trained by

determining the weight and adjusting it using a set of training data. (Negnevitsky, 2005)

Neuron receive signals from input link, performing activation level and sending signals as output signals through output link. Input signal can be raw data or output from other neuron, while output signal can be a final solution to the problem or input for other neurons.

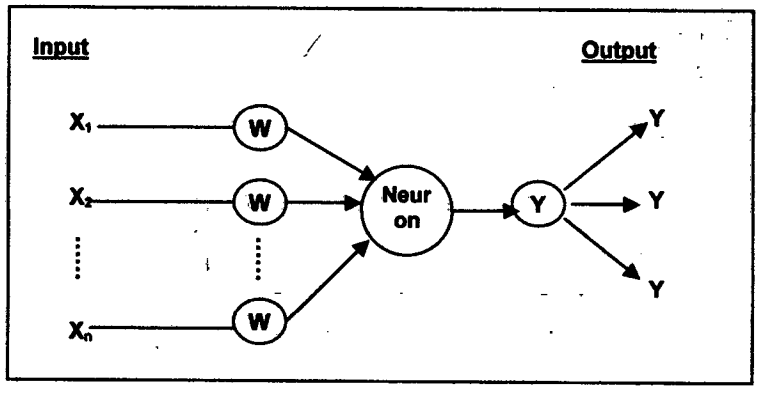

Figure 2.4: Neuron

Figure 2.4 shows the process of neuron operation. (McCulloch & Pitts, 1943) suggested an idea to explain how neuron producing an output. Neuron computes the weighted sum of the input signals and compares the results with the threshold value, θ. If the net input is less than θ, the neuron output is -1 but if the net input is greater or equal to θ, neuron will become active and output is +1. That means neuron use the following activation function:

$X = \Sigma\, xi\, wi$, x: input , w: weight.

$Y = +1$ if $x \geq \theta$, or $Y = -1$ if $x \leq \theta$. (2.2)

Where, X is the net weighted input to the neuron.

Activation function is a function performed to the neural processing unit to produce output signal from input average weights. The most common used activation function is sigmoid function. Beside sigmoid, there are three more common

functions; step function, sign function and linear function. Step and sign functions are also called hard limit function. They are usually used in decision making task for classification and pattern recognition task. Sigmoid function used in back propagation network. Figure 2.5, 2.6, 2.7, and 2.8 show the common activation function used in neural network. (Negnevitsky, 2005)

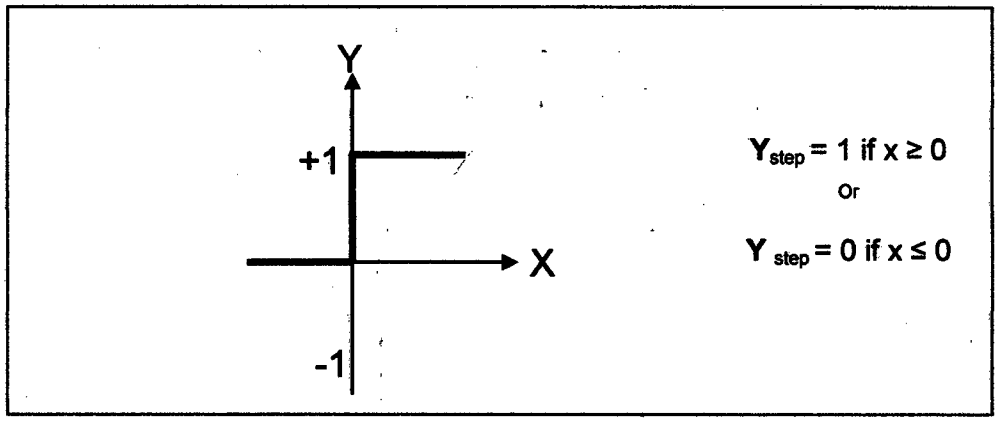

Figure 2.5: Step Activation Function

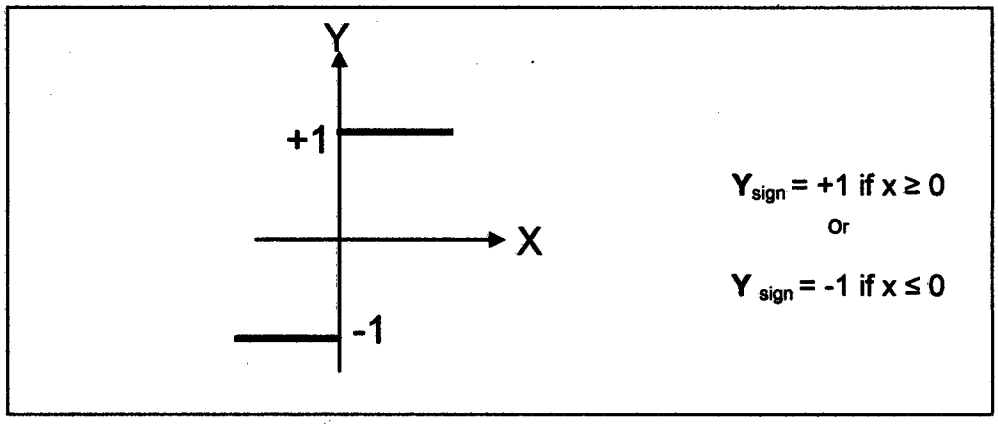

Figure 2.6: Sign Activation Function

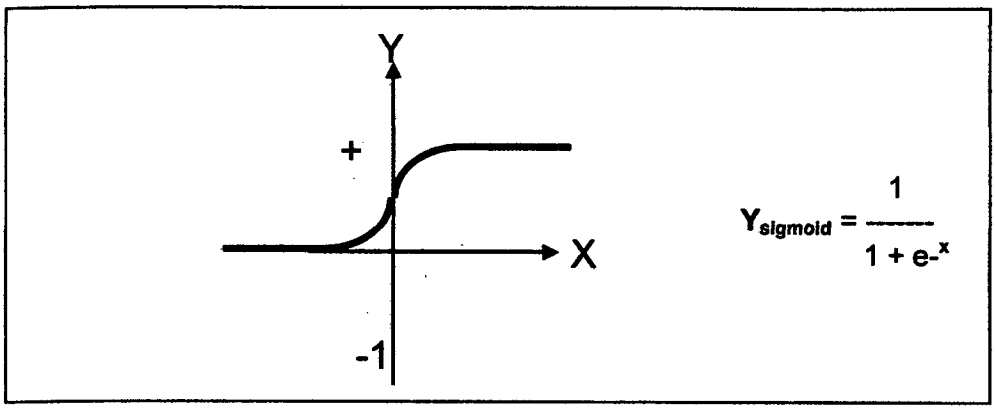

Figure 2.7: Sigmoid Activation Function

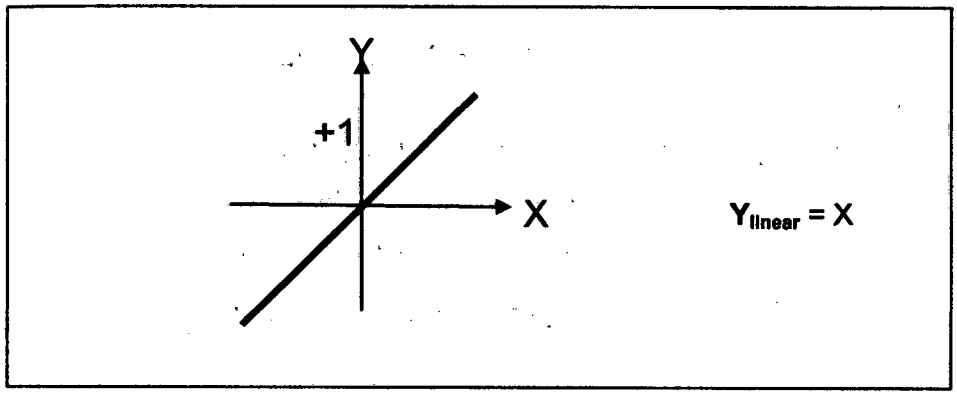

Figure 2.8: Linear Activation Function

2.5.1 Perceptron

Perceptron is the oldest and simplest form of neural network introduced by Frank Rosenblatt in1958. It consists of a single neuron with adjustable weights and a hard limiter. A hard limiter is transfer function that has ability to classify input vectors by dividing input space into two areas by hyperplane. Perceptron learns with supervised algorithm.

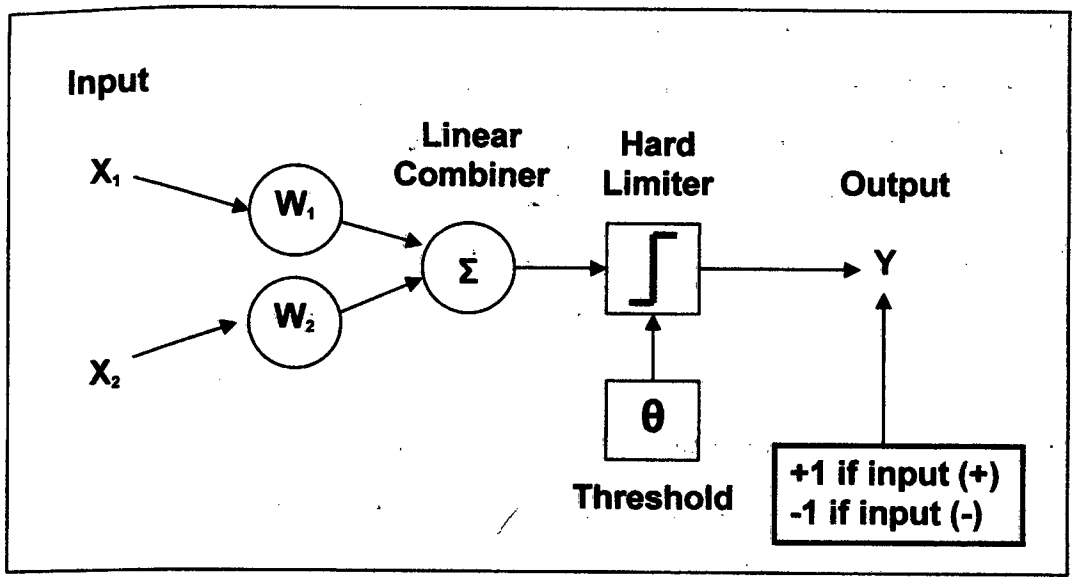

Figure 2.9: Single Layer Perceptron with 2 Inputs

Figure 2.9 show the single layer perceptron with 2 inputs. (Negnevitsky, 2005) A perceptron is used for classification, to classify inputs into classes, C_1, C_2 by hyperplane. It is show in figure 2.10, about inputs perceptron. The equation below describes a hyperplane in the input space given by linearly separable function:

$$\sum_{i=1}^{n} x_i w_i - \theta = 0 \qquad (2.3)$$

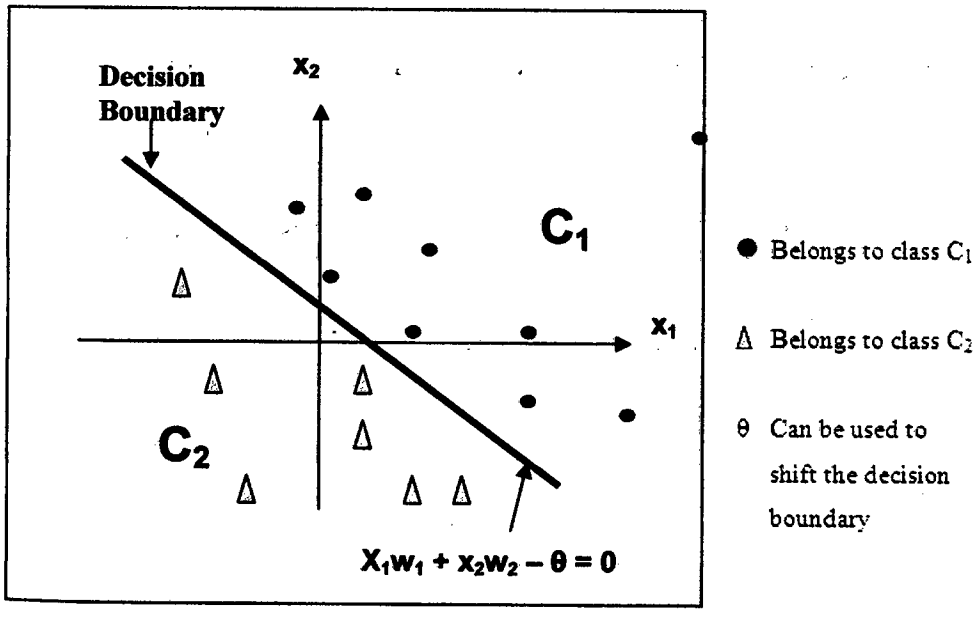

Figure 2.10: Inputs Perceptron

Perceptron networks have several limitations. First, the output values of a perceptron can take on only one of two values (True or False). Second, perceptrons can only classify linearly separable sets of vectors. If a straight line or plane can be drawn to separate the input vectors into their correct categories, the input vectors are linearly separable and the perceptron will find the solution. If the vectors are not linearly separable learning will never reach a point where all vectors are classified properly.

The most famous example of the perceptron's inability to solve problems with linearly nonseparable vectors is the boolean exclusive-or problem.

2.5.2 Perceptron Learning

Step I: Initialization

- Set initial weight, w_1, w_2,.., w_n and bias, θ in the range [-0.5,0.5].

Step II: Activation

- Apply input, x_1, x_2,…, x_n and target output, $y_d(p)$. Calculate the actual output, $y(p)$ at iteration $p = 1$

$$y(p) = \text{step}\left[\sum_{i=1}^{n} x_i(p)\, w_i(p) - \theta\right], \qquad (2.4)$$

where n = number of input, step = step activation function.

Step III: Weight Training

- Adjust weight value :

$$w_i(p+1) = w_i(p) + \Delta w_i(p), \qquad (2.5)$$