# A PROPOSED METHOD FOR SELECTING THE CLUSTERING ALGORITHM IN RADIAL BASIS FUNCTION

## ANG SAU LOONG

## UNIVERSITI SAINS MALAYSIA

## 2008

# A PROPOSED METHOD FOR SELECTING THE CLUSTERING ALGORITHM IN RADIAL BASIS FUNCTION

by

**ANG SAU LOONG**

Thesis submitted in fulfillment of the
requirements for the degree
of Master of Science

**JANUARY 2008**

# ACKNOWLEDGEMENTS

Dedicated to my parents, and my siblings for their continuous warm support and

love.

**TABLE OF CONTENTS**

Page

**CHAPTER 1 : INTRODUCTION**

# CHAPTER 2 : LITERATURE REVIEW

# CHAPTER 3 : METHODOLOGY

# CHAPTER 4 : CASE STUDIES

# CHAPTER 5 : CONCLUSIONS

# BIBLIOGRAPHY

**APPENDICES**

# LIST OF TABLES

Page

# LIST OF FIGURES

# LIST OF APPENDICES

# LIST OF PUBLICATIONS & SEMINARS

1.  Ang, S.L., Ong, H.C., and Low, H.C. (2006). Comparison between modified Radial Basis Functional Link Nets and Radial Basis Function. *Proceedings of the 2nd IMT-GT, Regional Conference on Mathematics, Statistics and Applications*, **4**, 120-125.

2.  Ang, S.L., Ong, H.C., and Low, H.C. (2007). Determining the Preprocessing Clustering Algorithm in Radial Basis Function Neural Network. *Proceedings of ICMS' 07, Integrating Mathematical Sciences within Society*, 572-579.

# SUATU KAEDAH YANG DICADANGKAN UNTUK MEMILIH ALGORITMA BERKELOMPOK DALAM FUNGSI ASAS RADIAL

# ABSTRAK

Rangkaian Fungsi Asas Radial telah digunakan dengan meluas untuk menganggarkan dan mengelaskan data. Model biasa bagi Fungsi Asas Radial menentukan pusat, serakan dan menyesuaikan pemberat sehingga ia dapat menganggarkan data. Dalam penyelidikan terdahulu, terdapat beberapa masalah wujud dalam mencari pusat terbaik bagi lapisan tersembunyi dalam Rangkaian Fungsi Asas Radial. Walaupun beberapa kaedah berkelompok seperti K-min atau K-median digunakan untuk mencari pusat, namun tidak terdapat keputusan yang konsisten yang menunjukkan keputusan terbaik untuk taburan data yang berbeza. Objektif utama tesis ini adalah untuk menentukan kaedah yang lebih baik untuk mendapatkan pusat dalam Rangkaian Fungsi Asas Radial dan juga model yang lebih maju yang dikenali sebagai Rangkaian Pautan Fungsi Asas Radial untuk mengelaskan data. Tiga jenis kaedah digunakan dalam kajian ini untuk mencari pusat bagi kedua-dua model di atas, antaranya pemilihan pusat secara rawak, algoritma berkelompok K-min dan juga algoritma berkelompok K-median. Andaian-andaian yang dibuat dalam kedua-dua kaedah algoritma berkelompok akan membantu meningkatkan prestasi kedua-dua model daripada menggunakan kaedah pemilihan rawak. Oleh itu, kesan kedua-dua jenis algoritma berkelompok dalam pemilihan pusat bagi kedua-dua model tersebut, iaitu Rangkaian Fungsi Asas Radial dan Rangkaian Pautan Fungsi Asas Radial, dari segi kejituan dan kelajuan ditunjukkan dalam kajian ini. Untuk menentukan kaedah kelompok yang dapat

memberikan penyelesaian yang lebih baik, kami menggunakan pengiraan kepencongan Mardia awal untuk mencari kaedah yang terbaik bagi mendapatkan pusat terbaik bagi kedua-dua model mencapai satu penyelesaian yang lebih baik untuk pengelasan data. Oleh itu, kepencongan data dikira sebelum memutuskan untuk memilih sama ada kaedah berkelompok K-min atau K-median dalam pencarian pusat bagi Rangkaian Pautan Fungsi Asas Radial. Di samping itu, suatu kriteria pemilihan awal ini digunakan untuk menunjukkan peningkatan keberkesanan dalam pengelasan data. Kami juga menggunakan dua set data sebenar untuk mengemukakan kaedah cadangan kami.

# A PROPOSED METHOD FOR SELECTING THE CLUSTERING ALGORITHM IN RADIAL BASIS FUNCTION

## ABSTRACT

Radial Basis Function Networks have been widely used to approximate data and classify data. In the common model for radial basis function, the centres and spreads are fixed while the weights are adjusted until it manages to approximate the data. From past research, there exist some problems in finding the best centres for the hidden layer of Radial Basis Function. Eventhough some clustering methods like K-means or K-median are used in finding the centres, there are no consistent results that show which one is better due to the different distributions of the data. The main objective in this thesis is to determine the better method to be used to find the centres in Radial Basis Function and the more advanced model which is Radial Basis Functional Link Nets for data classification. There are three types of method used in this study to find the centres of both models above; these include random selections, K-means clustering algorithm and also K-median clustering algorithm. The assumptions made in both clustering methods will help to enhance the performance of both models rather than using random selections. Therefore, the effects of the two types of clustering algorithm on centres selection for both models which are Radial Basis Function and Radial Basis Functional Link Nets in terms of accuracy and speed are shown in this study. To determine which clustering methods is the best solution, we apply preliminary Mardia's skewness calculation to find the best method to obtain the centres of both models in order to achieve a better solution for the data classification. Therefore, the skewness

of the data is calculated before deciding to choose between the K-means or K-median clustering method in finding the centre of Radial Basis Function Network. Besides, an initial selection criterion is used to show the improvement of efficiency in data classification. We use two sets of real data to demonstrate our proposed method.

# CHAPTER 1

# INTRODUCTION

## 1.1    The Human Brain and Artificial Neurons

In the human brain, there are approximately $10^{11}$ to $10^{13}$ neurons in the nervous system (Nauck *et al.*, 1997 and Patterson, 1995). A neuron is a small cell which receives input or more precisely stimuli in electrical signal from multiple sources such as the sensory or other types of cells and responds by generating electrical impulses which are transmitted to other neurons or effectors organs such as muscles and glands (Patterson, 1995). A neuron is composed of a nucleus, a cell body, dendrites and synapses (refer to Figure 1.1).
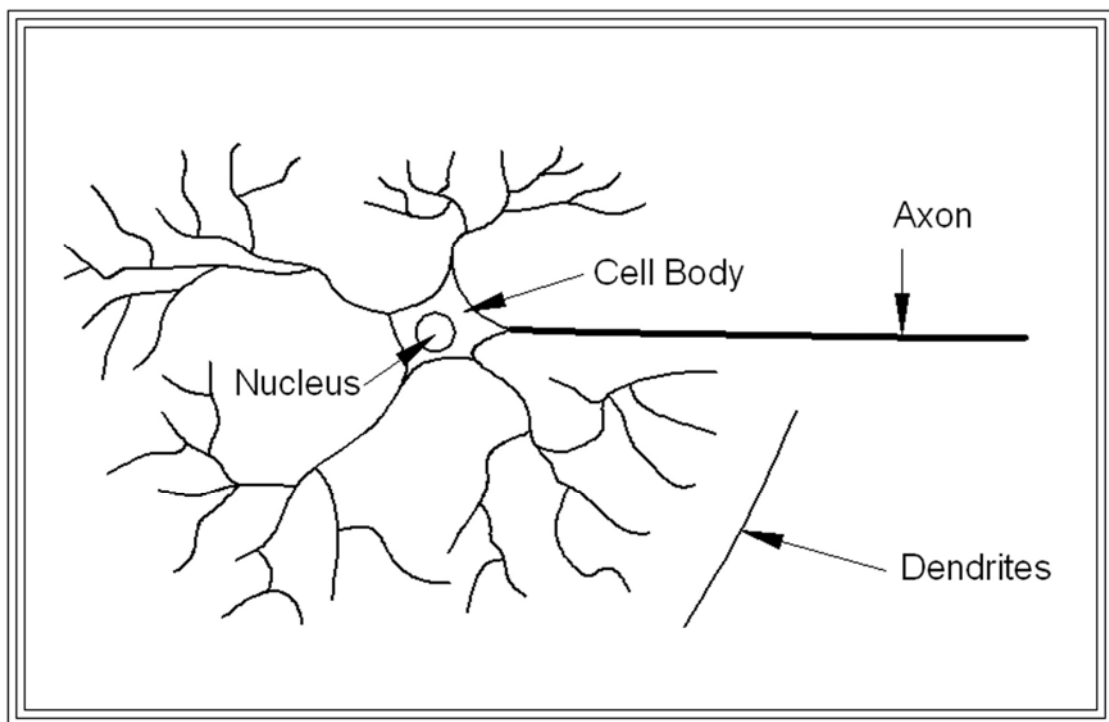


Figure 1.1: Components of a Neuron

The neurons are connected to each other by axons which are projected from their cell body or soma. The electrical signal in a neuron is sent through the axons to other neurons. Signal is also received from other neurons and passed to a neuron via the axons' connections and its dendrites. Joints which exist in between the axons are called synapses and the distance of the tiny gap is about 200 nm wide (Muller *et al.,* 1995). A synapse allows interchange of nerve impulses from one neuron to another neuron (refer to Figure 1.2).
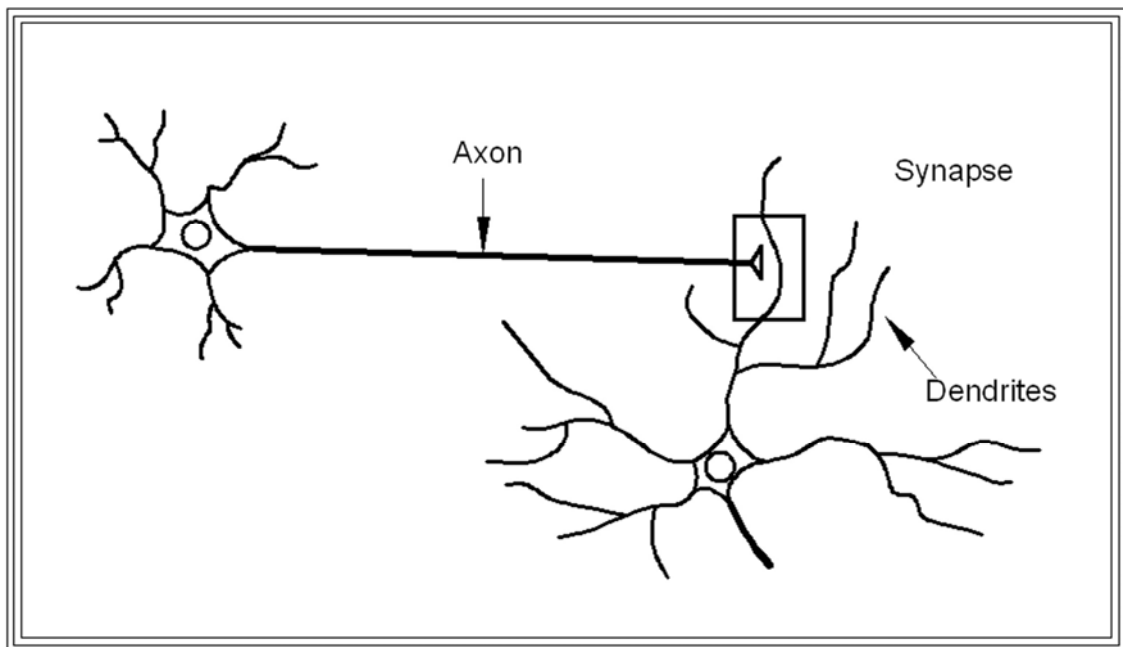


Figure 1.2: Structure of a Synapse

If the electrical inputs' signal received from other cell is large and continuously accumulated, a neuron sends a spike of electrical activity down to its axon to other neurons to excite or inhibit other neurons via synapses.

## 1.2 A Brief Introduction on Artificial Neural Networks

Artificial neural networks are simulated and simplified models of the central nervous system which are based on the structure of a real biological neuron system. These networks are highly interconnected by a group of artificial neurons in which each of them acts as small processing unit to process information. The artificial neurons react to the input stimuli, learn from training examples of information and adapt to various kinds of information. There are synaptic weights or weighted connections which link all the artificial neurons so that these neurons are able to communicate with each other and to work in a parallel way to solve problems.

## 1.3 The Multilayered Perceptrons Neural Networks

The multilayered perceptrons neural networks had been used widely in function approximation, control, forecasting and pattern recognition (Looney, 1997 and Haykin, 1999). The reason for using multilayered perceptrons neural networks in these applications is due to their properties and advantages. The advantages of the multilayered perceptrons neural networks are stated as follows.

i. The assumptions on the model are not necessary because the multilayered perceptrons neural networks are able to train and produce results for any numerical problem which is generated from pattern recognition or function approximation problem. On the other hand, different kinds of problems (linear or non-linear) can be solved only by using the same multilayered perceptrons neural networks with small

adjustments in numbers of the input nodes and output nodes together with the parameters in the networks.

ii.    The complexity of solving non-linear problems using multilayered perceptrons neural networks is reduced compared to linear regression analysis. If we apply linear regression method (Fukunaga, 1990) in non-linear problem, we are demanded to verify the non-linearity of the system. In inserting the square of some input variable or product of two input variables causes the linear regression model to become more complicated. In contrast, the multilayered perceptrons neural networks do not face the same limitations as these models do not acquire the analysis of the non-linearity of the problem given (Abe, 1995).

iii.    The multilayered perceptrons neural networks are packed with high generalization of the activation functions. As the sigmoid function in multilayered perceptrons neural networks is a global approximator (Looney, 1997), the output depends on any input from the feature space. We can replace the sigmoid function with the local approximator such as Gaussian function in order to train the data efficiently (Hagan *et al.*, 1995). Since the structure in multilayered perceptrons neural networks is constructed with many layers, the time to compute the results using different activation functions is distinct.

iv.    Based on the mapping connectivity which exists in multilayered perceptrons neural networks, the neural networks are able to run in a parallel way. The key to compute the training in a parallel way is according to the neurons in the same layer which stand independent of one another. Therefore, no matter how the data set is processed in

forward propagation or backpropagation, the neural networks are still able to go through the training in a parallel style. The parallel learning algorithm is time saving and powerful (Looney, 1997).

Despite the application of the multilayered perceptrons neural networks in many areas, there are some issues on the implementation of the neural networks. The weak points of the neural networks are summarized as below.

i. Determination of the number of hidden layers is one of the crucial problems in applying multilayered perceptrons neural networks. Since the number of hidden layers is determined by trial and error or heuristic way (Looney, 1997 and Haykin, 1999), it is hard to obtain the optimum solution from the training of multilayered perceptrons neural networks. The number of hidden layers plays an important role in the efficiency of the neural networks because the neurons which lie on each of the hidden layers have direct impact on the convergence of training and generalization ability. Optimal number of hidden layers is required to enhance the performance of the multilayered perceptrons neural networks. Overtraining due to too many hidden layers or under training caused by too few hidden layers is mostly not encouraged in building the structure of the neural networks (Looney, 1997 and Looney, 2002).

ii. The number of hidden neurons to be used in each layer in the multilayered perceptrons neural networks is hard to be optimized. Some researchers had been trying hard to search for the optimum number of hidden neurons during training and after training. The determination of the number of neurons made during training is usually divided into three

methods. The first method adds hidden neurons during training as mentioned in Ash (1989), Azimi-Sadjadi *et al.* (1993), Wang *et al.* (1994) and Fahlman and Lebiere (1990) but the papers in Sietsma and Dow (1991) and Oshino *et al.* (1993) proposed the deletion of the neurons. Another alternative way is found in Bartlett (1994) which proposed the deletion and addition of neurons during training. The determination of neurons after training was discussed in Xue *et al.* (1990), Fogel, (1991), Wada and Kawato (1991) and Kayama *et al.* (1990). Derivation is also made upon the upper and lower bounds of the number of hidden neurons of the three-layered neural network by Huang and Huang (1991).

iii. Since the training of multilayered perceptrons neural networks is based on error backpropagation algorithm which is a hill climbing technique (Haykin, 1999), it has a risk of being trapped in a local minimum where every little changes in the weights increase the error function. This problem is exposed in a simple demonstration by Gori and Tesi (1992). They had shown that in a non-linear example where the error backpropagation algorithm in a single layered neural network failed to converge due to being trapped in local minimum. If one single layer can fulfill the requirements to converge, then there are chances for the multilayered perceptrons neural networks to have bad convergence and the solution may be faulty. Even though some may adjust the initial parameter such as weights until we yield the most satisfactory outcome with the lowest sum of squared error, Rumelhart concluded that the

outcome with the lowest sum of squared error does not guarantee the learning is the best (Rumelhart *et al.*, 1986).


## 1.4 A Brief Introduction on the Radial Basis Function Neural Networks

Radial basis function neural networks (RBFNN) are robust and powerful feedforward artificial neural networks (Looney, 1997). This neural network model originated in the 1960s and started to gain popularity since late 1980s (Patterson, 1995). The basic idea is that from M continuous number of radial basis functions which cover the feature space $[0, 1]^N$ where N is the dimension of the input data, it is necessary that every input vector be mapped to each centre of the radial basis function or the Gaussian function (Looney, 1997). The approach to this neural network can be viewed as curve fitting on a multidimensional space (Bishop, 1995). Radial basis function neural networks are used in function approximation and interpolation (Sgarbi *et al.*, 1998), pattern classification and recognition (Haykin, 1999), protein sequence residue spatial distance prediction (Zhang and Huang, 2004), data mining applications (Buchtala *et al.*, 2005), and the training is faster than its rival, the multilayered perceptrons neural networks (Bishop, 1995). In general, a radial basis function neural network has three major parts in its structure:

i. The first layer of the RBFNN is normally known as the input layer where each input neuron represents a feature vector.

ii. The second layer which is the hidden layer consists of hidden nodes where each node has the centre corresponding to any input that is near to it.

iii.    The output layer where the output we obtain from the hidden layer is compared with the target values (refer to Figure 1.3).
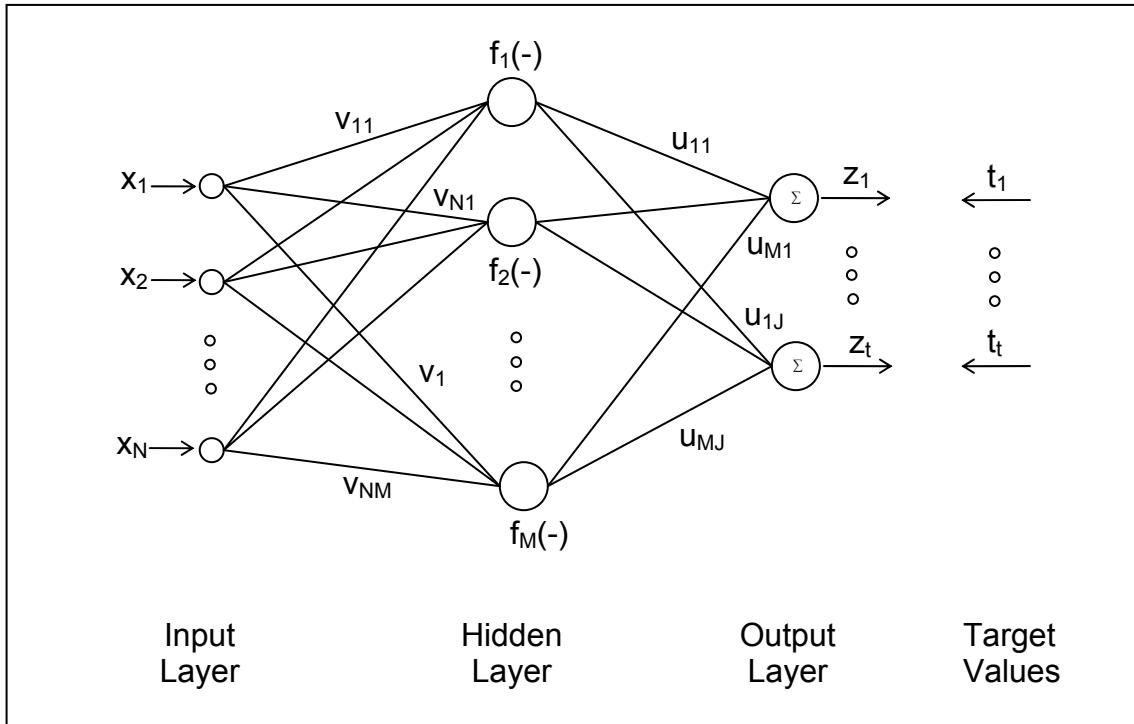


Figure 1.3:  Structure of RBF

The activation function used in this thesis is the Gaussian function. The Gaussian function is well-known for its robustness as a universal approximator (Lo, 1998). Therefore, the Gaussian function is suitable to be used as the kernel function in the radial basis function neural networks.

In between the first layer and the second layer, in non-linear approximation, the input data are fed into the Gaussian functions for further processing. When the input data become closer to the activation function, the activation function will be activated. Then by a weighted sum of the output of the

hidden units, output z which is located at the last layer of the neural networks will be compared with the target values. The training processes continue as long as the differences between the output and the target values have not reach a stopping criterion. In order to obtain the differences between output and the target values, we use the sum of squared error since it is the most efficient method for calculation of differences (Bishop, 1995).

## 1.5 The Rationale for Using Radial Basis Function Neural Networks

In this section, we state the reasons for using radial basis function neural networks instead of multilayered perceptrons neural networks. There are several aspects where radial basis function neural networks outperform multilayered perceptrons neural networks. The advantages are as follows:

i. Radial basis function neural networks learn without getting trapped in local minima as compared to multilayered perceptrons neural networks (Bianchini *et al.*, 1995)

ii. The training time for radial basis function neural networks is much less than the multilayered perceptrons neural networks.

iii. Restraining of the weight value is not necessary when training the radial basis function neural networks. However, for multilayered perceptrons neural networks, the weights have to be restrained (Looney, 1997).

iv. The parameters need not have to undergo non-linear optimization of the neural network to be determined (Bishop, 1995). In applications using neural networks, there are possible cases where the input data has no target values. The process to label the input data with corresponding target variables may be time consuming especially when the input data

are huge. Therefore, radial basis function neural networks which perform two stage training processes have the advantage of determining the non-linear representation given by first layer of the neural network. This process leaves a comparatively smaller number of parameters in the second layer to be determined by using the labelled input data. For good generalization, the number of data points is required to be huge compared with the number of parameters which are determined by the labelled data.

Hence, we use the radial basis function neural networks as our training network based on the accuracy of the output, simplicity in structure and faster learning concepts rather than multilayer perceptrons neural networks (Moody and Darken, 1989; Hassoun, 1995; Hwang and Bang, 1997; Haykin 1999; Li *et al.*,2002; Sing *et al.*, 2003 and Pang, 2005) .

## 1.6    Major Problems of Radial Basis Function Neural Networks

In our study, we only focus on the centre of each Gaussian function in each feature region of interest, which is one of the main keys for boosting the performance of the radial basis function. As the performance of the radial basis function neural networks can be improved in terms of speed and accuracy, the selection of centres for each hidden node becomes very significant.

Researches had been done to find the best centres for the hidden nodes which are situated at the hidden layer. Since the centre of the Gaussian function plays the most important role for non-linear approximation, the method for

finding these centres so that the input data is able to map to the right circular disks becomes a challenge to many researchers (Schwenker *et al.*, 2001 and Sing *et al.*, 2003). In order to choose the best centres, some researchers use clustering methods such as K-means (Sing *et al.*, 2003), dynamic clustering (Looney,1997), K-nearest-neighbor algorithm which combines with fuzzy c-mean algorithm (Wang *et al.*, 2002), clustering-based symmetric (Chen *et al.*, 2007) and density based clustering algorithms to solve the problem.

The most promising results can be obtained if the distance for each input vector in the feature space from the core centres of the Gaussian functions reduce to a minimum value. On the other hand, when a good set of centres in radial basis function is obtained, it means that all input data are covered by the Gaussian Function with its good position of centre and its optimum spread.

Different clustering methods lead to different quality of the outcome obtained from the neural networks. Among the clustering algorithms applied in finding the centres of the Gaussian function, K-means clustering algorithm is widely used for selection of centres. Another method, using K-median algorithm to choose the centres of the Gaussian function can also work since there are a lot of similarities in the structure of both K-means and K-median clustering methods.

However, the determination of which clustering method should be used in different distributions of data set remains to be identified. Therefore, we are

motivated to search for a good solution to determine which clustering method is better in partitioning the different types of data set.

## 1.7    Hierarchical and Non-Hierarchical Clustering Methods

Clustering methods can be divided into two major groups which are hierarchical and non-hierarchical. In hierarchical clustering algorithm, it is divided again into two different types of methods. The first method is the agglomerative hierarchical methods. This hierarchical method starts with the individual object so that we have a lot of clusters in the data space. Next, the most similar objects are assigned into the same partition and form the first grouped set of clusters. The process is repeated by combining the new set of clusters according to their similarities. Therefore, we obtain a few clusters at the end by setting the stopping criteria. Some of the examples of agglomerative hierarchical methods are single linkage, complete linkage and average linkage which can found in Patterson (1995). Another hierarchical type of clustering is the divisive hierarchical methods which work in a contradictory way to the agglomerative hierarchical methods. The whole object is first assigned into a big group which is then separated into two subgroups where the objects in the two subgroups have different attributes to each other. The clustering steps are carried on by assigning more dissimilar objects into further subgroups until the stopping criterion is matched. Examples of divisive hierarchical methods can be seen in Anderberg (1973) and Everit (1993).

The classical K-means was first found by MacQueen (1967). It is a non-hierarchical and unsupervised clustering method which separates the data sets

into their different clusters according to the number of K and also depending on the distances between each input data from the K centroids. The K-means clustering algorithm is used in data mining and data exploring areas such as protein synthesis and genes expression data analysis.

Somehow there are certain drawbacks about the K-means clustering algorithm from Johnson and Wichern (1998) and Cheung (2003). The first drawback is the priori knowledge on the number of K is heuristic and will undergo the process of trial and error. The preliminary number of K is crucial in determining the quality of the clustering results. The second drawback is the initial centre of each cluster is hard to choose for getting good results. The initial centres of each cluster must be chosen in such a way that it is sufficient to cover the input feature space and attract those input data which are close enough to it and group the particular input data into the respective clusters. This process goes on until there is no more assignment of input data to other clusters. The third drawback of K-means clustering algorithm is the sensitivity of the mean to the outliers. The outliers influence the process of cluster assignment and result in wrong data classification. Due to the weaknesses as stated, most researchers argue that K-means may not give the best solution for the data clustering.

However, some researchers propose the opposite opinion on K-means clustering method (Looney, 1997; Johnson and Wichern, 1998 and Chen *et al.*, 2005). The simplicity of the K-means clustering structure is efficient in reducing the time of processing as compared to other clustering methods such as density

based clustering and hierarchical clustering. Moreover, the efficiency of K-means clustering method is proven in terms of computational cost (Chen *et al.*, 2005). Since the basic data sets do not have to be stored during the compilation of K-means, K-means clustering method can be applied to much larger data sets than the hierarchical clustering method (Johnson and Wichern, 1998).

The K-median algorithm is a hybrid of the K-means clustering algorithm. In K-median algorithm, we have to rearrange the order of input data inside the cluster in ascending order. Then only can we apply the median for obtaining the new centre of each cluster. There is also another attribute that distinguishes between the K-means from the K-median clustering algorithm which is the sensitivity of outliers. In handling the outliers which are far away from all the centre of the clusters, K-median algorithm performs better when compared with K-means algorithm. The outliers will not have great influences on the results obtained.

Both clustering methods partition the data set into linearly separable classes (Fisher and Van Ness, 1971).

## 1.8 Objectives of the Study

There are three major objectives in this study:

i.  To improve the performance of radial basis function by applying K-means and K-median clustering method in the selection of centres of the hidden nodes in the aspects of number of iterations and accuracy.

ii.     To determine the clustering method to be chosen from either K-means or K-median by doing preprocess calculation of multivariate data skewness. The value of the calculation will show which method is better to be used to increase the efficiency of radial basis function.

iii.    To extend the above results from the radial basis function neural networks to the radial basis functional link nets and to show the same principle applies.

## 1.9     Format of Thesis

There are all together five chapters in the thesis. The structure of radial basis function neural networks is briefly introduced in Chapter 1. Radial basis function neural networks use the Gaussian Function as their activation function. The types of different clustering methods on the selection of centres are explained. Having briefly explained on the hierarchical and non-hierarchical clustering methods, we provide a clear reason for applying the methods in selection of centres for both models. Besides, we state the objectives which we aim to achieve in our study.

Chapter 2 gives literature review on the artificial neural networks, radial basis function neural networks and radial basis functional link nets. The literature review also covers K-means and K-median clustering methods and Mardia's Multivariate Skewness.

The preprocessing methodology is explained in Chapter 3. For the contents of Chapter 3, we present the formulae which are used for the training

models, the radial basis function neural networks and the radial basis functional link nets. We also show every step involved in K-means and K-median in finding the centres for the activation function in these two training models. By looking into the measure of Mardia's Multivariate Skewness, we summarize the flow of the calculation steps involved in determining the skewness of the data set.

Chapter 4 begins with the introduction for the case studies. The data sets which we used in our proposed method are discussed. We also state our objective of case studies to achieve better performance in terms of accuracy and speed for radial basis function. We extend our study by using the more powerful radial basis functional neural networks. We separate each data set into the test data set and training data set. The way to use Mardia's multivariate skewness in measuring the skewness for the data sets is explained. We also explained the method to obtain the centres of the Gaussian functions. Guidelines for classification are stated to determine whether an output is correctly classified or wrongly classified. The results obtained are shown in the form of tables and graphs. We first show the results of radial basis function neural network and radial basis functional link nets under the effects of the clustering methods. We then show the results that support the aim of the research which is the determination of the clustering methods used in both models by Mardia's multivariate skewness. Then, we discuss the results which we obtained based on the two real data sets. Conclusions are stated in the last section of Chapter 4 where we summarize the clustering method which is suitable to be used and the effectiveness of the radial basis functional link nets.

In the last chapter, we present the conclusions and future work. The conclusions are summarized while the future work will give some idea and directions which are useful for further research in the area of the radial basis function neural networks. Our contributions in this study are stated clearly in the last part of this chapter.

# CHAPTER 2

# LITERATURE REVIEW

## 2.1    Artificial Neural Networks

Based on the knowledge about biological neurons, the first artificial neuron was created by McCulloch and Pitts (1943) in their work which starts the new era of artificial neural networks. This synthetic neural device modelled the human intelligence in decision making and also thinking abilities in solving problems such as pattern recognition and combinatorial problems. This simple artificial neuron was formulated using the studies of neurophysiology, which is the research on the central and peripheral nervous system by recording of bioelectrical activity, whether spontaneous or stimulated, and the methods of mathematical logic and become the disciplines of neural networks.

In this model, there are some major components such as input neurons, input lines which consist of adaptive weights or synaptic weights, threshold or activation functions, and output. The input of the artificial neuron can be either excitatory or inhibitory.  The input lines lead the input to be summed up and activate the output. The threshold functions are labelled with a set of fixed threshold values to determine the condition for the sum of input to activate the output and the output that we obtained is the class identifier (refer to Figure 2.1).
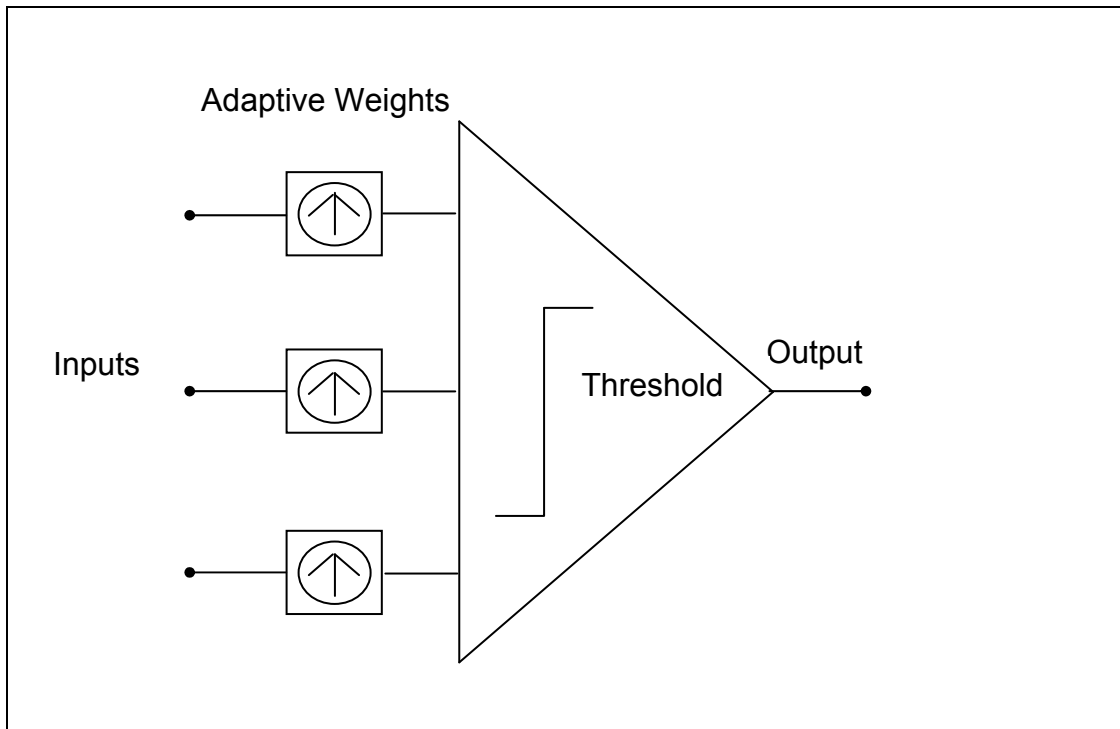
Figure 2.1:  A McCulloch-Pitts Neuron

Even though the artificial neuron was created, there are some limitations compared to the real biological neuron. The first limitation is that the artificial neuron does not contain any mechanism for learning but functions by just setting the threshold value for activation function. The model of McCulloch and Pitts did not explain sufficiently on the functionality of human brain including the learning ability of the brain and the physical changes in the neural connections. The second limitation is that the artificial neuron only represents a type of neuron whereas there are other types of neuron. One example of different types of neuron is the pyramid cell (Nauck *et al.*, 1997).

In 1949, an important achievement in the development of neural networks was made by Donald O. Hebb. Hebb postulated that a learning process which states that the effectiveness of a variable synapse between two

neurons is increased by continued activation of one neuron by the other across the synapse (Hebb, 1949). For instance, if Neuron A is stimulated continuously by Neuron B while Neuron A is still active, then Neuron A will become more sensitive to stimuli from B (Looney, 1997). Hebb's studies provide the learning rule for artificial neural networks but his studies did not contribute much in engineering of the artificial neural networks (Haykin, 1999).

Computer simulations on artificial neural networks were done in 1956 by Rochester's research group (Rochester *et al.*, 1956). They generalized the model of artificial neural networks by including inhibition, so that active cell or input could inhibit others from turning active. The normalization of synaptic weights is made to prevent unbounded growth in the weights of the neural networks model (Patterson, 1995). Nevertheless, the neural networks were too small and not conclusive to have any chance to exhibit behaviour that could reasonably be identified with thinking.

The first artificial neural network was developed in 1958 by Rosenblatt, Wightman and Martin. They devised a machine called the perceptrons which operate almost the same way as the human mind (Rosenblatt, 1958). It was a progression after the work of McCulloch and Pitts and Hebb's research. The perceptrons were built by two major layers which are the input layer and the output layer. Each layer is fully connected to each other but there is no connection for the neurodes in the same layer. The input layer was connected to output layer via the associated weights. By using the perceptrons model, the field of the neural networks started to boom based on the assumption that the

perceptrons model was able to guarantee the learning process of the input which would lead to the output or the solutions of the given problem.

Having almost the same structure and learning ability of the perceptron model, Bernard Widrow and Ted Hoff proposed the first practical application of artificial neural networks which is called ADALINE (ADAptive LInear NEuron) (Widrow and Hoff, 1960). If a number of ADALINEs built up the networks, then he named the ADALINEs as MADALINE (Multiple ADALINEs). They also contributed in introducing the new learning rule which is called the LMS (Least Mean Square) or also known as Widrow-Hoff learning method. LMS algorithm was applied into the famous error backpropagation learning method used in multilayered perceptrons neural networks (Patterson, 1995). The difference between the perceptrons model with ADALINE model is only at the part where the transfer function is linear for ADALINE networks while the hard-limiting transfer function is for Perceptrons networks. Both models were outstanding in solving the linearly separable problems which is used in pattern recognition where the patterns must be sufficiently and linearly separable from each other to make sure that the decision surface consists of a hyperplane (Hagan *et al.*, 1995 and Haykin, 1999).

However, perceptrons network was later heavily critiqued by Minsky and Papert in 1969. These researches pointed out the failure of a single layer of the perceptrons to learn the XOR logic (Exclusive Or) (Minsky and Papert, 1969). XOR logic can be viewed as the logic in classifying points where each of them is either in the class 0 or class 1 in the unit hypercube (Haykin, 1999). The

limitation of perceptrons which was only applicable in solving linearly separable problems disappointed the researcher of neural networks for almost two decades (Looney, 1997). Nevertheless, the researches on neural networks were carried on by several researchers such as Stephen Grossberg who concentrated his work on cooperative–competitive learning systems and the recall of spatial-temporal patterns (Grossberg, 1968 and 1969), Teuvo Kohonen and James Anderson on Associative memory (Kohonen, 1972; Anderson, 1968 and Anderson, 1977), Fukushima who investigated into the Cognitron (Fukushima, 1975) and Werbos who contributed in investigating the adjustment of the weights in multilayered perceptrons neural networks (Werbos, 1974).

The researches on neural networks had a turning point in 1980s when John Hopfield proposed the Hopfield network (Hopfield, 1982). In Hopfield network, Hopfield applied a feedback neural network with threshold units and a mutually connected, symmetrical links. Inside the neural network, the rule of the linear associator became the learning rule for input data. The associated energy function was based on the use of Lyapunov energy function for the nonlinear equations. The energy function dissipated with elapse of time and converged to a state where minimum energy was used. The patterns of the learning process memory were stored as dynamically and stable attractors (Patterson, 1995).

Four years later, Rumelhart, Hinton and Williams came out with the idea of error backpropagation algorithm which replaced the model of McCulloch and Pitts with a model which consisted of continuous and differentiable activation functions (Rumelhart *et al.*, 1986). This helped to solve the constraints of the

single layer perceptron which was not able to solve the XOR logic problem. Therefore, the neural networks had made a big step forward to a more stable model of neural networks that was able to overcome the non-linear problem.

The neural networks not only inherited some properties from the real biological neuron networks, it is also strongly related to statistics (Sarle, 1994). Linear neural networks and perceptrons are almost similar to the linear regression which is one of the statistical models. For the non-linear part, multilayered perceptrons neural networks are considered as another model which is also strongly related to statistics. The highlight of the multilayered perceptrons neural networks compared to the statistical model in non–linear problem solving aspect is that the multilayered perceptrons neural networks need not have any knowledge of the relationship of the input which stands for the independent variable and the output which is the dependent variable. Moreover, we can easily extend the capability of the model by adding more input and output without having difficulties unlike some of the statistical models such as splines or polynomials which suffer from an exponential increase in the parameters used in the equations for overcoming the regression problems (Nauck *et al.*, 1997).

### 2.1.1 Architecture of Artificial Neural Networks

Basically, there are two architectures which build the artificial neural networks. One is feedforward artificial neural networks while the other one is feedback artificial neural networks. In feedforward artificial neural networks, input signals are allowed to travel from input layer to output layer via the middle

layer in one way mode (Hagan *et al.*, 1995; Patterson, 1995 and Looney, 1997). Since the input signals are moving in one direction, we can associate the input to output in straight forward networks. Multilayered perceptrons neural networks and radial basis function neural networks are the examples of feedforward artificial neural networks.

In contrast to the feedforward artificial neural networks, feedback artificial neural networks or recurrent artificial neural networks can have signals that propagate in a lateral, forward or backward manner among the neurons (Hagan *et al.*, 1995; Patterson, 1995 and Looney, 1997). Recurrent artificial neural networks can be very complicated since the neurons of these artificial neural networks compete to determine an output. At the end, only one neuron will have the desired output (Hagan *et al.*, 1995 and Looney, 1997). Hopfield neural network is one of the examples of feedback artificial neural networks.

### 2.1.2 Learning Paradigms of Artificial Neural Networks

Supervised learning, which is also known as learning with a teacher, is one of the learning paradigms in artificial neural networks (Haykin, 1999). In this method, each output has the knowledge about its response to the input signals. By following the desired response of outputs, the neural network will learn in such a way to perform optimum action to input signals (Bishop, 1995). The neural networks are trained and the parameters of the neural networks are adjusted under the influences of the training vector and the error minimization effect (Haykin, 1999).