ARTIFICIAL NEURAL NETWORKS APPROACH FOR THE DETERMINATION OF AQUIFER PARAMETERS

by

A. Ufuk Şahin

B.S., in C.E., Boğaziçi University, 2005

Submitted to the Institute for Graduate Studies in
Science and Engineering in partial fulfillment of
the requirements for the degree of
Master of Science

Graduate Program in Civil Engineering
Boğaziçi University

2008

ACKNOWLEDGEMENTS

First, I would like to express my sincere gratitude to my thesis supervisor, Dr. Cem Avcı for his guidance, support and encouragement throughout the preparation of this thesis. Without his sympathy, patience and guidance, the accomplishment would be impossible.

I would also like to thank Dr. Osman Börekçi and Dr. Nadim Copty for their kind and supportive attitude to me and showing high interests and valuable advices to my thesis, also for the courses they gave me in my graduate education.

I can not find the proper words to thank my dear friends, Emin Çiftçi, Yavuz Tokmak, and Cenk Güngör for their generous help and motivating me whenever I have disappointed during the preparation of this research.

I am grateful to my parents for the endless support they have given me throughout my life. I would like to express my gratefulness to my engaged, Başak, without whom probably I would have never done this work.

ABSTRACT

ARTIFICIAL NEURAL NETWORKS APPROACH FOR THE DETERMINATION OF AQUIFER PARAMETERS

The determination of the aquifer parameters with sufficient accuracy is an important issue in the application of the mathematical models which have been developed for the groundwater systems. Recently, Artificial Neural Networks (ANNs) approach has become popular trend in the solutions of several hydrological problems. ANNs have the ability of learning and processing the introduced data without the need for the full understanding the physical world of the problems at the hand. In this research, ANN approach has been utilized to determine confined aquifer parameters such as transmissivity and storativity. Furthermore, an iterative ANN model has been proposed to determine leaky confined aquifer parameters. The results that have been obtained by ANN models have been also compared to the conventional curve matching procedures that are employed for the determination of aquifer parameters. As a second dimension of this thesis, a numerical experiment has been conducted to contour transmissivity distribution of a hypothetical aquifer by the ANN approach. The performance of the ANN model has been investigated by comparing the solutions of mathematical methods, namely Radial Basis Function and Ordinary Kriging, which are used in data interpolation.

As a conclusion, the ANN approach has been successfully applied to determine aquifer parameters. ANN models demonstrate that the ANN approach can be an alternative modeling technique for the solution of various Hydrological problems.

ÖZET

AKİFER PARAMETRELERİNİN BELİRMESİNDE YAPAY SİNİR AĞLARI YAKLAŞIMI

Akifer parametrelerinin sağlıklı ve güvenilir bir biçimde belirlenmesi, yeraltı suları için geliştirilen matematiksel modellerin daha kesin sonuçlarlar vermesi bakımından önemlidir. Son zamanlarda, çeşitli Hidroloji problemlerinin çözümlerinde matematiksel modellerin yanı sıra yapay sinir ağları modellerinin de kullanıldığı görülmektedir. Yapay sinir ağları, doğası gereği problemin fiziksel yapısının anlaşılmasına gerek kalmadan karmaşık ilişkileri rahatlıkla öğrenip yeni bilgileri de işleyebilmektedir. Bu çalışmada zemin iletme kapasitesi ve su biriktirme katsayısı gibi basınçlı akifer parametrelerinin belirlenmesi için yapay sinir ağları yaklaşımından yararlanılmıştır. Ayrıca sızdıran basınçlı akifer için yeni bir yaklaştırmalı yapay sinir ağı modeli geliştirilmiştir. Yapay sinir ağları yaklaşımı ile elde edilen sonuçlar akifer parametrelerinin belirmesi için geliştirilen geleneksel eğri eşleştirme yöntemleriyle de karşılaştırılmıştır. Bunlara ek olarak, sahadan toplanan zemin iletme kapasitesinin bütün bir akifer içinde dağılımı bulmak amacıyla yapay sinir ağları modeli ile bir sayısal deney yapılmış, modelin verimliliği aynı amaç için geliştirilen matematiksel yöntemlerin sonuçları ile sorgulanmıştır.

Sonuç olarak, yapay sinir ağları yaklaşımının akifer parametrelerinin belirmesi için başarıyla uygulanabileceği gösterilmiştir. Hidroloji alanında, yapay sinir ağları modellerinin alternatif bir modelleme seçeneği olduğu vurgulanmıştır.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS/ ABBREVIATIONS

B	Leakage Factor
b_j	Bias or threshold term of the j^{th} neuron
b'	Thickness of leaky layer
C_{ij}	Entry for Covariance Matrix at i^{th} row and j^{th} column
Cov	Covariance Operator
c^2	Shape Parameter
$d_j(n)$	Target value of the output neuron j for iteration n
$E(\cdot)$	Exponential integral
$e_j(n)$	Error of the output neuron j for iteration n
$H(x)$	Sum squares of instantaneous errors
h	Hydraulic Head
I	Identity matrix
I_0	Modified zero order Bessel function of first kind
J	Jacobian Matrix
J^T	Transpose of Jacobian Matrix
K_0	Modified zero order Bessel function of second kind
K'	Vertical Hydraulic Conductivity
L	Layer number
N	Number of data
n	Iteration number
Q	Pumping Rate
q	Vertical recharge
$R(x_0)$	Error of the estimated sampling point
R^2	Coefficient of determination
S	Storativity
s	Drawdown
T	Transmissivity

t	Time
Var	Variance Operator
$V(x_0)$	Value of sampling point
$\hat{V}(x_0)$	Estimation value of sampling value
$W(u)$	Theis Well Function
$W(u, r/B)$	Hantush- Jacob Well Function
w_{ji}	The synaptic weights of the j^{th} neuron due i^{th} pair of input data
Δw	Weight changes
X	Input data matrix
Δx	Horizontal distance
$y_j(n)$	Output value of the output neuron j for iteration n
y_{output}	Output value of neural network
Δy	Vertical distance
α	Coefficients for Radial Basis Function
$\delta_j(n)$	Local gradient at iteration n
ε_{av}	Average instantaneous errors
$\varepsilon(n)$	Instantaneous error for iteration n
$\varphi(\cdot)$	Transfer function
$\varphi'(\cdot)$	First derivative of transfer function
$\gamma(h)$	Semivariogram function of the separation distance h
η	Learning Rate Parameter
λ	Marquardt scalar
μ	Lagrangian multiplier
μ_c	Momentum Constant
σ^2	Variance of sampling point
σ_R^2	Error variance of sampling point
υ_j	Net function of the j^{th} neuron in the hidden layer

ANN	Artificial Neural Network
BPA	Back Propagation Algorithm
FFBPN	Feedforward Back Propagation Network
LM	Levenberg-Marquardt Algorithm
MAE	Mean Absolute Error
MLP	Multilayer Perceptron
MSE	Mean Square Error
OK	Ordinary Kriging
RBF	Radial Basis Function
RBFCM	Radial Basis Function Collocation Method
RMSE	Root Mean Square Error
RNN	Recurrent Neural Network
RRMSE	Relative Root Mean Square Error
SE	Standard Error

1. INTRODUCTION

Physical-based mathematical models have been developed and employed to analyze complex behavior of groundwater systems for various scenarios. The complex nature of groundwater flow, however, often creates difficulties in the application of mathematical models which ultimately requires simplifications of the physical world of the problem. Artificial Neural Network (ANN) models have recently become attractive in groundwater studies because of their ability to map complex relations without the need of using the physical-based mathematical model laws.

Ground water pumping tests are field tests used to determine aquifer parameters which provide input to the mathematical models. The success of the mathematical model depends on the accuracy of the aquifer parameters. Analytical methods based on the mathematical models have been developed to determine aquifer parameters since the pioneering work of Theis in 1935. ANN models have the potential to be utilized in order to minimize the possible error risks on the determination of aquifer parameters. Another important study on the aquifer parameters is the estimation of aquifer parameters for the entire aquifer. In practice, several statistical and mathematical models have been used to predict the unknown parameters. Such methods require a high level of effort and can cause errors in the aquifer approximation. ANN models, again, can be applied to estimate the unknown values of an aquifer as an alternative to several statistical and mathematical models.

In this research, ANN based approaches have been developed for the aquifer parameter estimation. Existing ANN models in the literature have been reviewed and subsequently applied to determine confined aquifer parameters such as transmissivity and storativity as well as the development of an iterative ANN model for estimating leaky confined aquifer parameters like transmissivity, storativity and the vertical hydraulic conductivity. The performance of ANN models for both aquifer types was compared to traditional curve matching procedures.

The ANN based aquifer estimation potential was also investigated in the data interpolation problem. A numerical experiment has been conducted to estimate transmissivity values of a hypothetical aquifer. The experiment, using an ANN model in addition to the traditional mathematical models such as Radial Basis Function and Ordinary Kriging was developed to predict transmissivity values at unsampled points.

The thesis document is organized in eight chapters. The objectives of this study are elaborated in Chapter 1. A brief introduction to the ANN concept is presented in Chapter 2. The Back Propagation Algorithm which has been used in this study is explained in Chapter 3. The fundamentals of confined aquifer and pumping tests are given in Chapter 4. The development and application of the ANN model to determine aquifer parameters are discussed in the Chapter 5 and 6, respectively. In Chapter 7, the second goal of this research is demonstrated by a numerical experiment. The conclusions are summarized in Chapter 8.

2. ARTIFICIAL NEURAL NETWORKS

2.1. Introduction

There is a general view point that the human brain and nervous system are the most sophisticated computer and information-processing machine ever devised. Because of this belief, there is a tendency to compare the brain to the computers. However, such comparisons are fraught with difficulties and misconceptions. Probably, the most fundamental difference between brains and computers is that today's computers operate by performing often sequential instructions from an input program, while no clear analogy of a program appears in human brains. In addition, the brain is massively parallel and interwoven, whereas programming of this kind is extremely difficult for computer software developers.

The human brain is a very complicated system which is able to think, recognize, remember and analyze. The most distinguished feature of the human brain with respect to other organs is that the human brain is the source of the conscious and for that reason it has the ability of “learning”. Although, it is impossible to claim that functioning of the human brain is definitely known, it has been known that the biological neuron cells are responsible for data transfers to the brain. A typical biological nerve cell called a neuron is basically comprised from a cell body, a signal carrying axon, and a cluster of dendrites as shown in the Figure 2.1. Neurons behave as microprocessors in computer terminology. Each neuron receives the signal of many other neurons throughout the dendrites. If this signal is strong enough, the neuron is activated and produces an output signal which is transmitted through the axons.

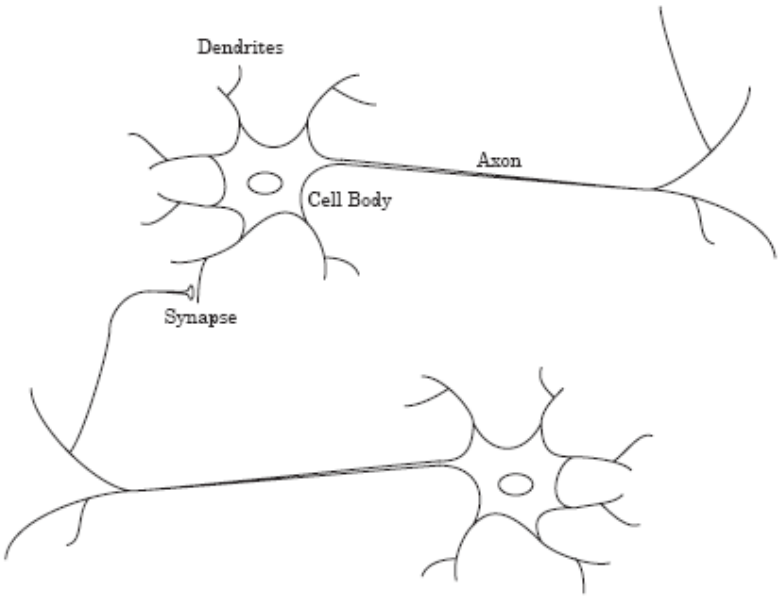

Figure 2.1. Biological Neuron

Artificial Neural Networks (ANNs), also referred as neural networks in the literature, are the mathematical or conceptual models of the biological neurons. ANNs can be simply defined as the mechanisms developed by inspiring structure of biological nerve cell that work with input-output responds. One of the most comprehensive definitions of ANN in the literature is that ANNs are distributed, adaptive, generally nonlinear learning machines composed of many different processing elements (or neurons) (Principe *et al.*, 2000).

ANNs have been widely used in many branches of science and engineering for the last few decades. First studies on neural network concept date back to the pioneering work of McCulloh and Pitts in 1943. In their research, McCulloh and Pitts created a neuron model that was the mathematical imitation of the functioning of the biological neuron. Moreover, it has been proved that artificial neuron could, in principle, compute any arithmetic or logical function (Hagan *et al.*, 1996). In Figure 2.2, a schematic view of the artificial neuron superimposed on the biological neuron is shown.

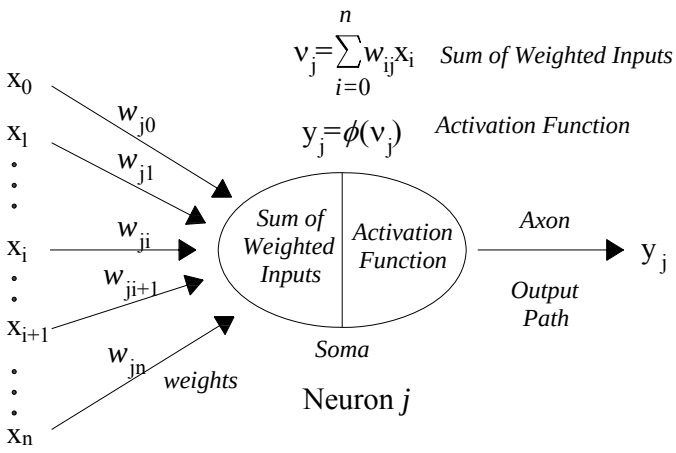

Figure 2.2. Schematic view of the artificial neuron

According to Figure 2.2, each neuron has two significant missions as signal receiving from input variables and data processing. The net function determines how to receive the signals from the input layer by synaptic weights (or simply called weights). Signals are transferred to neuron as

$$\upsilon_j = \sum_{i=1}^{N} w_{ji} x_i + b_j \tag{2.1}$$

where υ_j is the net function of j^{th} neuron, x_i is the i^{th} pair of the input vector, w_{ji} is the synaptic weights of the j^{th} neuron linked to the i^{th} pair of the input vector, and b_j is the bias or threshold term of the j^{th} neuron. Equation (2.1) can be rewritten for simplicity of the calculation as

$$\upsilon_j = \sum_{i=0}^{N} w_{ji} x_i \tag{2.2}$$

In Equation (2.2), the bias (also known as threshold) unit is implicitly represented by w_{j0} and its corresponding input signal x_0 value is equal to -1. However, it is not

necessary to transfer input signals to the neuron as the summation of the weighted input pairs. In Table 2.1 some types of the net functions has been exhibited.

Table 2.1. Summary of Net Functions (Hu and Hwang, 2002)

Net Function	**Formula**	**Comments**
Linear	$\upsilon_j = \sum_{i=1}^{N} w_{ji} x_i + b_j$	Most commonly used
High Order (2^{nd} Order Formula exhibited)	$\upsilon_j = \sum_{k=1}^{N} \sum_{i=1}^{N} w_{ji} x_i x_k + b_k$	The weighted linear combination of higher order polynomials term of input variables
Delta	$\upsilon_j = \prod_{i=1}^{N} w_{ji} x_i$	Seldomly used

The second mission of a neuron is to transfer the processed data as an output signal. For this reason, net function is activated by a transfer function which has great importance for determining how to transfer the received data. In general, neural network ability depends on the transfer function type. A transfer function is generally a non-linear function and has the derivates for learning procedures. In the neural network literature, there are many transfer functions which satisfy these features. Table 2.2 represents the most commonly used transfer functions.

Table 2.2. Some Transfer Functions

Transfer Function	**Formula**	**Derivative**	**Output Range**
Linear	$\varphi(\upsilon) = m\upsilon + c$	m	$(-\infty, \infty)$
Sigmoid	$\varphi(\upsilon) = \dfrac{1}{1+\exp(-\upsilon)}$	$\varphi(\upsilon)\left(1-\varphi(\upsilon)\right)$	$(0,1)$
Hyperbolic Tangent Sigmoid	$\varphi(\upsilon) = \dfrac{\exp(\upsilon)-\exp(-\upsilon)}{\exp(\upsilon)+\exp(-\upsilon)}$	$1-\left(\varphi(\upsilon)\right)^2$	$(-1,1)$
Gaussian Radial Basis	$\varphi(\upsilon) = \exp\left[-\dfrac{\lVert\upsilon - m\rVert^2}{\sigma^2}\right]$	$-\dfrac{2(\upsilon-m)\varphi(\upsilon)}{\sigma^2}$	$(0,1)$

2.2. Neural Network Topologies

ANNs can be examined in two major categories in terms of signal flow. Figure 2.3 describes the signal flow of the ANN models. In the left figure, signal flow in network is feed-forward. Such network types do not contain a feedback connection from the output signal as shown in the right figure.

(a) Acyclic topology (b) Cyclic topology

Figure 2.3. Illustration of (a) an acyclic graph and (b) a cyclic graph (Hu and Hwang, 2002)

a) **Feed-forward Neural Networks (FFNNs):** The data flow from input unit to output unit is strictly feed forward. The data processing can extend over single or multiple layers of units (Kröse and Smagt, 1996). FFNNs are often used for

nonlinear mapping of input-output relation (Hu and Hwang, 2002) FFNN can be modeled as a single layer perceptron or multi-layer perceptrons according to specific problems. FFNNs need an external teacher which constructs non-linear relation between input and outputs.

b) **Recurrent Neural Networks (RNNs):** The second category of the network classification in terms of data flow is recurrent neural network structures which contain the feedback loop. Due to the presence of the feedback connection, recurrent networks are used to approximate the nonlinear dynamic systems which have internal memory. (Hu and Hwang, 2002) RNNs are more complex network models rather than the feed-forward neural networks since there can not be an external teacher.

2.3. Feed-Forward Neural Network Architectures

ANNs are the layered structure which is responsible for data transfer. A typical neural network is comprised of an input layer, one or more hidden layers and an output layer. Input layer contains the input variable vector that will be introduced to the network. The hidden layer which is built of the neurons is the processor layer of the network and it is responsible for transfer of data. Each neuron in the hidden layer has the net and transfer functions. An output layer produces the network output by processing the data which are transferred from the hidden layer. The first neural network model developed by McCulloch and Pitts had composed of only one neuron which was connected to the input layer. At approximately the same time, Rosenblatt (1958) also proposed the first generation neural networks known as the perceptron. In the perceptron model, a single neuron with a weighted linear net function and transfer function is utilized (Hu and Hwang, 2002). The difference between the two models is that the former model suggested the mathematical explanations of a neuron whereas the latter proposed a learning algorithm which provides convergence when the training samples are linearly separable.

2.3.1. Single Layer Perceptron Models

When the input signal is transferred over a single layer and its single neuron, such a network is named as a single layer feed-forward network. This single neuron architecture is the same as the McCulloch and Pitt's neuron model. Single layer perceptron models have one or more input pairs in the input layer. In Figure 2.4, a single layer perceptron models are classified according to input pair vector.

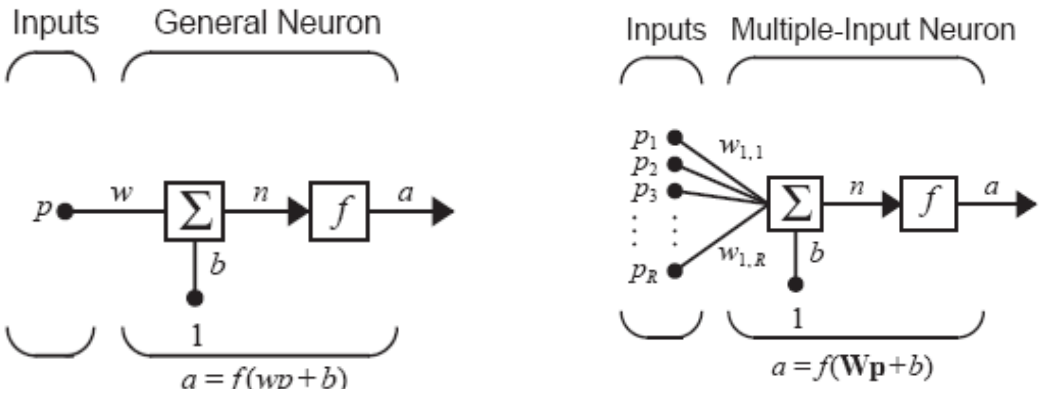

Figure 2.4. (a) Single-Input Neuron Model (b) Multiple-Input Neuron Model (Hagan *et. al*, 1996)

2.3.2. Multi-Layer Perceptron (MLP) Models

Single layer models are insufficient to map non-linear relation between the input and outputs, network models with multi-layer neurons are often employed. A Multilayer Perceptron (MLP) neural network model consists of layered network of neurons as shown in Figure 2.5. Each neuron in an MLP has nonlinear transfer function that is often continuously differentiable (Hu and Hwang, 2002).

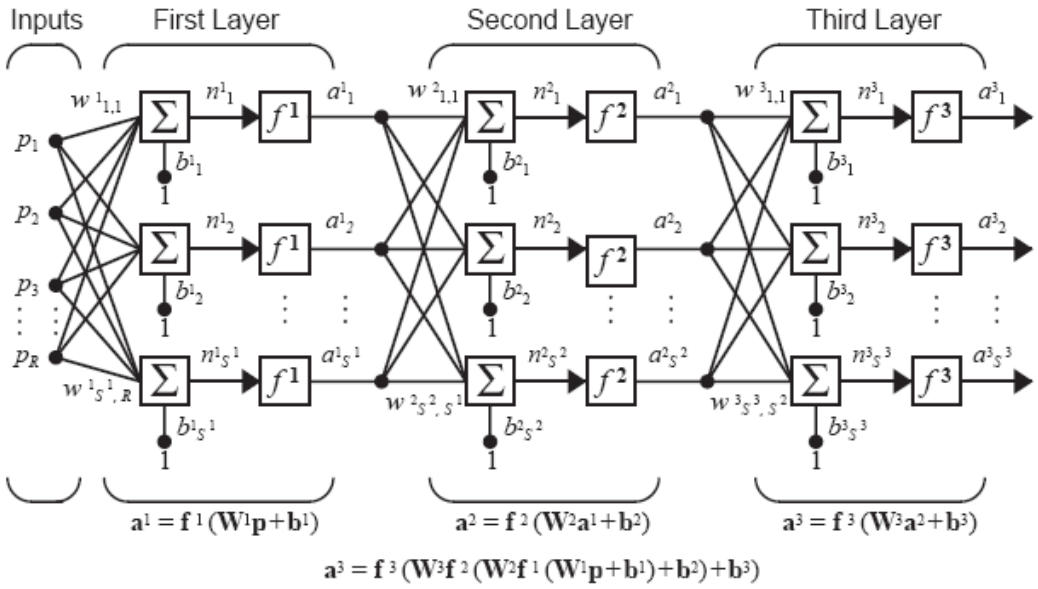

Figure 2.5. Schematic illustration of MLP (Hagan *et al.,* 1996)

2.4. Learning Strategies

In the neural network literature, there are two fundamental learning strategies which depend on the neural network architecture.

a) **Supervised Learning:** Supervised learning or sometimes called active learning implies the availability of an external agent that means having knowledge of a set of input- output examples (Haykin, 1994). A learning algorithm such as back-propagation algorithm is utilized to perform supervised learning in most feed-forward MLPs.

b) **Unsupervised Learning:** In unsupervised learning strategy, there is no external teacher which defines the environment to perform learning process. In other words, there are no specific examples of the function to be learned by the network. Self-organizing networks are trained in the manner of unsupervised learning by employing a special learning algorithm such as Hebbian Learning algorithm (Kröse and Smagt, 1996).

2.5. Important Aspects of ANN Modeling

In the previous sections, a brief introduction to ANN has been described to construct the background about ANN models. In this section, some important features of ANN models which lead to better approximation will be elaborated. It is important to note that these explanations have been mainly prepared for the back-propagation algorithm which was used in this study.

There is no theoretical formula or fixed rules for developing ANN model. Although the previous studies have a great impact on ANN modeling, the success of ANN model mostly depends on available data of the problem (ASCE, 2000 a.).

2.5.1. Selection of Input and Output Variables

ANNs are powerful models that can learn the nonlinear relation of input and output responses. The selection of an appropriate input vector for the problem is important for constructing correct relationship with desired target data. In general, ANNs are misunderstood as black box model; however, neural networks can not process any relation between the input and target variables if these are irrelevant (ASCE, 2000 a.). A sensitivity analysis can be performed to determine the relative importance of a variable when sufficient data is available (Maier and Dandy, 1996). One of other key points is that the selected input and target pairs should be representative for the problem domain.

2.5.2. Preparing Input and Target Vectors for Training

ANNs utilize the transfer function to process the signal flow between the input and output layer. Most commonly used transfer functions presented in Table 2.2 produce results based on the output range. For that reason, input and target pairs of training patters should be normalized into the unit range of transfer function before they are given to the network as a teaching signal. For instance, training patterns can be normalized to the range of [0, 1] or [-1, 1] when log-sigmoid transfer function or the tangent hyperbolic transfer

function, respectively, are utilized. When the network is trained to the desired error level, outputs of the network would be back-transformed to their original range.

When the large size training patterns are available, Principal Component Analysis (PCA) is recommended to reduce the dimensionality of data while retaining as much as possible the variation present in the original data set.

2.5.3. Initializing Network Parameters

For an initial step of learning, the network parameters such as weights, bias, momentum and learning rate should be initialized. The weights and bias values are initially assigned small random variables as (-0.3 ~ 0.3) (ASCE, 2000). Haykin (1994) suggested the initial values for the network parameters as given in Table 2.3.

Table 2.3. Initial Network Parameters

Parameters	Value
Number of Hidden Layer	1
Learning Rate	0.1
Momentum Constant	0.5

2.5.4. Determining ANN Configuration

As elaborated earlier, there is no fixed rule to determine the ANN structure. The number of neurons in the hidden layer is determined by a trial-error produce. In practice, two methods to find the optimal number of neurons in the hidden layer are preferred. As a first case, a small network structure is constructed and trained by the training data set. Adding the new neurons, the optimal size of network that satisfies the training error tolerance is then obtained. Instead of growing manner of network, some practitioners prefer to trim the network size using by large size network. In this case, the number of neurons in the hidden layer is reduced gradually until the minimum number of neurons that satisfy the chosen error tolerance (Kubat, 2000).

2.6. Applications of ANN in Hydrology

From a civil engineering point of view, Flood and Kartman (1994 a., b.) provided insight into the usage and potential application of ANNs in a two-paper series. In the first paper, a general introduction of the ANN concept was explained and demonstrated in a simple structural analysis as an example. In the subsequent paper, a number of applications in which ANNs were utilized in solving engineering problems were described to demonstrate the capability of the technique.

Pezeshk *et al.* (1996) used the FFBN to complement manual log interpolation. In their study, a neural network was developed to analyze geophysical well logs and to produce information on the subsurface strata classification. In addition, an overview has been presented on the neural network optimization techniques, limitations, and the strength of the approach in well-log interpolation.

Morshed and Kaluarachchi (1998) suggested that ANN could be used to simulate the groundwater flow and contaminant transport (GFCT) response explicitly. In their research, a one dimensional unsaturated flow and contamination problem was developed to investigate the effects of specific GFCT parameters that were simulated by ANN on overall results.

Nayak *et al.* (2006) reported the potential of neural computing techniques for forecasting groundwater levels by developing ANN models for a shallow aquifer of Central Godavari Delta System in India. In this study, inputs for ANN model were determined by using a combined approach which uses the field knowledge and statistical analysis of the data series. The reported results indicated that ANN model is an effective toll for monthly groundwater levels forecasting.

.

El Tabach *et al.* (2006) proposed a metamodel based on artificial neural networks for estimating the depth of the contaminated zone and the volume of pollutant infiltration in the soil in a two layer soil (a silty cover layer protecting a chalky aquifer) after a pollutant discharge at the soil surface. In this research, the ANN database was generated using a specific non-aqueous phase liquids (NAPL) simulator. For each case, the extent of contamination was computed as a function of cover layer permeability and thickness, water

table depth and soil surface–pollutant contact time. The ANN performance was also compared with a metamodelling method using multi-linear regression approximation.

Coppola *et al.* (2003) developed an ANN model for accurately predicting transient water levels in a complex multilayered ground-water system under variable state, pumping, and climate conditions. In that paper, the ANN was trained to predict transient water levels at 12 monitoring well locations screened in different aquifers in response to changing pumping and climate conditions. The trained ANN was then validated with ten sequential seven-day periods, and the results were compared against both measured and numerically simulated ground-water levels.

Singh and Datta (2004) used an ANN model to simultaneously solve the problems of estimating unknown groundwater pollution sources and estimating unknown hydrogeologic parameters such as hydraulic conductivity, porosity, and dispersivity. The universal function approximation property of a multilayer, feed-forward ANN was utilized to estimate temporally and spatially varying unknown pollution sources, as well as to provide a reliable estimation of unknown flow and transport parameters. The authors also reported that the proposed methodology performs reasonably well even with large measurement errors in their work.

Uddameri (2007) used a FFBPN forecast monthly and quarterly time-series water levels at a well that taps into the deeper Evangeline formation of the Gulf Coast aquifer in Victoria, Texas in the USA. The root mean square error, mean absolute deviation and correlation coefficient were noted to be 1.40, 0.33 and 0.77 m, respectively, for an evaluation data set of quarterly measurements and 1.17, 0.46, and 0.88 m for an evaluative monthly data set not used to train or test the model. The author also claimed that error analysis of the ANN model were better than statistical regression techniques.

Yang and Chan (2005) utilized an ANN model for simulating velocity profiles, velocity contours and estimating the discharges accordingly. Based on that study, the velocity profiles were measured by an acoustic doppler velocimeter in the open channel of the Chihtan purification plant, Taipei, with different discharges at fixed measuring section

and different depths. The results demonstrated that the velocity profiles that were modeled by ANN nicely fit to experimental velocity profiles of the condition investigated.

Kholgi and Hosseini (2006) employed the ANN model for interpolating transmissivity values in a case study. The efficiency of the Adaptive Network based Fuzzy Inference System (ANFIS), ANNs and Ordinary Kriging was investigated for interpolation of transmissivity in an unconfined aquifer. It was also shown that ANFIS model was more efficient to estimate the transmissivity in comparison with the ANN and Kriging models.

Balkhair (2002) developed an ANN model to estimate aquifer parameter values, namely transmissivity and storage coefficient, from pumping test data for a large diameter well. The ANN was trained to map time–drawdown and well diameter data (input vector) to its corresponding transmissivity and storage coefficient values (output vector). Based upon a pre-specified range of aquifer parameters, the input vectors were generated from the analytical solution of Papadopulos and Copper for large diameter well in a homogeneous, isotropic, non-leaky confined aquifer. The results obtained with the ANN indicated that estimated aquifer parameters were in good agreement with traditional type curve matching method.

Lin and Chen (2006) introduced a new ANN model to estimate aquifer parameter values. The authors criticized the Balkhair (2002) model and demonstrated the advantages of the proposed algorithm in terms of accuracy, computation time and the estimation range over the existing model. Additionally, the proposed algorithm was able to estimate the aquifer parameters better than the conventional type curve methods.

Samani *et al.* (2007) modified the Lin and Chen ANN model that was developed to determine aquifer parameters for a confined aquifer. The researchers suggested that utilizing the Principal Component Analysis to reduce the input vector dimension provides better approximation results and lesser computation time to train the neural network model in their paper. They also claimed that the proposed ANN appears to be a simpler and more accurate alternative to the type curve-matching techniques and previous ANN methods.

There are also some editorial articles which introduce the neural network concepts in hydrology. Adeli (2001) reviewed the ANN applications in civil engineering areas, structural engineering, traffic engineering, environmental and water resources engineering, to name a few, between 1989 and 2000. Another editorial report on the application of ANN in Hydrology was published by the ACSE Task Committee (2000 a., b.). In this two paper series, the important structural and functional aspects of ANN were introduced and the role of ANNs in various branches of hydrology has been examined. In the second part of this series, applications of ANN in rainfall-runoff modeling, stream-flow modeling, water quality modeling, groundwater modeling, and precipitation modeling were presented. The future applications of ANN in hydrology were also discussed.

3. BACK PROPAGATION ALGORITHM

The Back Propagation Algorithm (BPA) is probably one of the most popular learning algorithms for neural networks (ASCE, 2000 a.). It has been used in an application variety of non-linear engineering optimization problems and universal function approximation. The first discovery of BPA was demonstrated by Werbos in his Ph.D thesis in 1974. Although Werbos gave the first description of BPA, it was popularized with its rediscovery by Rumelhart, Hinton and Williams and their most famous seminar book entitled *Parallel Distributed Processing* (Haykin, 1994).

The fundamental logic behind the BPA is that the errors that are produced in forward phase are propagated back toward the input layer by adjusting the synaptic weights using the steepest gradient method. This forward and backward process continues iteratively until the training error is minimized.

3.1. Derivation of Back Propagation Algorithm[1]

Error of the output neuron j for iteration n is described by Equation (3.1),

$$e_j(n) = d_j(n) - y_j(n) \tag{3.1}$$

where $d_j(n)$ is the target data for iteration n and $y_j(n)$ is the output value of the network at iteration n.

The instantaneous error value $\varepsilon(n)$ is correspondingly the sum squared errors which is obtained by summing $0.5e_j^2(n)$ over all neurons in the output layer. Therefore, the instantaneous sum of squared errors of the network can be shown in Equation (3.2)

[1] Notations and algorithm derivation were adapted from Neural Networks: A comprehensive foundation by Simon Haykin.

$$\varepsilon(n) = \frac{1}{2}\sum_{j \in C} e_j^2(n) \tag{3.2}$$

where the set C involves all neurons in the output layer of the network. If the size of training input data set is N, the average error can be calculated by arithmetic averaging of the instantaneous sum of squared error with respect to training data set size N, as shown by Equation (3.3)

$$\varepsilon_{av} = \frac{1}{N}\sum_{n=1}^{N} \varepsilon(n) \tag{3.3}$$

While the instantaneous sum of squared error indicates the error of each neuron at output layer, average squared error can be utilized to obtain overall error of network for entire size of the training input set. Therefore, instantaneous and average error terms depend on the all free parameters (i.e. connection weights, thresholds or bias) of the networks. The aim of the training algorithm is to minimize average error term and then adjust these free parameters of the network.

The transfer signal at j^{th} neuron in first adjacent hidden layer which is caused by i^{th} neuron in the input layer is shown in Equation (3.4);

$$\upsilon_j(n) = \sum_{i=0}^{p} w_{ji}(n) y_i(n) \tag{3.4}$$

where p is size of the input set excluding the bias. In Equation (3.4), the bias unit is implicitly represented by w_{j0} and its corresponded input y_0 value is equal to -1. Schematic illustration of signal flow of the one hidden layer network is described in Figure 3.1. In this Figure, output of the hidden layer neuron is shown by υ_j, and then its signal is transferred to output layer neuron y_j at iteration n with an activation function $\phi(\cdot)$. Hence the value of output neuron j at iteration n is expressed in Equation (3.5)

$$y_j(n) = \phi_j\left(\upsilon_j(n)\right) \tag{3.5}$$

Considering Least Mean Square (LMS) algorithm, the back propagation algorithm corrects the synaptic or connection weights that are proportional to instantaneous gradient $\frac{\partial \varepsilon(n)}{\partial w_{ji}(n)}$ in a similar manner.

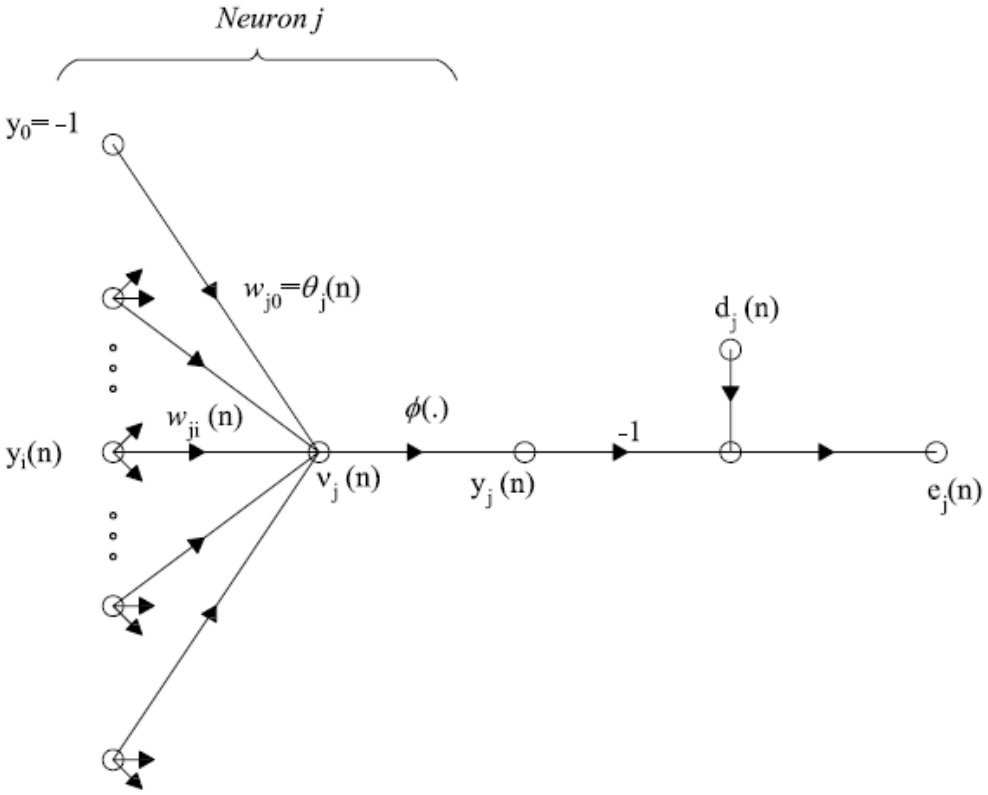

Figure 3.1. Schematic Illustration of Signal Flow of the One Hidden Layer Network

Applying the chain rule to instantaneous gradient $\frac{\partial \varepsilon(n)}{\partial w_{ji}(n)}$;

$$\frac{\partial \varepsilon(n)}{\partial w_{ji}(n)} = \frac{\partial \varepsilon(n)}{\partial e_j(n)} \frac{\partial e_j(n)}{\partial y_j(n)} \frac{\partial y_j(n)}{\partial \upsilon_j(n)} \frac{\partial \upsilon_j(n)}{\partial w_{ji}(n)} \tag{3.6}$$

Considering Equation (3.4), differentiation both sides of it with respect to $w_{ji}(n)$;

$$\frac{\partial \upsilon_j(n)}{\partial w_{ji}(n)} = y_i(n) \tag{3.7}$$

In a similar way, differentiating Equation (3.5), (3.1), and (3.2) with respect to $\upsilon_j(n)$, $y_j(n)$, $e_j(n)$ respectively, results are as follows:

$$\frac{\partial y_j(n)}{\partial \upsilon_j(n)} = \phi_j'\left(\upsilon_j(n)\right) \tag{3.8}$$

$$\frac{\partial e_j(n)}{\partial y_j(n)} = -1 \tag{3.9}$$

$$\frac{\partial \varepsilon(n)}{\partial e_j(n)} = e_j(n) \tag{3.10}$$

Thus, combining Equation (3.7), (3.8), (3.9), and (3.10) in Equation (3.6) yields

$$\frac{\partial \varepsilon(n)}{\partial w_{ji}(n)} = -e_j(n)\phi_j'\left(\upsilon_j(n)\right) y_i(n) \tag{3.11}$$

The synaptic weight of j^{th} neuron in the hidden layer due to i^{th} neuron in the input layer $w_{ji}(n)$ is adjusted according to *delta rule*

$$\Delta w_{ji}(n) = -\eta \frac{\partial \varepsilon(n)}{\partial w_{ji}(n)} \tag{3.12}$$

where η is learning rate parameter. The minus sign in Equation (3.12) shows that gradient descent in weight space. Substituting Equation (3.11) in Equation (3.12)

$$\Delta w_{ji}(n) = \eta \delta_j(n) y_i(n) \tag{3.13}$$

where $\delta_j(n)$ is local gradient at iteration n. Equation (3.14) defines the local gradient.

$$\begin{aligned} \delta_j(n) &= -\frac{\partial \varepsilon(n)}{\partial e_j(n)} \frac{\partial e_j(n)}{\partial y_j(n)} \frac{\partial y_j(n)}{\partial \upsilon_j(n)} \\ &= e_j(n)\phi_j'\left(\upsilon_j(n)\right) \end{aligned} \tag{3.14}$$

Local gradient describes the required weight changes in synaptic weight space. Otherwise stated, local gradient indicates role of the synaptic weights on the error between target output and network output.

A critical question can come to the mind at this point, what would be happen if the neuron j is not output node? There are two choices to answer this question. In the first case, if the neuron j is located at the output layer of the network, all formulations that have been derived are still valid. On the other hand, when the neuron j is located at the hidden layer of the network, there would be no desired response for that neuron to calculate error signal for second case. Figure 3.2 shows such network topology where the neuron j is located at the hidden layer of the network.

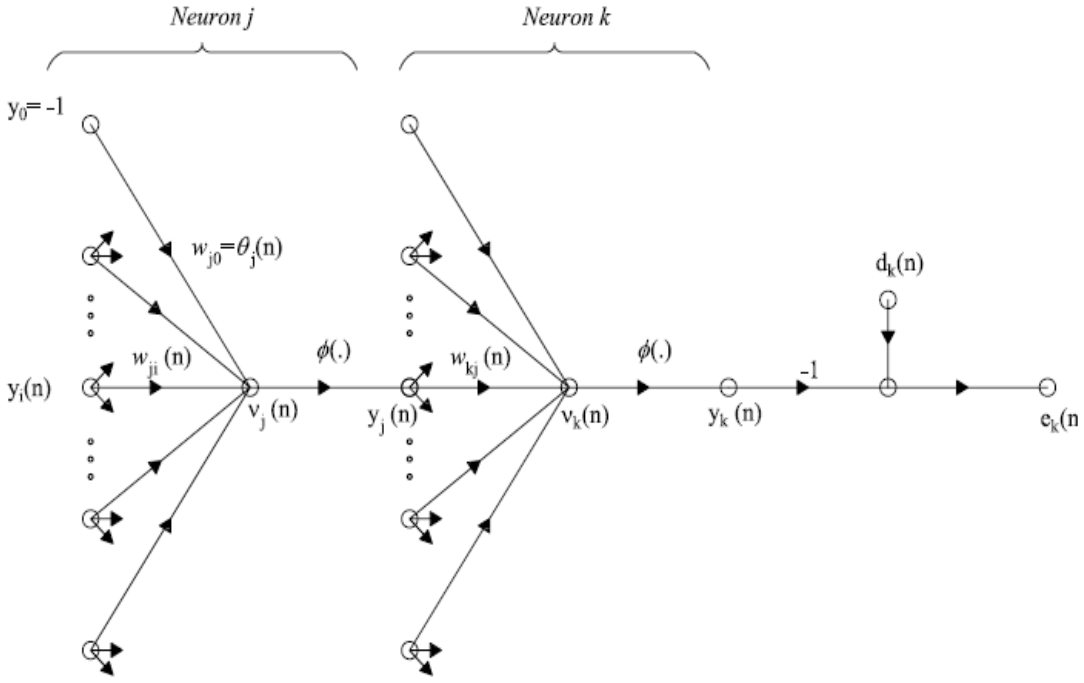

Figure 3.2. Schematic Illustration of Signal Flow of the Two Hidden Layers Network

Error of the output neuron k for iteration n is defined as follow

$$\varepsilon(n) = \frac{1}{2}\sum_{k \in C} e_k^2(n) \tag{3.15}$$

Equation (3.15) is rewritten form of Equation (3.2) assuming that neuron k is the output node. It is important to note that local gradient of hidden neuron j which was defined in Equation (3.14) should be expressed according to Figure 3.2. The reason of this arrangement is due to while neuron j was the output node in Figure 3.1, neuron j is located in the hidden layer, and therefore there is no corresponded target error term which network produces for this neuron. Equation (3.16) shows the mathematical formulation of described condition.

$$\begin{aligned}\delta_j(n) &= -\frac{\partial \varepsilon(n)}{\partial y_j(n)}\frac{\partial y_j(n)}{\partial \upsilon_j(n)} \\ &= -\frac{\partial \varepsilon(n)}{\partial y_j(n)}\phi_j'\left(\upsilon_j(n)\right)\end{aligned} \tag{3.16}$$

It can be inferred from Equation (3.15) and (3.16) that local gradient of hidden neuron j is related with error term of the next hidden layer. The aim of the following formulations is to prove this argument.

Based on Figure 3.2, error of the output neuron k for iteration n is found as

$$e_k(n) = d_k(n) - y_k(n) \tag{3.17}$$

where $y_k(n)$ is the output value of the network shown below:

$$y_k(n) = \phi_k\left(\upsilon_k(n)\right) \tag{3.18}$$

Rearranging Equation (3.17) yields

$$e_k(n) = d_k(n) - \phi_k\left(\upsilon_k(n)\right) \tag{3.19}$$

The internal activity of neuron k ;

$$\upsilon_k(n) = \sum_{j=0}^{q} w_{kj}(n) y_j(n) \tag{3.20}$$

where q is size of the input set excluding the bias. The bias unit is implicitly represented by w_{k0} and its corresponded input y_0 value is equal to -1 as the same manner of Equation (3.4).

The changes in the error term of $\varepsilon(n)$ with respect to signal of the hidden neuron j can be expressed as follow

$$\frac{\partial \varepsilon(n)}{\partial y_j(n)} = \sum_k e_k(n) \frac{\partial e_k(n)}{\partial y_j(n)} \tag{3.21}$$

Applying chain rule to Equation (3.21) yields

$$\frac{\partial \varepsilon(n)}{\partial y_j(n)} = \sum_k e_k(n) \frac{\partial e_k(n)}{\partial \upsilon_k(n)} \frac{\partial \upsilon_k(n)}{\partial y_j(n)} \tag{3.22}$$

Differentiating both sides of Equation (3.19) and (3.20) with respect to $\upsilon_k(n)$ and $y_j(n)$, respectively;

$$\frac{\partial e_k(n)}{\partial \upsilon_k(n)} = -\phi_k'\left(\upsilon_k(n)\right) \tag{3.23}$$

$$\frac{\partial \upsilon_k(n)}{\partial y_j(n)} = w_{kj}(n) \tag{3.24}$$

Combining Equation (3.23) and (3.24) in Equation (3.22) yields

$$\frac{\partial \varepsilon(n)}{\partial y_j(n)} = -\sum_k e_k(n) \phi_k'\left(\upsilon_k(n)\right) w_{kj}(n) \tag{3.25}$$

Rewriting Equation (3.25) using by the definition of local gradient $\delta_k(n)$ shown in Equation (3.14) with index k substituted for j ;

$$\frac{\partial \varepsilon(n)}{\partial y_j(n)} = -\sum_k \delta_k(n) w_{kj}(n) \tag{3.26}$$

Finally, combining Equation (3.16) and (3.26), the local gradient $\delta_j(n)$ for hidden neuron j can be shown as follow:

$$\delta_j(n) = \phi_k'\left(\upsilon_k(n)\right) \sum_k \delta_k(n) w_{kj}(n) \tag{3.27}$$

As a consequence, weight changes over the iterations can be written as

$$w_{ji}^L(n+1) = w_{ji}^L(n) + \eta \sum_{k=1}^{N} \delta_j^L(k) y_j^{L-1}(k) + \mu_c \left[w_{ji}^L(n) - w_{ji}^L(n-1) \right] \tag{3.28}$$

where L is the layer number, N is the total number of training data, n is the iteration number.

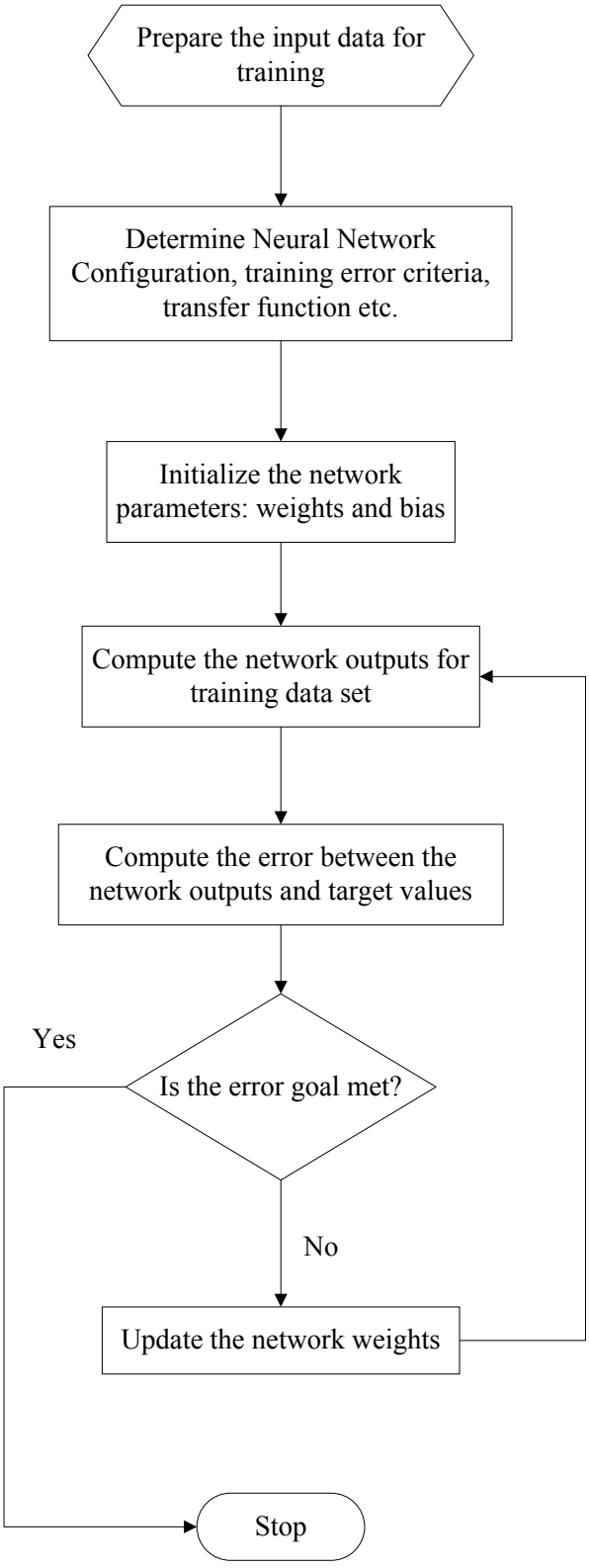

Figure 3.3. Network Training Flow Chart

3.2. Heuristic Modifications of Back Propagation Algorithm

3.2.1. Momentum Constant

Momentum constant is employed to prevent the oscillations in weight changes. Another benefit of momentum constant is that it can speed up the training in the very flat region of error surface (ASCE, 2000 a.). When successive gradient vectors form of zig-zag search pattern, the effective gradient direction is adjusted by this momentum term. Therefore, it helps minimize the mean square error (Hu and Hwang, 2002). The momentum constant value is assigned in the range of [0, 1]. When the momentum constant value approaches to zero, a weight change only depends on the gradient. On the other hand, if the momentum constant is set as 1, the weight change will equal to the last weight change, thus the gradient is ignored (Demuth et al, 2006). In practice, momentum constant can usually take the value in the range of [0.6, 0.9] (Hu and Hwang, 2002).

3.2.2. Learning Rate

A learning rate is used to increase the chance of avoiding the training process being trapped in local minima instead of global minima (ASCE, 2000). The performance of the BPA is very sensitive the appropriate setting of the learning rate. If the learning rate is selected as too high, the weight changes oscillate thus the network produces unstable results whereas the learning rate is set too small, training produce takes too long time to converge (Demuth et al, 2006). In general the learning rate is assigned as smaller values between 0 and 0.3 (Hu and Hwang, 2002).

3.2.3. The Levenberg-Marquardt Method

While the BPA is a steepest descent algorithm, the Lebenberg-Marquardt (LM) is an alternative method which is a modification of the classic Newton algorithm for finding the optimum solution to a minimization problem (Samani *et al*., 2007). Assuming that $H(x)$ is sum squares of error function presented as

$$H(x) = \sum_{i=1}^{N} e_i^2 \tag{3.29}$$

The derivative term in the steepest gradient method shown in Equation (3.11) can be rewritten as the derivative of $H(x)$ with respect to the synaptic weights in the Levenberg-Marquardt Method as follow

$$\Delta w = \left[J^T J + \lambda I \right]^{-1} J^T e \tag{3.30}$$

where J is the Jacobian matrix, λ is a scalar that controls learning process, e is the residual error vector. Jacobian matrix can be computed from following formula

$$J = \begin{bmatrix} \frac{\partial e_1}{\partial w_1} & \frac{\partial e_1}{\partial w_2} & \cdots & \frac{\partial e_1}{\partial w_N} \\ \frac{\partial e_2}{\partial w_1} & \frac{\partial e_2}{\partial w_2} & \cdots & \frac{\partial e_2}{\partial w_n} \\ \vdots & \vdots & \cdots & \vdots \\ \vdots & \vdots & \cdots & \vdots \\ \frac{\partial e_N}{\partial w_1} & \frac{\partial e_N}{\partial w_2} & \cdots & \frac{\partial e_N}{\partial w_N} \end{bmatrix} \tag{3.31}$$

In general, the LM algorithm provides a rapid and more accurate training rather the classical steepest descent algorithm (Hagan and Menhaj, 1994). In this study, FFBP was developed using by Matlab Neural Network Toolbox 5.0.

4. CONFINED AQUIFER AND PUMP TESTING

4.1. Flow in Completely Confined Aquifer

The studies on response of confined aquifers to pumping date back to Thiem's work in 1906. In his research, one of the first quantitative approaches was developed for steady state flow in confined aquifer. Theis (1935) suggested the first mathematical analysis of transient (non-steady) drawdown effects in a confined aquifer. Figure 4.1 represents the schematic illustration of a fully confined aquifer. The 2-D groundwater flow equation describing hydraulic head in a homogenous, isotropic confined aquifer shown in Figure 4.1 can be expressed in polar coordinates as;

$$\frac{\partial^2 h}{\partial r^2} + \frac{1}{r}\frac{\partial h}{\partial r} = \frac{S}{T}\frac{\partial h}{\partial t} \tag{4.1}$$

where h is the hydraulic head $[L]$, r is the radial distance from the pumping well $[L]$, t is time $[T]$, T is the transmissivity $\left[L^2/T\right]$ and S is the dimensionless parameter called storativity. Solution to Equation (4.1) was first derived by Theis in 1935 with the following assumptions (Fetter, 2001):

- All geologic formations are horizontal and have infinite horizontal extent
- The potentiometric surface of the aquifer is horizontal before the pumping test
- All changes in the position of the potentiometric surface are due to the effect of the pumping well alone.
- Aquifer is homogeneous and isotropic, all flow is radial toward the well
- Groundwater flow is horizontal and Darcy's law is valid
- The pumping well has an infinitesimal diameter and fully penetrating, pumping rate is constant and there is no recharge source to the aquifer

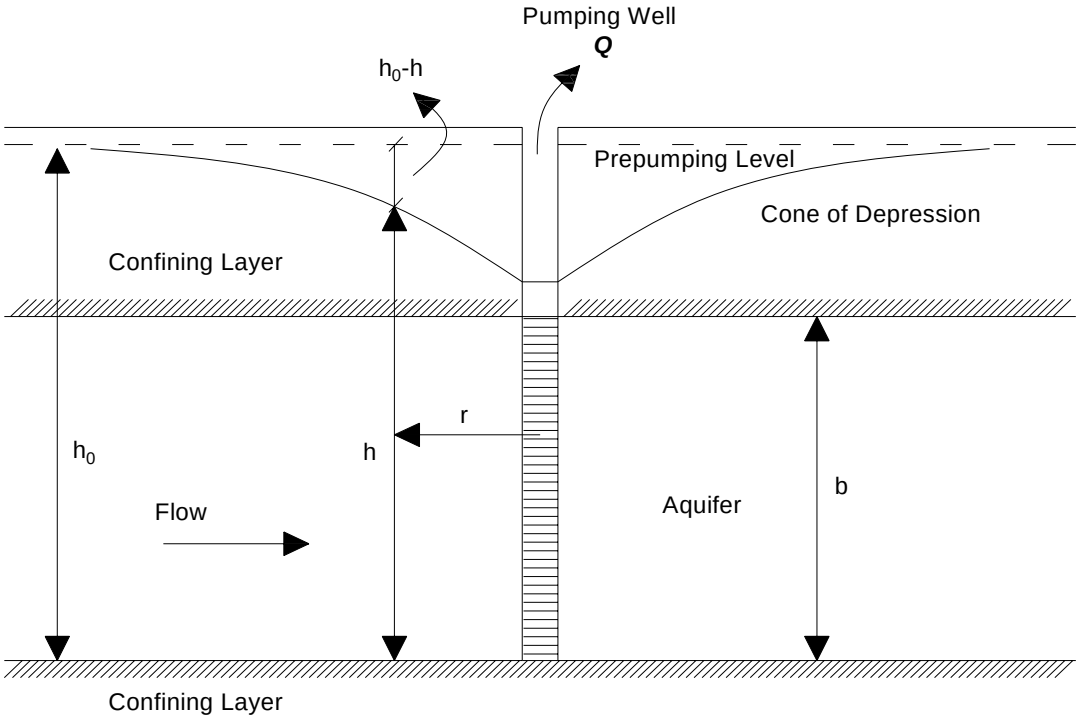

Figure 4.1. Schematic Illustration of a Fully Confined Aquifer

The following initial and two boundary conditions respectively are necessary to find the analytical solution;

$$\begin{aligned} &\text{Initial Value} && h(r,0) = h_0 \\ &1^{\text{st}}\text{ Boundary Condition} && h(\infty,t) = h_0 \\ &2^{\text{nd}}\text{ Boundary Condition} && \lim_{r\to 0}\left(r\frac{\partial h}{\partial r}\right) = \frac{Q}{2\pi T} \end{aligned} \tag{4.2}$$

The first condition in Equation 4.2 implies that at time zero and any distance r from the pumping well, the head is equal to pre-pumping level h_0. The second condition means that at an infinite radius for all time the hydraulic head has an unchanged value denoted as h_0. Finally, the last condition requires that the pumping well has a constant withdrawal rate (Schwartz and Zhang, 2003).

Solution to Equation (4.1), is also known as Theis or non-equilibrium equation, is presented as

$$h_0 - h = s = \frac{Q}{4\pi T} W(u) \tag{4.3}$$

where s is the drawdown $[L]$, Q is the pumping rate $\left[L^3/T\right]$, T is the transmissivity of the aquifer. The well function $W(u)$ and the dimensionless variable u are presented as

$$W(u) = \int_u^\infty \frac{e^{-x}}{x} dx \tag{4.4}$$

$$u = \frac{r^2 S}{4Tt} \tag{4.5}$$

where r is the radial distance from pumping well, t is the time, T and S are the transmissivity and the storativity of the aquifer, respectively.

The exponential integration is given by the infinite series as (Domenico and Schwartz, 1990)

$$W(u) = \int_u^\infty \frac{e^{-x}}{x} dx = -0.577216 - \ln u + u - \frac{u^2}{2 \times 2!} + \frac{u^3}{3 \times 3!} - \frac{u^4}{4 \times 4!} + \ldots \tag{4.6}$$

Equation (4.6) can be written in summation notation as

$$W(u) = -0.577216 - \ln u + \sum_{i=1}^{\infty} \frac{(-1)^{i+1} u^i}{i \times i!} \tag{4.7}$$

Table 4.1. Values of W(u) for values of u (from Wenzel, 1942)

u	1	2	3	4	5	6	7	8	9
x 1	0.2194	0.0489	0.013	0.0038	0.0011	0.00038	0.00012	0.000038	0.000012
x 10^{-1}	1.82	1.22	0.91	0.70	0.56	0.45	0.37	0.31	0.26
x 10^{-2}	4.04	3.35	2.96	2.68	2.47	2.30	2.15	2.03	1.92
x 10^{-3}	6.33	5.64	5.23	4.95	4.73	4.54	4.39	4.26	4.14
x 10^{-4}	8.63	7.94	7.53	7.25	7.02	6.84	6.69	6.55	6.44
x 10^{-5}	10.94	10.24	9.84	9.55	9.33	9.14	8.99	8.86	8.74
x 10^{-6}	13.24	12.55	12.14	11.85	11.63	11.45	11.29	11.16	11.04
x 10^{-7}	15.54	14.85	14.44	14.15	13.93	13.75	13.60	13.46	13.34
x 10^{-8}	17.84	17.15	16.74	16.46	16.23	16.05	15.90	15.76	15.65
x 10^{-9}	20.15	19.45	19.05	18.76	18.54	18.35	18.20	18.07	17.95
x 10^{-10}	22.45	21.76	21.35	21.06	20.84	20.66	20.50	20.37	20.25
x 10^{-11}	24.75	24.06	23.65	23.36	23.14	22.96	22.81	22.67	22.55
x 10^{-12}	27.05	26.36	25.96	25.67	25.44	25.26	25.11	24.97	24.86
x 10^{-13}	29.36	28.66	28.26	27.97	27.75	27.56	27.41	27.28	27.16
x 10^{-14}	31.66	30.97	30.56	30.27	30.05	29.87	29.71	29.58	29.46
x 10^{-15}	33.96	33.27	32.86	32.58	32.35	32.17	32.02	31.88	31.76

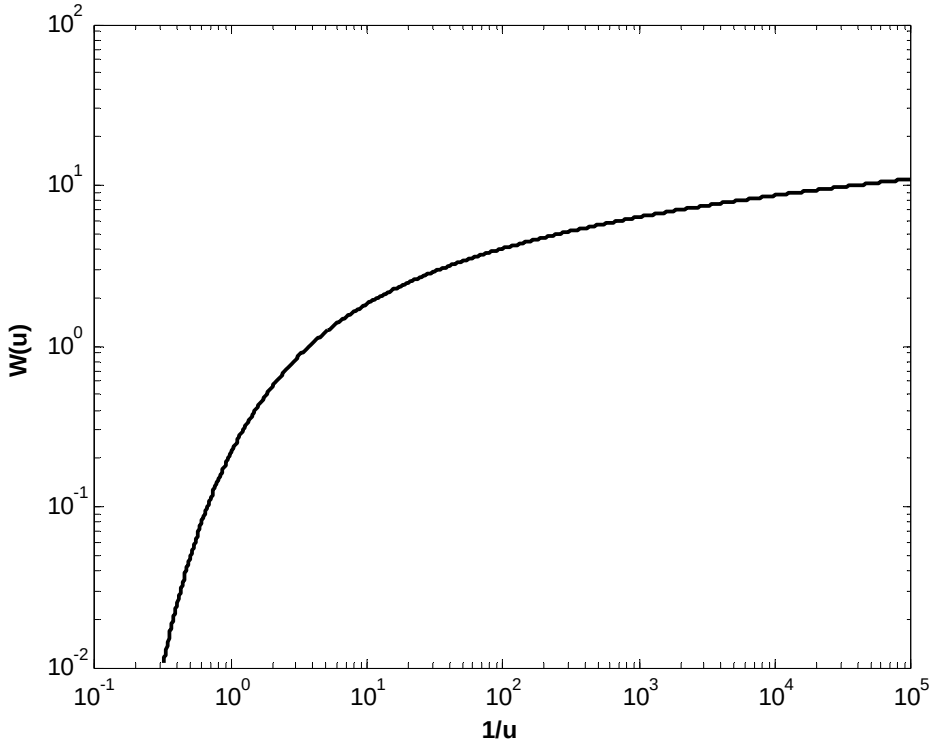

Figure 4.2. Values of W(u) plotted against values of 1/u

Pumping test plays a key role in groundwater engineering to determine the confined aquifer parameters. A variety of aquifer testing approaches are available in the literature. Theis Type-Curve Method is the one of the most popular techniques which relies on a curve matching procedure involving the following steps;

i. The well function $W(u)$ versus $1/u$ is plotted on log-log scale. The well function $W(u)$ can be generated using Equation (4.6) or copied from the textbooks available.
ii. A matching point that serves a reference point is defined on the type curve and can be located anywhere on the graph. In general, the matching point is chosen as $W(u)=1$ on the y-axis of the graph while $1/u$ can take 10, 100 or 1000 on x-axis of the type curve graph for the sake of simplicity in aquifer parameters computation.
iii. Field data composed of recorded drawdown-time data which was collected from non-pumping monitoring well are plotted on logarithmic paper of the same scale as the type curve.
iv. The field data and type curve plots are superimposed. Keeping the axes of the two curves parallel to each other, the field data curve is moved until it coincides with the type curve.
v. If a better matching is satisfied, the coordinates of matching point are noted. This point has four coordinates points which are $W(u)$ and $1/u$ coming from the type curve and $s(t)$ and t coming from the field data curve.
vi. Aquifer parameters namely transmissivity and storativity are then computed the following equations, respectively

$$T=\frac{Q}{4\pi s}W(u) \tag{4.8}$$

$$S=\frac{4Ttu}{r^2} \tag{4.9}$$

4.2. Flow in a Leaky Confined Aquifer

In the previous section, fundamentals of a completely confined aquifer were elaborated. However, in certain geological conditions, the vertical recharge of confining layer can not be omitted into the confined aquifer from adjacent units. Since aquitards, either above or below the aquifer, can leak water to aquifer, the net effect of this leakage causes to a reduction of the drawdown in the confined aquifer from that which is expected from completely confined aquifer response.

The vertical recharge rate due to aquitard leakage can be written by applying Darcy's law as

$$q = K' \frac{h_0 - h}{b'} \tag{4.10}$$

where K' is the hydraulic conductivity of confining bed $\left[L/T\right]$, b' is the thickness of the leaky layer $\left[L\right]$, and $h_0 - h$ is the drawdown $\left[L\right]$.

The two dimensional groundwater flow equation in polar coordinates as given in Equation (4.1) can be rewritten by considering the effect of leakage as follow

$$\frac{\partial^2 h}{\partial r^2} + \frac{1}{r}\frac{\partial h}{\partial r} - \frac{q}{T} = \frac{S}{T}\frac{\partial h}{\partial t} \Rightarrow \frac{\partial^2 h}{\partial r^2} + \frac{1}{r}\frac{\partial h}{\partial r} - K' \frac{h_0 - h}{Tb'} = \frac{S}{T}\frac{\partial h}{\partial t} \tag{4.11}$$

Substituting the drawdown s, Equation (4.11) yield as,

$$\frac{\partial^2 s}{\partial r^2} + \frac{1}{r}\frac{\partial s}{\partial r} - \frac{K's}{Tb'} = \frac{S}{T}\frac{\partial s}{\partial t} \tag{4.12}$$

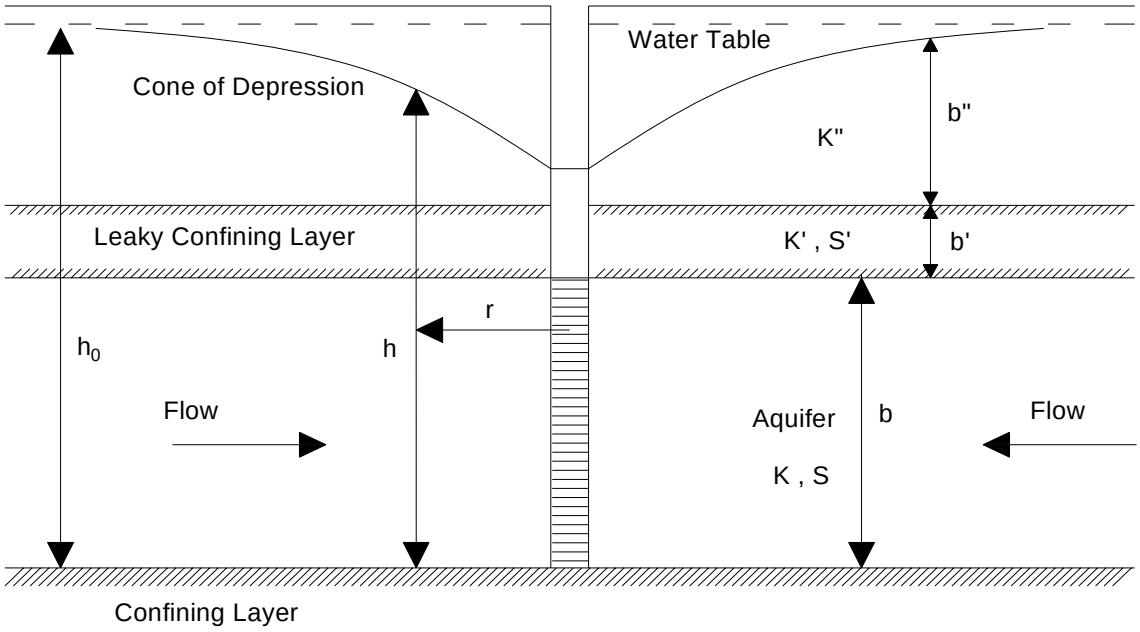

Figure 4.3. Schematic Illustration of a Leaky Confined Aquifer

The solution to Equation (4.11) was first derived by Hantush and Jacob (1955), and Hantush (1956). It is important to note the following assumptions that enable to derive analytical solution (Fetter 2001, Schwartz and Zhang, 2003).

- The pumping well fully penetrates the aquifer and has a constant pumping rate.
- Groundwater flow in the aquitard (leaky confining layer) is vertical and that layer has a uniform hydraulic conductivity (K') and thickness (b').
- Storage in the aquitard is negligible.
- Leakage across the aquitard comes from an aquifer whose head is assumed to be not affected by pumping.

Based on these assumptions, the solution of Equation (4.11) can be given as

$$h_0 - h = s = \frac{Q}{4\pi T} W\left(u, r/B\right) \tag{4.13}$$

where $B = \sqrt{\frac{Tb'}{K'}}$ and $W(u, r/B)$ is the leaky artesian well function.

The Hantush-Jacob well function $W(u, r/B)$ is expressed as

$$W(u, r/B) = \int_u^{\infty} \frac{1}{x} \exp\left(-x - \frac{r^2}{4B^2 x}\right) dx \tag{4.14}$$

Equation (4.14) can be approximated the following formula as (Prodanoff *et al*., 2006)

$$\begin{aligned} W(u, r/B) \cong\ & 2K_0(r/B) - I_0(r/B)\left[-E_i\left(\frac{r^2}{4B^2 u}\right)\right] \\ & + \left[\exp\left(-\frac{r^2}{4B^2 u}\right)\right] \times \left\{\gamma + \ln u + \left[-E_i(-u)\right] - u + u\left[\frac{I_0(r/B) - 1}{r^2/4B^2}\right]\right. \\ & \left. - u^2 \sum_{n=1}^{\infty} \sum_{m=1}^{n} \frac{(-1)^{n+m}(n-m+1)!}{(n+2)!^2} \left(\frac{r^2}{4B^2}\right)^m u^{n-m}\right\} \end{aligned} \tag{4.15}$$

where K_0 is the modified zero order Bessel function of second kind, I_0 is the modified zero order Bessel function of first kind, γ is the constant value equal to 0.577216, $-E_i()$ term is the exponential integral defined as

$$W(u) = E_1(u) = -E_i(-u) = \int_u^{\infty} \frac{1}{x} \exp(-x)\, dx, \quad u > 0 \tag{4.16}$$

Equation (4.15) indicates the some special cases depend on the well function variables u and r/B values. If the r/B value approaches to zero, the Hantush-Jacob solution reduces to the Theis solution since the aquitard becomes impermeable as $K' = 0$ (Freeze and Cherry, 1979). If the steady-state condition is satisfied, well function then depends on the function of r/B value. When u values approach infinity, the Hantush-Jacob well function vanishes. (Prodanoff *et al*., 2006) The following equation summarizes these conditions as

$$\begin{aligned} W(u,0) &= W(u) \\ W(0,r/B) &= 2K_0(r/B) \\ W(\infty,r/B) &= 0 \end{aligned} \tag{4.17}$$

Walton (1962) devised the graphical curve matching method for leaky confined aquifer. In a similar manner to the Theis curve matching procedure, the field time drawdown curve and theoretical family of type curves are plotted and superimposed. The following steps are applied with the modifications to leaky aquifer as follows

i. The field data curve should be matched to one of the type curves for r/B, if it is not, an imaginary type curve is drawn by interpolating between two r/B curves to satisfy matching.
ii. Coordinates of the matching point and r/B value are recorded, aquifer parameters then are computed as

$$T = \frac{Q}{4\pi s} W(u, r/B) \tag{4.18}$$

$$S = \frac{4uTt}{r^2} \tag{4.19}$$

$$K' = \frac{Tb'(r/B)^2}{r^2} \tag{4.20}$$

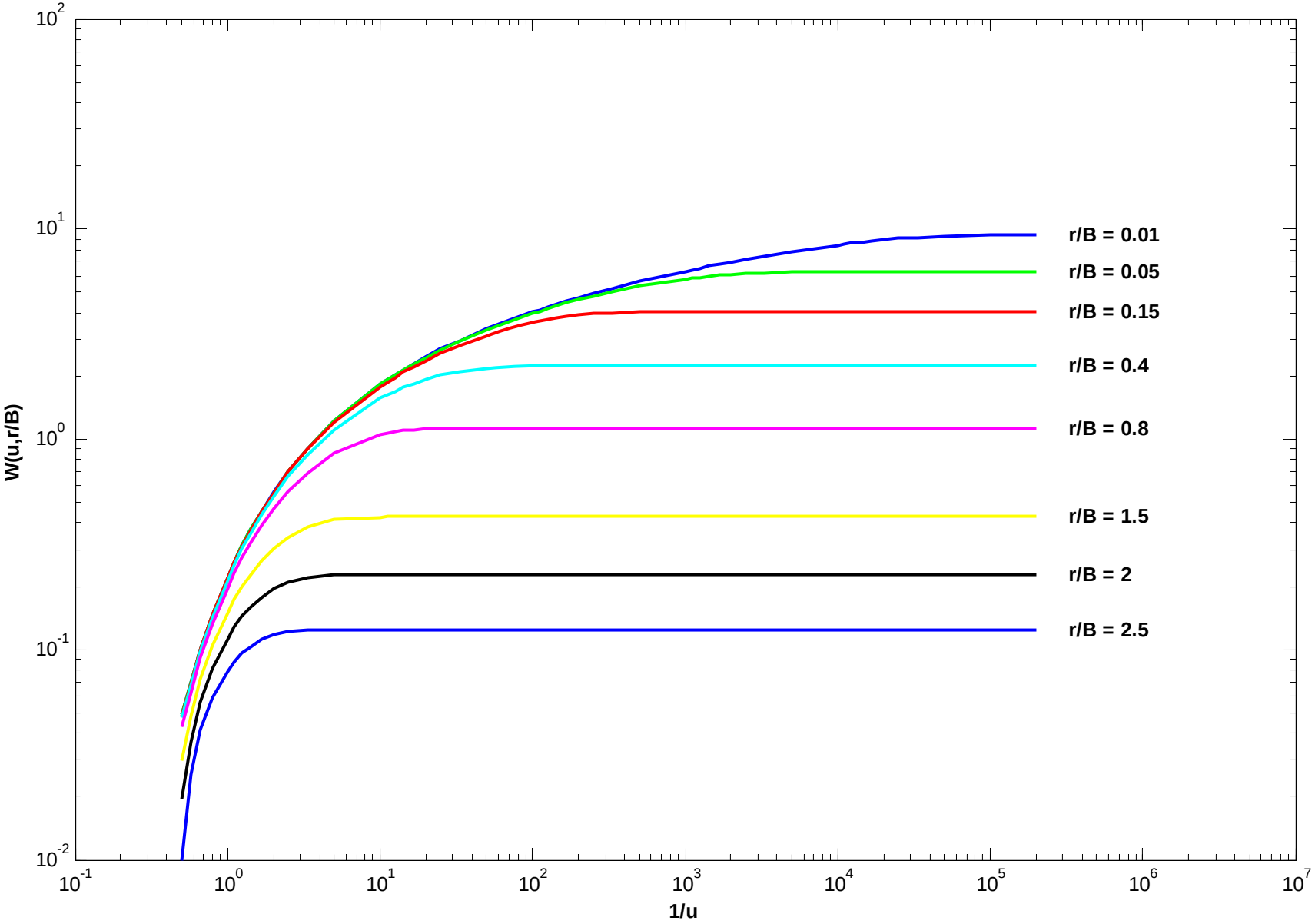

Figure 4.4. Non-equilibrium Leaky Aquifer Type Curves

Table 4.2. Values of W(u, r/B) (Hantush, 1956)

r/B u	0.1	0.015	0.03	0.05	0.075	0.1	0.15	0.2	0.3	0.4
0.000001	9.4425	8.6319	7.2471	6.2285	5.4228	4.8541	4.0601	3.5054	2.7449	2.2291
0.000005	9.4413	8.6319	7.2471	6.2285	5.4228	4.8541	4.0601	3.5054	2.7449	2.2291
0.00001	9.4176	8.6313	7.2471	6.2285	5.4228	4.8541	4.0601	3.5054	2.7449	2.2291
0.00005	8.8827	8.4533	7.245	6.2285	5.4228	4.8541	4.0601	3.5054	2.7449	2.2291
0.0001	8.3983	8.1414	7.2123	6.2282	5.4228	4.8541	4.0601	3.5054	2.7449	2.2291
0.0005	6.975	6.9152	6.6219	6.0821	5.4062	4.853	4.0601	3.5054	2.7449	2.2291
0.001	6.3069	6.2765	6.1202	5.7965	5.3078	4.8292	4.0595	3.5054	2.7449	2.2291
0.005	4.7212	4.7152	4.6829	4.6084	4.4713	4.296	3.8821	3.4567	2.7429	2.229
0.01	4.0356	4.0326	4.0167	3.9795	3.9091	3.815	3.5725	3.2875	2.7104	2.2253
0.05	2.4675	2.467	2.4642	2.4576	2.4448	2.4271	2.3776	2.311	2.1371	1.9283
0.1	1.8227	1.8225	1.8213	1.8184	1.8128	1.805	1.7829	1.7527	1.6704	1.5644
0.5	0.5598	0.5597	0.5596	0.5594	0.5589	0.5581	0.5561	0.5533	0.5453	0.5343
1	0.2194	0.2194	0.2193	0.2193	0.2192	0.219	0.2185	0.2179	0.216	0.2134
5	0.0011	0.0011	0.0011	0.0011	0.0011	0.0011	0.0011	0.0011	0.0011	0.0011

r/B u	0.5	0.6	0.7	0.8	0.9	1.0	1.5	2	2.5
0.000001	1.8488	1.555	1.321	1.1307	0.9735	0.842	0.4276	0.2278	0.1247
0.000005	1.8488	1.555	1.321	1.1307	0.9735	0.842	0.4276	0.2278	0.1247
0.00001	1.8488	1.555	1.321	1.1307	0.9735	0.842	0.4276	0.2278	0.1247
0.00005	1.8488	1.555	1.321	1.1307	0.9735	0.842	0.4276	0.2278	0.1247
0.0001	1.8488	1.555	1.321	1.1307	0.9735	0.842	0.4276	0.2278	0.1247
0.0005	1.8488	1.555	1.321	1.1307	0.9735	0.842	0.4276	0.2278	0.1247
0.001	1.8488	1.555	1.321	1.1307	0.9735	0.842	0.4276	0.2278	0.1247
0.005	1.8488	1.555	1.321	1.1307	0.9735	0.842	0.4276	0.2278	0.1247
0.01	1.8486	1.555	1.321	1.1307	0.9735	0.842	0.4276	0.2278	0.1247
0.05	1.7075	1.4927	1.2955	1.121	0.9701	0.8409	0.4276	0.2278	0.1247
0.1	1.4422	1.3115	1.1791	1.0505	0.9297	0.819	0.4271	0.2278	0.1247
0.5	0.5206	0.5044	0.486	0.4657	0.4439	0.421	0.3006	0.1943	0.1173
1	0.2101	0.2062	0.2017	0.1965	0.1909	0.1849	0.1499	0.1127	0.0792
5	0.0011	0.0011	0.0011	0.0011	0.0011	0.0011	0.0011	0.0011	0.0011

5. ARTIFICIAL NEURAL NETWORK APPROACH

Although there are many ANN applications on hydrology in the literature, there have been limited studies on the determination of aquifer parameters in lieu of traditional curve matching procedure. In recent years, some ANN approaches for aquifer parameter studies have been suggested by Aziz and Wong (1992), Balkhair (2002), and Lin and Chen (2006). These researchers used feed-forward back propagation networks in their works. In the Balkhair study, ANN was trained according to pre-defined aquifer parameters namely transmissivity T and storativity S ranges and subsequently the trained ANN was simulated by site data to obtain that parameters as network outputs. On the other hand, it was observed that there are some difficulties in selecting an appropriate trained range since there is no prior information of the aquifer parameters available. Lin and Chen (2006) have suggested a new ANN approach which combined analytical formulas derived by Theis (1935) and ANN as a function estimator of the well function to determine aquifer parameters. In their research, ANN produced a matching point like in the conventional curve matching produces rather than obtaining the parameters as ANN outputs, and that matching point is substituted in the analytical formula. This method seems to be a better approach because it avoids the need of an appropriate trained range.

The methodology used in this research to determine confined aquifer parameters is mostly based on the existing ANN approach which has been developed by Lin and Chen (2006). A new methodology combining Neural Network Approach and classical iteration techniques is proposed to determine Leaky Confined Aquifer parameters.

5.1. ANN Approach for Confined Aquifer Parameters

There is an implicit relationship among the time – drawdown curve and theoretical type curves. In the traditional approach, the type curve graphical method provides a matching point u_m to enable calculating the aquifer parameters. The ANN work was conducted using three layered ANN in order to detect the aquifer parameters from a set of

N observed time-drawdown data, this ANN was designed to produce a matching point u_m by processing relative drawdown ratios with respect to first recorded drawdown data in the input layer. This matching point u_m was then substituted in the analytical solution presented by Theis (1935).

Training data were generated assuming that time-drawdown curve and the type curve are identical to achieve this goal. First, $W(u)$ versus $1/u$ and time-drawdown curves were drawn in the logarithmic scale. The first time and corresponding drawdown data were taken as a matching point in the time-drawdown curve. To find the corresponding $1/u_m$ values in the theoretical Theis curve, the horizontal and vertical distances between the matching point and rest of observed data were calculated as shown in Figure 5.1.

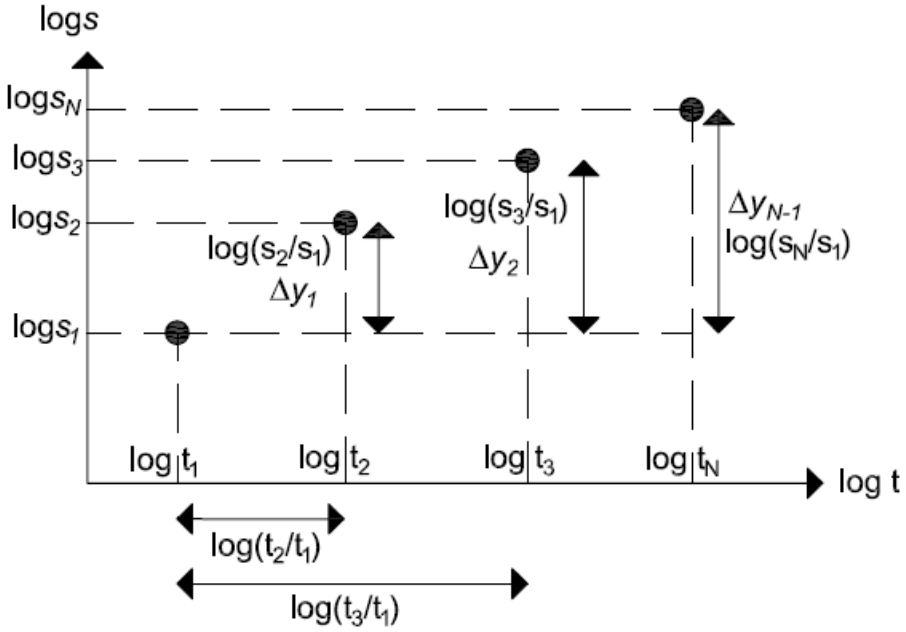

Figure 5.1. Time-Drawdown Data Graph in Logarithmic Scale

According to above figure, vertical distances can be written as following;

$$\Delta y_i = \log\left(s_{i+1}\right) - \log\left(s_1\right) = \log\left(\frac{s_{i+1}}{s_1}\right),\ i = 1, 2, ..., N-1 \tag{5.1}$$

Whereas the horizontal distances are given below formula;

$$\Delta x_i = \log\left(t_{i+1}\right) - \log\left(t_1\right) = \log\left(\frac{t_{i+1}}{t_1}\right),\ i = 1, 2, ..., N-1 \tag{5.2}$$

The projections of these points on $W(u)$ vs. $1/u$ graph shown in Figure 5.2 were calculated considering that time t_1 is assigned as a matching point that is value of $1/u_m$ away from the origin. Therefore horizontal distances of the remaining data on $W(u)$ vs. $1/u$ graph are computed as;

$$\Delta X_i = \log\left(\frac{1}{u_m}\right) + \Delta x_i = \log\left(\frac{1}{u_m}\right) + \log\left(\frac{t_{i+1}}{t_1}\right) = \log\left(\frac{1}{u_m} \times \frac{t_{i+1}}{t_1}\right),\ i = 1, 2, ..., N-1 \tag{5.3}$$

If the $1/u$ variables are equivalent to above formula, the corresponding $W(u)$ values are given as;

$$W(u_i) = W\left(u_m \times \frac{t_1}{t_{i+1}}\right),\ i = 1, 2, ..., N-1 \tag{5.4}$$

Thus, the vertical distances on $W(u)$ vs. $1/u$ graph will equal to following formula;

$$\Delta Y_i = \log\left[\frac{W\left(u_m \times t_1 / t_{i+1}\right)}{W\left(u_m\right)}\right],\ i = 1, 2, ..., N-1 \tag{5.5}$$

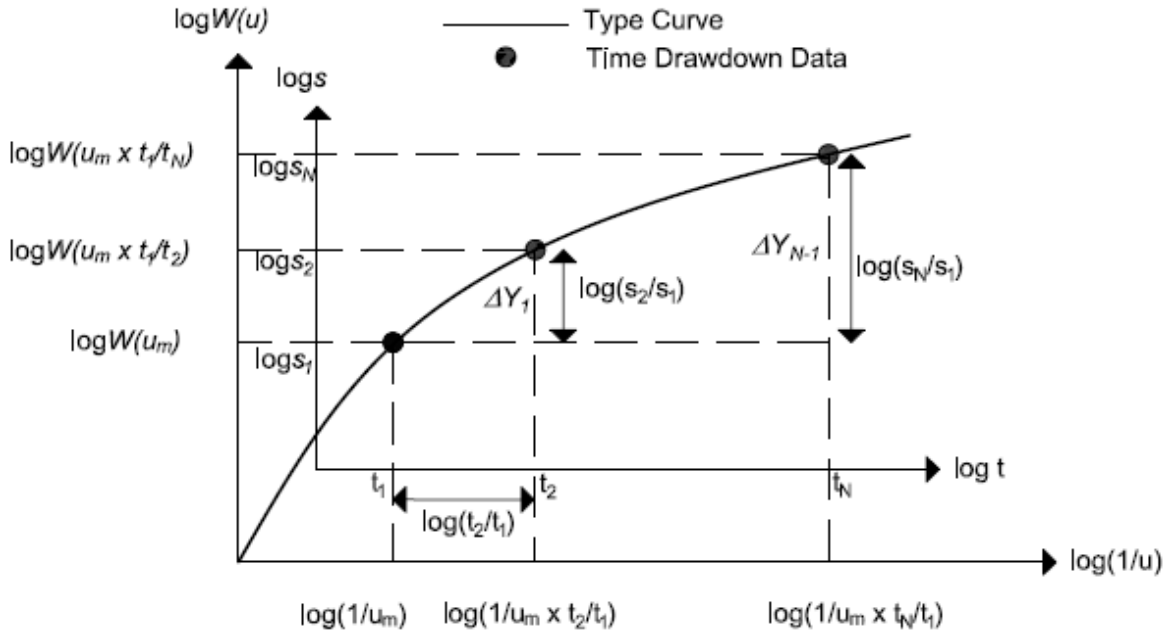

Figure 5.2. Preparation of Training Inputs

As stated earlier, $W(u)$ vs. $1/u$ graph and time-drawdown curves are identical. For that reason, the training data was generated using by Equation (5.5), while the Equation (5.4) was used to obtain matching point u_m in the simulation of ANN. Figure 5.3 and 5.4 represent the training and simulation data of ANN.

Once the ANN is trained according to a chosen error criteria, it is capable of producing an output when an input vector is introduced based on time-drawdown data (Lin and Chen, 2006). The aquifer parameters are then calculated from following formulas;

$$y_{output} = \log\left(\frac{1}{u_m}\right) \Rightarrow u_m = 10^{-y_{output}} \tag{5.6}$$

$$W_m = W\left(u_m\right) = W\left(\frac{1}{10^{y_{output}}}\right) \tag{5.7}$$

$$s_1 = s_m \tag{5.8}$$

$$t_1 = t_m \tag{5.9}$$

Finally, combining equations (5.6), (5.7), (5.8) and (5.9), transmissivity T and storativity S for a confined aquifer are determined by

$$T = \frac{Q}{4\pi s_m} W_m(u) \tag{5.10}$$

and

$$S = \frac{4Tu_m t_m}{r^2} \tag{5.11}$$

It is also possible to compute the estimated drawdowns using the estimated transmissivity and storativity values which were produced by ANN. The estimated drawdowns can be computed following equations;

$$u_i = \frac{r^2 \hat{S}}{4\hat{T}t_i} \tag{5.12}$$

$$\hat{s}_i = \frac{Q}{4\pi\hat{T}} W(u_i) \tag{5.13}$$

where $\hat{T}$ and $\hat{S}$ denote the estimated transmissivity and storativity, respectively, t_i is time for the i^{th} component of recorded time data, u_i is the corresponding well function value for the i^{th} component of recorded time data and W is the Theis well function.

The accuracy of ANN drawdown estimations with respect to field data was checked by utilizing the root mean squared error (RMSE) and relative root mean squared error (RRMSE) which are presented as;

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(\hat{s}_i - s_i\right)^2} \tag{5.14}$$

$$RRMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(\frac{\hat{s}_i - s_i}{s_i}\right)^2} \tag{5.15}$$

where N is the number of recorded site data, $\hat{s}_i$ and s_i are the estimated and the observed drawdown data, respectively, for the i^{th} component of a comparison set.

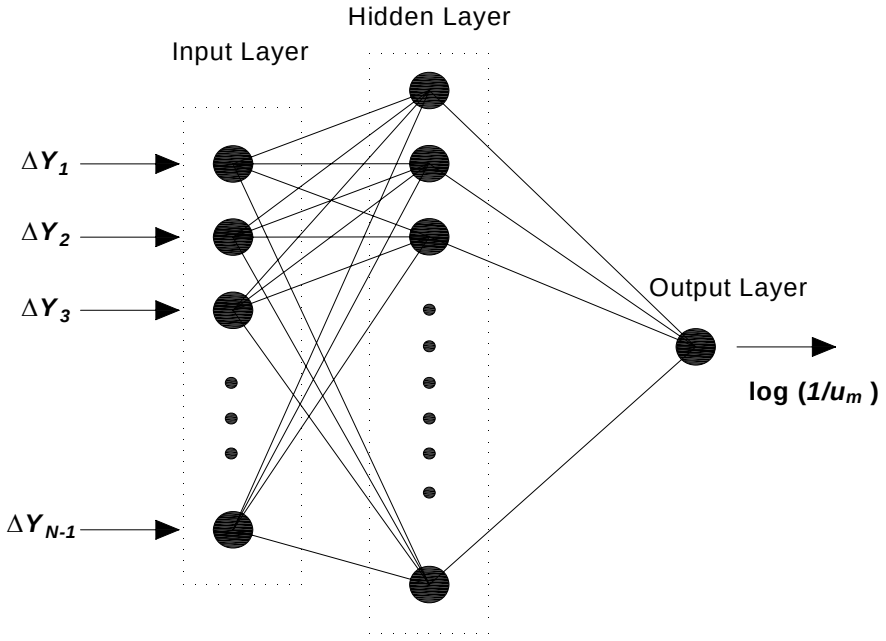

Figure 5.3. Schematic View of ANN with Training Inputs

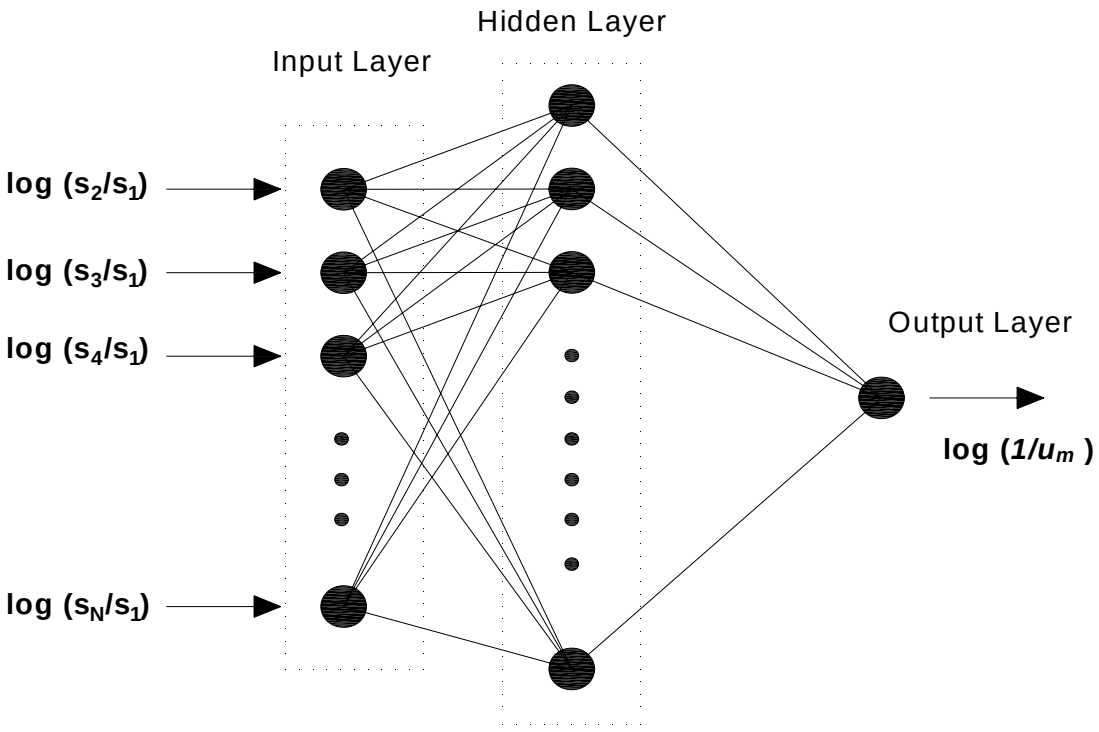

Figure 5.4. Schematic View of ANN for Simulation

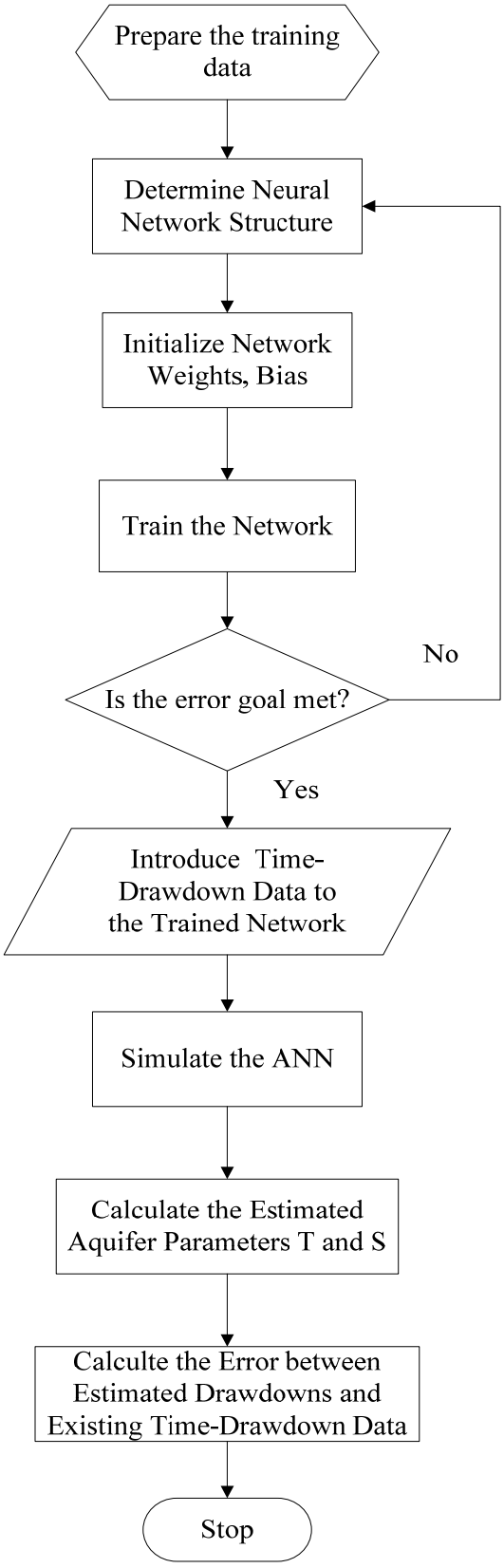

Figure 5.5. Flow Chart of the ANN Model for Confined Aquifer

5.2. ANN Approach for Leaky Confined Aquifer Parameters

When the leaky confined aquifer parameters are to be estimated, a new technique which is a hybrid method that enables ANN to make the iterations in order to detect r/B is proposed in this research. According to the literature, leaky aquifer parameters are computed by superimposing the time-drawdown data and a family of type curves created by Hantush (1956) well function. In the proposed algorithm, the iteration of ANN starts from an initial guess of r/B. For this procedure, the initial iteration value is selected from a given set of r/B vectors instead of producing a matching point u_m and r/B as output vector of the network. ANN computes a matching point u_m and corresponding drawdown data using the procedures which have been explained in the previous section. RRMSE between estimated and field drawdown data is calculated for that r/B value. RRMSE is compared by a chosen tolerance value. If the computed error is not in acceptable limits, this produce will repeated until the convergence is satisfied.

The training data were generated in a similar manner to Equation (5.5), namely, by contrast Theis well function was replaced by Hantush well function in here. Training data can be written as follow;

$$\Delta Y_i = \log\left[\frac{W\left(u_m \times t_1 / t_{i+1}, r/B\right)}{W\left(u_m, r/B\right)}\right],\ i = 1, 2, ..., N-1 \tag{5.16}$$

Equation (5.6), (5.8), and (5.9) are still valid to determine aquifer parameters. Equation (5.7) was modified for leaky confined aquifer well function given as;

$$W_m = W\left(u_m, \left(r/B\right)_m\right) = W\left(\frac{1}{10^{y_{output}}}, \left(r/B\right)_m\right) \tag{5.17}$$

The leaky confined aquifer parameters are then calculated from following formulas;

$$T = \frac{Q}{4\pi s_m} W\left(u_m, (r/B)_m\right) \tag{5.18}$$

$$S = \frac{4Tu_m t_m}{r^2} \tag{5.19}$$

$$K' = \frac{Tb'\left(r/B\right)_m^2}{r^2} \tag{5.20}$$

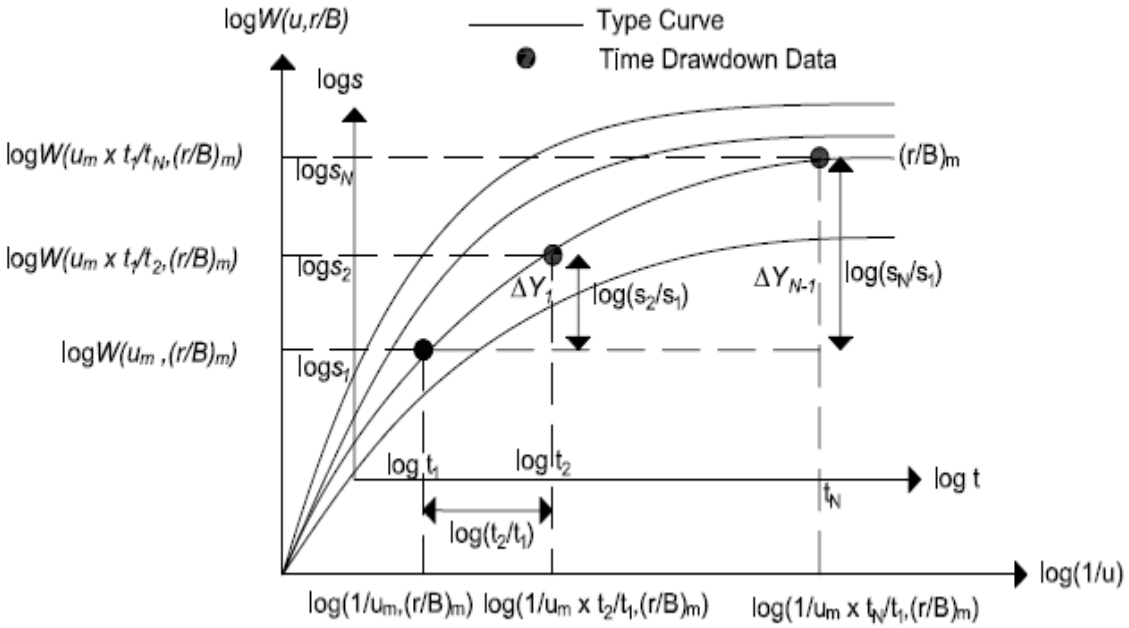

Figure 5.6. Preparation of Training Data for Leaky Aquifer

RMSE and RRMSE were computed using by Equation (5.14) and (5.15), respectively. Figure 5.7 represents flowchart of the proposed algorithm.

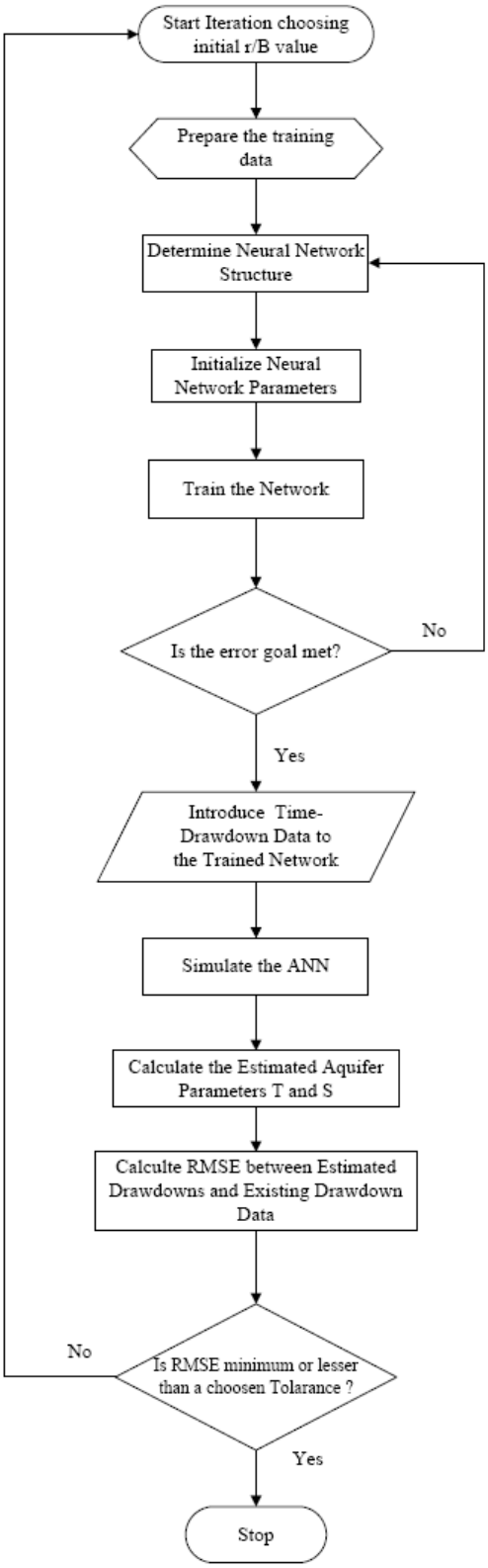

Figure 5.7. Flowchart of the Proposed ANN Approach

6. ANN PUMP TEST ANALYIS

The developed methodologies were used to determine the aquifer parameters for confined aquifer and leaky- confined aquifer with data obtained from the literature. The results were compared to the conventional curve matching methods.

6.1. Aquifer Parameters Determination for Completely Confined Aquifer by ANN

The data for determining the confined aquifer parameters were taken from Schwartz and Zhang (2003). The problem is defined as a well that fully penetrates a confined aquifer discharging at a uniform pumping rate of $500\ m^3/day$. The time-drawdown data was collected from an observation well located 300 m away the pumping well. Table 6.1 shows the time-drawdown data of the observation well.

Table 6.1. Time-Drawdown Data for Confined Aquifer

Time (min)	Drawdown (m)	Time (min)	Drawdown (m)
1	0.03	35.62	1.79
1.27	0.05	45.2	1.97
1.61	0.09	57.36	2.15
2.04	0.15	72.79	2.33
2.59	0.22	92.37	2.52
3.29	0.31	117.21	2.7
4.18	0.41	148.74	2.89
5.3	0.53	188.74	3.07
6.72	0.66	239.5	3.26
8.53	0.8	303.92	3.45
10.83	0.95	385.66	3.64
13.74	1.11	489.39	3.83
17.43	1.27	621.02	4.02
22.12	1.44	788.05	4.21
28.07	1.61	1000	4.39

6.1.1. Training Data Preparation

Based on the type curves presented in Figure 4.4, $\log(1/u)$ is always greater than -0.5 and when $\log(1/u)$ is greater than 4.0, corresponding logarithmic well function $\log W(u)$ values approach to constant value (Walton, 1962). For that reason, the training range of ANN outputs was selected from this $\log(1/u)$ range as -0.5 to 4.0. The input vector components of the training phase were then generated according to available field time- drawdown data using the Equation (5.5) as

$$\Delta Y_i = \log\left[\frac{W\left(u_m \times t_1 / t_{i+1}\right)}{W\left(u_m\right)}\right],\ i = 1, 2, ..., N-1 \tag{6.1}$$

where N denotes the number of time-drawdown data. A total of 10205 training patterns were generated using $\log(1/u)$ values from -0.5 to 4.0 with a step size of 4.41×10^{-4}. Based on the available time-drawdown data, Equation (6.1) implies that each row of training input data matrix having the size of 29×10205 was comprised from time components of the field data. This relatively huge matrix was reduced to size of 1×10205 by applying the Principal Component Analysis (PCA). The PCA is a powerful method that reduces the dimensionality of large data set. The input vectors were normalized to new vectors which have zero mean and unit variance. Also the minimum fraction variance as 0.02 was utilized for eliminating the principal components that contribute less than 2 % to the total variation in the data set (Samani *et al*., 2007).

6.1.2. Determining ANN Configuration

As explained in Section 2.4., determining ANN configuration is generally a time-consuming procedure that depends on trial-error. Neural network parameters such as learning rate η and the momentum constant μ_c were assigned as 0.5 and 0.6, respectively, based on the Lin and Chen study (2006). In order to determine the optimum number of neurons in the hidden layer, ANN was started to train for 3 neurons in the hidden layer for

300 epochs. In a growing manner, the number of neurons in the hidden layer was gradually increased and the training mean square errors (MSEs) were compared. When the MSE was plotted over epoch numbers shown in Figure 6.1, the number of neurons in the hidden layer was determined as 8 neurons which indicated the optimal solution with respect to alternative neuron numbers.

Table 6.2. Training Errors for Different Number of Neurons in the Hidden Layer

Epoch No	**n = 3**	**n = 5**	**n = 8**	**n = 10**
0	0.469538	2.54084	0.904255	1.90958
50	1.05E-04	2.78E-05	3.72E-07	1.30E-06
100	7.43E-05	1.29E-05	2.31E-07	3.98E-07
150	2.55E-05	8.86E-06	1.81E-07	3.34E-07
200	1.34E-05	4.71E-06	1.52E-07	2.81E-07
250	1.13E-05	9.87E-07	1.32E-07	2.40E-07
300	1.08E-05	5.25E-07	1.17E-07	2.09E-07
Required Training Time (sec)	19.27	25.23	39.80	49.22

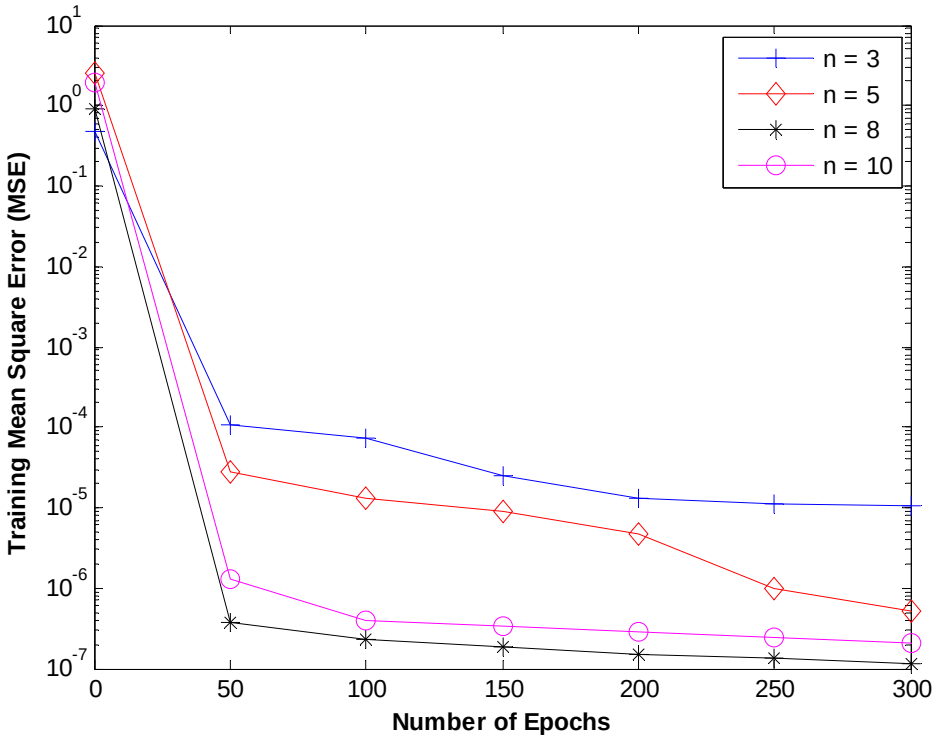

Figure 6.1. Training Errors for the Alternative Neuron Numbers

6.1.3. Training and Testing of ANN

After constructing the appropriate ANN structure, ANN was ready to train with the input training data that was prepared using by Equation (6.1). The Table 6.3 summarizes the neural network configuration and parameters.

Table 6.3. ANN Configuration for Confined Aquifer Parameters Estimation

Parameter	**Value**
Learning Rate	0.5
Momentum Constant	0.6
Number of Neuron in the hidden layer	8
Network Configuration	1-8-1
Error Criteria (MSE)	1.00E-08
Maximum Number of Epoch	10000
Number of Training Patterns	10205

A total of 100 test patterns that were not used during the training procedure were employed to compute the generalization performance of the existing ANN. The test patterns were randomly generated from combinations of idealized T and S values ranging from 10^2 to 10^5 m^2/day and 10^{-2} to 10^{-4}, respectively. (Lin and Chen, 2006) Two different error criteria namely RRMSE and coefficient of determination were utilized to evaluate the ANN performance between the target values and estimated values.

$$RRMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(\frac{y_i - \hat{y}_i}{y_i}\right)^2} \tag{6.2}$$

$$R^2 = 1 - \frac{\sum_{i=1}^{N}\left(y_i - \hat{y}_i\right)^2}{\sum_{i=1}^{N}\left(y_i - \overline{y}\right)^2} \tag{6.3}$$

where y_i is target value, $\hat{y}_i$ is the simulated value computed by neural network, $\overline{y}$ is the mean of the target values, and N is the number of test patterns. Best linear fit between observed and calculated values would have RRMSE = 0 and $R^2 = 1$. The following figures present the testing performance of trained ANN for transmissivity and storativiy estimations, respectively.

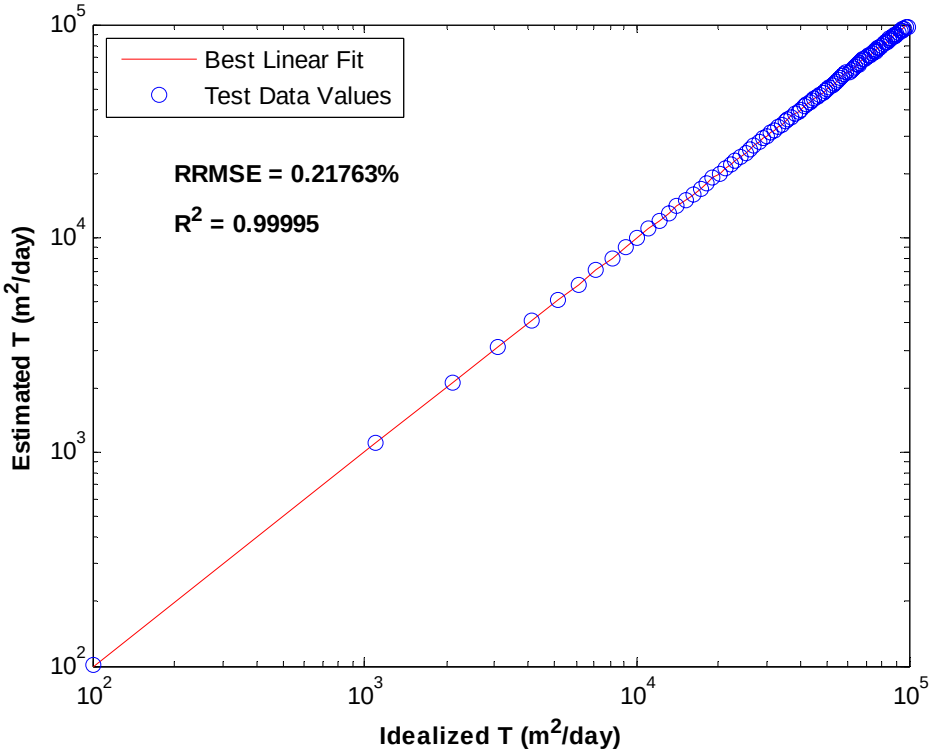

Figure 6.2. Training Range of Estimated Transmissivity Values

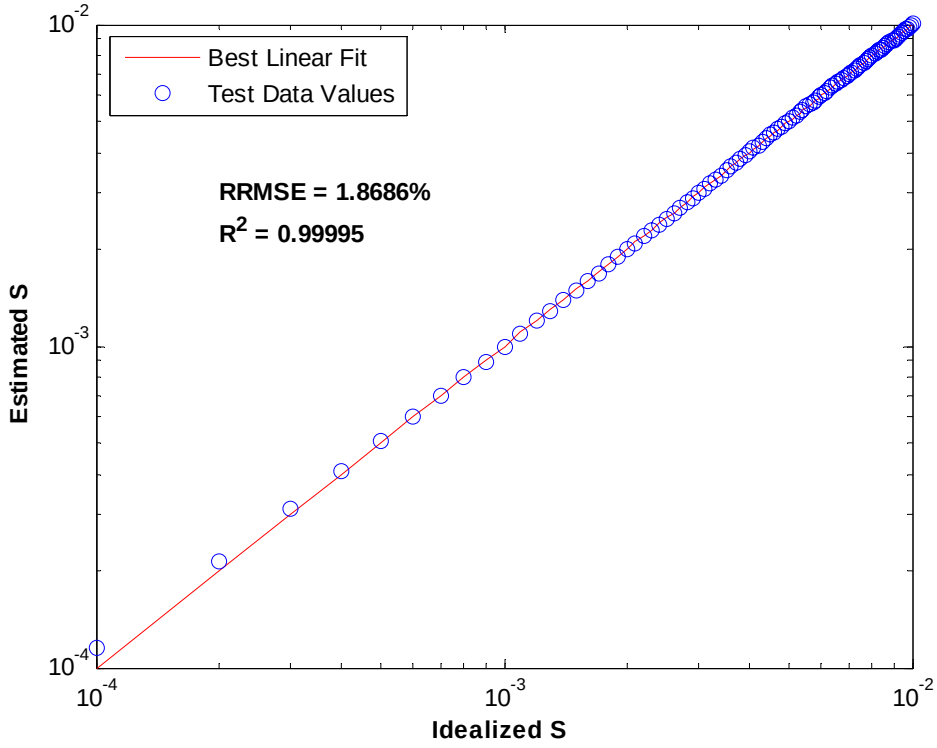

Figure 6.3. Training Range of Estimated Storativity Values

6.1.4. Comparison of ANN Solution and Conventional Methods

The field time-drawdown data were introduced to well-trained and tested ANN to compute the confined aquifer parameters. The aquifer parameters were also determined by Theis type-curve graphical method and Cooper-Jacob straight line method. Table 6.5 shows the results of the three methods.

Table 6.4. Comparison of Estimated Aquifer Parameters

	ANN Output	**Theis Solution**	**Cooper – Jacob Solution**
Estimated T (m^2/day)	51.522	52.354	51.125
Estimated S	3.45E-06	3.39E-06	3.02E-06
RMSE	0.0483	0.0673	0.0621
R-RMSE (%)	3.73	5.51	17.30
R^2	0.9988	0.9977	0.9980

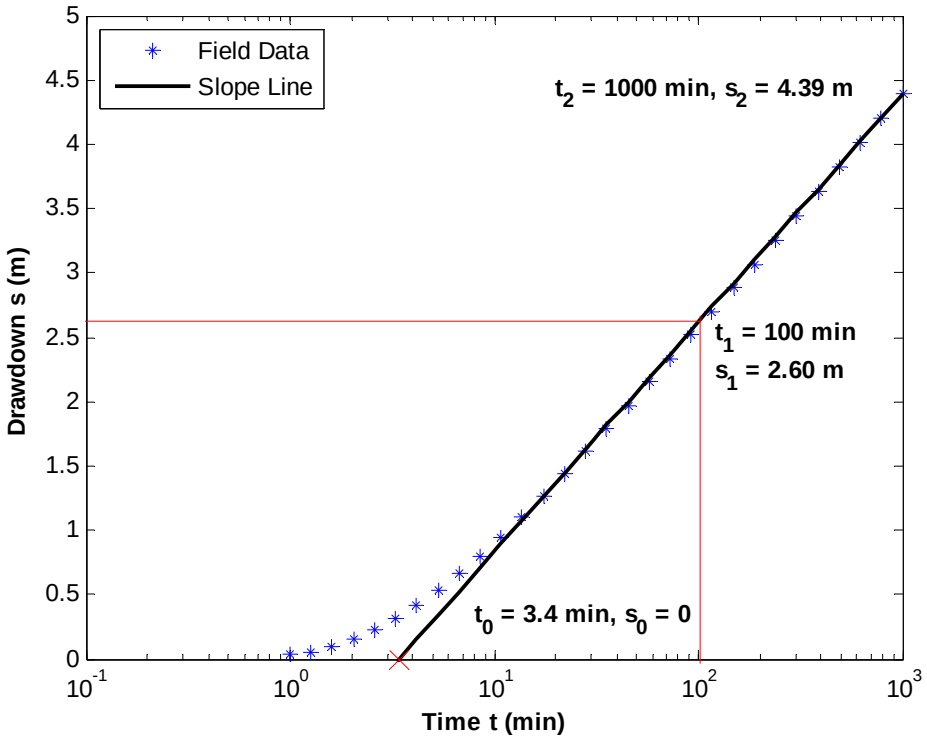

Figure 6.4. Cooper-Jacob Solution

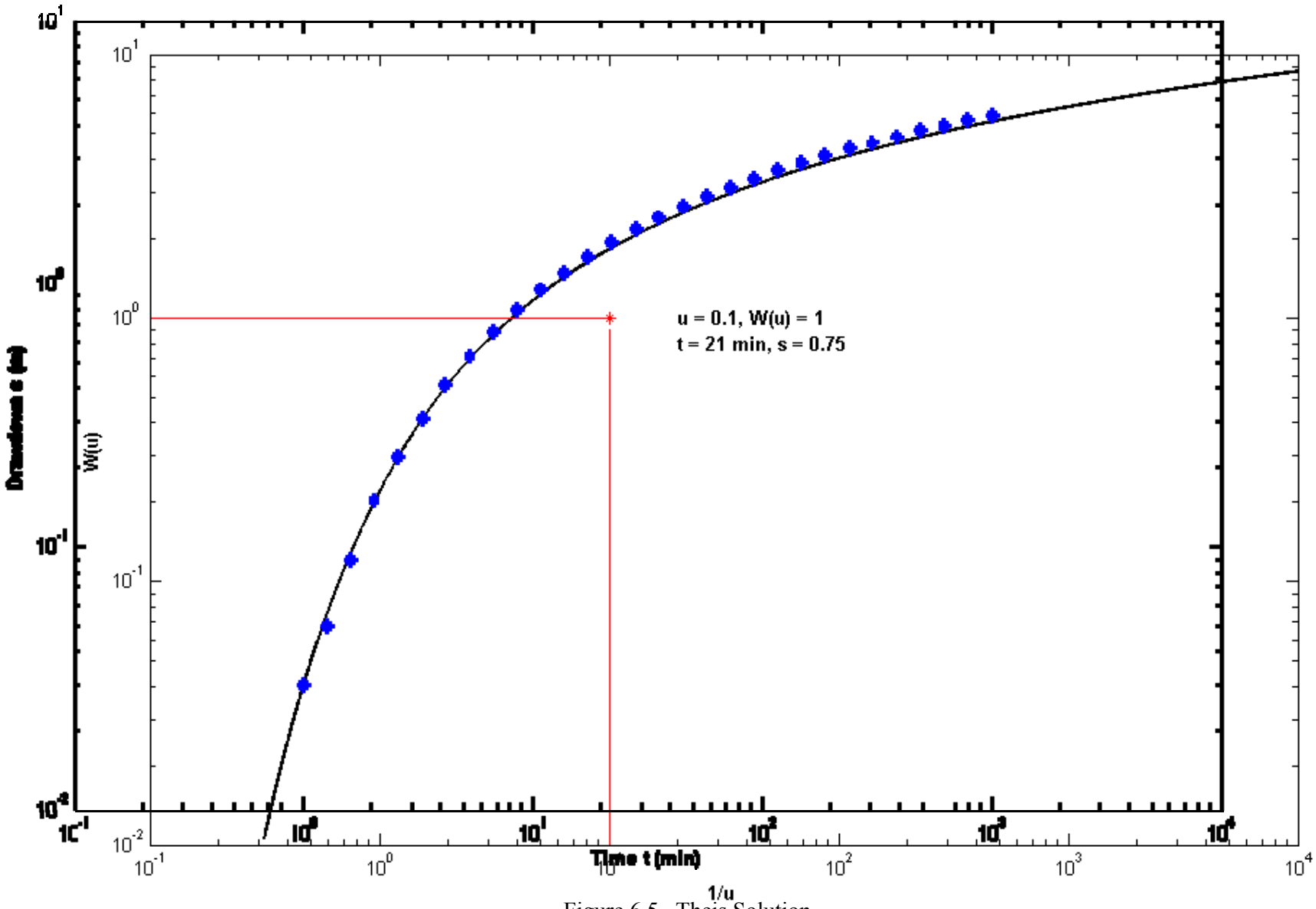

Figure 6.5. Theis Solution

6.2. Aquifer Parameters Determination for Leaky Confined Aquifer by ANN

The data for determining leaky aquifer parameters were taken from Schwartz and Zhang (2003). In this example, a test of a leaky, confined aquifer was conducted at a pumping rate of $10872\ m^3/day$. The time-drawdown data were collected from an observation well that was located from 154 m away the pumping well. The thickness of semi-confining layer was assumed to be 20 m. Table 6.6 shows the time-drawdown data of the observation well.

Table 6.5. Time-Drawdown Data for Leaky Confined Aquifer

Time (min)	Drawdown (m)	Time (min)	Drawdown (m)
0.1	0.27	11.72	2.54
0.14	0.39	16.1	2.59
0.19	0.53	22.12	2.63
0.26	0.68	30.39	2.64
0.36	0.85	41.75	2.65
0.49	1.03	57.36	2.65
0.67	1.21	78.8	2.65
0.92	1.39	108.26	2.65
1.27	1.58	148.74	2.65
1.74	1.76	204.34	2.65
2.4	1.93	280.72	2.65
3.29	2.09	385.66	2.65
4.52	2.23	529.83	2.65
6.21	2.36	727.9	2.65
8.53	2.46	1000	2.65

6.2.1. Training Data Preparation and Determination of NN Configuration

The Hantush-Jacob well function depends on two variables, namely dimensionless u and r/B ratio as elaborated in the Chapter 4. According to the proposed algorithm, training data were prepared considering $\log(1/u)$ and r/B ranges. As described in Section 6.1.1, $\log(1/u)$ is always greater than -0.5 and when $\log(1/u)$ is greater than

4.0, when the corresponding logarithmic Hantush-Jacob well function $\log W(u, r/B)$ values approach constant value (Walton, 1962). A total of 100 training patterns were generated using $\log(1/u)$ values from -0.5 to 4.0 with a step size of 4.5×10^{-2} for each r/B value. The r/B values were selected as shown in Table 6.6.

Table 6.6. Simulated r/B values

r/B	0.01	0.015	0.03	0.05	0.075	0.1	0.15	0.2	0.3	0.4
	0.5	0.6	0.7	0.8	0.9	1	1.5	2	2.5	

For one r/B value, the 29×100 training matrix was generated. PCA was then applied to reduce the matrix dimension to 1×100. ANN was trained with 19 training matrices that were generated by utilizing 19 different r/B values.

The same neural network configuration in Section 6.1.2 was used to determine leaky aquifer parameters since there is no difference among the neural network topologies. The only difference between determination of confined aquifer and leaky confined aquifer parameters is that different well functions are assigned to generate training data. However, ANN started to estimate aquifer parameters for each r/B value and computed the RMSEs between the field and estimated drawdowns in the proposed algorithm. When the minimum RMSE was obtained, ANN stopped the simulation.

6.2.2. Training and Testing of ANN

A total of 100 test patterns that were not used during the training procedure were employed to determine estimation ranges of transmissivity T and storativity S values. Assuming $r/B = 0.01$, the test patterns were randomly generated from combinations of idealized T and S values ranging from 10^2 to 10^5 m^2/day and 10^{-2} to 10^{-4}, respectively. Figure 6.6 and 6.7 characterize the testing performance of the implemented ANN.

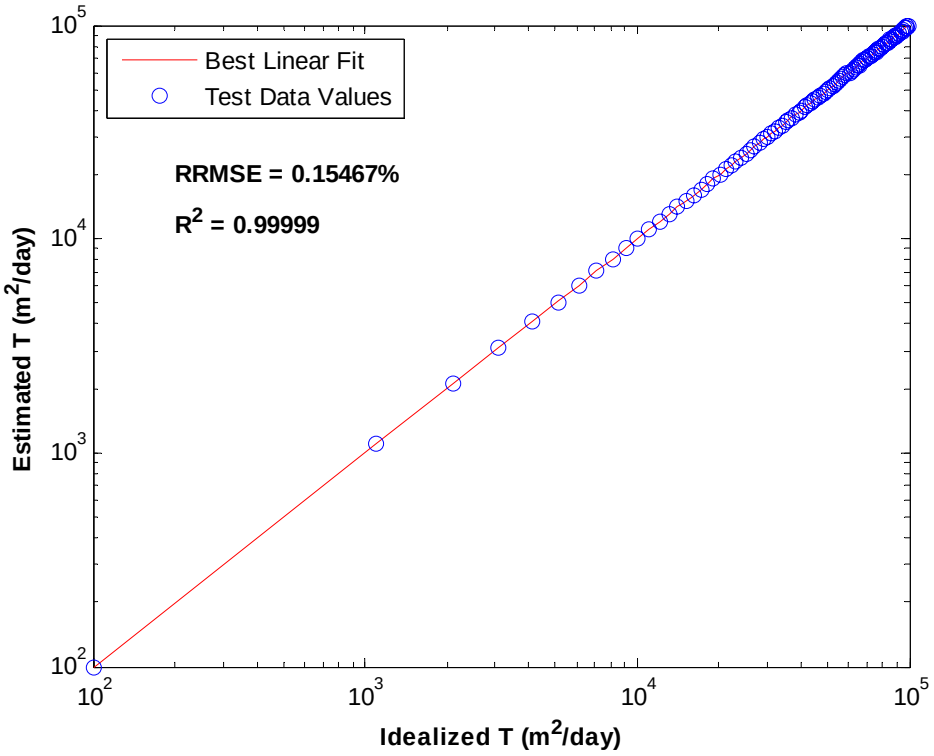

Figure 6.6. Training Range of Estimated Transmissivity for r/B = 0.01

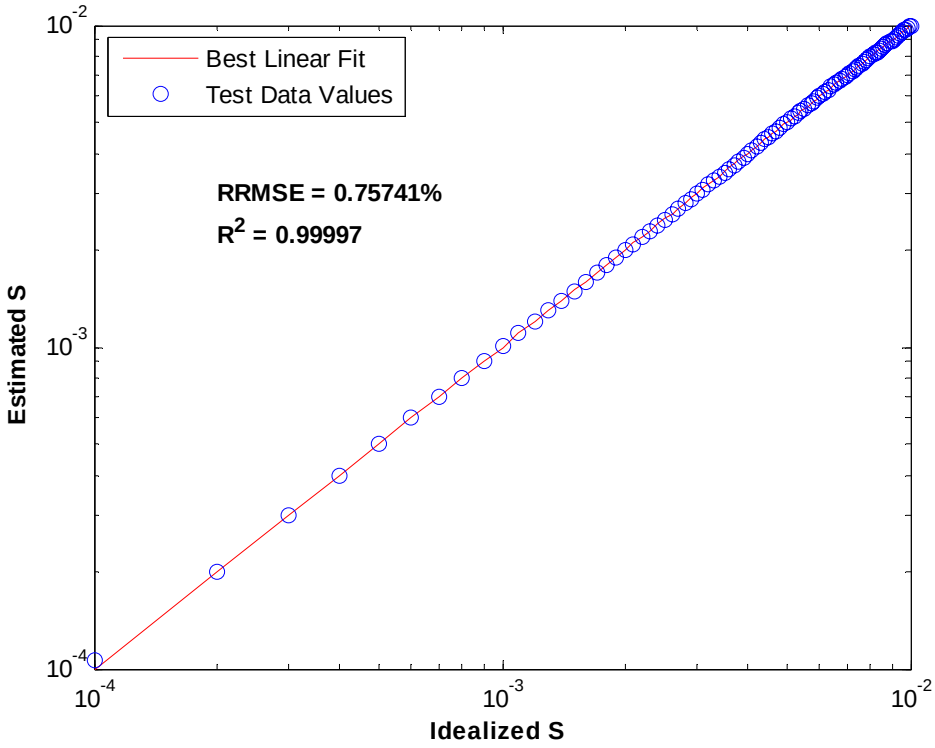

Figure 6.7. Training Range of Estimated Storativity for r/B = 0.01

6.2.3. Comparison of ANN Solution and Conventional Methods

After the ANN was trained, the field time-drawdown data were introduced for simulation. Aquifer parameters were also determined by the conventional type curve graphical method. ANN solution was illustrated in Figure 6.8. According to Figure, minimum RMSE value was obtained when ANN was simulated with the 7^{th} component of the r/B set.

Table 6.7. Comparison of Estimated Aquifer Parameters for Leaky Confined Aquifer

	ANN Output	**Walton Type Curve Graphical Method**
r/B	0.15	0.15
Estimated T (m^2/day)	1319.70	1373.30
Estimated S	9.998E-06	9.97E-06
Estimated K (m/day)	0.0250	0.0261
RMSE	0.0483	0.0824
R-RMSE (%)	0.44	3.60
R^2	0.9994	0.9895

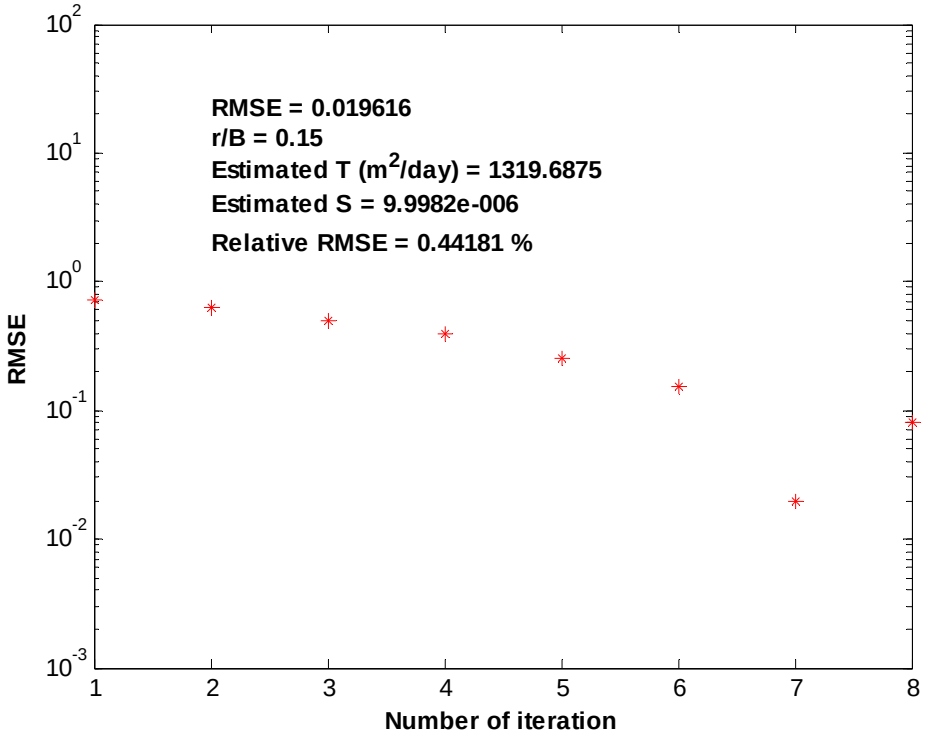

Figure 6.8. RMSE of the Proposed ANN

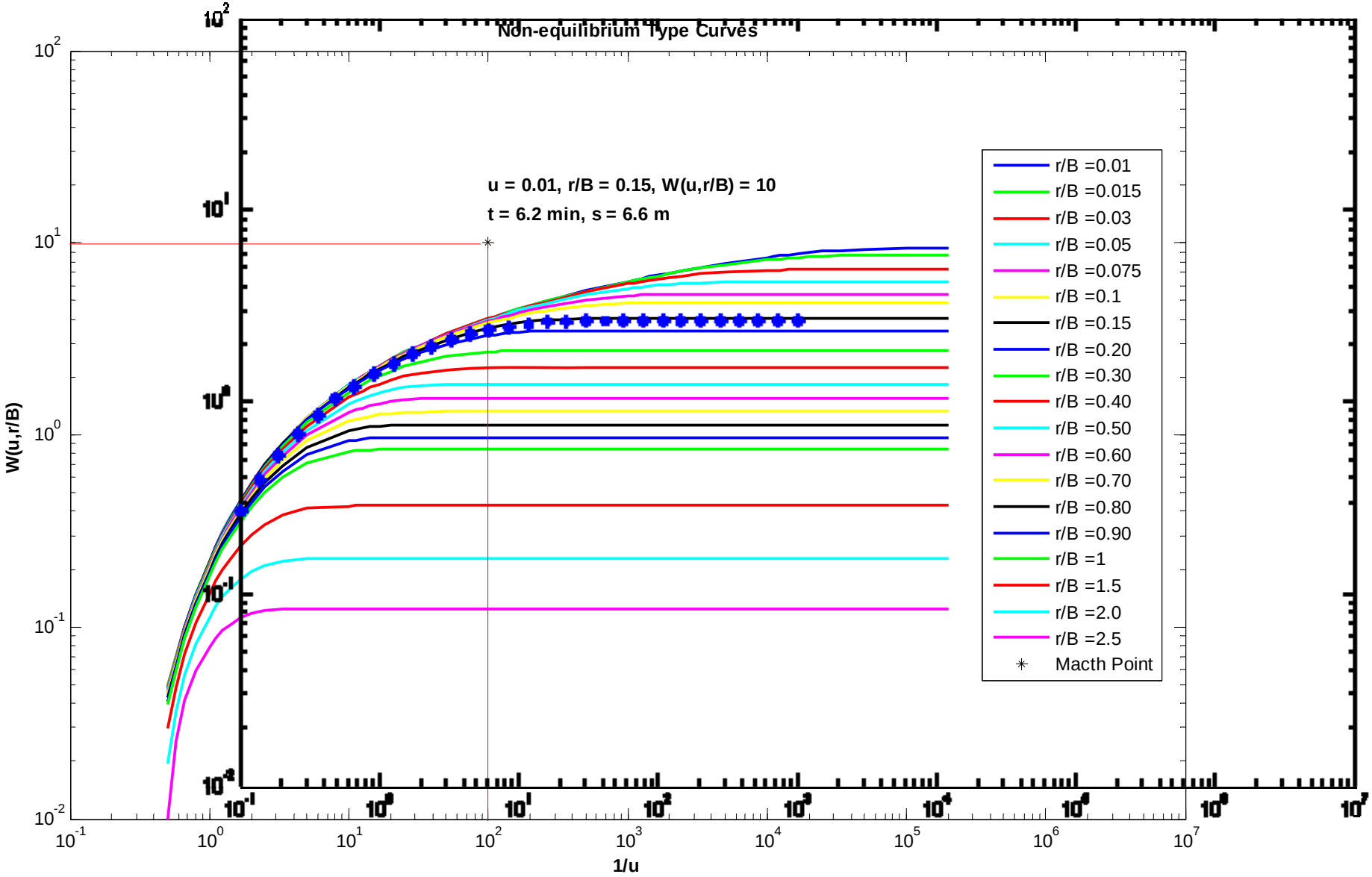

Figure 6.9. Walton Type Curve Matching Solution

6.3. Assessment

In these examples, ANN was employed to approximate the well functions of confined and leaky aquifer, respectively. The results indicate that the ANN approaches perform more accurate and reliable for the both situations. One of the advantages of using the ANN model is that the errors from superimposition of the curves disappear, and the accuracy of estimation increases. Furthermore, the ANN model can be easily adapted to any time drawdown data within the investigated cases. When a new time-drawdown data is introduced to ANN models, regenerating the training input data is enough to estimate the aquifer parameters. Table 6.8 summarizes the computation efforts of the ANN models in this study. The computation efforts were obtained by Intel Pentium 4 CPU 3.20 GHz, 512 MB Ram PC.

Table 6.8. Computation Effort of the Models

	Training Data Generation		**Training**	**Simulation**
Model	**Matrix size**	**Elapsed Time (sec)**	**Elapsed Time (sec)**	**Elapsed Time (sec)**
Confined Aquifer Parameters Estimation	29 x 10205	95.3	5.97	0.086
Leaky Aquifer Parameters Estimation	19 x 29 x 100	210.9	36.2	0.145

7. ANN AQUIFER TRANSMISSIVITY ESTIMATIONS

In this chapter, a numerical experiment was conducted to show the ability of artificial neural network on the estimations of transmissivity value. For this purpose, a hypothetical confined aquifer was generated using by Turning Band Algorithm (TBA). TBA produces a non-conditional simulation of a multi-normal random function with a given covariance structure (Mantoglou and Wilson, 1981). The synthetic transmissivity values were ranging from 6.77 m^2/day to 115.46 m^2/day with the mean value of 44.21 m^2/day in $150\,m \times 150\,m$ rectangular domain size. A uniform grid size of 10 m was utilized for the generation of the synthetic transmissivity values. The transmissivity map of the hypothetical aquifer is represented in Figure 7.1 and 7.2.

In this section, 25 sampling points were randomly selected from the entire domain to develop the transmissivity values of the aquifer by utilizing ANN, Radial Basis Function Collocation Method (RBFCM) and Ordinary Kriging (OK) Method. The number of sampling points was deliberately limited in order to review the mapping efficiency of each method. Consequently, a pumping test was conducted by simulating the transmissivity values of each case in MODFLOW software. The four different monitoring wells (MWs) were used to observe the drawdown response of the aquifer. The estimation performance of ANN, RBFCM and OK method was compared based on RMSE each well.

The univariate statistical analysis for the synthetic transmissivity values of the entire aquifer was summarized in Table 7.1. The preceding figure also exhibits the histograms of generated and log transformed transmissivity values, respectively.

Table 7.1. Statistical Analysis for Hypothetical Transmissivity Values

Statistical Criteria	**Transmissivity Values**	**Log_{10} Transmissivity Values**
Number of Data	225	225
Min	6.77	0.8306
Max	115.46	2.0624
Mean	44.2129	1.5183
Lower quantile (25%)	15.72	1.1965
Median	40.29	1.6052
Upper quantile (75%)	71.475	1.8542
Variance	868.6463	0.1303
Standard Deviation	29.4728	0.3609
Coefficient of Variance	0.6666	0.2377

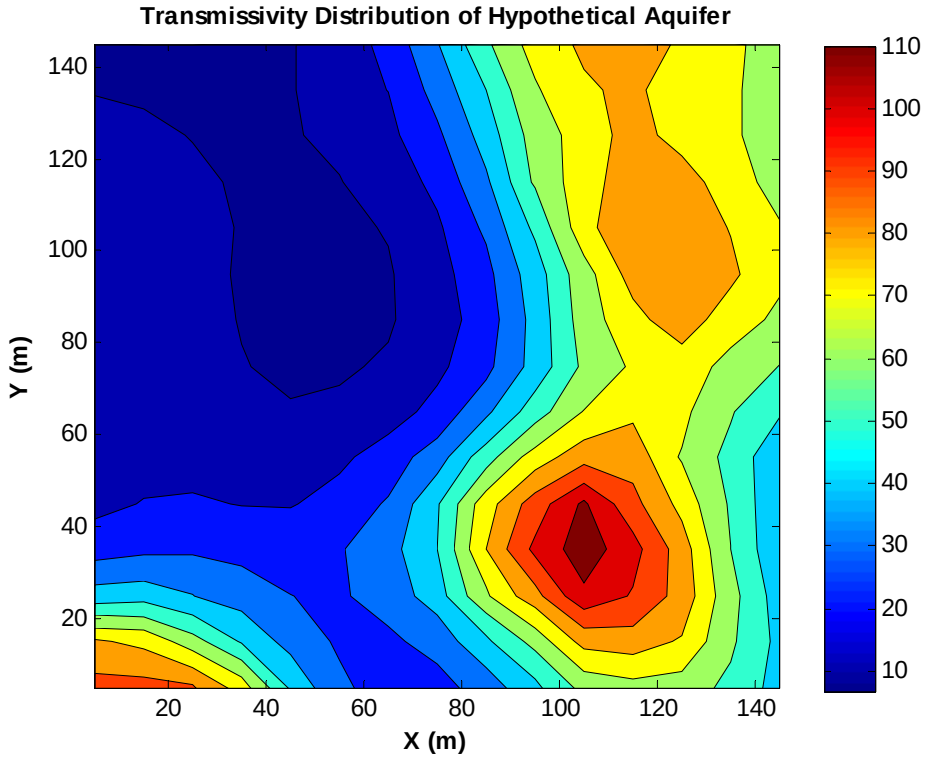

Figure 7.1. Contour Plot of the Hypothetical Aquifer Transmissivity

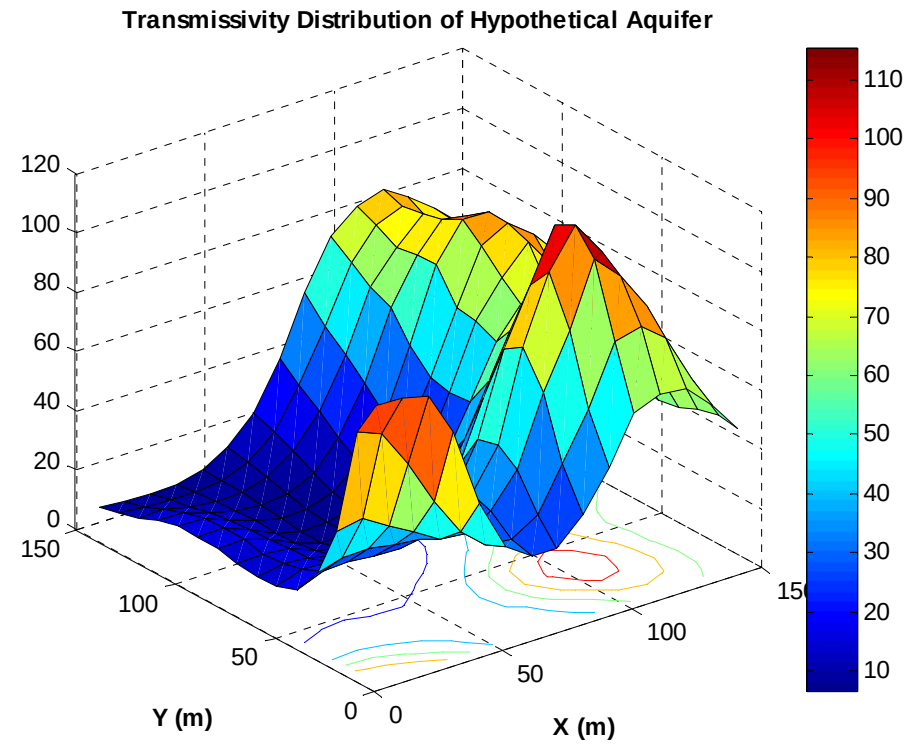

Figure 7.2. 3-D View of the Hypothetical Aquifer Transmissivity

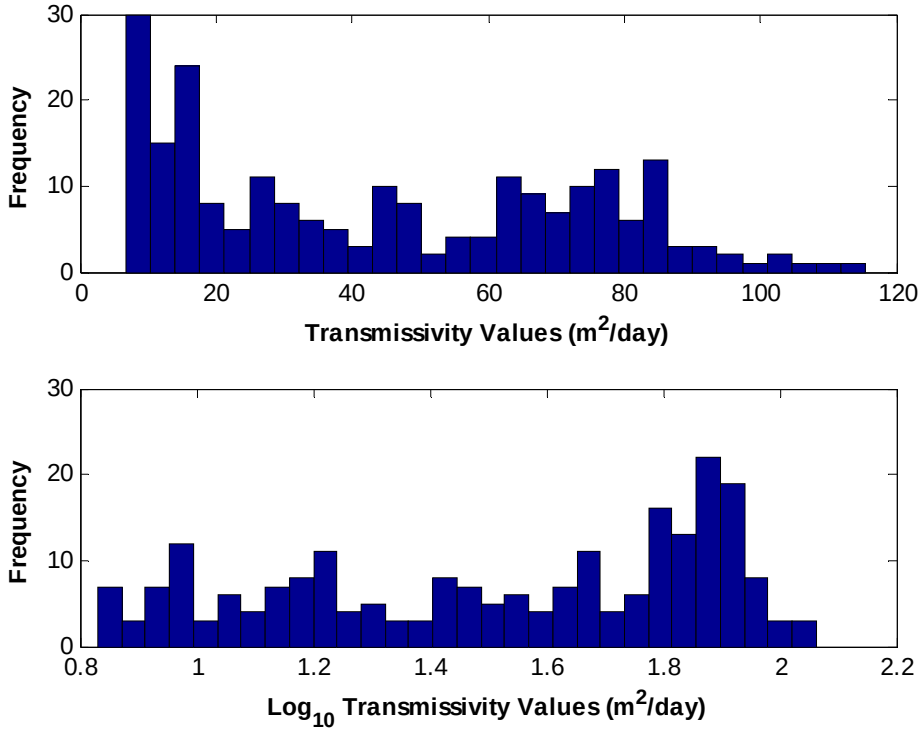

Figure 7.3. Histograms of Hypothetical Transmissivity Values

7.1. Radial Basis Function Approach

The Radial Basis Functions (RBFs) were first used in multivariate data and function interpolation (Frankie, 1982). Kansa (1990 a., b.) used them for scattered data approximation and for the solution of the partial differential equations (PDEs). Radial Basis function Collocation Method (RBFCM), also known as the Kansa method, is one of the solution technique which is widely used in mesh-free methods. The fundamental logic behind the RBFCM is the construction of unknown function matrix based on the available data for the problem. The function to be approximated can be written as

$$u_i = f_{ij}(r)\alpha_j \quad i, j = 1, 2, ..., N \tag{7.1}$$

where $f_{ij}(r)$ is radial basis function which evaluates radial distance from i^{th} collation point to j^{th} collation point, α_j is coefficient of the j^{th} collation point, N denotes the number of nodes or centers of the available data. The radial distance r among the node is presented by

$$r_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2 + (z_i - z_j)^2} \qquad i, j = 1, 2, ..., N \tag{7.2}$$

The spatial derivative of u can be derived from Equation (7.1) as

$$\frac{\partial^n u_i}{\partial \xi^n} = \frac{\partial^n f_{ij}}{\partial \xi^n}\alpha_j \tag{7.3}$$

where ξ denotes the coordinate directions. Applying chain rule, Equation (7.3) yields

$$\frac{\partial f_{ij}}{\partial \xi} = \frac{\partial f_{ij}}{\partial r}\frac{\partial r}{\partial \xi} \Rightarrow \frac{\partial f_{ij}}{\partial \xi} = \frac{\partial f_{ij}}{\partial r}\frac{(\xi_{i-}\xi_j)}{r} \tag{7.4}$$

There are many radial basis functions which are widely used in the literature. Table 7.2 depicts most commonly used RBFs and their properties.

Table 7.2. Some Common RBFs

RBF	**Formula**	**Comment**
Multiquadrics (MQ)	$\left(r^2+c^2\right)^{\beta/2}$	$\beta>0, \beta\in 2N+1, N$ is integer
Inverse Multiquadric	$\left(r^2+c^2\right)^{-\beta/2}$	$\beta>0, \beta\in 2N+1, N$ is integer
Splines	r^{β}	$\beta>0, \beta\in 2N+1, N$ is integer
Thin Plate Splines (TPS)	$r^{\beta}\ln r$	$\beta>0, \beta\in 2N+1, N$ is integer
Gaussian	$\exp\left(-c^2r^2\right)$	

Based on the Table 7.2, RBFs within the form of multiquadric and Gaussian contain c^2 term called as shape parameter. The value of shape parameter directly affects the accuracy of approximation. When the shape parameter $c^2 >> r_{\max}^2$, shape parameter dominates the f matrix described in Equation1, so that the approximated u function will have more or less the same values. On the other hand, when the shape parameter value approaches zero, the radial distances govern the f matrix; therefore the approximation function will lose its smoothness (Börekçi, 2005). Considering this dilemma, the selection of appropriate shape parameter value has great importance on the approximation accuracy. Although there are no mathematical explanations for the selection of c^2 term, the previous studies in the RBF literature have suggested the some heuristic method to define shape parameter. Hardy (1971) proposed the starting value for a trial and error procedure to find an optimal c^2 value as

$$c = 0.815 r_{av} \tag{7.5}$$

where r_{av} is the average distance between each node and its nearest neighbor. Hon *et al.*, (1999) reported that satisfactory results were obtained for c between $5r_{av}$ and $8r_{av}$. Wu and Hon (2003) suggested another shape parameter criterion that guarantee the invertible f matrix as

$$c = 4r_{\min} \tag{7.6}$$

where $r_{\min}$ is the minimum radial distance between the nodes. The commercial software SURFER utilize the shape parameter as

$$c^2 = \frac{l^2}{25 \times N} \tag{7.7}$$

where l is the length of diagonal of the data extent, N is the number of data points.

7.2. Kriging Method

Kriging method was initially introduced to provide estimates for unsampling values by Krige (1951). Kriging is often assumed as the best linear unbiased estimator (BLUE) (Lee, 2000). The reasons to named as the acronym BLUE is that Kriging is linear due to that estimated points are composed of the weighted linear combination of available data, Kriging is unbiased since it has a mean of residual or error, and additionally Kriging is the best due to it tries to minimize the variance of the errors (Isaaks and Sritastava, 1989). In reality, it is difficult to attain the goals of Kriging because there is no data which resemble the entire non-samples values. For that reason, variogram models which can model the entire domain character from the available sample points are assigned to attain these ambitious goals of Kriging.

A variogram is defined as the measure of spatial variability. In the geostatistic literature, there are many variogram types that represent experimental measures of spatial variability or continuity. In this example, semivariogram was used to determine the spatial variability of the domain at the hand. The semivariogram is half of the squared difference between the paired data values approximately separated by h distance as follows

$$\gamma(h) = \frac{1}{2N(h)} \sum_{(i,j)|h_{ij}=h} (x_i - y_i)^2 \tag{7.8}$$

where $\gamma(h)$ is estimated variogram at the separation distance h, $N(h)$ is the total number of pairs attributes that are separated by h, x_i is the value at the start or tail of pair i, y_i is the corresponding end or head value (Deutsch and Journel, 1998).

In Ordinary Kriging technique[2], the estimation point can be written in terms of the weighted linear combination of sample points as

$$\hat{V}(x_0) = \sum_{i=1}^{n} w_i V(x_i) \tag{7.9}$$

where $\hat{V}(x_0)$ is the estimated value, $V(x_i)$ is the sample value at node i, w_i is the weight of node i, and n is the total number of sampling points. The error or residual of the estimated point is the difference between the true and the corresponding estimate as follow

$$R(x_0) = \hat{V}(x_0) - V(x_0) \tag{7.10}$$

The variance of a weighted linear combination is given by

$$Var\left\{\sum_{i=1}^{n} w_i V_i\right\} = \sum_{i=1}^{n}\sum_{j=1}^{n} w_i w_j Cov(V_i V_j) \tag{7.11}$$

where Cov denotes the covariance function value between the nodes. Combining Equation (7.10) and Equation (7.11), the variance of the error can be expressed as

$$\begin{aligned} Var\{R(x_0)\} = Cov\{\hat{V}(x_0)\hat{V}(x_0)\} - 2Cov\{\hat{V}(x_0)V(x_0)\} \\ + Cov\{V(x_0)V(x_0)\} \end{aligned} \tag{7.12}$$

The first term of Equation (7.12) represents the variance of $\hat{V}(x_0)$, and in a similar fashion, the last term of Equation (7.13) implies that the variance of $V(x_0)$ which is assumed to be the same for all random variables as follows

[2] Notations and derivations were adapted from Isaaks and Sritastava, 1989

$$Cov\left\{\hat{V}\left(x_0\right)\hat{V}\left(x_0\right)\right\} = Var\left\{\sum_{i=1}^{n} w_i V_i\right\} = \sum_{i=1}^{n}\sum_{j=1}^{n} w_i w_j C_{ij} \tag{7.13}$$

$$Cov\left\{V\left(x_0\right)V\left(x_0\right)\right\} = \sigma^2 \tag{7.14}$$

The second term of Equation (7.12) can be expanded using the statistical expectation operator E as

$$\begin{aligned} 2Cov\left\{\hat{V}\left(x_0\right)V\left(x_0\right)\right\} &= 2Cov\left\{\left(\sum_{i=1}^{n} w_i V_i\right)V_0\right\} \\ &= 2E\left\{\sum_{i=1}^{n} w_i V_i \cdot V_0\right\} - 2E\left\{\sum_{i=1}^{n} w_i V_i\right\} \cdot E\left\{V_0\right\} \\ &= 2\sum_{i=1}^{n} w_i \cdot \left[E\left\{V_i, V_0\right\} - E\left\{V_i\right\} \cdot E\left\{V_0\right\}\right] \\ &= 2\sum_{i=1}^{n} w_i Cov\left\{V_i V_0\right\} \end{aligned} \tag{7.15}$$

Combining these three equations, the error variance is written as

$$\sigma_R^2 = \sigma^2 + \sum_{i=1}^{n}\sum_{j=1}^{n} w_i w_j C_{ij} - 2\sum_{i=1}^{n} w_i C_{i0} \tag{7.16}$$

Kriging goal is to minimize the error variance. This minimization problem can be solved by Lagrangian multiplier as

$$\begin{aligned} &Min\left(\sigma_R^2\right) \\ &Subject\ to \sum_{i=1}^{n} w_i = 1 \end{aligned} \tag{7.17}$$

The unbiasedness condition is rearranged as follow

$$\sum_{i=1}^{n} w_i = 1 \Rightarrow 2\left(\sum_{i=1}^{n} w_i - 1\right) = 0 \tag{7.18}$$

Adding this equation to Equation (7.16), objective function yield as

$$\sigma_R^2 = \sigma^2 + \sum_{i=1}^{n}\sum_{j=1}^{n} w_i w_j C_{ij} - 2\sum_{i=1}^{n} w_i C_{i0} + 2\mu\left(\sum_{i=1}^{n} w_i - 1\right) \tag{7.19}$$

where μ is the Lagrangian multiplier.

Applying Lagrangian multiplier rule to solve this minimization problem as

$$\begin{aligned} \frac{\partial\left(\sigma_R^2\right)}{\partial w_i} &= 2\sum_{j=1}^{n} w_j C_{ij} - 2C_{i0} + 2\mu = 0 \\ \frac{\partial\left(\sigma_R^2\right)}{\partial \mu} &= 2\sum_{i=1}^{n} w_i - 2 = 0 \end{aligned} \tag{7.20}$$

Rewriting Equation (7.20), Ordinary Kriging system can be expressed as

$$\begin{aligned} &\sum_{j=1}^{n} w_j C_{ij} + \mu = C_{i0} \qquad i = 1, 2, ..., n \\ &\sum_{i=1}^{n} w_i = 1 \end{aligned} \tag{7.21}$$

Equation (7.21) can also be written in matrix notation presented as follow

$$\begin{bmatrix} C_{11} & . & . & C_{1n} & 1 \\ . & . & . & . & 1 \\ . & . & . & . & 1 \\ C_{n1} & . & . & C_{nn} & 1 \\ 1 & 1 & 1 & 1 & 0 \end{bmatrix} \times \begin{bmatrix} w_1 \\ . \\ . \\ w_n \\ \mu \end{bmatrix} = \begin{bmatrix} C_{i0} \\ . \\ . \\ C_{n0} \\ 1 \end{bmatrix} \tag{7.22}$$

It is also possible to write the Ordinary Kriging system described in Equation (7.21) in terms of semivariogram model as preceding equation

$$\sum_{j=1}^{n} w_j \gamma_{ij} - \mu = \gamma_{i0} \qquad i = 1, 2, ..., n$$
$$\sum_{i=1}^{n} w_i = 1 \tag{7.23}$$

Equations (7.21) and (7.23) strongly imply that selection of appropriate variogram models directly affects the accuracy of the approximation like the selection of appropriate shape parameters in RBFCM.

7.3. Applications of the Models

The location map of sampling points was shown in below Figure 7.4

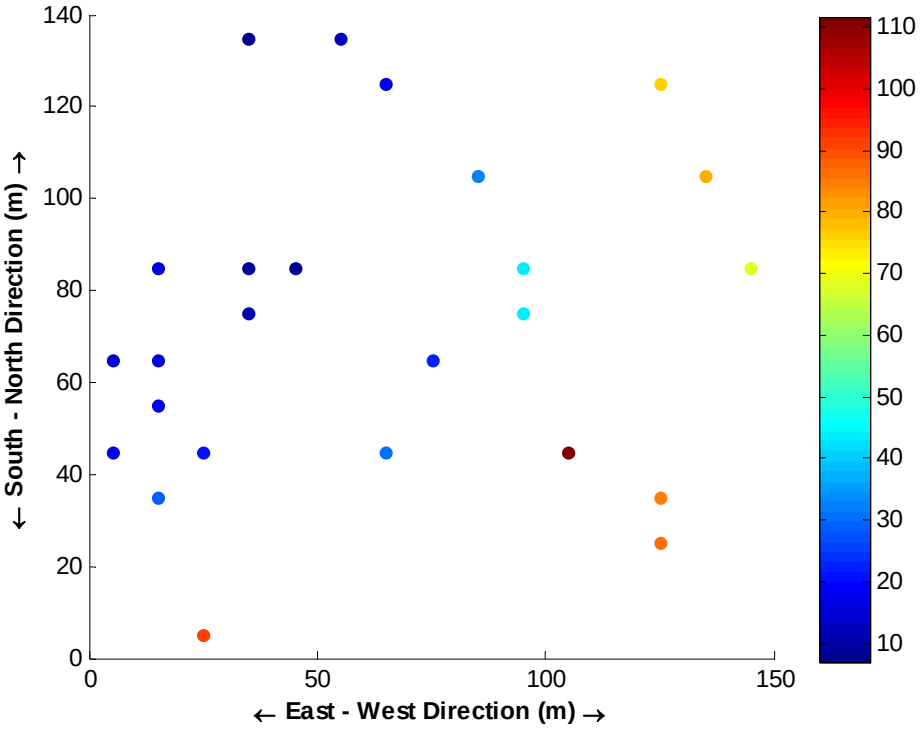

Figure 7.4. Location Map of Sampling Data

The statistical analysis of the sampling data was summarized in Table 7.3.

Table 7.3. Statistical Analysis of Sampling Data

Statistical Criteria	**Transmissivity Values**	**Log_{10} Transmissivity Values**
Number of Data	25	25
Min	6.99	0.8445
Max	111.72	2.0481
Mean	38.6548	1.441
Lower quantile (25%)	15.31	1.1844
Median	22.84	1.3587
Upper quantile (75%)	70.1825	1.8456
Standard Deviation	32.034	0.3677
Coefficient of Variance	0.8287	2.0481

To apply the RBFCM to the experiment, multi-quadric RBF was utilized to map the transmissivity distribution of the hypothetical aquifer. The shape parameter was chosen according to SURFER Software formula

$$c^2 = \frac{l^2}{25 \times N} \tag{7.24}$$

In Ordinary Kriging method, first semivariogram models that resembled the sample data character as the best as possible were investigated. For this purpose, linear, Gaussian and exponential models have been used to provide best fit with experimental semivariogram values.

Gaussian model and exponential model formulations are shown the following equations

$$\gamma(h) = \begin{cases} C_0 + C_1\left(1 - \exp\left(-\dfrac{h^2}{a^2}\right)\right), & h \le a \\ C_1, & h > a \end{cases} \tag{7.25}$$

$$\gamma(h)=\begin{cases} C_0+C_1\left(1-\exp\left(-\dfrac{h}{a}\right)\right), & h\le a \\ C_1, & h>a \end{cases} \tag{7.26}$$

where C_0 is the nugget effect, C_0+C_1 is the sill, a is the integral scale, and h is the distance. The semivariogram of the sample data is shown in Figure 7.5.

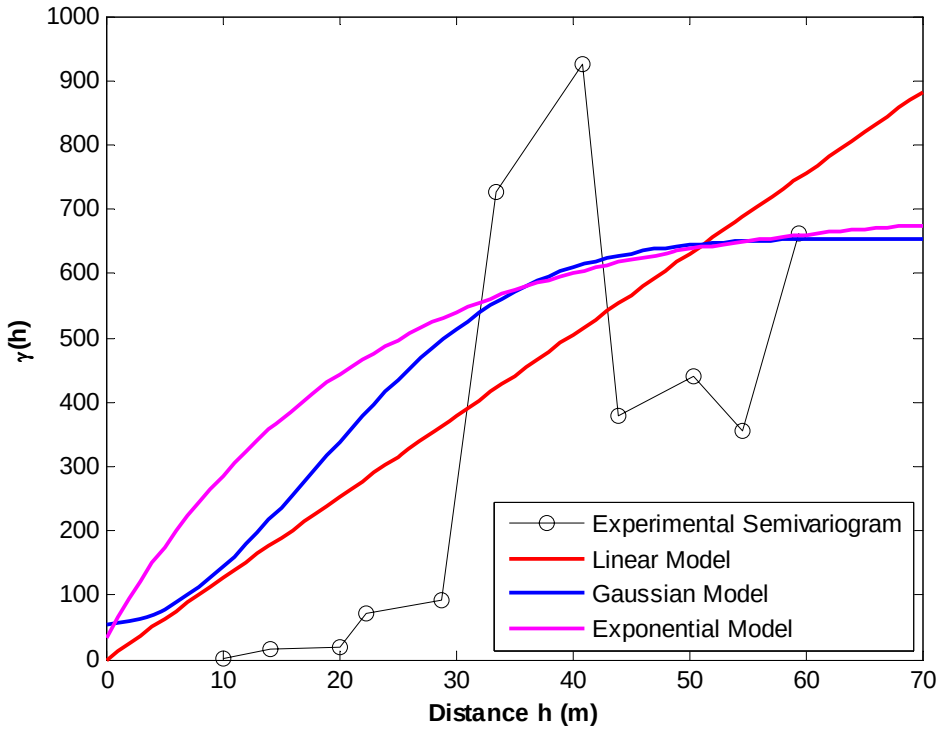

Figure 7.5. Semivariogram Models for Sampling Data

According to the Figure 7.5, best fit was obtained by the Gaussian model as

$$\gamma(h)=55+600\left(1-\exp\left(-\frac{h^2}{625}\right)\right) \tag{7.27}$$

For ANN approach, the x-coordinate and y-coordinate of each sample point were set as input data vector, and transformed values of corresponding transmissivity in logarithm base 10 were considered as the target vector for the training of the neural network. After the training of the network, the coordinates of the estimation points were

introduced to the network to map the transmissivity distribution of the entire aquifer. The tangent hyperbolic function was utilized as the transfer function in the hidden layer. Table 7.4 summarizes the ANN model configuration which was used to map transmissivity distribution.

Table 7.4. ANN Parameters for the Experiment

Parameter	**Value**
Learning Rate	0.05
Momentum Constant	0.9
Number of Neuron in the hidden layer	8
Network Configuration	2-8-1
Error Criteria (MSE)	1.00E-06
Maximum Number of Epoch	1000

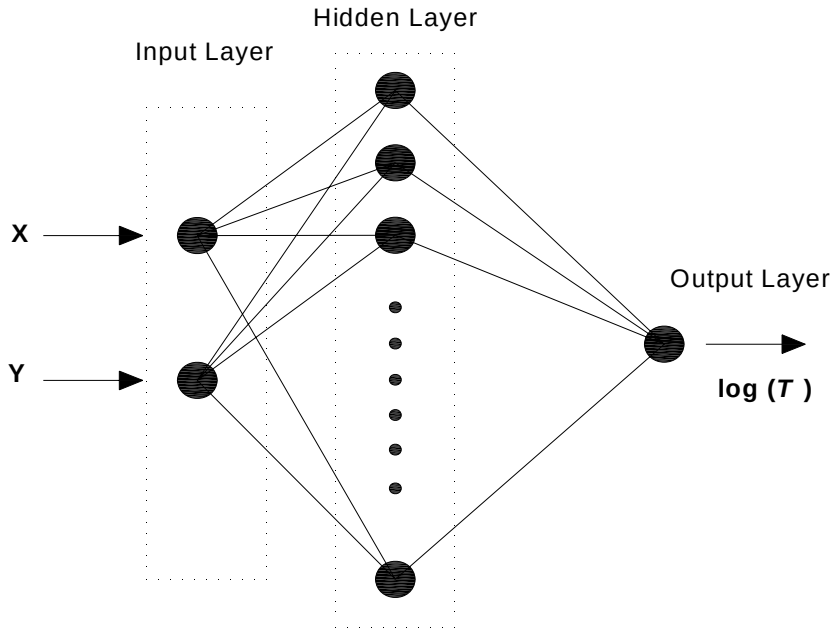

Figure 7.6. Schematic Illustration of ANN Model to Estimate Transmissivity

For the pump test analysis, four different monitoring wells were selected from the hypothetical confined aquifer. Each edge of the problem domain was assumed to be

constant head boundary conditions. The following figure characterizes the boundary conditions of the aquifer and the locations of monitoring wells. Model parameters used in pump test analysis were listed in Table 7.5.

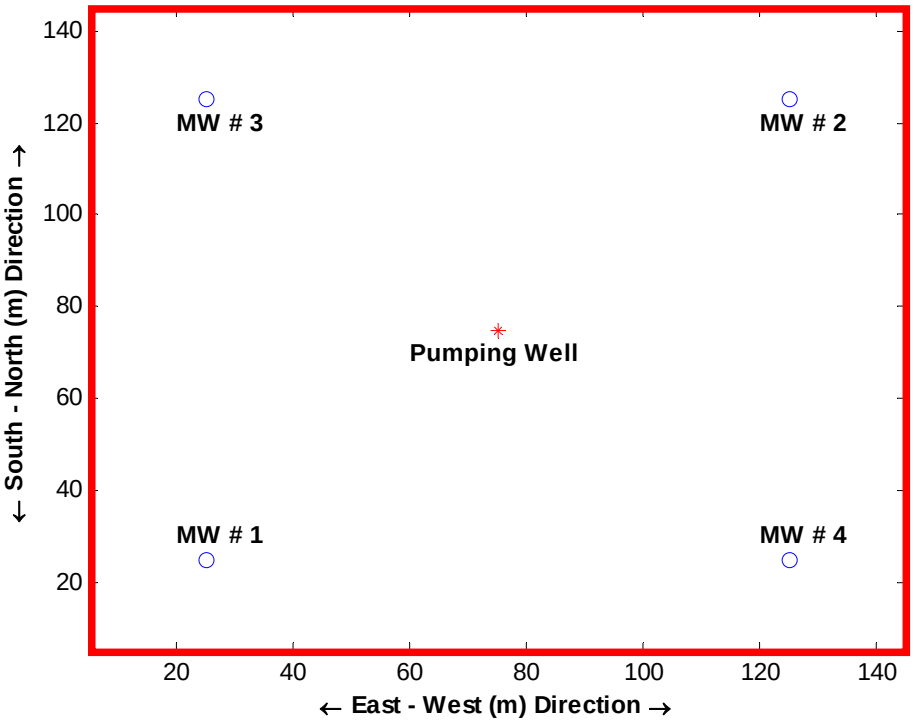

Figure 7.7. Locations of MWs and Boundary Conditions

Table 7.5. Model Parameters of Pumping Test

Model Parameter	Value
Initial Head (m)	100
Pumping Rate (m^3/day)	500
Aquifer Thickness (m)	20
Storage Coefficient	0.001
Effective Porosity	0.25
Specific Yield	0.25

7.4. Estimation Results of the Models

In Table 7.6, results that have obtained employed by RBFCM, Ordinary Kriging Method and ANN are presented. Maximum Absolute Error (*max*AE), Mean Absolute Error (MAE) and Standard Error (SE) were defined to improve the comparisons of the model as follows

$$\max AE = \max \left| y - \hat{y} \right| \tag{7.28}$$

$$MAE = \left| \frac{1}{N} \sum_{i=1}^{N} y_i - \hat{y}_i \right| \tag{7.29}$$

$$SE = \frac{\sqrt{\frac{1}{N} \sum_{i=1}^{N} (y_i - \hat{y}_i)^2}}{\overline{y}} \tag{7.30}$$

where y is the target value, $\hat{y}$ is the estimated values, $\overline{y}$ is the mean of the target values and N denotes the total number of data.

Table 7.6. Transmissivity Estimation of the Implemented Methods

Statistical Criteria	**RBF**	**OK (Gaussian Model)**	**ANN**
Min	6.92	6.99	6.87
Max	111.72	111.72	118.77
Mean	49.50	45.57	43.70
Standard Deviation	31.85	25.00	29.87
Variance	1014.48	625.24	892.12
maxAE (Max. Absolute Error)	65.72	31.78	28.06
MAE (Absolute Mean Error)	10.73	8.20	7.30
RMSE	17.47	11.31	10.27
Standard Error (SE)	0.35	0.25	0.24
R^2	0.7305	0.8606	0.8838

When the performances of the implemented methods are compared according to Table 7.6, ANN transmissivity estimation lead to better results rather than the conventional

methods. Although, Ordinary Kriging also presented acceptable accuracy as well as ANN model, ANN approach can be preferred since Ordinary Kriging requires more application effort. The weakness of the ANN model is the robustness of the solution when the insufficient data are available. To increase the stability of ANN model, some of the sampling data which are not used during training process can be assigned as testing patterns. Experiment results imply that ANN model can be employed as an alternative method to estimate unknown values when sufficient data are collected from the field. In RBFCM, there is no need for a pre-processing procedure such as selection of variogram in OK or training scheme in ANN. Thus, RBF method can be directly applied to data interpolation problem. In Figure 7.8, 7.9, and 7.10, transmissivity distribution of RBFCM is shown. The transmissivity map of OK method is illustrated in Figure 7.11, 7.12 and 7.13. The estimated transmissivity values of the ANN model are represented in Figure 7.14, 7.15 and 7.16.

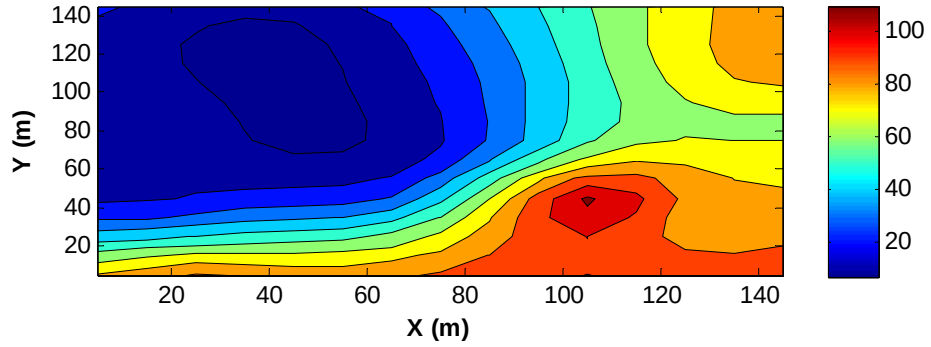

Figure 7.8. Transmissivity Distribution by RBFCM

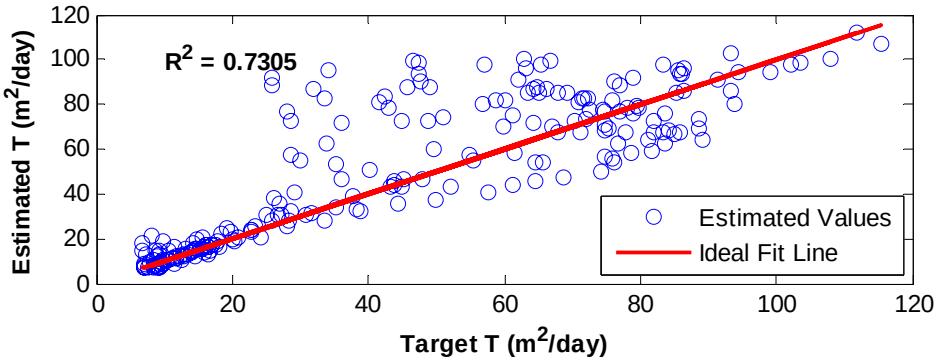

Figure 7.9. Regression Analysis for RBFCM Solution

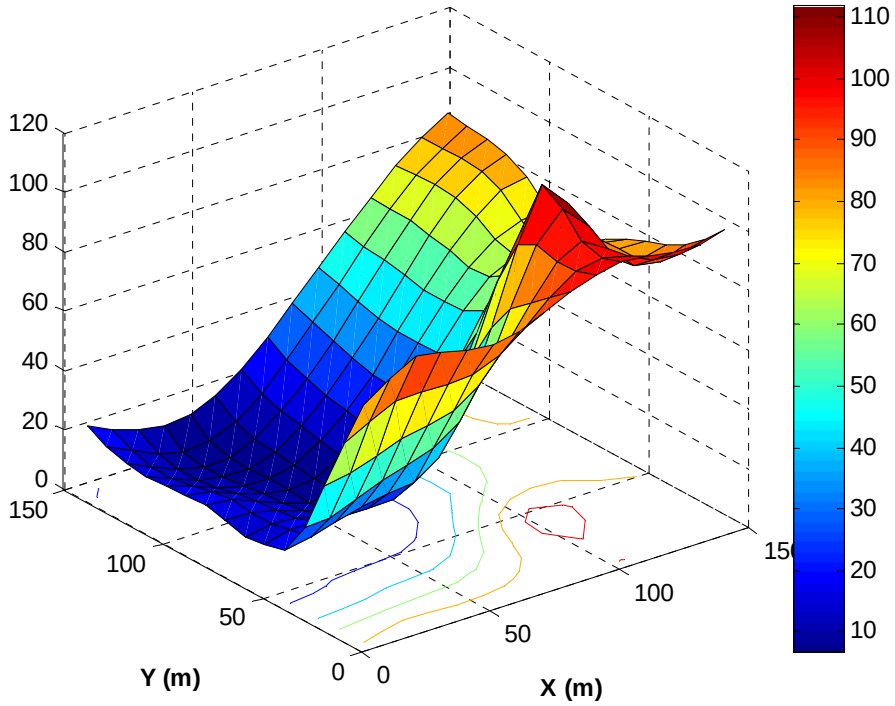

Figure 7.10. 3-D View of Transmissivity Map for RBFCM Solution

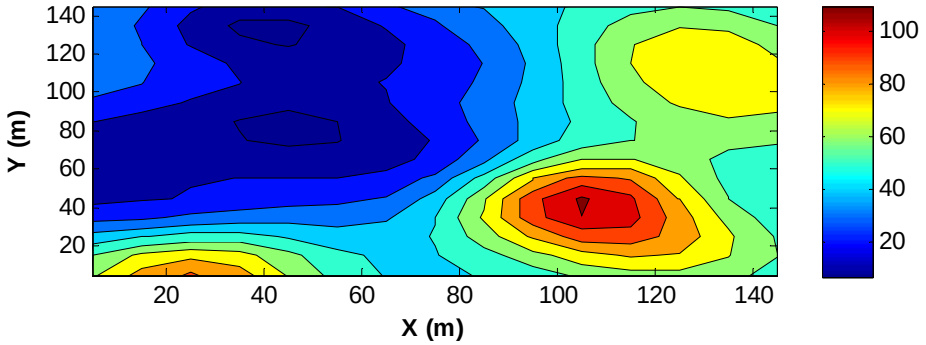

Figure 7.11. Transmissivity Distribution by OK (Gaussian Model)

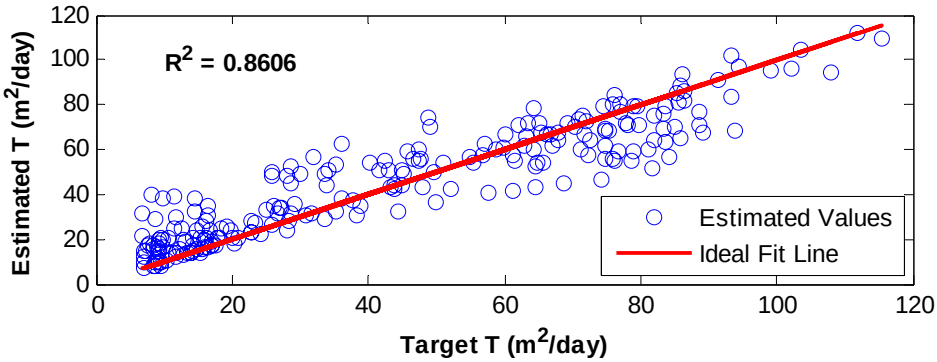

Figure 7.12. Regression Analysis for OK Solution

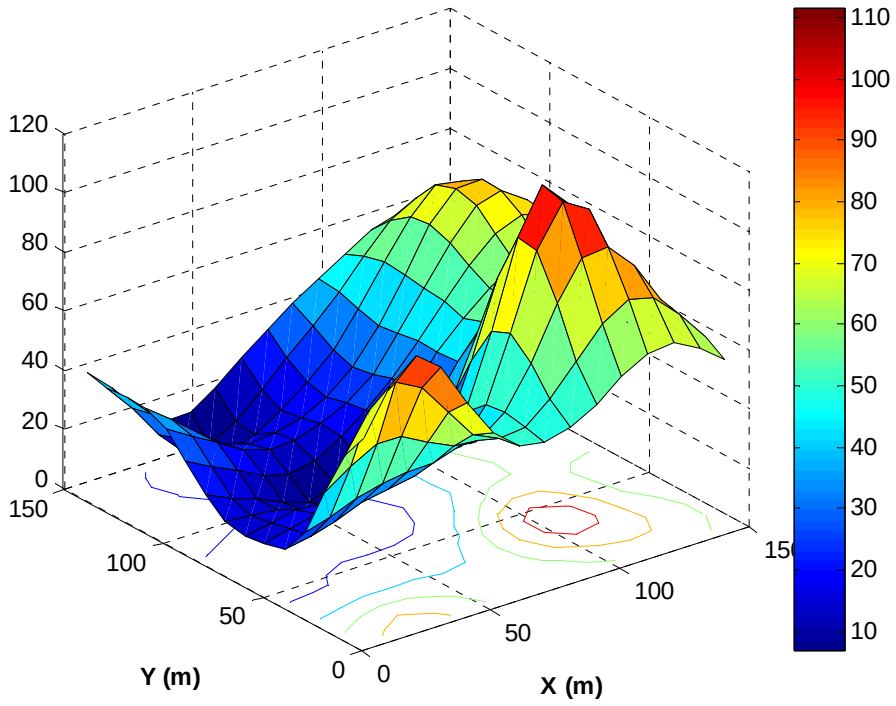

Figure 7.13. 3-D View of Transmissivity Map for OK Solution

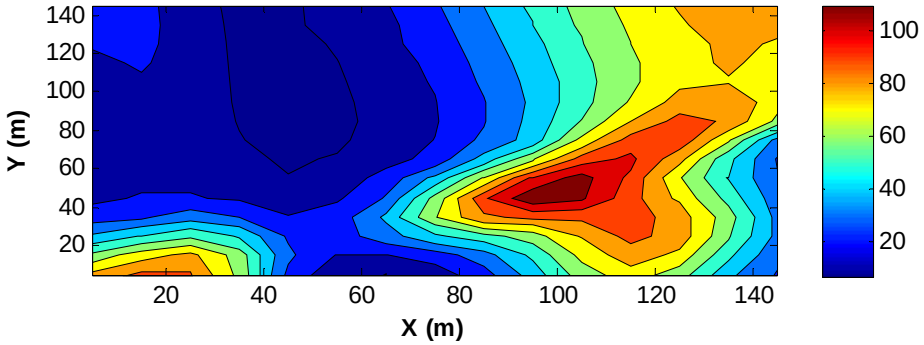

Figure 7.14. Transmissivity Distribution by ANN

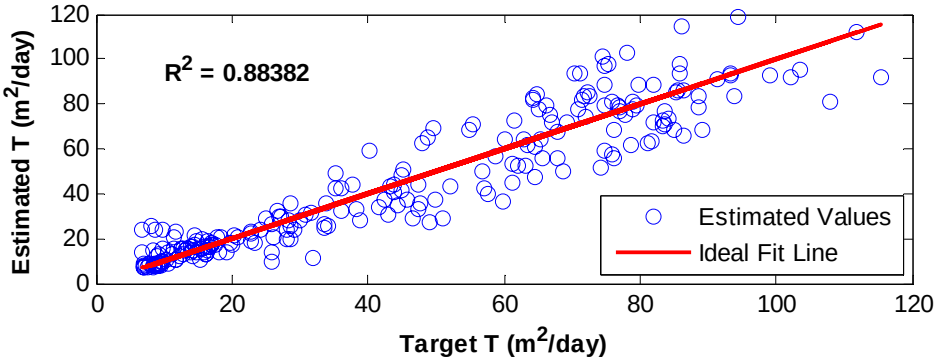

Figure 7.15. Regression Analysis for ANN Solution

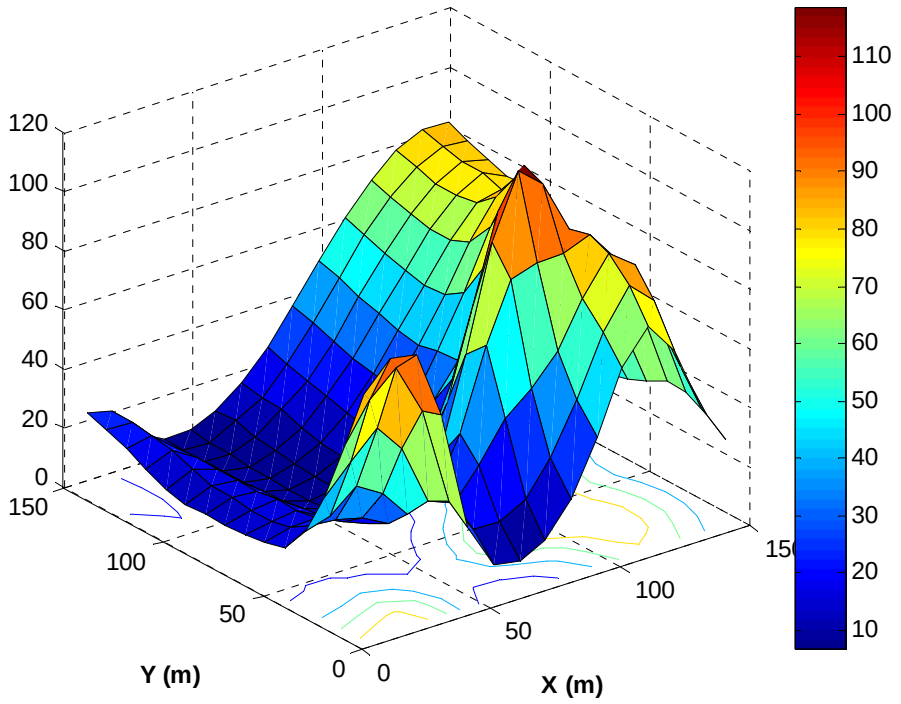

Figure 7.16. 3-D View of Transmissivity Map for ANN Solution

Estimated time-drawdown curves of the monitoring wells are depicted in Figure 7.17.

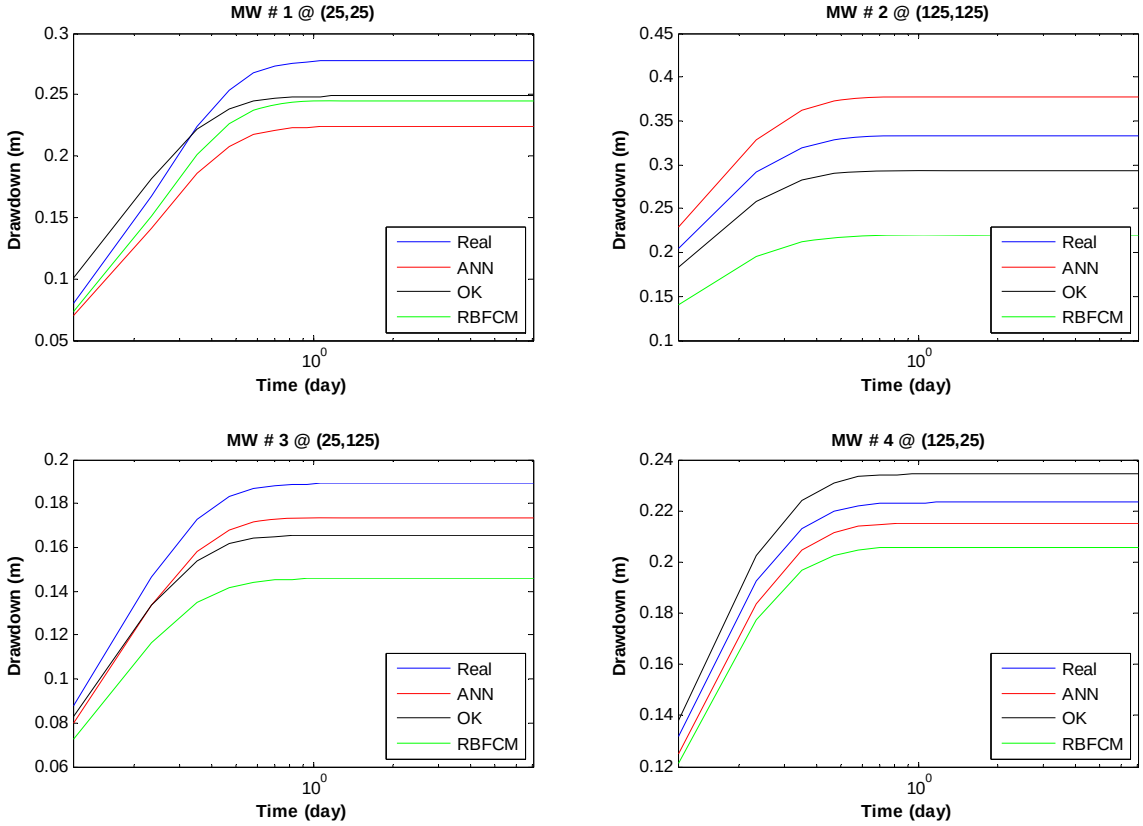

Figure 7.17. Time-Drawdown Curves of MWs

Table 7.7. The Estimation Performance of the Model on the monitoring wells

	ANN	OK	RBFCM
Well No	**RMSE**	**RMSE**	**RMSE**
1	0.0271	0.0144	0.0164
2	0.0228	0.0207	0.0582
3	0.0079	0.0120	0.0221
4	0.0041	0.0057	0.0089

In Table 7.7, the estimation errors of each method have been detailed. According to the above figure and table, ANN model performed better drawdown estimations at MW # 3

and MW # 4. Although OK method generated the best estimation in MW # 2, ANN approach was also estimated the drawdown as well as OK method. The overall drawdown estimations of each model on the entire domain are summarized in Table 7.8.

Table 7.8. Overall Drawdown Estimations Performance of the Models

	Error Criteria		
Model	**MAE**	**RMSE**	**R^2**
ANN	0.0648	0.1283	0.9947
OK	0.0976	0.1954	0.9910
RBFCM	0.1018	0.1623	0.9912

From the drawdown estimation performance point of view, ANN model has again proved to be alternative method for transmissivity mapping of the aquifer.

8. CONCLUSIONS

The primary aim of this research was to review the ability of ANN as a universal function approximator for estimating aquifer parameter and mapping of aquifer transmissivity. The investigation took place in two parts: in the first part, the existing ANN approaches as presented in the literature was applied to determine aquifer parameters estimation for confined aquifers. In this approach, a new ANN approach was proposed to determine leaky confined aquifer parameters. The ANN results were compared to conventional curve matching procedures to review the performance of ANN models. In the second part of the research work, a numerical experiment was conducted to compare the mapping performance of the ANN model where the Radial Basis Function method and Ordinary Kriging method for limited sampling transmissivity data.

According to obtained results, an ANN model approach provides a successful alternative for aquifer parameters determination and transmissivity mapping. In the determination of aquifer parameters problem, ANN produced a better estimation parameter value with respect to the conventional methods. The potential errors due to curve matching procedures disappeared when the ANN model was utilized. Another advantage of the ANN model is that it can be easily applied to any time-drawdown data that was not recorded in regular time interval.

An ANN model can also be utilized to map the transmissivity distribution of an aquifer. Such ANN model is recommended as an alternative technique to Ordinary Kriging or RBFCM in the interpolation problems since it is more easily applied than Kriging method and also performs better approximation values than classical methods. The ANN model has a simpler application procedure than Ordinary Kriging method that requires the selection of appropriate variogram model. In the Kriging technique, the accuracy of approximation is very sensitive to variogram model. Therefore, selecting variogram model is a time-consuming and error producing work. The variogram selection is not needed if ANN model is employed.

The main weakness of the ANN models is the need to determine a network configuration which performs an optimum approximation. According to the ANN theory, ANN models process the implicit relation between input and target vectors. In other words, the higher performance is obtained when proper input-target data set that reflect the domain character is selected. Even if the ANN is trained by a proper training data set, ANN models should be tested with the testing patterns which are not used during training of the network. In this research, ANN model for the transmissivity map of a hypothetical aquifer was not tested with testing patterns since there were limited site data available. This problem can be overcome if a large data set is available, which can then be divided for training and test patterns of the network.

In conclusion, neural network models have been successfully applied to determine aquifer parameters from pumping tests and to estimate transmissivity distribution of an aquifer. Considering this research and the previous studies, ANN models should be able to find more application areas in hydrology as an alternative to already established conventional techniques.

REFERENCES

Adeli, Hojjat, 2001, "Neural Networks in Civil Engineering: 1989-2000", *Computer Aided Civil and Infrastructure Engineering*, 16, pp. 126-142.

ASCE Task Committee, 2000 a., "Artificial Neural Networks in Hydrology I: Preliminary Concepts", *Journal of Hydrologic Engineering*, Vol.5, No.2, pp. 115-123.

ASCE Task Committee, 2000 b., "Artificial Neural Networks in Hydrology II: Hydrologic Applications", *Journal of Hydrologic Engineering*, Vol. 5, No. 2, pp. 115-123.

Aziz, A.R.A., and K.V. Wong, 1992, "A Neural Network Approach on the Determination of Aquifer Parameters", *Ground Water*, 30 (2):164-166.

Balkhair, S. Khaled, 2002, "Aquifer Parameters Determination for Large Diameter Wells Using Neural Network Approach", *Journal of Hydrology*, 265: 118-128.

Börekçi, Osman, 2005, *The Radial Basis Function Collocation Method: A Meshless Method for the Solution of Boundary Value Problems*, Lecture Notes, Boğaziçi University, İstanbul.

Coppala, Emery Jr., Ferenc Szidarovszky, Mary Poulton, Emmanuel Charles, 2003, "Artificial Neural Network Approach for Predicting Transient Water Levels in a Multilayered Groundwater System under Variable State, Pumping, and Climate Conditions", *Journal of Hydrologic Engineering*, Vol. 8, No. 6, pp. 348-360.

Demuth, Howard, Mark Beale, and Martin Hagan, 2006, *Neural Network Toolbox User's Guide*, The MathWorks Inc.

Deutsch, Clayton V., and André G. Journel, 1998, *GSLIB Geostatistical Software Library and User's Guide*, 2nd edition, Oxford University Press, New York.

Domenico, Patrick A., and Franklin W. Schwartz, 1990, *Physical and Chemical Hydrogeology*, John Wiley & Sons. Inc., New York.

El Tabach, Eddy, Louren Lancelot, Isam Shahrour, Yacaub Najjar, 2007, "Use of Artificial Neural Network Simulation Metamodelling to Asses Groundwater Contamination in a Road Project", *Mathematical and Computer Modeling*, 45: 766-776.

Fetter, C.W., 2001, *Applied Hydrogeology*, 4th edition, Prentice- Hall., New Jersey.

Flood, I., and N. Kartam, 1994 a., "Neural Networks in Civil Engineering I: Principal and Understanding, *Journal of Computation Civil Engineering*, Vol. 8, No. 2, pp. 131-148.

Flood, I., and N. Kartam, 1994 b., "Neural Networks in Civil Engineering II: Systems and Applications, *Journal of Computation Civil Engineering*, Vol. 8, No. 2, pp. 149-162.

Freeze, R.A., and J.A. Cherry, 1979, *Groundwater*, Prentice Hall, New Jersey.

Hagan, Martin, and Mohammed B. Menhaj, 1994, "Training Feedforward Networks with the Marquardt Algorithm", *IEEE Transactions on Neural Networks*, Vol. 5, No. 6, pp. 989-993.

Hagan, Martin, Howard B. Demuth, Mark Beale, 1996, *Neural Network Design*, Thomson Learning Publisher, New York.

Hantush, M.S., 1956, "Analysis of Data from Pumping Test in Leaky Aquifers", *Trans. Am. Geophysics Union*, Vol. 65, pp. 702-714.

Hantush, M.S., and C.E. Jacob, 1955, "Non-steady Radial Flow in an Infinite Leaky Aquifer", *Trans. Am. Geophysics Union*, Vol. 36, pp. 95-100.

Hardy, R.L., 1971, "Multiquadric Equations of Topography and Other Irregular Surfaces", *Geophys. Res*., 176: 1905-1910.

Haykin, Simon, 1994, *Neural Networks: A Comprehensive Foundation*, Macmillan College Publishing Company, New York.

Hon, Y.C., K.F. Cheung, and E.J. Kansa, 1999, "Multiquadric Solutions for Shallow Water Equations", *Journal of Hydraulic Engineering*, 125:524-533.

Hu, Y. Hen, and Jenq-Neng Hwang, (editors), 2002, *Handbook of Neural Network Signal Processing*, CRC Press LLC.

Isaaks, H. Edward, and R.M. Srivastava, 1989, *An Introduction to Applied Geostatistics*, Oxford University Press, New York.

Kansa, E.J, 1990 a., "Multiquadrics - A Scattered Data Approximation Scheme with Application to Computational Fluid Dynamics-I, Surface Approximations and Partial Derivative Estimations", *Computer & Mathematics with Applications*, Vol. 19, pp. 127-145.

Kansa, E.J, 1990 b., "Multiquadrics - A Scattered Data Approximation Scheme with Application to Computational Fluid Dynamics-II, Solutions to Parabolic, Hyperbolic and Elliptic Partial Differential Equations, *Computer & Mathematics with Applications*, Vol. 19, pp. 147-161.

Kholgi, M, and S.M. Husseini, 2006, "Estimation of Aquifer Transmissivity using Kriging, Artificial Neural Network, and Neuro-Fuzzy Models", *Journal of Spatial Hydrology*, Vol. 6, No.2, pp. 68-81.

Krigie, D.G., 1951, *A Statistical Approach to Some Mine Valuations and Allied Problems at the Witwatersrand,* Master Thesis, University of Witwatersrand, South Africa.

Kröse, Ben and Patrick van der Smagt, 1996, *An Introduction to Neural Networks*, 8th edition, University of Amsterdam.

Kubat, Miroslav, 2000, "Designing Neural Network Architectures for Pattern Recognition", *The Knowledge Engineering Review*, Vol. 15:2, pp. 151-170.

Lee, Stanley E., 2000, "Neuro-Fuzzy Estimation in Spatial Statistics", *Journal of Mathematical Analysis and Applications*, 249: 221-231.

Lin, Gwo-Fong and Guo-Rong Chen, 2006, "An Improved Neural Network Approach to the Determination Aquifer Parameters", *Journal of Hydrology*, 316: 281-289.

Maier, H. R., and G. C. Dandy, 1996, ''The Use of Artificial Neural Networks for the Prediction of Water Quality Parameters'', *Water Resour. Res*., 32 (4): 1013–1022.

Mantoglou, Aristotelis and John L. Wilson, 1981, *Simulation of Random Fields with the Turning Band Methods*, Ralph M. Parson Lab, Technical Report No: 264, MIT Cambridge.

McCullloch W., and W. Pitts, 1943, "A Logical Calculus of Ideas Imminent in Nervous Activity", *Bulletin of Mathematical Biophysics*, Vol. 5, pp. 115-133.

Morshed, Jahangir and Jagath J. Kaluarachchi, 1998, "Application of Artificial Neural Network and Genetic Algorithm in Flow and Transport Simulations", *Advances in Water Resources*, Vol. 22, No 1, pp.145-158.

Nayak, Purna C., Y.R. Satyaji Rao, K.P. Sudheer, 2006, "Groundwater Level Forecasting in a Shallow Aquifer Using Artificial Neural Network Approach", *Water Resources Management*, 20: 77-90.

Pezeshk,S, C.V. Camp, S. Karprapu, 1996, "Geophysical Log Interpretation Using Neural Network", *Journal of Computing in Civil Engineering*, Vol.10, No. 2, pp. 136-142.

Principe, Joce C., Neil R. Euliano, W. Curt Lefebvre, 2000, *Neural and Adaptive Systems: Fundamentals through Simulations*, John Wiley & Sons. Inc., New York.

Prodanoff, J.H.A., and W.J. Mansur, F.C.B. Mascarenhas, 2006, "Numerical Evaluation of Theis and Hantush-Jacob Well Functions", *Journal of Hydrology,* 318: 173-183.

Rosenblatt, F, 1958, "The Perceptron: A Probabilistic Model for Information Storage and the Organization in the Brain", *Psychological Review*, Vol.65, pp. 386-408.

Samani, N., M.G. Moghadam, A.A. Safavi, 2007, "A Simple Neural Network Model for the Determination of Aquifer Parameters", *Journal of Hydrology*, 340: 1-11.

Schwartz, W. Franklin and Hubao Zhang, 2003, *Fundamentals of Groundwater*, John Wiley & Sons Inc., New York.

Singh, Raj Mohan and Bithin Datta, 2004, "Groundwater Pollution Source Identification and Simultaneous Parameter Estimation Using Pattern Matching by Artificial Neural Network", *Environmental Forensics*, 5: 143-153.

Surfer Version 8, 2002, *Radial Basis Functions*, Help Files, Golden Software Inc.

Theis, C.V., 1935, "The Relation between the Lowering of the Piezometric Surface and the Rate and Duration of Discharge of a Well Using Groundwater Storage", *American Geophysical Union Transcript*, Vol. 2, No. 16, pp. 519-524.

Thiem, G., 1906, *Hydrologische Methode*, Gebhardt, Leipzig.

Uddameri V, 2007, "Using Statistical and Artificial Neural Network Models to Forecast Potentiometric Levels at a Deep Well in South Texas, *Environmental Geology*, 51: 885-895.

Walton, W.C., 1962, *Leaky Artesian Aquifer Conditions in Illinois*, Illinois State Water Survey, Illinois.

Wenzel, L.K., 1942, "Methods for Determining Permeability of Water Bearing Material with Special Reference to Discharging Well Methods", *U.S. Geological Survey Water Supply Paper*, 887.

Wu, Zongmin and Y.C. Hon, 2003, "Convergence Error Estimate in Solving Free Boundary Diffusion Problem by Radial Basis Functions Method", *Engineering Analysis with Boundary Elements*, 27: 73-79.