A COMPRESSIVE CONCRETE STRENGTH PREDICTION MODEL USING ARTIFICIAL NEURAL NETWORKS

ZANG GUOJI

INTELLIGENT SYSTEM, MASTER OF SCIENCE

UNIVERSITI UTARA MALAYSIA

2017

Permission to Use

In presenting this project in partial fulfillment of the requirements for a postgraduate degree from the Universiti Utara Malaysia, I agree that the University Library may make it freely available for inspection. I further agree that permission for copying of this project in any manner in whole or in part, for scholarly purposes may be granted by my supervisor(s) or in their absence by the Dean of Postgraduate Studies and Research. It is understood that any copying or publication or use of this project or parts thereof for financial gain shall not be allowed without my written permission. It is also understood that due recognition shall be given to me and to Universiti Utara Malaysia for any scholarly use which may be made of any material from my project.

Requests for permission to copy or to make other use of materials in this project, in whole or in part, should be addressed to

Dean of Postgraduate Studies and Research

College of Arts and Sciences

Universiti Utara Malaysia

06010 UUM Sintok

Kedah DarulAman

Malaysia

Abstract (English)

A building is at a high risk of destruction if the compressive concrete strength does not meet the required specification. Thus, the prediction of compressive concrete strength has become an important research area. Previous prediction models are based on fix numbers of attributes. Consequently, when the number of attributes increase or decrease, the models could not be used. Thus, a compressive concrete strength prediction model which can work with different numbers of attribute is needed. The purpose of this study is to develop compressive concrete strength prediction models using different combinations of attributes. This study includes five stages: data collection, normalization, parameters identification, model construction and evaluation. The employed data set consists of nine attributes: water, cement, fine aggregate, coarse aggregate, age, fly ash, super plasticizer, blast furnace slag and compressive concrete strength. This study produced eight prediction models where each model has different combination of attributes. It also identified appropriate weights, learning rate, momentum and number of hidden nodes for each of the proposed model, and design a general artificial neural network (ANN) architecture. Model eight of the study produced a higher correlation coefficient (i.e., 0.973) than the existing study (i.e., 0.953). This study has successfully produced eight concrete strength prediction models with good coefficient correlation. The compressive strength prediction models would benefit civil engineers as they can use the models to identify the suitability of additional materials in concrete mix.

Keywords: Compressive concrete strength, Different combinations of attributes, Artificial neural networks, Prediction models.

Abstrak (Bahasa malaysia)

Sesebuah bangunan adalah berisiko tinggi untuk runtuh jika kekuatan mampatan konkrit tidak memenuhi spesifikasi yang dikehendaki. Oleh itu, ramalan kekuatan mampatan konkrit telah menjadi satu topik penyelidikan yang penting. Model ramalan sebelum ini adalah berasaskan kepada bilangan atribut yang tetap. Akibatnya, apabila berlaku peningkatan atau penurunan bilangan atribut, model tersebut tidak boleh digunakan. Oleh itu, model ramalan kekuatan mampatan konkrit yang boleh berfungsi dengan bilangan atribut yang berlainan adalah diperlukan. Tujuan kajian ini adalah untuk membangunkan model ramalan kekuatan mampatan konkrit yang menggunakan kombinasi atribut berlainan. Kajian ini merangkumi lima peringkat: pengumpulan data, penormalan, pengenalpastian parameter, pembinaan model dan penilaian. Data set yang digunakan terdiri daripada sembilan atribut: air, simen, agregat halus, agregat kasar, usia, abu terbang, *super plasticizer*, sanga relau bagas dan kekuatan mampatan konkrit. Kajian ini menghasilkan lapan model ramalan yang mana setiap model mempunyai kombinasi atribut yang berbeza. Kajian itu juga mengenalpasti berat, kadar pembelajaran, momentum dan bilangan nod tersembunyi yang sesuai untuk setiap model ramalan yang dicadangkan, dan rekabentuk umum seni bina rangkaian neural buatan (ANN). Model lapan dalam kajian ini menghasilkan pekali korelasi yang lebih tinggi (0.973) daripada kajian yang sedia ada (0.953). Kajian ini telah berjaya menghasilkan lapan model ramalan kekuatan mampatan konkrit dengan pekali korelasi yang baik. Model ramalan kekuatan mampatan konkrit ini akan memberi manfaat kepada jurutera awam untuk mengenal pasti kesesuaian bahan tambahan untuk campuran konkrit.

Kata Kunci: Kekuatan konkrit mampatan, Kombinasi sifat-sifat, Rangkaian neural buatan, Model ramalan.

Acknowledgement

Heartfelt thanks are to my dear supervisor, Prof. Dr. Faudziah Ahamad for patiently navigating and generously sharing her rich source of knowledge with me. She always give me a hand when I cannot progress my dissertation. She is indeed a teacher of "pen hand, open mind and open heart".

Equally, I need to thanks to my parents for all their love, unlimited help and the great support they offered to me.

I also need to say "thank you" to the friends around me for giving me suggestions and sharing their experiences with me. They really help me a lot.

In the end, thanks to Universiti Utara Malaysia for supporting me a good place for studying knowledge and experiences.

Table of Contents

List of Tables

List of Figures

List of Appendices

CHAPTER ONE

INTRODUCTION

1.1 Background

Concrete is one of the most indispensable building and engineering material in the world. It has been used for more than 10 decades (Aggarwal, Kumar, Sharma, & Sharma, 2015). Concrete becomes more and more popular in the world because of its capabilities. For example, it can take up any shape before it becomes hard, and strengthens when it hardens. This construction material is widely used in buildings, bridges, roads, runways, docks, military engineering, nuclear power stations and so on (Wankhade & Kambekar, 2013). If there is a high quality building, it must have a strong compressive strength of concrete. Because of this, compressive concrete strength becomes an important element building construction. If the compressive concrete strength do not meet the required specification for a building then there will a high risk of destruction when unfortunate incidents happened such as natural disasters or damages caused by humans. For example on May 12, 2008, an earthquake of magnitude 7.9, struck western Sichuan province causing many buildings to be destroyed and casualties. Many experts agree that casualties and damages could have been avoided if the buildings were built using high quality components. The question is, how quality is the buildings? Obviously, when such catastrophic incident occur, the buildings can said to be below the quality standard – that is, the compressive concrete strength was below than the standard procedure(Michele et al., 2010). Again on July28, 1976, the city of Tangshan, China was struck by a 7.8magnitude earthquake. Many people were injured and buildings were severely destroyed (Huixian, Housner, Lili, & Duxin, 2002). Thus, these disasters have attracted many researchers, especially to predict concrete compressive strength so that new buildings are safe to withstand disasters such as earthquake or equivalent incidents.

Over the last decade, artificial neural network (ANN), have become popular and have been used by many scholars to solve engineering related problems. The positive side of ANN is that there is no requirement for assuming a model form and do not need to make any specific equation form. ANN automatically handles the relationships among variables and adapt according to the data used for their training. So using a large number of experimental data, a model can be developed (Rasa, Ketabchi, & Afshar, 2009). According to Muthupriya, Subramanian, & Vishnuram (2011), ANN is like a powerful and useful weapon that can handle classification and able to learn based on samples or existing datasets well.

In terms of prediction of compressive concrete strength, for instance, Vakhshouri & Nejadi (2015) constructed an Adaptive Neural-Fuzzy inference System incorporating both neural networks and fuzzy systems to predict compressive strength of concrete. In another study, Aggarwal et al (2015) used Multiple Regression to predict strength of concrete. Gilan, Ali, & Ramezanianpour (2011) used fuzzy function which based on support vector regression also to forecast compressive strength of concrete.

In terms of attributes used for prediction, different approaches were used. Since, the main attributes of concrete are cement, aggregate (sand, stone) and water; most researchers would use all of the attributes to make concrete (Ferraris, 1999). Others would add other attributes such as Blast Furnace Slag, Fly Ash, Super plasticizer, Coarse Aggregate, Fine Aggregate and Age (Wankhade & Kambekar, 2013).

Based on the above discussion, it can be seen that previous prediction models are static. Static prediction models always focus on specified number of attributes. However, when the numbers of attributes increase or decrease, the models will not work anymore. This indicates that static models have a limitation – that is these models cannot work with different numbers of attributes. Thus, a compressive concrete strength model which can work with different combinations of attributes is needed.

In this study, several different set of attributes were used. In total the number of attributes used was 9. Data for this study was taken from the University of California, Irvine (UCI)'s repository. The dataset consists of 9 attributes and 1030 instances. This dataset is the data that were used to perform prediction by Yeh (1998). The last attribute, compressive strength of concrete is the dependent attribute, while other 8 attributes are the independent attributes. The following information describes the dataset in greater detail. Out of 9 attributes, five attributes are the basic attributes (cement, water, age, coarse aggregate and fine aggregate) and is shown in Table 2.1

and Table 2.2. Three attributes, fly ash, super plasticizer and blast furnace slag are additional attributes. The final attribute (compressive concrete strength) is the class or defendant attribute.

1.2 Problem Statement

Compressive strength of concrete is one of the most important and useful properties that is employed to resist compressive stresses. However, at locations where tensile strength or shear strength is of primary importance, the compressive strength is used to measure properties of hardened concrete (Gupta, 2007). Even though most of the existing studies obtain good accuracy on predicting compressive concrete strength, their models still have some weaknesses. In general, the problems relate to attributes. Specifically, previous researches only used specific attributes to predict compressive concrete strength.

In civil engineering, engineers will not always use the same attributes to make concrete. In usual situation, engineers only use some main attributes to predict compressive concrete strength (Rasa, Ketabchi, & Afshar, 2009). But in other situations, civil engineers need to add extra materials (make the strength of concrete stronger) to predict the compressive concrete strength (Wankhade & Kambekar, 2013; Martinez-Molina et al., 2014; Nikoo, Torabian Moghadam, & Sadowski, 2015; De Melo & Banzhaf, 2016).

But existing models have not explored using basic attributes with additional attributes to predict compressive concrete strength. Because of this, a prediction model which can use the basic attributes and additional attributes for predicting compressive strength of concrete is needed (Rasa, Ketabchi, & Afshar, 2009;Deepa, Sathiya Kumari, & Pream Sudha, 2010; Alilou & Teshnehlab, 2010; Gilan et al., 2011; Muthupriya et al., 2011; Kabir, Hasan, & Miah, 2012; Kabir, Hasan, & Miah, 2013;Martinez-Molina et al., 2014; Nikoo, Torabian Moghadam, & Sadowski, 2015; De Melo & Banzhaf, 2016).

1.3 Research Questions

The main question is can a prediction model predict compressive concrete strength with good correlation coefficient when new materials are added to the basic prediction model?

Specific question would be:

1) What are the basic attributes for predicting compressive strength of concrete?

2) What is the suitable technique for predicting concrete compressive strength?

3) What are the suitable parameters for Weights, Learning Rate, Momentum, numbers of hidden layers and numbers of hidden nodes that can be used to construct a compressive concrete strength prediction model?

1.4 Research Objectives

The main objective of this study is to construct a prediction model which could predict compressive concrete strength accurately using basic attributes with additional attributes.

Specific objectives are:

1) to identify the basic attributes that can predict compressive concrete strength with good correlation coefficient;

2) to identify additional attributes that can be used to predict compressive concrete strength with good correlation coefficient;

3) to determine suitable parameters for weights, learning rate, momentum and numbers of hidden nodes.

4) to design a main ANN architecture for predicting compressive strength of concrete and construct a compressive concrete strength prediction model.

1.5 Significance of the study

The study will benefit civil engineers. This work supports the combinations of attributes (basic attributes + additional attributes) to predict compressive strength of concrete for civil engineers. It explores ANN architectures (it includes learning rate, momentum, number of hidden layer and number of hidden nodes) for prediction. So, civil engineers can use the ANN architectures to predict the compressive concrete strength.

1.6 Scope of this study

This study used the secondary dataset from Yeh et al. (1998). The dataset contains 1030 instances, and 9 attributes (age, water, cement, fine aggregate, coarse aggregate, super plasticizer, fly ash, blast furnace slag and compressive strength of concrete). The compressive concrete strength is the output (target). Based on the features and different combinations of attributes (5 basic attributes + 3 additional attributes), this secondary data set was separated into 8 sets of data (Table 3.2).

1.7 Thesis organization

This dissertation report is separated into six chapters. Chapter One is the background and introduction about concrete and compressive concrete strength. It also describes the problem statement, research questions, objectives, significance of the study, and scope of this study. Chapter Two presents the literature review which includes the information about compressive strength of concrete, attributes existing scholars used to predict compressive concrete strength, existing techniques for prediction, and artificial neural networks. Chapter Three discusses the methodology used in this study. The methodology consists of five main phases which are Data Collection, Normalization, Determine Parameters, Model Construction and Evaluations. Chapter Four presents the deliverables for objectives 1, 2, 3 and 4. Chapter Five discusses the evaluation results of 8 models, discussion, and comparison results between model 8 and one existing compressive concrete strength prediction model. Chapter Six highlights the overall achievement, contributions and future works of this study.

CHAPTER TWO

LITERATURE REVIEW

2.1 Theoretical background

This chapter includes three sections. Section 2.1.1 describes the background of concrete and the importance of concrete. Section 2.1.2 provides discussion on existing attributes used to predict compressive concrete strength and Section 2.1.3 discusses on various existing techniques that have been used for prediction.

2.1.1 Compressive strength of concrete

Concrete is an important and most common building material of civil engineering. It has useful capabilities such as able to take any shape before it solidifies and hardens strongly, giving a good strength. This construction material is widely used in buildings, bridges, roads, runways, docks, military engineering, nuclear power stations (Wankhade & Kambekar, 2013). In addition,concrete is an artificial conglomerate stone. That is, it includes several basic elements such as cement, fine aggregate, coarse aggregate and water.Using different amounts of elements will contribute to different compressive concrete strength (Chou, Chiu, Farfoura, & Altaharwa, 2011).

In the process of making concrete, civil engineers will add other materials, such as fly-ash, supper plasticizer and blast furnace slag to improve the property of concrete. In simple words, civil engineers will use other materials to make the compressive strength of basic concrete stronger.Basic concrete in general consist of materials such as cement, fine aggregat, coarse aggregate and water (Yeh et al., 2003).

The issue of damages after an earthquake is serious and at most times frightening. People cannot stop earthquake, but people can avoid unnecessary losses. Therefore, compressive concrete strength plays an important role because buildings' damages can be reduced if the compressive concrete strength can withstand strong earth movements. The series of earthquakes that happened for example in British (2008), Yu shu, China (2010), New Zealand (2013) and Nepal (2015) caused buildings to collapse and many casualties (Musson, 2008; Bray et al., 2013;Jordans, Kohrt, & Tol, 2015). Thus, if concrete can be predicted for earthquake resistance, then buildings can be assured a safe place when such incidents happen. And according to Ghan, Peng, & Anson (1999), the high range of compressive concrete strength is between 70 to 140 MPa at 28 to 91 days and high-early strength is between 20 to 28 MPa at 3 to 12 hours or 1 to 3 days.

2.1.2 Attributes

Concrete consists of mixed materials. Some researchers defined the basic attributes such as water, fine aggregate and coarse aggregate (Ferraris, 1999). Others mentioned that it composed of cement, sand, aggregate, water, mineral admixtures and chemical admixtures (Liu, Sue, & Kou, 2009). A mixture of different materials will make different properties of concrete and in turn results to a different compressive concrete strength.

Previous studies on predicting compressive concrete strength use different composition of attributes. Table 2.1shows the attributes that have been used to predict compressive concrete strength and Table 2.2 shows the occurrences of the attributes in previous research works.

Table 2.1

Attributes used by existing researchers

NO.	AUTHORS	INDEPENDENT VARIABLES	DEPENDENT VARIABLES
1	Yeh (1998)	Cement, Blast Furnace Slag, Fly ash, Water, Super plasticizer, Coarse aggregate, Fine aggregate and age.	Compressive strength of concrete
2	Yeh (2003)	Cement, Blast Furnace Slag (BFS), Fly ash, Water, Super plasticizer, Coarse aggregate and Fine aggregate.	Compressive strength of concrete
3	Yaqub et al. (2006)	Water cement ratio, slump, cement content, age (days).	Compressive strength of concrete
4	Tanyildizi and Coskun (2007)	Cement (C), Fly ash (Fa), Aggregate, water (W) and	Compressive strength of concrete

		super plasticizer (SP).	
5	Rasa et al. (2009)	Water, Cement, Silica fume (SF), Super-plasticizer, Cement Type (CT).	Density and compressive strength
6	Bilim et al. (2009)	Cement, aggregate, age, blast furnace slag and plasticizer.	Compressive strength of concrete
7	Deepa et al. (2010)	Cement, Blast Furnace Slag, Fly ash, Water, Super plasticizer, Coarse aggregate , Fine aggregate and age.	Compressive strength of concrete
8	Atici (2011)	Age, cement, Blast furnace slag and Fly ash,	Compressive strength of concrete
9	Hasan and Kabir (2011)	Coarse aggregate, fine aggregate, cement, water, fineness modulus of sand and age (days).	Compressive strength of concrete
10	Muthupriya et al. (2011)	Age, Cement, Silica fume, Fly-ash, Water, Sand, Aggregate, and Super plasticizer.	Compressive strength of concrete
11	Kabir et al. (2013)	Coarse aggregate, Fine aggregate, Cement, Water, Age and W/C ratio (WCR).	Compressive strength of concrete
12	Wankhade and Kambekar (2013)	Age, Water, Cement, Super plasticizer, Blast Furnace	Compressive strength of concrete

		Slag, Fly-Ash. Fine aggregate and Coarse Aggregate.	
13	Wilfridoet al. (2014)	Cement, Sand (S), Gravel (G) and Water.	Compressive strength of concrete
14	Aggarwal et al. (2015)	water, fine aggregate-binder ratio (FA), coarse aggregate-binder (CA) and binder content (BC).	Compressive strength of concrete
15	Melo and Banzhaf (2015)	Cement, Blast Furnace Slag, Fly ash, Water, Super plasticizer, Coarse aggregate , Fine aggregate and age of testing.	Compressive strength of concrete

Table 2.2

The occurrences of attributes from previous attributes

Past Work	C	W	Fa	FA	CA	Age	SP	BFS	BC	SF	WCR	S	G	CT
1	√	√	√	√	√		√							
2		√		√	√				√					
3	√	√					√			√				√
4	√	√	√	√	√		√	√						
5		√		√	√	√					√			
6	√	√										√	√	
7	√	√	√	√	√	√				√		√		
8	√	√	√	√	√	√	√	√						

9	√	√	√	√	√	√	√	√						
10	√	√	√	√	√	√	√	√						
11	√	√	√	√	√	√	√	√						
12	√	√		√	√	√						√		
13	√					√					√	√		
14	√		√			√		√						
15	√			√	√	√	√	√						
Occurrences	13	12	8	11	11	10	8	7	1	2	2	4	1	1

Based on Table 2.1 and Table 2.2, it can be found that the common attributes are water, cement, fine aggregate, coarse aggregate and age (the occurrences are more than 10). These common attributes have been mentioned as the basic concrete components (Chou et al., 2011). Therefore, cement, water, coarse aggregate, fine aggregate and age were chosen as the basic attributes in this study. According to Yeh (2006), fly ash, super plasticizer and blast furnace slag are mineral admixtures which can improve compressive concrete strength. Because of these, the three attributes were selected as three additional attributes in this study.

2.1.3 Past studies on prediction techniques

Prediction, as people understand, is considered as forecasting short-term changes of certain phenomena. Examples are predicting the temperature of tomorrow at a given location or forecasting which asset to best invest next year (Cesa-Bianchi & Lugosi, 2006). In general, prediction is done based on precious experiences or historical data. Table 2.3 shows the various techniques used by previous researchers.

Table 2.3

Various prediction techniques

Techniques	Authors	Correlation Coefficient	Root Mean Square Error	Mean Absolute Error
Support Vector Machine (SVM)	Gupta (2007)	0.9910	0.9100	
	Chou, Chiu, Farfoura and Taharwa (2011)	0.9197	6.7248	14.9052
	Akande et al., (2014)	0.9773	23.1400	4.8900
	Suhad and Abbas (2015)	0.9900	-0.3208	
Genetic Operation Tree	Yeh and Lien (2009)	0.8669		
Multiple Statistical Regression	Liu et al. (2009)	0.9622	24.0800	5.5000
Levenberg -Marquardt	Alilouand Teshnehlab (2010)	0.9944		5.1080
	Chou, Chiu, Farfoura and Taharwa (2011)	0.9428	7.0364	11.6444
Multiple Regression	Deepa et al. (2010)	0.7908	9.9054	7.6780
	Chou, Chiu, Farfoura and Taharwa (2011)	0.6906	11.6391	36.6473
Linear Regression	Deepa et al. (2010)	0.7009	11.1066	8.8388

M5P Model Tree	Deepa et al. (2010)	0.8872	7.1874	5.0080
ANN (Back-Propagation)	Yeh (2003)	0.9940		
	Rasa et al. (2009)	0.9947	0.0348	
	Yeh & Lien (2009)	0.9338		
	Muthupriya et al. (2011)	0.9724	2.3729	-1.1138
	Wankhade and Kambekar (2013)	0.98	2.4500	1.8300

Based on Table 2.3, it can be seen that the popular methods that have been used for prediction compressive concrete strengths are SVM (Suhad & Abbas, 2015), Genetic Operation Tree (Yeh & Lien, 2009), Multiple Statistical Regression (Liu et al., 2009), Levenberg-Marquardt (Alilou &Teshnehlab, 2010), Multiple Regression (Chou et al., 2011), Linear Regression (Deepa et al., 2010), M5P Model Tree (Deepa et al., 2010), and Back Propagation (ANN) (Wankhade & Kambekar, 2013).

Support Vector Machine (SVM) is one of the good techniques for prediction. It is a statistical learning algorithm that can be applied to both classification and regression problems (Akande et al., 2014). As Figure 2.1 shows, SVM fits a hyperplane or function between 2 different classes given a maximum margin parameter. This hyperplane attempts to separate the classes so that each falls on either side of the plane, and by a specified margin. There is a specific cost function for this kind of model which adjusts the plane until error is minimized (Kasi, 2015).

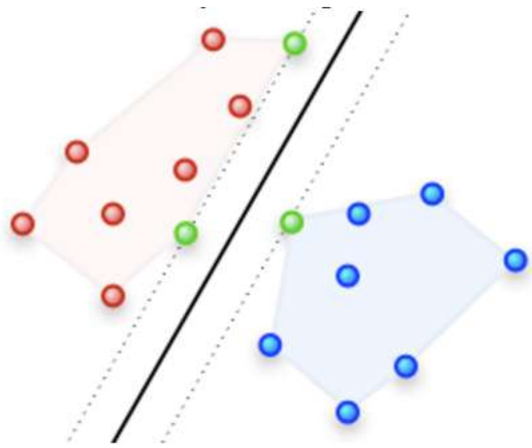

Figure 2.1: The general diagram of SVM
(http://algoholic.eu/faster-dot-product-for-svm/)

From Table 2.3, in the study of Akande et al., (2014), the researchers used SVM method to predict compressive concrete strength of concrete and used Coefficient of correlation (CC), root mean square error (RMSE) and absolute error (EA) to judge their model. The SVM method for predicting compressive concrete strength achieved good results which are 0.9773 (CC), 23.14 (RMSE) and 4.89 (EA). Therefore, the results proved that SVM is a good technique for prediction. In other studies from Suhad & Abbas (2015); Preetham, Shivaraj, Prema kumar, & Kumar (2014), SVM also showed good results.

In 2007, Gupta used SVM to predict compressive concrete strength with small number of data. Gupta and Fred (2014) found that SVM achieved a better performance with smaller number of training data but requires a heuristic process.

Due to some limitations of SVM, several researchers such as researchers that Uppada, Balu, Gupta, & Dutta (2014); Betrie, Sadiq, Morin, & Tesfamariam (2014); and Sakr, Elhajj, & Mitri (2011) used ANN for compressive concrete strength prediction. ANN was found to perform better than SVM for prediction.

For other techniques, Yeh & Lien (2009) applied genetic operation tree (GOT) in their study. GOT is a combination of an operation tree and a genetic algorithm to automatically produce self-organized formulas for predicting the compressive strength of high performance concrete. Comparison results indicated that GOT (R^2=0.8669) obtained formulas that were more accurate than nonlinear regression formulas but less accurate than neural network models (R^2=0.9338).

Liu et al. (2009) estimated the strength of concrete by using multiple statistics regression with the nondestructive test (NDT) surface hardness rebound value. In their study, they used 146 examples for training, and 20 examples for testing. In addition, they used 10 attributes (cement, coarse aggregate, fine aggregate, slag, flay ash, chemical admixture, water, age, moisture content and rebound value) as inputs and one attribute (compressive strength) as output. In the result of this study, the correlation coefficient achieved was 0.9622.

ANN models have been widely studied with the goal of achieving human-like performance, especially in the area of pattern recognition and system identification.

The networks are made of a number of nonlinear calculative units that manipulate in parallel and are arranged in a mode reminiscent of biological neural inter-connections (Alilou & Teshnehlab, 2010).

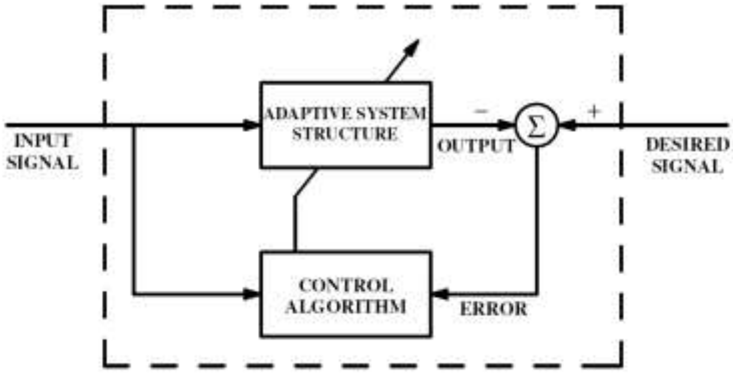

Figure 2.2: Block diagram of an adaptive system
Alilou (2010)

In Figure 2.2, Alilou and Teshnehlab used five methods of ANN for predicting concrete compressive strength. The methods are Levenberg-Marquardt, Polak-Ribiere Conjugate Gradient, Fletcher-Powell Conjugate Gradient, Gradient Descent and Quasi-Newton. All five methods achieved good accuracy and Levenberg-Marquardt obtained the best correlation coefficient (99.436) and shortest time (7.7 ms).

In the research of Rashid & Mansur (2009), they indicated that the significance of the composition materials to product high quality strength of concrete combined with the results of a previous study on finding nice quality value of compressive concrete strength. Chou, Chiu, Farfoura and Al-Taharwa (2011) used Data Mining method to predict the compressive strength of concrete with good accuracy. The compressive strength of high performance concrete was the class (target) attribute. The independent attributes (inputs) were cement, fly ash, blast furnace slag, water, super plasticizer, age, and coarse and fine aggregate. Table 2.2 shows the five different methods of data mining that they used for quantitative analysis and these are artificial neural network, support vector machines (SVM), multiple regression (MR), multiple additive regression trees (MART) and bagging regression trees (BRT). The performance comparison of this prediction model was tested by cross-validation. It showed that MART had high workability in prediction correlation coefficient, avoid to over fitting, and made short training time. The result of this study also showed that multiple additive regression trees can also be used to predict high performance concrete with different ages.

In 2011, Gilan et al. constructed a new fuzzy function model by using support vector regression to predict compressive strength of concrete and they called this model as evolutionary fuzzy function model (EFF-SVP). This model is a alteration of the fuzzy function (FF) models. For validation purpose, they examined the results based on several previous system modeling methods, artificial neural network (ANN) (Kosko, 1992), adaptive neural-fuzzy inference system (ANFIS) (Jang, 1993), fuzzy function with least squared estimation (FF-LSE) (Turksen, 2008), and enhanced FF with LSE (IFFLSE) (Celikyilmaz & Turksen, 2008). They also used eight independent attributes and one dependent attribute.

Table 2.4

RMSE for several modeling methods

Modeling Methods	RMSE			
	Train	**Validate**	**Test**	**All**
ANN-1	5.7507	6.9689	7.0205	6.1577
ANN-2	4.6006	5.6884	5.9848	5.0091
ANFIS	3.1172	14.8931	10.877	7.6109
FF-LSE	4.9397	6.4908	8.9668	5.9609
IFF-LSE	4.7435	6.8167	5.1823	5.1826
EFF-SVR	3.6922	6.3789	5.0965	4.4221

Saduf (2013)

Based on Table 2.4, it indicates that EFF-SVR was the best modeling method for predicting compressive concrete strength as the method produced the lowest value of RMSE.

Deepa et al. (2010) chose three data mining methods, Multiplayer perceptron, Linear regression and M5P model tree for predicting compressive concrete strength and compared with them. The target of this research was to find a good algorithm for prediction with shortest time. The independent attributes of this study were Cement, Blast Furnace Slag, Fly Ash, Water, Super plasticizer, Coarse aggregate, Fine aggregate and age. The result shown in Table 2.5 indicates that M5P model tree is the best algorithm for predicting compressive strength of concrete, although the taken time was not the shortest one. But it achieved the highest correlation and lowest Root

Mean Square Error (RMSE) and Mean Absolute Error (MAE).

Table 2.5

Prediction results for three algorithms

Techniques	Correlation	RMSE	MAE	Time taken In (sec)
Multilayer perceptron	0.7908	9.9054	7.678	2.06
Linear regression	0.7009	11.1066	8.8388	0.02
M5P model tree	0.8872	7.1874	5.008	0.41

Deepa (2010)

Another popular prediction method is the Bayesian network or Bayesian prediction. A Bayesian network is a graphical model that encodes probabilistic relationships among variables of interest. The model takes prior knowledge and data, and enables estimation of posterior probabilities of outcomes (Thomas, 2015). For example, Vale (2014) used Bayesian prediction method to forecast the winds of winter and MacKay (1994) did a prediction of competition based on Bayesian non-linear model.

In 2011, Pradhan & Kundu used Bayesian prediction to predict the two-parameter gamma distribution. Their result indicated that Bayesian estimates with non-informative priors behave like maximum likelihood estimates, but for informative priors the Bayesian estimates behave much better than maximum likelihood estimates. They also found that Bayesian prediction is an method for prediction based on the data that are already known.

Based on Table 2.3 and the discussion above, it is obvious that ANN is a much better technique for solving prediction problems. This is because ANN (using Back Propagation) obtained a high correlation coefficient (around 0.93-0.99) (Yeh & Lien, 2009; Muthupriya et al., 2011;Yeh, 2003;Rasa et al., 2009;Wankhade & Kambekar, 2013). Table 2.3 also shows that the average correlation coefficient (around 0.99) of back propagation is higher than the correlation coefficient average of other techniques (lower than 0.98).

Therefore, ANN was used in this study for predicting compressive strength of concrete. Section 2.2 below describes ANN in more detail.

2.2 ANN concepts and architecture

ANN consists parallel architectures that are can learn and generalize from given datasets to produce meaningful solutions even when data contain errors and are incomplete. This makes ANN a powerful tool for handling complicated engineering problems. Basically, the process of a neural network is similar to the process of neurons in the brain.

The basic strategy for developing a neural network-based model for prediction a certain data is to train a neural network on the results of a series of experiments using that dataset. If the experimental results contain the relevant information about the data behaviour, then the trained neural network will contain sufficient information about data's behaviour to qualify as a ANN model (Noorzaei, Hakim, Jaafar, & Thanoon, 2007). A trained neural network not only can reproduce the experimental results, but also it can predict the results for other similar experiments based on its powerful capability.

2.2.1 Construction of Neural Network Model and Parameters

A neural network architecture talks about how many layers in a network, how many hidden layers, how many hidden nodes in hidden layers and the relationship between each unit. The best architecture is selected from several architectures that are developed through an iteration process. How to select a most suitable ANN architecture is an open problem of investigation and depends on the area of applications. It can be determined by training, testing and validating several networks having different conditions. Connecting such units in various ways leads to different architectures of neural networks. The ANN learns from existing examples which is the process to get the final weights that are adapted. The basic unit of all ANNs is the neuron. The basic scheme of the neuron is shown in Figure 2.3. This process is represented by a learning algorithm (Oravec, Petráš, & Pilka, 2008).

The basic neuron model is shown in Figure 2.3 below:

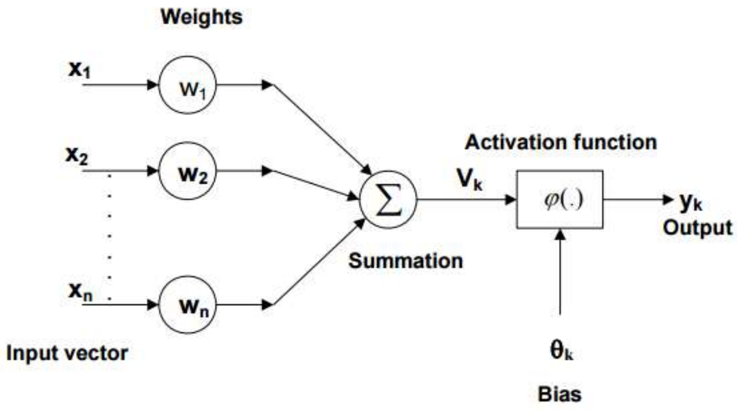

Figure 2.3: Basic neuron model

As shown in Figure 2.3, neural network models can be obtained by the number of hidden layers, number of hidden nodes in each hidden layer, type of activation function, value of learning rate and value of momentum term.

Learning rate coefficient is one of the most significant elements in network development. Every time a pattern is presented to the network, the weights leading to a neuron are modified slightly during learning in the direction required to produce a smaller error at the outputs the next time the same pattern is presented. The amount of weight modification is proportional to the learning rate. The range of leaning rate is between 0 to 1. If the value of learning rate is close to 1, it means that important modification in weight is needed, but if a value is close to 0, it presents little modification is needed (Plagianakos, Magoulas, & Vrahatis, 2001).

However, the learning rate in a parameter is the one that determines the size of the weights adjustment each time the weights are changed during training. Small values of learning rate lead to small weight changes and large values lead to large changes. The most suitable learning rate for model cannot be found directly. If the value of learning rate is 0, the network will not learn. Therefore, the learning rate is very significant in identifying over-learning and when to stop training (Noorzaei et al., 2007).

2.2.2 A simple neural network model

The simplest type of neural network feed forward network. It is a single-layer perceptron network that includes one single layer of output nodes, one layer of input nodes, and one layer of hidden layer nodes. The inputs are fed directly to the outputs via a series of weights. In this network, the information moves in only one direction, forward, from the input nodes, through the hidden nodes (if any) and to the output nodes. There are no cycles or loops in the network. Figure 2.4 shows the diagram.

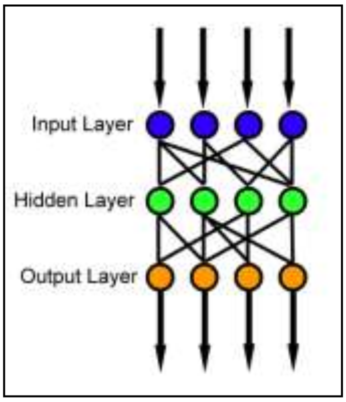

Figure 2.4: A simple neural network model

The operation of the network can be divided into two phases, learning phase, and classification phase. The Back-propagation algorithm and Feed Forward are popular techniques that have been used to perform the learning process. Back-propagation is a training algorithm that includes of 2 steps: 1) Feed forward the input values, 2) calculate the error and propagate it back to the earlier layers. Both Feed Forward and Back-propagation algorithms are used in training neural network.

In this research, Feed Forward and Back-propagation algorithms were used to develop the ANN model.

2.3 Summary

Several topics were investigated to determine the input, and techniques to be used in the study. Basically, five basic attributes, cement, fine aggregate, coarse aggregate, water and age were chosen as input. Besides these, 3 other inputs (blast furnace slag, fly ash, and super plasticizer) were selected as additional attributes. Back-propagation algorithm was also selected to be used in the model development process.

CHAPTER THREE

METHODOLOGY

This chapter elaborates the process of achieving the objectives and constructing the compressive concrete strength prediction model.

3.1 Research Process

The general goal of this study is to construct a compressive concrete strength prediction model. Thus, the process of constructing the model involves five (5) phases: Data collection (data information), Normalization, Determine parameters, Prediction model construction (construct the main architecture), and Evaluation. The flow diagram is shown in Figure 3.1:

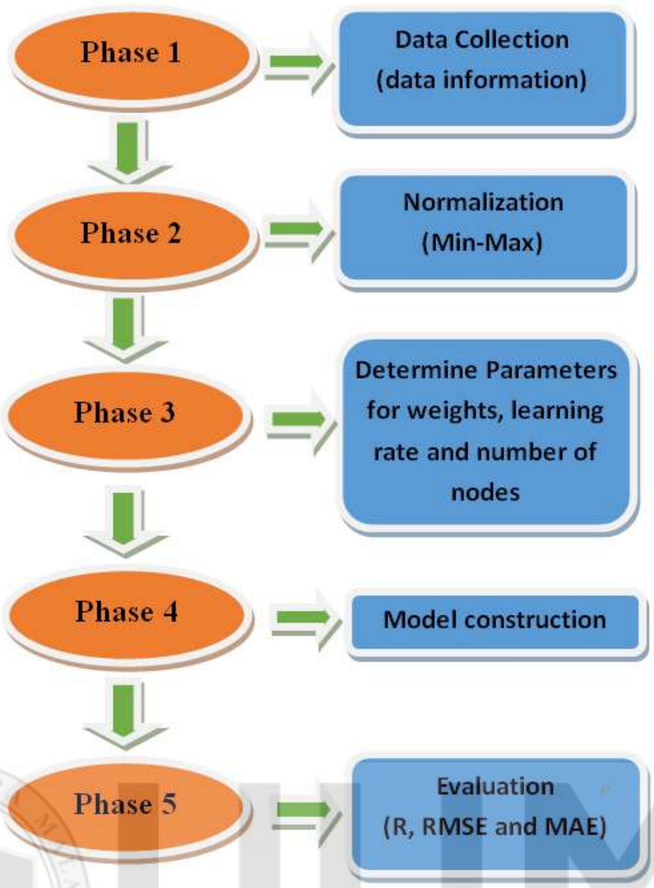

Figure 3.1: The research process diagram

3.1.1 Phase 1 Data Collection

This study used a secondary dataset (Concrete Compressive Strength Data Set) that was taken from the UCI repository. The dataset was separated to 8 sub datasets (different combinations of attributes). Detailed information on the datasets is shown in Table 3. The sample of data are shown in Appendix A.

Table 3.1

Inputs and output attributes

No. of Datasets	**No. of input attributes**	**Instances of Data**	**Output attribute (units)**
1	5	209	
2	6	209	
3	6	209	
4	6	209	Compressive Strength of Concrete
5	7	209	
6	7	209	
7	7	209	
8	8	209	

Based on Table 3.1, Model 1 focused on 5 basic attributes (cement, water, age, fine aggregate and coarse aggregate), and the other models focused on different combinations of attributes (5 basic attributes + additional attributes). All 8 models have the same target which is compressive strength of concrete.

The statistical descriptions of datasets for each model is specified in Table 3.2.

Table 3.2 (a)

Statistical descriptions for Dataset 1

Statistic / Attributes	Minimum	Maximum	Mean	StdDev
Cement	200	540	354.187	85.623
Water	146	228	192.114	12.183
Coarse Aggregate	838.4	1125	1018.21	72.394
Fine Aggregate	594	945	773.097	81.492
Age	1	365	61.995	90.721
Concrete Compressive Strength	6.27	74.99	29.806	14.645

Table 3.2 (b)

Statistical descriptions for Dataset 2

Statistic / Attributes	Minimum	Maximum	Mean	StdDev
Cement	165	540	349.823	89.144
Fly Ash	0	143.6	3.0350	20.115
Water	146	228	191.493	12.524
Coarse Aggregate	838.4	1125	1018.059	71.543
Fine Aggregate	594	945	776.54	82.784
Age	1	365	59.871	87.88
Concrete Compressive	6.27	74.99	29.592	14.551

Strength				

Table 3.2 (c)

Statistical descriptions for Dataset 3

Statistic / **Attributes**	**Minimum**	**Maximum**	**Mean**	**StdDev**
Cement	102	540	288.601	113.239
Blast Furnace Slag	0	359.400	81.596	98.380
Water	146	228	194.971	14.116
Coarse Aggregate	879	1125	997.672	70.917
Fine Aggregate	594	945	760.491	91.398
Age	1	365	55.024	84.969
Concrete Compressive Strength	3.32	74.99	28.438	14.858

Table 3.2 (d)

Statistical descriptions for Dataset 4

Statistic / **Attributes**	**Minimum**	**Maximum**	**Mean**	**StdDev**
Cement	200	540	368.285	92.462
Water	140	228	189.562	15.813
Super plasticizer	0	28.2	1.171	4.621

Coarse Aggregate	801	1125	1006.465	78.131
Fine Aggregate	594	945	773.194	82.984
Age	1	365	56.254	85.553
Concrete Compressive Strength	6.27	79.99	32.157	15.986

Table 3.2 (e)

Statistical descriptions for Dataset 5

Statistic / **Attributes**	**Minimum**	**Maximum**	**Mean**	**StdDev**
Cement	102	540	277.976	104.788
Blast Furnace Slag	0	359.400	93.500	99.345
Fly Ash	0	143.6	3.035	20.115
Water	146	228	196.282	15.710
Coarse Aggregate	838.4	1145	988.485	67.078
Fine Aggregate	594	945	755.125	90.204
Age	1	365	64.512	96.999
Concrete Compressive Strength	2.330	74.99	28.551	14.809

Table 3.2 (f)

Statistical descriptions for Dataset 6

Statistic / Attributes	Minimum	Maximum	Mean	StdDev
Cement	144	540	308.598	103.055
Fly Ash	0	194.9	59.576	62.872
Water	141.8	228	182.451	17.650
Super plasticizer	0	28.2	4.921	5.670
Coarse Aggregate	801.1	1125	1002.633	68.539
Fine Aggregate	594	945	792.537	74.501
Age	1	365	50.311	70.490
Concrete Compressive Strength	6.27	79.99	31.779	13.693

Table 3.2 (g)

Statistical descriptions for Dataset 7

Statistic / Attributes	Minimum	Maximum	Mean	StdDev
Cement	102	540	309.589	110.917
Blast Furnace Slag	0	359.400	91.553	96.363
Water	127.3	228	189.148	20.771
Super plasticizer	0	32.200	2.925	5.782

Coarse Aggregate	801	1134.3	978.956	73.776
Fine Aggregate	594	945	757.001	87.249
Age	1	365	62.459	89.335
Concrete Compressive Strength	4.57	82.600	35.922	18.436

Table 3.2 (h)

Statistical descriptions for Dataset 8

Statistic / Attributes	**Minimum**	**Maximum**	**Mean**	**StdDev**
Cement	116	540	272.344	93.748
Blast Furnace Slag	0	305.300	86.481	92.614
Fly Ash	0	193	50.236	62.300
Water	121.800	228	184.844	24.122
Super plasticizer	0	32.200	5.869	5.728
Coarse Aggregate	822	1134.3	978.376	70.028
Fine Aggregate	594	945	762.811	85.395
Age	1	365	59.847	85.491
Concrete Compressive Strength	4.83	82.600	36.695	17.652

This dataset does not contain any missing values (using WEKA). In addition, concrete compressive strength is a highly nonlinear function of building materials (Chou et al., 2011).

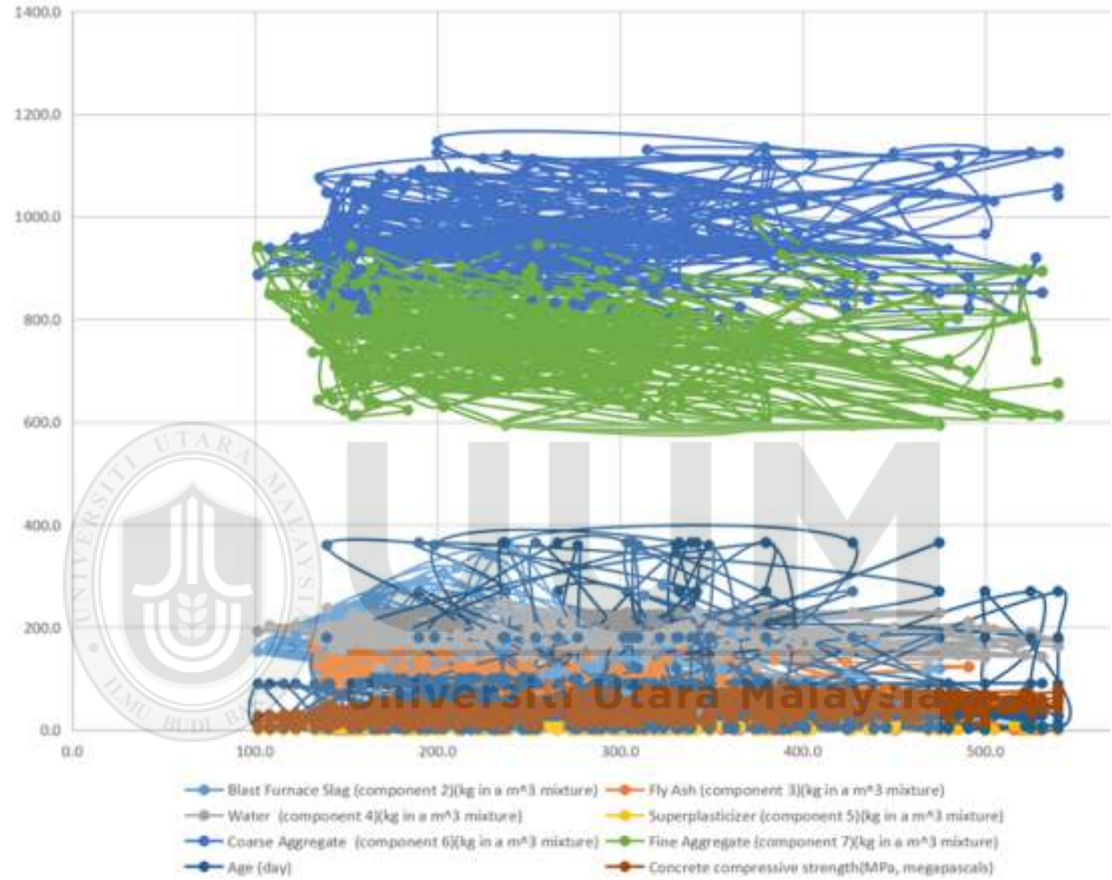

Figure 3.2: The scatter plot of dataset.

Figure 3.2 shows the scatter plot of the dataset. Based on the figure, it can be seen that it is nonlinear and cannot be solved using a linear solving method such as regression.

3.1.2 Phase 2 Normalization

This study used ANN method to do data training, so initially the dataset need to be normalized. Because that it is good for comparison between the results for the various sensory outputs, and it also can enhance the reliability of the trained network (Jayalakshmi & Santhakumaran, 2011). Min-Max normalization (Nayak, Misra, & Behera, 2014) was used. Normalization was done using WEKA 3.6, and the formula is shown below:

$$V' = \frac{V - minA}{maxA - minA}(new\_maxA - new\_minA) + new\_minA \qquad (3\text{-}1)$$

Where, V' is a new value

V is the original value

minA is the minimum value of the attributes

maxA is the maximum value of the attributes

new_ maxA is a maximum value of the new value

new_ minA is a minimum value of the new value

The data was also set to two decimal places, and the samples of raw data and normalized data are shown in Table 3.3.

Table 3.3

Sample of raw data (before) and normalized data (after)

Names:	**Cement**	**BFS**	**Fly Ash**	**Water**	**SP**	**CA**	**FA**	**Age**	**CS**
	213.70	98.10	24.50	181.70	6.90	1065.80	785.40	3	18.00

Raw Data	213.70	98.10	24.50	181.70	6.90	1065.80	785.40	14	30.39
	213.70	98.10	24.50	181.70	6.90	1065.80	785.40	28	45.71
	213.70	98.10	24.50	181.70	6.90	1065.80	785.40	56	50.77
	213.70	98.10	24.50	181.70	6.9	1065.80	785.40	100	53.90
Norma--lized Data	0.26	0.27	0.12	0.48	0.21	0.77	0.48	0.01	0.20
	0.26	0.27	0.12	0.48	0.21	0.77	0.48	0.04	0.35
	0.26	0.27	0.12	0.48	0.21	0.77	0.48	0.07	0.54
	0.26	0.27	0.12	0.48	0.21	0.77	0.48	0.15	0.60
	0.26	0.27	0.12	0.48	0.21	0.77	0.48	0.27	0.64

3.1.3 Phase 3 Determine Parameters

In this study, the value for four parameters (Weights, Learning Rate, Momentum factor and numbers of hidden nodes) were determined. The process is shown below (Figure 3.3):

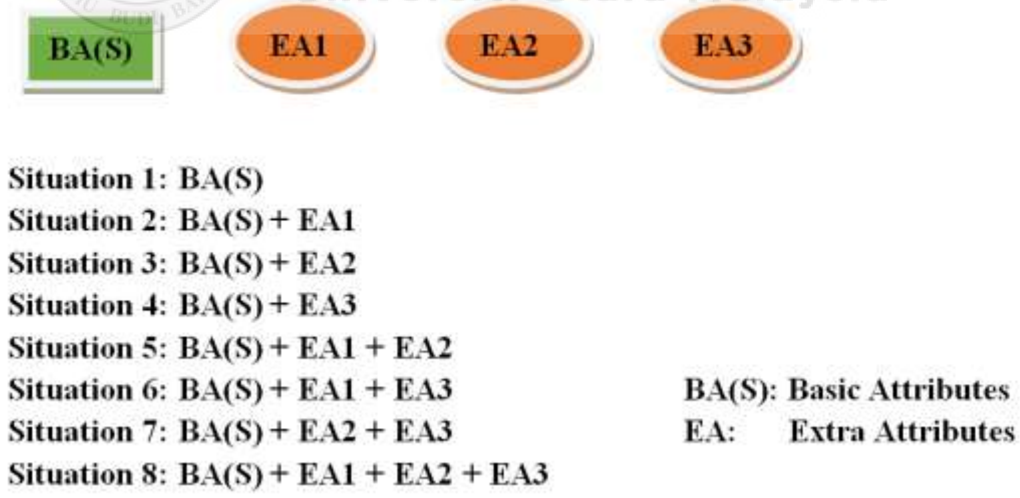

Figure 3.3: Eight situations of different combinations of attributes

In Figure 3.3, because that there are five basic attributes (BAs) and three extra attributes (EA1, EA2 and EA3), eight training models were considered. The five basic attributes creates the basic training model, while the three additional attributes were added to construct other training models.

In total, there 8 prediction models were constructed. The following sub sections explains the process of determining the parameters.

3.1.3.1 Determine Weights

Eight sets of weights were determined based on different combinations of attributes. Back-Propagation algorithm (Makin, 2006) were used in data training, specifically for updating the weights. The formulas (for one hidden layer) are shown below:

Feed-Forward:

$$z\_in_j = v_{0j} + \sum_{i=1}^{n} x_i v_{ij} \tag{3-2}$$

$$z_j = f(z\_in_j) = 1/(1 + \exp(-z\_in_j)) \tag{3-3}$$

Each hidden unit $(z_j, j = 1,2 \ldots . p)$ sums its weighted input signals, applies its activation function to compute its output signal, and sends this signal to all units in the output layer.

$$y\_in_k = w_{0k} + \sum_{j=1}^{p} z_j w_{jk} \tag{3-4}$$

$$y_k = f(y\_in_k) = 1/(1 + \exp(-y\_in_k)) \tag{3-5}$$

Each output unit $(y_k, k = 1,2 \ldots m)$ sums its weighted input signals, and applies its activation function to compute its output signal.

Back Propagation of Error:

$$\sigma_k = (t_k - y_k)f(x)[1 - f(x)] \tag{3-6}$$

$$\Delta w_{jk} = \alpha\delta_k z_j \tag{3-7}$$

$$\Delta w_{0k} = \alpha\delta_k \tag{3-8}$$

Each of output units $(y_k, k = 1,2 \ldots m)$ gets a target pattern corresponding to the input training pattern, calculates its error information term, computes its weight correction term (it will use for updating w_{jk}), computes its bias correction term (it will use for updating w_{0k}) and transfers σ_k to units in the layer below.

$$\sigma\_in_j = \sum_{k=1}^{m} \sigma_k w_{jk} \tag{3-9}$$

$$\sigma_j = \sigma_{in_j} f\left(z_{in_j}\right)[1 - f\left(z_{in_j}\right)] \tag{3-10}$$

$$\Delta v_{ij} = \alpha\delta_j x_i \tag{3-11}$$

$$\Delta v_{0j} = \alpha\delta_j \tag{3-12}$$

Each hidden unit $(z_j = 1,2, \ldots p)$ adds delta inputs (from units in the layer above) multiplies by the derivative of its activation function to calculate its error information term, calculates its weight correction term (it will use for updating v_{ij}), and computes its bias correction term (it will use for updating v_{0j}).

Update Weights and Biases:

Each output unit $(y_k, k = 1,2, \ldots m)$ updates its bias and weights (j=0,...p):

$$w_{jk}(\text{new}) = w_{jk}(\text{old}) + \Delta w_{jk} \tag{3-13}$$

Each hidden unit $(z_j, j = 1,2, \ldots p)$ updates its bias and weights (i = 0,...n):

$$v_{ij}(new) = v_{ij}(old) + \Delta v_{ij} \quad (3\text{-}14)$$

3.1.3.2 Other Parameters

In this part, several parameters were determined. The value of learning rate that 0.01, 0.1, 0.3, 0.5 and 0.9 (Wankhade & Kambekar, 2013) were tested. The momentum factors which are 0.0, 0.25, 0.5 and 0.75 (Yeh, 2006;Wankhade & Kambekar, 2013)also were tested in this study. The suitable learning rate and momentum which made the prediction model achieve best results were used to construct the prediction model for compressive concrete strength. Based on the study of Panchal, Ganatra, Kosta, & Panchal (2011), 1 hidden layer is sufficient for nearly all problems, and 2 hidden layers are required for modeling data with discontinuities like a saw tooth wave pattern. As the result, all models of this study used one hidden layer but different hidden nodes. The testing parameters of each models were mentioned in Table 3.4.

Table 3.4

Testing parameters

No. of attributes	**5**	**6**			**7**			**8**
Attributes in each model	**M 1**	**M 2**	**M 3**	**M 4**	**M 5**	**M 6**	**M 7**	**M 8**
	BVs	**BVs+ EV1**	**BVs+ EV2**	**BVs+EV3**	**BVs+EV1 +EV2**	**BVs+EV1 +EV3**	**BVs+ EV2+ EV3**	**All**
Hidden layers	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**
No. of hidden nodes	**2~6**	**2~6**	**2~6**	**2~6**	**2~6**	**2~6**	**2~6**	**2~6**
Learning rate	**0.01, 0.1, 0.3, 0.5 and 0.9.**							
Momentum	**0.0, 0.25, 0.5 and 0.75**							

Table 3.4 shows the summary of values used to obtain the best parameters for learning rate, momentum, number of hidden layer and number of hidden nodes. For the hidden nodes, there is no formula or algorithm to figure out how many hidden nodes should be in a hidden layer. However, according to Doug (2016), the number of neurons in hidden layer is the mean of the neurons in the input and output layers.

So in this study, the mean value of model 1 is 3 (i.e number of inputs plus the number of outputs, then divide by 2); the mean value of model 2 to model 4 is 3.5; the mean value of model 5 to model 7 is 4; and the mean value of model 8 is 4.5. Based on the mean value of each model, this study tested (2~6) number of hidden nodes, which is approximate to the mean of inputs and output. In addition, values of parameters were measured by correlation coefficient (the higher, the better), mean absolute error (the lower, the better) and root mean square error (the lower, the better).

3.1.4 Phase 4 Construct Prediction Model

There are five basic independent attributes and three extra independent attributes in this study. In total, there are 8 independent attributes (inputs) and one dependent attribute (output). So considering all situations, it should have 8 prediction models (Table 3.5), each with different number of independent attributes. The general prediction model is shown in Figure 3.4. When users choose different number of independent attributes, the model will change. In other words, the parameters, "r", "weights"," and hidden nodes" will be changed. The best model chosen is the model that has the highest correlation coefficient, lowest Mean Absolute Error, and lowest Root Mean Square Error. The models are presented in Chapter 5.

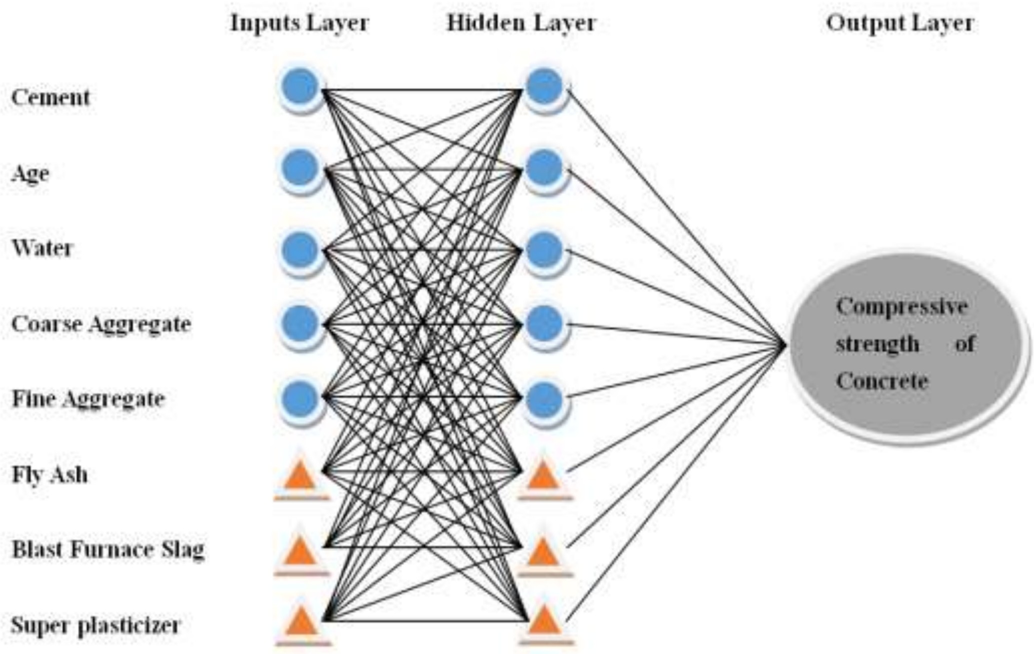

Figure 3.3: The general model for compressive strength of concrete

Table 3.5

Models and Attributes

No. of Models	Attributes
Model 1	5 basic attributes (cement, water, fine aggregate, coarse aggregate and age)
Model 2	5 Basic attributes + Fly Ash (FA)
Model 3	5 Basic attributes + Blast Furnace Slag (BFS)
Model 4	5 Basic attributes + Super Plasticizer (SP)
Model 5	5 Basic attributes + FA + BFS
Model 6	5 Basic attributes + FA + SP
Model 7	5 Basic attributes + BFS + SP
Model 8	5 Basic attributes + FA + BFS + SP

3.1.5 Phase 5 Evaluation

In this study, the percentage split method was used to evaluate all models. Specifically, 90% of the data was used for training and 10% was used for testing. The performance was measured based on correlation coefficient, root mean square error (RMSE) and mean absolute error (MAE) (Wankhade & Kambekar, 2013).

3.1.5.1 Correlation coefficient

Correlation coefficient tests the level of linear relation among the goal and the predicted result. It is a method to identify how far the tendency in predicted values follows those in real observed values. The value of R is numeric in the range of 0-1.A high value of correlation coefficient shows that the model is good. The correlation coefficient (R) formula used is:

$$R = \frac{\sum_{i=1}^{n}(x_i)(y_i)}{\sqrt{\sum_{i=1}^{n}(x_i^2)\sum_{i=1}^{n}(y_i^2)}} \tag{3-16}$$

Where, $x_i = X_i - \overline{X}$, $y_i = Y_i - \overline{Y}$

$X_i = i^{th}$ observed value, $\overline{X} = \text{mean of } X$,

$Y_i = i^{th}$ predicted value, $\overline{Y} = \text{mean of } Y$,

$n = \text{number of observation of } X_i \text{ and } Y_i$

If correlation coefficient is equal to 1, it shows that the model is perfect. Values of correlation coefficient in the range of 0.9 to 0.99 show that the model performs well (good correlation coefficient). However, if the value of correlation coefficient is between 0.8 and 0.89, the model is said to be satisfactory and can still be accepted. Any values of correlation coefficient that is less than 0.8 shows the model is not good (Wankhade & Kambekar, 2013). In this study, the value required is more than 0.9. If the value of correlation coefficient cannot achieve 0.9, the prediction model was reconstructed.

3.1.5.2 Root mean square error (RMSE)

The root mean square error is suitable to iterative algorithms and is quite a good method for higher values. This is the formula for calculating RMSE as below:

$$\mathrm{RMSE} = \sqrt{\frac{\sum_{i=1}^{n}(X_i - Y_i)^2}{n}} \qquad (3\text{-}17)$$

(Source from: https://www.kaggle.com/wiki/RootMeanSquaredError)

It supports a general representation of the errors involved in the prediction. If the value of result is lower, it means that the result is better.

3.1.5.3 Mean absolute error (MAE)

The mean absolute error (MAE) is to measure how close forecasts or predictions are to the outcomes. The formula is:

$$\mathrm{MAE} = \frac{\sum_{i=1}^{n}|X_i - Y_i|}{n} \qquad (3\text{-}18)$$

(Source from: https://www.kaggle.com/wiki/MeanAbsoluteError)

A lower MAE shows that the prediction is better.

3.2 Summary

This chapter describes the methodology of this study. Five phases were involved: Data collection (8 groups of data), Normalization, Determine parameters, Prediction model construction (it includes all the 8 prediction models), and Evaluation (correlation coefficient, mean absolute error and root mean square error were mentioned).

CHAPTER FOUR

RESULTS

This chapter presents deliverables for objective 1, objective 2, objective 3 and objective 4.

4.1 Deliverables for objective 1

As stated in Chapter One, the first objective of this study is to identify the basic attributes for prediction.

The basic attributes have been obtained through examining past researches and have been presented in Table 2.1 (Chapter Two). In summary, the basic attributes are cement fine aggregate, coarse aggregate, water and age. Table 4.1 below shows the descriptions for each attribute:

Table 4.1

Description of basic attributes

Basic attributes	**Description**
Cement	A substance used in construction that sets and hardens and can bind other materials together.
Water	A colourless liquid that is used to mix with cement in making concrete.
Fine aggregate	Consist of natural sand or crushed stone with most particles passing through a 3/8-inch sieve.
Coarse aggregate	Particles that are greater than 0.19 inch, but generally range between 3/8 and 1.5 inches in diameter.

Age	The days for making concrete become harden.

4.2 Deliverables for objective 2

The second objective of this study is to identify additional attributes that can be used to predict compressive concrete strength.

The three additional attributes are fly ash, blast furnace slag and super plasticizer. Table 4.2 below shows the descriptions for each attribute:

Table 4.2

Description of additional attributes

Basic attributes	Description
Fly Ash	Finely divided residue that results from the combustion of pulverized coal and is transported from the combustion chamber by exhaust gases.
Blast Furnace Slag	Consists primarily of silicates, alumina silicates, and calcium-alumina-silicates.
Super Plasticizer	Chemical admixtures used where well-dispersed particle suspension is required. These polymers are used as dispersants to avoid particle segregation (gravel, coarse and fine sands), and to improve the flow characteristics (rheology) of suspensions such as in concrete applications.

4.3 Deliverables for Objective 3

As mentioned in Chapter 1, the third objective is to determine the parameters for

Weights, Learning Rate, Momentum, number of hidden nodes in hidden layer for 8 prediction models. The models were:

Model 1 (5 basic attributes (cement, water, fine aggregate, coarse aggregate and age)**)**

Model 2 (5 Basic attributes + Fly Ash (FA)**)**

Model 3 (5 Basic attributes + Blast Furnace Slag (BFS)**)**

Model 4 (5 Basic attributes + Super Plasticizer (SP)**)**

Model 5 (5 Basic attributes + FA + BFS**)**

Model 6 (5 Basic attributes + FA + SP**)**

Model 7 (5 Basic attributes + BFS + SP**)**

Model 8 (5 Basic attributes + FA + BFS + SP**)**

The parameters were obtained through several experimentation. The following sections explain the results for each model.

4.3.1 Model 1 parameters:

In this section, five learning rates that 0.01, 0.1, 0.3, 0.5 and 0.9 (Wankhade & Kambekar, 2013) were tested. The momentum values 0.0, 0.25, 0.5 and 0.75 (Yeh, 2006;Wankhade & Kambekar, 2013) also were tested.

For the number of hidden layers, according to the study of Panchal, Ganatra, Kosta, & Panchal (2011), one hidden layer is sufficient for nearly all problems, and two hidden layers are required for modeling data with discontinuities like a saw tooth wave pattern. Therefore, all models of this study used one hidden layer but different hidden nodes.

For the number of hidden nodes, too few hidden nodes in hidden layer will lead to the problem called under fitting, but too many hidden nodes will lead to over fitting and it will take longer training time (Panchal et al., 2011). Based on that, 2, 3, 4, 5 and 6 hidden nodes were used for training and testing.

Thus, considering all the parameters (learning rate, momentum and number of hidden nodes) in this study, a total of 100 different combinations of parameters (each set of parameters must includes learning rate, momentum and number of hidden nodes) for model 1 were tested (Figure 4.1).

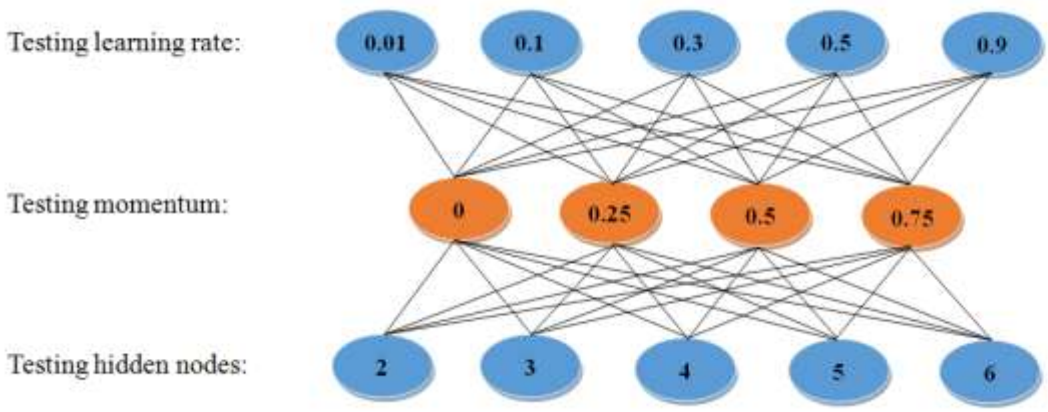

Figure 4.1: The combinations of parameters

The model performance was measured using R, MAE, RMSE and Time taken. The experiments used Percentage split of 90% for training and 10% for testing. The best values (i.e the highest value of R, lowest value of MAE and lowest value of RMSE) were chose as the final parameters for model 1.

The detail of parameters tested (100 combinations of parameters) and results obtained are shown as follows:

Table 4.3

Experiments using 5 attributes

Test 1: Hidden nodes= 2 and momentum =0

Learning Rate	**0.01**	**0.1**	**0.3**	**0.5**	**0.9**
R	0.8804	0.9134	0.7869	0.7809	0.8019
MAE	4.0947	3.9977	5.7728	5.9514	4.6447
RMSE	6.4014	5.8833	8.1277	8.2254	7.975
TT(s)	0.47	0.42	0.45	0.42	0.41

Test 2: Hidden nodes= 2 and momentum =0.25

Learning Rate	**0.01**	**0.1**	**0.3**	**0.5**	**0.9**
R	0.8893	0.9125	0.7829	0.7660	0.8428
MAE	3.9685	4.1009	5.9431	6.2876	6.1876
RMSE	6.1943	5.9870	8.1871	8.5692	7.8970
TT(s)	0.48	0.44	0.43	0.45	0.44

Test 3: Hidden nodes= 2 and momentum =0.5

Learning Rate	**0.01**	**0.1**	**0.3**	**0.5**	**0.9**
R	0.9002	0.9103	0.7724	0.7457	0.7694
MAE	3.8302	4.4744	6.2828	7.7918	18.8094
RMSE	5.9326	6.3422	8.4906	10.5650	20.8214
TT(s)	0.43	0.43	0.44	0.43	0.44

Test 4: Hidden nodes= 2 and momentum =0.75

Learning Rate	**0.01**	**0.1**	**0.3**	**0.5**	**0.9**
R	0.9099	0.7846	0.8777	0.8778	0.8904
MAE	3.7194	6.3157	4.6237	11.0227	10.6924
RMSE	5.6839	8.7157	6.3087	12.5657	12.2117
TT(s)	0.44	0.43	0.43	0.43	0.39

Test 5: Hidden nodes= 3 and momentum =0

Learning Rate	**0.01**	**0.1**	**0.3**	**0.5**	**0.9**
R	0.9336	0.9590	0.9619	0.8181	0.8331
MAE	3.1908	2.5046	3.1601	5.6332	4.9197
RMSE	4.7421	3.8438	4.2815	7.6559	7.3802
TT(s)	0.6	0.6	0.59	0.59	0.6

Test 6: Hidden nodes= 3 and momentum =0.25

Learning Rate	**0.01**	**0.1**	**0.3**	**0.5**	**0.9**
R	0.9413	0.9594	0.9629	0.8073	0.8756
MAE	2.9353	2.5231	2.8156	5.8294	7.0976
RMSE	4.4682	3.8311	3.8196	7.7954	8.2749
TT(s)	0.6	0.61	0.61	0.59	0.6

Test 7: Hidden nodes= 3 and momentum =0.5

Learning Rate	**0.01**	**0.1**	**0.3**	**0.5**	**0.9**
R	0.9485	0.9586	0.9626	0.7864	0.8791
MAE	2.7192	2.6414	2.8974	6.2488	11.8380
RMSE	4.1985	3.9793	3.9863	8.5837	12.8561
TT(s)	0.6	0.61	0.6	0.59	0.6

Test 8: Hidden nodes= 3 and momentum =0.75

Learning Rate	**0.01**	**0.1**	**0.3**	**0.5**	**0.9**
R	0.9510	0.9535	0.8718	0.8742	0.3354
MAE	2.7822	3.3341	8.1529	8.8121	9.2185
RMSE	4.0752	4.6930	9.7265	10.9323	13.2009
TT(s)	0.6	0.59	0.6	0.6	0.6

Test 9: Hidden nodes= 4 and momentum =0

Learning Rate	**0.01**	**0.1**	**0.3**	**0.5**	**0.9**
R	0.9129	0.9427	0.9460	0.9337	0.9290
MAE	3.6862	2.9915	4.7634	5.3747	8.9899
RMSE	5.4981	4.7114	6.3335	6.9037	10.2559
TT(s)	0.75	0.78	0.76	0.76	0.74

Test 10: Hidden nodes= 4 and momentum =0.25

Learning Rate	**0.01**	**0.1**	**0.3**	**0.5**	**0.9**
R	0.9056	0.9438	0.9658	0.9334	0.9330
MAE	3.7392	2.8825	2.5220	5.7189	10.9118
RMSE	5.8302	4.5740	3.6008	7.2815	12.1103
TT(s)	0.74	0.75	0.75	0.74	0.76

Test 11: Hidden nodes= 4 and momentum =0.5

Learning Rate	**0.01**	**0.1**	**0.3**	**0.5**	**0.9**
R	0.9433	0.9463	0.9612	0.9386	0.9445
MAE	2.8576	2.7670	3.1551	7.3113	14.8149
RMSE	4.3924	4.4281	4.3184	8.7113	15.8161
TT(s)	0.81	0.83	0.76	0.76	0.78

Test 12: Hidden nodes= 4 and momentum =0.75

Learning Rate	**0.01**	**0.1**	**0.3**	**0.5**	**0.9**
R	0.9278	0.9196	0.9021	0.1812	-0.2787
MAE	3.3279	3.8040	9.5353	25.9760	9.2175
RMSE	5.0691	5.9779	11.165	27.3232	13.2046
TT(s)	0.76	0.74	0.75	0.74	0.73

Test 13: Hidden nodes= 5 and momentum =0

Learning Rate	**0.01**	**0.1**	**0.3**	**0.5**	**0.9**
R	0.9221	0.9423	0.9463	0.9396	0.9059
MAE	3.5480	3.9592	3.6870	4.2475	8.5610
RMSE	5.1594	5.4206	4.7616	5.2401	10.1191
TT(s)	0.92	0.91	0.93	0.9	84.1585

Test 14: Hidden nodes= 5 and momentum =0.25

Learning Rate	**0.01**	**0.1**	**0.3**	**0.5**	**0.9**
R	0.9268	0.9444	0.9452	0.9355	0.9323
MAE	3.4002	3.8172	3.5581	4.1521	8.1561
RMSE	5.0156	5.1465	4.6506	5.1889	9.4475
TT(s)	0.9	0.92	0.9	0.93	0.91

Test 15: Hidden nodes= 5 and momentum =0.5

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.8815	0.9103	0.9429	0.9341	0.9186
MAE	3.8284	4.8413	3.8406	5.8334	13.0642
RMSE	6.5487	9.8674	4.9255	7.4281	14.1241
TT(s)	0.9	0.93	0.9	0.92	0.9

Test 16: Hidden nodes= 5 and momentum =0.75

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.5962	0.9658	0.9295	0.8157	-0.3745
MAE	4.9869	2.4902	5.6578	5.9071	9.2174
RMSE	10.6039	3.5597	6.6116	10.3654	13.2045
TT(s)	0.92	0.9	0.92	0.91	0.84

Test 17: Hidden nodes= 6 and momentum =0

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9303	0.9429	0.9479	0.9397	0.8424
MAE	3.2577	3.9271	4.0513	5.0696	16.6086
RMSE	4.8640	5.4865	5.1605	6.6148	18.3994
TT(s)	1.07	1.06	1.08	1.1	1.09

Test 18: Hidden nodes= 6 and momentum =0.25

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9364	0.9500	0.9444	0.9366	0.6845
MAE	2.9223	3.4084	4.2300	5.3145	10.5619
RMSE	4.6896	4.5527	5.4057	6.8506	12.5164
TT(s)	1.11	1.08	1.09	1.07	1.07

Test 19: Hidden nodes= 6 and momentum =0.5

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9305	0.9259	0.9256	0.8430	0.5504
MAE	3.0043	3.7053	4.8049	6.8165	18.3035
RMSE	5.1360	5.5568	6.6059	8.6411	19.3555
TT(s)	1.07	1.09	1.08	1.08	1.08

Test 20: Hidden nodes= 6 and momentum =0.75

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9358	0.9112	0.8910	0.7303	-0.2282
MAE	3.6943	5.4957	8.6233	31.3474	9.2175
RMSE	5.9498	13.0798	10.1576	32.7483	13.2046
TT(s)	1.11	1.09	1.09	1.06	1.12

Table 4.3 shows all results using different combination of parameters for model 1. The best results were achieved from Test 16 which is learning rate = 0.1, momentum = 0.75 and the number of hidden nodes is 5. The results are: R achieved the highest value i.e0.9658, MAE achieved the lowest value i.e 2.4902 and RMSE also achieved the lowest value i.e 3.5597.

The weights and thresholds parameters for model 1 is shown in Appendix J.

4.3.2 Model 2 to Model 8 parameters

In order to determine the best parameters to be used for models 2 to 8, several experiments were conducted using different learning rate, momentum, and hidden nodes. The weights and threshold parameters for Model 2 to Model 8 are shown in Appendix J. Table 4.4 shows the values used for determining suitable parameters for model 2 to model 8.

Table 4.4

Values used for determining suitable parameter for Models 2 to 8

Models	Learning rate	Momentum	Hidden layer	Hidden nodes	Percentage Split
Model 2	0.01, 0.1, 0.3, 0.5 &0.9	0, 0.25, 0.5 & 0.75	1	2,3,4,5&6	90%
Model 3	0.01, 0.1, 0.3, 0.5 &0.9	0, 0.25, 0.5 & 0.75	1	2,3,4,5&6	90%
Model 4	0.01, 0.1, 0.3, 0.5 &0.9	0, 0.25, 0.5 & 0.75	1	2,3,4,5&6	90%
Model 5	0.01, 0.1, 0.3, 0.5 &0.9	0, 0.25, 0.5 & 0.75	1	2,3,4,5&6	90%
Model 6	0.01, 0.1, 0.3, 0.5 &0.9	0, 0.25, 0.5 & 0.75	1	2,3,4,5&6	90%
Model 7	0.01, 0.1, 0.3, 0.5 &0.9	0, 0.25, 0.5 & 0.75	1	2,3,4,5&6	90%
Model 8	0.01, 0.1, 0.3, 0.5 &0.9	0, 0.25, 0.5 & 0.75	1	2,3,4,5&6	90%

Based on the experiments, the suitable parameters for models 2 to 8 are presented in Table 4.5 below. And for the parameters testing for model 2 to model 8, each of mode also has 100 combinations of attributes need to test like Figure 4.1 shown. The experimental results (parameters testing) for Model 2 to Model 8 are shown in Appendix B to H.

Table 4.5

Best parameters for Models 2 to 8

Models	**Learning rate**	**Momentum**	**Hidden layer**	**Hidden nodes**
Model 2 (6 attributes)	0.1	0.75	1	5
Model 3 (6 attributes)	0.5	0.25	1	5
Model 4 (6 attributes)	0.01	0.75	1	5
Model 5 (7 attributes)	0.01	0.25	1	3
Model 6 (7 attributes)	0.01	0.5	1	4
Model 7 (7 attributes)	0.1	0.75	1	5
Model 8 (8 attributes)	0.1	0	1	6

4.4 Deliverables for objective 4

The purpose of the fourth objective is to determine an ANN architecture.

Figure 4.2 shows the main architecture. Based on the main architecture, 8 different architectures were obtained from using the combination of 5 basic attributes (cement, water, fine aggregate, coarse aggregate and age) with different additional attributes (fly ash, blast furnace slag or super plasticizer).

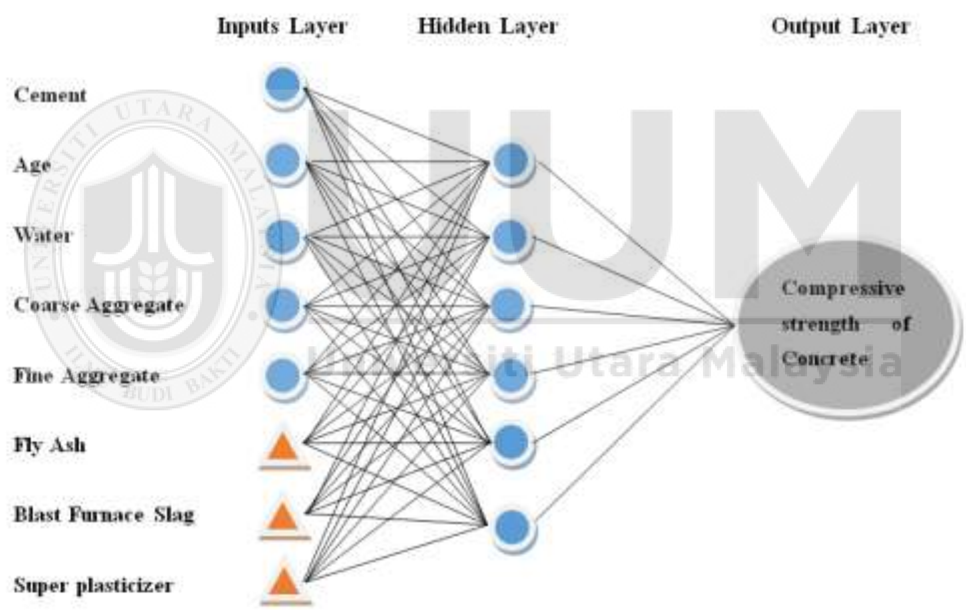

Figure 4.2: The main architecture for the study

From Figure 4.2, there are 8 attributes as inputs in total (5 are basic attributes and 3 are additional attributes) for compressive concrete strength prediction. There are 6 hidden nodes because the maximum number of hidden nodes is 6. It has 1 output which is the compressive strength of concrete.

Based on Table 4.3 and Table 4.5, the ANN architectures of all the 8 prediction models are shown as below (Figure 4.3 to Figure 4.10).

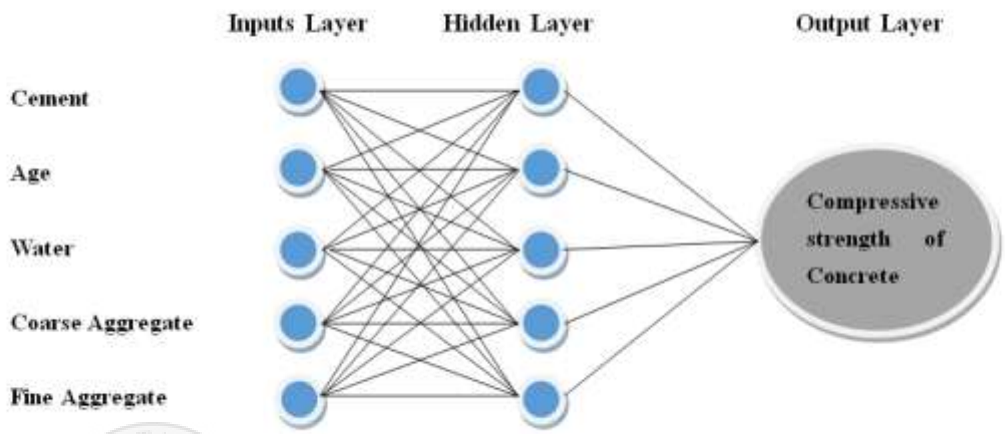

Figure 4.3: The ANN architecture for Model 1

In Figure 4.3, the parameters of model 1 are learning rate equal to 0.1 and momentum equal to 0.75.

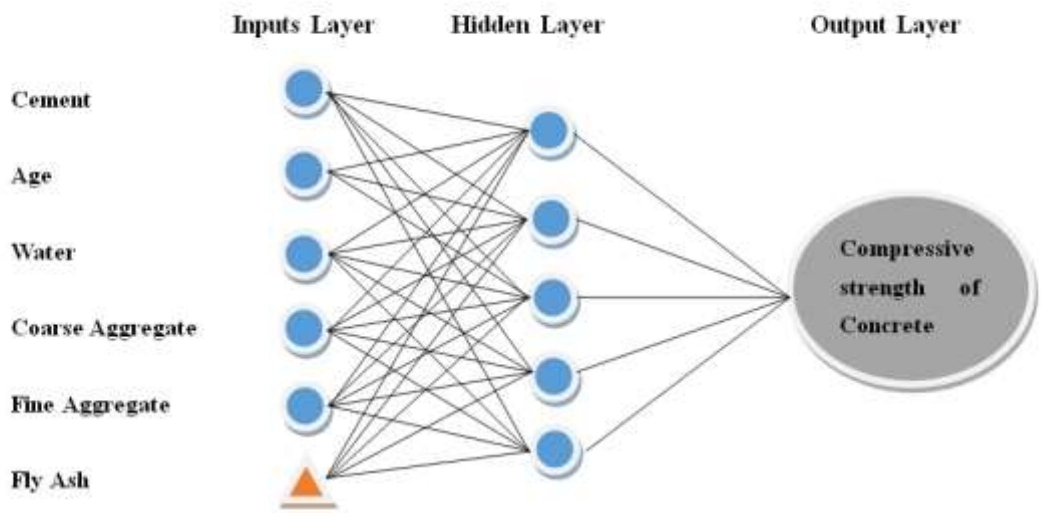

Figure 4.4: The ANN architecture for Model 2

In Figure 4.4, the parameters of model 2 are learning rate equal to 0.5 and momentum equal to 0.25.

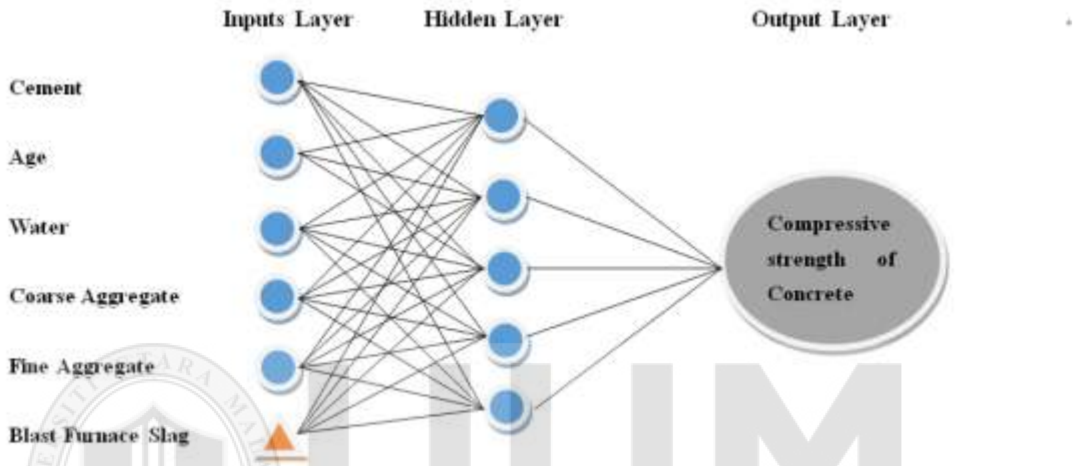

Figure 4.5: The ANN architecture of Model 3

Figure 4.5 is the architecture of model 3, and the parameters that learning rate equal to 0.01 and momentum equal to 0.75.

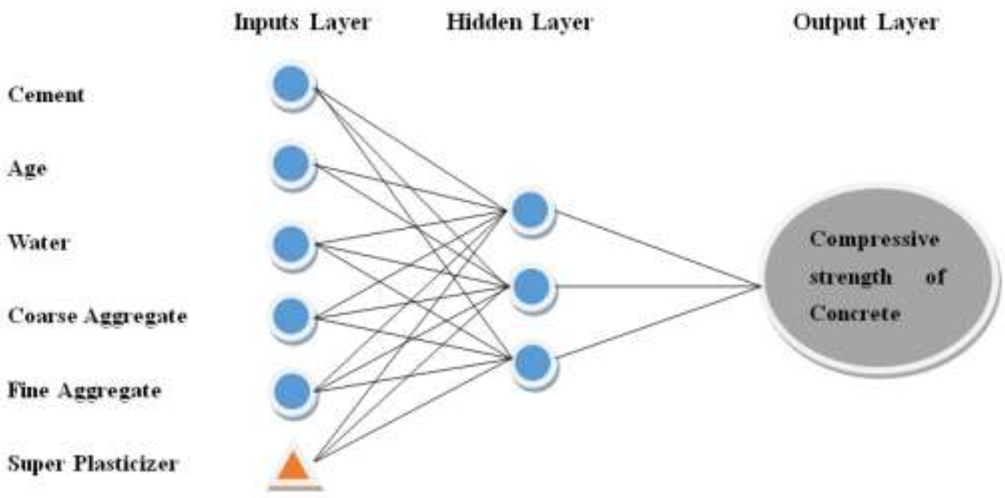

Figure 4.6: The ANN architecture of Model 4

Figure 4.6 shows the ANN architecture for model 4. The parameters in this architecture are learning rate 0.01 and momentum 0.25.

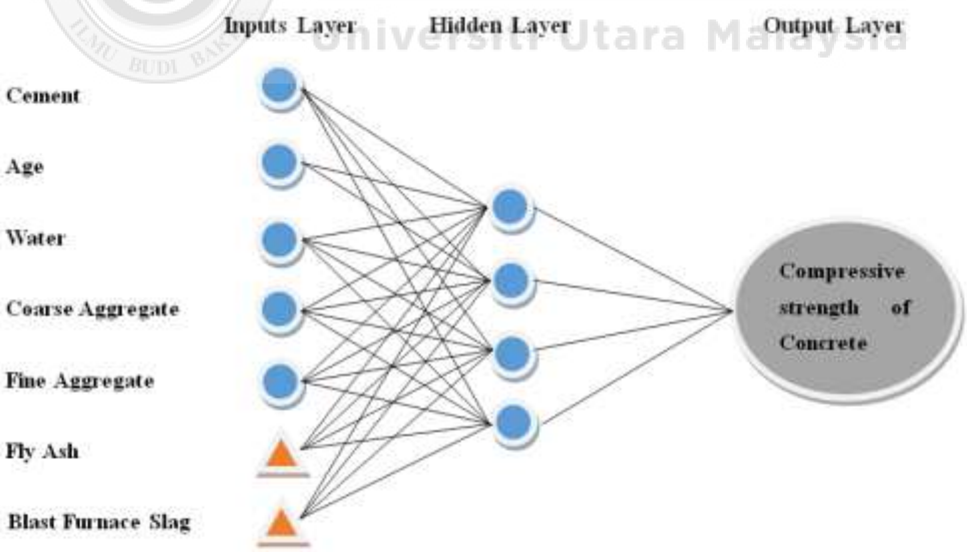

Figure 4.7: The ANN architecture for Model 5

In Figure 4.7, the parameters of model 5 are learning rate equal to 0.01 and momentum equal to 0.5.

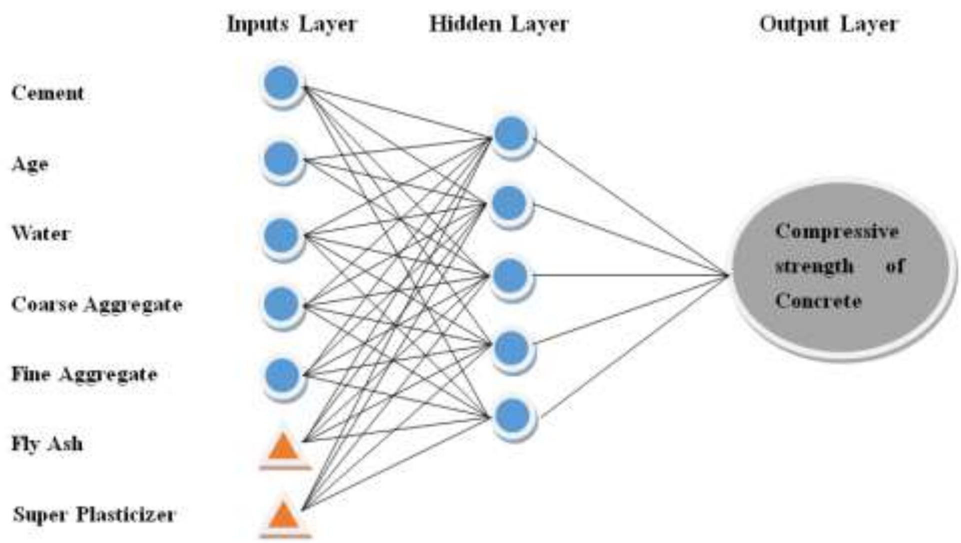

Figure 4.8: The ANN architecture for Model 6

Figure 4.8 gives the information about the ANN architecture of model 6. The parameters of model 6 are learning rate equal to 0.1 and 0.75.

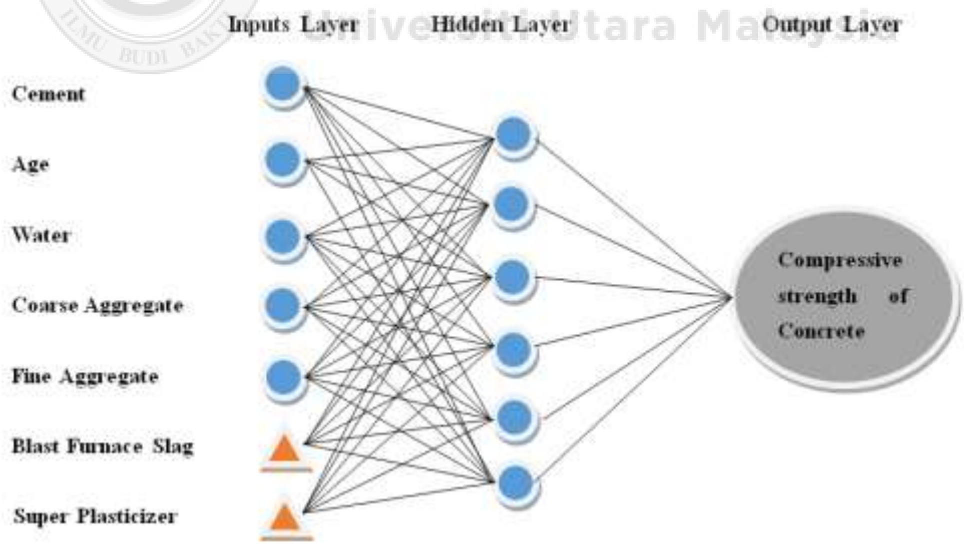

Figure 4.9: The ANN architecture for Model 7

Figure 4.9 shows the ANN architecture of model 7, and the parameters in model 7 are learning rate equal to 0.1 and momentum equal to 0.

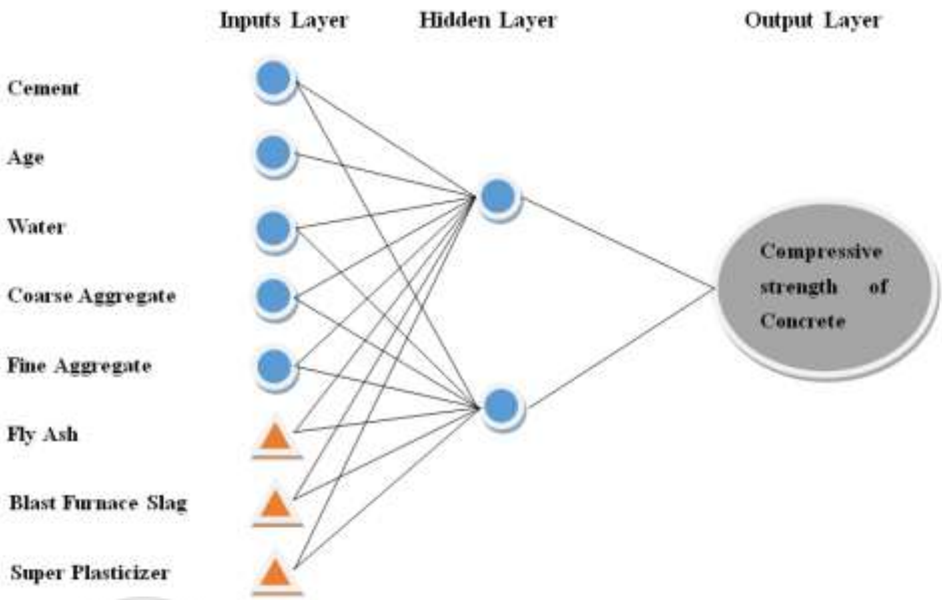

Figure 4.10: The ANN architecture of model 8

In Figure 4.10, it shows the ANN architecture for model 8. In model 8, the final parameters are learning rate equal to 0.1 and momentum equal to 0.5.

4.5 Summary

This chapter presents the deliverables for objective 1, 2, 3 and 4. The main output, which is the compressive concrete strength prediction model is presented in Chapter 5. The performance evaluation results of the model is also shown in Chapter 5, section 5.2 and 5.3.

CHAPTER FIVE

EVALUATION AND DISCUSSION

This chapter consists of two parts. The first part presents the 8 prediction models while the second part shows the evaluation results and discussion.

5.1 Prediction models

As mentioned in Chapter 3, eight prediction models were constructed based on 8 combinations of attributes. The models are listed in Table 5.1.The parameters that are suitable for each model are also shown in the Table 5.1:

Table 5.1

8 prediction models and parameters

Models	Learning rate	Momentum	Hidden layer	Hidden nodes
Model 1 (5 attributes)	0.1	0.75	1	5
Model 2 (6 attributes)	0.5	0.25	1	5
Model 3 (6 attributes)	0.01	0.75	1	5
Model 4 (6 attributes)	0.01	0.25	1	3
Model 5 (7 attributes)	0.01	0.5	1	4

Model 6 (7 attributes)	0.1	0.75	1	5
Model 7 (7 attributes)	0.1	0	1	6
Model 8 (8 attributes)	0.1	0.5	1	2

5.2 Evaluation Results and Discussion

The prediction models were evaluated based on correlation coefficient (R), mean absolute error (MAE) and root mean square error (RMSE). Table 5.2 shows the results:

Table 5.2

Evaluation results

No. of Models	R	MAE	RMSE	TT
Model 1	0.9658	2.4902	3.5597	0.90
Model 2	0.9690	2.2398	2.8786	0.99
Model 3	0.9655	2.6267	3.4045	1.01
Model 4	0.9781	1.9613	2.5950	0.64
Model 5	0.9585	3.2410	3.8710	0.86
Model 6	0.9634	3.3257	3.9412	1.02
Model 7	0.9414	4.7335	7.1735	1.21
Model 8	0.9492	5.1556	6.1640	0.53

Figure5 .1 shows the results in graphical form.

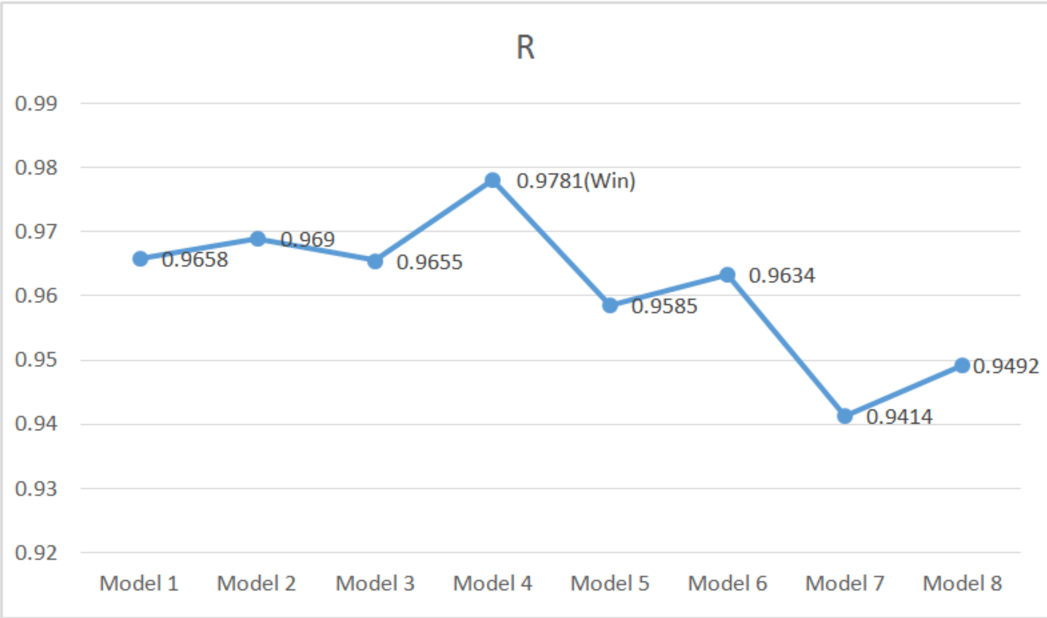

Figure 5.1: Correlation coefficient of 8 models

Table 5.1 shows the most suitable sets of parameters for each of 8 prediction models, it includes learning rate, momentum and number of hidden nodes. The statistical performance measures of correlation coefficient, mean absolute error and root mean square error were used to judge the parameters in each of 8 models. Table 5.2 gives the best results (R, MAE and RMSE) of each model. It can be observed that ANN performs better with correlation coefficient in the range of 0.9414 to 0.9781, lower mean absolute error in the range of 1.9613 to 5.1556 Mpa and the lower root mean square error in the range of 2.5950 to 7.1735 Mpa.

Table 5.2 and Figure 5.1 show the values of correlation coefficient for 8 models, and each model has different combinations of attributes. From these results, model 1, which used 5 basic attributes (cement, water, fine aggregate, coarse aggregate and age) achieved quite good results (correlation coefficient equal to 0.9658). Model 2 with 5 basic attributes + one additional attribute (fly ash) and model 4 with 5 basic attributes + one additional attribute (super plasticizer) obtained better results than model 1. This indicates that when additional attributes such as fly ash or super plasticizer were used to predict compressive concrete strength, the value of correlation coefficient increases.

Model 4 achieved the best results (highest value of correlation coefficient) among the 8 models. However, model 5 (5 basic attributes + two additional attributes fly ash and blast furnace slag), model 6 (5 basic attributes + two additional attributes fly ash and super plasticizer), model 7 (5 basic attributes + two additional attributes blast furnace slag and super plasticizer), and model 8 (5 basic attributes + three additional attributes fly ash, blast furnace slag and super plasticizer) obtained lower results than model 1. However, from the results, all values of correlation coefficient are more than 0.94. These models are considered as good and are acceptable because value of correlation coefficient that is more than 0.9, which means that the relationship between predict value and actual value are close (Wankhade & Kambekar, 2013).

Model 7 obtained the lowest correlation coefficient (0.9414). For model 3 which used 5 basic attributes + one additional attribute blast furnace slag to predict compressive concrete strength, it achieved that the value of correlation coefficient equal to 0.9655, this result is near to the result of model 1 (R=0.9658).

For MAE and RMSE, the lower the values, the better is the result. According to Moriasi et al. (2007), if the value of MAE and RMSE is zero, it means that the result or work is perfect, the accuracy will be 100% and correlation coefficient will be1.

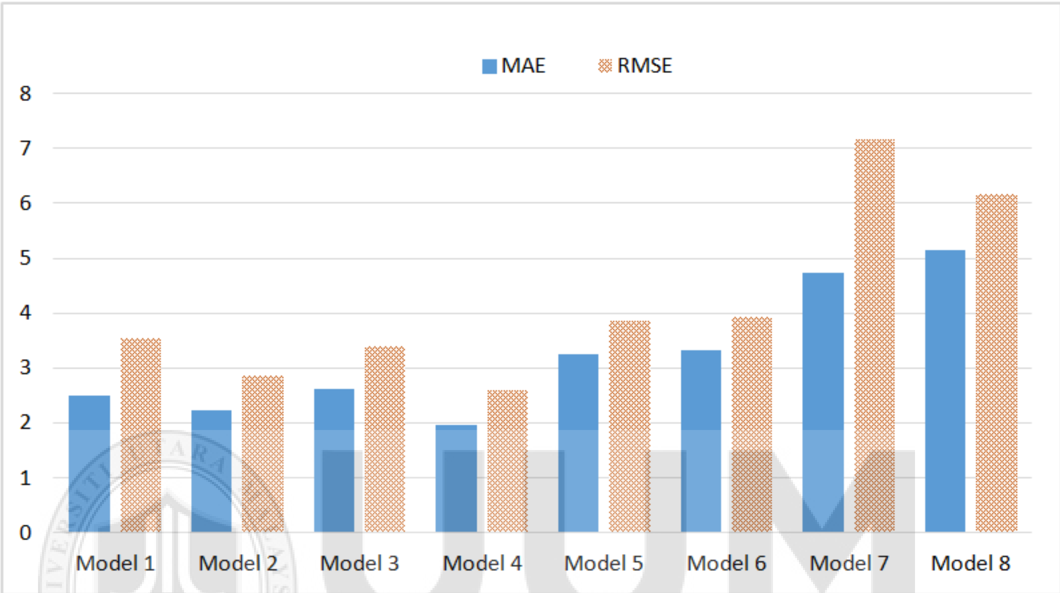

Figure 5.2: MAE and RMSE of 8 models

Figure 5.2 shows the bar chart for the mean absolute error (MAE) and root mean square error (RMSE) of each model. MAE and RMSE are another two statistical measures for judging these 8 prediction models. Based on Table 5.2 and Figure 5.2, two lowest MAE and RMSE values come from Model 2 and 4. The results of Model 3 are close to Model 1. Model 5, 6, 7 and 8 obtained the higher MAE and RMSE values than the basic model. These two statistical performance measures also proved that when adding super plasticizer or fly ash into the basic model (model 1), the model could achieve better correlation coefficient.

According to Kattan (2011), a model can achieved achieved different results (correlation coefficient, mean absolute error and root mean square error) because of three possible reasons. First, the attributes are different from each model, thus giving different results. Second, for each model is different (different ANN architectures). Each model has a different learning rate, momentum and number of hidden nodes, recording to different results. Third, different training algorithms could also give different results. But, in this study, all of the 8 prediction models used the same training algorithm (Back propagation) and the same number of hidden layers (1). Thus, the main reasons influence the results of this study are different attributes of datasets and different ANN prediction architectures.

Figure 5.3 to Figure 5.10 are line charts that compare predicted value with the actual value in this study.

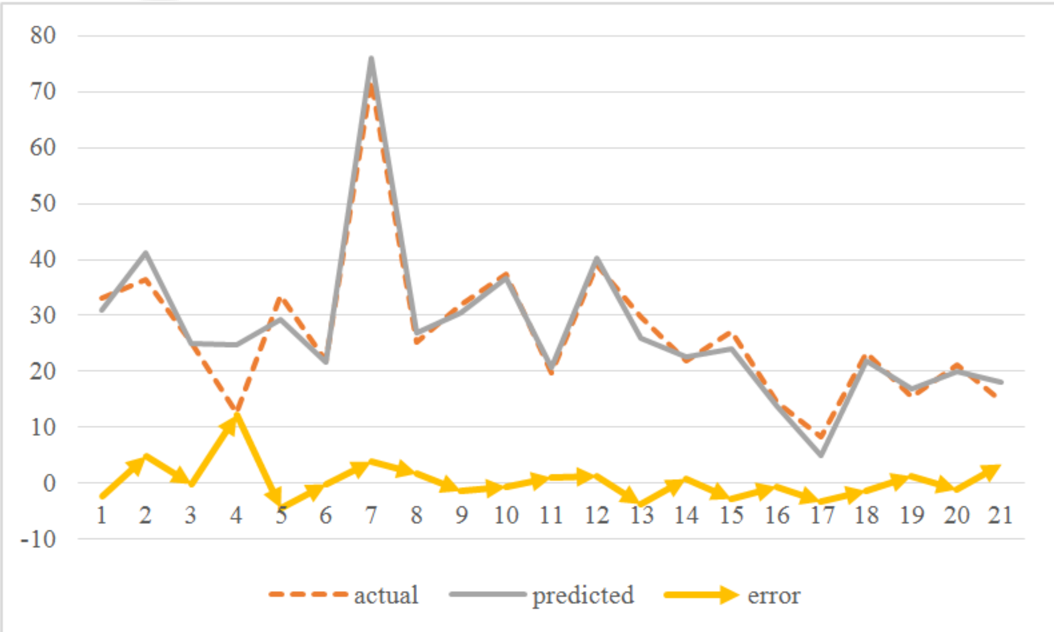

Figure 5.3: Comparison for Model 1

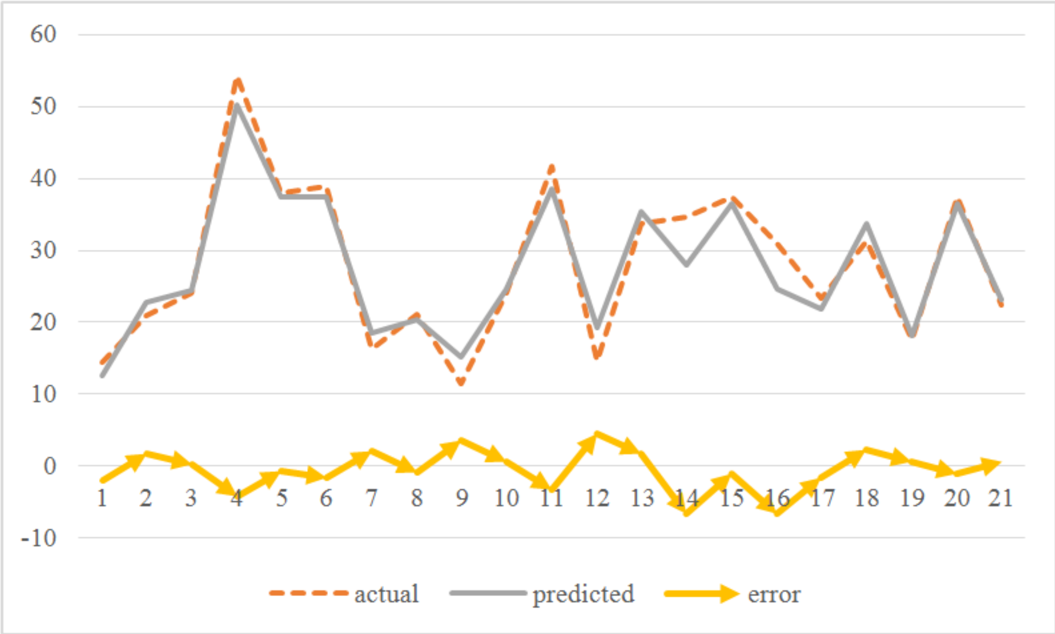

Figure 5.4: Comparison for Model 2

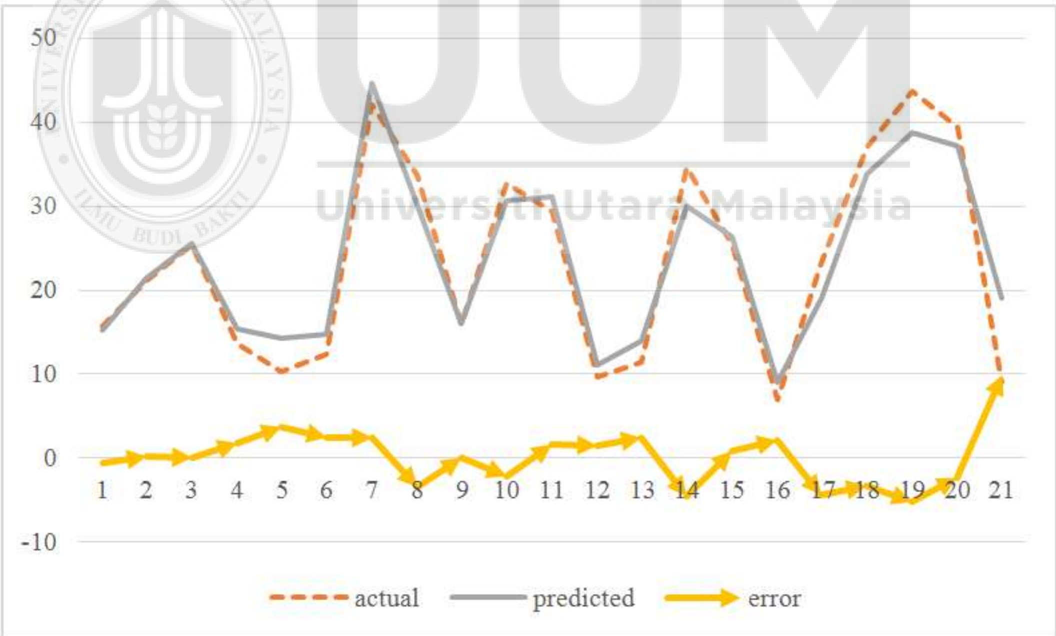

Figure 5.5: Comparison for Model 3

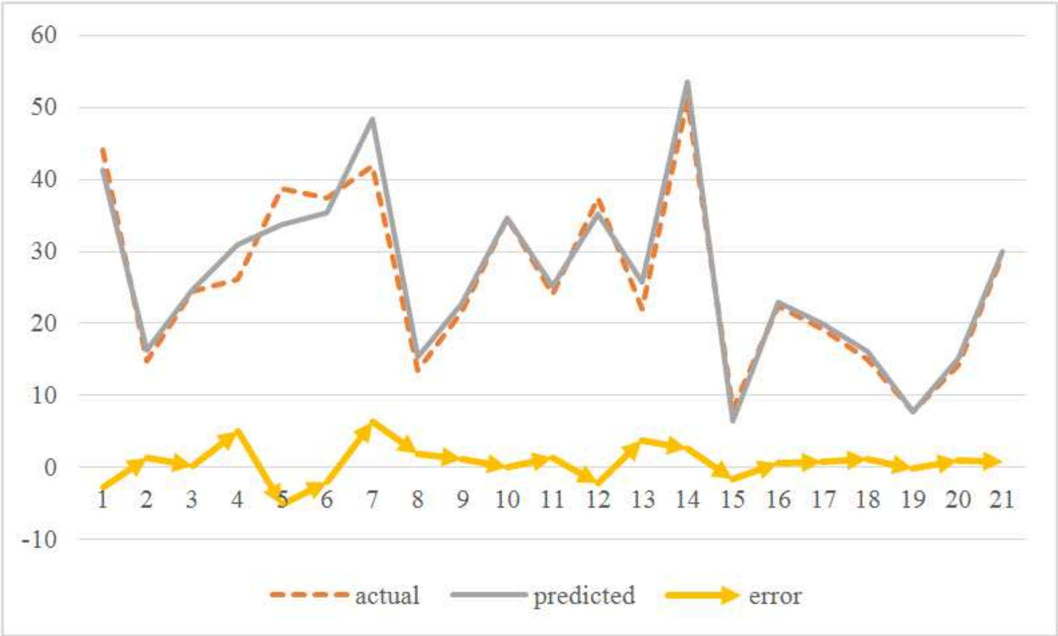

Figure 5.6: Comparison for Model 4

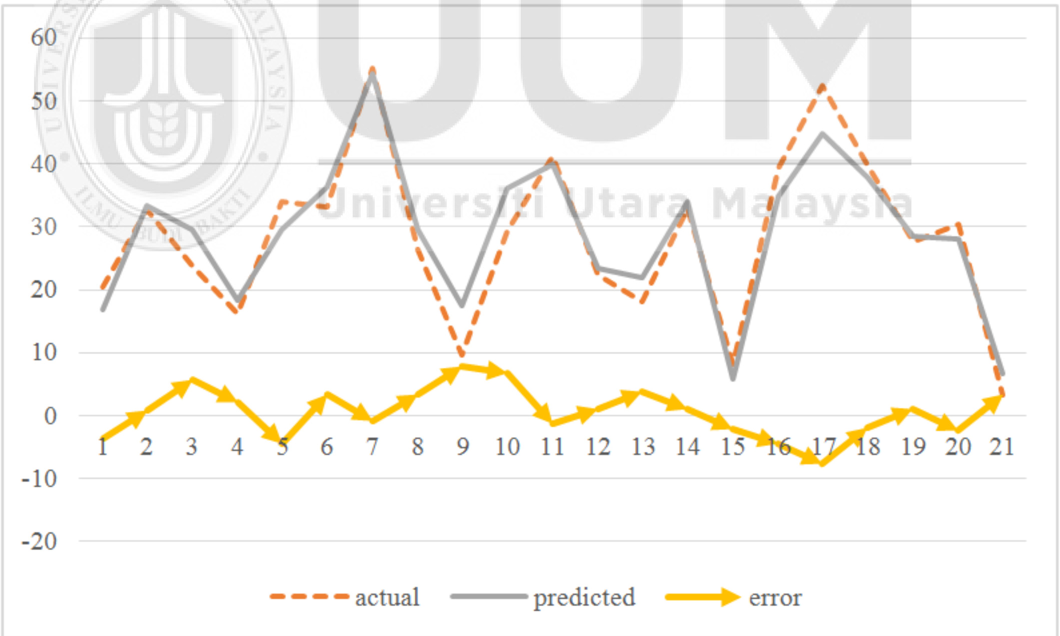

Figure 5.7: Comparison for Model 5

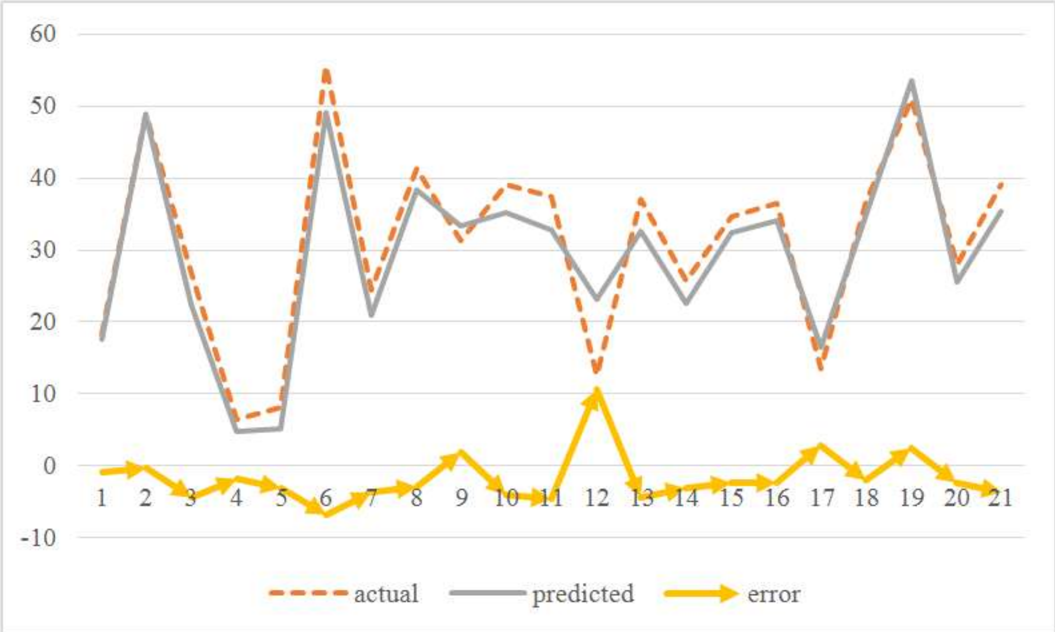

Figure 5.8: Comparison for Model 6

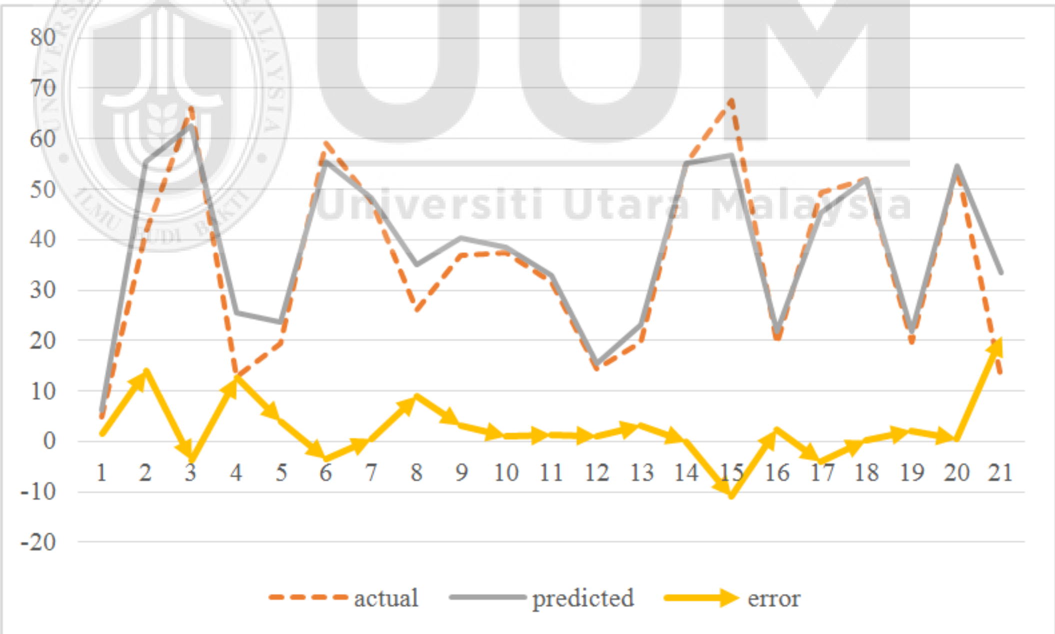

Figure 5.9: Comparison for Model 7

Figure 5.10: Comparison for Model 8

From these figures, it can be seen that the predicted values and the actual values are close together indicating that the models predicted well. The errors are close to 0 which also indicates that the models are good.

Based on the testing results and discussion above, predicting compressive concrete strength of concrete using different combinations of attributes (5 basic attributes + additional attributes) are acceptable. Even though each of these 8 models obtained different results, all the results are considered as good because the R values are more than 0.94. It also proved that the additional attributes (fly ash, super plasticizer and blast furnace slag) have some influence on the prediction models. In other words, the additional attributes can improve the values of correlation coefficient, and decrease the mean absolute error and root mean square error of the prediction models.

5.3 Comparison with past work

Model 8 were compared with the model by Wankhade & Kambekar (2013). In their study, they used the same attributes (cement, water, age, fine aggregate, coarse aggregate, fly ash, super plasticizer and blast furnace slag) but with different parameters. The parameters used by them were 0.9 (learning rate), 0.01(momentum), 1 (hidden layer) and 17 (hidden nodes).

Wankhade's work separated his dataset into 8 groups for different ages (days). In his work, he mentioned that prediction of compressive concrete strength at 28 days is more important for carrying out construction activities, thus, Wankhade used 28 days data to predict compressive concrete strength. This study only chose the data for 28 days from Dataset 8 and used the same parameters of model 8 to compare with Wankhade's work.

The comparison results are as shown in Table 5.3.

Table 5.3

The comparison with past work for 28 days

NAMES	r	m	Hidden layer	Hidden nodes	Percentage Split	R	MAE	RMSE
Wankhade's Work	0.9	0.01	1	17	90%	0.9526	4.1347	4.3864
Model 8 in this study	0.1	0.5	1	2	90%	0.9730	3.1417	3.4516

Based on Table 5.3, the testing results of model 8 are better than Wankhade's work. This is because model 8 used different set of parameters which are learning rate = 0.1, momentum = 0.5 and hidden nodes = 2 for predicting compressive concrete strength. The work of Wankhade achieved 0.9526 for R, 4.1347for MAE and 4.3864 for RMSE. Model 8 achieved better results i.e R = 0.9730, MAE = 3.1417 and RMSE = 3.4516.

Therefore, Table 5.3 proved that the architecture (include parameters) of model 8 could produce a better compressive concrete strength prediction model. It also proved that parameters are one of the most important elements for prediction; the reason is that different sets of parameters would achieve different results.

5.3 Summary

This chapter shows the evaluation results of all 8 compressive concrete strength prediction models. It also discusses reasons for obtaining such results. This study also compared model 8 with an existing compressive concrete strength prediction work. Model 8 of this study performed better than the existing work.

CHAPTER SIX

CONCLUSION

6.1 Work summary

This study aims to construct a compressive concrete strength prediction model based on 8 selected attributes. The attributes were cement fine aggregate, coarse aggregate, water, age, fly ash, blast furnace slag and super plasticizer. The objectives were to (i) identify the basic attributes that can predict compressive strength of concrete with good correlation coefficient; (ii) identify additional attributes that can be used to predict concrete strength with good correlation coefficient; (iii) determine the parameters for weights, learning rate, momentum factor and numbers of hidden nodes, and (iv) design a general ANN architecture for predicting compressive strength of concrete and construct a compressive concrete strength prediction model.

The study has successfully accomplished all objectives. The summary of results is as follows:

Objective 1: Five basics attributes were identified and the attributes are presented in Chapter 4, section 4.1.

Objective 2: Three additional attributes were identified and shown in Chapter 4, section 4.2.

Objective 3: The parameters chosen for 8 different prediction models were obtained through several experiments. These parameters are presented in Chapter 4, section 4.3.

Objective 4: The ANN architecture were produced and shown in Chapter 4, section 4.4. The prediction models are constructed and presented in Chapter 5, section 5.1. Evaluation results are shown in Chapter 5, section 5.2 and 5.3.

On overall, a prediction model from this study (model 8) was compared with an existing work by Wankhade & Kambekar (2013). The Model 8 produced by this study showed better results (prediction of compressive concrete strength at 28 days) than the model produced by Wankhade &Kambekar (2013).

6.2 Contribution

This study made several contributions:

- Used additional prediction attributes: Most study used five attributes to construct the compressive concrete strength prediction model. However, this study attempted to include three more attributes in constructing the prediction model. Good results were obtained when these additional attributes were incorporated to the basic prediction model.
- Parameters for learning rate, weights, momentum, hidden layer, and hidden nodes: This study has successfully identified different parameters for each eight prediction models. When tested using these parameters, good results were obtained.
- Better prediction results: the prediction model (Model 8) showed better

results (prediction of compressive concrete strength at 28 days) than the model by Wankhade & Kambekar (2013).

6.3 Future works

This work determined 5 basic attributes and 3 additional attributes for predicting compressive concrete strength. In order to improve the work several suggestions are listed below:

- Include more extra attributes so that the model can be more robust.
- Develop new architectures and determine new parameters (learning rate, momentum etc.).
- Implement the concrete strength prediction model using computer language such as C-sharp, Java or others so that the application can be used in many platforms.

REFERENCES

Aggarwal, R., Kumar, M., Sharma, R. K., & Sharma, M. K. (2015). Predicting Compressive Strength of Concrete. *International Journal of Applied Science and Engineering*, (April), 171–185.

Akande, K., Owolabi, T., Twaha, S., & Olatunji, S. (2014). Performance Comparison of SVM and ANN in Predicting Compressive Strength of Concrete. *IOSR Journal of Computer Engineering*, *16*(5), 88–94. http://doi.org/10.9790/0661-16518894

Alilou, V. ., & Teshnehlab, M. (2010). Prediction of 28-day compressive strength of concrete on the third day using artificial neural networks. *International Journal of Engineering (IJE)*, (3), 565–576. Retrieved from http://www.cscjournals.org/csc/manuscript/Journals/IJE/volume3/Issue6/IJE-126.pdf

Atici, U. (2011). Prediction of the strength of mineral admixture concrete using multivariable regression analysis and an artificial neural network. *Expert Systems With Applications*, *38*(8), 9609–9618. http://doi.org/10.1016/j.eswa.2011.01.156

Betrie, G. D., Sadiq, R., Morin, K. A., & Tesfamariam, S. (2014). *Uncertainty quanti fi cation and integration of machine learning techniques for predicting acid rock drainage chemistry : A probability bounds approach*. *Science of the Total Environment* (Vol. 490). Elsevier B.V. http://doi.org/10.1016/j.scitotenv.2014.04.125

Bilim, C., Atis, C. D., Tanyildizi, H., & Karahan, O. (2009). Predicting the compressive strength of ground granulated blast furnace slag concrete using artificial neural network. *Advances in Engineering Software*, *40*, 334–340. http://doi.org/10.1016/j.advengsoft.2008.05.005

Bray, J. D., Rourke, T. D. O., Cubrinovski, M., Zupan, J. D., Taylor, M., Toprak, S., … Ballegooy, S. Van. (2013). *Liquefaction Impact on Critical Infrastructure in Christchurch*. Christchurch. Retrieved from http://www.cee.cornell.edu/cee/people/profile.cfm?netid=tdo1

Celikyilmaz, A., & Turksen, I. B. (2008). Enhanced fuzzy system models with improved fuzzy clustering algorithm. *IEEE Transactions on Fuzzy Systems*, *16*(3), 779–794. http://doi.org/10.1109/TFUZZ.2007.905919

Cesa-Bianchi, N., & Lugosi, G. (2006). *Prediction, Learning, and Games*. (N. Cesa-Bianchi & G. Lugosi, Eds.). Barcelona: Cambridge University Press. http://doi.org/10.1017/CBO9780511546921

Chou, J., Chiu, K., Farfoura, M., & Al-Taharwa, I. (2011). Optimizing the prediction accuracy of concrete compressive strength based on a comparison of data mining techniques. *Journal of Computing in Civil Engineering*, *25*(3), 242–253.

De Melo, V. V., & Banzhaf, W. (2016). Predicting high-performance concrete compressive strength using features constructed by kaizen programming. *Proceedings - 2015 Brazilian Conference on Intelligent Systems, BRACIS 2015*, 80–85. http://doi.org/10.1109/BRACIS.2015.56

Deepa, C., Sathiya Kumari, K., & Pream Sudha, V. (2010). Prediction of the Compressive Strength of High Performance Concrete Mix using Tree Based Modeling. *International Journal of Comuputer Apllications*, *6*(5), 18–24.

Doug. (2016). How to choose the number of hidden layers and nodes in a feedforward neural network? Retrieved from http://stats.stackexchange.com/questions/181/how-to-choose-the-number-of-hidden-layers-and-nodes-in-a-feedforward-neural-netw

Fred, F. (2014). What are advantages of Artificial Neural Network over Support Vector Machine. Retrieved from http://stackoverflow.com/questions/11632516/what-are-advantages-of-artificial-neural-networks-over-support-vector-machines

Ghan, Y. N., Peng, G. F., & Anson, M. (1999). Residual strength and pore structure of high-strength concrete and normal strength concrete after exposure to high temperatures, *21*, 23–27.

Gilan, S. S., Ali, A. M., & Ramezanianpour, A. A. (2011). Evolutionary fuzzy function with support vector regression for the prediction of concrete compressive strength. *Proceedings - UKSim 5th European Modelling Symposium on Computer Modelling and Simulation, EMS 2011*, 263–268. http://doi.org/10.1109/EMS.2011.28

Gupta, S. M. (2007). Support Vector Machines based Modelling of Concrete Strength. *World Academy of Science, Engineering and Technology*, *36*(1), 305–311.

Huixian, L., Housner, G. W., Lili, X., & Duxin, H. (2002). The Great Tangshan Earthquake of 1976. *Technical Report: Caltech EERL.2002.001*.

Jang, J. S. R. (1993). ANFIS : Adaptive Network Based Fuzzy Inference System. *Systems, Man and Cybernetics, IEEE Transactions on*, *23*(3), 665–685. Retrieved from http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=256541

Jayalakshmi, T., & Santhakumaran, A. (2011). Statistical normalization and back propagation for classification. *International Journal of Computer Theory and Engineering*, *3*(1), 89–93. Retrieved from http://www.ijcte.org/papers/288-L052.pdf

Jordans, M., Kohrt, B., & Tol, W. (2015). *Nepal Earthquakes 2015*. Nepal. Retrieved from https://interagencystandingcommittee.org/system/files/20150622_nepal_earthquakes_mhpss_desk_review_150619.pdf

Kabir, A., Hasan, M., & Miah, K. (2012). Predicting 28 Days Compressive Strength of Concrete from 7 Days Test Result. *Advances in Design and Construction of Structures*, 18–22.

Kasi, K. (2015). What is difference between SVM and Neural Networks? Retrieved from https://www.quora.com/What-is-difference-between-SVM-and-Neural-Networks

Kattan, M. W. (2011). Factors Affecting the Accuracy of Prediction Models Limit the Comparison of Rival Prediction Models When Applied to Separate Data Sets. *European Urology*, *59*(4), 566–567. http://doi.org/10.1016/j.eururo.2010.11.039

Kosko, B. (1992). Neural Networks and Fuzzy Systems. *Journal of Englewood Cliffs*, 449.

Liu, J. C., Sue, M. L., & Kou, C. H. (2009). Estimating the strength of concrete using surface rebound value and design parameters of concrete material. *Tamkang Journal of Science and Engineering*, *12*(1), 1–7.

MacKay, D. J. C. (1994). Bayesian non-linear modelling for the energy prediction competition. *ASHRAE Transactions*, *100*, 1053–1062.

Makin, J. G. (2006). Backpropagation. *University of California, Berkeley*, 1–8.

Martinez-Molina, W., Torres-Acosta, A. A., Jauregui, J. C., Chavez-Garcia, H. L., Alonso-Guzman, E. M., Graff, M., & Arteaga-Arcos, J. C. (2014). Predicting concrete compressive strength and modulus of rupture using different NDT techniques. *Advances in Materials Science and Engineering*, *2014*, 1–15. http://doi.org/10.1155/2014/742129

Michele, M., Raucoules, D., Lasserre, C., Pathier, E., Klinger, Y., Van der Woerd, J., … Xu, X. (2010). The M-w 7.9, 12 May 2008 Sichuan earthquake rupture measured by sub-pixel correlation of ALOS PALSAR amplitude images. *Earth Planets And Space*, *62*(11), 875–879. http://doi.org/10.5047/eps.2009.05.002

Moriasi, D. N., Arnold, J. G., Liew, M. W. Van, Bingner, R. L., Harmel, R. D., & Veith, T. L. (2007). Model evaluation guidelines for systematic quantification of accuracy in watershed simulations. *Transactin of the ASABE*, *50*(3), 885–900.

Musson, R. M. W. (2008). *The seismicity of the British Isles to 1600*. British. Retrieved from http://www.earthquakes.bgs.ac.uk/historical/data/studies/MUSS008/MUSS008.pdf

Muthupriya, P., Subramanian, K., & Vishnuram, B. G. (2011). Prediction of Compressive Strength and Durability of High Performance Concrete By Artificial Neural Networks. *International Journal of Optimization in Civil Engineering*, 189–209.

Nayak, S. C., Misra, B. B., & Behera, H. S. (2014). Impact of Data Normalization on

Stock Index Forecasting. *International Journal of Computer Information Systems and Industrial Management Applications*, *6*, 257–269.

Nikoo, M., Torabian Moghadam, F., & Sadowski, L. (2015). Prediction of concrete compressive strength by evolutionary artificial neural networks. *Advances in Materials Science and Engineering*, *2015*, 1–8. http://doi.org/10.1155/2015/849126

Noorzaei, J., Hakim, S. J. S., Jaafar, M. S., & Thanoon, W. A. M. (2007). Development of Artificial Neural Networks for Predicting Concrete Compressive Strength. *International Journal of Engineering and Technology*, *4*(2), 141–153.

Oravec, M., Petráš, M., & Pilka, F. (2008). Video Traffic Prediction Using Neural Networks. *Electrical Engineering*, *5*(4), 59–78.

Panchal, G., Ganatra, A., Kosta, Y. P., & Panchal, D. (2011). Behaviour Analysis of Multilayer Perceptrons with Multiple Hidden Neurons and Hidden Layers. *Internatinal Journal of Computer Theory and Engineering*, *3*(2), 332–337.

Plagianakos, V. P., Magoulas, G. D., & Vrahatis, M. N. (2001). Leaning rate adaptation in stochastic gradient descent. *In Advances in Convex Analysis and Global optimization.Springer US*, 433–444.

Pradhan, B., & Kundu, D. (2011). Bayes estimation and prediction of the two-parameter gamma distribution. *Journal of Statistical Computation and Simulation*, *81*(9), 1187–1198. http://doi.org/10.1080/00949651003796335

Preetham, S., Shivaraj, M., Prema kumar, W. P., & Kumar, H. R. (2014). Support Vector Machine Technique in Analysis of Concrete-Critical Review. *International Journal of Emerging Technologies and Engineering*, *1*(9), 199–203.

Rasa, E., Ketabchi, H., & Afshar, M. H. (2009). Predicting Density and Compressive Strength of Concrete Cement Paste Containing Silica Fume Using Arti cial Neural Networks. *Journal of Civil Engineering*, *16*(1), 33–42.

Rashid, M. A., & Mansur, M. A. (2009). Considerations in producing high strength concrete. *Journal of Civil Engineering*, *37*(1), 53–63.

Sakr, G. E., Elhajj, I. H., & Mitri, G. (2011). *Efficient forest fire occurrence prediction for developing countries using two weather parameters*. *Engineering Applications of Artificial Intelligence* (Vol. 24). http://doi.org/10.1016/j.engappai.2011.02.017

Suhad, M. A., & Abbas, M. A. (2015). SUPPORT VECTOR MACHINE (SVM) FOR MODELLING THE STRENGTH OF LIGHTWEIGHT FOAMED CONCRETE. *Diyala Jounal of Engineering Sciences*, 29–36.

Thomas, W. K. (2015). What is the difference between a Bayesian network and Bayesian neural network? Retrieved August 18, 2016, from https://www.researchgate.net/post/What_is_the_difference_between_a_Bayesian_network_and_Bayesian_neural_network

Turksen, I. B. (2008). Fuzzy functions with LSE. *Applied Soft Computing Journal*, *8*(3), 1178–1188. http://doi.org/10.1016/j.asoc.2007.12.004

Uppada, S. R., Balu, A., Gupta, A. K., & Dutta, J. R. (2014). Modeling Lipase Production From Co-cultures of Lactic Acid Bacteria Using Neural Networks and Support Vector Machine with Genetic Algorithm Optimization. *International Journal of Emerging Technologies in Computational and Applied Sciences (IJETCAS)*, 38–43.

Vakhshouri, B., & Nejadi, S. (2015). Predicition Of Compressive Strength In Light-Weight Self-Compacting Concrete By ANFIS Analytical Model. *Archives of Civil Engineering*, *61*(2). http://doi.org/10.1515/ace-2015-0014

Vale, R. (2014). Bayesian Prediction for The Winds of Winter. Retrieved from http://arxiv.org/abs/1409.5830

Wankhade, M. W., & Kambekar, A. R. (2013). Prediction of compressive strength of concrete using Artificial Neural Network. *International Journal of Scientific Research and Reviews*, *2*(2), 11–26.

Yaqub, M., Taxila, T., Bukhari, I., & Taxila, T. (2006). Development of Mix Design for High Strength Development of Mix Design for High Strength. *Conference on Our World in Concrete & Structures*, 31–35.

Yeh, I.-C. (1998). Modeling of Strength of High-Performance Concrete Using

Artificial Neural Networks. *Journal of Civil Engineering*, *28*(12), 1797–1808. Retrieved from http://www.sciencedirect.com/science/article/pii/S0008884698001653

Yeh, I.-C. (2003). A Mix Proportioning Methodology for Fly Ash and Slag Concrete Using Artificial Neural Networks. *Chung Hua Journal of Science and Engineering*, *1*(1), 77–84.

Yeh, I.-C. (2006). Exploring Concrete Slump Model Using Artificial Neural Networks. *Journal of Computing in Civil Engineering*, 217–221. http://doi.org/10.1016/j.neunet.2008.04.002

Yeh, I. C., & Lien, L. C. (2009). Knowledge discovery of concrete material using Genetic Operation Trees. *Expert Systems with Applications*, *36*(3 PART 2), 5807–5812. http://doi.org/10.1016/j.eswa.2008.07.004

APPENDICES

Appendix A. Sample of Raw Dataset

Table 8.1 50 Samples of Dataset 1

Cement (component 1)(kg in a m^3 mixture)	Water (component 4)(kg in a m^3 mixture)	Coarse Aggregate (component 6)(kg in a m^3 mixture)	Fine Aggregate (component 7)(kg in a m^3 mixture)	Age (day)	Concrete compressive strength(MPa, megapascals)
475.0	228.0	932.0	594.0	28	39.29
380.0	228.0	932.0	670.0	90	52.91
475.0	228.0	932.0	594.0	180	42.62
380.0	228.0	932.0	670.0	365	52.52
380.0	228.0	932.0	670.0	270	53.30
475.0	228.0	932.0	594.0	7	38.60
475.0	228.0	932.0	594.0	270	42.13
475.0	228.0	932.0	594.0	90	42.23
380.0	228.0	932.0	670.0	180	53.10
349.0	192.0	1047.0	806.9	3	15.05
475.0	228.0	932.0	594.0	365	41.93
310.0	192.0	971.0	850.6	3	9.87
485.0	146.0	1120.0	800.0	28	71.99
376.0	214.6	1003.5	762.4	3	16.28
376.0	214.6	1003.5	762.4	14	25.62
376.0	214.6	1003.5	762.4	28	31.97
376.0	214.6	1003.5	762.4	56	36.30
376.0	214.6	1003.5	762.4	100	43.06
405.0	175.0	1120.0	695.0	28	52.30
436.0	218.0	838.4	719.7	28	23.85
289.0	192.0	913.2	895.3	90	32.07
289.0	192.0	913.2	895.3	3	11.65
393.0	192.0	940.6	785.6	3	19.20

393.0	192.0	940.6	785.6	90	48.85
393.0	192.0	940.6	785.6	28	39.60
480.0	192.0	936.2	712.2	28	43.94
480.0	192.0	936.2	712.2	7	34.57
480.0	192.0	936.2	712.2	90	54.32
480.0	192.0	936.2	712.2	3	24.40
333.0	192.0	931.2	842.6	3	15.62
255.0	192.0	889.8	945.0	90	21.86
255.0	192.0	889.8	945.0	7	10.22
289.0	192.0	913.2	895.3	7	14.60
255.0	192.0	889.8	945.0	28	18.75
333.0	192.0	931.2	842.6	28	31.97
333.0	192.0	931.2	842.6	7	23.40
289.0	192.0	913.2	895.3	28	25.57
333.0	192.0	931.2	842.6	90	41.68
393.0	192.0	940.6	785.6	7	27.74
255.0	192.0	889.8	945.0	3	8.20
397.0	185.7	1040.6	734.3	28	33.08
382.5	185.7	1047.8	739.3	7	24.07
295.8	185.7	1091.4	769.3	7	14.84
397.0	185.7	1040.6	734.3	7	25.45
381.4	185.7	1104.6	784.3	28	22.49
295.8	185.7	1091.4	769.3	28	25.22
238.1	185.7	1118.8	789.3	28	17.58
339.2	185.7	1069.2	754.3	7	21.18
381.4	185.7	1104.6	784.3	7	14.54
339.2	185.7	1069.2	754.3	28	31.90

Table 8.2 50 Samples of Dataset 2

Cement (component 1)(kg in a m^3 mixture)	Fly Ash (component 3)(kg in a m^3 mixture)	Water (component 4)(kg in a m^3 mixture)	Coarse Aggregate (component 6)(kg in a m^3 mixture)	Fine Aggregate (component 7)(kg in a m^3 mixture)	Age (day)	Concrete compressive strength(MPa, megapascals)
475.0	0.0	228.0	932.0	594.0	28	39.29
380.0	0.0	228.0	932.0	670.0	90	52.91
475.0	0.0	228.0	932.0	594.0	180	42.62
380.0	0.0	228.0	932.0	670.0	270	53.30
475.0	0.0	228.0	932.0	594.0	7	38.60
475.0	0.0	228.0	932.0	594.0	270	42.13
475.0	0.0	228.0	932.0	594.0	90	42.23
380.0	0.0	228.0	932.0	670.0	180	53.10
349.0	0.0	192.0	1047.0	806.9	3	15.05
475.0	0.0	228.0	932.0	594.0	365	41.93
310.0	0.0	192.0	971.0	850.6	3	9.87
485.0	0.0	146.0	1120.0	800.0	28	71.99
376.0	0.0	214.6	1003.5	762.4	3	16.28
376.0	0.0	214.6	1003.5	762.4	14	25.62
376.0	0.0	214.6	1003.5	762.4	28	31.97
376.0	0.0	214.6	1003.5	762.4	56	36.30
376.0	0.0	214.6	1003.5	762.4	100	43.06
505.0	60.0	195.0	1030.0	630.0	28	64.02
405.0	0.0	175.0	1120.0	695.0	28	52.30
165.0	143.6	163.8	1005.6	900.9	14	16.88
165.0	143.6	163.8	1005.6	900.9	28	26.20
165.0	143.6	163.8	1005.6	900.9	56	36.56
165.0	143.6	163.8	1005.6	900.9	100	37.96
436.0	0.0	218.0	838.4	719.7	28	23.85

289.0	0.0	192.0	913.2	895.3	90	32.07
289.0	0.0	192.0	913.2	895.3	3	11.65
393.0	0.0	192.0	940.6	785.6	3	19.20
393.0	0.0	192.0	940.6	785.6	90	48.85
393.0	0.0	192.0	940.6	785.6	28	39.60
480.0	0.0	192.0	936.2	712.2	28	43.94
480.0	0.0	192.0	936.2	712.2	7	34.57
480.0	0.0	192.0	936.2	712.2	90	54.32
480.0	0.0	192.0	936.2	712.2	3	24.40
333.0	0.0	192.0	931.2	842.6	3	15.62
255.0	0.0	192.0	889.8	945.0	90	21.86
255.0	0.0	192.0	889.8	945.0	7	10.22
289.0	0.0	192.0	913.2	895.3	7	14.60
255.0	0.0	192.0	889.8	945.0	28	18.75
333.0	0.0	192.0	931.2	842.6	28	31.97
333.0	0.0	192.0	931.2	842.6	7	23.40
289.0	0.0	192.0	913.2	895.3	28	25.57
333.0	0.0	192.0	931.2	842.6	90	41.68
393.0	0.0	192.0	940.6	785.6	7	27.74
255.0	0.0	192.0	889.8	945.0	3	8.20
397.0	0.0	185.7	1040.6	734.3	28	33.08
382.5	0.0	185.7	1047.8	739.3	7	24.07
295.8	0.0	185.7	1091.4	769.3	7	14.84
397.0	0.0	185.7	1040.6	734.3	7	25.45
381.4	0.0	185.7	1104.6	784.3	28	22.49
295.8	0.0	185.7	1091.4	769.3	28	25.22

Table 8.3 50 Samples of Dataset 3

Cement (component 1)(kg in a m^3 mixture)	Blast Furnace Slag (component 2)(kg in a m^3 mixture)	Water (component 4)(kg in a m^3 mixture)	Coarse Aggregate (component 6)(kg in a m^3 mixture)	Fine Aggregate (component 7)(kg in a m^3 mixture)	Age (day)	Concrete compressive strength(MPa, megapascals)
332.5	142.5	228.0	932.0	594.0	270	40.27
332.5	142.5	228.0	932.0	594.0	365	41.05
198.6	132.4	192.0	978.4	825.5	360	44.30
475.0	0.0	228.0	932.0	594.0	28	39.29
198.6	132.4	192.0	978.4	825.5	90	38.07
198.6	132.4	192.0	978.4	825.5	28	28.02
427.5	47.5	228.0	932.0	594.0	270	43.01
190.0	190.0	228.0	932.0	670.0	90	42.33
304.0	76.0	228.0	932.0	670.0	28	47.81
380.0	0.0	228.0	932.0	670.0	90	52.91
266.0	114.0	228.0	932.0	670.0	365	52.91
198.6	132.4	192.0	978.4	825.5	180	41.72
475.0	0.0	228.0	932.0	594.0	270	42.13
190.0	190.0	228.0	932.0	670.0	365	53.69
237.5	237.5	228.0	932.0	594.0	270	38.41
237.5	237.5	228.0	932.0	594.0	28	30.08
427.5	47.5	228.0	932.0	594.0	7	35.08
349.0	0.0	192.0	1047.0	806.9	3	15.05
380.0	95.0	228.0	932.0	594.0	180	40.76
237.5	237.5	228.0	932.0	594.0	7	26.26
380.0	95.0	228.0	932.0	594.0	7	32.82
332.5	142.5	228.0	932.0	594.0	180	39.78
190.0	190.0	228.0	932.0	670.0	180	46.93
237.5	237.5	228.0	932.0	594.0	90	33.12
304.0	76.0	228.0	932.0	670.0	90	49.19

139.6	209.4	192.0	1047.0	806.9	7	14.59
198.6	132.4	192.0	978.4	825.5	7	14.64
475.0	0.0	228.0	932.0	594.0	365	41.93
485.0	0.0	146.0	1120.0	800.0	28	71.99
376.0	0.0	214.6	1003.5	762.4	100	43.06
480.0	0.0	192.0	936.2	712.2	7	34.57
480.0	0.0	192.0	936.2	712.2	90	54.32
480.0	0.0	192.0	936.2	712.2	3	24.40
333.0	0.0	192.0	931.2	842.6	3	15.62
255.0	0.0	192.0	889.8	945.0	90	21.86
255.0	0.0	192.0	889.8	945.0	7	10.22
289.0	0.0	192.0	913.2	895.3	7	14.60
255.0	0.0	192.0	889.8	945.0	28	18.75
333.0	0.0	192.0	931.2	842.6	28	31.97
333.0	0.0	192.0	931.2	842.6	7	23.40
289.0	0.0	192.0	913.2	895.3	28	25.57
333.0	0.0	192.0	931.2	842.6	90	41.68
393.0	0.0	192.0	940.6	785.6	7	27.74
255.0	0.0	192.0	889.8	945.0	3	8.20
158.8	238.2	185.7	1040.6	734.3	7	9.62
239.6	359.4	185.7	941.6	664.3	7	25.42
238.2	158.8	185.7	1040.6	734.3	7	15.69
239.6	359.4	185.7	941.6	664.3	28	39.44
220.8	147.2	185.7	1055.0	744.3	28	25.75
397.0	0.0	185.7	1040.6	734.3	28	33.08

Table 8.4 50 Samples of Dataset 4

Cement (component 1)(kg in a m^3 mixture)	Water (component 4)(kg in a m^3 mixture)	Superplasticizer (component 5)(kg in a m^3 mixture)	Coarse Aggregate (component 6)(kg in a m^3 mixture)	Fine Aggregate (component 7)(kg in a m^3 mixture)	Age (day)	Concrete compressive strength(MPa, megapascals)
540.0	162.0	2.5	1040.0	676.0	28	79.99
540.0	162.0	2.5	1055.0	676.0	28	61.89
475.0	228.0	0.0	932.0	594.0	28	39.29
380.0	228.0	0.0	932.0	670.0	90	52.91
475.0	228.0	0.0	932.0	594.0	180	42.62
380.0	228.0	0.0	932.0	670.0	365	52.52
380.0	228.0	0.0	932.0	670.0	270	53.30
475.0	228.0	0.0	932.0	594.0	7	38.60
475.0	228.0	0.0	932.0	594.0	270	42.13
475.0	228.0	0.0	932.0	594.0	90	42.23
380.0	228.0	0.0	932.0	670.0	180	53.10
349.0	192.0	0.0	1047.0	806.9	3	15.05
475.0	228.0	0.0	932.0	594.0	365	41.93
310.0	192.0	0.0	971.0	850.6	3	9.87
485.0	146.0	0.0	1120.0	800.0	28	71.99
531.3	141.8	28.2	852.1	893.7	3	41.30
531.3	141.8	28.2	852.1	893.7	7	46.90
531.3	141.8	28.2	852.1	893.7	28	56.40
531.3	141.8	28.2	852.1	893.7	56	58.80
531.3	141.8	28.2	852.1	893.7	91	59.20
376.0	214.6	0.0	1003.5	762.4	3	16.28
376.0	214.6	0.0	1003.5	762.4	14	25.62
376.0	214.6	0.0	1003.5	762.4	28	31.97
376.0	214.6	0.0	1003.5	762.4	56	36.30
376.0	214.6	0.0	1003.5	762.4	100	43.06

500.0	140.0	4.0	966.0	853.0	28	67.57
451.0	165.0	11.3	1030.0	745.0	28	78.80
516.0	162.0	8.2	801.0	802.0	28	41.37
520.0	170.0	5.2	855.0	855.0	28	60.28
528.0	185.0	6.9	920.0	720.0	28	56.83
520.0	175.0	5.2	870.0	805.0	28	51.02
500.1	200.0	3.0	1124.4	613.2	28	44.13
405.0	175.0	0.0	1120.0	695.0	28	52.30
516.0	162.0	8.3	801.0	802.0	28	41.37
475.0	162.0	9.5	1044.0	662.0	28	58.52
500.0	151.0	9.0	1033.0	655.0	28	69.84
436.0	218.0	0.0	838.4	719.7	28	23.85
289.0	192.0	0.0	913.2	895.3	90	32.07
289.0	192.0	0.0	913.2	895.3	3	11.65
393.0	192.0	0.0	940.6	785.6	3	19.20
393.0	192.0	0.0	940.6	785.6	90	48.85
393.0	192.0	0.0	940.6	785.6	28	39.60
480.0	192.0	0.0	936.2	712.2	28	43.94
480.0	192.0	0.0	936.2	712.2	7	34.57
480.0	192.0	0.0	936.2	712.2	90	54.32
480.0	192.0	0.0	936.2	712.2	3	24.40
333.0	192.0	0.0	931.2	842.6	3	15.62
255.0	192.0	0.0	889.8	945.0	90	21.86
255.0	192.0	0.0	889.8	945.0	7	10.22
289.0	192.0	0.0	913.2	895.3	7	14.60

Table 8.5 50 Samples of Dataset 5

Cement (component 1)(kg in a m^3 mixture)	Blast Furnace Slag (component 2)(kg in a m^3 mixture)	Fly Ash (component 3)(kg in a m^3 mixture)	Water (component 4)(kg in a m^3 mixture)	Coarse Aggregate (component 6)(kg in a m^3 mixture)	Fine Aggregate (component 7)(kg in a m^3 mixture)	Age (day)	Concrete compressive strength(MPa, megapascals)
332.5	142.5	0.0	228.0	932.0	594.0	270	40.27
332.5	142.5	0.0	228.0	932.0	594.0	365	41.05
198.6	132.4	0.0	192.0	978.4	825.5	360	44.30
266.0	114.0	0.0	228.0	932.0	670.0	90	47.03
380.0	95.0	0.0	228.0	932.0	594.0	365	43.70
380.0	95.0	0.0	228.0	932.0	594.0	28	36.45
139.6	209.4	0.0	192.0	1047.0	806.9	90	39.36
342.0	38.0	0.0	228.0	932.0	670.0	365	56.14
380.0	95.0	0.0	228.0	932.0	594.0	90	40.56
475.0	0.0	0.0	228.0	932.0	594.0	180	42.62
427.5	47.5	0.0	228.0	932.0	594.0	28	37.43
475.0	0.0	0.0	228.0	932.0	594.0	7	38.60
304.0	76.0	0.0	228.0	932.0	670.0	365	55.26
266.0	114.0	0.0	228.0	932.0	670.0	365	52.91
198.6	132.4	0.0	192.0	978.4	825.5	180	41.72
475.0	0.0	0.0	228.0	932.0	594.0	270	42.13
190.0	190.0	0.0	228.0	932.0	670.0	365	53.69
237.5	237.5	0.0	228.0	932.0	594.0	270	38.41
237.5	237.5	0.0	228.0	932.0	594.0	28	30.08
237.5	237.5	0.0	228.0	932.0	594.0	180	36.25
342.0	38.0	0.0	228.0	932.0	670.0	90	50.46
427.5	47.5	0.0	228.0	932.0	594.0	365	43.70
237.5	237.5	0.0	228.0	932.0	594.0	365	39.00
380.0	0.0	0.0	228.0	932.0	670.0	180	53.10

427.5	47.5	0.0	228.0	932.0	594.0	90	41.54
139.6	209.4	0.0	192.0	1047.0	806.9	7	14.59
198.6	132.4	0.0	192.0	978.4	825.5	7	14.64
475.0	0.0	0.0	228.0	932.0	594.0	365	41.93
198.6	132.4	0.0	192.0	978.4	825.5	3	9.13
304.0	76.0	0.0	228.0	932.0	670.0	180	50.95
332.5	142.5	0.0	228.0	932.0	594.0	28	33.02
304.0	76.0	0.0	228.0	932.0	670.0	270	54.38
266.0	114.0	0.0	228.0	932.0	670.0	270	51.73
310.0	0.0	0.0	192.0	971.0	850.6	3	9.87
190.0	190.0	0.0	228.0	932.0	670.0	270	50.66
266.0	114.0	0.0	228.0	932.0	670.0	180	48.70
342.0	38.0	0.0	228.0	932.0	670.0	270	55.06
376.0	0.0	0.0	214.6	1003.5	762.4	100	43.06
505.0	0.0	60.0	195.0	1030.0	630.0	28	64.02
405.0	0.0	0.0	175.0	1120.0	695.0	28	52.30
200.0	200.0	0.0	190.0	1145.0	660.0	28	49.25
165.0	0.0	143.6	163.8	1005.6	900.9	14	16.88
165.0	0.0	143.6	163.8	1005.6	900.9	28	26.20
165.0	0.0	143.6	163.8	1005.6	900.9	56	36.56
165.0	0.0	143.6	163.8	1005.6	900.9	100	37.96
436.0	0.0	0.0	218.0	838.4	719.7	28	23.85
289.0	0.0	0.0	192.0	913.2	895.3	90	32.07
289.0	0.0	0.0	192.0	913.2	895.3	3	11.65
393.0	0.0	0.0	192.0	940.6	785.6	3	19.20
289.0	0.0	0.0	192.0	913.2	895.3	7	14.60

Table 8.6 50 Samples of Dataset 6

Cement (component 1)(kg in a m^3 mixture)	Fly Ash (component 3)(kg in a m^3 mixture)	Water (component 4)(kg in a m^3 mixture)	Superplasticizer (component 5)(kg in a m^3 mixture)	Coarse Aggregate (component 6)(kg in a m^3 mixture)	Fine Aggregate (component 7)(kg in a m^3 mixture)	Age (day)	Concrete compressive strength(MPa, megapascals)
540.0	0.0	162.0	2.5	1040.0	676.0	28	79.99
540.0	0.0	162.0	2.5	1055.0	676.0	28	61.89
475.0	0.0	228.0	0.0	932.0	594.0	28	39.29
475.0	0.0	228.0	0.0	932.0	594.0	270	42.13
475.0	0.0	228.0	0.0	932.0	594.0	90	42.23
380.0	0.0	228.0	0.0	932.0	670.0	180	53.10
349.0	0.0	192.0	0.0	1047.0	806.9	3	15.05
475.0	0.0	228.0	0.0	932.0	594.0	365	41.93
531.3	0.0	141.8	28.2	852.1	893.7	28	56.40
531.3	0.0	141.8	28.2	852.1	893.7	56	58.80
531.3	0.0	141.8	28.2	852.1	893.7	91	59.20
222.4	96.7	189.3	4.5	967.1	870.3	3	11.58
222.4	96.7	189.3	4.5	967.1	870.3	14	24.45
222.4	96.7	189.3	4.5	967.1	870.3	28	24.89
222.4	96.7	189.3	4.5	967.1	870.3	56	29.45
194.7	100.5	165.6	7.5	1006.4	905.9	100	37.34
190.7	125.4	162.1	7.8	1090.0	804.0	3	15.04
212.1	121.6	180.3	5.7	1057.6	779.3	28	24.90
212.1	121.6	180.3	5.7	1057.6	779.3	56	34.20
212.1	121.6	180.3	5.7	1057.6	779.3	100	39.61
230.0	118.3	195.5	4.6	1029.4	758.6	3	10.03
230.0	118.3	195.5	4.6	1029.4	758.6	14	20.08
230.0	118.3	195.5	4.6	1029.4	758.6	28	24.48
230.0	118.3	195.5	4.6	1029.4	758.6	56	31.54

166.1	163.3	176.5	4.5	1058.6	780.1	3	10.76
166.1	163.3	176.5	4.5	1058.6	780.1	14	25.48
166.1	163.3	176.5	4.5	1058.6	780.1	28	21.54
166.1	163.3	176.5	4.5	1058.6	780.1	56	28.63
166.1	163.3	176.5	4.5	1058.6	780.1	100	33.54
238.1	94.1	186.7	7.0	949.9	847.0	3	19.93
238.1	94.1	186.7	7.0	949.9	847.0	14	25.69
238.1	94.1	186.7	7.0	949.9	847.0	28	30.23
238.1	94.1	186.7	7.0	949.9	847.0	56	39.59
238.1	94.1	186.7	7.0	949.9	847.0	100	44.30
250.0	95.7	187.4	5.5	956.9	861.2	3	13.82
250.0	95.7	187.4	5.5	956.9	861.2	14	24.92
250.0	95.7	187.4	5.5	956.9	861.2	28	29.22
250.0	95.7	187.4	5.5	956.9	861.2	56	38.33
250.0	95.7	187.4	5.5	956.9	861.2	100	42.35
212.5	100.4	159.3	8.7	1007.8	903.6	3	13.54
212.5	100.4	159.3	8.7	1007.8	903.6	14	26.31
212.5	100.4	159.3	8.7	1007.8	903.6	28	31.64
212.6	100.4	159.4	10.4	1003.8	903.8	100	47.74
212.0	124.8	159.0	7.8	1085.4	799.5	3	19.52
212.0	124.8	159.0	7.8	1085.4	799.5	14	31.35
231.8	121.6	174.0	6.7	1056.4	778.5	14	26.77
231.8	121.6	174.0	6.7	1056.4	778.5	28	33.73
231.8	121.6	174.0	6.7	1056.4	778.5	56	42.70
251.4	118.3	188.5	6.4	1028.4	757.7	56	36.64
251.4	118.3	188.5	6.4	1028.4	757.7	100	44.21

Table 8.7 50 Samples of Dataset 7

Cement (component 1)(kg in a m^3 mixture)	Blast Furnace Slag (component 2)(kg in a m^3 mixture)	Water (component 4)(kg in a m^3 mixture)	Superplasticizer (component 5)(kg in a m^3 mixture)	Coarse Aggregate (component 6)(kg in a m^3 mixture)	Fine Aggregate (component 7)(kg in a m^3 mixture)	Age (day)	Concrete compressive strength(MPa, megapascals)
540.0	0.0	162.0	2.5	1040.0	676.0	28	79.99
540.0	0.0	162.0	2.5	1055.0	676.0	28	61.89
332.5	142.5	228.0	0.0	932.0	594.0	270	40.27
332.5	142.5	228.0	0.0	932.0	594.0	365	41.05
198.6	132.4	192.0	0.0	978.4	825.5	360	44.30
266.0	114.0	228.0	0.0	932.0	670.0	90	47.03
266.0	114.0	228.0	0.0	932.0	670.0	28	45.85
475.0	0.0	228.0	0.0	932.0	594.0	28	39.29
198.6	132.4	192.0	0.0	978.4	825.5	90	38.07
198.6	132.4	192.0	0.0	978.4	825.5	28	28.02
427.5	47.5	228.0	0.0	932.0	594.0	270	43.01
190.0	190.0	228.0	0.0	932.0	670.0	90	42.33
304.0	76.0	228.0	0.0	932.0	670.0	28	47.81
342.0	38.0	228.0	0.0	932.0	670.0	365	56.14
380.0	95.0	228.0	0.0	932.0	594.0	90	40.56
475.0	0.0	228.0	0.0	932.0	594.0	180	42.62
427.5	47.5	228.0	0.0	932.0	594.0	180	41.84
198.6	132.4	192.0	0.0	978.4	825.5	180	41.72
475.0	0.0	228.0	0.0	932.0	594.0	270	42.13
190.0	190.0	228.0	0.0	932.0	670.0	365	53.69
237.5	237.5	228.0	0.0	932.0	594.0	270	38.41
237.5	237.5	228.0	0.0	932.0	594.0	28	30.08
342.0	38.0	228.0	0.0	932.0	670.0	90	50.46
427.5	47.5	228.0	0.0	932.0	594.0	365	43.70

237.5	237.5	228.0	0.0	932.0	594.0	365	39.00
380.0	0.0	228.0	0.0	932.0	670.0	180	53.10
332.5	142.5	228.0	0.0	932.0	594.0	180	39.78
190.0	190.0	228.0	0.0	932.0	670.0	180	46.93
237.5	237.5	228.0	0.0	932.0	594.0	90	33.12
266.0	114.0	228.0	0.0	932.0	670.0	180	48.70
342.0	38.0	228.0	0.0	932.0	670.0	270	55.06
139.6	209.4	192.0	0.0	1047.0	806.9	360	44.70
425.0	106.3	151.4	18.6	936.0	803.7	3	36.30
286.3	200.9	144.7	11.2	1004.6	803.7	3	24.40
475.0	118.8	181.1	8.9	852.1	781.5	7	55.60
469.0	117.2	137.8	32.2	852.1	840.5	7	54.90
425.0	106.3	153.5	16.5	852.1	887.1	7	49.20
388.6	97.1	157.9	12.1	852.1	925.7	7	34.90
531.3	0.0	141.8	28.2	852.1	893.7	7	46.90
425.0	106.3	153.5	16.5	852.1	887.1	7	49.20
439.0	177.0	186.0	11.1	884.9	707.9	7	56.10
337.9	189.0	174.9	9.5	944.7	755.8	28	49.90
388.6	97.1	157.9	12.1	852.1	925.7	56	55.20
531.3	0.0	141.8	28.2	852.1	893.7	56	58.80
425.0	106.3	153.5	16.5	852.1	887.1	56	64.30
318.8	212.5	155.7	14.3	852.1	880.4	56	66.10
401.8	94.7	147.4	11.4	946.8	852.1	56	73.70
362.6	189.0	164.9	11.6	944.7	755.8	91	79.30
379.5	151.2	153.9	15.9	1134.3	605.0	91	56.50
362.6	189.0	164.9	11.6	944.7	755.8	91	79.30

Table 8.8 50 Samples of Dataset 8

Cement (component 1)(kg in a m^3 mixture)	Blast Furnace Slag (component 2)(kg in a m^3 mixture)	Fly Ash (component 3)(kg in a m^3 mixture)	Water (component 4)(kg in a m^3 mixture)	Superplasticizer (component 5)(kg in a m^3 mixture)	Coarse Aggregate (component 6)(kg in a m^3 mixture)	Fine Aggregate (component 7)(kg in a m^3 mixture)	Age (day)	Concrete compressive strength(MPa, megapascals)
540.0	0.0	0.0	162.0	2.5	1040.0	676.0	28	79.99
540.0	0.0	0.0	162.0	2.5	1055.0	676.0	28	61.89
332.5	142.5	0.0	228.0	0.0	932.0	594.0	270	40.27
332.5	142.5	0.0	228.0	0.0	932.0	594.0	365	41.05
198.6	132.4	0.0	192.0	0.0	978.4	825.5	360	44.30
266.0	114.0	0.0	228.0	0.0	932.0	670.0	90	47.03
304.0	76.0	0.0	228.0	0.0	932.0	670.0	28	47.81
380.0	0.0	0.0	228.0	0.0	932.0	670.0	90	52.91
139.6	209.4	0.0	192.0	0.0	1047.0	806.9	90	39.36
342.0	38.0	0.0	228.0	0.0	932.0	670.0	365	56.14
380.0	95.0	0.0	228.0	0.0	932.0	594.0	90	40.56
139.6	209.4	0.0	192.0	0.0	1047.0	806.9	3	8.06
139.6	209.4	0.0	192.0	0.0	1047.0	806.9	180	44.21
380.0	0.0	0.0	228.0	0.0	932.0	670.0	365	52.52
380.0	0.0	0.0	228.0	0.0	932.0	670.0	270	53.30
475.0	0.0	0.0	228.0	0.0	932.0	594.0	7	38.60
304.0	76.0	0.0	228.0	0.0	932.0	670.0	36	55.26

							5	
266.0	114.0	0.0	228.0	0.0	932.0	670.0	365	52.91
237.5	237.5	0.0	228.0	0.0	932.0	594.0	28	30.08
332.5	142.5	0.0	228.0	0.0	932.0	594.0	90	37.72
475.0	0.0	0.0	228.0	0.0	932.0	594.0	90	42.23
237.5	237.5	0.0	228.0	0.0	932.0	594.0	180	36.25
342.0	38.0	0.0	228.0	0.0	932.0	670.0	90	50.46
427.5	47.5	0.0	228.0	0.0	932.0	594.0	365	43.70
237.5	237.5	0.0	228.0	0.0	932.0	594.0	365	39.00
380.0	0.0	0.0	228.0	0.0	932.0	670.0	180	53.10
427.5	47.5	0.0	228.0	0.0	932.0	594.0	90	41.54
427.5	47.5	0.0	228.0	0.0	932.0	594.0	7	35.08
349.0	0.0	0.0	192.0	0.0	1047.0	806.9	3	15.05
332.5	142.5	0.0	228.0	0.0	932.0	594.0	180	39.78
139.6	209.4	0.0	192.0	0.0	1047.0	806.9	7	14.59
198.6	132.4	0.0	192.0	0.0	978.4	825.5	7	14.64
475.0	0.0	0.0	228.0	0.0	932.0	594.0	365	41.93
198.6	132.4	0.0	192.0	0.0	978.4	825.5	3	9.13
304.0	76.0	0.0	228.0	0.0	932.0	670.0	180	50.95
332.5	142.5	0.0	228.0	0.0	932.0	594.0	28	33.02
190.0	190.0	0.0	228.0	0.0	932.0	670.0	27	50.66

							0	
266.0	114.0	0.0	228.0	0.0	932.0	670.0	180	48.70
342.0	38.0	0.0	228.0	0.0	932.0	670.0	270	55.06
139.6	209.4	0.0	192.0	0.0	1047.0	806.9	360	44.70
332.5	142.5	0.0	228.0	0.0	932.0	594.0	7	30.28
190.0	190.0	0.0	228.0	0.0	932.0	670.0	28	40.86
485.0	0.0	0.0	146.0	0.0	1120.0	800.0	28	71.99
374.0	189.2	0.0	170.1	10.1	926.1	756.7	3	34.40
313.3	262.2	0.0	175.5	8.6	1046.9	611.8	3	28.80
425.0	106.3	0.0	153.5	16.5	852.1	887.1	3	33.40
425.0	106.3	0.0	151.4	18.6	936.0	803.7	3	36.30
323.7	282.8	0.0	183.8	10.3	942.7	659.9	3	28.30
379.5	151.2	0.0	153.9	15.9	1134.3	605.0	3	28.60
362.6	189.0	0.0	164.9	11.6	944.7	755.8	3	35.30

Appendix B. Tables of testing parameters for Model 2

Table 8.9 Parameters testing for Model 2

Test 1: Hidden nodes = 2, momentum = 0

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.8947	0.9150	0.8826	0.8697	0.8577
MAE	3.5464	3.3398	4.2231	5.3121	4.7976
RMSE	5.1782	4.7411	5.8403	6.4966	6.3011
TT	0.52	0.49	0.48	0.52	0.47

Test 2: Hidden nodes = 2, momentum = 0.25

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9007	0.9124	0.9079	0.8584	0.7635
MAE	3.4832	3.4796	4.3330	6.5025	18.5923
RMSE	5.0419	4.8960	5.5677	7.4516	20.1191
TT	0.5	0.47	0.5	0.5	0.47

Test 3: Hidden nodes = 2, momentum = 0.5

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9089	0.9057	0.8422	0.8935	0.2831
MAE	3.3303	3.8415	4.4651	5.3692	12.3925
RMSE	4.8398	5.2725	6.1108	7.1294	15.7901
TT	0.49	0.49	0.48	0.47	0.49

Test 4: Hidden nodes = 2, momentum = 0.75

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9176	0.8870	0.8915	0.6960	-0.2305
MAE	3.1583	4.7141	5.0371	8.2613	27.8475
RMSE	4.5924	6.1993	6.7898	9.6591	29.8496
TT	0.48	0.49	0.48	0.47	0.5

Test 5: Hidden nodes = 3, momentum = 0

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.8913	0.9017	0.8927	0.8938	0.9198
MAE	3.5292	3.1916	3.8185	4.2863	3.2708
RMSE	4.9660	4.7875	5.1597	5.5334	4.4942
TT	0.66	0.67	0.68	0.7	0.64

Test 6: Hidden nodes = 3, momentum = 0.25

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.8983	0.9027	0.8880	0.8342	0.7390
MAE	3.3525	3.2281	4.6016	5.5137	6.9621
RMSE	4.8090	4.7752	5.7101	7.9303	8.5623
TT	0.67	0.66	0.66	0.64	0.66

Test 7: Hidden nodes = 3, momentum = 0.5

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9025	0.9022	0.9233	0.8944	0.3316
MAE	3.2109	3.4393	4.8092	4.9529	13.6598
RMSE	4.7310	4.9325	6.0267	6.4508	16.8670
TT	0.66	0.78	0.66	0.65	0.61

Test 8: Hidden nodes = 3, momentum: = 0.75

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.8969	0.8815	0.9061	0.7760	-0.2570
MAE	3.4411	4.4859	3.8121	6.6068	27.8475
RMSE	4.9512	5.9246	5.2742	8.5125	29.8496
TT	0.65	0.66	0.65	0.65	0.75

Test 9: Hidden nodes = 4, momentum = 0

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.8825	0.9131	0.9069	0.9268	0.9141
MAE	3.6946	3.0150	3.1059	3.0362	3.8048
RMSE	5.1191	4.3984	4.7484	4.1726	4.6344
TT	0.84	0.82	0.84	0.83	0.83

Test 10: Hidden nodes = 4, momentum = 0.25

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.8872	0.9077	0.9098	0.8903	0.2517
MAE	3.6694	3.0994	3.0434	3.8227	10.5972
RMSE	5.0125	4.5330	4.6330	4.9612	14.0669
TT	0.83	0.83	0.83	0.83	0.83

Test 11: Hidden nodes = 4, momentum = 0.5

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.8925	0.8985	0.8810	0.7366	0.3270
MAE	3.6291	3.2846	4.2166	4.5487	10.6464
RMSE	4.9128	4.7511	5.4169	7.7206	13.5994
TT	0.84	0.83	0.83	0.83	0.79

Test 12: Hidden nodes = 4, momentum = 0.75

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9029	0.9098	0.9208	0.7440	0.6604
MAE	3.4847	4.1681	5.2170	6.4193	10.1964
RMSE	4.7988	5.2616	6.2178	8.3268	12.6584
TT	0.84	0.83	0.84	0.81	0.83

Test 13: Hidden nodes = 5, momentum = 0

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.8777	0.9177	0.8959	0.9367	0.8658
MAE	3.8340	2.9101	3.1885	2.4528	5.9052
RMSE	5.2092	4.3820	4.8062	3.8427	7.5386
TT	1.02	1.0	1.0	1.0	0.99

Test 14: Hidden nodes = 5, momentum = 0.25

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.8808	0.9194	0.9039	0.9690	0.7505
MAE	3.7043	2.9241	3.0469	2.2398	6.0634
RMSE	5.1694	4.3037	4.8103	2.8786	7.1472
TT	1.01	0.99	0.99	0.99	0.97

Test 15: Hidden nodes = 5, momentum = 0.5

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.8919	0.9249	0.8652	0.9104	0.3953
MAE	3.4900	2.8335	4.7548	3.3617	13.3152
RMSE	4.9932	4.1170	6.5505	4.5772	16.4282
TT	1.0	0.99	1.01	1.0	0.98

Test 16: Hidden nodes = 5, momentum = 0.75

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9068	0.9018	0.9132	0.7007	0.9345
MAE	3.3330	3.4350	3.4435	8.4421	3.3723
RMSE	4.6744	4.8375	4.7545	9.8090	4.5616
TT	0.99	0.99	1.0	0.98	1.04

Test 17: Hidden nodes = 6, momentum = 0

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.8906	0.9515	0.9178	0.9252	0.9188
MAE	3.5529	2.3292	2.5949	2.6494	4.2623
RMSE	4.9651	3.4281	4.2953	4.1177	4.8560
TT	1.16	1.18	1.17	1.17	1.18

Test 18: Hidden nodes = 6, momentum = 0.25

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.8968	0.9252	0.9096	0.8834	0.7435
MAE	3.3850	3.6624	2.7204	2.9959	7.4641
RMSE	4.8218	5.0171	4.7518	5.8951	9.1404
TT	1.19	1.16	1.16	1.16	1.18

Test 19: Hidden nodes = 6, momentum = 0.5

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.8941	0.9471	0.9263	0.8838	0.7080
MAE	3.4880	2.4966	3.0064	4.1150	14.7202
RMSE	4.8873	3.8466	4.1545	5.7991	23.2614
TT	1.15	1.16	1.16	1.17	1.15

Test 20: Hidden nodes = 6, momentum = 0.75

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9112	0.9391	0.9148	-0.1542	-0.2571
MAE	3.4540	2.8646	3.3110	9.3124	27.8495
RMSE	4.6221	3.9225	4.5223	10.7491	29.8520
TT	1.17	1.18	1.16	1.25	1.16

Appendix C. Tables of testing parameters for Model 3

Table8.10 Parameters testing for Model 3

Test 1: Hidden nodes = 2, momentum = 0

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9151	0.9246	0.9307	0.9296	0.7643
MAE	3.5466	3.5531	3.5948	3.8443	6.3633
RMSE	4.7266	4.4888	4.3155	4.4540	8.6636

TT	**0.52**	**0.47**	**0.48**	**0.49**	**0.48**

Test 2: Hidden nodes = 2, momentum = 0.25

Learning Rate	**0.01**	**0.1**	**0.3**	**0.5**	**0.9**
R	**0.9218**	**0.9258**	**0.9374**	**0.9319**	**0.7456**
MAE	**3.4168**	**3.6132**	**3.6481**	**3.8109**	**6.0130**
RMSE	**4.5471**	**4.4385**	**4.4238**	**4.4400**	**8.1793**
TT	**0.49**	**0.48**	**0.48**	**0.49**	**0.47**

Test 3: Hidden nodes = 2, momentum = 0.5

Learning Rate	**0.01**	**0.1**	**0.3**	**0.5**	**0.9**
R	**0.9296**	**0.9271**	**0.9323**	**0.8448**	**0.5902**
MAE	**3.2731**	**3.7300**	**3.9023**	**5.2440**	**11.7961**
RMSE	**4.3278**	**4.4310**	**4.5745**	**7.3838**	**13.0151**
TT	**0.49**	**0.48**	**0.47**	**0.49**	**0.48**

Test 4: Hidden nodes = 2, momentum = 0.75

Learning Rate	**0.01**	**0.1**	**0.3**	**0.5**	**0.9**
R	**0.9373**	**0.9290**	**0.9136**	**0.6983**	**0.7735**
MAE	**3.1169**	**3.9145**	**5.1526**	**18.3598**	**6.7165**
RMSE	**4.1095**	**4.6690**	**6.2908**	**20.6185**	**8.8574**
TT	**0.48**	**0.47**	**0.47**	**0.47**	**0.48**

Test 5: Hidden nodes = 3, momentum = 0

Learning Rate	**0.01**	**0.1**	**0.3**	**0.5**	**0.9**
R	**0.9489**	**0.9442**	**0.9510**	**0.9222**	**0.7258**
MAE	**2.9444**	**3.2401**	**2.6804**	**3.8037**	**7.4728**
RMSE	**3.7952**	**3.9513**	**3.7193**	**4.6948**	**10.4331**
TT	**0.63**	**0.64**	**0.64**	**0.63**	**0.65**

Test 6: Hidden nodes = 3, momentum = 0.25

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9521	0.9465	0.9462	0.9214	0.7434
MAE	2.8401	3.2394	2.8226	3.8426	8.0265
RMSE	3.6813	3.8956	3.9359	4.5798	10.9608
TT	0.64	0.65	0.65	0.65	0.64

Test 7: Hidden nodes = 3, momentum = 0.5

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9517	0.9557	0.9273	0.9053	0.1108
MAE	2.8363	3.0981	3.1598	4.2424	11.4559
RMSE	3.6774	3.5982	4.5195	5.2560	12.8390
TT	0.64	0.64	0.65	0.63	0.63

Test 8: Hidden nodes = 3, momentum = 0.75

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9500	0.9532	0.9089	0.9081	0.1323
MAE	2.7782	3.2728	5.2774	6.3514	10.9816
RMSE	3.7160	4.4456	6.4441	8.0717	12.4565
TT	0.65	0.64	0.64	0.65	0.88

Test 9: Hidden nodes = 4, momentum = 0

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9567	0.9511	0.9447	0.9403	0.7480
MAE	2.8985	3.0162	3.1622	3.5322	7.5967
RMSE	3.6416	3.7193	3.9746	4.3621	10.2088
TT	0.8	0.8	0.82	0.82	0.82

Test 10: Hidden nodes = 4, momentum = 0.25

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9591	0.9497	0.9457	0.9306	0.5899
MAE	2.9777	3.0481	3.1997	3.3255	13.2838
RMSE	3.5761	3.7672	4.0048	4.4270	15.5897
TT	0.86	0.82	0.83	0.83	0.83

Test 11: Hidden nodes = 4, momentum = 0.5

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9586	0.9499	0.9494	0.9072	0.4777
MAE	2.9823	3.0414	3.2022	4.1238	13.0222
RMSE	3.5564	3.7844	3.9592	4.9642	15.4492
TT	0.84	0.82	0.85	0.82	0.83

Test 12: Hidden nodes = 4, momentum = 0.75

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9571	0.9538	0.8933	0.4064	0.0851
MAE	2.7769	3.1651	4.6102	10.3809	10.9806
RMSE	3.5287	4.0106	6.1486	11.7422	12.4559
TT	0.84	0.83	0.84	0.82	0.83

Test 13: Hidden nodes = 5, momentum = 0

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9508	0.9537	0.9546	0.9529	0.8830
MAE	2.8971	2.8208	3.1850	3.2522	6.3927
RMSE	3.7293	3.6358	3.9366	4.0057	8.0838
TT	1.01	1.0	1.0	1.02	1.13

Test 14: Hidden nodes = 5, momentum = 0.25

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9529	0.9615	0.9294	0.9346	0.5801
MAE	2.8358	2.6309	3.8071	3.6343	13.6490
RMSE	3.6478	3.3049	4.8424	4.3674	15.9199
TT	1.0	1.01	1.0	1.01	0.99

Test 15: Hidden nodes = 5, momentum = 0.5

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9545	0.9645	0.9543	0.9250	0.6598
MAE	3.0450	2.5575	2.9682	3.7586	11.9497
RMSE	3.6308	3.1981	3.9174	4.5314	14.0060
TT	0.98	1.01	1.0	1.0	1.08

Test 16: Hidden nodes = 5, momentum = 0.75

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9655	0.9611	0.8676	0.5661	0.0706
MAE	2.6267	2.9401	4.6520	9.8602	10.9817
RMSE	3.4045	3.5409	5.8971	11.0372	12.4566
TT	1.01	1.0	0.99	1.01	0.96

Test 17: Hidden nodes = 6, momentum = 0

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9531	0.9560	0.9520	0.9603	0.8680
MAE	2.9268	2.8268	2.6708	3.2446	7.0010
RMSE	3.6817	3.5989	3.6554	4.5406	8.9107
TT	1.17	1.16	1.23	1.19	1.17

Test 18: Hidden nodes = 6, momentum = 0.25

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9550	0.9544	0.9117	0.9276	0.6584
MAE	2.9188	2.8048	3.8237	3.5713	7.3742
RMSE	3.6201	3.5982	5.0132	4.4116	9.3582
TT	1.18	1.2	1.16	1.17	1.16

Test 19: Hidden nodes = 6, momentum = 0.5

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9557	0.9583	0.9573	0.9181	0.4959
MAE	3.0416	2.6842	2.7826	3.7402	13.4364
RMSE	3.6436	3.4069	3.6411	4.6934	16.7401
TT	1.19	1.19	1.17	1.21	1.19

Test 20: Hidden nodes = 6, momentum = 0.75

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9616	0.9566	0.9201	0.8283	0.0361
MAE	2.8569	2.5436	5.5999	13.4854	11.1547
RMSE	3.4090	3.6608	7.0229	15.3210	12.5145
TT	1.17	1.28	1.21	1.18	1.22

Appendix D. Tables of testing parameters for Model 4

Table 8.11 Parameters testing for Model 4

Test 1: Hidden nodes = 2, momentum = 0

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9501	0.9503	0.9256	0.9389	0.5860
MAE	3.2978	3.3307	3.9718	3.8876	10.9728
RMSE	4.6596	4.2287	4.9127	4.9463	13.9705

TT	**0.51**	**0.49**	**0.47**	**0.49**	**0.48**

Test 2: Hidden nodes = 2, momentum = 0.25

Learning Rate	**0.01**	**0.1**	**0.3**	**0.5**	**0.9**
R	**0.9564**	**0.9487**	**0.9204**	**0.9154**	**0.5902**
MAE	**3.0535**	**3.4415**	**3.9981**	**4.0901**	**11.5850**
RMSE	**4.3711**	**4.3197**	**4.8395**	**4.9905**	**14.4911**
TT	**0.48**	**0.5**	**0.47**	**0.47**	**0.49**

Test 3: Hidden nodes = 2, momentum = 0.5

Learning Rate	**0.01**	**0.1**	**0.3**	**0.5**	**0.9**
R	**0.9645**	**0.9469**	**0.9147**	**0.9116**	**-0.7602**
MAE	**2.9058**	**3.5141**	**4.1202**	**4.5584**	**15.9443**
RMSE	**3.9480**	**4.4259**	**5.2062**	**5.8092**	**20.0409**
TT	**0.48**	**0.47**	**0.49**	**0.48**	**0.47**

Test 4: Hidden nodes = 2, momentum = 0.75

Learning Rate	**0.01**	**0.1**	**0.3**	**0.5**	**0.9**
R	**0.9690**	**0.9308**	**0.8880**	**-0.6791**	**0.2988**
MAE	**2.7194**	**4.4427**	**8.4566**	**23.9399**	**13.8723**
RMSE	**3.5144**	**6.2169**	**9.7962**	**26.8994**	**17.5802**
TT	**0.48**	**0.5**	**0.47**	**0.47**	**0.75**

Test 5: Hidden nodes = 3, momentum = 0

Learning Rate	**0.01**	**0.1**	**0.3**	**0.5**	**0.9**
R	**0.9766**	**0.9768**	**0.9730**	**0.7976**	**0.6584**
MAE	**2.0385**	**2.2062**	**2.4698**	**5.3383**	**8.3082**
RMSE	**2.7071**	**2.7737**	**3.2861**	**7.6283**	**10.9861**
TT	**0.64**	**0.64**	**0.65**	**0.64**	**0.65**

Test 6: Hidden nodes = 3, momentum = 0.25

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9781	0.9765	0.9724	0.9123	0.6020
MAE	1.9613	2.1975	2.9105	4.8251	12.5916
RMSE	2.5950	2.7839	3.5953	6.2733	15.7982
TT	0.64	0.65	0.65	0.64	0.65

Test 7: Hidden nodes = 3, momentum = 0.5

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9768	0.9759	0.9263	0.6221	-0.3783
MAE	2.0695	2.1716	4.0993	13.1195	14.0251
RMSE	2.6865	2.8140	5.2227	16.0562	17.8948
TT	0.66	0.64	0.64	0.63	0.65

Test 8: Hidden nodes = 3, momentum = 0.75

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9766	0.9751	0.6152	0.4660	0.7004
MAE	2.2803	2.2523	18.8293	28.1950	26.7948
RMSE	2.8351	3.1832	21.0878	30.1211	28.1307
TT	0.65	0.64	0.64	0.64	0.59

Test 9: Hidden nodes = 4, momentum = 0

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9724	0.9772	0.9738	0.9505	0.6923
MAE	2.3641	1.6897	3.1766	3.6475	8.8520
RMSE	3.1094	2.8513	3.8759	4.4735	11.3958
TT	0.8	0.8	0.8	0.81	0.8

Test 10: Hidden nodes = 4, momentum = 0.25

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9725	0.9743	0.9414	0.9119	0.6617
MAE	2.2560	2.4027	4.1267	4.2970	11.7568
RMSE	3.1069	3.4733	5.3820	6.0037	14.4205
TT	0.8	0.81	0.82	0.81	0.81

Test 11: Hidden nodes = 4, momentum = 0.5

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9718	0.9587	0.9623	0.4477	0.6880
MAE	1.8772	3.4735	4.2071	13.3164	14.6002
RMSE	3.0441	4.3253	5.1727	16.7253	18.1973
TT	0.83	0.82	0.81	0.81	0.8

Test 12: Hidden nodes = 4, momentum = 0.75

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9721	0.9705	0.7211	0.2961	-0.3020
MAE	1.9674	3.0409	7.4173	23.9827	17.5568
RMSE	3.0315	4.1006	9.0921	28.3718	21.1437
TT	0.8	0.82	0.81	0.8	0.78

Test 13: Hidden nodes = 5, momentum = 0

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9735	0.9729	0.9636	0.9529	0.6831
MAE	2.3642	2.6897	3.5331	3.3240	8.8941
RMSE	3.0832	3.6155	4.6815	4.1103	11.4708
TT	0.99	0.97	0.97	0.99	0.98

Test 14: Hidden nodes = 5, momentum = 0.25

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9734	0.9750	0.9487	0.9202	0.6973
MAE	2.0556	2.7387	3.6038	5.3550	9.7953
RMSE	3.0094	3.6035	4.5408	6.3894	12.4221
TT	0.95	0.98	0.98	0.99	0.97

Test 15: Hidden nodes = 5, momentum = 0.5

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9699	0.9613	0.9360	0.4710	0.3449
MAE	1.9519	3.0474	3.9784	15.0775	18.2928
RMSE	3.1454	3.9020	5.1468	18.4411	22.6824
TT	0.97	0.96	0.97	0.97	0.97

Test 16: Hidden nodes = 5, momentum = 0.75

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9688	0.9405	0.5507	0.3389	-0.3299
MAE	2.3006	2.8463	11.4623	20.5887	13.8718
RMSE	3.5111	5.1699	13.9899	23.8105	17.5797
TT	0.96	0.97	0.97	0.97	0.99

Test 17: Hidden nodes = 6, momentum = 0

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9737	0.9681	0.9474	0.9423	0.6707
MAE	2.2819	3.3562	4.0780	3.8333	7.9855
RMSE	2.8436	4.2279	5.4032	4.8021	10.7322
TT	1.15	1.16	1.15	1.19	1.18

Test 18: Hidden nodes = 6, momentum = 0.25

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9659	0.9639	0.9524	0.9283	0.6717
MAE	2.5462	2.9309	4.2239	4.4707	8.5818
RMSE	3.4983	3.8627	5.3174	5.8097	11.4385
TT	1.16	1.15	1.16	1.15	1.17

Test 19: Hidden nodes = 6, momentum = 0.5

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9677	0.9725	0.9330	0.5587	0.7185
MAE	2.5206	3.7362	5.4149	15.5499	10.9888
RMSE	3.4519	4.7582	6.8245	18.4284	13.5023
TT	1.15	1.16	1.15	1.16	1.13

Test 20: Hidden nodes = 6, momentum = 0.75

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9657	0.9656	0.5731	0.2491	0.8615
MAE	3.2144	3.8509	8.5282	19.3508	5.7945
RMSE	3.8990	4.9530	10.1723	23.0435	7.8252
TT	1.14	1.16	1.14	1.16	1.09

Appendix E. Tables of testing parameters for Model 5

Table 8.12 Parameters testing for Model 5

Test 1: Hidden nodes = 2, momentum = 0

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.8096	0.8141	0.8765	0.8707	0.7047
MAE	6.2356	6.0359	4.4321	4.8170	9.5024

RMSE	**7.7398**	**7.6451**	**6.3900**	**6.7315**	**11.8038**
TT	**0.52**	**0.49**	**0.48**	**0.5**	**0.49**

Test 2: Hidden nodes = 2, momentum = 0.25

Learning Rate	**0.01**	**0.1**	**0.3**	**0.5**	**0.9**
R	**0.8138**	**0.8136**	**0.8735**	**0.7840**	**0.4756**
MAE	**6.1091**	**6.0508**	**4.3648**	**6.3427**	**12.3273**
RMSE	**7.6541**	**7.6531**	**6.4159**	**8.2421**	**15.7636**
TT	**0.49**	**0.5**	**0.5**	**0.49**	**0.49**

Test 3: Hidden nodes = 2, momentum = 0.5

Learning Rate	**0.01**	**0.1**	**0.3**	**0.5**	**0.9**
R	**0.8167**	**0.7855**	**0.8680**	**0.7822**	**0.6774**
MAE	**5.9830**	**6.5691**	**4.3589**	**6.3765**	**8.3503**
RMSE	**7.5834**	**8.2596**	**6.4993**	**8.1808**	**11.1784**
TT	**0.51**	**0.51**	**0.49**	**0.5**	**0.49**

Test 4: Hidden nodes = 2, momentum = 0.75

Learning Rate	**0.01**	**0.1**	**0.3**	**0.5**	**0.9**
R	**0.8148**	**0.7782**	**0.7744**	**0.8044**	**0**
MAE	**5.8962**	**6.5915**	**6.3397**	**6.4875**	**10.2658**
RMSE	**7.6221**	**8.3898**	**8.3187**	**8.5646**	**13.0368**
TT	**0.49**	**0.5**	**0.49**	**0.49**	**0.48**

Test 5: Hidden nodes = 3, momentum = 0

Learning Rate	**0.01**	**0.1**	**0.3**	**0.5**	**0.9**
R	**0.7936**	**0.8981**	**0.8766**	**0.7666**	**0.7359**
MAE	**6.4639**	**4.5476**	**4.7942**	**6.9506**	**8.7035**
RMSE	**8.0149**	**5.9798**	**6.6257**	**8.7013**	**10.9060**
TT	**0.68**	**0.67**	**0.68**	**0.67**	**0.68**

Test 6: Hidden nodes = 3, momentum = 0.25

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.8007	0.8947	0.7709	0.7795	0.6108
MAE	6.2720	4.6312	6.7763	6.5133	11.5276
RMSE	7.8842	6.1337	8.5101	8.2534	14.3393
TT	0.68	0.65	0.67	0.67	0.67

Test 7: Hidden nodes = 3, momentum = 0.5

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.8061	0.8875	0.7683	0.7443	0.3567
MAE	6.0787	4.8478	6.7155	6.5596	14.2846
RMSE	7.7732	6.4284	8.5577	8.7634	21.4357
TT	0.68	0.68	0.67	0.67	0.67

Test 8: Hidden nodes = 3, momentum = 0.75

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.8246	0.7742	0.7325	0.6929	-0.1199
MAE	5.7634	6.5514	6.6298	8.1063	10.2662
RMSE	7.4202	8.4160	9.0082	11.4645	13.0367
TT	0.66	0.68	0.67	0.66	0.68

Test 9: Hidden nodes = 4, momentum = 0

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9226	0.8733	0.8680	0.8216	0.7328
MAE	3.8278	4.5385	4.7283	6.0878	10.6927
RMSE	5.1185	6.4369	6.7531	7.5806	12.9783
TT	0.84	0.86	0.84	0.85	0.85

Test 10: Hidden nodes = 4, momentum = 0.25

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9429	0.8748	0.8606	0.8040	0.4098
MAE	3.5619	4.5339	4.7990	6.2046	14.2449
RMSE	4.4895	6.4078	6.8958	7.8210	20.6469
TT	0.85	0.88	0.86	0.85	0.83

Test 11: Hidden nodes = 4, momentum = 0.5

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9585	0.8778	0.8048	0.7794	0.4608
MAE	3.2410	4.6053	6.3525	6.3117	9.4950
RMSE	3.8710	6.4105	7.9052	8.2465	12.4485
TT	0.86	0.84	0.85	0.84	0.84

Test 12: Hidden nodes = 4, momentum = 0.75

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9178	0.9312	0.7778	0.7967	-0.3472
MAE	3.9015	4.0666	6.5688	6.5621	10.2659
RMSE	5.3043	4.8468	8.4065	8.6952	13.0368
TT	0.85	0.86	0.84	0.86	0.76

Test 13: Hidden nodes = 5, momentum = 0

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9071	0.9040	0.8362	0.8747	0.7331
MAE	4.7154	4.0686	4.3722	5.0246	10.6622
RMSE	5.7255	5.6363	7.1722	6.4781	12.9522
TT	1.03	1.0	1.02	1.02	1.02

Test 14: Hidden nodes = 5, momentum = 0.25

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9250	0.8934	0.9003	0.9117	0.4768
MAE	3.8654	4.1672	4.5094	4.7976	12.7833
RMSE	5.1037	5.9035	5.7081	5.6554	17.0449
TT	1.02	1.03	1.0	1.05	1.01

Test 15: Hidden nodes = 5, momentum = 0.5

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9255	0.8573	0.8958	0.8466	-0.2458
MAE	3.7282	4.2425	4.6545	5.0105	10.3851
RMSE	5.0582	6.7560	5.8113	7.0351	13.3280
TT	1.03	1.0	1.03	1.02	1.05

Test 16: Hidden nodes = 5, momentum = 0.75

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9050	0.9129	0.8575	0.8549	-0.2894
MAE	3.6332	4.1882	5.8669	5.8296	10.2659
RMSE	5.6053	5.4076	7.2743	7.4809	13.0368
TT	1.03	1.02	1.03	1.03	1.0

Test 17: Hidden nodes = 6, momentum = 0

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9172	0.8703	0.8454	0.8536	0.7282
MAE	3.9325	3.7351	4.5765	4.8584	10.6768
RMSE	5.2640	6.4426	7.5966	6.9500	12.9362
TT	1.21	1.23	1.22	1.21	1.2

Test 18: Hidden nodes = 6, momentum = 0.25

Learning Rate	0.01	0.1	0.3	0.5	0.9

R	**0.9517**	**0.8824**	**0.8846**	**0.8661**	**0.3764**
MAE	**3.1121**	**3.8645**	**4.4182**	**4.7802**	**15.5210**
RMSE	**4.1199**	**6.1791**	**6.7168**	**6.7875**	**28.4648**
TT	**1.21**	**1.22**	**1.21**	**1.25**	**1.2**

Test 19: Hidden nodes = 6, momentum = 0.5

Learning Rate	**0.01**	**0.1**	**0.3**	**0.5**	**0.9**
R	**0.9525**	**0.8959**	**0.9416**	**0.8859**	**0.5094**
MAE	**3.2125**	**4.4503**	**4.0269**	**4.9218**	**11.4662**
RMSE	**4.0603**	**5.8701**	**4.6767**	**6.0510**	**14.4429**
TT	**1.22**	**1.22**	**1.23**	**1.23**	**1.22**

Test 20: Hidden nodes = 6, momentum = 0.75

Learning Rate	**0.01**	**0.1**	**0.3**	**0.5**	**0.9**
R	**0.9446**	**0.8543**	**0.8863**	**0.7978**	**-0.5126**
MAE	**3.5552**	**4.3052**	**5.2195**	**6.4352**	**10.2658**
RMSE	**4.3173**	**7.2242**	**6.1467**	**8.5849**	**13.0368**
TT	**1.22**	**1.22**	**1.22**	**1.22**	**1.13**

Appendix F. Tables of testing parameters for Model 6

Table 8.13 Parameters testing for Model 6

Test 1: Hidden nodes = 2, momentum = 0

Learning Rate	**0.01**	**0.1**	**0.3**	**0.5**	**0.9**
R	**0.8999**	**0.9134**	**0.9159**	**0.9138**	**0.7531**
MAE	**4.2599**	**3.8102**	**3.8276**	**5.1183**	**9.6898**
RMSE	**6.1785**	**5.7194**	**6.1313**	**6.5152**	**12.8284**
TT	**0.53**	**0.50**	**0.51**	**0.5**	**0.5**

Test 2: Hidden nodes = 2, momentum = 0.25

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9028	0.9163	0.9157	0.9093	0.6171
MAE	4.2869	3.7005	4.2436	4.6710	13.1674
RMSE	6.0962	5.6323	6.5003	6.5262	15.5065
TT	0.51	0.51	0.53	0.49	0.53

Test 3: Hidden nodes = 2, momentum = 0.5

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9062	0.9196	0.9098	0.7992	0.6499
MAE	4.3197	3.5213	5.2511	7.8003	22.2702
RMSE	6.0137	5.5610	7.4600	8.8327	25.4132
TT	0.52	0.51	0.52	0.49	0.5

Test 4: Hidden nodes = 2, momentum = 0.75

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9101	0.9197	0.8087	0.9021	-0.0603
MAE	4.2894	3.5117	7.9083	5.1321	21.1271
RMSE	5.8859	5.9487	8.9816	5.9033	24.6844
TT	0.49	0.48	0.49	0.48	0.49

Test 5: Hidden nodes = 3, momentum = 0

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9245	0.8906	0.9160	0.9294	0.8502
MAE	4.0029	4.7399	4.2092	4.1284	9.6358
RMSE	5.4933	6.8146	6.1431	6.1030	11.9360
TT	0.67	0.69	0.69	0.68	0.69

Test 6: Hidden nodes = 3, momentum = 0.25

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9289	0.8893	0.9174	0.9199	0.7044
MAE	3.9390	4.6641	5.1667	4.3078	10.8566
RMSE	5.2802	6.7872	6.4960	5.6035	12.9670
TT	0.69	0.67	0.71	0.69	0.69

Test 7: Hidden nodes = 3, momentum = 0.5

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9351	0.8867	0.9048	0.8530	0.9376
MAE	3.8631	4.4119	5.0045	5.8363	6.4969
RMSE	5.0569	6.7426	6.1954	7.1089	8.0581
TT	0.7	0.71	0.69	0.7	0.69

Test 8: Hidden nodes = 3, momentum = 0.75

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9407	0.9149	0.8710	0.1943	-0.2007
MAE	3.6495	4.7446	5.7616	10.7861	21.1334
RMSE	4.7704	6.1029	7.3233	13.4138	24.6904
TT	0.69	0.69	0.7	0.69	0.69

Test 9: Hidden nodes = 4, momentum = 0

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9273	0.9394	0.9382	0.8874	0.8115
MAE	4.0157	3.5695	3.5074	5.5720	8.0285
RMSE	5.3385	5.0201	5.4281	6.5901	10.2811
TT	0.86	0.85	0.85	0.86	0.86

Test 10: Hidden nodes = 4, momentum = 0.25

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9306	0.9486	0.9001	0.9096	0.2101
MAE	3.9135	3.1400	4.5096	4.6544	18.1402
RMSE	5.2053	4.6189	6.4577	6.1090	27.2915
TT	0.87	0.86	0.87	0.86	0.85

Test 11: Hidden nodes = 4, momentum = 0.5

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9339	0.9310	0.8930	0.8408	0.8800
MAE	3.9176	3.5799	4.6833	6.2153	6.8425
RMSE	5.1286	5.2667	7.0470	7.8165	9.2484
TT	0.86	0.86	0.87	0.85	0.84

Test 12: Hidden nodes = 4, momentum = 0.75

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9350	0.9216	0.8481	0.9163	0.7938
MAE	3.9110	4.0854	6.0114	7.7080	10.48898
RMSE	5.9110	5.5396	7.6114	8.8293	12.4944
TT	0.87	0.87	0.87	0.87	0.78

Test 13: Hidden nodes = 5, momentum = 0

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9292	0.9289	0.9332	0.9053	-0.0193
MAE	3.9254	3.6010	3.5496	8.1110	13.6349
RMSE	5.2617	5.1829	5.4374	9.7651	33.2809
TT	1.04	1.04	1.06	1.04	1.04

Test 14: Hidden nodes = 5, momentum = 0.25

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9320	0.9350	0.9142	0.9048	0.0463
MAE	3.8776	3.7283	4.0412	4.5978	11.4598
RMSE	5.1502	4.8382	5.8728	6.0896	15.6676
TT	1.04	1.05	1.03	1.02	1.04

Test 15: Hidden nodes = 5, momentum = 0.5

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9342	0.9468	0.8560	-0.0688	0.9007
MAE	3.9243	3.8452	4.5471	15.0498	10.7439
RMSE	5.1102	4.6933	7.7163	33.6546	12.6483
TT	1.04	1.03	1.03	1.04	1.01

Test 16: Hidden nodes = 5, momentum = 0.75

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9300	0.9634	0.0230	0.7769	-0.1835
MAE	3.9406	3.3257	12.7752	9.4989	21.1334
RMSE	5.1685	3.9412	31.2413	14.4464	24.6904
TT	1.07	1.02	1.05	1.02	1.06

Test 17: Hidden nodes = 6, momentum = 0

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.8868	0.9269	0.9253	0.9110	0.7256
MAE	5.2061	4.2748	3.2449	4.5067	6.7166
RMSE	6.7959	5.4416	5.5111	5.7120	9.5143
TT	1.2	1.21	1.21	1.22	1.22

Test 18: Hidden nodes = 6, momentum = 0.25

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9115	0.9212	0.9237	0.8888	0.5313
MAE	4.7420	4.7557	3.8592	5.4334	14.4899
RMSE	6.1880	5.7840	5.3766	6.9814	18.6597
TT	1.21	1.22	1.2	1.22	1.2

Test 19: Hidden nodes = 6, momentum = 0.5

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9231	0.9545	0.8627	-0.0668	0.8248
MAE	4.3592	3.5705	4.3512	19.2055	13.5695
RMSE	5.7389	4.4640	7.3810	48.0607	15.5826
TT	1.21	1.22	1.2	1.22	1.21

Test 20: Hidden nodes = 6, momentum = 0.75

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9308	0.9483	0.8700	0.8969	0.0824
MAE	4.0454	3.6750	5.7146	5.5630	13.1281
RMSE	5.4767	4.3429	7.1850	6.9425	16.4154
TT	1.2	1.23	1.22	1.2	1.6

Appendix G. Tables of testing parameters for Model 7

Table 8.14 Parameters testing for Model 7

Test 1: Hidden nodes = 2, momentum = 0

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.8663	0.8632	0.8407	0.8669	0.4452
MAE	6.5435	7.7062	9.1377	8.5056	14.7507

RMSE	**9.4690**	**9.9840**	**11.4269**	**10.5543**	**20.1099**
TT	**0.54**	**0.52**	**0.49**	**0.49**	**0.49**

Test 2: Hidden nodes = 2, momentum = 0.25

Learning Rate	**0.01**	**0.1**	**0.3**	**0.5**	**0.9**
R	**0.8705**	**0.8531**	**0.8287**	**0.8634**	**0.6116**
MAE	**6.4212**	**8.2826**	**9.8836**	**9.3066**	**13.4605**
RMSE	**9.3246**	**10.6314**	**12.1728**	**11.3567**	**18.4477**
TT	**0.5**	**0.5**	**0.49**	**0.48**	**0.5**

Test 3: Hidden nodes = 2, momentum = 0.5

Learning Rate	**0.01**	**0.1**	**0.3**	**0.5**	**0.9**
R	**0.8745**	**0.8364**	**0.8214**	**0.4862**	**0.5106**
MAE	**6.3284**	**9.0401**	**10.2723**	**14.4228**	**20.9321**
RMSE	**9.1988**	**11.4493**	**12.2873**	**18.9284**	**26.1692**
TT	**0.5**	**0.5**	**0.5**	**0.5**	**0.5**

Test 4: Hidden nodes = 2, momentum = 0.75

Learning Rate	**0.01**	**0.1**	**0.3**	**0.5**	**0.9**
R	**0.8775**	**0.8245**	**0.7695**	**-0.2312**	**0**
MAE	**6.2132**	**9.8627**	**13.5592**	**35.1924**	**18.3714**
RMSE	**9.1043**	**12.1518**	**15.8166**	**39.9124**	**2.5908**
TT	**0.5**	**0.49**	**0.53**	**0.47**	**0.49**

Test 5: Hidden nodes = 3, momentum = 0

Learning Rate	**0.01**	**0.1**	**0.3**	**0.5**	**0.9**
R	**0.8941**	**0.9175**	**0.8145**	**0.8566**	**0.6939**
MAE	**5.3763**	**5.3408**	**7.6685**	**9.3407**	**11.7917**
RMSE	**8.4875**	**7.6559**	**10.9428**	**11.6632**	**15.2666**

TT	**0.69**	**0.67**	**0.69**	**0.68**	**0.66**

Test 6: Hidden nodes = 3, momentum = 0.25

Learning Rate	**0.01**	**0.1**	**0.3**	**0.5**	**0.9**
R	**0.8989**	**0.9094**	**0.8093**	**0.8547**	**0.6562**
MAE	**5.2905**	**5.3242**	**8.2695**	**8.5628**	**14.6822**
RMSE	**8.2923**	**7.8406**	**11.2557**	**10.9904**	**18.6647**
TT	**0.69**	**0.68**	**0.68**	**0.68**	**0.69**

Test 7: Hidden nodes = 3, momentum = 0.5

Learning Rate	**0.01**	**0.1**	**0.3**	**0.5**	**0.9**
R	**0.9031**	**0.9025**	**0.8475**	**0.6724**	**0.5462**
MAE	**5.4004**	**5.3831**	**10.0832**	**12.7031**	**19.5521**
RMSE	**8.1606**	**8.1766**	**12.2665**	**16.1073**	**24.2933**
TT	**0.67**	**0.69**	**0.7**	**0.67**	**0.68**

Test 8: Hidden nodes = 3, momentum = 0.75

Learning Rate	**0.01**	**0.1**	**0.3**	**0.5**	**0.9**
R	**0.9070**	**0.8089**	**0.6736**	**0.3676**	**0.2612**
MAE	**5.6108**	**8.3362**	**16.1715**	**35.1915**	**18.3714**
RMSE	**8.1116**	**11.1044**	**19.7993**	**39.9116**	**22.5908**
TT	**0.68**	**0.68**	**0.68**	**0.68**	**0.59**

Test 9: Hidden nodes = 4, momentum = 0

Learning Rate	**0.01**	**0.1**	**0.3**	**0.5**	**0.9**
R	**0.8802**	**0.9292**	**0.8173**	**0.8434**	**0.8959**
MAE	**6.7233**	**4.3526**	**7.7461**	**10.3520**	**7.2441**
RMSE	**9.0381**	**7.1478**	**11.0485**	**12.9364**	**9.4350**

TT	**0.86**	**0.87**	**0.85**	**0.87**	**0.85**

Test 10: Hidden nodes = 4, momentum = 0.25

Learning Rate	**0.01**	**0.1**	**0.3**	**0.5**	**0.9**
R	**0.8855**	**0.9271**	**0.8856**	**0.8553**	**0.6306**
MAE	**6.7923**	**5.7836**	**6.9340**	**8.5182**	**14.8972**
RMSE	**8.8882**	**7.5860**	**8.9283**	**10.7720**	**19.1450**
TT	**0.86**	**0.86**	**0.87**	**0.85**	**0.85**

Test 11: Hidden nodes = 4, momentum = 0.5

Learning Rate	**0.01**	**0.1**	**0.3**	**0.5**	**0.9**
R	**0.8957**	**0.9182**	**0.8516**	**0.8158**	**0.4180**
MAE	**6.5143**	**5.1150**	**10.3250**	**11.2317**	**13.4139**
RMSE	**8.5362**	**7.5181**	**12.7253**	**12.9002**	**19.9895**
TT	**0.86**	**0.86**	**0.86**	**0.86**	**0.86**

Test 12: Hidden nodes = 4, momentum = 0.75

Learning Rate	**0.01**	**0.1**	**0.3**	**0.5**	**0.9**
R	**0.9292**	**0.9172**	**0.7643**	**0.3893**	**-0.1656**
MAE	**5.5589**	**5.5241**	**20.1682**	**35.0094**	**18.3714**
RMSE	**7.1569**	**8.3330**	**23.8903**	**39.7511**	**22.5908**
TT	**0.87**	**0.85**	**0.83**	**0.83**	**0.82**

Test 13: Hidden nodes = 5, momentum = 0

Learning Rate	**0.01**	**0.1**	**0.3**	**0.5**	**0.9**
R	**0.8737**	**0.9285**	**0.9065**	**0.8474**	**0.7914**
MAE	**6.6971**	**4.4501**	**5.0780**	**7.9462**	**9.5235**
RMSE	**9.2606**	**7.2295**	**8.4554**	**10.4424**	**12.5803**
TT	**1.05**	**1.06**	**1.02**	**1.03**	**1.03**

Test 14: Hidden nodes = 5, momentum = 0.25

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.8846	0.9261	0.9256	0.8546	0.5483
MAE	6.2341	4.3783	6.2473	11.0514	16.4979
RMSE	8.8991	7.3431	7.5795	13.6073	23.5646
TT	1.05	1.03	1.04	1.05	10.5

Test 15: Hidden nodes = 5, momentum = 0.5

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.8960	0.9215	0.8993	0.6467	-0.4137
MAE	5.8733	4.3243	7.4489	14.3664	20.9094
RMSE	8.5171	7.5190	9.5031	18.9885	26.5581
TT	1.04	1.03	1.04	1.03	1.03

Test 16: Hidden nodes = 5, momentum = 0.75

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9358	0.9083	0.7365	0.3386	0.4732
MAE	4.8074	4.9094	10.8294	32.0173	17.7845
RMSE	6.8124	8.2800	13.0404	36.6032	26.6810
TT	1.03	1.05	1.03	1.04	1.03

Test 17: Hidden nodes = 6, momentum = 0

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.8688	0.9414	0.8969	0.8552	0.8704
MAE	6.9988	4.7335	5.7155	9.4612	9.8567
RMSE	9.3909	7.1735	9.5908	12.3297	12.5975
TT	1.21	1.21	1.21	1.22	1.21

Test 18: Hidden nodes = 6, momentum = 0.25

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.8685	0.9303	0.8846	0.8218	0.6968
MAE	6.9137	4.6916	6.5631	13.7148	13.5069
RMSE	9.4096	7.1683	9.2239	18.4869	15.9879
TT	1.21	1.21	1.22	1.21	1.22

Test 19: Hidden nodes = 6, momentum = 0.5

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.8728	0.9054	0.8396	0.7020	0.5001
MAE	6.2113	5.2260	7.6254	13.5155	17.6964
RMSE	9.2173	8.1436	11.6874	19.1017	22.1606
TT	1.22	1.22	1.22	1.2	1.22

Test 20: Hidden nodes = 6, momentum = 0.75

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.8810	0.9409	0.4827	0.1755	0.6334
MAE	5.8973	4.609	11.8765	35.1921	15.8521
RMSE	8.9722	6.406	21.1199	39.9122	23.0239
TT	1.23	1.23	1.22	1.2	1.21

Appendix H. Tables of testing parameters for Model 8

Table 8.15 Parameters testing for Model 8

Test 1: Hidden nodes = 2, momentum = 0

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.8882	0.9398	0.9448	0.9383	0.7413
MAE	6.4816	5.2643	7.2302	10.1469	12.1841

RMSE	8.3473	6.5649	8.2841	11.1090	14.8746
TT	0.57	0.55	0.57	0.55	0.56

Test 2: Hidden nodes = 2, momentum = 0.25

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.8889	0.9447	0.9409	0.9323	0.8105
MAE	6.4496	5.2124	7.2480	8.0459	8.7979
RMSE	8.3263	6.3771	8.34	9.5155	10.6554
TT	0.54	0.53	0.55	0.54	0.54

Test 3:Hidden nodes = 2, momentum = 0.5

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.8958	0.9492	0.9297	0.8064	0.7993
MAE	6.3123	5.1556	7.1186	9.5848	13.5152
RMSE	8.0982	6.1640	8.3359	12.4395	15.9949
TT	0.54	0.53	0.55	0.56	0.52

Test 4: Hidden nodes = 2, momentum = 0.75

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9135	0.9236	0.8102	0.2781	0.4388
MAE	5.9269	6.2665	7.2495	25.2034	20.5762
RMSE	7.5071	7.1526	12.1197	32.4034	26.8507
TT	0.57	0.59	0.6	0.76	0.57

Test 5: Hidden nodes = 3, momentum = 0

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.8823	0.9198	0.9198	0.9318	0.6743
MAE	7.2339	5.5125	5.6294	8.3501	11.2530
RMSE	8.5695	7.1347	7.1374	9.5678	14.2259

TT	**0.83**	**0.76**	**0.8**	**0.93**	**0.75**

Test 6: Hidden nodes = 3, momentum = 0.25

Learning Rate	**0.01**	**0.1**	**0.3**	**0.5**	**0.9**
R	**0.8998**	**0.8980**	**0.9218**	**0.9239**	**0.7478**
MAE	**6.7025**	**6.4824**	**5.7487**	**8.8003**	**12.5562**
RMSE	**7.9704**	**8.0267**	**7.2286**	**10.4236**	**14.4120**
TT	**0.95**	**1.05**	**0.82**	**0.82**	**0.8**

Test 7: Hidden nodes = 3, momentum = 0.5

Learning Rate	**0.01**	**0.1**	**0.3**	**0.5**	**0.9**
R	**0.9057**	**0.8931**	**0.9099**	**0.7231**	**0.7377**
MAE	**6.3963**	**6.6506**	**6.9624**	**11.9117**	**11.8853**
RMSE	**7.8055**	**8.1900**	**8.7641**	**14.9117**	**14.2731**
TT	**0.79**	**1.54**	**2.18**	**0.77**	**2.04**

Test 8: Hidden nodes = 3, momentum = 0.75

Learning Rate	**0.01**	**0.1**	**0.3**	**0.5**	**0.9**
R	**0.9115**	**0.8775**	**0.4823**	**0.4671**	**0.5478**
MAE	**6.1697**	**7.2378**	**14.5762**	**36.2297**	**20.5689**
RMSE	**7.6366**	**8.7657**	**30.8547**	**40.2406**	**26.8334**
TT	**1.09**	**0.79**	**0.84**	**0.8**	**0.75**

Test 9 : Hidden nodes = 4, momentum = 0

Learning Rate	**0.01**	**0.1**	**0.3**	**0.5**	**0.9**
R	**0.9039**	**0.9358**	**0.9302**	**0.9368**	**0.5228**
MAE	**6.7358**	**4.5262**	**5.0381**	**6.2262**	**12.4191**
RMSE	**7.8593**	**6.4243**	**7.0244**	**7.2949**	**18.5592**
TT	**1.43**	**2.3**	**1.0**	**2.23**	**1.0**

Test 10: Hidden nodes = 4, momentum = 0.25

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9003	0.9340	0.8977	0.9247	0.7687
MAE	6.8791	4.6258	6.3928	6.2689	18.9040
RMSE	8.0136	6.5272	8.1072	7.8505	23.6058
TT	1.03	0.99	2.22	1.41	1.54

Test 11: Hidden nodes = 4, momentum = 0.5

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.8926	0.9284	0.9201	0.7511	0.7597
MAE	6.8849	4.9773	6.5358	10.9946	12.384
RMSE	8.4898	6.9172	7.7812	15.9441	15.7142
TT	0.98	1.55	0.98	1.05	1.0

Test 12: Hidden nodes = 4, momentum = 0.75

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.8727	0.8973	0.8160	0.5461	-0.2309
MAE	7.7743	6.8302	10.1822	21.9670	20.5798
RMSE	9.5415	8.6018	16.5148	28.3245	26.8554
TT	1.18	48.1861	0.98	1.18	1.11

Test 13: Hidden nodes = 5, momentum = 0

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9278	0.9073	0.9141	0.8885	0.5297
MAE	5.4762	6.8374	6.6419	8.1484	17.2155
RMSE	7.2763	8.2243	7.9177	9.9660	27.7915
TT	1.26	1.56	1.54	1.82	2.19

Test 14: Hidden nodes = 5, momentum = 0.25

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9165	0.8529	0.9058	0.9230	0.7119
MAE	6.1134	8.0845	7.1028	7.4972	16.2696
RMSE	7.8977	10.5937	9.8251	9.5544	24.7129
TT	2.05	1.17	1.55	2.7	1.99

Test 15: Hidden nodes = 5, momentum = 0.5

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9352	0.9162	0.8483	0.8105	0.3348
MAE	5.2283	6.1956	8.5487	12.2318	17.8194
RMSE	6.9345	7.5026	10.5056	14.3665	24.1150
TT	1.84	2.06	1.51	1.54	2.02

Test 16: Hidden nodes = 5, momentum = 0.75

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9331	0.9033	0.8994	0.0506	0.1712
MAE	5.6733	7.1383	9.1078	21.9733	20.5759
RMSE	7.1627	8.7850	10.6120	28.3285	26.8501
TT	1.97	1.16	1.76	1.22	1.68

Test 17: Hidden nodes = 6, momentum = 0

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9069	0.9037	0.9282	0.8205	0.7443
MAE	6.8073	7.0776	5.4681	9.6882	13.7753
RMSE	8.1494	8.8066	6.8795	12.7525	24.2835
TT	1.75	2.73	1.88	2.38	2.68

Test 18: Hidden nodes = 6, momentum = 0.25

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9161	0.8909	0.9164	0.8037	0.7300
MAE	6.1609	7.6182	5.6651	8.6515	16.9769
RMSE	7.6536	9.6269	7.6127	12.7291	23.0919
TT	1.87	2.63	1.38	1.52	1.35

Test 19: Hidden nodes = 6, momentum = 0.5

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.9306	0.8263	0.8878	0.5307	0.6410
MAE	5.8139	8.5078	7.6299	13.2003	13.8648
RMSE	7.0354	10.8618	10.3119	18.7043	17.5864
TT	1.35	1.37	1.36	1.49	1.59

Test 20: Hidden nodes = 6, momentum = 0.75

Learning Rate	0.01	0.1	0.3	0.5	0.9
R	0.8930	0.9428	0.8312	-0.5702	-0.3198
MAE	7.8240	5.5061	18.2939	40.4560	20.5802
RMSE	9.1551	6.4958	30.2939	48.3556	26.8555
TT	1.39	1.39	1.55	1.55	1.44

Appendix I. Table of information for and final results

Table 8.16 Final parameters and evaluation results for each model

No. of Models	Learning Rate	Momentum	No. of hidden layers	No. of hidden nodes	R	MAE	RMSE	TT
Model 1	**0.1**	**0.75**	**1**	**5**	**0.9658**	**2.4902**	**3.5597**	**0.90**
Model 2	**0.5**	**0.25**	**1**	**5**	**0.9690**	**2.2398**	**2.8786**	**0.99**

Model 3	**0.01**	**0.75**	**1**	**5**	**0.9655**	**2.6267**	**3.4045**	**1.01**
Model 4	**0.01**	**0.25**	**1**	**3**	**0.9781**	**1.9613**	**2.5950**	**0.64**
Model 5	**0.01**	**0.5**	**1**	**4**	**0.9585**	**3.2410**	**3.8710**	**0.86**
Model 6	**0.1**	**0.75**	**1**	**5**	**0.9634**	**3.3257**	**3.9412**	**1.02**
Model 7	**0.1**	**0**	**1**	**6**	**0.9414**	**4.7335**	**7.1735**	**1.21**
Model 8	**0.1**	**0.5**	**1**	**2**	**0.9492**	**5.1556**	**6.1640**	**0.53**

Appendix J. Figures of weights and threshold in each model

```
Linear Node 0
    Inputs    Weights
    Threshold    -0.0014189132871085808
    Node 1    -1.7154657036500796
    Node 2    1.9256176235670397
    Node 3    -2.2889560422063284
    Node 4    -0.8935955823217944
    Node 5    1.9531395672677068
Sigmoid Node 1
    Inputs    Weights
    Threshold    -5.0702727684076025
    Attrib Cement(component1)    3.2229644672584636
    Attrib Water(component2)    -0.712578765513904
    Attrib Coarse-Aggregate(component3)    -1.7090670899024003
    Attrib Fine-Aggregate(component4)    -5.760242868034724
    Attrib Age(component5)    2.0458047745270225
Sigmoid Node 2
    Inputs    Weights
    Threshold    -4.803724899889724
    Attrib Cement(component1)    -6.3046403743595105
    Attrib Water(component2)    -0.9883703391780873
    Attrib Coarse-Aggregate(component3)    -1.084020679386968
    Attrib Fine-Aggregate(component4)    -11.931958466894198
    Attrib Age(component5)    1.1106378209358425
Sigmoid Node 3
    Inputs    Weights
    Threshold    -14.033154617473892
    Attrib Cement(component1)    1.7396776059664942
    Attrib Water(component2)    -0.2339419055631619
    Attrib Coarse-Aggregate(component3)    0.7103815499681975
    Attrib Fine-Aggregate(component4)    1.6662027295697497
    Attrib Age(component5)    -12.973985493300917
Sigmoid Node 4
    Inputs    Weights
    Threshold    -1.9843546824646128
    Attrib Cement(component1)    -6.189666938580757
    Attrib Water(component2)    0.8497002783452134
    Attrib Coarse-Aggregate(component3)    -1.5564094152769656
    Attrib Fine-Aggregate(component4)    -3.676161719311089
    Attrib Age(component5)    -0.8390319884287402
Sigmoid Node 5
    Inputs    Weights
    Threshold    -3.4765325697463396
    Attrib Cement(component1)    3.9740569515631177
    Attrib Water(component2)    -5.378175185900631
    Attrib Coarse-Aggregate(component3)    -0.9398574228668345
    Attrib Fine-Aggregate(component4)    -2.664894057277998
    Attrib Age(component5)    0.926934123405688
```

Figure 8.1: Weights and threshold in Model 1

```
Linear Node 0
    Inputs    Weights
    Threshold    0.054894183935341165
    Node 1    1.8086536637531119
    Node 2    -1.8840475716159006
    Node 3    -0.9250905633393068
    Node 4    -1.5322436578357284
    Node 5    0.8451120772517112
Sigmoid Node 1
    Inputs    Weights
    Threshold    2.8325652700525343
    Attrib Cement(component1)    5.173024672454119
    Attrib Fly-Ash(component2)    6.839896091970756
    Attrib Water(component3)    -2.7821210192098356
    Attrib Coarse-Aggregate(component4)    -2.464419339707945
    Attrib Fine-Aggregate(component5)    3.9830273256431763
    Attrib Age(component6)    0.5865910032263066
Sigmoid Node 2
    Inputs    Weights
    Threshold    -1.0378168396942262
    Attrib Cement(component1)    -1.613423806446127
    Attrib Fly-Ash(component2)    0.9905173552214117
    Attrib Water(component3)    -1.3494521211889074
    Attrib Coarse-Aggregate(component4)    -2.0701563198041146
    Attrib Fine-Aggregate(component5)    9.95092693902368E-4
    Attrib Age(component6)    0.1063257755718064
Sigmoid Node 3
    Inputs    Weights
    Threshold    -6.28175619104406
    Attrib Cement(component1)    -2.1517506251431304
    Attrib Fly-Ash(component2)    0.43328963512285434
    Attrib Water(component3)    2.2058493485328308
    Attrib Coarse-Aggregate(component4)    2.7058669355304574
    Attrib Fine-Aggregate(component5)    0.5872539238293014
    Attrib Age(component6)    -5.3912194120031005
Sigmoid Node 4
    Inputs    Weights
    Threshold    -15.624618689595371
    Attrib Cement(component1)    1.1277311422384677
    Attrib Fly-Ash(component2)    6.875873372129792
    Attrib Water(component3)    -1.32480951451281
    Attrib Coarse-Aggregate(component4)    -0.6633154557019605
    Attrib Fine-Aggregate(component5)    0.11581488114002819
    Attrib Age(component6)    -22.12281144541191
Sigmoid Node 5
    Inputs    Weights
    Threshold    -0.16805645359000324
    Attrib Cement(component1)    -5.235792632930763
    Attrib Fly-Ash(component2)    3.245610235029983
    Attrib Water(component3)    -8.21628858672327
    Attrib Coarse-Aggregate(component4)    -1.0441697046986793
    Attrib Fine-Aggregate(component5)    -11.47878002562719
    Attrib Age(component6)    0.7075973856789589
```

Figure8.2: Weights and threshold in Model 2

```
Linear Node 0
    Inputs    Weights
    Threshold    -0.16014382923185963
    Node 1    3.617964995451901
    Node 2    -1.0221240641796911
    Node 3    -1.8050718281187204
    Node 4    -3.463917568816838
    Node 5    1.170175686325494
Sigmoid Node 1
    Inputs    Weights
    Threshold    -0.46298095745176976
    Attrib Cement(component1)    1.1065722893791305
    Attrib Blast-Furnace-Slag(component2)    0.8289619231142987
    Attrib Water(component3)    -0.9660037041994028
    Attrib Coarse-Aggregate(component4)    0.42049505162895245
    Attrib Fine-Aggregate(component5)    0.3107435019284669
    Attrib Age(component6)    -0.22708600128415155
Sigmoid Node 2
    Inputs    Weights
    Threshold    -0.14377638113839059
    Attrib Cement(component1)    1.2586297379859819
    Attrib Blast-Furnace-Slag(component2)    1.898046764324217
    Attrib Water(component3)    -0.7066021210458036
    Attrib Coarse-Aggregate(component4)    0.41862954615231
    Attrib Fine-Aggregate(component5)    -2.2788669421829955
    Attrib Age(component6)    -1.0679366129024135
Sigmoid Node 3
    Inputs    Weights
    Threshold    -1.2561781456852452
    Attrib Cement(component1)    -0.9787601471671572
    Attrib Blast-Furnace-Slag(component2)    -0.45885509285841203
    Attrib Water(component3)    0.9667538137885255
    Attrib Coarse-Aggregate(component4)    0.7222965299542229
    Attrib Fine-Aggregate(component5)    2.324205927857906
    Attrib Age(component6)    -0.9470824404466595
Sigmoid Node 4
    Inputs    Weights
    Threshold    -9.2198457572843
    Attrib Cement(component1)    0.7474385497682018
    Attrib Blast-Furnace-Slag(component2)    0.9215935083960106
    Attrib Water(component3)    -0.28397616993395874
    Attrib Coarse-Aggregate(component4)    0.017110550082197667
    Attrib Fine-Aggregate(component5)    0.40911381635924204
    Attrib Age(component6)    -8.497046043846906
Sigmoid Node 5
    Inputs    Weights
    Threshold    -1.4071683631497527
    Attrib Cement(component1)    -1.137801857293191
    Attrib Blast-Furnace-Slag(component2)    0.04485495546881399
    Attrib Water(component3)    2.125143020560837
    Attrib Coarse-Aggregate(component4)    -0.4954020294829398
    Attrib Fine-Aggregate(component5)    0.5647023184716973
    Attrib Age(component6)    -1.0777493682831407
Class
    Input
    Node 0
```

Figure 8.3: Weights and threshold in Model 3

```
Linear Node 0
    Inputs    Weights
    Threshold    2.127445842331551
    Node 1    -3.9592500026973996
    Node 2    -2.159892384502291
    Node 3    -1.9785424579949136
Sigmoid Node 1
    Inputs    Weights
    Threshold    -7.419457371486786
    Attrib Cement(component1)    0.6409881271184251
    Attrib Water(component2)    0.3722035471528739
    Attrib Superplasticizer(component3)    -0.2027243168122589
    Attrib Coarse-Aggregate(component4)    0.8855583674512059
    Attrib Fine-Aggregate(component5)    0.8247999986376342
    Attrib Age(component6)    -5.545089487745404
Sigmoid Node 2
    Inputs    Weights
    Threshold    -0.37365531199117596
    Attrib Cement(component1)    -1.4647505607738727
    Attrib Water(component2)    -1.5595108269127853
    Attrib Superplasticizer(component3)    1.074283500524466
    Attrib Coarse-Aggregate(component4)    -0.5732510221886931
    Attrib Fine-Aggregate(component5)    -0.3911842598063993
    Attrib Age(component6)    0.022012667715312063
Sigmoid Node 3
    Inputs    Weights
    Threshold    1.0201775157310085
    Attrib Cement(component1)    -1.2422200045952034
    Attrib Water(component2)    1.8041088321543215
    Attrib Superplasticizer(component3)    -1.5573027606540784
    Attrib Coarse-Aggregate(component4)    -1.2390261236452227
    Attrib Fine-Aggregate(component5)    -0.8169898437568693
    Attrib Age(component6)    0.7241850018412982
Class
    Input
    Node 0
```

Figure 8.4: Weights and threshold in Model 4

```
Linear Node 0
    Inputs    Weights
    Threshold    -0.47459835316232807
    Node 1    1.771942091939398
    Node 2    -1.9809223252973827
    Node 3    -3.25466544379295
    Node 4    3.352000226955241
Sigmoid Node 1
    Inputs    Weights
    Threshold    1.0120114901324173
    Attrib Cement(component1)    -1.221204834078431
    Attrib Blast-Furnace-Slag(component2)    -1.5274523924185612
    Attrib Fly-Ash(component3)    0.12592980863573283
    Attrib Water(component4)    -0.15611816844410378
    Attrib Coarse-Aggregate(component5)    0.902213239866004
    Attrib Fine-Aggregate(component6)    1.652524552405703
    Attrib Age(component7)    -0.1365100586899701
Sigmoid Node 2
    Inputs    Weights
    Threshold    -0.9646811244386393
    Attrib Cement(component1)    -0.5277804106869897
    Attrib Blast-Furnace-Slag(component2)    -1.1678561185838108
    Attrib Fly-Ash(component3)    -1.607989431482484
    Attrib Water(component4)    -1.9089649348947058
    Attrib Coarse-Aggregate(component5)    0.03161937371235206
    Attrib Fine-Aggregate(component6)    1.4985980690503544
    Attrib Age(component7)    -0.9583341161346174
Sigmoid Node 3
    Inputs    Weights
    Threshold    -4.091237011567978
    Attrib Cement(component1)    0.3936753655137853
    Attrib Blast-Furnace-Slag(component2)    0.6326688560629284
    Attrib Fly-Ash(component3)    3.253848145786897
    Attrib Water(component4)    -1.0779717211725108
    Attrib Coarse-Aggregate(component5)    0.015108234885585967
    Attrib Fine-Aggregate(component6)    0.07997894957976837
    Attrib Age(component7)    -6.962636754876101
Sigmoid Node 4
    Inputs    Weights
    Threshold    2.065522024062573
    Attrib Cement(component1)    1.122979917147397
    Attrib Blast-Furnace-Slag(component2)    0.7768995984435983
    Attrib Fly-Ash(component3)    2.795304232064349
    Attrib Water(component4)    -2.0491947064317544
    Attrib Coarse-Aggregate(component5)    -0.42413386785578083
    Attrib Fine-Aggregate(component6)    -0.16070672941159314
    Attrib Age(component7)    -0.53405125076359
Class
    Input
    Node 0
```

Figure 8.5: Weights and threshold in Model 5

```
Linear Node 0
    Inputs    Weights
    Threshold    -0.5921461281472398
    Node 1    1.6802419488286984
    Node 2    -1.6674065820233863
    Node 3    -1.3536645174947695
    Node 4    1.0123017052621814
    Node 5    0.638490442389993
Sigmoid Node 1
    Inputs    Weights
    Threshold    -1.3257957252478219
    Attrib Cement(component1)    0.29927672153949114
    Attrib Fly-Ash(component2)    -1.9548476098306817
    Attrib Water(component3)    1.3637173390394175
    Attrib Superplasticizer(component4)    6.9088809118003995
    Attrib Coarse-Aggregate(component5)    -0.4117155430572222
    Attrib Fine-Aggregate(component6)    2.852781703011666
    Attrib Age(component7)    0.2918261049876156
Sigmoid Node 2
    Inputs    Weights
    Threshold    -15.80178451266743
    Attrib Cement(component1)    1.3658929198415284
    Attrib Fly-Ash(component2)    0.9400250162158067
    Attrib Water(component3)    0.6918996714118768
    Attrib Superplasticizer(component4)    -0.6815416349654653
    Attrib Coarse-Aggregate(component5)    0.34116916088149885
    Attrib Fine-Aggregate(component6)    0.9241016840815122
    Attrib Age(component7)    -15.068068397249043
Sigmoid Node 3
    Inputs    Weights
    Threshold    -4.0194009951902565
    Attrib Cement(component1)    0.5116520207092253
    Attrib Fly-Ash(component2)    1.8705867355512864
    Attrib Water(component3)    -0.18054473640841462
    Attrib Superplasticizer(component4)    4.429535671243353
    Attrib Coarse-Aggregate(component5)    -0.46800898408619385
    Attrib Fine-Aggregate(component6)    0.20189939241236463
    Attrib Age(component7)    -4.393640119805373
Sigmoid Node 4
    Inputs    Weights
    Threshold    -2.545833077872557
    Attrib Cement(component1)    6.532318236417699
    Attrib Fly-Ash(component2)    6.492871493181739
    Attrib Water(component3)    -3.1611210408301122
    Attrib Superplasticizer(component4)    -3.2974511618701237
    Attrib Coarse-Aggregate(component5)    -0.015923103839274217
    Attrib Fine-Aggregate(component6)    0.027311630005257246
    Attrib Age(component7)    -0.08535983153373554
Sigmoid Node 5
    Inputs    Weights
    Threshold    8.18206952325063
    Attrib Cement(component1)    7.278168731581168
    Attrib Fly-Ash(component2)    1.2465697173277852
    Attrib Water(component3)    1.3778793880485412
    Attrib Superplasticizer(component4)    6.565791008225748
    Attrib Coarse-Aggregate(component5)    1.197207295197689
    Attrib Fine-Aggregate(component6)    1.8008848430801958
    Attrib Age(component7)    0.7641494251812395
Class
    Input
    Node 0
```

Figure 8.6: Weights and threshold in Model 6

```
Linear Node 0
    Inputs    Weights
    Threshold    1.0286620872836427
    Node 1    0.8409197250023231
    Node 2    -1.308701749675082
    Node 3    -1.6801875311060421
    Node 4    -1.537657843384131
    Node 5    -3.3306134532475964
    Node 6    1.4606668102880174
Sigmoid Node 1
    Inputs    Weights
    Threshold    1.69352203718179
    Attrib Cement(component1)    2.406463826381346
    Attrib Blast-Furnace-Slag(component2)    3.4420839130946357
    Attrib Water(component3)    4.402878999542579
    Attrib Superplasticizer(component4)    2.0781726598749404
    Attrib Coarse-Aggregate(component5)    -2.5314429669133247
    Attrib Fine-Aggregate(component6)    0.5414974088832528
    Attrib Age(component7)    0.9292849950862613
Sigmoid Node 2
    Inputs    Weights
    Threshold    0.5703546952035797
    Attrib Cement(component1)    1.531881009360674
    Attrib Blast-Furnace-Slag(component2)    1.5881676539022258
    Attrib Water(component3)    4.302467652164625
    Attrib Superplasticizer(component4)    -0.35813158614313095
    Attrib Coarse-Aggregate(component5)    2.0920547008881547
    Attrib Fine-Aggregate(component6)    0.20658417688798095
    Attrib Age(component7)    0.7956915425744204
Sigmoid Node 3
    Inputs    Weights
    Threshold    -2.3160469710159464
    Attrib Cement(component1)    -3.0082957934380645
    Attrib Blast-Furnace-Slag(component2)    -2.146069317264187
    Attrib Water(component3)    -2.3092512864624286
    Attrib Superplasticizer(component4)    0.21465207689268237
    Attrib Coarse-Aggregate(component5)    -1.825385791429848
    Attrib Fine-Aggregate(component6)    -0.2118939755440297
    Attrib Age(component7)    -0.701952118846712
Sigmoid Node 4
    Inputs    Weights
    Threshold    1.7389868289766792
    Attrib Cement(component1)    1.7792196278596242
    Attrib Blast-Furnace-Slag(component2)    2.456902733153759
    Attrib Water(component3)    -0.8640012059993144
    Attrib Superplasticizer(component4)    3.0577300506237988
    Attrib Coarse-Aggregate(component5)    -2.373310198631559
    Attrib Fine-Aggregate(component6)    -2.869136435622514
    Attrib Age(component7)    -0.03593086120030961
Sigmoid Node 5
    Inputs    Weights
    Threshold    -10.754705675350118
    Attrib Cement(component1)    0.3805903383700793
    Attrib Blast-Furnace-Slag(component2)    0.6448263915079901
    Attrib Water(component3)    -0.8069098977031398
    Attrib Superplasticizer(component4)    -0.7408052684610003
    Attrib Coarse-Aggregate(component5)    0.14814560321034287
    Attrib Fine-Aggregate(component6)    -0.10892480931181488
    Attrib Age(component7)    -9.333795732207665
Sigmoid Node 6
    Inputs    Weights
    Threshold    1.7552980535647609
    Attrib Cement(component1)    2.175293072481637
    Attrib Blast-Furnace-Slag(component2)    2.418000249049408
    Attrib Water(component3)    -2.361666775751841
    Attrib Superplasticizer(component4)    2.36100268574002
    Attrib Coarse-Aggregate(component5)    -0.12303538198739258
    Attrib Fine-Aggregate(component6)    -2.3698496935604574
    Attrib Age(component7)    0.0063017603589959215
Class
    Input
    Node 0
```

Figure 8.7: Weights and threshold in Model 7

```
Linear Node 0
    Inputs    Weights
    Threshold    -1.6951411799780498
    Node 1    1.6479907722788283
    Node 2    0.9286085637426785
Sigmoid Node 1
    Inputs    Weights
    Threshold    8.474244896066635
    Attrib Cement(component1)    0.2346391716004374
    Attrib Blast-Furnace-Slag(component2)    -0.2873721856520531
    Attrib Fly-Ash(component3)    -0.6982353141843244
    Attrib Water(component4)    0.9928964303120608
    Attrib Superplasticizer(component5)    0.6414406597890544
    Attrib Coarse-Aggregate(component6)    0.11475092936243375
    Attrib Fine-Aggregate(component7)    -0.34575470156095023
    Attrib Age(component8)    9.186174339687996
Sigmoid Node 2
    Inputs    Weights
    Threshold    2.0029682560619952
    Attrib Cement(component1)    7.980654293675804
    Attrib Blast-Furnace-Slag(component2)    5.616308377041829
    Attrib Fly-Ash(component3)    5.231555127916812
    Attrib Water(component4)    -1.9401119071324833
    Attrib Superplasticizer(component5)    -1.2028663422302028
    Attrib Coarse-Aggregate(component6)    1.2936154440072098
    Attrib Fine-Aggregate(component7)    4.405524416265276
    Attrib Age(component8)    -4.283733387362535
Class
    Input
    Node 0
```

Figure 8.8: Weights and threshold in Model 8