

**MODELLING BEHAVIOURAL DESPAIR
WITH
ARTIFICIAL NEURAL NETWORK**

**M.Sc.Thesis by
Aynur ZÖNGÖR
(504031400)**

**Date of submission : 25 December 2006
Date of defence examination: 21 February 2007**

**Supervisor (Chairman): Prof. Dr. İnci Çilesiz (İTÜ.)
Prof. Dr. Reşit Canbeyli (BÜ.)
Members of the Examining Committee Prof.Dr. Ethem Alpaydın (BÜ.)
Prof.Dr. Cem Say (BÜ.)
Assoc. Prof.Dr. Neslihan Şengör (İTÜ.)**

MAY 2007

ACKNOWLEDGEMENTS

I would like to thank my supervisors Prof.Dr. Reşit Canbeyli and Prof.Dr.İnci Çilesiz for their support.I would like to special thank Assoc.Prof.Dr. Neslihan Şengör. This work would not be possible without her generous support.Additionally, I would like to thank Prof.Dr. Ethem Alpaydın for giving advice and adjustments.

May 2007

Aynur Zöngör

TABLE OF CONTENTS

ACKNOWLEDGEMENTS	i
TABLE OF CONTENTS	iii
ABBREVIATIONS	v
LIST OF TABLES	vi
LIST OF FIGURES	viii
SUMMARY	xi
ÖZET	xii
1. INTRODUCTION	1
2. BEHAVIORAL DESPAIR	5
2.1 An Overview of Animal Depression Models	6
2.1.1 The Learned Helplessness Model	7
2.1.2 The Behavioral Despair Model	8
2.2 Porsolt Test (The Forced Swimming Test)	9
3. ARTIFICIAL NEURAL NETWORKS (ANN)	10
3.1 A Neuron Structure	11
3.2 Network Architectures	12
3.3 Multilayer Perceptron (MLP)	12
3.4 Back Propagation Algorithm.	13
4. PREDICTION OF BEHAVIORAL DESPAIR RATIO AND DURATION OF IMMOBILITY WITH MLP	16
4.1 Prediction BD Ratio with Immobility and Head-Shake Behaviors	17
4.1.1 Statistical analysis to determine the number of neurons and	18
4.1.2 Analyzing the effect of initial conditions for training and	
test set results	22
4.1.3 Results	35
4.2. A General Model for Prediction of BD Ratio with Seasonal data of Rat Groups	37
4.2.1 Modeling general seasonal behavior with immobility values at different minutes	37
4.2.2 Determining the effect of initial weights and neuron number	49
4.2.3 Results	64
4.3 A Different General ANN Model to Predict the Behavioral Despair Ratio for All Seasons	65
4.3.1 Determining the effect of initial weights and neuron number	65
4.3.2 Results	75
4.4 A Model for Prediction of Immobility Behavior During PST2 (Second Day of Porsolt Test)	79
4.4.1 A model for prediction of PST2 considering only third, fourth and fifth minutes immobility	79
4.4.2 A model for prediction of PST2 considering each of the first five minutes immobility	80
4.4.3 A model for prediction of PST2 considering each of the first ten	

minutes immobility	81
5. DISCUSSION AND CONCLUSION	83
REFERENCES	86
APPENDIX A.	88
BIOGRAPHY	94

ABBREVIATIONS

ANN	: Artificial Neural Network
MLP	: Multilayer Perceptron
BD	: Behavioral Despair Ratio
BPA	: Back-Propagation Algorithm
dogsh_kth	: Number of wet-dog-shake's during k th minute of PST1
imm1_kth	: Duration of immobility during k th minute of PST1
imm2	: Duration of immobility during PST2

LIST OF TABLES

	<u>Page Number</u>
Table 4.1.1 Mean squared error and standard deviation, training set results for Data set of 17 rats	20
Table 4.1.2 General Mean squared error and standard deviation.....	21
Table 4.1.3 Mean squared error and standard deviation, training set results for Data set of 16 rats	21
Table 4.1.4 General Mean squared error and standard deviation.....	22
Table 4.1.5 Mean squared error and standard deviation, training set results for Data set of 16 rats	24
Table 4.1.6 Mean squared error and standard deviation, test set results for Data set of 16 rats	26
Table 4.1.7 Mean squared error and standard deviation, training set results for Data set of 16 rats	28
Table 4.1.8 Mean squared error and standard deviation, test set results for Data set of 16 rats	30
Table 4.1.9 General test results for three hidden neurons	31
Table 4.1.10 General test results for ten hidden neurons	31
Table 4.1.11 Mean squared error and standard deviation, training set results for Data set of 16 rats	33
Table 4.1.12 Mean squared error and standard deviation, test set results for Data set of 16 rats	34
Table 4.1.13 General Results for third simulation	34
Table 4.2.1 Mean squared error and standard deviation, Training set results for 3 and 5 hidden neurons.....	51
Table 4.2.2 Mean squared error and standard deviation, Training set results for 7 and 10 hidden neurons.....	53
Table 4.2.3 Mean squared error and standard deviation, Test set results for 3 and 5 hidden neurons	56
Table 4.2.4 Mean squared error and standard deviation, Test set results for 7 and 10 hidden neurons	58
Table 4.3.1 Mean squared error and standard deviation, Training set	

	results for 3 and 5 hidden neurons	67
Table 4.3.2	Mean squared error and standard deviation, Training set results for 7 and 10 hidden neurons	69
Table 4.3.3	Mean squared error and standard deviation, Test set results for 3 and 5 hidden neurons	71
Table 4.3.4	Mean squared error and standard deviation, Test set results for 7 and 10 hidden neurons	73
Table A.1	Outputs in Training Phase for Data Set1	89
Table A.2	Outputs in Test Phase for Data Set1	89
Table A.3	Outputs in Training Phase for Data Set2	89
Table A.4	Outputs in Test Phase for Data Set2	89
Table A.5	Outputs in Training Phase for Data Set3	89
Table A.6	Outputs in Test Phase for Data Set3	89
Table A.7	Normalized Inputs of the ANN in Training Phase	90
Table A.8	Outputs of the ANN in Training Phase	90
Table A.9	Outputs of the MLP in Training Phase	90
Table A.10	Outputs of the MLP in Training Phase	90
Table A.11	Outputs of the MLP in Training Phase	90
Table A.12	Outputs of the MLP in Training Phase	91
Table A.13	Outputs of the MLP in Training Phase	91
Table A.14	Outputs of the MLP in Training Phase	91
Table A.15	Outputs of the MLP in Training Phase	92
Table A.16	Outputs of the MLP in Test Phase	92
Table A.17	Outputs of the MLP in Training Phase	92
Table A.18	Outputs of the MLP in Test Phase	92
Table A.19	Outputs of the MLP in Training Phase	93

LIST OF FIGURES

	<u>Page Number</u>
Figure 2.1 An Illustration Porsolt Test	7
Figure 3.1 Structure of Neuron	11
Figure 3.2 Multilayer Perceptron	13
Figure 4.1 Structure of using MLP	16
Figure 4.2 Graph of Sigmoid Activation function	17
Figure 4.1.1 Outputs of Training Phase	35
Figure 4.1.2 Outputs of Test Phase	35
Figure 4.1.3 Outputs of Training Phase	36
Figure 4.1.4 Outputs of Test Phase	36
Figure 4.1.5 Outputs of Training Phase	36
Figure 4.1.6 Outputs of Test Phase	36
Figure 4.2.1 Desired and ANN Output	38
Figure 4.2.2 Mean Error of Training Process	38
Figure 4.2.3 Desired and ANN Output of August	38
Figure 4.2.4 Desired and ANN Output of February	38
Figure 4.2.5 Desired and ANN Output of May	39
Figure 4.2.6 Desired and ANN Output of November	39
Figure 4.2.7 Desired and ANN Output	40
Figure 4.2.8 Mean Error of Training Process	40
Figure 4.2.9 Desired and ANN Output of August	40
Figure 4.2.10 Desired and ANN Output of February	40
Figure 4.2.11 Desired and ANN Output of May	40
Figure 4.2.12 Desired and ANN Output of November	40
Figure 4.2.13 Desired and ANN Output	41
Figure 4.2.14 Mean Error of Training Process	41
Figure 4.2.15 Desired and ANN Output of August	42
Figure 4.2.16 Desired and ANN Output of February	42
Figure 4.2.17 Desired and ANN Output of May	42
Figure 4.2.18 Desired and ANN Output of November	42
Figure 4.2.19 Desired and ANN Output	43

Figure 4.2.20	Mean Error of Training Process	43
Figure 4.2.21	Desired and ANN Output of August	43
Figure 4.2.22	Desired and ANN Output of February	43
Figure 4.2.23	Desired and ANN Output of May	44
Figure 4.2.24	Desired and ANN Output of November	44
Figure 4.2.25	Desired and ANN Output	44
Figure 4.2.26	Mean Error of Training Process	44
Figure 4.2.27	Desired and ANN Output of August	45
Figure 4.2.28	Desired and ANN Output of February	45
Figure 4.2.29	Desired and ANN Output of May	45
Figure 4.2.30	Desired and ANN Output of November	45
Figure 4.2.31	Desired and ANN Output	46
Figure 4.2.32	Mean Error of Training Process	46
Figure 4.2.33	Desired and ANN Output of August	46
Figure 4.2.34	Desired and ANN Output of February	46
Figure 4.2.35	Desired and ANN Output of May	46
Figure 4.2.36	Desired and ANN Output of November	46
Figure 4.2.37	Desired and ANN Output	47
Figure 4.2.38	Mean Error of Training Process	47
Figure 4.2.39	Desired and ANN Output of August	47
Figure 4.2.40	Desired and ANN Output of February	47
Figure 4.2.41	Desired and ANN Output of May	48
Figure 4.2.42	Desired and ANN Output of November	48
Figure 4.2.43	Outputs of Training Phase	59
Figure 4.2.44	Test results for Spring.....	60
Figure 4.2.45	Test results for Autumn.....	60
Figure 4.2.46	Test results for Summer.....	60
Figure 4.2.47	Test results for Winter.....	60
Figure 4.2.48	Outputs of Training Phase	61
Figure 4.2.49	Test results for Spring.....	61
Figure 4.2.50	Test results for Autumn.....	61
Figure 4.2.51	Test results for Summer.....	61
Figure 4.2.52	Test results for Winter.....	61
Figure 4.2.53	Outputs of Training Phase	62
Figure 4.2.54	Test results for Spring.....	62
Figure 4.2.55	Test results for Autumn.....	62

Figure 4.2.56	Test results for Summer.....	62
Figure 4.2.57	Test results for Winter.....	62
Figure 4.2.58	Outputs of Training Phase	63
Figure 4.2.59	Test results for Spring.....	63
Figure 4.2.60	Test results for Autumn.....	63
Figure 4.2.61	Test results for Summer.....	63
Figure 4.2.62	Test results for Winter.....	63
Figure 4.3.1	Outputs of Training Phase	75
Figure 4.3.2	Outputs of Test Phase	75
Figure 4.3.3	Outputs of Training Phase	76
Figure 4.3.4	Outputs of Test Phase	76
Figure 4.3.5	Outputs of Training Phase	77
Figure 4.3.6	Outputs of Test Phase	77
Figure 4.3.7	Outputs of Training Phase	78
Figure 4.3.8	Outputs of Test Phase	78
Figure 4.4.1	Desired and ANN Output	79
Figure 4.4.2	Mean Error of Training Process	79
Figure 4.4.3	Desired and ANN Output of Test Phase	80
Figure 4.4.4	Desired and ANN Output	80
Figure 4.4.5	Mean Error of Training Process	80
Figure 4.4.6	Desired and ANN Output of Test Phase	81
Figure 4.4.7	Desired and ANN Output	81
Figure 4.4.8	Mean Error of Training Process	81
Figure 4.4.8	Desired and ANN Output of Test Phase	82

MODELLING BEHAVIOURAL DESPAIR WITH ARTIFICIAL NEURAL NETWORK

SUMMARY

Behavioral despair test, which is often known as Porsolt test, is used many areas of psychology and medicine sector. Also, behavioral despair used as a model to understand depression mechanism. Neural Network is also a popular and powerful approach/tool, which is used to solve various problems in different disciplines. In this thesis study, we totally achieved four different behavioral despair modeling studies with artificial neural networks. In first modeling study, duration of immobility and wet-dog-shake behaviors of 17 rats are considered. Prediction of behavioral despair is tried to make real using these two behaviors. In second and third modeling study, 37 rats that belonged to different seasons were used. For two models, only immobility behavior was considered. Main aim of these two models was searching seasonal effects and impact of immobility in different minutes on behavioral despair. Porsolt test has been done in two consecutive days. At the last modeling study, it was tried to predict immobility at second day by considering the data of immobility in first day.

In order to achieve this study, data from the research team carrying out their studies under the supervision of Reşit Canbeyli at Psychology Department in Boğaziçi University were obtained. The forced swimming test, i.e., Porsolt test results are used as data set, where the Porsolt test is carried out with 17 rats and 37 rats at different seasons and four different groups are considered.

DAVRANIŞSAL ÇARESİZLİĞİN YAPAY SINIR AĞLARI İLE MODELLENMESİ

ÖZET

Davranışsal çaresizlik testi, bilinen adıyla Porsolt testi, psikoloji alanında ve ilaç sanayinde yaygın olarak kullanılan bir yöntemdir. Ayrıca bir model olarak depresyonun anlaşılması için kullanılmaktadır. Yapay sinir ağları ise günümüzde çeşitli disiplinlerde değişik problemleri çözmek için kullanılan gözde ve güçlü bir yaklaşımdır. Bu tez çalışmasında yapay sinir ağları kullanılarak davranışsal çaresizlikle ilgili toplam dört farklı modelleme çalışması gerçekleştirildi. İlk modelleme çalışmasında, 17 sıçanın hareketsizlik ve kafa sallama davranışları göz önüne alındı. Bu iki davranış yapay sinir ağlarına uygulanarak öğrenilmiş çaresizlik önceden öngörüldü. Bu modelleme çalışmasında hareketsizlik ve kafa sallama davranışının öğrenilmiş çaresizlik üzerinde etkili olduğu görüldü. İkinci ve üçüncü modelleme çalışmasında ise farklı mevsimler ait 37 sıçanın verisi de kullanıldı. Bu iki modelleme çalışmasında sadece hareketsizlik davranışı gözönüne alındı. Bu iki modelin amacı değişik dakikalardaki hareketsizlik sürelerinin ve mevsimsel faktörlerin davranışsal çaresizlik davranışına etkisini belirlemektir. Porsolt testi arka arkaya iki günde gerçekleşen bir test çalışmasıdır. Son modelleme çalışmasında, birinci günün hareketsizlik sürelerinden yola çıkılarak, hayvanın ikinci günkü hareketsizlik süresi tahmin edilmeye çalışıldı.

Bu çalışmayı gerçekleştirmek için Reşit Canbeyli yönetiminde Boğaziçi Psikoloji Labrotuvarında yapılan deneylerde elde edilen veriler kullanılmıştır. 17 sıçan grubu ve farklı mevsim gruplarında oluşan 37 sıçan grupları ile gerçekleştirilen Porsolt testi, diğer bir adıyla zorlanmış yüzme testi sonuçları veri olarak alındı.

1. INTRODUCTION

In a few decades, there will arise more need of interdisciplinary studies to understand human behavior and thought. To investigate the nature and origins of thought and behavior, cognitive science which embraces philosophy, neuroscience and psychology is dedicated precisely to the study of how the mind works. Cognitive Science searches answers to the fundamental questions about the mental processes and it does this in a dynamic, interdisciplinary approach. In cognitive science, scientists in several fields work together to develop theories of mind based on complex representations and computational procedures.

Cognitive scientists use methods, perspectives and expertises from a number of different disciplines. Despite differences in methods of investigation, cognitive scientists have a commitment to a set of ideas: that the mind is a function of the brain, that thinking is a kind of computation. Indeed, cognitive science tries to unify various divergent theoretical ideas which researches in different fields bring to the study of mind and brain.

Psychology which is defined as an academic and applied discipline involving the *scientific study of mental processes and behavior* is a fundamental component of cognitive science. Psychology also refers to the application of this knowledge to various aspects of human activity, including problems of individuals' daily lives and the treatment of mental illness. Psychology differs from neurophysiology and neuroscience as it is primarily concerned with the interaction of mental processes and behavior on a systemic level, while neuroscience deals more about the biological or neural processes themselves [5].

In this sense, cognitive psychologists are commonly interested in theorizing and computational modeling. Meanwhile, they also benefit from experimentations with human participants and animals as a primary method. Physiological experiments are crucial for cognitive science to understand the nature of mind and mental processes in many ways. Considering only thought experiments in deriving hypothesis about human behavior could give rise to absurd results, so real experimental set ups are important. Computational models and physiological experiments evolve together improving each other. This is best stated as following, "*To address the crucial*

questions about the nature of mind, the physiological experiments need to be interpretable within a theoretical framework that postulates mental representation and procedures. One of best ways of developing theoretical frameworks is by forming and testing computational models intended to be analogues to mental operations.” [5].

Cognitive science has several approaches which are broadly classified as symbolic, connectionist and dynamic systems. Cognitive science has a fundamental hypothesis is expressed in terms representational structures in the mind and computational procedures that operate on those structures. The fundamental hypothesis incorporates diversity approaches despite disagreement about the nature of the representations and computations that constitute thinking.

The central hypothesis of cognitive science is that thinking can best be understood in terms of representational structures in the mind and computational procedures that operate on those structures. While there is much disagreement about the nature of the representations and computations that constitute thinking, the central hypothesis is general enough to encompass the current range of thinking in cognitive science. According to connectionist theories, thought can be modeled using artificial neural networks (ANN) [5,6].

The brain consists of simple processing units linked to each other by excitatory and inhibitory connections. Processing knowledge not only occurs between the units via their connections, but also modifying the connections plays an important role especially in forming the plasticity property of the brain. The activation and learning which is spread to the units produces the behavior. Connectionist approach is inspired by this actuality and tries to capture this property of the brain in giving rise to mind [7].

Connectionist networks consist of nodes capable of processing simple nonlinear functions and their connections. These models are powerful tools especially to understand the psychological processes that involve satisfaction of parallel constraints. Similarly processes appear in vision, decision making, action selection, and meaning making in language comprehension. These models can be used to simulate learning by methods that include Hebbian, reinforcement and error back-propagation learning [4].

Relation between connection models and psychological results has been evolved by simulations of various psychological experiments, although these models are only rough approximations to the actual neural networks. Nowadays, more realistic

computational models of the brain are realized by using more realistic neurons and simulating the interactions between different areas [6,8].

In cognitive science, generally ANN structures are used to obtain the simulations or models of behavior and executive functions [9-12]. For testing, the outputs of models are often compared with subject behavior. Some models based on the ANN structures are developed to explain the neural substrates underlying in psychological behavior as fear conditioning [9,10].

In this thesis, artificial neural network is used for behavioral analysis of Porsolt tests. The Porsolt test (also called the behavioral despair test or forced swimming test) is a test used to measure the effect of antidepressant drugs on the behavior of laboratory animals (typically rats or rat). Porsolt swim test is also the most commonly used test for assessment of depression in animal models [3].

In this study, we tried to develop an ANN model which aimed to predict ratio (Behavioral Despair Ratio). BD ratio is the quotient of the immobility in first day to the immobility in second day and thought as diagnostic parameter to forecast depression risk. To achieve this objective, three different ANN simulations are realized. Multilayer perceptron structure (MLP) is preferred because the problems considered in this thesis correspond to functional approximation problem. These simulations are summarized briefly in the following.

In the first simulation, ANN is used to predict BD ratio given immobility and head-shake behaviors. There is just one testing group consisting data of 17 rats. Data of ten rats are used as training set and data of rest are used as test set. In [3], the same study has been done with same data set. The difference is in that thesis ADALINE network is used and the results are tested using the training set. The better results obtained in this thesis are due to the ANN structure used as MLP is a better function approximation than ADALINE. Other two simulations used only immobility behavior to predict BD ratio. In second simulation, a general model is developed to predict of BD ratio with different rat groups in different seasons. The aim of this simulation is to demonstrate the effect of seasons on behavioral despair. Data of previous rat group is used as training set. In the test phase, updated MLP is tested for each rat groups which belonged to different seasons. In third simulation, data of four rat groups which were obtained in different seasons are used as test and training sets.

Additionally, ANN model is obtained which tries to forecast immobility in the second day of test by using immobility in the first day. Again like in the previous experiments, the MLP's structure is used for this simulation.

2. BEHAVIORAL DESPAIR

To explain behavioral despair, we need first to define “Learned Helplessness”. Learned helplessness is described as a psychological condition in which a human or animal has learned to believe that there is no chance of improving the situation after exposed to uncontrolled, unpleasant and/or harmful situations. When similar unpleasant situation repeats again, he does not show any reaction and stays passive and encounters damage. This is due to fact that, he thinks that he has no control over the ongoing situation and whatever he does is useless. Learned helplessness may also occur in everyday situation when environment in which people experience different events make them feel they have no control over what is happening or they really have no control over what is going on. In most cases, when people experience learned helplessness, they have a tendency to give up easily or fail more often at somewhat easier tasks [13].

Learned helplessness is accepted as a phenomenon which has three main parts: contingency, cognition, and behavior, defined below. Contingency means the uncontrollability of the condition. Cognition means the characteristic thoughts about their situation. Behavior refers to performance about what subjects will do in the uncontrolled situation [14].

The learned helplessness model is applied in the areas. First, it is a valuable method to test the effects of stress on the immune system. Second, learning problems are observed in animals after exposure to uncontrolled aversive events, so learned helplessness can be used as method to search the interference in learning. Also, due to similarities between learned helplessness and depressive symptoms (such as learning deficits, slowed response, and passivity) researches have used learned helplessness to develop a model of reactive depression [15].

In this thesis, we were exclusively interested in the relation between depression and learned helplessness. Learned helplessness offered a model to explain human depression. It has been argued that the model has validity because there is similarity between the behavioral characteristics of learned-helplessness in animals and signs of depression in humans. For example, learned helpless animals exhibit

loss of appetite and weight, decreased locomotors activity, and poor performance in both tempting and aversively motivated tasks. These behavioral characteristics of learned helpless animals are considered equivalent to loss of appetite and weight, psychomotor retardation demonstrated by depressed humans (DSM-IV). [13]

Behavioral despair paradigm is a variant of learned helplessness phenomenon. In this modified paradigm rats forced to swim in an inescapable container on consecutive two days. Generally, the behavioral despair is accepted as milder version of learned helplessness [15].

2.1 An Overview of Animal Depression Models

An animal model is defined as setting up experiments with animals to mimic a disorder. Generally, animal studies are carried to fulfill two main purposes. The first aim is to understand the animal species and to learn more about their behavior. The other purpose is pursued for the ultimate purpose of learning about human species, as most of the experiments are harmful and/or involve some kind of unpleasant experience to subjects, so researchers preferred animals instead of human subjects in these experiments [15].

In psychology, animal models are mainly used to study four aims. First aim is to mimic a psychiatric syndrome in its entirety. In this case, homology between the behavior of the affected animal and the syndrome must be constituted. The second aim is systematically studying the effects of potential therapeutic treatments. In this case, only the efficacy of known therapeutic agents is searched to develop new pharmacotherapy. The third one is simulating only specific signs or symptoms of (related to) psychopathologic conditions. Last usage of animal models is studying more theoretically hypotheses [16].

We briefly discuss main validating criterions of animal models. Validation criteria are described as general standards to the evaluation of any model. There are many different types of validity criteria: predictive; construct; concurrent or convergent; discriminate; etiological; and face validity. Which one of them is used depend on the desired purpose of the test. In the following, main validity criterions; *predictive validity* means the ability of a test to predict an interesting behavior. *Construct validity* is most commonly defined as the theoretical rationale of the model. *Face validity* refers to the degree of resemblance between the animal model and the clinical condition [13,16].

The etiology of depression contains numerous risk factors which have psychological, social, and biological effects. But in general, major animal models of depression assume a single causal factor. The attempt to simulate depression using a single psychological or behavioral manipulation may be counterproductive, since a few of the identified etiological factors appear sufficiently potent to precipitate depression in an otherwise risk-free individual. Indeed, the diversity of animal models of depression may prove to be a particularly valuable source of theoretical insights. It follows that while there may be many good reasons to reject certain models, based on wide different etiological assumptions; these differences should be seen as complementary rather than as competitors. In the following sections, aspects of learned helplessness model and the behavioral despair model will be discussed [13].

2.1.1 The Learned Helplessness Model

In 1967, Seligman and co-workers accidentally discovered helplessness phenomena while studying the effects of inescapable shock on active avoidance learning in dogs.

Seligman had studied classical conditioning the simplest mechanism whereby organisms learn about relations between stimuli and come to alter their behavior. Seligman applied several inescapable shocks (UCS) paired with a conditioned stimulus (CS) to dogs in cage. Then these dogs were replaced in another cage where they could escape by jumping over a barrier. Consequently, most of the dogs couldn't learn that avoiding shock was possible by jumping over a barrier [17].

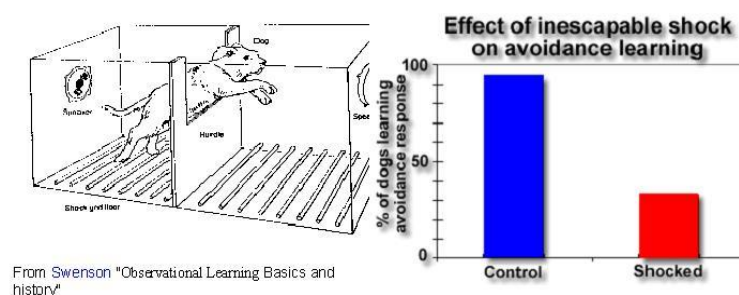


Figure 2.1: An Illustration Porsolt Test

By these studies, Seligman showed that after exposure to inescapable shock; even in an avoidance situation the ability to learn was degraded. Seligman used the term *“Learned Helplessness”* to describe this phenomenon [17].

The central idea in learned helplessness is based on the observation that exposure to uncontrollable stress produces performance deficits in subsequent learning tasks that are not seen in subjects exposed to identical stressors that are under the subjects' control. This marks a sharp change in the direction of previous studies of learning which had focused on learning in controllable situations [17].

The theoretical rationale of learned helplessness as a model of depression has usually been assumed to lie within the ‘learned helplessness hypothesis of depression’ (Seligman, 1975) and consists, in effect, of three assertions: that animals exposed to uncontrollable aversive events do become helpless; that a similar state is induced in people by uncontrollability; and that helplessness in people is the central symptom of depression [13].

2.1.2 The Behavioral Despair Model

A variant of the learned helplessness model is the behavioral despair paradigm. In this model, rat or rats are forced to swim in a confined space. The animal initially swims around and attempts to escape, and eventually assumes an immobile posture. On the subsequent test, the latency to immobility is decreased. In a modification of this paradigm, animals are first exposed to uncontrollable stress before the swim test. These paradigms are conceptually similar to the learned helplessness paradigm in assuming that after uncontrollable stress, animals have learned to "despair" (i.e., learned helplessness). As such, the behavioral despair model involves conceptually similar inducing conditions and dependent variables, and thus has the potential of providing convergent support for the construct of learned helplessness [13,14].

Generally ‘behavioral despair’ is considered as a milder version of learned helplessness. In actuality, both phenomena seem to share similar physiological substrates. Learned helplessness includes unavoidable painful stimuli, on other hand in behavioral despair test the subjects are exposed to an unpleasant situation that is inescapable. The animal model of behavioral despair is called as Porsolt test detailed following section [13].

2.2 Porsolt Test (The Forced Swimming Test)

The Porsolt test or forced swimming test is also called “The behavioral despair Test”. The Porsolt test, which is a standard method used to measure the effect of antidepressant drugs on the behavior of laboratory animals, is utilized typically to induce behavioral despair in rats or rat.

Rats are subjected to two trials during which they are forced to swim in an acrylic cylinder filled with water, and from which they can not escape. The first trial lasts 15 minutes. The rats struggle and try to escape in the first few minutes, but later they cease to move and only keep their head above the water. In first trial, rats learn that there is no possibility to escape from the unpleasant situation. Then, after 24-hours, a second trial is performed that lasts 5 minutes. Rats show immobility most of 5 minutes in last trail. The time that the test animal spends without moving (duration of immobility) during the second trial is measured. This immobility time is shown to be decreased by antidepressants [15].

Behavioral despair ratio (BD) is defined as the ratio of the durations of immobilization measured in first five minutes of first trial day (PST1), and during immobilization of second trial day (PST2). BD parameter is considered critical for determining degree risk of depression. In this study, we consider two behaviors of rats which are potentially crucial to predict degree of BD. First is immobility that is motionless of rats in the water. The less important second behavior is wet-dog-shake, where the animal twitches its head in a fashion similar to the trembling of a wet dog.

$$BD = \frac{imm_2}{imm_{1-5}} \quad (2.2)$$

In this thesis, results of the forced Porsolt test is evaluated and simulated with ANN. This simulation studies are equivalent to studying function approximation problem with ANN structures. We realized four different simulations by using ANN. First three of them we tried to predict BD parameters. In the last simulation, durations of immobility in second day are tried to be predicted. In first simulation, immobility and wet-dog-shake behaviors of 17 rats are considered. In the rest of the simulations, only immobility behaviors of 37 rats which belonged to groups tested in different seasons are considered. Details of each simulation are explained in the fourth chapter.

3. ARTIFICIAL NEURAL NETWORKS (ANN)

In its most general form, a neural network is a machine that is designed to model the way in which the brain performs a particular task or function of interest; the network is usually implemented by using electronic components or is simulated in software on a digital computer. Performance of the ANN is determined by interconnection of simple processing units termed as 'neurons'.

A definition of a neural network is following:

"A neural network is a massively parallel distributed processor made up of simple processing units which has a propensity for storing experiential knowledge and making it available for use. It resembles brain in two ways.

- 1. Knowledge is acquired by the network from its environment through a learning process.*
- 2. Interneuron connection strengths, known as synaptic weights, are used to store the acquired knowledge."* [4].

Superiority of its computing power gets from its *massively parallel distributed structure* and *ability of learning*. Learning ability provide generalization that refers to the neural network producing reasonable outputs for inputs not encountered during training.

Main advantageous properties and capabilities of ANN:

- Nonlinearity
- Input-Output Mapping
- Adaptability
- Evidential Response
- Contextual Information
- Fault Tolerance
- Neurobiological Analogy

3.1 A Neuron Structure

A neuron is fundamental unit processing of ANN. Main function of its: Each input is multiplied by weights and these results are added then applied limiter. So that output of neuron is gotten.

It consist three basic elements:

Weight: This refers to synapses or connecting links.

Adder: A linear combiner summing input signals, weighted by the respective synapses of the neuron.

An Activation Function: it is called limiter.

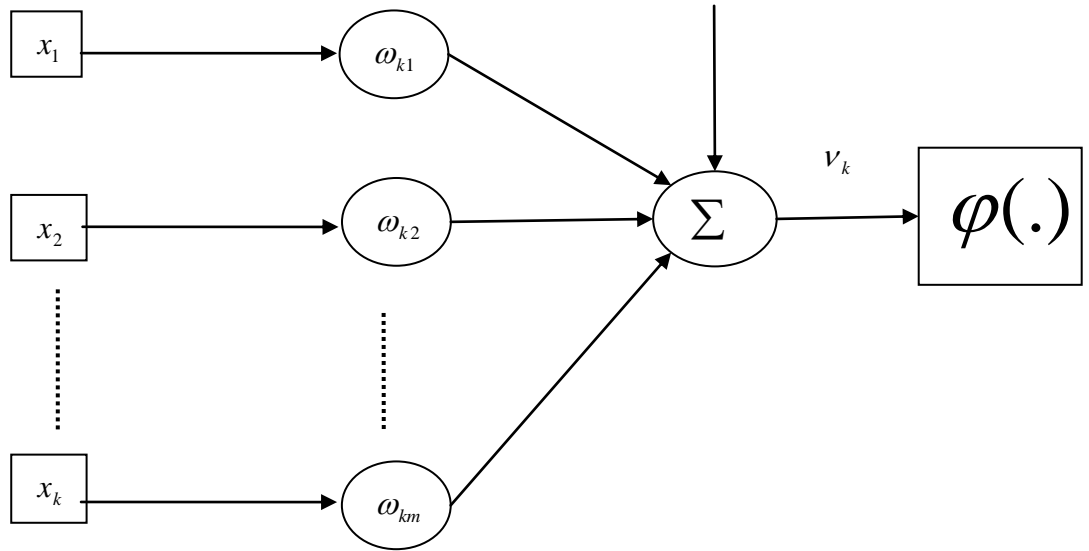


Figure 3.1: Structure of a Neuron

The structure of a neuron is expressed in mathematical terms as following:

$x_1, x_2, \dots, x_j \rightarrow$ Inputs of the Neuron

$w_{k1}, w_{k2}, w_{km} \rightarrow$ Weights

$$v_k = \sum_{j=1}^m w_{kj} \cdot x_j \rightarrow y_k = \phi(v_k + b_k) \rightarrow \text{Output of the Neuron} \quad (3.1)$$

First computational model of neurons is developed by McColloc-Pitts in 1946. By time, neuron structure is improved as above structure.

3.2 Network Architectures

There are various neural network structures. Generally, network architectures can be divided into three different classes. First of them is single-layer feed-forward networks which basically have one input layer and one output layer. Input layer of source nodes only projects from environment onto output layer and don't perform any computation. Output layer of neurons have computation ability. The second class is multilayer feedforward class which has one or more hidden layers differently. They contain computations neurons which are called hidden neurons. The network is enabled to extract higher-order statistics by adding one or more hidden layers. Last class is recurrent networks which have at least one feedback loop differently. The presence of feedback loop provides nonlinearity and profound impact on the learning capability.

In our thesis, we preferred multilayer feed forward networks as network architecture. Because the prediction of BD or prediction immobility in second day problems correspond function approximation in neural network area. Multilayer perceptron is one of best and fit network structures for function approximation application.

3.3 Multilayer Perceptron (MLP)

Well known architecture of ANN is multilayer perceptrons (MLP) which is member of multilayer feed forward networks. In this network structure, a set of neurons get together and then constitute typical network architecture that includes a input layer, one or more hidden layer and an output layer. Input layer contains sensory units (source nodes), other layer consists of computation nodes. The input signals propagate through the network in a forward direction, on a layer-by-layer basis.

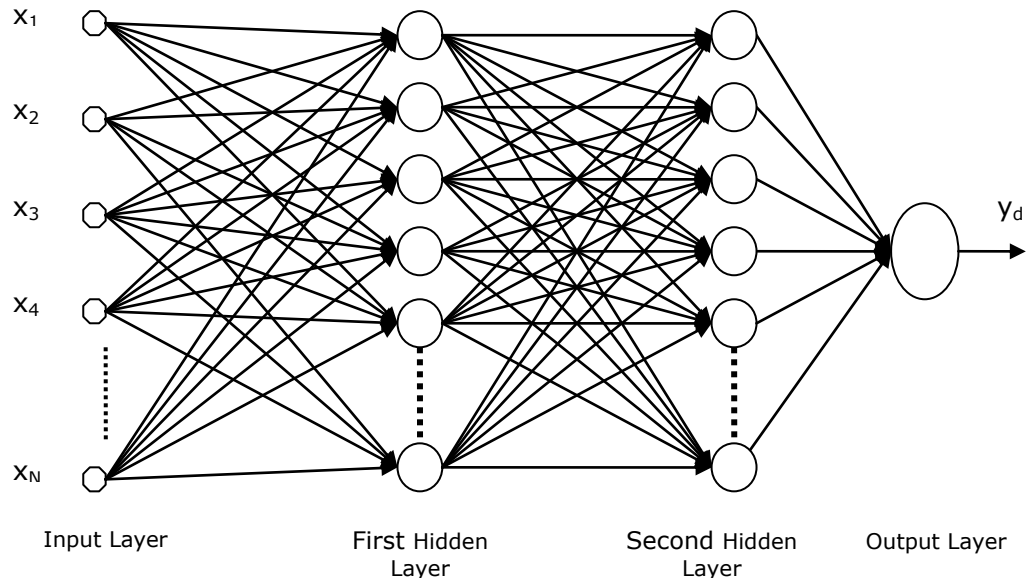


Figure 3.2: Multilayer Perceptron

Training of the MLP is in supervised manner with *error-back propagation algorithm* which is a highly popular. MLP is applied to solve various and sophisticated problem

3.4 Back Propagation Algorithm.

Back Propagation Algorithm is a supervised learning technique which is widely used for training feed forward multilayer neural networks and also known as Delta Rule. Back Propagation Algorithm consists of two passes named forward pass and backward pass based on error-correction rule. In forward pass, inputs applied to the neurons and signal is then propagated through the network in forward direction, on a layer by layer base. Synaptic weights of network are unchanged. In backward pass, error propagated through backward direction and synaptic weights adjusted to minimize error that is the difference between desired output and MLP output [4].

We explain back propagation algorithm step by step in following:

1. Step : Calculate output of each neuron of the ANN

Calculating each neuron's output layer by layer in forward direction.

i: layer index

j: neuron index

$$v_j^{(l)}(n) = \sum_{i=0}^m w_{ij}^{(l)}(n) y^{(l-1)}_i(n) \quad (3.4a)$$

$$y^{(l)}_j = \varphi(v^{(l)}_j(n)) \quad (3.4b)$$

2. Step: Estimate error

Calculating error at output layer according to desired and ANN output.

$$e_i(k) = y_d(k) - y_i(k) \rightarrow \text{error of } i.\text{neuron at output in } k.\text{iteration} \quad (3.4c)$$

3. Step: Estimate Local Gradient.

After finding error, local gradients are calculated due to errors.

Local Gradients:

Gradient of i.neuron at output layer of the ANN :

$$\delta_i^o(k) = e_i^o(k) \varphi'(v_i^o(k)) \quad (3.4d)$$

Gradient of j.neuron at hidden layer of the ANN

$$\delta_j^{(l)}(k) = \varphi'(v_j^{(l)}(k)) \sum_i \delta_i^{(l+1)}(k) w_{ij}^{(l+1)}(k) \quad (3.4e)$$

4. Step: Maintenance of weights

Weights of the ANN are updated according to learning rate, local gradient and input.

$$w_{ij}(k+1) = w_{ij}(k) + \Delta w_{ij} \quad (3.4f)$$

$$w_{ij}^{(l)}(k+1) = w_{ij}^{(l)}(k) + \eta \delta_j^{(l)} y_i^{(l-1)}(k) \quad (3.4g)$$

$$\Delta w_{ij} = -\eta \frac{\partial E}{\partial w_{ij}} = \eta \delta_j^{(l)} y_i^{(l-1)}(k) \rightarrow \text{updating term for weight.} \quad (3.4h)$$

5. Step:

These steps are repeated for all training set.

4. PREDICTION OF BEHAVIORAL DESPAIR RATIO AND DURATION OF IMMOBILITY WITH MLP

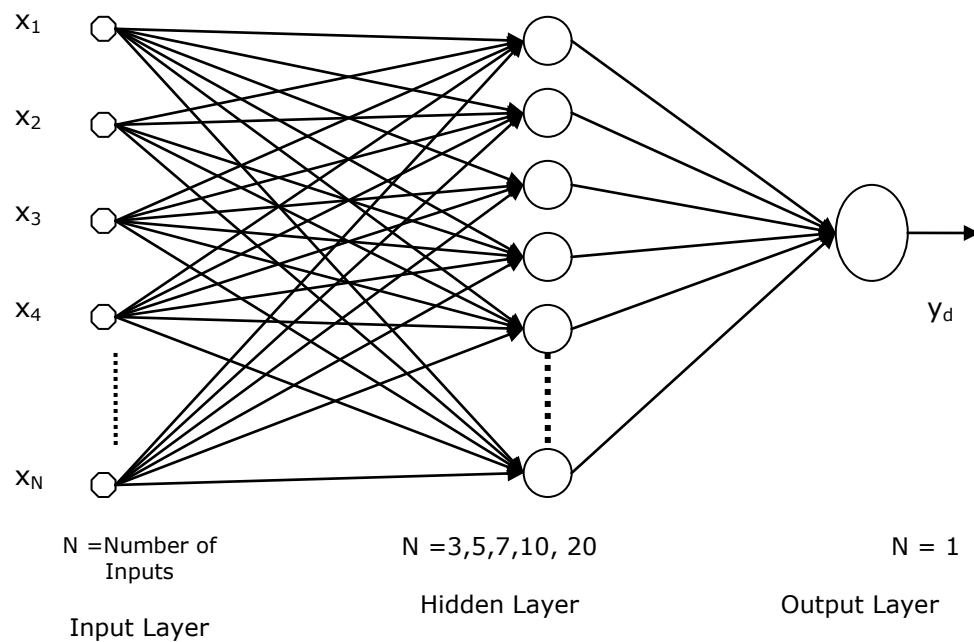


Figure 4.1: Structure of using MLP

Our Multilayer Perceptron (MLP) consists of three layers, which are input layer, hidden layer and output layer. Input layer that contains sensory neurons (nodes), hidden and output layer which are consists of computational neurons. Number of Input and hidden neurons is variant according but, in output layer there is only one neuron.

The learning method using is online and supervised. In the online method, weights are changed by applying each input. According supervised method, ANN tries to minimize cost function that is mean square error. The learning algorithm which is used is Back Propagation algorithm.

In this study, the MLP model fulfills two operations mainly prediction of BD ratio and prediction duration of immobility behaviors in second day. First, predicting BD ratio which is considered as a distinctive sign of depression by applying various size inputs. For this goal, three different MLP models are realized and evaluated. Each model contains vary number of distinctive simulations. Main difference between simulations is training sets that contain changeable number of inputs. In first MLP model, training set contains data about duration of immobility in first day and also differently number of head shakes behaviors. In other two MLP model which also try to predict BD ratio, training sets contain only data of immobility behaviors and size of training patterns vary for each simulations. The second fulfill of MLP models is prediction duration immobility in second day (PST2) with considering immobility in first day (PST1) as inputs. Last fourth MLP model is designed for this aim.

The MLP can have different activation functions. Which activation function is selected depend on structure of problem that is want to solve. In this work, the activation function of the ANN is chosen as sigmoid.

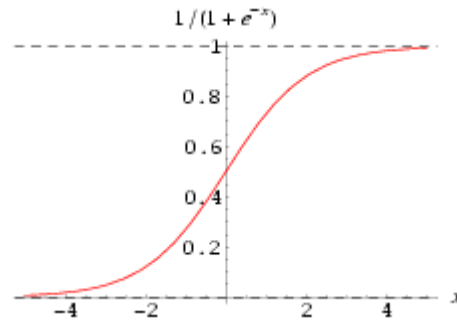


Figure 4.2: Graph of Sigmoid Activation function

4.1 Prediction BD Ratio with Immobility and Head-Shake Behaviors

Our aim in this section is to predict BD ratio which is claimed critical parameter in forecasting depression. Especially, the effect of immobility and head-shake behaviors in depression will be considered and what is their effect on BD ratio will be investigated. Maybe, depression risk can be prevented by prediction of BD ratio. Inputs of the MLP are third minute of immobilization in PST1 ($imm_{1,3}$) and average number of wet-dog-shakes in the fifth and the sixth minutes on PST1 ($dogsh_{5,5}$).

In the thesis of İ.Oruç [3], entirely same study which we are explained in this section has done and used same data set in both test and training phase. The architecture

has been used in her work is, a single-layer network, ADALINE which is simple and inadequate structure for function approximation problems. Because of this, her results are not as precise as our results. In the below given subsections, all simulation results are explained with tables and graphs, successively. Discussions for each simulation are also given.

4.1.1 Statistical analysis to determine the number of neurons and training/test sets

In this section, statistical analysis is carried out in order to determine the effect of hidden layer neuron number and the training/test set combinations on the performance of artificial neural network (ANN) model. Considering these results, the number of hidden layer neurons and the training/test set discrimination will be determined.

In this analysis, as inputs only the duration of immobility and the number of head shakes are considered. The role of ANN is to determine a relation between these inputs and BD ratio, thus the BD ratio corresponds to the output of ANN. Behavioral despair (BD) ratio is equal to the duration of immobility during the first five minutes of the first day to the immobility during the first five minutes of the second day. It has been argued that BD ratio is a measure of depression and bad mood [3]. Thus, it is important to determine how BD ratio is predicted or which parameters are operative, so there would be a chance to forecast depression and/or bad mood and take some precautions.

In the data set obtained from Porsolt experiments carried out in Canbeyli's Physiology Laboratory, there are values for 17 rats. In this group, the BD value of the 14th rat is very much different that the values of the other rat. While BD ratio is 10.94 for the 14th rat, the BD ratio of other rat changes in the interval of 4.25 to 0.2.

The ANN structure used is multilayer perceptron and the activation used is of sigmoid type where the function takes value between -1 and 1. Thus while implementing the ANN structure, below given function is used.

$$y_{MLP}(x) = \tanh(ax) \quad (4.1a)$$

Since the activation function takes values between -1 and 1 in order to prevent the effect of saturation regions which would cause poor learning phase, the input and output values of the data set are normalized to the interval -0.9 and 0.9.

$$x_{norm} = \frac{(x - x_{min}) \times 1.8}{x_{max}} - 0.9 \quad (4.1b)$$

When the normalization procedure is completed considering all 17 rat, except the 14th rat with extraordinary BD ratio all the other values are negative and the normalized set does not have a normal distribution. In order to disregard the negative effect of the 14th rat during training phase, a second data set has been constructed considering only 16 rat data. So, with two different data set composed of 16 and 17 rat, two similar statistical analysis are carried out separately.

Since multilayer perceptron (MLP) is used as ANN structure, for training backpropagation algorithm is considered. The MLP structure is composed of three layers one being the hidden layer, the others is input and output layers. While the input layer has two neurons, different number of the hidden layer neurons are tried to understand the effect of neuron number on the performance of the MLP for the considered problem. So four different numbers, namely, three, five, seven and ten are considered. The activation function is sigmoid type as mentioned above. The learning rate is 0.5 and the constant of argument (a) is taken unity. In each simulation 5000 iteration is carried out and the weights are updated in online mode and test phase is realized considering the weights obtained at the end of 5000 iterations.

For data set of 17 rat, 17 different case are considered where in each four different number of neurons (3,5,7,10) are taken into consideration to investigate the effect of neuron number, thus 68 different simulations are carried out. In each case one rat data is taken as test set while other 16 compose the training set. During the training phase of each simulation mentioned above mean of squared error and standard deviation of the 16 data are calculated.

$$e_{avg} = \frac{1}{16} \frac{1}{2} \sum_1^{16} (y_d - y_{MLP})^2 \quad (4.1c)$$

In Table 4.1.1, the simulation index Sim1 means that the data for the first rat is taken as test set, while other 16 rat data is used in training set, similarly Sim2 means the data for the second rat is used as test set while others are used during training phase. In Table 4.1.1 a summary of the results obtained for training set is given. In each simulation, once 5000 iteration is ended the value of weights are kept and used for the training set to calculate the mean square error. The effect of different number of neurons in hidden layer and different sets of rat can be followed from the Table 4.1.1 considering the mean square error and the standard deviations.

Table 4.1.1: Mean squared error and standard deviation, training set results for
Data set of 17 rats

Simulation index	Number of neurons in hidden layer = 3	Number of neurons in hidden layer = 5	Number of neurons in hidden layer = 7	Number of neurons in hidden layer = 10
Sim1	0.3639 \pm 0.4547	0.3639 \pm 0.4546	0.3640 \pm 0.4541	0.3640 \pm 0.4542
Sim2	0.3584 \pm 0.4398	0.3584 \pm 0.4397	0.2547 \pm 0.4902	0.3583 \pm 0.4395
Sim3	0.2347 \pm 0.3464	0.1541 \pm 0.2005	0.0833 \pm 0.2320	0.0228 \pm 0.0263
Sim4	0.3407 \pm 0.4776	0.2735 \pm 0.5042	0.3406 \pm 0.4777	0.3406 \pm 0.4775
Sim5	0.2082 \pm 0.3625	0.3066 \pm 0.4959	0.3056 \pm 0.4917	0.3052 \pm 0.4886
Sim6	0.1759 \pm 0.3048	0.1712 \pm 0.2580	0.1663 \pm 0.2448	0.1707 \pm 0.2551
Sim7	0.2919 \pm 0.3895	0.2446 \pm 0.3650	0.3612 \pm 0.4607	0.3344 \pm 0.7846
Sim8	0.1353 \pm 0.2569	0.0979 \pm 0.1978	0.2235 \pm 0.3300	0.0439 \pm 0.0548
Sim9	0.2518 \pm 0.4803	0.0356 \pm 0.0618	0.0106 \pm 0.0175	0.0487 \pm 0.1186
Sim10	0.3013 \pm 0.5734	0.3305 \pm 0.5595	0.2845 \pm 0.4107	0.2844 \pm 0.3855
Sim11	0.3614 \pm 0.4532	0.3614 \pm 0.4530	0.3972 \pm 0.7828	0.2761 \pm 0.5148
Sim12	0.2367 \pm 0.3511	0.0741 \pm 0.1376	0.0290 \pm 0.0438	0.0716 \pm 0.0763
Sim13	0.2873 \pm 0.3567	0.2873 \pm 0.3567	0.2872 \pm 0.3566	0.2872 \pm 0.3566
Sim14	0.2673 \pm 0.3417	0.2625 \pm 0.3783	0.1451 \pm 0.1857	0.1641 \pm 0.1985
Sim15	0.2073 \pm 0.2974	0.0221 \pm 0.0394	0.0177 \pm 0.0180	0.0365 \pm 0.0756
Sim16	0.2764 \pm 0.5613	0.3634 \pm 0.4610	0.1136 \pm 0.2083	0.2869 \pm 0.5670
Sim17	0.3523 \pm 0.4008	0.3523 \pm 0.4009	0.3518 \pm 0.4010	0.3080 \pm 0.5608

In order to investigate the results given in Table 4.1.1 further, general mean is calculated considering the neuron numbers of hidden layer and these are given in Table 4.1.2. When these general mean values are taken into consideration the best results are obtained for 10 hidden layer neurons, while the worst results are obtained for three hidden neurons. Considering these results a conclusion can be drawn that for training phase as the number of neurons increase the results got better. So, in order to have a concrete result for the number of hidden layer neurons both of these two cases will be considered in the sequel.

Table 4.1.2: General Mean squared error and standard deviation

Hidden Neuron Numbers	General Mean Error
3	0.2736 \pm 0.4028
5	0.2388 \pm 0.3391
7	0.2198 \pm 0.3297
10	0.2178 \pm 0.3432

A second trial is carried out with data set where the values for 14th rat is not considered, thus only values for 16 rat are taken into consideration. The reason of excluding the 14th rat has been explained in the above paragraphs. The data set composed of 16 rats has a normal distribution when compared to data set composed of 17 rats. The inputs and the output are same as in the previous case and all inputs and output values are normalized.

Simulation index again indicate which rat value is taken as test value and again four different hidden layer number is considered and 16 data sets are simulated for four different hidden layer neuron numbers, thus 64 simulations are carried out. The results are summarized in Table 4.1.2.

Table 4.1.3: Mean squared error and standard deviation, training set results for
Data set of 16 rats

Simulation index	Number of neurons in hidden layer = 3	Number of neurons in hidden layer = 5	Number of neurons in hidden layer = 7	Number of neurons in hidden layer = 10
Sim1	0.2295 \pm 0.2912	0.3280 \pm 0.3492	0.1029 \pm 0.1405	0.0438 \pm 0.1175
Sim2	0.2252 \pm 0.3014	0.0618 \pm 0.0576	0.1684 \pm 0.1487	0.0959 \pm 0.1025
Sim3	0.2422 \pm 0.3485	0.2400 \pm 0.3402	0.3558 \pm 0.3563	0.3558 \pm 0.3563
Sim4	0.3152 \pm 0.3164	0.0859 \pm 0.0960	0.1854 \pm 0.2455	0.1070 \pm 0.1370
Sim5	0.1925 \pm 0.3806	0.2096 \pm 0.2487	0.1925 \pm 0.3807	0.1928 \pm 0.3810
Sim6	0.2565 \pm 0.2810	0.2565 \pm 0.2810	0.2568 \pm 0.2802	0.2565 \pm 0.2812
Sim7	0.0755 \pm 0.1195	0.1298 \pm 0.1808	0.0692 \pm 0.1112	0.0220 \pm 0.0264
Sim8	0.1110 \pm 0.1445	0.1169 \pm 0.1788	0.1610 \pm 0.3484	0.1542 \pm 0.2416
Sim9	0.3190 \pm 0.3201	0.1037 \pm 0.0900	0.1178 \pm 0.1770	0.0763 \pm 0.1035
Sim10	0.1369 \pm 0.3196	0.1164 \pm 0.2508	0.1162 \pm 0.2508	0.1159 \pm 0.2498
Sim11	0.1237 \pm 0.1816	0.3003 \pm 0.3030	0.0849 \pm 0.2154	0.2774 \pm 0.4392
Sim12	0.3145 \pm 0.3305	0.3527 \pm 0.3840	0.4007 \pm 0.6022	0.2120 \pm 0.4659
Sim13	0.3391 \pm 0.3193	0.2690 \pm 0.3087	0.0138 \pm 0.0254	0.2690 \pm 0.3087
Sim14	0.1104 \pm 0.1044	0.1032 \pm 0.1144	0.0762 \pm 0.0867	0.0675 \pm 0.1211
Sim15	0.2516 \pm 0.2874	0.2589 \pm 0.4997	0.2589 \pm 0.4986	0.0684 \pm 0.1331
Sim16	0.3061 \pm 0.3216	0.1056 \pm 0.1654	0.3159 \pm 0.3940	0.2422 \pm 0.2171

In order to investigate these results with previous case and to understand the effect of neuron numbers again a general mean is calculated for each hidden layer neuron number. These are given in Table 4.1.4.

Table 4.1.4: General Mean squared error and standard deviation

Hidden Neuron Numbers	General Mean Error
3	0.2218 ± 0.2730
5	0.1899 ± 0.2405
7	0.1798 ± 0.2663
10	0.1598 ± 0.2301

Based on the results summarized in Table 4.1.2 and Table 4.1.4, it can be followed that results for the second case where 16 rat are considered are better in the over all evaluation. Thus from now on only this data set will be considered for the further investigations on determining the training/test set.

The number of neurons to be considered is thus determined, now the training and test sets will be determined. Three different cases will be constructed. In the first case test set will be composed of rat values that give the worst results, in the second case test set will be composed of rat values that give the best results and in the last case a test set will be a combination of good and bad. These three different cases will be constructed for hidden layer neuron number of three and ten separately and for each case 20 different initial values of weights will be considered, to investigate the effect of initial weight values on the results.

4.1.2 Analyzing the effect of initial conditions for training and test set results

Considering the results of above carried analysis, in this subsection training and test sets will be constructed according to above stated argument for 16 rats. The results summarized in Table 4.1.3 will be guiding for this procedure. Only two cases for hidden layer neurons will be considered, so test and training sets will be determined and simulations will be carried both for three and ten hidden layer neurons. The training and test sets will be formed for three cases, as an example considering the neuron number of three, the best result obtained is Sim7, and thus the 7th rat is not in the training set. This means that the 7th rat decreases the performance of ANN, so it will be considered in the set of bad case. Again for neuron number three, sim9 corresponds to worst results and this means that when 9th rat value is not in the training set the results got worse, so rat 9 improves the training phase and should

be in the set of good case. Following this procedure, three different test sets will be formed corresponding to five best, five worst and five mixed rat values and the remaining 11 will form the training set for these three different cases.

Thus, three test sets are formed for hidden layer neuron number three, where the data related to rat 7, 8, 10, 11 and 14 form the test set for the best case (data set 1) as when these values are not in the training set the performance of training decreases. The second case which corresponds to worst case (data set 2) is constructed by forming the test set from data of rat 4, 9, 12, 13 and 16. The test set for third case corresponding to mixed case (data set 3) is formed by data of rat 4, 7, 13, 14 and 16. In all cases the remaining values form the training set.

Similar procedure is followed for hidden layer neuron number 10 to form the training and test sets. In this case the test set for the best case (data set 4) is composed of 1,7,9,14 and 15, the worst case (data set 5) is composed of 3,6,11,13 and 16 and the mixed case (data set 6) is composed of 1,3,11,15 and 16. Again in all cases the remaining values form the training set

Table 4.1.5: Mean squared error and standard deviation, training set results for
First Trial with 16 rats

Number of neurons in hidden layer	Simulation index	Data Set 1	Data Set 2	Data Set 3
3	Sim1	0.0043 \pm 0.0152	0.0010 \pm 0.0056	0.0269 \pm 0.1164
	Sim2	0.0357 \pm 0.1148	0.1764 \pm 0.2793	0.0268 \pm 0.1164
	Sim3	0.0370 \pm 0.1142	0.0010 \pm 0.0042	0.0269 \pm 0.1164
	Sim4	0.0358 \pm 0.1147	0.1764 \pm 0.2793	0.0268 \pm 0.1164
	Sim5	0.0366 \pm 0.1148	0.0010 \pm 0.0057	0.0269 \pm 0.1162
	Sim6	0.0366 \pm 0.1142	0.1764 \pm 0.2793	0.0269 \pm 0.1164
	Sim7	0.0330 \pm 0.1266	0.0010 \pm 0.0057	0.0268 \pm 0.1164
	Sim8	0.0399 \pm 0.1170	0.1764 \pm 0.2793	0.0261 \pm 0.1173
	Sim9	0.0360 \pm 0.1144	0.0010 \pm 0.0055	0.0269 \pm 0.1164
	Sim10	0.0618 \pm 0.2617	0.1764 \pm 0.2793	0.0269 \pm 0.1164
	Sim11	0.0043 \pm 0.0152	0.001 \pm 0.0056	0.0269 \pm 0.1164
	Sim12	0.0153 \pm 0.0356	0.0010 \pm 0.0057	0.0269 \pm 0.1164
	Sim13	0.0331 \pm 0.1267	0.0010 \pm 0.0057	0.0269 \pm 0.1164
	Sim14	0.0024 \pm 0.0074	0.0010 \pm 0.0057	0.0269 \pm 0.1163
	Sim15	0.1418 \pm 0.3500	0.0010 \pm 0.0057	0.0268 \pm 0.1165
	Sim16	0.0358 \pm 0.1145	0.0010 \pm 0.0056	0.0269 \pm 0.1164
	Sim17	0.0371 \pm 0.1143	0.0010 \pm 0.0057	0.0269 \pm 0.1163
	Sim18	0.0371 \pm 0.1147	0.1764 \pm 0.2793	0.0268 \pm 0.1165
	Sim19	0.0359 \pm 0.1145	0.0010 \pm 0.0054	0.0268 \pm 0.1164
	Sim20	0.0329 \pm 0.1263	0.0010 \pm 0.0054	0.0268 \pm 0.1165
		Data Set 4	Data Set 5	Data Set 6
10	Sim1	0.0039 \pm 0.0109	0.0276 \pm 0.08	0.0021 \pm 0.0044
	Sim2	0.0087 \pm 0.0164	0.0728 \pm 0.358	0.0724 \pm 0.3581
	Sim3	0.0091 \pm 0.0339	0.0758 \pm 0.1479	0.0726 \pm 0.359
	Sim4	0.0062 \pm 0.021	0.011 \pm 0.0184	0.0993 \pm 0.2732
	Sim5	0.1115 \pm 0.1674	0.034 \pm 0.0939	0.1898 \pm 0.314
	Sim6	0.0091 \pm 0.0186	0.0500 \pm 0.2256	0.0723 \pm 0.3581
	Sim7	0.1115 \pm 0.1674	0.0728 \pm 0.358	0.0021 \pm 0.007
	Sim8	0.0045 \pm 0.0119	0.0726 \pm 0.3573	0.0168 \pm 0.0288
	Sim9	0.0037 \pm 0.0105	0.0779 \pm 0.287	0.1898 \pm 0.314
	Sim10	0.0043 \pm 0.0081	0.0000 \pm 0.0001	0.0137 \pm 0.0376
	Sim11	0.1115 \pm 0.1675	0.031 \pm 0.0718	0.1898 \pm 0.314
	Sim12	0.0041 \pm 0.0199	0.1128 \pm 0.2516	0.1898 \pm 0.314
	Sim13	0.0124 \pm 0.0409	0.0738 \pm 0.3569	0.0047 \pm 0.0085
	Sim14	0.1115 \pm 0.1675	0.0738 \pm 0.3565	0.1898 \pm 0.314
	Sim15	0.0083 \pm 0.0125	0.0728 \pm 0.358	0.1898 \pm 0.314
	Sim16	0.0041 \pm 0.0117	0.0115 \pm 0.036	0.1898 \pm 0.314
	Sim17	0.007 \pm 0.0157	0.0509 \pm 0.2264	0.1898 \pm 0.314
	Sim18	0.0047 \pm 0.0129	0.0241 \pm 0.0626	0.0244 \pm 0.0361
	Sim19	0.0054 \pm 0.015	0.0000 \pm 0.0000	0.1898 \pm 0.314
	Sim20	0.1115 \pm 0.1672	0.0059 \pm 0.008	0.0724 \pm 0.3584

In Table 4.1.5, the training results obtained for the three different training/test set constructed for best case, worst case and mixed case are summarized both for hidden layer neuron number three and ten.

In order to investigate the effect of initial weights on the performance of ANN during the training phase, 20 simulations are carried out for each case and mean value of squared error and standard deviation is calculated for all. The below given equation is used to calculate mean of squared error:

$$e_{avg} = \frac{1}{11} \frac{1}{2} \sum_{i=1}^{16} (y_d - y_{MLP})^2 \quad (4.1d)$$

When the results summarized in Table 4.1.5 is considered, for the training/test sets of best case (data set 1 and 4), best result is obtained for simulation nine when hidden layer neuron number is 10. For the training/test sets of worst case (data set 2 and 5), best result is obtained for simulation nineteen when hidden layer neuron number is again ten. For the training/test sets of mixed case (data set 3 and 6), best result is obtained for the first simulation when hidden layer neuron number is ten. So, for hidden layer number ten the effect of initial weights is somewhat effective, while for hidden layer neuron number, the value of initial weights are not that much effective, only for data set 1 and 2 the results do depend on initial values.

The test results of the simulations summarized in Table 4.1.5 are given in Table 4.1.6. In this Table once the training phase is completed the weights obtained are used for the data that were not used during training. Again mean of squared error calculated according to below given equation and the standard deviations are calculated.

$$e_{avg} = \frac{1}{5} \frac{1}{2} \sum_{i=1}^{16} (y_d - y_{MLP})^2 \quad (4.1e)$$

In Table 4.1.6 test results are given indicating the data set, the number of hidden layer neurons and the simulation number.

Table 4.1.6: Mean squared error and standard deviation, Test set results for
First Trial with 16 rats

Number of neurons in hidden layer	Simulation index	Data Set 1	Data Set 2	Data Set 3
3	Sim1	0.2672 \pm 0.3445	0.1304 \pm 0.0883	0.3193 \pm 0.4242
	Sim2	0.1289 \pm 0.1754	0.1657 \pm 0.2253	0.3202 \pm 0.4247
	Sim3	0.1169 \pm 0.156	0.1173 \pm 0.0913	0.3186 \pm 0.4238
	Sim4	0.1277 \pm 0.1737	0.1657 \pm 0.2253	0.3199 \pm 0.4247
	Sim5	0.1357 \pm 0.1753	0.1303 \pm 0.0884	0.3173 \pm 0.4238
	Sim6	0.1191 \pm 0.1598	0.1657 \pm 0.2253	0.3195 \pm 0.4243
	Sim7	0.1944 \pm 0.1990	0.1292 \pm 0.0886	0.3199 \pm 0.4247
	Sim8	0.0909 \pm 0.1218	0.1657 \pm 0.2253	0.2756 \pm 0.3835
	Sim9	0.1238 \pm 0.1679	0.1269 \pm 0.0886	0.3193 \pm 0.4241
	Sim10	0.3822 \pm 0.3198	0.1657 \pm 0.2253	0.3195 \pm 0.4243
	Sim11	0.2671 \pm 0.3444	0.1279 \pm 0.0886	0.3197 \pm 0.4245
	Sim12	0.2329 \pm 0.2284	0.1297 \pm 0.0886	0.3188 \pm 0.4239
	Sim13	0.1939 \pm 0.1987	0.1284 \pm 0.0886	0.3195 \pm 0.4243
	Sim14	0.2728 \pm 0.3506	0.1294 \pm 0.0886	0.3188 \pm 0.4243
	Sim15	0.2053 \pm 0.1674	0.1298 \pm 0.0886	0.3201 \pm 0.4245
	Sim16	0.1263 \pm 0.1721	0.1270 \pm 0.0886	0.3198 \pm 0.4244
	Sim17	0.1168 \pm 0.1552	0.1296 \pm 0.0886	0.3165 \pm 0.4227
	Sim18	0.1386 \pm 0.1714	0.1657 \pm 0.2253	0.3203 \pm 0.4247
	Sim19	0.1252 \pm 0.1701	0.1251 \pm 0.0888	0.3199 \pm 0.4245
	Sim20	0.1958 \pm 0.2000	0.1255 \pm 0.0888	0.3213 \pm 0.4252
		Data Set 4	Data Set 5	Data Set 6
10	Sim1	0.4358 \pm 0.3449	0.4162 \pm 0.5912	0.1265 \pm 0.1565
	Sim2	0.0785 \pm 0.0890	0.2351 \pm 0.2457	0.1615 \pm 0.2058
	Sim3	0.2523 \pm 0.2916	0.4362 \pm 0.5321	0.1713 \pm 0.1589
	Sim4	0.3513 \pm 0.4921	0.0673 \pm 0.0842	0.4389 \pm 0.6467
	Sim5	0.1079 \pm 0.1077	0.3939 \pm 0.4848	0.1393 \pm 0.1625
	Sim6	0.5532 \pm 0.4567	0.2555 \pm 0.2600	0.3706 \pm 0.3912
	Sim7	0.1171 \pm 0.0994	0.4419 \pm 0.5923	0.1738 \pm 0.1812
	Sim8	0.5280 \pm 0.5252	0.4184 \pm 0.4977	0.9059 \pm 0.4448
	Sim9	0.5147 \pm 0.4956	0.3431 \pm 0.6428	0.1529 \pm 0.179
	Sim10	0.4541 \pm 0.4699	0.1696 \pm 0.1740	0.1388 \pm 0.1655
	Sim11	0.3819 \pm 0.3336	0.6386 \pm 0.6927	0.0802 \pm 0.1188
	Sim12	0.1171 \pm 0.0994	0.1054 \pm 0.0853	0.1393 \pm 0.1625
	Sim13	0.1171 \pm 0.0994	0.6166 \pm 0.5966	0.1477 \pm 0.1260
	Sim14	0.1171 \pm 0.0994	0.1211 \pm 0.0877	0.0838 \pm 0.1259
	Sim15	0.0251 \pm 0.0156	0.4394 \pm 0.5893	0.3865 \pm 0.2625
	Sim16	0.1388 \pm 0.1085	0.4763 \pm 0.6951	0.1393 \pm 0.1625
	Sim17	0.1176 \pm 0.0980	0.4070 \pm 0.4565	0.0802 \pm 0.1188
	Sim18	0.2169 \pm 0.3599	0.4036 \pm 0.6077	0.4124 \pm 0.5084
	Sim19	0.5308 \pm 0.5661	0.0196 \pm 0.0232	0.0845 \pm 0.1485
	Sim20	0.1171 \pm 0.0994	0.3924 \pm 0.6772	0.6397 \pm 0.6758

Now a discussion which is similar to the one carried out for the training set results given in Table 4.1.5 will be done for the test set results given in Table 4.1.6 in order to understand the effect of initial weight values on test set.

When the hidden layer neuron number is three, for data set 1 the worst result is obtained in simulation 14 ($0,2728 \pm 0,3506$), the best result is obtained in simulation 8 ($0,0909 \pm 0,1218$), as can be followed from these, there is an effective difference between these two cases. Similarly, for the data set 2, the initial values of weights are effective on the test results. On the other hand for the data set 3, where mixed values are considered, there is no such difference depending on the initial values of weights.

When hidden layer neuron number is 10, for all data sets the effect of different initial values is important. These results reveal that the choice of initial weights do have effect on the results especially on test results, thus in order to see this effect one simulation is not enough and more simulations has to be carried out to have reliable results.

So the same case will be repeated once more and then both cases will reconsidered in order to decide the most suitable training/test set and the effect of initial values of weights. In Table 4.1.7 similar to Table 4.1.5 the results for the training set are given.

Table 4.1.7: Mean squared error and standard deviation, Training set results for
Second Trial with 16 rats

Number of neurons in hidden layer	Simulation index	Data Set 1	Data Set 2	Data Set 3
3	Sim1	0.1195 ± 0.4117	0.1764 ± 0.2793	0.0268 ± 0.1164
	Sim2	0.0257 ± 0.1146	0.1764 ± 0.2793	0.0269 ± 0.1164
	Sim3	0.0043 ± 0.0152	0.001 ± 0.0056	0.0269 ± 0.1164
	Sim4	0.0357 ± 0.1148	0.1764 ± 0.2793	0.0268 ± 0.1164
	Sim5	0.037 ± 0.1142	0.001 ± 0.0042	0.0269 ± 0.1164
	Sim6	0.0358 ± 0.1147	0.1764 ± 0.2793	0.0268 ± 0.1164
	Sim7	0.0366 ± 0.1148	0.001 ± 0.0057	0.0269 ± 0.1162
	Sim8	0.0366 ± 0.1142	0.1764 ± 0.2793	0.0269 ± 0.1164
	Sim9	0.033 ± 0.1266	0.001 ± 0.0057	0.0268 ± 0.1164
	Sim10	0.0399 ± 0.117	0.1764 ± 0.2793	0.0261 ± 0.1173
	Sim11	0.036 ± 0.1144	0.001 ± 0.0055	0.0269 ± 0.1164
	Sim12	0.0618 ± 0.2617	0.1764 ± 0.2793	0.0269 ± 0.1164
	Sim13	0.0043 ± 0.0152	0.001 ± 0.0056	0.0269 ± 0.1164
	Sim14	0.0153 ± 0.0356	0.001 ± 0.0057	0.0269 ± 0.1164
	Sim15	0.0331 ± 0.1267	0.001 ± 0.0057	0.0269 ± 0.1164
	Sim16	0.0024 ± 0.0074	0.001 ± 0.0057	0.0269 ± 0.1163
	Sim17	0.1418 ± 0.35	0.001 ± 0.0057	0.0268 ± 0.1165
	Sim18	0.0358 ± 0.1145	0.001 ± 0.0056	0.0269 ± 0.1164
	Sim19	0.0371 ± 0.1143	0.001 ± 0.0057	0.0269 ± 0.1163
	Sim20	0.0371 ± 0.1147	0.1764 ± 0.2793	0.0268 ± 0.1165
		Data Set 4	Data Set 5	Data Set 6
10	Sim1	0.1115 ± 0.1677	0.0003 ± 0.0006	0.0168 ± 0.056
	Sim2	0.0042 ± 0.0084	0.0729 ± 0.3579	0.0724 ± 0.3582
	Sim3	0.0063 ± 0.0241	0.0511 ± 0.2238	0.0051 ± 0.0141
	Sim4	0.0108 ± 0.0346	0.0728 ± 0.3578	0.0724 ± 0.3583
	Sim5	0.0068 ± 0.0217	0.0728 ± 0.3575	0.1898 ± 0.314
	Sim6	0.0020 ± 0.0039	0.0738 ± 0.3567	0.0434 ± 0.0914
	Sim7	0.0078 ± 0.0142	0.0031 ± 0.0056	0.1898 ± 0.314
	Sim8	0.0289 ± 0.0506	0.0000 ± 0.0000	0.0723 ± 0.358
	Sim9	0.0022 ± 0.0066	0.0000 ± 0.0000	0.0085 ± 0.0191
	Sim10	0.1115 ± 0.1674	0.0728 ± 0.3578	0.0053 ± 0.0074
	Sim11	0.0067 ± 0.0128	0.0167 ± 0.0291	0.0158 ± 0.0255
	Sim12	0.0181 ± 0.064	0.0000 ± 0.0000	0.1898 ± 0.314
	Sim13	0.1115 ± 0.1672	0.0496 ± 0.222	0.0723 ± 0.358
	Sim14	0.1115 ± 0.1674	0.0179 ± 0.0417	0.0725 ± 0.3586
	Sim15	0.0087 ± 0.0216	0.0000 ± 0.0000	0.0135 ± 0.0377
	Sim16	0.1114 ± 0.1678	0.0000 ± 0.0000	0.0044 ± 0.0098
	Sim17	0.1115 ± 0.1675	0.0045 ± 0.0103	0.1898 ± 0.314
	Sim18	0.0217 ± 0.0464	0.0000 ± 0.0000	0.1898 ± 0.314
	Sim19	0.0175 ± 0.0269	0.0268 ± 0.0729	0.06 ± 0.2238
	Sim20	0.0521 ± 0.1174	0.0000 ± 0.0000	0.0037 ± 0.0092

When we evaluate this trial by itself; when hidden layer neuron number is three, for data set 1 and 2 which correspond to best and worst cases, the initial values of the

weights are effective, while for data set 3, which corresponds to mixed case the initial values of the weights are not effective. In the other case where hidden layer neuron number is 10, the initial values of weights are effective on all three data set.

Considering on the whole i.e., when hidden layer neuron number is three and 10, for the worst case (data set 1 and data set 4) the best result is obtained for neuron number ten on the sixth simulation (0.0020 ± 0.0039). The worst result is obtained for neuron number three on the 17th simulation (0.1418 ± 0.35).

In the best case (data set 2 and data set 5), the best result is obtained for neuron number 10 and for 8th, 9th, 15th, 16th 18th and 19th simulations. In all these cases the mean squared error is so small that it can be considered zero. The worst case is obtained again when the neuron number is three in the 1st, 2nd, 4th, 8th, 10th, 12th and 20th simulations where in all these simulations mean squared error is 0.1764 ± 0.2793 . For the data set 3 and data set 6, where mixed values are considered the best result is obtained for neuron number 10 in simulation 20 where the mean squared error is 0.0037 ± 0.0092 , and the worst case is obtained for neuron number 10 in 5th, 7th, 12th, 17th and 18th simulations where the mean squared error is 0.1898 ± 0.3140 .

Table 4.1.8: Mean squared error and standard deviation, Test set results for
Second Trial with 16 rats

Number of neurons in hidden layer	Simulation index	Data Set 1	Data Set 2	Data Set 3
3	Sim1	0.2336 ± 0.1654	0.1657 ± 0.2253	0.3203 ± 0.4249
	Sim2	0.1018 ± 0.1178	0.1657 ± 0.2253	0.3192 ± 0.4241
	Sim3	0.2672 ± 0.3445	0.1304 ± 0.0883	0.3193 ± 0.4242
	Sim4	0.1289 ± 0.1754	0.1657 ± 0.2253	0.3202 ± 0.4247
	Sim5	0.1169 ± 0.156	0.1173 ± 0.0913	0.3186 ± 0.4238
	Sim6	0.1277 ± 0.1737	0.1657 ± 0.2253	0.3199 ± 0.4247
	Sim7	0.1357 ± 0.1753	0.1303 ± 0.0884	0.3173 ± 0.4238
	Sim8	0.1191 ± 0.1598	0.1657 ± 0.2253	0.3195 ± 0.4243
	Sim9	0.1944 ± 0.199	0.1292 ± 0.0886	0.3199 ± 0.4247
	Sim10	0.0909 ± 0.1218	0.1657 ± 0.2253	0.2756 ± 0.3835
	Sim11	0.1238 ± 0.1679	0.1269 ± 0.0886	0.3193 ± 0.4241
	Sim12	0.3822 ± 0.3198	0.1657 ± 0.2253	0.3195 ± 0.4243
	Sim13	0.2671 ± 0.3444	0.1279 ± 0.0886	0.3197 ± 0.4245
	Sim14	0.2329 ± 0.2284	0.1297 ± 0.0886	0.3188 ± 0.4239
	Sim15	0.1939 ± 0.1987	0.1284 ± 0.0886	0.3195 ± 0.4243
	Sim16	0.2728 ± 0.3506	0.1294 ± 0.0886	0.3188 ± 0.4243
	Sim17	0.2053 ± 0.1674	0.1298 ± 0.0886	0.3201 ± 0.4245
	Sim18	0.1263 ± 0.1721	0.127 ± 0.0886	0.3198 ± 0.4244
	Sim19	0.1168 ± 0.1552	0.1296 ± 0.0886	0.3165 ± 0.4227
	Sim20	0.1386 ± 0.1714	0.1657 ± 0.2253	0.3203 ± 0.4247
		Data Set 4	Data Set 5	Data Set 6
10	Sim1	0.1079 ± 0.1077	0.1852 ± 0.1921	0.4423 ± 0.2992
	Sim2	0.1171 ± 0.0994	0.2376 ± 0.2459	0.1769 ± 0.2043
	Sim3	0.1805 ± 0.2358	0.1605 ± 0.2147	0.1771 ± 0.1839
	Sim4	0.1177 ± 0.0975	0.2023 ± 0.2215	0.1644 ± 0.0912
	Sim5	0.2605 ± 0.189	0.2178 ± 0.2079	0.3939 ± 0.2702
	Sim6	0.5639 ± 0.5337	0.1634 ± 0.2410	0.6008 ± 0.5004
	Sim7	0.5283 ± 0.5198	0.3804 ± 0.6842	0.1529 ± 0.1790
	Sim8	0.4809 ± 0.4477	0.4764 ± 0.7107	0.2458 ± 0.3789
	Sim9	0.2540 ± 0.1968	0.4300 ± 0.4268	0.3696 ± 0.3971
	Sim10	0.1171 ± 0.0994	0.1280 ± 0.1028	0.2216 ± 0.1787
	Sim11	0.5186 ± 0.5266	0.3668 ± 0.4625	0.5289 ± 0.3603
	Sim12	0.4862 ± 0.5062	0.4384 ± 0.5249	0.1393 ± 0.1625
	Sim13	0.1171 ± 0.0994	0.5604 ± 0.6602	0.1749 ± 0.1914
	Sim14	0.5311 ± 0.5291	0.1698 ± 0.2150	0.6466 ± 0.6680
	Sim15	0.1418 ± 0.1237	0.4243 ± 0.3123	0.5780 ± 0.6669
	Sim16	0.3089 ± 0.4607	0.1501 ± 0.1460	0.3981 ± 0.3386
	Sim17	0.1171 ± 0.0994	0.3337 ± 0.1673	0.0837 ± 0.1259
	Sim18	0.4709 ± 0.5678	0.1543 ± 0.1520	0.1356 ± 0.2146
	Sim19	0.2417 ± 0.2635	0.6527 ± 0.5970	0.1276 ± 0.1090
	Sim20	0.2538 ± 0.2789	0.3524 ± 0.4361	0.2892 ± 0.2003

The test results summarized in Table 4.1.8 will also be evaluated in a similar way.

Again both neuron number will be considered simultaneously, and for worst case which corresponds to data set 1 and data set 4, the best result is obtained when the neuron number is three and in 10th simulation. In this case the mean squared error is 0.0909 ± 0.1218 .

The worst result is obtained when the neuron number is 10 and in 6th simulation. For data set 1 which corresponds to worst case for neuron number three, the mean squared error changes in the interval of 0.0909 ± 0.1218 and 0.3822 ± 0.3198 depending on initial weight values. For data set 4, this interval is 0.1079 ± 0.1077 and 0.5639 ± 0.5337 , thus test set the results are better for neuron number three.

For the best case, which corresponds to data set 2 and data set 5, the best result is obtained when the neuron number is three and the simulation is the 5th one. In this case the mean squared error is 0.1173 ± 0.0913 . The worst result is obtained in the 19th simulation when the neuron number is 10 and in this case the mean squared error is 0.6527 ± 0.5970 . While for the neuron number 3 and the data set 2 is considered, the variation of mean squared error is small, i.e., between 0.1173 ± 0.0913 and 0.1657 ± 0.2253 depending on the initial weight values, for neuron number 10 and data set 5 this variation is more and is between 0.1280 ± 0.1028 and 0.6527 ± 0.5970 .

In case of mixed data set corresponding to data set 3 and data set 6, the best result is obtained in the 17th simulation and for neuron number 10 where the mean squared error is 0.1171 ± 0.0994 . The worst result is obtained again for neuron number 10 in the 5th simulation and in this case the mean squared error is 0.6008 ± 0.5004 . When results for each neuron number is considered, the mean squared error varies between 0.2756 ± 0.3835 and 0.3203 ± 0.4247 for neuron number three and between 0.1171 ± 0.0994 and 0.6008 ± 0.5004 for neuron number 10 depending on initial value of weights.

Table 4.1.9: General test results for three hidden neurons

	Number of neurons in hidden layer = 3		
	Data Set 1	Data Set 2	Data Set 3
First Trial	0.1781 ± 0.0748	0.1390 ± 0.0181	0.3172 ± 0.0098
Second Trial	0.1788 ± 0.0768	0.1431 ± 0.0191	0.3171 ± 0.0098

Table 4.1.10: General test results for ten hidden neurons

	Number of neurons in hidden layer = 10		
	Data Set 4	Data Set 5	Data Set 6
First Trial	0.2636 ± 0.1832	0.3399 ± 0.1728	0.2487 ± 0.2169
Second Trial	0.2958 ± 0.1735	0.3092 ± 0.1538	0.3024 ± 0.1789

In order to compare these two trials which are carried out separately a general mean is calculated for data sets and neuron numbers and the results are given in Table 4.1.9 and 10. So the results given in Table 4.1.10 are in a way summary of results given in Table 6 and 8

When the results are considered on the whole, even though the initial values of the weights do have an effect on the results when each single case is considered for the same number of hidden layer neurons, they do not change the results dramatically when the results in Table 4.1.9 and Table 4.1.10 are reconsidered.

When the test performance is compared considering the effect of different hidden layer neuron number, three hidden layer neuron case give better result than hidden layer number 10. This is contradicting the previous results obtained, where better results have been obtained with hidden layer neuron number 10, but it must be kept in mind that the data sets, initial values of weights are all different and there are more than one parameter affecting the performance.

In order to understand the effect of hidden layer neuron number, the effect of data set will be negated and one more analysis will be done. In this analysis, the ANN structure with ten hidden layer neurons will be trained with the data sets used in training the three hidden layer neuron structure. So the results to be compared will be obtained using the same data sets but different neuron numbers and different initial weight values. It is expected that these would enlighten more the effect of neuron number.

Table 4.1.11: Mean squared error and standard deviation, Training set results
for Third Trial with 16 rats

Number of neurons in hidden layer	Simulation index	Data Set 1	Data Set 2	Data Set 3
10	Sim1	0.0219 \pm 0.0741	0.0001 \pm 0.0001	0.0268 \pm 0.1164
	Sim2	0.0471 \pm 0.0756	0.0007 \pm 0.0032	0.0268 \pm 0.1166
	Sim3	0.0284 \pm 0.1029	0.0006 \pm 0.0030	0.0268 \pm 0.1164
	Sim4	0.0746 \pm 0.2909	0.0000 \pm 0.0000	0.0387 \pm 0.1286
	Sim5	0.0277 \pm 0.0679	0.0021 \pm 0.0032	0.0268 \pm 0.1165
	Sim6	0.0376 \pm 0.1143	0.0007 \pm 0.0039	0.0268 \pm 0.1164
	Sim7	0.0710 \pm 0.1197	0.0007 \pm 0.0021	0.0269 \pm 0.1162
	Sim8	0.0176 \pm 0.0440	0.1764 \pm 0.2793	0.0268 \pm 0.1165
	Sim9	0.0755 \pm 0.1578	0.0018 \pm 0.0055	0.0269 \pm 0.1162
	Sim10	0.0638 \pm 0.1839	0.0014 \pm 0.0030	0.0261 \pm 0.1129
	Sim11	0.0469 \pm 0.1105	0.0024 \pm 0.0042	0.0266 \pm 0.1153
	Sim12	0.0900 \pm 0.2170	0.0007 \pm 0.0035	0.0269 \pm 0.1162
	Sim13	0.0261 \pm 0.0434	0.0002 \pm 0.0006	0.0267 \pm 0.1160
	Sim14	0.0358 \pm 0.1146	0.0008 \pm 0.0041	0.0268 \pm 0.1164
	Sim15	0.0024 \pm 0.0033	0.0001 \pm 0.0001	0.0268 \pm 0.1164
	Sim16	0.0124 \pm 0.0379	0.0008 \pm 0.0040	0.0420 \pm 0.1241
	Sim17	0.0451 \pm 0.1663	0.0008 \pm 0.0043	0.0269 \pm 0.1164
	Sim18	0.0145 \pm 0.0376	0.1764 \pm 0.2793	0.0268 \pm 0.1165
	Sim19	0.0435 \pm 0.1271	0.0003 \pm 0.0007	0.0269 \pm 0.1161
	Sim20	0.0248 \pm 0.0782	0.1764 \pm 0.2793	0.0268 \pm 0.1165

When the results obtained for the training set is considered for all three data set these results are better than the results obtained in trial one and two for neuron number three. Thus, when hidden layer neuron number is 10, the results obtained for training set is improved.

Similarly, to see the performance of test set the data sets used during test phase for neuron number three is used for neuron number 10 and these results are given in Table 4.1. 12. Even though for training set better results were obtained for hidden layer neuron number 10, for test set the results are better with hidden layer neuron number three. This is most probably due to overtraining which resulted in poor performance in the test set even though the training phase performance is high.

Table 4.1.12: Mean squared error and standard deviation, Test set for
Third Trial with 16 rats

Number of neurons in hidden layer	Simulation index	Data Set 1	Data Set 2	Data Set 3
10	Sim1	0.6650 \pm 0.3336	0.1319 \pm 0.1233	0.1768 \pm 0.2498
	Sim2	0.4975 \pm 0.4877	0.7828 \pm 0.6682	0.3543 \pm 0.3188
	Sim3	0.4263 \pm 0.3066	0.7490 \pm 0.6404	0.1848 \pm 0.1809
	Sim4	0.2852 \pm 0.1986	0.4725 \pm 0.4668	0.2918 \pm 0.3743
	Sim5	0.5709 \pm 0.5214	0.3586 \pm 0.4073	0.7446 \pm 0.5053
	Sim6	0.3741 \pm 0.3824	0.7627 \pm 0.6385	0.1708 \pm 0.1691
	Sim7	0.1970 \pm 0.1298	0.3553 \pm 0.3956	0.3847 \pm 0.4816
	Sim8	0.8476 \pm 0.4151	0.1555 \pm 0.1581	0.6015 \pm 0.4786
	Sim9	0.4993 \pm 0.4043	0.1277 \pm 0.1341	0.5925 \pm 0.4779
	Sim10	0.1385 \pm 0.1177	0.2528 \pm 0.2714	0.4636 \pm 0.5816
	Sim11	0.3101 \pm 0.2480	0.6217 \pm 0.5918	0.2204 \pm 0.2868
	Sim12	0.1094 \pm 0.1048	0.1208 \pm 0.1308	0.7417 \pm 0.5047
	Sim13	0.1738 \pm 0.1760	0.3541 \pm 0.4021	0.7623 \pm 0.6483
	Sim14	0.1860 \pm 0.2593	0.3259 \pm 0.3906	0.4008 \pm 0.2615
	Sim15	0.3957 \pm 0.5694	0.3610 \pm 0.4099	0.7448 \pm 0.5410
	Sim16	0.5865 \pm 0.3818	0.3434 \pm 0.3941	0.4271 \pm 0.5483
	Sim17	0.4950 \pm 0.4402	0.2483 \pm 0.4072	0.1788 \pm 0.2311
	Sim18	0.1223 \pm 0.2205	0.1555 \pm 0.1581	0.7446 \pm 0.5093
	Sim19	0.3136 \pm 0.3979	0.9219 \pm 0.6534	0.3941 \pm 0.2495
	Sim20	0.2234 \pm 0.2037	0.1555 \pm 0.1581	0.6852 \pm 0.8081

In the test phase, the data used are new since these are not used during training phase. In Table 4.1.13, the means of the results obtained in Table 4.1. 12 are given and when these are compared with the results summarized in Table 4.1. 9 and 10, they are poor.

Table 4.1.12: General Results for third simulation

	Number of neurons in hidden layer = 10		
	Data Set 1	Data Set 2	Data Set 3
Third Trial	0.3709 \pm 0.2020	0.3879 \pm 0.2500	0.4633 \pm 0.2215

When the hidden layer neuron number is 10, the good performance obtained for the training is not observed in the test set. This is most probably due to the overtraining phenomena mentioned in the above paragraph. In order to get better test performance, instead of stopping the training phase only considering the iteration number, a limit on the upper bound of error can be assigned. In this case, training would end in a shorter time and test performance will probably increase.

When all these trials are evaluated on the whole, the results vary in each simulation with the initial values of weights. On the other hand, when each 20 simulation result is considered again by taking their means as in Table 4.1.9 and Table 4.1.10 there is not much difference in the results for the same training/test set and neuron number. As the data set, hidden layer neuron number and initial value of the weights are the parameters that affect the performance of ANN structure considered, these analysis are carried out. It is concluded that to carry out the simulations for different initial values of weights will give more reliable results, thus in the sequel simulations will be repeated for different initial weight values and their means will be given.

4.1.3 Results

As a consequence of the statistical studies made in this section, the best test results for the three different data sets are depicted in the following figures. The statistical studies were made for different numbers of hidden layer neurons, namely three and ten. Although the performance of the training phase is satisfactory in general, the test results are not as good as expected. As we mentioned above, this problem may have emerged because the generalization property of ANN became poorer due to overtraining. For this reason, it would be beneficial to let the training process end using suitable error criteria in order to avoid overtraining.

From the data set 1, five rats which reduce the performance of ANN mostly were chosen as the test set. After training is performed using the remaining 11 rat, the ANN is applied to the test phase. The results on training and test are illustrated in the Figure 4.1.1 and 4.1.2, respectively. Also, test and training results are given as numerical values on Tables A.1 and A.2 in appendix. As can be seen from the figures the test results are not as good as the results of the training phase.

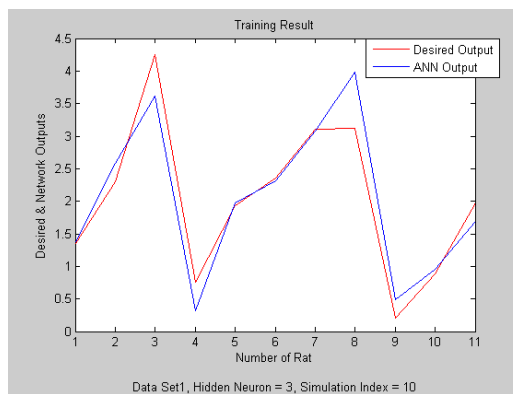


Figure 4.1.1: Outputs of Training Phase

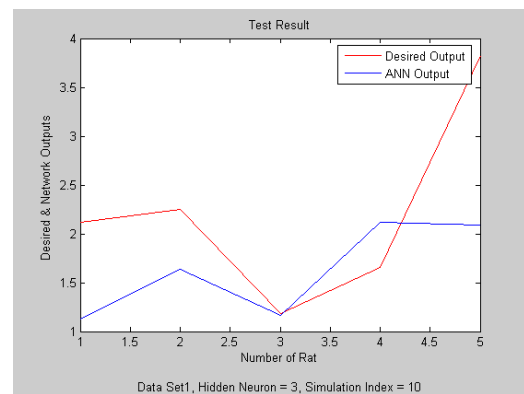


Figure 4.1.2: Outputs of Test Phase

Similarly, from the data set 2, five rats which increase the performance of ANN mostly are chosen as the test set. Training was performed using the remaining 11 rat and the ANN is applied to the test phase. The results on training and test are illustrated in the Figure 4.1.3 and 4.1.4, respectively. As can be seen from the figures the results of the training phase is perfect whereas the test phase is again not as expected.

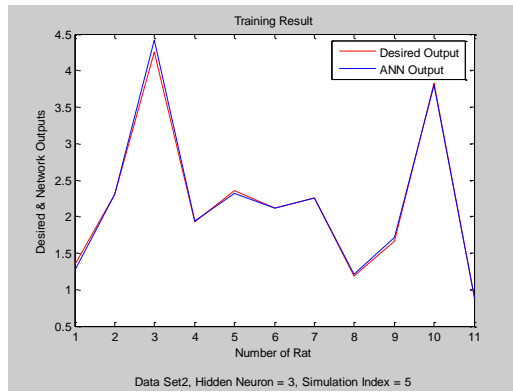


Figure 4.1.3: Outputs of Training Phase

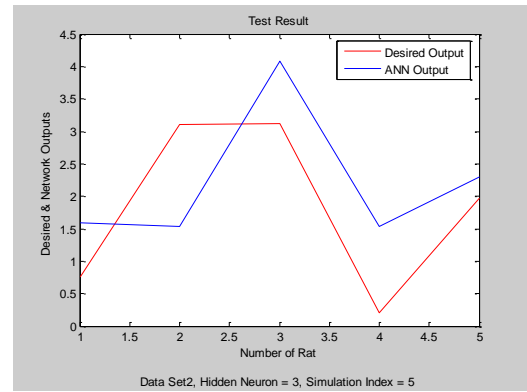


Figure 4.1.4: Outputs of Test Phase

The test and training sets are constructed randomly for the data set 3. The result can be followed from the Figure 4.1.5 and 4.1.6. The test results for this case are much worse, which points out the effect of the convenience of test and training sets.

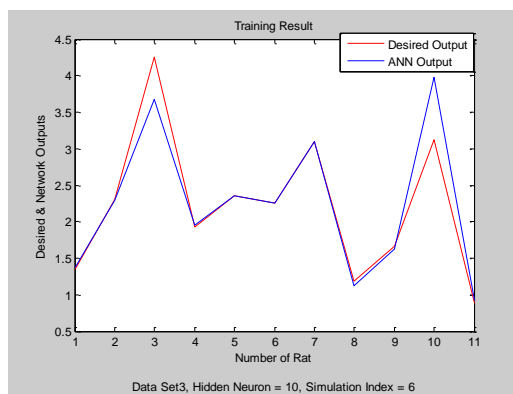


Figure 4.1.5: Outputs of Training Phase

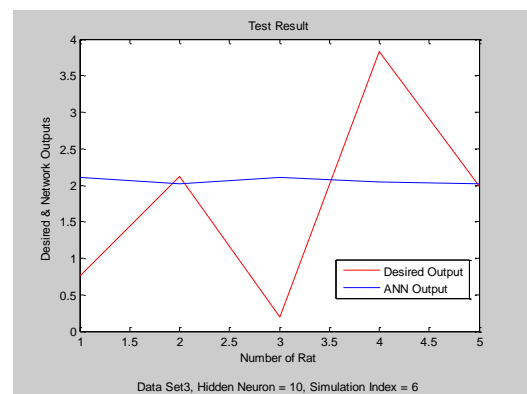


Figure 4.1.6: Outputs of Test Phase

Consequently, head shake and immobility behavior has a considerable effect on BD ratio. In other terms there are relationship between these behaviors and behavioral despair.

4.2. A General Model for Prediction of BD Ratio with Seasonal data of Rat Groups

The aim of this section is to get a general model revealing the effect of four seasons on the behavior of rat groups attended to Porsolt test. In test phase, the general model is also tried to predict BD ratio. The differences is using seasonal datasets is test phase and only immobility behavior is considered as input of the ANN. In this section, two main studies are realized to achieve same goal. These studies are explained and detailed in following.

First modeling is summarized as follow. During the training phase, the MLP structure is trained with data which is composed of previously used data related to rat group of 17 rats. Then, in test phase the trained MLP structure is used for each seasonal group, so data of 37 rats are employed during the test phase. We are interested in which seasons the test results give valid information. So we could understand whether the model obtained is a general one that can be used for any season. In this modeling attempt, number of neurons in the input layer enlarged as the data size is now more. In hidden layer, there are twenty computational neurons and one neuron in the output layer. The inputs of MLP are various immobilization durations in different minutes of PST1 (First Day of Porsolt test). Output is BD ratio of each rat like in the previous model.

Second model realized with training set is data of sixteen rats without extreme rat like previous 4.1 sections. Test set contains data of 37 rats which are belonged to different seasonal groups. Four different data sets are constituted to apply as inputs of ANN. First data set consist of immobility in fourth, fifth, sixth and seventh minutes (imm4, imm5, imm6 and imm7) as inputs of the ANN. Second dataset contain difference duration between fifth and fourth minutes (imm5-imm4). In third dataset, the duration between fifth and sixth minutes (imm6-imm5) was taken as input of the ANN. Fourth dataset included the difference durations between fifth and fourth minutes (imm5-imm4) and between fifth and sixth minutes (imm6-imm5).

4.2.1 Modeling general seasonal behavior with immobility values at different minutes

In this model, seven simulations are realized to achieve same goal. For each simulation different input datasets are used and the structure of the ANN isn't changed. Details of simulations are given follow.

Modeling of the general seasonal behavior considering only third, fourth and fifth minute's immobility

As our aim was to obtain a general model for all seasons, we considered as training data the data used in the previous section. Thus, training set contains data of 17 rats with extreme rat. The inputs of ANN in this case are durations of immobilization measured in third, fourth and fifth minutes of PST1. The output of ANN is BD ratio of each rat. The input, desired output and ANN output values are given on A.7 and A.8 tables in appendix.

The stopping criteria for this model is that when iteration number reach 5000, training phase terminate The training performance of the ANN, is not good enough, as can be followed from below figures where mean error for each iteration is shown. The desired values of outputs and ANN outputs are shown following.

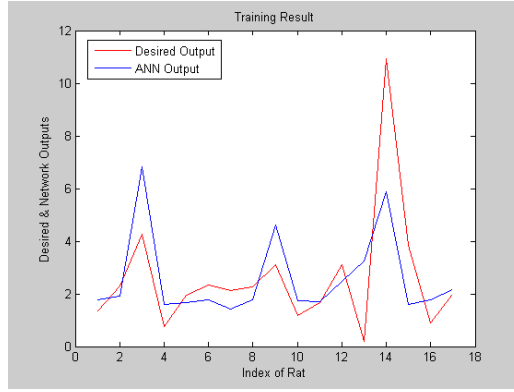


Figure 4.2.1: Desired and ANN Output

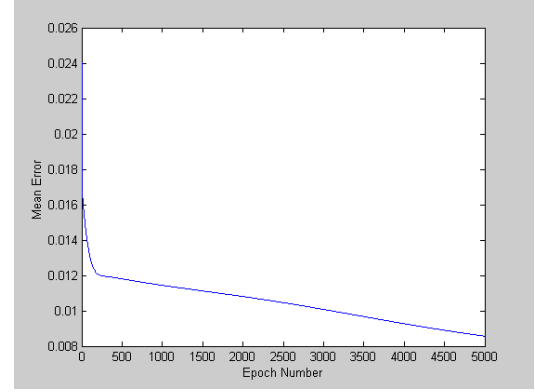


Figure 4.2.2: Mean Error of Training Process

After training phase, in which the weights of the ANN are maintained, the performance of the ANN is tested. In test phase, there are data for four rat groups related to different seasons. Months of experiments, when data related to rat groups are obtained, are the following: August, February, May and November.

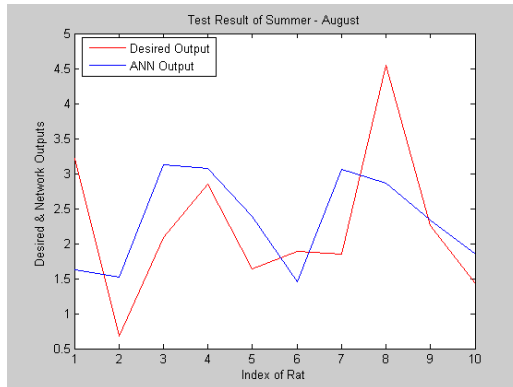


Figure 4.2.3: Desired and ANN Output of August

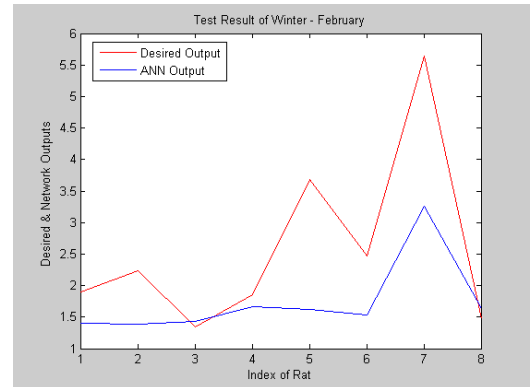


Figure 4.2.4: Desired and ANN Output of February

Test results obtained are not satisfactory. There may be many causes but most important one is we are using different rat groups and ANN has interpolation capability, not extrapolation ability so for ANN the results would be meaningful only when the training and test data are from the same data set.

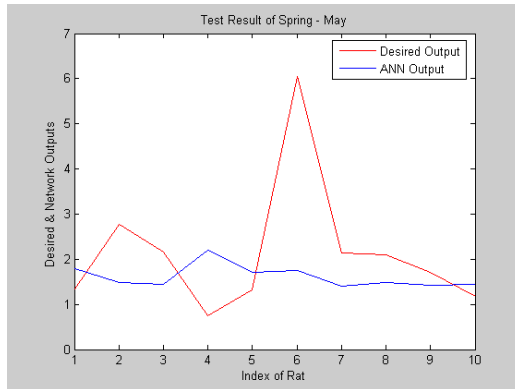


Figure 4.2.5: Desired and ANN Output Of May

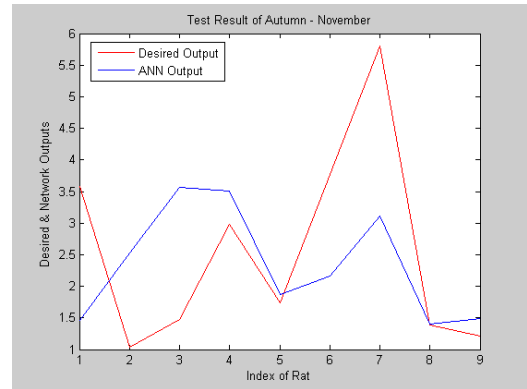


Figure 4.2.6: Desired and ANN Output of November

In order to use ANN with extrapolation capability we have to use more information about the phenomena implemented into ANN structure. The other reasons may be due to bad training performance, few inputs, seasonal effects etc.

Modeling of the general seasonal behavior considering each of the first five minutes immobility

Now, simulation results for five inputs which correspond to first five minutes of immobility's values are obtained. Number of rats and structure of ANN are same as in the previous subsection. The output is again BD. Inputs and output of the MLP are given in A.7 and A.9 tables respectively on appendix.

Results of training phase is better than previous experiment' training phase. Iteration number is 5000 and last mean error that is critical performance parameter value is 0.0030. In below on left graphs outputs of the MLP and desired outputs are plotted together. Mean square error is also shown below and its last value is 0.0030

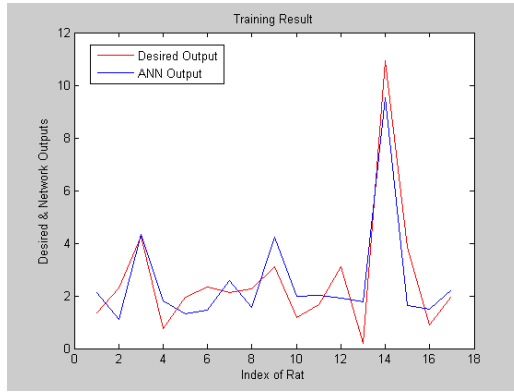


Figure 4.2.7: Desired and ANN Output

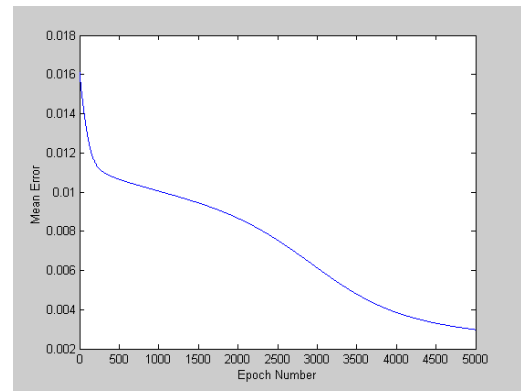


Figure 4.2.8: Mean Square Error during Training Process

Again, data of different seasonal rat groups are taken as test pattern. The structure of the ANN has constituted in training phase. Now, this new updated structure is tested in this phase with wholly different rat groups which are consist of four seasonal groups. Results of four rats groups of different seasons are given below graphs separately.

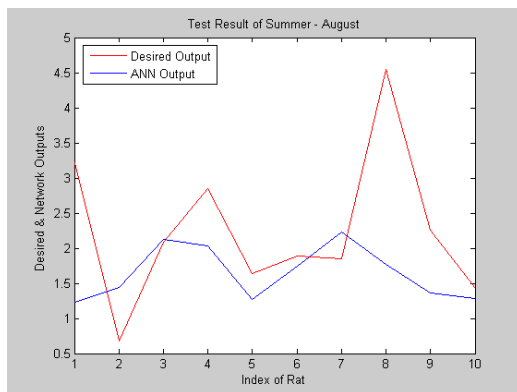


Figure 4.2.9: Desired and ANN Output of August

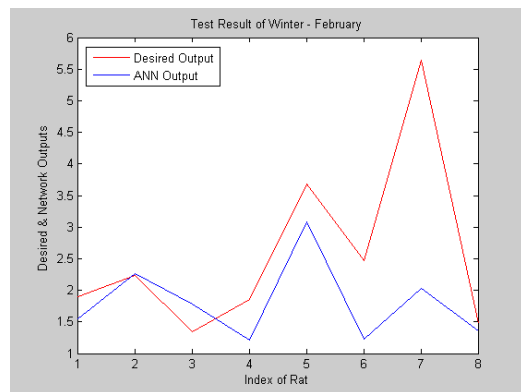


Figure 4.2.10: Desired and ANN Output of February

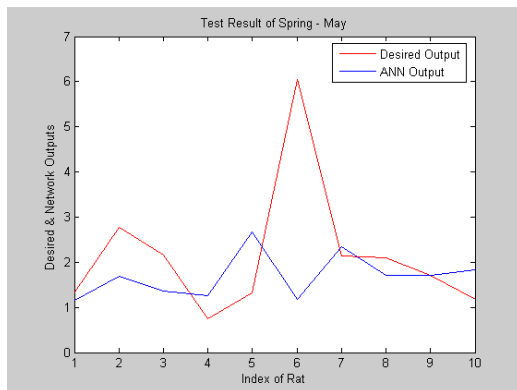


Figure 4.2.11: Desired and ANN Output Of May

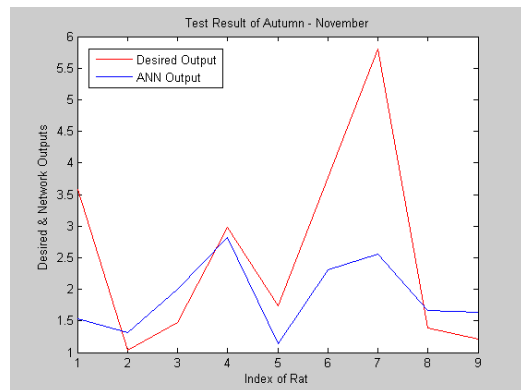


Figure 4.2.12: Desired and ANN Output of November

Training performance of this experiment is better than first experiment. Also, test results are relatively better than first. But results aren't good enough because of different test and training groups, insufficient of patterns etc

Modeling of the General Seasonal behavior considering each of the first six minutes immobility

The training pattern contains data of 17 rats like previous simulations. The inputs of the MLP are only difference from second simulation adding the duration of immobilization in sixth minutes. The inputs and outputs are given A.7 and A.10 tables in appendix. In each iteration imm_{11} , imm_{12} ... imm_{16} applied to the MLP as inputs and then the weights of the MLP are updated. These processes continue until iteration reach 5000.

The performance of training phase gets better than previous simulations. A main criterion of performance is mean square error. So, comparisons are done according to last mean square error that is 0.0024 in this simulation. When the last mean square error decrease the ANN produce better outputs.

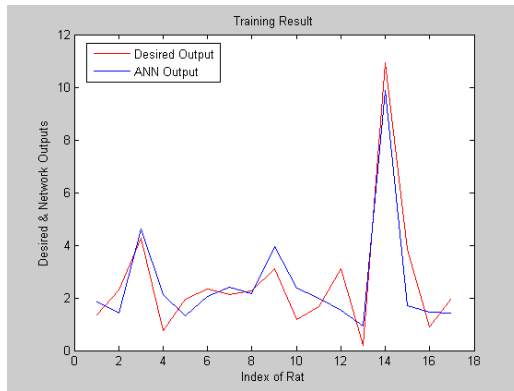


Figure 4.2.13: Desired and ANN Output

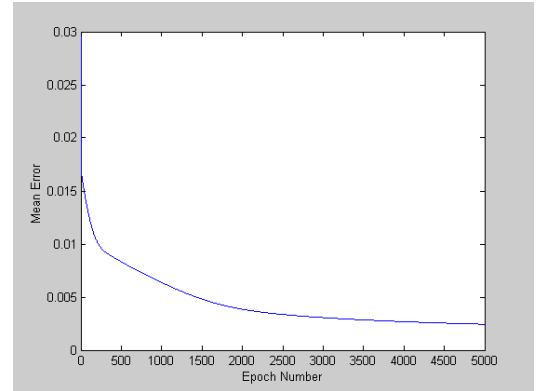


Figure 4.2.14: Mean Square Error of Training

The updated structure of the MLP is tested for each seasonal rat groups. We searched that a general simulation modeling for all season or not, what is difference these groups.

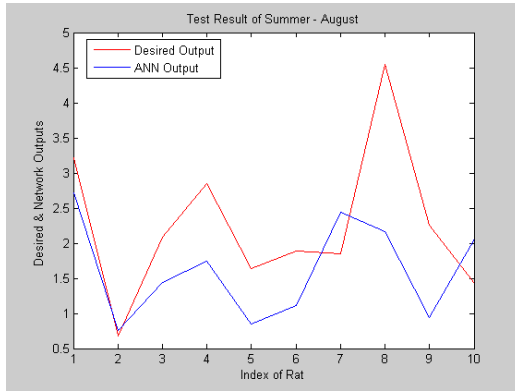


Figure 4.2.15: Desired and ANN Output of August

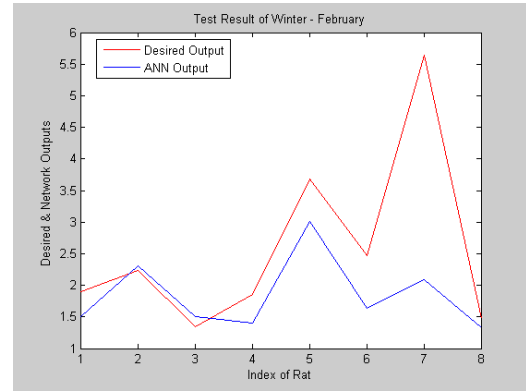


Figure 4.2.16: Desired and ANN Output of February

In test phase, the performance of ANN is relatively better than previous simulations. This circumstance is normal according to better performance of training.

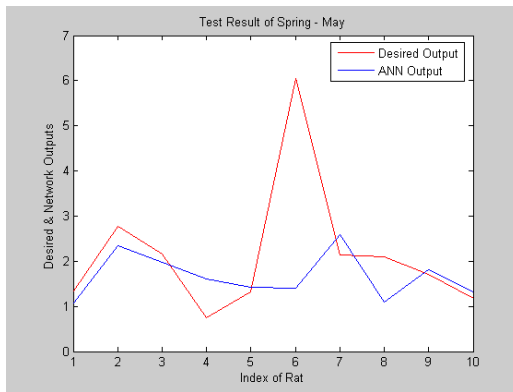


Figure 4.2.17: Desired and ANN Output of May

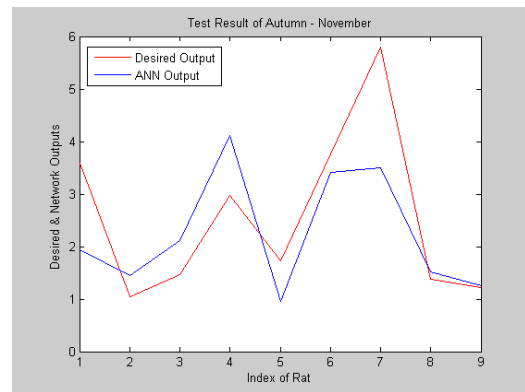


Figure 4.2.18: Desired and ANN Output of November

Training performance of this experiment is better than previous experiments.

Modeling of the general seasonal behavior considering each of the first seven minutes immobility

The inputs of the MLP are adding imm_{17} (duration of immobility in seventh minute) to previous simulation inputs. The outputs aren't change. All output and input data is given A.7 and A.11 tables in appendix. The structure of ANN is same only input layer is enlarged because of increasing of inputs by one.

The performance of ANN is remarkably getting better than the previous simulations by adding imm_{17} to inputs. Maybe, this value is worth parameters for simulations. The main criteria of performance that is mean square error is 0.0011 at last iteration.

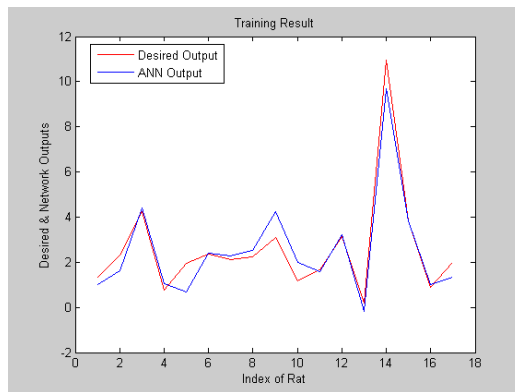


Figure 4.2.19: Desired and ANN Output

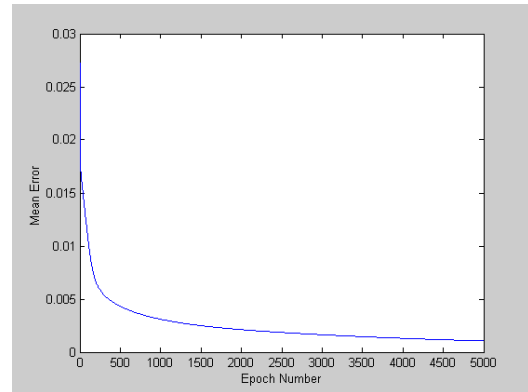


Figure 4.2.20: Mean Square Error of Training Phase

In test phase, updated structure of the ANN is trailed for four seasonal rat groups. On the contrary, any improvement can't observe in test phase. This situation has many causes such as unrelated data patterns, individual of each rat, seasonal difference ...etc.

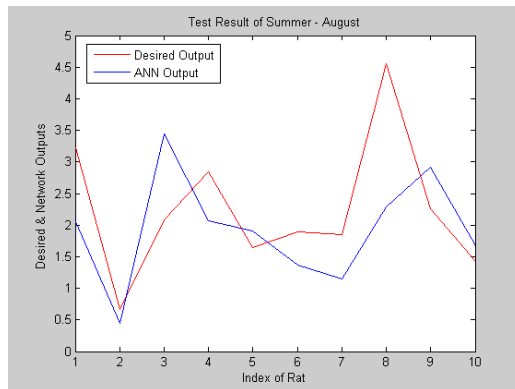


Figure 4.2.21: Desired and ANN Output of August

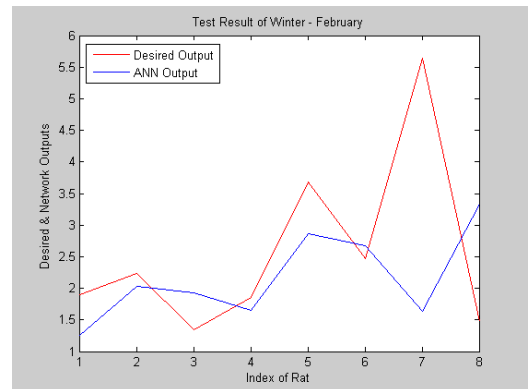


Figure 4.2.22: Desired and ANN Output of February

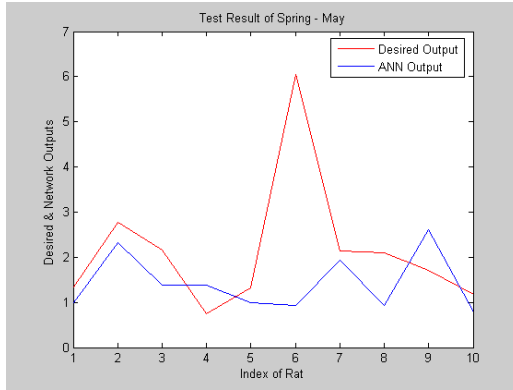


Figure 4.2.23: Desired and ANN Output of May

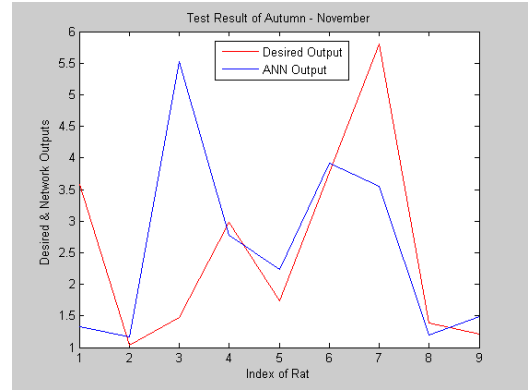


Figure 4.2.24: Desired and ANN Output of November

Modeling of the general seasonal behavior considering each of the first eight minutes immobility

Inputs are increased by adding imm_{18} to previous input pattern. Thus, the duration of first eight minutes of PST1 are applied the ANN as inputs. Number of nodes increased proportionally by enlarging inputs. The input and output data are given in A.7 and A.12 tables in appendix.

The performance criteria of these experiments are same like previous simulation. The last mean error is 0.0011 at 5000 iteration. Outputs and mean square error plotted on below graphs.

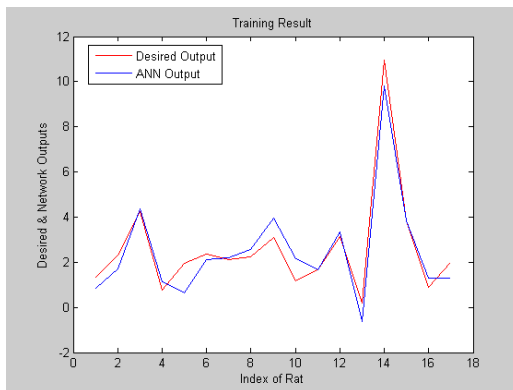


Figure 4.2.25: Desired and ANN Output

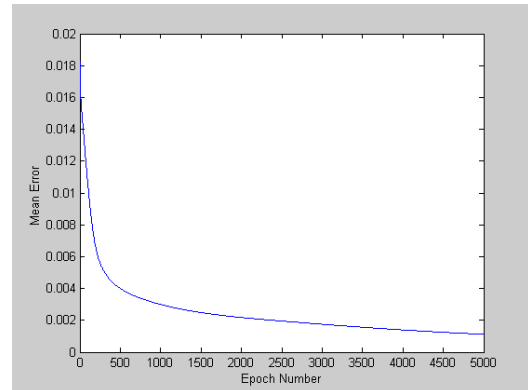


Figure 4.2.26: Mean Square Error of Training Phase

The new updated structure of MLP is trialed for each seasonal group. Results are showed below graphs sequentially. But expected improvement doesn't occur in test phase in spite of good performance of training.

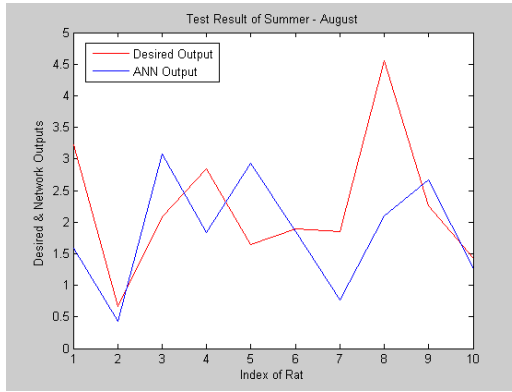


Figure 4.2.27: Desired and ANN Output of August

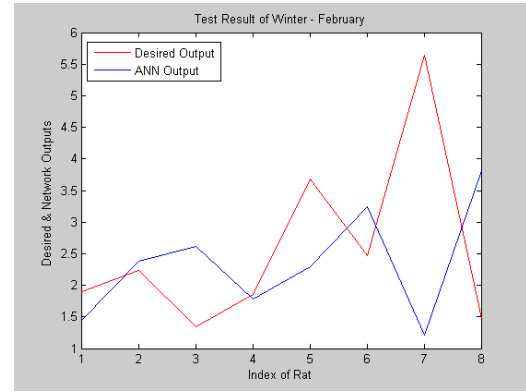


Figure 4.2.28: Desired and ANN Output of February

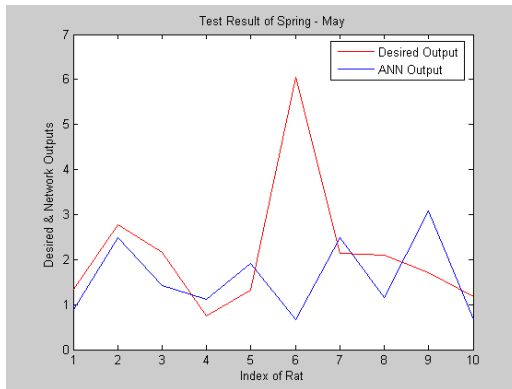


Figure 4.2.29: Desired and ANN Output of May

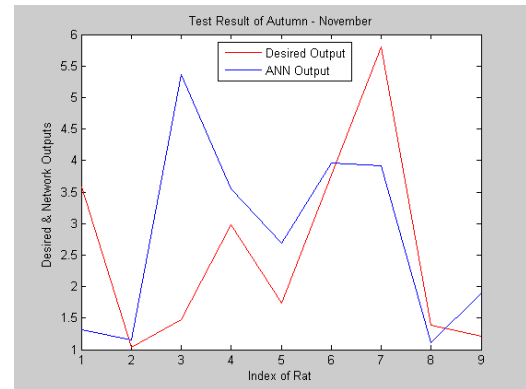


Figure 4.2.30: Desired and ANN Output of November

Modeling of the general seasonal behavior considering each of the first nine minutes immobility

The inputs of ANN are increased with added the duration immobilization in ninth minutes. Then, size of input vectors is nine and first nine immobilization value of PST1 are applied as inputs. Number of sensory neurons in input layer gets increase to 9. In training phase, weights of the MLP are updated to get better results. Output of the ANN doesn't change are BD ratio of rats. The input and output data are given in A.7 and A.13 tables in appendix.

The performance of the ANN in this simulation get better remarkably and last mean error is 0, 00055 nearly zero. Also, result of training phase is showed graphs.

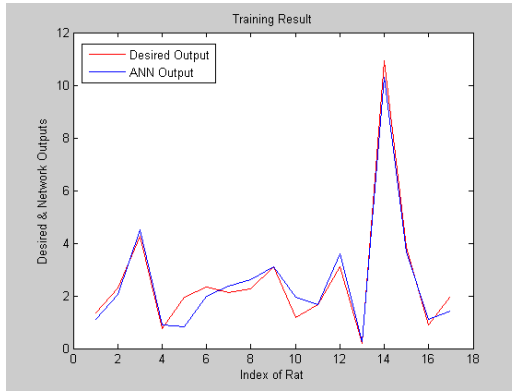


Figure 4.2.31: Desired and ANN Output

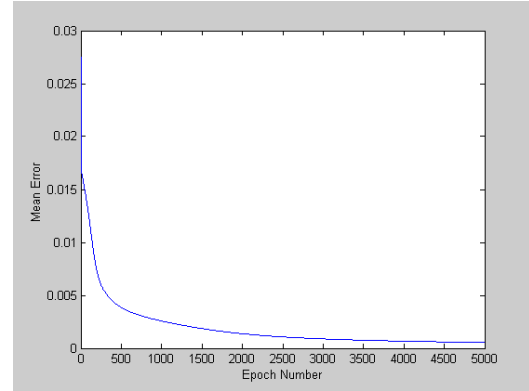


Figure 4.2.32: Mean Square Error of Training Phase

Unfortunately, in test phase expected performance doesn't occur. We can repeat same reasons which are declared before. Results fairly get better, the output of the ANN are almost converge desired outputs. The main performance criteria which is mean square error equal to 0.00054 is guaranteed

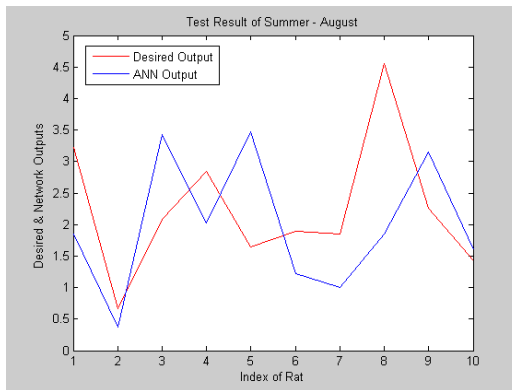


Figure 4.2.33: Desired and ANN Output of August

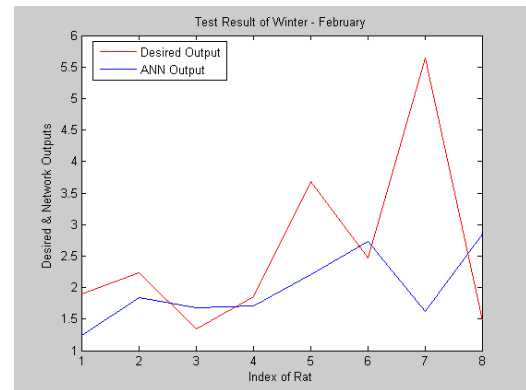


Figure 4.2.34: Desired and ANN Output of February

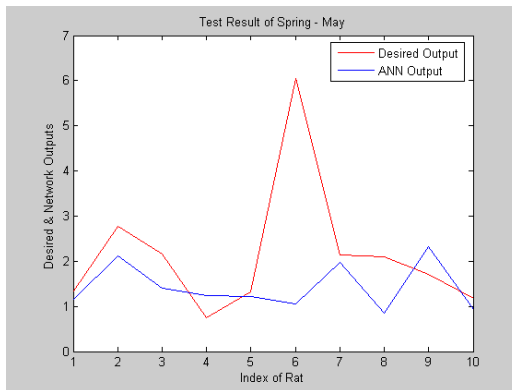


Figure 4.2.35: Desired and ANN Output of May

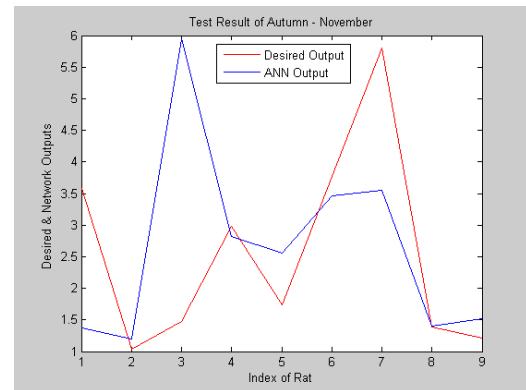


Figure 4.2.36: Desired and ANN Output of November

Modeling of the general seasonal behavior considering each of the first ten minutes immobility

This simulation is last of series of simulation. Inputs of the ANN are the first 10 minutes of PST1 which are 15 minute totally. Outputs are same previous simulations BD ratio of 17 rats. All data are normalized between 0.9 and -0.9 range because of sigmoid activation function. The performance of this simulation are nearly same previous. Value of last mean error is 0.00054. The input and output data are given in A.7 and A.14 tables in appendix.

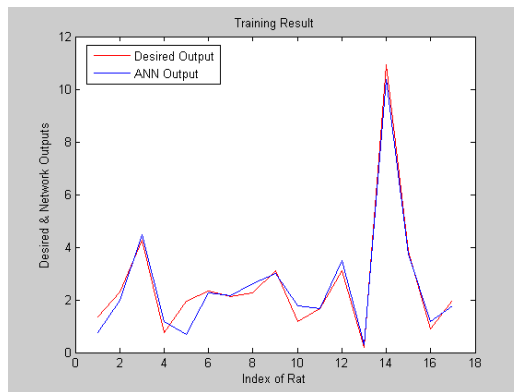


Figure 4.2.37: Desired and ANN Output

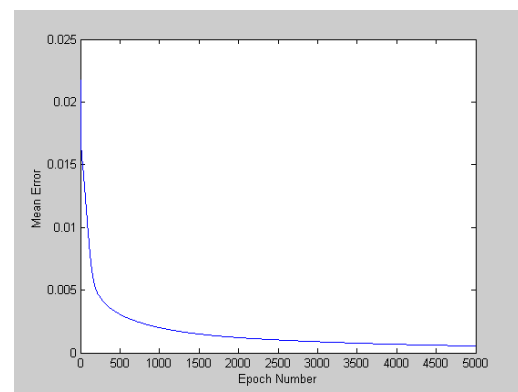


Figure 4.2.38: Mean Square Error of Training Phase

After training phase, the updated structure of ANN is tested for each seasonal rat groups. Results of test are given below graphs.

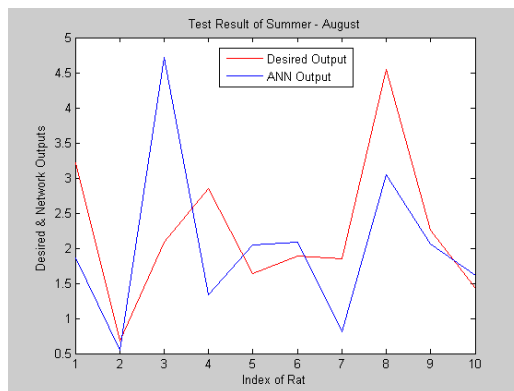


Figure 4.2.39: Desired and ANN Output of August

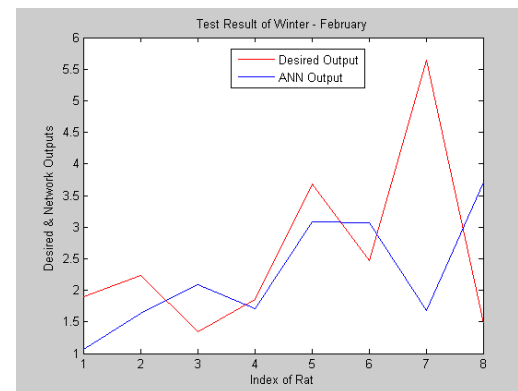


Figure 4.2.40: Desired and ANN Output of February

Results of test phase aren't reasonable as we expected. Only, a slightly improvement can be seen in comparison to previous modeling results.

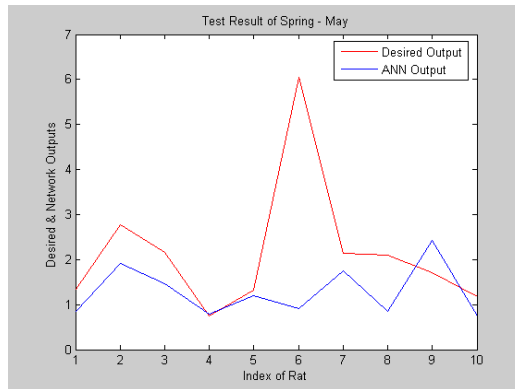


Figure 4.2.41: Desired and ANN Output of May

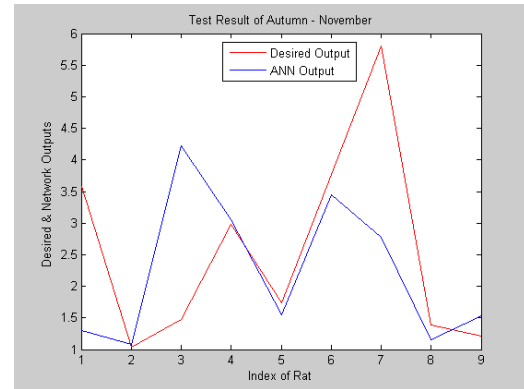


Figure 4.2.42: Desired and ANN Output of November

Considering all these simulations, we can conclude that even though more number of inputs has positive effect still the test results are poor. This is due to the irrelevance of training and test data.

4.2.2 Determining the effect of initial weights and neuron number

In this section we investigated whether it is possible to obtain an ANN based model to predict the behavioral despair (BD) ratio for all seasons. The aim was to seek an answer to the following questions: Is there a general model for all seasons? How does BD ratio vary according to seasons? What are the impacts of seasonal factors on BD?

For training set, the data used in section 4.1 and 4.2.1 was used but only considering the 16 rat data excluding the one with extreme value. For test set data of 37 rats which attain Porsolt test in different seasons are used. Thus test set is composed of four different groups: the first group contains data of 10 rat which attain Porsolt test in May, the second group contains data of nine rats that took the Porsolt test in November, the third group contains data of 10 rats and the fourth group contains the data of eight rat which attain Porsolt test in August and February, respectively. During the Porsolt test immobility duration for the first 15 minutes of the first day and for the first five minutes of the second day is kept.

The ANN structure used is three layered multilayer perceptron and while its inputs are changing according to different data sets for hidden layer neurons four different cases are considered and three, five, seven and ten neurons are used. For output layer only one neuron is used, BD ratio is the only parameter to be predicted. During training all four different cases of hidden layer neurons are considered and the learning rule is error backpropagation rule.

In this part of the thesis only immobility is considered as input and the effect of different combination of immobility durations on the prediction of BD ratio is investigated. Four different data set with four different inputs are constructed, in the first data set the inputs are the immobility durations in the first fourth, fifth, sixth and seventh minutes and these are indicated by imm4, imm5, imm6 and imm7, respectively. In the second data set, this time difference between the immobility duration during the fourth minute and fifth minute (imm5-imm4) is considered as input, in the third data set this time difference between the immobility duration during the fifth minute and sixth minute (imm6-imm5) is considered and in the fourth data set, both the difference between the immobility duration during the fourth minute and fifth minute (imm5-imm4) and the immobility duration during the fifth minute and sixth minute (imm6-imm5) are considered as inputs. Thus, for data set one there would be four inputs while for the second and third there is only one input and for

the fourth data set there is two inputs. The simulations are done for these four different data set and each time four different hidden layer neuron numbers are considered and in order to see the effect of initial weights simulations are repeated 20 times with different randomly selected initial weight values. Once the training phase is completed considering the 16 rat data, test phase is conducted for four different seasons and considering the 37 rat in total.

All training phase results for these simulations are summarized in Table 4.2.1 and Table 4.2.2, where mean squared error and standard deviations are given. In these tables sim_k , where k changes from 1 to 20 denotes each trial with different initial weight values and the means and deviations are for 16 different data values. For each data set besides running 20 times to see the effect of initial weight values, simulations are repeated for four different neuron numbers. The stopping criteria is again the number of iterations which is chosen to be 5000, like in the previous sections. Each time a simulation is completed the mean error is calculated according to the equation given in below.

$$e_{avg} = \frac{1}{16} \frac{1}{2} \sum_1^{16} (y_d - y_{MLP})^2 \quad (4.2a)$$

The performance of the training phase is evaluated considering the mean error, and in order to improve this performance the weights are updated to reduce the value of the mean error. This performance depends on parameters as the number of hidden layer neurons, number of hidden layers, learning rate, the learning rule, etc. Here only the effect of number of hidden layer neurons and the initial weight values are investigated.

Table 4.2.1: Mean squared error and standard deviation, Training set results for 3 and 5 hidden neurons

Number of Hidden Neurons	Simulation index	Data Set 1	Data Set 2	Data Set 3	Data Set 4
3	Sim1	0.0569 \pm 0.0829	0.4909 \pm 0.7866	0.5000 \pm 0.7489	0.3973 \pm 0.6428
	Sim2	0.0631 \pm 0.1367	0.4909 \pm 0.7866	0.3936 \pm 0.6116	0.3971 \pm 0.6420
	Sim3	0.0496 \pm 0.1090	0.4909 \pm 0.7866	0.5000 \pm 0.7493	0.3970 \pm 0.6415
	Sim4	0.0526 \pm 0.1038	0.4909 \pm 0.7866	0.3942 \pm 0.6158	0.3971 \pm 0.6423
	Sim5	0.0524 \pm 0.1186	0.4909 \pm 0.7866	0.3938 \pm 0.6154	0.4892 \pm 0.7726
	Sim6	0.0320 \pm 0.0520	0.4909 \pm 0.7866	0.3939 \pm 0.6140	0.3970 \pm 0.6413
	Sim7	0.0525 \pm 0.0843	0.4909 \pm 0.7866	0.3940 \pm 0.6156	0.4891 \pm 0.7731
	Sim8	0.0469 \pm 0.0770	0.4909 \pm 0.7866	0.5000 \pm 0.7489	0.3976 \pm 0.6457
	Sim9	0.0449 \pm 0.0951	0.4909 \pm 0.7866	0.3938 \pm 0.6151	0.4003 \pm 0.6580
	Sim10	0.0276 \pm 0.0367	0.4909 \pm 0.7866	0.3945 \pm 0.6157	0.3974 \pm 0.6438
	Sim11	0.0520 \pm 0.1319	0.4909 \pm 0.7866	0.3940 \pm 0.6157	0.3973 \pm 0.6427
	Sim12	0.0568 \pm 0.0917	0.4909 \pm 0.7866	0.3937 \pm 0.6136	0.3975 \pm 0.6430
	Sim13	0.0720 \pm 0.0931	0.4909 \pm 0.7866	0.3936 \pm 0.6127	0.4892 \pm 0.7727
	Sim14	0.0605 \pm 0.1298	0.4909 \pm 0.7866	0.3933 \pm 0.6090	0.3970 \pm 0.6419
	Sim15	0.0536 \pm 0.0973	0.4909 \pm 0.7866	0.3940 \pm 0.6142	0.3972 \pm 0.6424
	Sim16	0.0558 \pm 0.0834	0.4909 \pm 0.7866	0.3947 \pm 0.6150	0.3982 \pm 0.6441
	Sim17	0.0469 \pm 0.1170	0.4909 \pm 0.7866	0.4999 \pm 0.7488	0.3985 \pm 0.6443
	Sim18	0.0452 \pm 0.0677	0.4909 \pm 0.7866	0.3943 \pm 0.6149	0.4890 \pm 0.7734
	Sim19	0.0674 \pm 0.1320	0.4910 \pm 0.7867	0.3935 \pm 0.6110	0.3970 \pm 0.6421
	Sim20	0.0551 \pm 0.0925	0.4909 \pm 0.7866	0.3942 \pm 0.6148	0.4891 \pm 0.7731
		Data Set 1	Data Set 2	Data Set 3	Data Set 4
5	Sim1	0.0554 \pm 0.0970	0.4910 \pm 0.7867	0.5000 \pm 0.7492	0.3966 \pm 0.6484
	Sim2	0.0302 \pm 0.0395	0.4909 \pm 0.7866	0.3935 \pm 0.6161	0.3976 \pm 0.6442
	Sim3	0.0255 \pm 0.0516	0.4909 \pm 0.7866	0.3938 \pm 0.6170	0.3973 \pm 0.6529
	Sim4	0.0549 \pm 0.1028	0.4909 \pm 0.7866	0.3938 \pm 0.6164	0.4891 \pm 0.7729
	Sim5	0.0500 \pm 0.0945	0.4909 \pm 0.7867	0.3938 \pm 0.6156	0.3975 \pm 0.6433
	Sim6	0.0380 \pm 0.0854	0.4909 \pm 0.7867	0.3926 \pm 0.6125	0.3971 \pm 0.6459
	Sim7	0.0447 \pm 0.0864	0.4909 \pm 0.7866	0.3924 \pm 0.6104	0.3964 \pm 0.6502
	Sim8	0.0261 \pm 0.0449	0.4909 \pm 0.7866	0.5001 \pm 0.7496	0.4892 \pm 0.7724
	Sim9	0.0524 \pm 0.1134	0.4913 \pm 0.7868	0.3933 \pm 0.6160	0.4893 \pm 0.7721
	Sim10	0.0185 \pm 0.0552	0.4914 \pm 0.7869	0.5001 \pm 0.7496	0.3972 \pm 0.6484
	Sim11	0.0226 \pm 0.0335	0.4910 \pm 0.7867	0.3933 \pm 0.6155	0.3969 \pm 0.6475
	Sim12	0.0251 \pm 0.0483	0.4909 \pm 0.7866	0.3924 \pm 0.6119	0.4893 \pm 0.7722
	Sim13	0.0474 \pm 0.0970	0.4909 \pm 0.7866	0.3934 \pm 0.6145	0.3975 \pm 0.6477
	Sim14	0.0380 \pm 0.0737	0.4909 \pm 0.7866	0.3973 \pm 0.6243	0.3973 \pm 0.6433
	Sim15	0.0197 \pm 0.0506	0.4909 \pm 0.7866	0.3933 \pm 0.6147	0.3972 \pm 0.6428
	Sim16	0.0488 \pm 0.0937	0.4909 \pm 0.7866	0.3937 \pm 0.6174	0.3953 \pm 0.6546
	Sim17	0.0239 \pm 0.0418	0.4912 \pm 0.7868	0.3938 \pm 0.6169	0.3970 \pm 0.6454
	Sim18	0.0179 \pm 0.0440	0.4909 \pm 0.7866	0.3937 \pm 0.6149	0.3980 \pm 0.6451
	Sim19	0.0159 \pm 0.0395	0.4910 \pm 0.7867	0.3936 \pm 0.6149	0.3975 \pm 0.6474
	Sim20	0.0543 \pm 0.1104	0.4909 \pm 0.7866	0.5002 \pm 0.7499	0.4892 \pm 0.7724

When hidden layer neuron number is three the best result for data set 1, where the inputs are immobilities at fourth, fifth, sixth and seventh minutes, is obtained at simulation 10 (mean error= 0.0276 ± 0.0367). The worst result for this case is obtained at simulation 13 (mean error = 0.0720 ± 0.0931). This difference points out the effect of initial weight values on the results. For data set 2, there is almost no difference between the results and it seems that initial weight values do not affect the results very much. In this case the worst case is at simulation 19 (mean error= 0.4910 ± 0.7867), the all other have the same error value which is not much different than this result (mean error= 0.4909 ± 0.7866). For data set 3, the best result is obtained at simulation 14 (error= 0.3933 ± 0.6090) and the worst result is obtained at simulation three (mean error= 0.5000 ± 0.7493). The last data set is data set 4 and for this data set the best result is obtained at simulation 6 (mean error= 0.3970 ± 0.6413) and the worst result is obtained at simulation 5 (mean error= 0.4892 ± 0.7726). For neuron number three, when all four different data sets are considered the best results are obtained for the first data set, while worst results are obtained for data set 2. Thus, as data set, data set 1 is better than others so for the prediction of BD ratio, to consider immobilities at fourth, fifth, sixth and seventh minutes give better results than considering differences of immobilities as in data sets 2, 3 and 4.

Now the case when the hidden layer neuron is five will be investigated. For data set 1, the initial weight values are effective; the best result is obtained in simulation 19 (mean error= 0.0159 ± 0.0395), the worst case is obtained in simulation 1 (mean error= 0.0554 ± 0.0970). For the data set 2, the initial weight values are not much effective the best (mean error = 0.4914 ± 0.7869) and worst (mean error = 0.4909 ± 0.7866) results are obtained in simulation 10, and in many simulations, respectively. The initial conditions are also effective on the results obtained for the data set 3. The best result is obtained in simulation 7 and 12 (mean error= 0.3924 ± 0.6104). The worst result is obtained in simulation 20 where the mean error was 0.5002 ± 0.7499 . For the data set 4, the results did not change much with the initial values, the best result is obtained in simulation 14 (mean error = 0.3970 ± 0.6419) and the worst result is obtained in simulation 13 (mean error= 0.4892 ± 0.7727). When we evaluate the over all results for this four different data set, the best results are obtained for the data set1 and the worst results are obtained for the data set 2.

Table 4.2.2: Mean squared error and standard deviation, Training set results for 7 and 10 hidden neurons

Number of Hidden Neurons	Simulation index	Data Set 1	Data Set 2	Data Set 3	Data Set 4
7	Sim1	0.0117 ± 0.0338	0.4909 ± 0.7866	0.3928 ± 0.6161	0.3956 ± 0.6511
	Sim2	0.0254 ± 0.0557	0.4913 ± 0.7868	0.3945 ± 0.6204	0.3977 ± 0.6477
	Sim3	0.0301 ± 0.0653	0.4909 ± 0.7866	0.3918 ± 0.6169	0.3972 ± 0.6430
	Sim4	0.0215 ± 0.0542	0.4914 ± 0.7869	0.3937 ± 0.6186	0.3941 ± 0.6634
	Sim5	0.0198 ± 0.0518	0.4910 ± 0.7867	0.3943 ± 0.6172	0.3972 ± 0.6484
	Sim6	0.0290 ± 0.0636	0.4911 ± 0.7867	0.3930 ± 0.6149	0.3977 ± 0.6441
	Sim7	0.0162 ± 0.0336	0.4909 ± 0.7866	0.3917 ± 0.6199	0.3966 ± 0.6508
	Sim8	0.0256 ± 0.0596	0.4913 ± 0.7868	0.3917 ± 0.6199	0.3980 ± 0.6473
	Sim9	0.0169 ± 0.0454	0.4909 ± 0.7866	0.3958 ± 0.6206	0.3980 ± 0.6448
	Sim10	0.0227 ± 0.0610	0.4912 ± 0.7868	0.3924 ± 0.6193	0.3950 ± 0.6568
	Sim11	0.0190 ± 0.0486	0.4909 ± 0.7866	0.3929 ± 0.6178	0.3964 ± 0.6503
	Sim12	0.0250 ± 0.0515	0.4909 ± 0.7866	0.3937 ± 0.6164	0.3977 ± 0.6446
	Sim13	0.0486 ± 0.1012	0.4909 ± 0.7866	0.3944 ± 0.6185	0.3946 ± 0.6590
	Sim14	0.0626 ± 0.1177	0.4909 ± 0.7866	0.3925 ± 0.6171	0.3936 ± 0.6614
	Sim15	0.0188 ± 0.0381	0.4909 ± 0.7866	0.3936 ± 0.6157	0.3943 ± 0.6575
	Sim16	0.0264 ± 0.0519	0.4909 ± 0.7866	0.3923 ± 0.6176	0.3959 ± 0.6530
	Sim17	0.0251 ± 0.0536	0.4909 ± 0.7866	0.3921 ± 0.6174	0.3946 ± 0.6598
	Sim18	0.0359 ± 0.0684	0.4909 ± 0.7866	0.3922 ± 0.6172	0.3969 ± 0.6472
	Sim19	0.0499 ± 0.1189	0.4910 ± 0.7867	0.3946 ± 0.6200	0.3957 ± 0.6526
	Sim20	0.0147 ± 0.0435	0.4909 ± 0.7866	0.3925 ± 0.6163	0.3961 ± 0.6521
		Data Set 1	Data Set 2	Data Set 3	Data Set 4
10	Sim1	0.0286 ± 0.0509	0.4909 ± 0.7866	0.3925 ± 0.6201	0.3981 ± 0.6457
	Sim2	0.0493 ± 0.1020	0.4911 ± 0.7867	0.3907 ± 0.6288	0.3946 ± 0.6595
	Sim3	0.0148 ± 0.0397	0.4912 ± 0.7868	0.3929 ± 0.6197	0.3969 ± 0.6480
	Sim4	0.0162 ± 0.0417	0.4910 ± 0.7867	0.3909 ± 0.6263	0.3921 ± 0.6701
	Sim5	0.0228 ± 0.0570	0.4911 ± 0.7868	0.3912 ± 0.6239	0.3922 ± 0.6681
	Sim6	0.0175 ± 0.0449	0.4909 ± 0.7866	0.3915 ± 0.6226	0.3933 ± 0.6673
	Sim7	0.0321 ± 0.0692	0.4909 ± 0.7866	0.3910 ± 0.6264	0.3929 ± 0.6674
	Sim8	0.0166 ± 0.0497	0.4909 ± 0.7866	0.3909 ± 0.6268	0.3947 ± 0.6539
	Sim9	0.0164 ± 0.0463	0.4909 ± 0.7866	0.3911 ± 0.6251	0.3937 ± 0.6590
	Sim10	0.0204 ± 0.0566	0.4912 ± 0.7868	0.3933 ± 0.6186	0.3920 ± 0.6709
	Sim11	0.0155 ± 0.0495	0.4911 ± 0.7867	0.3968 ± 0.6230	0.3945 ± 0.6543
	Sim12	0.0236 ± 0.0541	0.4909 ± 0.7866	0.3909 ± 0.6269	0.3949 ± 0.6565
	Sim13	0.0138 ± 0.0419	0.4915 ± 0.7869	0.3914 ± 0.6251	0.3938 ± 0.6579
	Sim14	0.0198 ± 0.0550	0.4910 ± 0.7867	0.5001 ± 0.7501	0.3938 ± 0.6613
	Sim15	0.0476 ± 0.0969	0.4909 ± 0.7866	0.3928 ± 0.6191	0.3974 ± 0.6494
	Sim16	0.0235 ± 0.0483	0.4909 ± 0.7866	0.3911 ± 0.6252	0.3945 ± 0.6559
	Sim17	0.0220 ± 0.0455	0.4910 ± 0.7867	0.3921 ± 0.6214	0.4001 ± 0.6594
	Sim18	0.0196 ± 0.0575	0.4913 ± 0.7869	0.3948 ± 0.6184	0.3952 ± 0.6531
	Sim19	0.0168 ± 0.0387	0.4909 ± 0.7866	0.3921 ± 0.6213	0.3971 ± 0.6476
	Sim20	0.0255 ± 0.0558	0.4911 ± 0.7867	0.3930 ± 0.6209	0.3966 ± 0.6493

Similar to Table 4.2.1 in Table 4.2.2, the results obtained for the four different data set are given but this time the neuron number is 7 and 10. In order to investigate effect of initial weight values, the results are obtained for randomly chosen 20 different initial weight values.

First we will investigate the case when the hidden neuron number is seven. While the data set is 1, the initial weight values did affect the performance of the ANN and the best result is obtained in the first simulation (mean error = 0.0117 ± 0.0338) and the worst results are obtained in the fourteenth simulation (mean error = 0.0626 ± 0.1177). For the data set 2, the initial weight values did not affect the performance much and while the best result (mean error = 0.4909 ± 0.7866) is obtained for more than one simulation the worst case (mean error = 0.4910 ± 0.7867) is obtained for the fourth simulation. As can be followed from the mean error values there is only a minor difference between the worst and best cases. For data set 3, again the initial weight values did not affect much and the mean error varied in the interval $\{0.3943 \pm 0.6172, 0.3917 \pm 0.6199\}$. Similar results obtained for the data set 4, for which the mean error is varied in the interval $\{0.3941 \pm 0.6634 \text{ and } 0.3977 \pm 0.6446\}$.

Now the case when the hidden layer neuron number is 10 will be considered. In this case, for the data set 1 the best result is obtained in the 13th simulation (mean error = 0.0138 ± 0.0419) and the worst case is obtained in the second simulation (mean error = 0.0493 ± 0.1020). While for the data set 1 the initial weight values do affect the performance, for the data set 2, they are not much effective since for this case the mean error varies in the interval $\{0.4909 \pm 0.7866, 0.4915 \pm 0.7869\}$. For data set 3, the best result is obtained in the second simulation (mean error = 0.3907 ± 0.6288) and the worst result is obtained in the 14th simulation (mean error = 0.5001 ± 0.7501). There is not much variation in the mean error values for the data set 4, the worst result is obtained in the 17th simulation (mean error = 0.4001 ± 0.6594) and the best result is obtained in the 10th simulation (mean error = 0.3920 ± 0.6709) .

Now considering the results given in Table 4.2.2 an overall evaluation of the results will be given. The best result is obtained when the neuron number is seven and the data set is the data set 1 and the simulation is the first simulation. The increase in the neuron number did not affect the results much thus do not have much effect on the performance of the ANN model. Still for neuron number three the results are somewhat worse when compared to others. There is almost no difference between the neuron number five, seven and ten. The effect of initial weight values on the results is more significant than neuron number. For data set 2 there is almost no change in the results both for different number of neurons and different initial weight values. For data set 3, while the best result is obtained for neuron number 10, the initial weight values affected the performance more. For the data set 4, again the best result is obtained when the hidden neuron number is 10, but still the increase in the number of neurons did not cause very significant change in the performance.

When we consider the same data sets but this time focusing on the different neuron numbers, the following results are obtained: as the neuron number is increased with data set 1, for some initial conditions better results are obtained and the mean error is decreased on the general. Whereas for the data set 2, the neuron number does not much affect the performance. In a similar way, the neuron number is not much effective for the data set 3 and 4.

The performance of the training phase is mostly dependent on the data, the initial weight values is somewhat effective while the number of hidden layer neuron number is the least significant parameter on the performance of the results.

Table 4.2.3: Mean squared error and standard deviation, Test set results for 3 and 5 hidden neurons

Number of Hidden Neurons	Simulation index	Data Set 1	Data Set 2	Data Set 3	Data Set 4
3	Sim1	0.3070 ± 0.3817	0.1404 ± 0.1407	0.1126 ± 0.1447	0.1265 ± 0.2713
	Sim2	0.4471 ± 0.4036	0.1404 ± 0.1407	0.1252 ± 0.2332	0.1283 ± 0.2748
	Sim3	0.3509 ± 0.4498	0.1404 ± 0.1406	0.1127 ± 0.1445	0.1279 ± 0.2736
	Sim4	0.3570 ± 0.4306	0.1404 ± 0.1407	0.1253 ± 0.2354	0.1253 ± 0.2687
	Sim5	0.3727 ± 0.4367	0.1404 ± 0.1407	0.1254 ± 0.2383	0.1133 ± 0.1930
	Sim6	0.2793 ± 0.2662	0.1404 ± 0.1407	0.1255 ± 0.2432	0.1288 ± 0.2754
	Sim7	0.3181 ± 0.3672	0.1404 ± 0.1407	0.1254 ± 0.2386	0.1131 ± 0.1929
	Sim8	0.2965 ± 0.3079	0.1404 ± 0.1407	0.1126 ± 0.1447	0.1255 ± 0.2727
	Sim9	0.2623 ± 0.2916	0.1404 ± 0.1407	0.1254 ± 0.2367	0.1295 ± 0.2771
	Sim10	0.2370 ± 0.2483	0.1404 ± 0.1406	0.1254 ± 0.2394	0.1254 ± 0.2725
	Sim11	0.4830 ± 0.4291	0.1403 ± 0.1406	0.1254 ± 0.2381	0.1234 ± 0.2648
	Sim12	0.3422 ± 0.3965	0.1404 ± 0.1407	0.1253 ± 0.2343	0.1243 ± 0.2669
	Sim13	0.3174 ± 0.3858	0.1404 ± 0.1407	0.1253 ± 0.2337	0.1132 ± 0.1928
	Sim14	0.3212 ± 0.3328	0.1404 ± 0.1406	0.1252 ± 0.2323	0.1270 ± 0.2721
	Sim15	0.2796 ± 0.3018	0.1404 ± 0.1407	0.1255 ± 0.2426	0.1262 ± 0.2717
	Sim16	0.3724 ± 0.3814	0.1404 ± 0.1407	0.1254 ± 0.2407	0.1233 ± 0.2636
	Sim17	0.4125 ± 0.3483	0.1404 ± 0.1407	0.1126 ± 0.1447	0.1203 ± 0.2568
	Sim18	0.3822 ± 0.4187	0.1404 ± 0.1407	0.1253 ± 0.2336	0.1131 ± 0.1930
	Sim19	0.4423 ± 0.4088	0.1403 ± 0.1406	0.1252 ± 0.2329	0.1268 ± 0.2727
	Sim20	0.3053 ± 0.3462	0.1404 ± 0.1407	0.1254 ± 0.2415	0.1132 ± 0.1930
		Data Set 1	Data Set 2	Data Set 3	Data Set 4
5	Sim1	0.3289 ± 0.3648	0.1403 ± 0.1406	0.1127 ± 0.1445	0.1183 ± 0.2495
	Sim2	0.2732 ± 0.3114	0.1404 ± 0.1407	0.1261 ± 0.2266	0.1239 ± 0.2670
	Sim3	0.3060 ± 0.3359	0.1404 ± 0.1407	0.1255 ± 0.2344	0.1213 ± 0.2549
	Sim4	0.3426 ± 0.3655	0.1403 ± 0.1406	0.1255 ± 0.2326	0.1132 ± 0.1929
	Sim5	0.2982 ± 0.3604	0.1403 ± 0.1406	0.1255 ± 0.2327	0.1272 ± 0.2732
	Sim6	0.4150 ± 0.4398	0.1403 ± 0.1406	0.1281 ± 0.2176	0.1161 ± 0.2458
	Sim7	0.3998 ± 0.4513	0.1404 ± 0.1407	0.1283 ± 0.2169	0.1177 ± 0.2472
	Sim8	0.2768 ± 0.2658	0.1404 ± 0.1407	0.1128 ± 0.1443	0.1133 ± 0.1933
	Sim9	0.3774 ± 0.3857	0.1402 ± 0.1405	0.1262 ± 0.2259	0.1135 ± 0.1937
	Sim10	0.3172 ± 0.3564	0.1401 ± 0.1404	0.1128 ± 0.1443	0.1205 ± 0.2561
	Sim11	0.2550 ± 0.2668	0.1403 ± 0.1406	0.1263 ± 0.2251	0.1175 ± 0.2480
	Sim12	0.3319 ± 0.3172	0.1404 ± 0.1407	0.1288 ± 0.2157	0.1134 ± 0.1935
	Sim13	0.3888 ± 0.4351	0.1404 ± 0.1407	0.1258 ± 0.2288	0.1210 ± 0.2570
	Sim14	0.3196 ± 0.3485	0.1403 ± 0.1406	0.1258 ± 0.2307	0.1263 ± 0.2714
	Sim15	0.3406 ± 0.3953	0.1404 ± 0.1407	0.1261 ± 0.2263	0.1244 ± 0.2667
	Sim16	0.3444 ± 0.3574	0.1404 ± 0.1407	0.1263 ± 0.2253	0.1182 ± 0.2444
	Sim17	0.3114 ± 0.3377	0.1402 ± 0.1405	0.1257 ± 0.2304	0.1164 ± 0.2464
	Sim18	0.3208 ± 0.3779	0.1404 ± 0.1407	0.1254 ± 0.2330	0.1225 ± 0.2622
	Sim19	0.3235 ± 0.3742	0.1403 ± 0.1406	0.1262 ± 0.2254	0.1208 ± 0.2572
	Sim20	0.3453 ± 0.3752	0.1404 ± 0.1407	0.1129 ± 0.1441	0.1133 ± 0.1932

While in Tables 4.2.1 and 4.2.2 training set results are investigated, now the test set results will be considered. For the test set data, the data obtained from 37 rat attained to the Porsolt test at different seasons is used. These rat are grouped into

four different groups according to the seasons. These groups include data of 10 rat attained the test during May and August, nine rat attained the test during November and eight rat attained the test during February. Only the immobility behavior of the rat is considered as in the training set.

The weight values obtained at the end of the simulations for each training phase are used to obtain the results for the test set which are given in Table 4.2.3 and 4.2.4 In contrasts to what is expected, the results obtained for the test set are somewhat contrasting to the results obtained for the training set. While the training set results obtained for the data set 1 was good the test results for the same set is poor. The test results for the data set 2,3, and 4 are better even though their training set results were poor. There may be very different factors and reasons for this contrasting results obtained for the training set and test set data as not properly chosen training and test sets, long training phase which decreased the generalization ability of the ANN, the seasonal differences between the training and test sets, the different character properties of the rat, etc. The most important factor could be the difference between the characterizing of the training set and test set: while in the training set the data used belongs to any 16 rat without considering seasonal effect, the test set is composed of data obtained from totally different rat considering the seasonal effect.

Now we will interpret the results given in the Table 4.2.3, where test results for neuron number three and five are depicted. When the neuron number is three and the data set 1 is considered the best test result is obtained for the simulation 10 (mean error = 0.2370 ± 0.2483), the worst result is obtained in simulation 2 (mean error = 0.4471 ± 0.4036). The different weight values are effective for this data set. For the data set 2, this not the case and there is only a slight difference between test results obtained for different weight values and the mean error values are varied in the interval $\{0.1403 \pm 0.1406 \text{ and } 0.1404 \pm 0.1407\}$. For the data set 3 the best results are obtained in simulations 1, 8 and 17 (mean value = 0.1126 ± 0.1447) while the worst result is obtained with the weight values obtained in simulation 6 (mean value= 0.1255 ± 0.2432). For data set 4, the best and worst results are obtained for the weights obtained in the simulation 18 (mean value = 0.1131 ± 0.1930) and 9 (mean value = 0.1295 ± 0.2771), respectively.

The best test result for neuron number five and the data set 1 is obtained with the weight values got at the end of the simulation 11 (mean error = 0.2550 ± 0.2668) and the worst case is obtained for simulation 7 (mean error = 0.3998 ± 0.4513). For

the data set 2, there is not much change in the results obtained for the test set the mean error changes in the interval $\{0.1401 \pm 0.1404, 0.1404 \pm 0.1407\}$. Similarly, for the data set 4 the mean error changes in the interval $\{0.1127 \pm 0.1445 \text{ and } 0.1288 \pm 0.2157\}$.

Table 4.2.4: Mean squared error and standard deviation, Test set results for 7 and 10 hidden neurons

Number of Hidden Neurons	Simulation index	Data Set 1	Data Set 2	Data Set 3	Data Set 4
7	Sim1	0.3249 ± 0.3707	0.1404 ± 0.1407	0.1281 ± 0.2177	0.1165 ± 0.2407
	Sim2	0.3318 ± 0.3839	0.1402 ± 0.1405	0.1262 ± 0.2247	0.1214 ± 0.2597
	Sim3	0.3277 ± 0.3839	0.1404 ± 0.1407	0.1309 ± 0.2114	0.1264 ± 0.2712
	Sim4	0.3293 ± 0.3794	0.1401 ± 0.1404	0.1272 ± 0.2200	0.1213 ± 0.2488
	Sim5	0.3358 ± 0.3905	0.1403 ± 0.1406	0.1257 ± 0.2282	0.1210 ± 0.2580
	Sim6	0.2712 ± 0.3080	0.1403 ± 0.1406	0.1272 ± 0.2208	0.1249 ± 0.2687
	Sim7	0.2946 ± 0.3359	0.1404 ± 0.1407	0.1316 ± 0.2102	0.1184 ± 0.2488
	Sim8	0.3341 ± 0.3702	0.1402 ± 0.1404	0.1315 ± 0.2102	0.1256 ± 0.2720
	Sim9	0.3244 ± 0.3740	0.1404 ± 0.1407	0.1256 ± 0.2281	0.1224 ± 0.2646
	Sim10	0.3399 ± 0.3809	0.1402 ± 0.1405	0.1298 ± 0.2132	0.1205 ± 0.2507
	Sim11	0.2860 ± 0.3280	0.1404 ± 0.1406	0.1279 ± 0.2182	0.1183 ± 0.2481
	Sim12	0.2816 ± 0.2615	0.1403 ± 0.1406	0.1255 ± 0.2342	0.1234 ± 0.2649
	Sim13	0.3585 ± 0.4360	0.1404 ± 0.1407	0.1258 ± 0.2280	0.1208 ± 0.2503
	Sim14	0.3700 ± 0.3549	0.1403 ± 0.1406	0.1291 ± 0.2149	0.1198 ± 0.2451
	Sim15	0.2648 ± 0.2817	0.1404 ± 0.1406	0.1267 ± 0.2226	0.1195 ± 0.2460
	Sim16	0.3196 ± 0.3550	0.1404 ± 0.1406	0.1298 ± 0.2135	0.1190 ± 0.2484
	Sim17	0.3112 ± 0.3257	0.1404 ± 0.1407	0.1303 ± 0.2125	0.1199 ± 0.2462
	Sim18	0.4229 ± 0.3799	0.1404 ± 0.1406	0.1299 ± 0.2131	0.1183 ± 0.2506
	Sim19	0.3763 ± 0.3793	0.1403 ± 0.1406	0.1261 ± 0.2254	0.1178 ± 0.2444
	Sim20	0.3746 ± 0.4243	0.1404 ± 0.1406	0.1287 ± 0.2162	0.1187 ± 0.2482
		Data Set 1	Data Set 2	Data Set 3	Data Set 4
10	Sim1	0.3345 ± 0.3024	0.1404 ± 0.1406	0.1293 ± 0.2142	0.1231 ± 0.2656
	Sim2	0.3592 ± 0.3587	0.1403 ± 0.1406	0.1345 ± 0.2055	0.1203 ± 0.2473
	Sim3	0.3439 ± 0.3921	0.1402 ± 0.1405	0.1281 ± 0.2176	0.1176 ± 0.2485
	Sim4	0.3532 ± 0.4015	0.1403 ± 0.1406	0.1337 ± 0.2067	0.1224 ± 0.2463
	Sim5	0.3220 ± 0.3528	0.1403 ± 0.1405	0.1330 ± 0.2078	0.1221 ± 0.2469
	Sim6	0.2975 ± 0.3087	0.1404 ± 0.1407	0.1322 ± 0.2090	0.1215 ± 0.2454
	Sim7	0.3426 ± 0.3586	0.1404 ± 0.1407	0.1337 ± 0.2068	0.1227 ± 0.2502
	Sim8	0.3340 ± 0.3775	0.1404 ± 0.1406	0.1340 ± 0.2064	0.1168 ± 0.2376
	Sim9	0.3681 ± 0.4112	0.1403 ± 0.1406	0.1332 ± 0.2074	0.1194 ± 0.2436
	Sim10	0.3092 ± 0.3247	0.1402 ± 0.1405	0.1269 ± 0.2223	0.1230 ± 0.2482
	Sim11	0.3628 ± 0.3806	0.1403 ± 0.1405	0.1257 ± 0.2300	0.1164 ± 0.2355
	Sim12	0.3310 ± 0.3833	0.1403 ± 0.1406	0.1339 ± 0.2063	0.1188 ± 0.2440
	Sim13	0.3810 ± 0.4020	0.1401 ± 0.1404	0.1327 ± 0.2080	0.1178 ± 0.2368
	Sim14	0.3084 ± 0.2937	0.1403 ± 0.1406	0.1130 ± 0.1440	0.1202 ± 0.2446
	Sim15	0.4195 ± 0.4335	0.1404 ± 0.1406	0.1288 ± 0.2153	0.1180 ± 0.2488
	Sim16	0.3093 ± 0.3239	0.1404 ± 0.1407	0.1334 ± 0.2072	0.1181 ± 0.2409
	Sim17	0.3024 ± 0.3288	0.1403 ± 0.1406	0.1306 ± 0.2116	0.1273 ± 0.2771
	Sim18	0.3281 ± 0.3357	0.1402 ± 0.1404	0.1256 ± 0.2289	0.1167 ± 0.2382
	Sim19	0.2963 ± 0.3105	0.1404 ± 0.1406	0.1306 ± 0.2116	0.1188 ± 0.2524
	Sim20	0.3562 ± 0.3749	0.1403 ± 0.1405	0.1287 ± 0.2156	0.1170 ± 0.2449

When the hidden layer neuron number is seven the test results for the data set 1 is not good as the results obtained for the training set of this data set. The test results are changing very much with the weight values obtained for each simulation in this group. The variation of mean error is between $\{0.2648 \pm 0.2817$ and $0.4229 \pm 0.3799\}$. For data set 2, better results compared to test results of data set 1 is obtained, in this case there are minor changes in the mean error values varying in the set $\{0.1402 \pm 0.1405, 0.1404 \pm 0.1406\}$. The test results for the data set 3 are better than the expected and for this case the mean error varies in the set $\{0.1255 \pm 0.2342, 0.1316 \pm 0.2102\}$. The best results for the test sets is obtained for the data set 4, where the variation of the mean error is in between $\{0.1165 \pm 0.2407, 0.1256 \pm 0.2720\}$.

For the neuron number 10, while the test results obtained for the data set 2,3 and 4 are better, the results for data set 1 is very poor and for this data set the mean error of the test set changes in the interval $\{0.2963 \pm 0.3105, 0.4195 \pm 0.4335\}$.

After analyzing the overall results, in the following for each of the data sets we illustrate the test and learning results of the simulation with best test performance in the figures.

For the data set 1, the best test result is obtained in ten simulations when the hidden layer neuron number is three. The results for the learning phase are depicted in Figure 4.2.1.

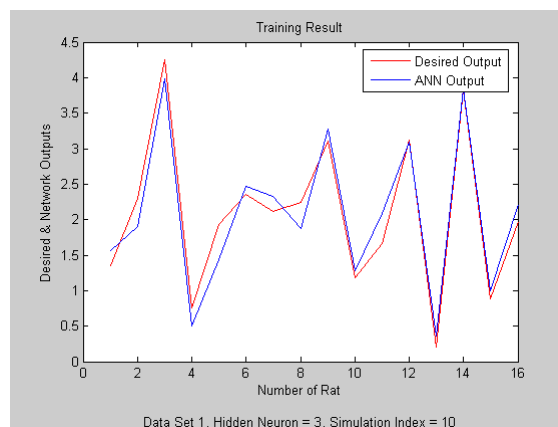


Figure 4.2.43: Outputs of Training Phase

The results obtained from the test phase are plotted separately for each different seasonal group.

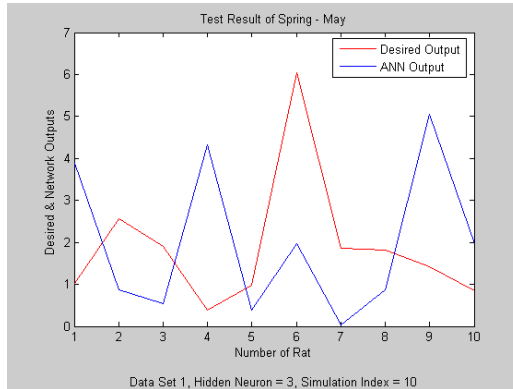


Figure 4.2.44: Test results for Spring

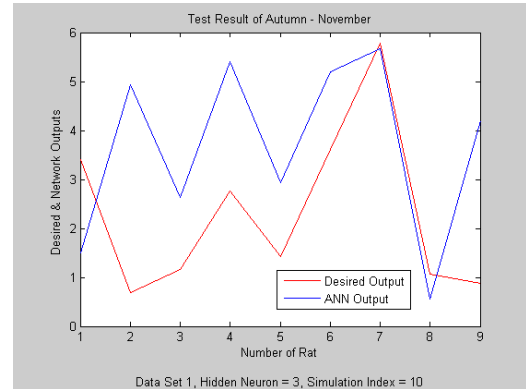


Figure 4.2.45: Test results for Autumn

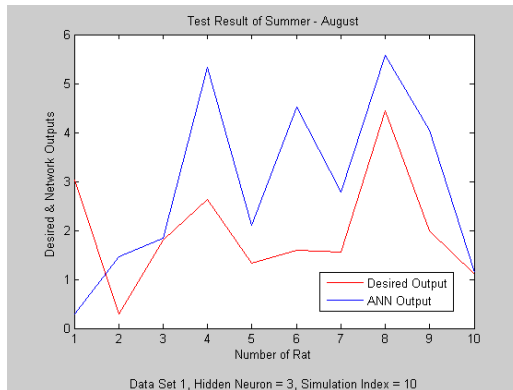


Figure 4.2.46: Test results for Summer

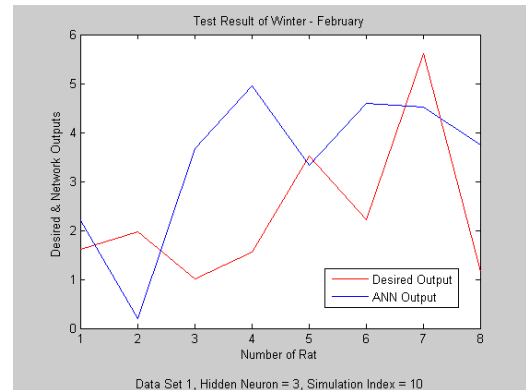


Figure 4.2.47: Test results for Winter

In spite of the good performance in learning phase the test results are not as good as expected. As we mentioned above, this may depend on different factors. Basically, the difference between training and test sets, seasonal differences or the extreme characteristics of rat are some of the possible factors. Because of the unsatisfactory test results we can not claim that the model is appropriate for all seasons.

For the data set 2, the meaningful results can't be obtained in training phase. So it is not possible to forecast the BD ratio by using the data of (Imm4 -Imm5) or in other words these data are not sufficient by themselves.



Figure 4.2.48: Outputs of Training Phase

Because of the bad training performance the test results are also not satisfactory (see Figure 4.2.7 and 4.2.8). Considering the obtained results, one can not propose an ANN model with this data set.

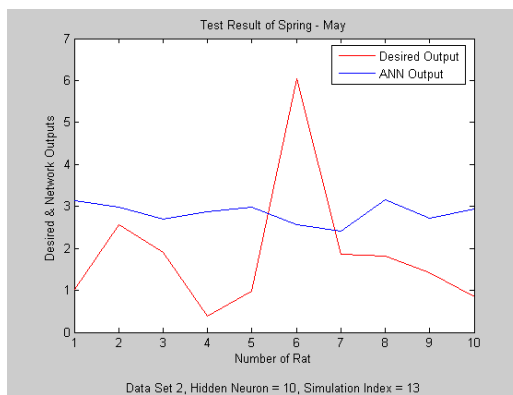


Figure 4.2.49: Test results for Spring

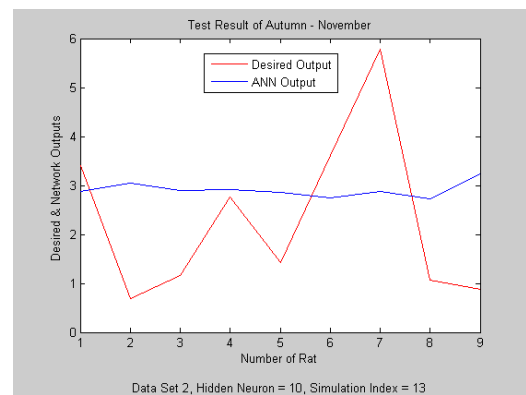


Figure 4.2.50: Test results for Autumn

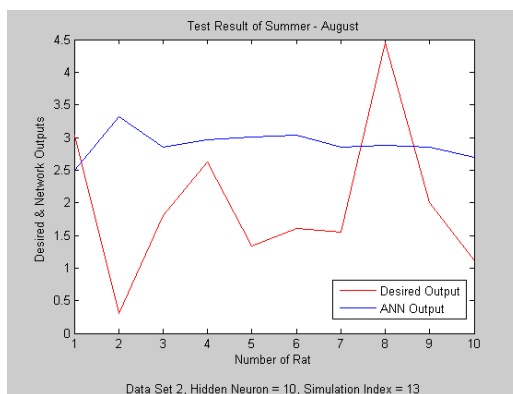


Figure 4.2.51: Test results for Summer

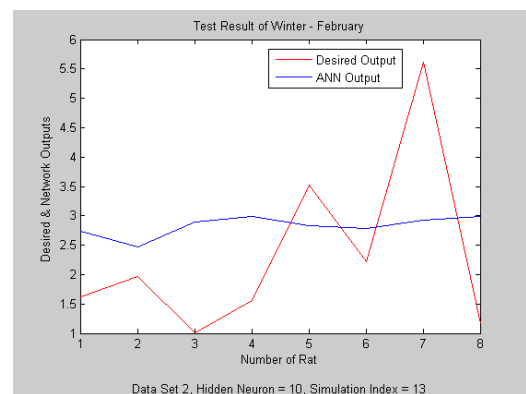


Figure 4.2.52: Test results for winter

For data set3, the BD rate is tried to be estimated using the data of (imm6-imm5). As can be seen from the Figure 4.2.11 the training performance is inconvenient.



Figure 4.2.53: Outputs of Training Phase

The test results are also unsatisfactory because of poor training performance. The results for the different seasons are given below.



Figure 4.2.54: Test results for Spring

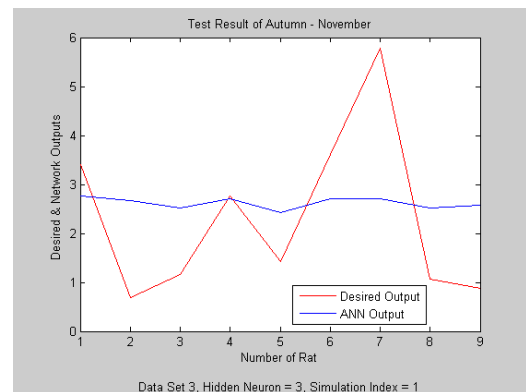


Figure 4.2.55: Test results for Autumn

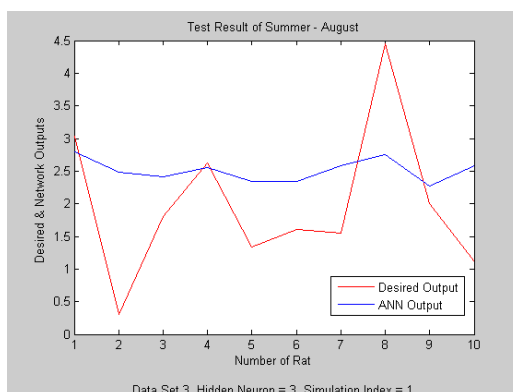


Figure 4.2.56: Test results for Summer

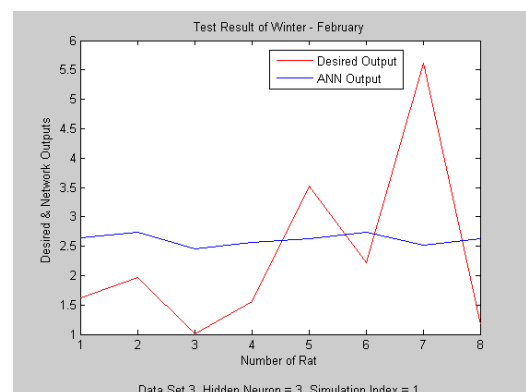


Figure 4.2.57: Test results for Winter

Again, we can not propose a model for BD estimation as the data (Imm5 - Imm6) are insufficient.

The results for the data set 4 are not different from the results for the data set 2 and data set 3.



Figure 4.2.58: Outputs of Training Phase

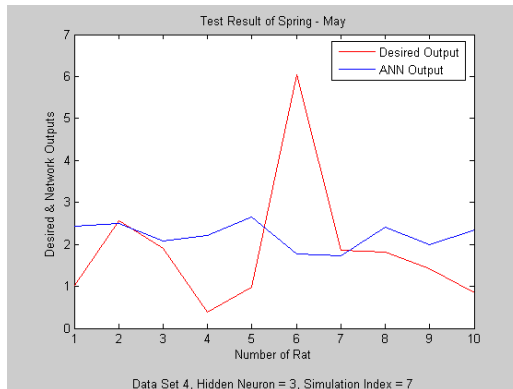


Figure 4.2.59: Test results for Spring

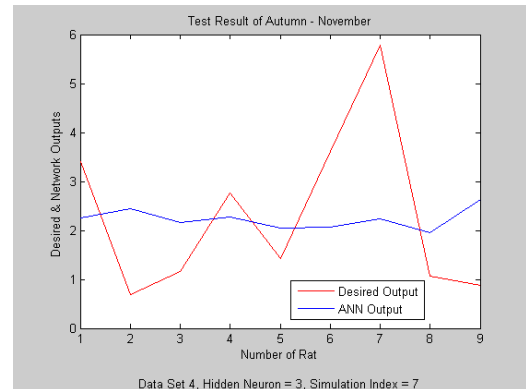


Figure 4.2.60: Test results for Autumn

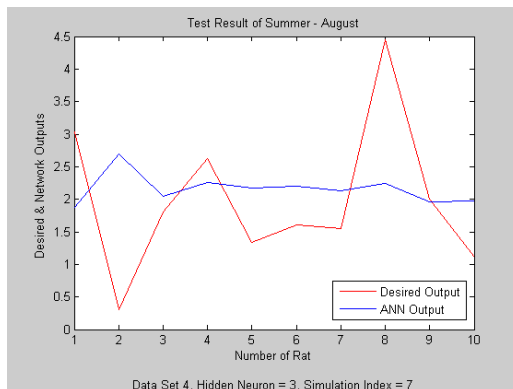


Figure 4.2.61: Test results for Summer

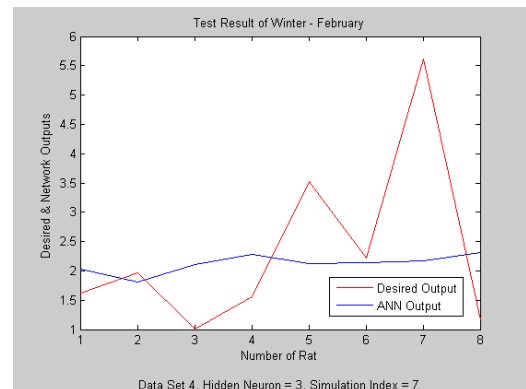


Figure 4.2.62: Test results for Winter

Similarly, we can not propose an ANN model for this data set as both the training and the test results are poor.

4.2.3 Results

In this section of the thesis, two ANN models are realized for the prediction of BD ratio considering only immobility behavior. In the following, we compare these models using the obtained results.

In the first model presented in Section 4.2.1, the duration of immobilities on different minutes were considered. Throughout the study always twenty hidden layer neuron were used. The training set consisted of 17 rats including the extreme one and during the test phase four different rat group for each season are used. The performance of the training phase increased as the number of inputs increased, whereas the test results were unsatisfactory.

In the second model presented in Section 4.2.2 four different cases were considered, namely, 3, 5, 7 and 10 hidden layer neurons and the simulations were performed with 20 different initial weights for each four data sets. Among the simulation results only the one corresponding to the data set 1 was reasonable. It was impossible to obtain an ANN model for BD estimation using the other data sets.

Comparing two models, one can conclude that the training gives good results when the duration of immobility at 7th or more later minutes are considered. On the other hand the second model is appropriate to use only with the data set 1. For both models the test result are worse than the training results which probably depends on the difference between training and test sets. Moreover comparing the first model and the second model with three hidden layer neurons we can claim that the numbers of hidden layer neurons are not affective. There isn't remarkable difference between seasonal groups on test phase. In general, we see that the factors affecting the training and test performance are the initial weights and the input set applied to ANN.

4.3 A Different General ANN Model to Predict the Behavioral Despair Ratio for All Seasons

As in the previous section, in this section we are going to design an ANN based model to predict the behavioral despair (BD) ratio for all seasons. The main difference of this work from the previous one is explained in Section 4.2 is the usage of different test and training sets. The aim is again to obtain a general ANN model which can predict the BD ratio for all seasons and to investigate the impacts of seasonal factors on BD. In this work the data of 37 rats for different season is used. For each season, three rat data are separated in order to be used for the test and the remaining rat data are taken as the training set. In the following, we will present shortly how each training and test set is formed.

4.3.1 Determining the effect of initial weights and neuron number

In data set 1, the inputs are the immobility durations at the 4th, 5th, 6th and 7th minutes which are indicated by imm4, imm5, imm6, imm7, respectively. The output is the BD ratio and the expectation is to be able to predict the BD ratio with the defined inputs via ANN structure. While the test set is composed of data related to rat number 2, 5, and 8 all attended the Porsolt test in May, rat number 11, 14, 19 all attended the Porsolt test in November, rat number 22, 25, 29 all attended the Porsolt test in August and finally, rat number 32, 35, 37 all attended the Porsolt test in February. The training set is composed of data related to other rat data again all related to different seasons.

For data set 2, the difference between the immobility durations of 5th minute and 4th minute are considered as the input of the ANN. For this data set, the test set is composed of data obtained for rat number 2,8,9 taken the Porsolt test in May, rat number 11,14,16 taken the test in November, rat number 24,28, 29 taken the test in August and finally the rat number 32,35,36 taken the test in February. The training set is composed of data related to 25 rats which have not been used in composing the test set.

For data set 3, the difference between the immobility durations of 6th minute and 5th minute are considered as the input of the ANN and the output is BD ratio. For this data set, the test set is composed of data obtained for rat number 3,6,7 taken the Porsolt test in May, rat number 13,16,18 taken the test in November, rat number

21,24, 27 taken the test in August and finally the rat number 30,33,35 taken the test in February. The training set is composed of data related to 25 rats which have not been used in composing the test set.

For data set 4, the difference between the immobility durations of 4th and 5th minutes and 6th minute and 5th minute are considered as the input of the ANN. For this data set, the test set is composed of data obtained for rat number 1,2,6 taken the Porsolt test in May, rat number 23,24,26 taken the test in November, rat number 13,17,18 taken the test in August and finally the rat number 35,36,37 taken the test in February. The training set is composed of data related to rat which have not been used in composing the test set.

The ANN structure used is three layered multilayer perceptron and while its inputs are changing according to different data sets, for hidden layer neurons four different cases are considered as done in the previous chapter that is three, five, seven and ten neurons. For output layer only one neuron is used. During training all four different cases of hidden layer neurons are considered and the learning rule is error back propagation rule.

Firstly, the ANN is trained with the data of 25 rat considering 20 different initial weights for each hidden layer neuron number and for each data set. After completing the training the ANN is tested using the data of 12 rats.

Table 4.3.1: Mean squared error and standard deviation, Training set results for 3 and 5 hidden neurons

Number of Hidden Neurons	Simulation index	Data Set 1	Data Set 2	Data Set 3	Data Set 4
3	Sim1	0.3070 ± 0.4039	1.2907 ± 2.0178	1.2523 ± 1.8744	0.5698 ± 0.5617
	Sim2	1.1171 ± 1.2732	1.2907 ± 2.0178	1.2522 ± 1.8744	0.5522 ± 0.5531
	Sim3	0.3395 ± 0.5214	1.2919 ± 2.0201	1.2522 ± 1.8744	0.5746 ± 0.5607
	Sim4	0.3304 ± 0.4869	1.2907 ± 2.0178	1.2523 ± 1.8744	0.5220 ± 0.5919
	Sim5	0.3423 ± 0.4826	1.2985 ± 2.0356	1.2522 ± 1.8744	0.5621 ± 0.5459
	Sim6	0.3488 ± 0.4785	1.2907 ± 2.0178	1.2522 ± 1.8744	0.5636 ± 0.5495
	Sim7	0.3098 ± 0.4024	1.2907 ± 2.0178	1.2522 ± 1.8744	0.5608 ± 0.5511
	Sim8	0.3891 ± 0.5482	1.2907 ± 2.0178	1.2525 ± 1.8745	0.6433 ± 0.6899
	Sim9	0.3082 ± 0.4130	1.2907 ± 2.0178	1.2522 ± 1.8744	0.5637 ± 0.5495
	Sim10	0.4277 ± 0.6587	1.2927 ± 2.0215	1.2522 ± 1.8744	0.5663 ± 0.5564
	Sim11	0.3482 ± 0.4803	1.2907 ± 2.0178	1.2524 ± 1.8745	0.5544 ± 0.5564
	Sim12	0.3787 ± 0.5137	1.2907 ± 2.0178	1.2523 ± 1.8744	0.5559 ± 0.5558
	Sim13	0.2918 ± 0.3833	1.2907 ± 2.0178	1.2522 ± 1.8744	0.5693 ± 0.5633
	Sim14	0.3434 ± 0.4599	1.2907 ± 2.0178	1.2526 ± 1.8826	0.6554 ± 0.7343
	Sim15	0.3377 ± 0.4602	1.2907 ± 2.0178	1.2522 ± 1.8744	0.5649 ± 0.5574
	Sim16	0.3388 ± 0.4663	1.2907 ± 2.0178	1.2522 ± 1.8744	0.5727 ± 0.5567
	Sim17	0.3115 ± 0.4154	1.2942 ± 2.0245	1.2523 ± 1.8744	0.5673 ± 0.5584
	Sim18	1.1132 ± 1.2940	1.2907 ± 2.0178	1.2522 ± 1.8744	0.5722 ± 0.5581
	Sim19	0.3094 ± 0.4121	1.2920 ± 2.0201	1.2522 ± 1.8744	0.5591 ± 0.5570
	Sim20	0.3460 ± 0.4765	1.2907 ± 2.0178	1.2523 ± 1.8744	0.5653 ± 0.5597
		Data Set 1	Data Set 2	Data Set 3	Data Set 4
5	Sim1	0.3483 ± 0.4838	1.2907 ± 2.0178	1.2524 ± 1.8745	0.5682 ± 0.5535
	Sim2	0.2941 ± 0.4085	1.2907 ± 2.0178	1.2523 ± 1.8744	0.5571 ± 0.5542
	Sim3	0.3508 ± 0.4976	1.2907 ± 2.0178	1.2525 ± 1.8745	0.5510 ± 0.5548
	Sim4	0.3462 ± 0.5216	1.2918 ± 2.0198	1.2457 ± 1.9058	0.5722 ± 0.5578
	Sim5	0.3075 ± 0.4579	1.2907 ± 2.0178	1.2457 ± 1.9032	0.5621 ± 0.5597
	Sim6	0.4430 ± 0.6130	1.2908 ± 2.0179	1.2523 ± 1.8745	0.5580 ± 0.5512
	Sim7	0.3524 ± 0.4702	1.2910 ± 2.0184	1.2523 ± 1.8744	0.5547 ± 0.5506
	Sim8	0.2595 ± 0.3880	1.2907 ± 2.0178	1.2523 ± 1.8744	0.5608 ± 0.5541
	Sim9	1.1155 ± 1.2676	1.2907 ± 2.0178	1.2523 ± 1.8744	0.6557 ± 0.7348
	Sim10	0.3154 ± 0.4349	1.2907 ± 2.0178	1.2457 ± 1.9052	0.5601 ± 0.5594
	Sim11	0.3704 ± 0.5546	1.2907 ± 2.0178	1.2525 ± 1.8745	0.5626 ± 0.5586
	Sim12	0.3514 ± 0.5276	1.2913 ± 2.0188	1.2523 ± 1.8744	0.6538 ± 0.7321
	Sim13	0.2665 ± 0.3762	1.2907 ± 2.0178	1.2456 ± 1.8983	0.5557 ± 0.5551
	Sim14	0.3247 ± 0.4662	1.2907 ± 2.0178	1.2523 ± 1.8744	0.5738 ± 0.5577
	Sim15	0.3365 ± 0.5156	1.2907 ± 2.0178	1.2459 ± 1.9093	0.5615 ± 0.5507
	Sim16	0.2753 ± 0.3878	1.2907 ± 2.0178	1.2526 ± 1.8746	0.5702 ± 0.6054
	Sim17	0.4130 ± 0.5722	1.2907 ± 2.0178	1.2522 ± 1.8744	0.5620 ± 0.5600
	Sim18	0.3479 ± 0.5393	1.2939 ± 2.0239	1.2456 ± 1.8994	0.5577 ± 0.5579
	Sim19	0.3271 ± 0.4716	1.2907 ± 2.0178	1.2523 ± 1.8744	0.5574 ± 0.5431
	Sim20	0.3338 ± 0.4726	1.2907 ± 2.0178	1.2522 ± 1.8744	0.5668 ± 0.5883

As the performance criteria we use the mean error. The training performance increases as the mean error decreases. In Table 4.3.1, mean errors obtained from the simulations using four different data set and 20 different initial weights are listed for two cases, namely, three and five hidden layer neurons. These data can be used in analyzing the effect of the number of hidden layer neurons, the data set and the initial weights on the performance of ANN. In the following the results and comments for each of these cases are given.

When hidden layer neuron number is three the training performance for the data set 1 has different values for different initial weights. The mean error takes values from the range $\{1.1132 \pm 1.2940, 0.2918 \pm 0.3833\}$. Considering other data sets the result for this data set is better. For data set 2 while the error is in general 1.2907 ± 2.0178 , for some initial weight values it ranges in the interval $\{1.2985 \pm 2.0356, 1.2942 \pm 2.0245\}$. On the whole, the performance of ANN does not change much with initial weight values. This is similar for data set 3, for which the mean error ranges in the interval $\{1.2522 \pm 1.8744, 1.2526 \pm 1.8826\}$. For data set 4, only small changes occur in the mean error values during each simulation and these error values range in the interval $\{0.5722 \pm 0.5581 \text{ and } 0.6433 \pm 0.6899\}$. For this hidden layer neuron, the best results are obtained for data set 1 and the worst results are obtained for data set 2.

The results for the hidden layer neuron number five is summarized in Table 4.3.1. For data set 1, the initial weight values are affective on the performance and it changes in the interval $\{0.2941 \pm 0.4085, 1.1155 \pm 1.2676\}$. For data set 2, the mean error value is 1.2907 ± 2.0178 in general, but for some initial weight values it increases up to 1.2918 ± 2.0198 . For data set 3, the mean error varies in the interval $\{1.2456 \pm 1.8994, 1.2523 \pm 1.8744\}$. There is very little variance with different initial weight values. For data set 4, the mean error varies in the interval $\{0.5510 \pm 0.5548, 0.6557 \pm 0.7348\}$. For neuron number 5 the best results are obtained in the simulations carried out with data set 1 and 4.

Table 4.3.2: Mean squared error and standard deviation, Training set results for 7 and 10 hidden neurons

Number of Hidden Neurons	Simulation index	Data Set 1	Data Set 2	Data Set 3	Data Set 4
7	Sim1	0.3099 ± 0.4206	1.2923 ± 2.0207	1.2427 ± 1.9105	0.5661 ± 0.5582
	Sim2	0.3318 ± 0.4676	1.2929 ± 2.0220	1.2425 ± 1.9053	0.5566 ± 0.5581
	Sim3	0.2952 ± 0.4218	1.2908 ± 2.0179	1.2523 ± 1.8744	0.5594 ± 0.5557
	Sim4	0.3148 ± 0.4639	1.2971 ± 2.0314	1.2427 ± 1.9103	0.5640 ± 0.5544
	Sim5	0.2524 ± 0.3832	1.2907 ± 2.0178	1.2527 ± 1.8746	0.5558 ± 0.5578
	Sim6	0.2926 ± 0.4084	1.2942 ± 2.0245	1.2523 ± 1.8744	0.5569 ± 0.5536
	Sim7	0.3124 ± 0.4404	1.2907 ± 2.0178	1.2523 ± 1.8745	0.5586 ± 0.5572
	Sim8	0.2338 ± 0.3375	1.2907 ± 2.0178	1.2524 ± 1.8745	0.5642 ± 0.5592
	Sim9	0.3558 ± 0.5247	1.2907 ± 2.0178	1.2427 ± 1.9103	0.5559 ± 0.5567
	Sim10	0.3323 ± 0.4703	1.2907 ± 2.0178	1.2525 ± 1.8745	0.5645 ± 0.5566
	Sim11	0.3492 ± 0.4920	1.2947 ± 2.0256	1.2431 ± 1.9093	0.5582 ± 0.5541
	Sim12	0.3214 ± 0.4366	1.2954 ± 2.0270	1.2426 ± 1.9097	0.5713 ± 0.5554
	Sim13	0.2920 ± 0.4217	1.2908 ± 2.0179	1.2426 ± 1.9095	0.5638 ± 0.5779
	Sim14	0.2496 ± 0.2907	1.2907 ± 2.0178	1.2522 ± 1.8744	0.5613 ± 0.5975
	Sim15	0.3271 ± 0.4675	1.2907 ± 2.0178	1.2526 ± 1.8746	0.5579 ± 0.5548
	Sim16	0.2506 ± 0.3533	1.2907 ± 2.0178	1.2523 ± 1.8744	0.5569 ± 0.5569
	Sim17	0.3012 ± 0.4326	1.2907 ± 2.0178	1.2522 ± 1.8744	0.5605 ± 0.5596
	Sim18	0.1057 ± 0.1372	1.2938 ± 2.0238	1.2428 ± 1.9084	0.5606 ± 0.5584
	Sim19	0.3583 ± 0.5361	1.2908 ± 2.0179	1.2526 ± 1.8746	0.5601 ± 0.5760
	Sim20	0.3644 ± 0.5360	1.2910 ± 2.0183	1.2426 ± 1.9094	0.5651 ± 0.5528
		Data Set 1	Data Set 2	Data Set 3	Data Set 4
10	Sim1	0.2247 ± 0.2989	1.2922 ± 2.0206	1.2523 ± 1.8744	0.5579 ± 0.5572
	Sim2	0.2986 ± 0.4340	1.2916 ± 2.0194	1.2527 ± 1.8746	0.5590 ± 0.5541
	Sim3	0.2966 ± 0.4444	1.2910 ± 2.0183	1.2419 ± 1.9090	0.5558 ± 0.5552
	Sim4	0.1443 ± 0.1841	1.2954 ± 2.0272	1.2525 ± 1.8745	0.5585 ± 0.5575
	Sim5	0.3072 ± 0.4401	1.2927 ± 2.0215	1.2524 ± 1.8745	0.5538 ± 0.5568
	Sim6	0.2155 ± 0.2547	1.2977 ± 2.0328	1.2523 ± 1.8745	0.5596 ± 0.5518
	Sim7	0.2857 ± 0.4039	1.2920 ± 2.0203	1.2525 ± 1.8745	0.5531 ± 0.5555
	Sim8	0.3214 ± 0.4599	1.2907 ± 2.0178	1.2525 ± 1.8745	0.5594 ± 0.5570
	Sim9	0.3091 ± 0.4265	1.2913 ± 2.0188	1.2403 ± 1.9065	0.5315 ± 0.5792
	Sim10	0.3616 ± 0.5049	1.2907 ± 2.0178	1.2408 ± 1.9088	0.5526 ± 0.5526
	Sim11	0.3091 ± 0.4258	1.2910 ± 2.0183	1.2525 ± 1.8745	0.5589 ± 0.5564
	Sim12	0.3307 ± 0.4776	1.2919 ± 2.0200	1.2405 ± 1.9083	0.5593 ± 0.5579
	Sim13	0.3735 ± 0.5141	1.2907 ± 2.0178	1.2523 ± 1.8744	0.5576 ± 0.5538
	Sim14	0.3349 ± 0.5213	1.2913 ± 2.0188	1.2408 ± 1.9092	0.5572 ± 0.5532
	Sim15	0.3327 ± 0.4871	1.2976 ± 2.0327	1.2404 ± 1.9074	0.5557 ± 0.5605
	Sim16	0.3296 ± 0.5144	1.2907 ± 2.0178	1.2404 ± 1.9081	0.5569 ± 0.5537
	Sim17	0.2857 ± 0.4067	1.2907 ± 2.0178	1.2527 ± 1.8746	0.5607 ± 0.5560
	Sim18	0.3707 ± 0.5579	1.2926 ± 2.0214	1.2405 ± 1.9084	0.5595 ± 0.5573
	Sim19	0.3024 ± 0.4460	1.2909 ± 2.0182	1.2407 ± 1.9078	0.5664 ± 0.5514
	Sim20	0.3424 ± 0.4932	1.2938 ± 2.0237	1.2524 ± 1.8745	0.5537 ± 0.5597

The same approach is carried out for hidden layer neuron number 7 and 10 and the results are summarized in Table 4.3.2. Again for each hidden layer neuron number with four different data set, simulations are carried out 20 times with different initial weight values.

Following the results shown in Table 4.3.2, a summary can be given for neuron number 7. For data set 1, the performance criteria of the ANN structure which is taken to be mean error changes in the interval $\{0.1057 \pm 0.1372, 0.3644 \pm 0.5360\}$. The initial weight values somewhat affect performance. For data set 2, the mean error value does not change much with the initial weight values. It generally varies in the interval $\{1.2954 \pm 2.0270, 1.2907 \pm 2.0178\}$ for data set 3, the mean error varies in the interval $\{1.2426 \pm 1.9094, 1.2527 \pm 1.8746\}$ and the variation is very small. For data set 4, the mean error again varies very little in the interval $\{0.5640 \pm 0.5544, 0.5713 \pm 0.5554\}$. While the results are good for data set 1 and 4, they are poor for data set 2 and 3.

When hidden layer neuron number is 10, the results obtained for 20 different initial weight values can be summarized as following: For data set 1, the mean error varies in the interval $\{0.1443 \pm 0.1841, 0.3735 \pm 0.5141\}$, for data set 2, the mean error varies very little in the interval $\{1.2907 \pm 2.0178, 1.2938 \pm 2.0237\}$. For data set 3 and 4 the mean errors vary in the intervals $\{1.2403 \pm 1.9065, 1.2525 \pm 1.8745\}$ and $\{0.5315 \pm 0.5792, 0.5664 \pm 0.5514\}$, respectively.

When we consider the change in the mean error within the same data set, it seems that increasing the number of hidden layer neurons is not much effective on the performance of the ANN structure. But the choice of data set and the initial weight values have effect on the performance.

When the results are evaluated in the general, it seems that the increase in the neuron number is not much effective on the performance. The data set structure is much more effective than the other parameters, even though the initial values of weights do have some effect on the performance.

Once the training phase is completed, in order to decide the effectiveness of the model obtained by ANN, test results have to be evaluated. The main aim of the test phase is to check the performance of the ANN structure obtained, especially the generalization ability. These are summarized in Table 4.3.3 and 4.3.4.

Table 4.3.3: Mean squared error and standard deviation, Test set results for 3 and 5 hidden neurons

Number of Hidden Neurons	Simulation index	Data Set 1	Data Set 2	Data Set 3	Data Set 4
3	Sim1	0.1469 ± 0.2577	0.0950 ± 0.1880	0.1192 ± 0.2069	0.1903 ± 0.3219
	Sim2	0.0692 ± 0.1039	0.0950 ± 0.1880	0.1192 ± 0.2069	0.1999 ± 0.3526
	Sim3	0.1453 ± 0.3053	0.0949 ± 0.1878	0.1192 ± 0.2069	0.1896 ± 0.3218
	Sim4	0.1460 ± 0.3033	0.0950 ± 0.1880	0.1192 ± 0.2068	0.2310 ± 0.4515
	Sim5	0.1462 ± 0.2930	0.0942 ± 0.1847	0.1192 ± 0.2069	0.1857 ± 0.3132
	Sim6	0.1473 ± 0.2704	0.0950 ± 0.1880	0.1192 ± 0.2069	0.1918 ± 0.3287
	Sim7	0.1468 ± 0.2565	0.0950 ± 0.1880	0.1192 ± 0.2069	0.1919 ± 0.3297
	Sim8	0.1438 ± 0.2711	0.0950 ± 0.1880	0.1193 ± 0.2066	0.1859 ± 0.3018
	Sim9	0.1466 ± 0.2599	0.0950 ± 0.1880	0.1192 ± 0.2069	0.1879 ± 0.3189
	Sim10	0.1347 ± 0.2760	0.0948 ± 0.1877	0.1192 ± 0.2069	0.1852 ± 0.3122
	Sim11	0.1478 ± 0.2713	0.0950 ± 0.1880	0.1192 ± 0.2068	0.2015 ± 0.3566
	Sim12	0.1436 ± 0.2615	0.0950 ± 0.1880	0.1192 ± 0.2069	0.2012 ± 0.3551
	Sim13	0.1451 ± 0.2513	0.0950 ± 0.1880	0.1192 ± 0.2069	0.1929 ± 0.3306
	Sim14	0.1446 ± 0.2586	0.0950 ± 0.1880	0.1185 ± 0.2201	0.1905 ± 0.3116
	Sim15	0.1475 ± 0.2681	0.0950 ± 0.1880	0.1192 ± 0.2069	0.1906 ± 0.3252
	Sim16	0.1480 ± 0.2712	0.0950 ± 0.1880	0.1192 ± 0.2069	0.1923 ± 0.3288
	Sim17	0.1469 ± 0.2600	0.0947 ± 0.1872	0.1192 ± 0.2068	0.1877 ± 0.3182
	Sim18	0.0746 ± 0.1171	0.0950 ± 0.1880	0.1192 ± 0.2069	0.1914 ± 0.3266
	Sim19	0.1467 ± 0.2591	0.0949 ± 0.1878	0.1192 ± 0.2069	0.2022 ± 0.3573
	Sim20	0.1478 ± 0.2718	0.0950 ± 0.1880	0.1192 ± 0.2069	0.1959 ± 0.3362
		Data Set 1	Data Set 2	Data Set 3	Data Set 4
5	Sim1	0.1481 ± 0.2741	0.1665 ± 0.1973	0.1192 ± 0.2067	0.1938 ± 0.3334
	Sim2	0.1450 ± 0.2539	0.1494 ± 0.1792	0.1192 ± 0.2068	0.1934 ± 0.3340
	Sim3	0.1461 ± 0.2875	0.0996 ± 0.1548	0.1193 ± 0.2067	0.2012 ± 0.3574
	Sim4	0.1451 ± 0.3046	0.1556 ± 0.2716	0.1168 ± 0.2237	0.1919 ± 0.3280
	Sim5	0.1489 ± 0.2863	0.1232 ± 0.1436	0.1169 ± 0.2233	0.2012 ± 0.3548
	Sim6	0.1400 ± 0.2308	0.2538 ± 0.3138	0.1192 ± 0.2069	0.1926 ± 0.3317
	Sim7	0.1733 ± 0.2656	0.3110 ± 0.3873	0.1192 ± 0.2068	0.1969 ± 0.3439
	Sim8	0.1419 ± 0.2618	0.1539 ± 0.1857	0.1192 ± 0.2069	0.1986 ± 0.3472
	Sim9	0.0697 ± 0.1034	0.4057 ± 0.3438	0.1192 ± 0.2068	0.1906 ± 0.3118
	Sim10	0.1446 ± 0.2548	0.1559 ± 0.1962	0.1168 ± 0.2236	0.2021 ± 0.3572
	Sim11	0.1425 ± 0.2961	0.1343 ± 0.2447	0.1193 ± 0.2067	0.2013 ± 0.3538
	Sim12	0.1445 ± 0.3032	0.1504 ± 0.2652	0.1192 ± 0.2069	0.1908 ± 0.3124
	Sim13	0.1434 ± 0.2515	0.1567 ± 0.1865	0.1172 ± 0.2226	0.1994 ± 0.3507
	Sim14	0.1458 ± 0.2654	0.1517 ± 0.1885	0.1192 ± 0.2069	0.1914 ± 0.3266
	Sim15	0.1463 ± 0.3136	0.2058 ± 0.3197	0.1166 ± 0.2242	0.1896 ± 0.3240
	Sim16	0.1453 ± 0.2551	0.1473 ± 0.1783	0.1193 ± 0.2065	0.1920 ± 0.3263
	Sim17	0.1335 ± 0.2602	0.1407 ± 0.1710	0.1192 ± 0.2069	0.2008 ± 0.3538
	Sim18	0.1436 ± 0.2988	0.1310 ± 0.2255	0.1171 ± 0.2228	0.2013 ± 0.3547
	Sim19	0.1464 ± 0.2677	0.1282 ± 0.1579	0.1192 ± 0.2068	0.1976 ± 0.3445
	Sim20	0.1487 ± 0.2921	0.0956 ± 0.1395	0.1192 ± 0.2069	0.1929 ± 0.3299

In Table 4.3.3, test results for hidden layer neuron number 3 and 5 are given. When the hidden layer neuron number is three, for the data set 1, we obtained good results for two simulations compared to other 18 simulations. As we had mentioned previously, even though the results are good for the training set, they are not that good for the test set. We had explained that this may be result of overtraining, which causes poor generalization ability and decreases the test set performance. The two simulations with good results are simulation 2 and 18, for which the mean errors are 0.0692 ± 0.1039 and 0.0746 ± 0.1171 , respectively. In the other simulations for this case the mean error varies in the interval $\{0.1335 \pm 0.2602, 0.1480 \pm 0.2712\}$. For data set 2, there is not much difference in the mean error values with the changing initial weight values. While the error mean value is 0.0950 ± 0.1880 in general and its minimum is 0.0942 ± 0.1847 . Similarly, for data set 3, the results do not change much with the initial weight values and in most simulations its value is 0.1192 ± 0.2069 . For data set 4, in each simulation the mean error changes slightly in the interval $\{0.2022 \pm 0.3573, 0.1857 \pm 0.3132\}$. On the overall the best results are obtained for data set 2 and the worst results are obtained for data set 4.

When the hidden layer neuron is five, for data set 1, the mean error value has its minimum value at the 9th simulation which is 0.0697 ± 0.1034 . For this simulation, mean error obtained for the training phase was the maximum value. This is again due to the negative effect of overtraining. Instead of choosing the iteration number as a stopping criterion, another method was used there would not have been such differences in the mean error values between the training set and test set. For this data set, mean error values varies in the interval $\{0.1335 \pm 0.2602, 0.1733 \pm 0.2656\}$. For data set 2, mean error varies with initial weight values in the interval $\{0.0956 \pm 0.1395, 0.4057 \pm 0.3438\}$ and for data set 3 it does not change much with the initial weigh values. The interval is $\{0.1166 \pm 0.2242, 0.1193 \pm 0.2067\}$. Similarly, for data set 4 mean error value differs very little in the interval $\{0.1908 \pm 0.3124, 0.2021 \pm 0.3572\}$.

Table 4.3.4: Mean squared error and standard deviation, Test set results for 7 and 10 hidden neurons

Number of Hidden Neurons	Simulation index	Data Set 1	Data Set 2	Data Set 3	Data Set 4
7	Sim1	0.1472 ± 0.3111	0.0949 ± 0.1878	0.1163 ± 0.2249	0.2012 ± 0.3538
	Sim2	0.1477 ± 0.2669	0.0948 ± 0.1876	0.1166 ± 0.2241	0.2019 ± 0.3578
	Sim3	0.1423 ± 0.2551	0.0950 ± 0.1880	0.1192 ± 0.2069	0.2022 ± 0.3564
	Sim4	0.1493 ± 0.2858	0.0944 ± 0.1858	0.1164 ± 0.2248	0.1921 ± 0.3302
	Sim5	0.1447 ± 0.3561	0.0950 ± 0.1880	0.1193 ± 0.2065	0.2013 ± 0.3557
	Sim6	0.1431 ± 0.2506	0.0948 ± 0.1872	0.1192 ± 0.2069	0.1981 ± 0.3460
	Sim7	0.1517 ± 0.2875	0.0950 ± 0.1880	0.1192 ± 0.2068	0.1945 ± 0.3372
	Sim8	0.1550 ± 0.3538	0.0950 ± 0.1880	0.1192 ± 0.2067	0.1990 ± 0.3481
	Sim9	0.1437 ± 0.2972	0.0950 ± 0.1880	0.1164 ± 0.2249	0.2022 ± 0.3580
	Sim10	0.1498 ± 0.2897	0.0950 ± 0.1880	0.1193 ± 0.2066	0.1989 ± 0.3476
	Sim11	0.1436 ± 0.2551	0.0947 ± 0.1871	0.1164 ± 0.2246	0.2027 ± 0.3583
	Sim12	0.1427 ± 0.2534	0.0947 ± 0.1868	0.1164 ± 0.2248	0.1941 ± 0.3336
	Sim13	0.1436 ± 0.2614	0.0950 ± 0.1880	0.1164 ± 0.2248	0.1876 ± 0.3168
	Sim14	0.2611 ± 0.3588	0.0950 ± 0.1880	0.1192 ± 0.2069	0.2014 ± 0.3516
	Sim15	0.1437 ± 0.2577	0.0950 ± 0.1880	0.1193 ± 0.2065	0.1958 ± 0.3402
	Sim16	0.1348 ± 0.2338	0.0950 ± 0.1880	0.1192 ± 0.2068	0.2017 ± 0.3553
	Sim17	0.1448 ± 0.2632	0.0950 ± 0.1880	0.1192 ± 0.2069	0.1988 ± 0.3471
	Sim18	0.2015 ± 0.3767	0.0948 ± 0.1874	0.1165 ± 0.2245	0.1998 ± 0.3503
	Sim19	0.1457 ± 0.2954	0.0950 ± 0.1880	0.1193 ± 0.2065	0.2003 ± 0.3519
	Sim20	0.1427 ± 0.2970	0.0949 ± 0.1880	0.1164 ± 0.2247	0.1993 ± 0.3481
		Data Set 1	Data Set 2	Data Set 3	Data Set 4
10	Sim1	0.2132 ± 0.2739	0.0949 ± 0.1878	0.1192 ± 0.2068	0.1974 ± 0.3459
	Sim2	0.1461 ± 0.2686	0.0949 ± 0.1879	0.1193 ± 0.2064	0.1934 ± 0.3340
	Sim3	0.1432 ± 0.2898	0.0949 ± 0.1880	0.1164 ± 0.2248	0.1993 ± 0.3501
	Sim4	0.1768 ± 0.3750	0.0947 ± 0.1868	0.1193 ± 0.2066	0.1998 ± 0.3529
	Sim5	0.1471 ± 0.2723	0.0948 ± 0.1877	0.1192 ± 0.2068	0.2032 ± 0.3626
	Sim6	0.2085 ± 0.2759	0.0944 ± 0.1855	0.1192 ± 0.2069	0.1989 ± 0.3497
	Sim7	0.1412 ± 0.2453	0.0949 ± 0.1878	0.1193 ± 0.2067	0.2021 ± 0.3577
	Sim8	0.1484 ± 0.2750	0.0950 ± 0.1880	0.1193 ± 0.2066	0.1986 ± 0.3482
	Sim9	0.1401 ± 0.2383	0.0949 ± 0.1880	0.1165 ± 0.2247	0.2267 ± 0.4347
	Sim10	0.1385 ± 0.2800	0.0950 ± 0.1880	0.1164 ± 0.2250	0.1986 ± 0.3489
	Sim11	0.1404 ± 0.2399	0.0949 ± 0.1880	0.1193 ± 0.2067	0.2021 ± 0.3577
	Sim12	0.1305 ± 0.2248	0.0949 ± 0.1879	0.1164 ± 0.2250	0.1979 ± 0.3450
	Sim13	0.1172 ± 0.2142	0.0950 ± 0.1880	0.1192 ± 0.2068	0.1964 ± 0.3424
	Sim14	0.1459 ± 0.3105	0.0949 ± 0.1880	0.1163 ± 0.2251	0.1940 ± 0.3361
	Sim15	0.1502 ± 0.2970	0.0944 ± 0.1855	0.1164 ± 0.2248	0.2015 ± 0.3578
	Sim16	0.1443 ± 0.3200	0.0950 ± 0.1880	0.1164 ± 0.2250	0.1979 ± 0.3469
	Sim17	0.1446 ± 0.2591	0.0950 ± 0.1880	0.1193 ± 0.2064	0.2034 ± 0.3603
	Sim18	0.1411 ± 0.2913	0.0949 ± 0.1877	0.1164 ± 0.2250	0.1947 ± 0.3373
	Sim19	0.1379 ± 0.2705	0.0950 ± 0.1880	0.1164 ± 0.2248	0.1954 ± 0.3385
	Sim20	0.1464 ± 0.2886	0.0948 ± 0.1874	0.1192 ± 0.2067	0.2022 ± 0.3599

In Table 4.3.4, the results for the test phase is given when the hidden layer neuron number is seven and ten. While obtaining these results the weights obtained in the training phase for which the results are given in Table 2 are used and the data set is composed of data related to nine rats attained to the Porsolt test. How effective the training phase was will in a way determined from the results obtained for the test set.

For hidden layer neuron number seven, when the data set is the first one, the initial weight values are not much effective on the results. The best result is obtained at the 12th simulation and the worst result is obtained at the 18th simulation. These are 0.1427 ± 0.2534 and 0.2015 ± 0.3767 , respectively. For data set 2 and data set 3, again the initial weight values do not effect the mean error values much and it varies in the interval $\{0.0944 \pm 0.1858, 0.0950 \pm 0.1880\}$ and $\{0.1163 \pm 0.2249$ ve $0.1193 \pm 0.2065 \}$, respectively. Similarly, for data set 4, the mean error value varies in the interval $\{0.1876 \pm 0.3168, 0.2022 \pm 0.3564\}$

The last step of the evaluation of the test phase is the case with 10 hidden layer neuron numbers. For data set 1, the mean error value varies in the interval $\{0.1305 \pm 0.2248, 0.2132 \pm 0.2739\}$ and the initial weight values are somewhat effective on the results. For data set 2, 3 and 4, the initial weight values are not much effective on the results and it takes values in the interval $\{0.0944 \pm 0.1855, 0.0950 \pm 0.1880\}$, $\{0.1163 \pm 0.2251, 0.1193 \pm 0.2066\}$ and $\{0.1934 \pm 0.3340, 0.2034 \pm 0.3603\}$, respectively.

On the overall evaluation of the test results, the number of hidden layer neuron is not much effective on the mean error values. The results could have been improved for data set 1 and 4, if the stopping criteria for the training phase have been chosen differently. It could have been an upper bound on the error value rather than iteration number. This can be seen from the two cases, where the error value for the training phase was high but the test results were better than the other simulations. To be specific, these cases are the simulation2 and 18 for data set 1 with hidden layer number three, and simulations nine for hidden layer neuron number 5. While for data set 2, the results are better, they are not that much depending for the data set 3.

4.3.2 Results

The results of training and test simulations for four different data sets using different numbers of hidden layer neurons have been already given above. In Figure 4.3.1 and 4.3.2, respectively, results of training and test phase of the simulation having the best test performance among those given in Table 4.3.3 and 4.3.4 are illustrated.

The best test performance for the data set 1(imm4-7) is obtained when the hidden layer neuron number is three in the second simulation, whereas the training performance of this simulation is not extreme among the others.



Figure 4.3.1: Outputs of Training Phase

Considering the following figure, we can conclude that the test result of this simulation is worse than the training result.

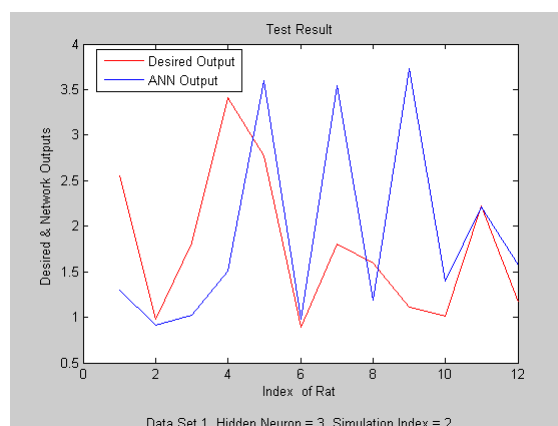


Figure 4.3.2: Outputs of Test Phase

The best test performance for the data set 2(imm4-5) is obtained in the simulation 5 when the hidden layer neuron number is three. The training performance for this case is not so satisfactory. Consequently, BD estimation seems does not possible using this data set.



Figure 4.3.3: Outputs of Training Phase

The performance of the test phase is also poor as a consequence of the poor training performance.



Figure 4.3.4: Outputs of Test Phase

The best test performance for the data set 3 (imm5-6) is obtained in the first simulation when the hidden layer neuron number is seven. In Figure 4.3.5 and 4.3.6, respectively, results of training and test phase of the simulation having the best test performance among those given in Table 4.3 and 4.3.4 are illustrated. The training performance for this case is not so satisfactory. Consequently, BD estimation seems not possible using this data set.

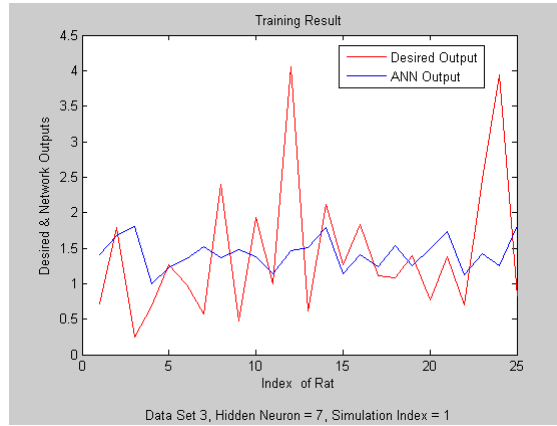


Figure 4.3.5: Outputs of Training Phase

The performance of the test phase is also poor as a consequence of the poor training performance.

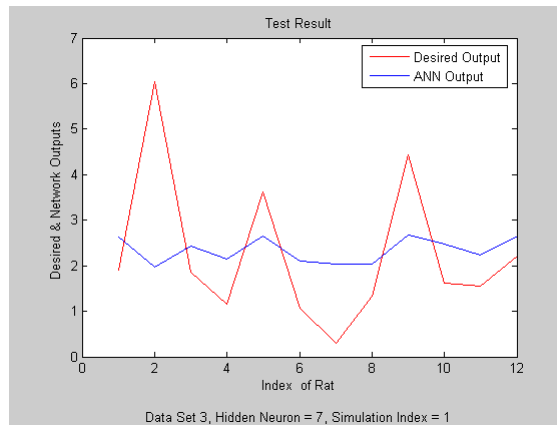


Figure 4.3.6: Outputs of Test Phase

The best test performance for the data set 4(imm5-imm4 and imm6—imm5) is obtained in simulation 10 when the hidden layer neuron number is three. In Figure 4.3.7 and 4.3.8, respectively, results of training and test phase of the simulation are illustrated. The training phase for this data set is more satisfactory than those for the data set 2 and 3. Nevertheless, the training performance is not perfect. Consequently, BD estimation seems not easy using this data set.



Figure 4.3.7: Outputs of Training Phase



Figure 4.3.8: Outputs of Test Phase

Considering the best cases for all data sets illustrated in the above figures, the model for BD estimation can only be achieved using the data set 1 to a certain extent. It is not possible to perform BD estimation using the other data sets. Moreover, from the results we observed that the number of hidden layer neurons is not much affective over the performance of ANN. The basic factors affecting the performance are the initial weights and data sets.

4.4 A Model for Prediction of Immobility Behavior During PST2 (Second Day of Porsolt Test)

The Porsolt Test used to induce behavioral despair in rats, consist of two sessions. First session is PST1, other data set is obtained from the experiment which takes place following day, and its label is PST2. In PST1 sessions, the rat is placed in a container filled with water and forced to swim 15 minutes. Next day, the rat is placed back for 5 minutes. It shows immobility behavior mostly. Immobility is characteristic behavior and points to depressive mood in rat.

ANN consists of three layers which are input, hidden that has 20 neurons and output layer which include only one neuron. Learning rate is 0.5 and iteration number is 5000. The ANN intends to predict immobility of PST2 by various immobility of PST1 in different time intervals lasting in minutes.

There are 37 rats that consist of four groups and are tested in different seasons. In training phase, 25 rats are used, rest of them are used for test phase. Test and training pattern sets are composed of four group rats for which experiments are done in different seasons.

4.4.1 A model for prediction of PST2 considering only third, fourth and fifth minutes immobility

Immobility is third, fourth and fifth minutes in PST1 considered as input of ANN. Immobility of PST2 session are desired output of the ANN. inputs of the MLP and outputs are given on Table A.7 and A.15 respectively in appendix.

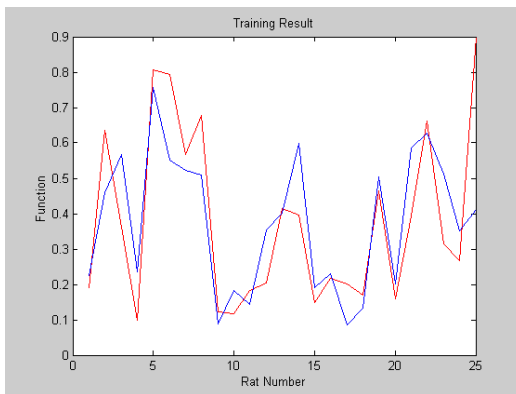


Figure 4.4.1: Desired and ANN Output
— : ANN Output
— : Desired Output

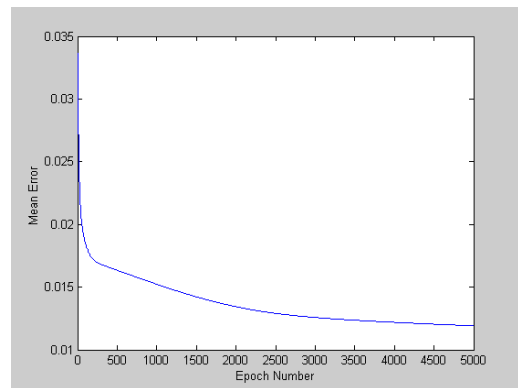


Figure 4.4.2: Mean Error of Training Phase

End of training phase, mean square error is 0.0119, this parameter is critical for performance.

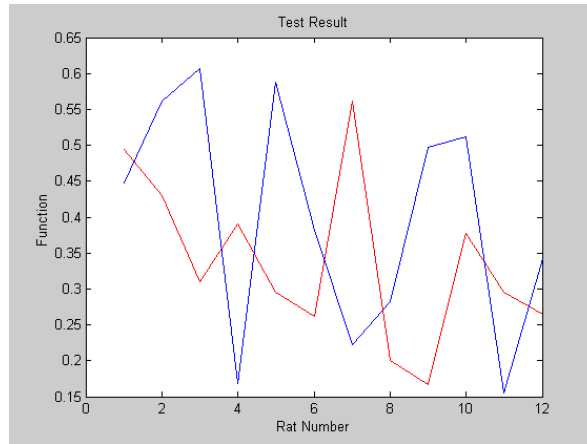


Figure 4.4.3: Desired and ANN Output of Test Phase

— : ANN Output
— : Desired Output

Test results aren't as good as training results. This may be caused due to many reasons as seasonal differences, characteristic differences between rats, etc. Because of all these reasons a general ANN model can't be designed to predict PST2.

4.4.2 A model for prediction of PST2 considering each of the first five minutes immobility

First five minutes of PST1 are given as ANN inputs. Output of the ANN isn't changed in this experiment. Training results are better than first experiment. This improvement is due to more inputs. Inputs and outputs of training phase are given Table A.7 and A.15 in appendix.

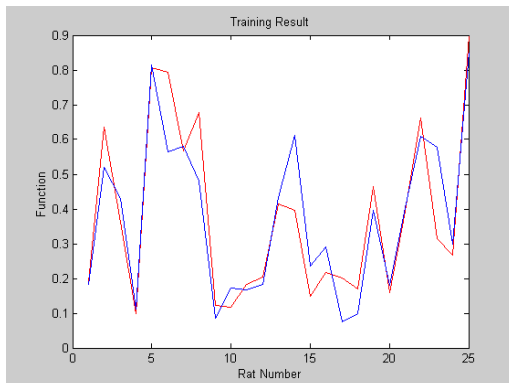


Figure 4.4.4: Desired and ANN Output

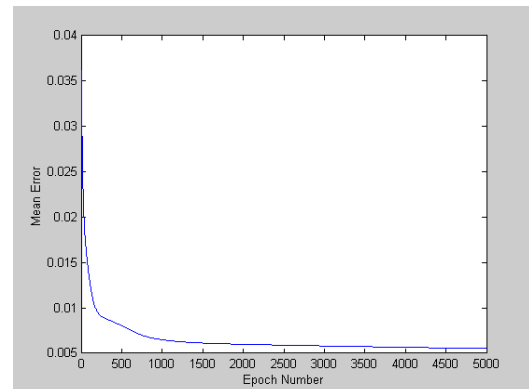


Figure 4.4.5: Mean Error of Training Phase

Last square mean error is 0.00555. Performance gets better than previous simulation by adding extra inputs.

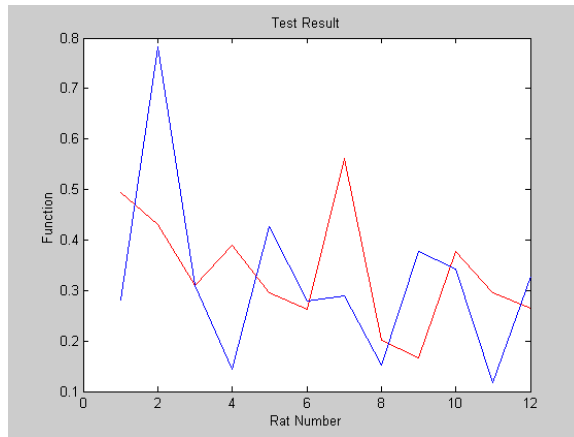


Figure 4.4.6: Desired and ANN Output Of Test Phase

— : ANN Output
— : Desired Output

Again, test performance is bad in comparison to training performance. Test result is given on Table A.18 in appendix.

4.4.3 A model for prediction of PST2 considering each of the first ten minutes immobility

Immobility of ten minutes are applied the ANN as inputs. Desired outputs are same as previous experiments.

As a seen in below graphs, performance is excellent. Last mean error that is critical parameter of performance is nearly zero.

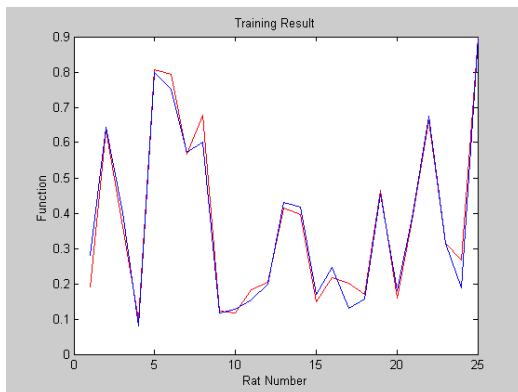


Figure 4.4.7: Desired and ANN Output

— : ANN Output
— : Desired Output

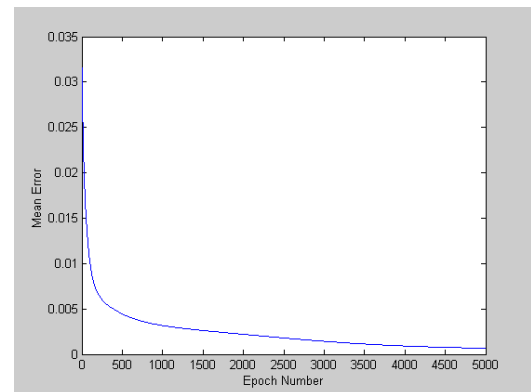


Figure 4.4.8: Mean Error of Training Phase

Results of training phase are also given on Table A.19 in appendix.

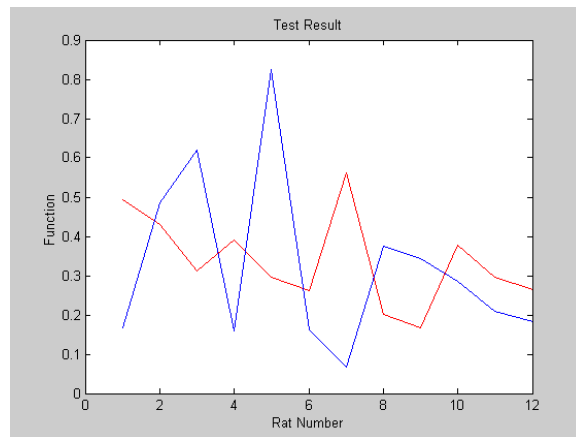


Figure 4.4.9: Desired and ANN Output of Test Phase

— : ANN Output
— : Desired Output

5. DISCUSSION AND CONCLUSION

The aim of this thesis was to obtain ANN based models in order to predict the behavioral despair. Behavioral despair is an approach used in pharmacology to develop antidepressants and to understand the mechanisms creating depression. In this thesis, two different aspects related to behavioral despair were considered, while one was to understand what type of animal behaviors give clues about the behavioral despair, the other was to investigate the effect of seasonal changes on the behavioral despair. A criteria for behavioral despair is the BD ratio and in this thesis this ratio is accepted as a critical parameter for depression risk or it is considered as a depressive mood indicator [3].

In order to fulfill the aim, data from the research team carrying out their studies under the supervision of Reşit Canbeyli at Psychology Department in Boğaziçi University were obtained. The forced swimming test, i.e., Porsolt test results are used as data set, where the Porsolt test is carried out with 17 rats and 37 rats at different seasons and four different groups are considered. In this thesis, four different modeling problems are considered. In the first modeling approach, immobility and wet-dog-shake behavior, in the second and third modeling approach only immobility behavior were considered and ANN structures for predicting the BD ratio in different seasons are obtained. In the last modeling approach, the aim was different and this time the ANN structure was developed to predict the duration of immobility behavior in second day, considering the immobility behavior at different times in the first day.

These modeling approaches correspond to the function approximation problem by ANN structures and since multilayer perceptrons are known to be best candidate for this type of problem, it is preferred as the ANN structure in this thesis. The modeling problem dealt with in the section 4.1 is the same as the problem considered in the thesis of İ.Oruç [3], where she obtained a model using ADALINE structure. When the results of this thesis are compared with the results of [3], it can be seen clearly that using MLP structure is much more advantageous. During all modeling approaches, three layered MLP with one hidden layer was used. The simulations of

the models were carried out in MATLAB 7.0, without using any toolbox but developing a parametric and adaptable code.

In the first modeling approach considered in section 4.1, the aim was to predict the BD ratio, considering wet-dog-shake and immobility as the inputs, thus as the parameters effecting the BD ratio. With the ANN structures obtained in this thesis, BD ratio can be predicted, given the wet-head-shake and the immobility durations. So that, wet-head-shake and immobility behaviors have affects on behavioral despair. In order to have better generalization ability which would give better test results, the stopping criteria could have been chosen as setting an upper bound on the error to prevent over training.

The second modeling approach given in section 4.2 deals with obtaining an ANN structure capable of predicting BD ratio for any season considering only immobility behavior. Two different works are carried out, while in section 4.2.1 the data of 17 rats used in the first modeling approach is used in the training set, and the data obtained at different seasons from 37 rats was used in the test set. Seven different data set were used each composed of immobility durations at different minutes considered during the Porsolt test. In this approach, ascending input number improved the training phase but did not much effect the test phase. In section 4.2.2 with four different data set and four different hidden layer neuron number (3,5,7,10) statistical work was carried out in order to investigate the effect of initial weight values on the results. Thus, each simulation was repeated with 20 randomly determined initial weight values. In this trial, data of 16 rats are used, ignoring one rat data with extreme value as the training set, and like in section 4.2.1 data of 37 rats are used as the test set. The only meaningful result obtained in this trial was the one obtained with data set 1. While, the ascending hidden layer number did not much affect the results, the initial weight values and different combinations of the ANN inputs affected the outputs. In both trials, while the results for the training set was successful, for the test set they were not that much successful. In this section, a general model could not be realized for all seasons in test phase, probably, because of seasonal factors, differences among rat groups and over training of the ANN.

In section 4.3, using only the immobility behavior and the data of 37 rats obtained in different seasons were used to obtain an ANN model to predict the BD ratio. For each season, three rat data are used to set up test set and the others are used to form the training set. Four different data set and four different hidden layer neuron number (3,5,7,10) was considered and statistical work was carried out in order to

investigate the effect of initial weight values on the results. As like in section 4.2.2, successful results are obtained only for data set 1. The results obtained for data1 set 1 in this section were better than the results in section 4.2.2. So, composing the training set and test set using the same rat group improves the performance. As like in section 4.2, there was not much difference between the different season groups during the test phase. It is clear that immobility behavior is critical in predicting the behavioral despair following the results obtained in section 4.2 and 4.3. On the other hand, failure in obtaining a general model for any season may have different reasons as seasonal effects, the difference among the rats and so on.

In section 4.4, the modeling problem was totally different than the previous ones; here the Porsolt tests first day immobility behavior was used to predict the second day's immobility behavior for any season. Data of 37 rats obtained at four different seasons are used. For each season, three rat data were used to set up test set and the others were used to form the training set. Three different training set were formed each containing the immobility duration at different times. While results are good for the training set, the test phase is not that much successful.

Cognitive science has important role in understanding different aspects of behavior and as it is an interdisciplinary field, scientists in different fields use many different methods and viewpoints while working. To understand nature of behavioral despair or behavioral despair as psychological behaviors, cognitive science is powerful approach and methods. We suggested that other behaviors such as wet-dog-shake, struggle, etc., environmental factors should be considered and analyzed. It is true that there is still much to be explored about behavioral despair. Behavioral despair and its consequences are not as simple as people once thought, and many details need to be worked out. Many important questions remain to be answered, and it is for sure that some important questions are not even asked yet.

It is often difficult to predict at the outset where research will be lead. This may be an advantage if negative consequences are appreciated and more work is carried on considering the inability to control important environmental events. It must be kept in mind that interpreting a similar inability produced a method of studying the behavioral despair phenomenon.

In this thesis, analysis of behavioral data obtained from Porsolt test is carried out using ANN structure as a function approximator and four different arguments are evaluated. Another approach could have been investigating the neural substrates, their relations and the mechanisms behind behavioral despair and realize its model.

REFERENCES

- [1] **Willner, P.**, 1990. Animal Models of Depression: An Overview, 425-445.
- [2] **McKinney W.T, Bunney W.E.**, Aug 1969. Animal Model of Depression, *Arch Gen Psychiat.*
- [3] **Oruç İ.**, 1999, *Prediction of Learned Despaired in Rats Using Artificial Neural Networks.*
- [4] **Haykin S.**, 1998, *Neural Networks: A Comprehensive Foundation.*
- [5] **P.S. Churchland**, 1988 , “*Neurophilosophy-toward a unified science of the mind-brain*”, A Bradford Book, The MIT Press.
- [6] **N. Kasabov**, 2003. “Evolving Connectionist Systems”, *Springer.*
- [7] **D.E. Rumelhart; G.E. Hinton, R.J. Williams**, 1986, “Learning internal representations by error propagation”, *Parallel and Distributed Processing: Explorations in the Microstructures of Cognition*, D.E. Rumelhart and J.L. McClelland (Eds.), Cambridge, MA: MIT Press, pp.318-362.
- [8] **S. Grossberg**, 1980, “How does a brain build a cognitive code?”, *Psychological Review* , pp. 1-51.
- [9] **J.L. Armony, D. Servan-Schreiber, J.D. Cohen, J.E. LeDoux**, 1995, “An anatomically constrained neural network model of fear conditioning”, *Behavioral Neuroscience*, pp. 246-257.
- [10] **J.L. Armony, D. Servan-Schreiber, J.D. Cohen, J.E. LeDoux**, 1997, “Computational modelling of emotion: explorations through the anatomy and physiology of fear conditioning”, *Trends in Cognitive Sciences*, no.1, pp.28-34.
- [11] **J.D. Cohen, D. Servan-Schreiber**, 1992 , “Context, cortex and dopamine: a connectionist approach to behavior and biology in schizophrenia”, *Psychological Review* , pp. 45-77.

- [12] **J.D. Cohen, K. Dunbar, J.L. Mc Clelland**, 1990, "On the control of automatic processes: a parallel distributed processing account of the Stroop effect", *Psychological Review*, pp. 332-361.
- [13] http://en.wikipedia.org/wiki/Learned_helplessness
- [14] http://findarticles.com/p/articles/mi_qa3673/is_200407/ai_n9422024
- [15] http://findarticles.com/p/articles/mi_m2405/is_n2_v125/ai_21240925
- [16] <http://www.acnp.org/G4/GN401000076/CH.html>
- [17] <http://flyfishingdevon.co.uk/salmon/year2/psy221depression/psy221depression.htm>

APPENDIX A.

Table A.1: Outputs in Training Phase
for Data Set

Desired Output	ANN Output
1.3500	1.3738
2.3000	2.5862
4.2500	3.6185
0.7600	0.3238
1.9300	1.9762
2.3500	2.3113
3.1000	3.0780
3.1200	3.9866
0.2000	0.4855
0.8800	0.9520
1.9800	1.6916

Table A.2: Outputs in Test Phase
for Data Set1

Desired Output	ANN Output
2.1200	1.1294
2.2500	1.6334
1.1800	1.1692
1.6600	2.1205
3.8300	2.0901

Table A.3: Outputs in Training Phase
for Data Set 2

Desired Output	ANN Output
1.3500	1.2791
2.3000	2.3054
4.2500	4.4196
1.9300	1.9340
2.3500	2.3171
2.1200	2.1220
2.2500	2.2600
1.1800	1.2140
1.6600	1.7081
3.8300	3.7971
0.8800	0.8892

Table A.4: Outputs in Test Phase
for Data Set2

Desired Output	ANN Output
0.7600	1.5960
3.1000	1.5358
3.1200	4.0799
0.2000	1.5303
1.9800	2.2967

Table A.5: Outputs in Training Phase
for Data Set3

Desired Output	ANN Output
1.3500	1.3772
2.3000	2.2982
4.2500	3.6735
1.9300	1.9582
2.3500	2.3545
2.2500	2.2577
3.1000	3.0950
1.1800	1.1196
1.6600	1.6200
3.1200	3.9757
0.8800	0.9189

Table A.6: Outputs in Test Phase
for Data Set3

Desired Output	ANN Output
0.7600	2.1038
2.1200	2.0247
0.2000	2.1028
3.8300	2.0382
1.9800	2.0152

Table A.7: Normalized Inputs of the ANN in Training Phase

Imm1_1	Imm1_2	Imm1_3	Imm1_4	Imm1_5	Imm1_6	Imm1_7	Imm1_8	Imm1_9	Imm1_10
0.2492	0.5068	0.4390	0.5068	0.4390	0.4254	0.2627	0.3169	0.6831	0.5068
0.3034	0.2627	0.3169	0.2627	0.3169	0.2763	0.2898	0.3712	0.6695	0.3034
0.1407	0.1271	0.1136	0.1271	0.1136	0.1678	0.2492	0.2627	0.3983	0.2627
0.1407	0.3712	0.4525	0.3712	0.4525	0.6424	0.4390	0.6153	0.8322	0.5610
0.2220	0.3305	0.3576	0.3305	0.3576	0.4390	0.3034	0.4661	0.3034	0.4254
0.3169	0.2356	0.3034	0.2356	0.3034	0.1949	0.2085	0.2356	0.2627	0.1949
0.4390	0.3305	0.5203	0.3305	0.5203	0.4119	0.3983	0.3305	0.3441	0.3034
0.1814	0.4390	0.3576	0.4390	0.3576	0.2356	0.2220	0.2492	0.2220	0.2085
0.1203	0.2085	0.1678	0.2085	0.1678	0.3441	0.3847	0.3847	0.1000	0.4661
0.3169	0.1949	0.3169	0.1949	0.3169	0.3305	0.2627	0.4525	0.3983	0.4390
0.2085	0.3847	0.3305	0.3847	0.3305	0.2627	0.2492	0.2627	0.4119	0.5475
0.1407	0.3034	0.2492	0.3034	0.2492	0.4797	0.5610	0.5203	0.5881	0.5746
0.1000	0.1000	0.3034	0.1000	0.3034	0.9000	0.5475	0.3983	0.2085	0.2356
0.1000	0.1136	0.1542	0.1136	0.1542	0.2627	0.4390	0.4661	0.5610	0.4119
0.2492	0.2898	0.4797	0.2898	0.4797	0.6153	0.8458	0.4932	0.3034	0.5203
0.3712	0.3441	0.3847	0.3441	0.3847	0.2356	0.2220	0.4119	0.4254	0.2356
0.3305	0.3441	0.2220	0.3441	0.2220	0.3034	0.2356	0.3712	0.4119	0.2220

Table A.8: Outputs of the ANN

Desired Output	ANN Output
1.3500	1.7646
2.3000	1.9247
4.2500	6.8313
0.7600	1.6039
1.9300	1.6810
2.3500	1.7650
2.1200	1.4365
2.2500	1.7632
3.1000	4.5970
1.1800	1.7381
1.6600	1.7044
3.1200	2.4774
0.2000	3.2399
10.9400	5.8677
3.8300	1.6033
0.8800	1.7698
1.9800	2.1661

Table A.9: Outputs of the ANN

Desired Output	ANN Output
1.3500	2.1258
2.3000	1.1002
4.2500	4.3435
0.7600	1.8020
1.9300	1.3076
2.3500	1.4541
2.1200	2.5654
2.2500	1.5778
3.1000	4.2312
1.1800	1.9859
1.6600	2.0001
3.1200	1.9113
0.2000	1.7590
10.9400	9.5145
3.8300	1.6203
0.8800	1.4746
1.9800	2.2454

Table A.10: Outputs of the ANN

Desired Output	ANN Output
1.3500	1.8542
2.3000	1.4162
4.2500	4.5970
0.7600	2.1072
1.9300	1.3304
2.3500	2.0372
2.1200	2.3917
2.2500	2.1707
3.1000	3.9524
1.1800	2.3728
1.6600	1.9803
3.1200	1.5427
0.2000	0.9281
10.9400	9.8926
3.8300	1.7165
0.8800	1.4453
1.9800	1.4301

Table A.11: Outputs the ANN

Desired Output	ANN Output
1.3500	1.0211
2.3000	1.6034
4.2500	4.3878
0.7600	1.0352
1.9300	0.6725
2.3500	2.4188
2.1200	2.2938
2.2500	2.5064
3.1000	4.2459
1.1800	2.0090
1.6600	1.5897
3.1200	3.2052
0.2000	-0.1908
10.9400	9.6910
3.8300	3.8456
0.8800	1.0021
1.9800	1.3555

Table A.12: Outputs of the ANN

Desired Output	ANN Output
1.3500	0.8648
2.3000	1.6871
4.2500	4.3784
0.7600	1.1209
1.9300	0.6541
2.3500	2.1247
2.1200	2.1786
2.2500	2.5470
3.1000	3.9628
1.1800	2.1596
1.6600	1.6840
3.1200	3.3442
0.2000	-0.6391
10.9400	9.8201
3.8300	3.8134
0.8800	1.3142
1.9800	1.2975

Table A.13: Outputs of the ANN

Desired Output	ANN Output
1.3500	1.1073
2.3000	2.0664
4.2500	4.5067
0.7600	0.9033
1.9300	0.8269
2.3500	1.9928
2.1200	2.3630
2.2500	2.6155
3.1000	3.0917
1.1800	1.9566
1.6600	1.6773
3.1200	3.5904
0.2000	0.2631
10.9400	10.3062
3.8300	3.6590
0.8800	1.0939
1.9800	1.4076

Table A.14: Outputs of the ANN in Training Phase

Desired Output	ANN Output
1.3500	0.7608
2.3000	1.9975
4.2500	4.4764
0.7600	1.1864
1.9300	0.7002
2.3500	2.2549
2.1200	2.1659
2.2500	2.6131
3.1000	3.0013
1.1800	1.7846
1.6600	1.6802
3.1200	3.4958
0.2000	0.2895
10.9400	10.3523
3.8300	3.7431
0.8800	1.1717
1.9800	1.7621

Table A.15: Outputs of the ANN
in Training Phase

Desired Output	ANN Output
0.1916	0.2253
0.6344	0.4586
0.3626	0.5669
0.1000	0.2362
0.8053	0.7577
0.7931	0.5525
0.5672	0.5221
0.6771	0.5087
0.1214	0.0889
0.1183	0.1835
0.1824	0.1430
0.2038	0.3539
0.4145	0.4005
0.3962	0.5995
0.1489	0.1920
0.2160	0.2307
0.2008	0.0855
0.1702	0.1324
0.4634	0.5051
0.1580	0.2010
0.3931	0.5850
0.6618	0.6264
0.3137	0.5127
0.2679	0.3520
0.9000	0.4132

Table A.16: Outputs of the ANN
in Test Phase

Desired Output	ANN Output
0.4939	0.4482
0.4298	0.5620
0.3107	0.6071
0.3901	0.1677
0.2954	0.5880
0.2618	0.3815
0.5611	0.2229
0.2008	0.2826
0.1672	0.4978
0.3779	0.5112
0.2954	0.1556
0.2649	0.3418

Table A.17: Outputs of the ANN
in Training Phase

Desired Out	ANN Output
0.1916	0.1828
0.6344	0.5204
0.3626	0.4292
0.1000	0.1110
0.8053	0.8132
0.7931	0.5634
0.5672	0.5793
0.6771	0.4812
0.1214	0.0858
0.1183	0.1724
0.1824	0.1667
0.2038	0.1839
0.4145	0.4338
0.3962	0.6110
0.1489	0.2362
0.2160	0.2897
0.2008	0.0752
0.1702	0.0991
0.4634	0.3954
0.1580	0.1811
0.3931	0.4045
0.6618	0.6091
0.3137	0.5784
0.2679	0.2986
0.9000	0.8513

Table A.18: Outputs of the ANN
in Test Phase

Desired Output	ANN Output
0.4939	0.2803
0.4298	0.7827
0.3107	0.3098
0.3901	0.1435
0.2954	0.4255
0.2618	0.2792
0.5611	0.2898
0.2008	0.1525
0.1672	0.3771
0.3779	0.3417
0.2954	0.1178
0.2649	0.3277

Table A.19: Outputs of the ANN in Training Phase

Desired Output	ANN Output
0.1916	0.2814
0.6344	0.6446
0.3626	0.4011
0.1000	0.0810
0.8053	0.8000
0.7931	0.7502
0.5672	0.5720
0.6771	0.6016
0.1214	0.1159
0.1183	0.1266
0.1824	0.1549
0.2038	0.1981
0.4145	0.4303
0.3962	0.4178
0.1489	0.1702
0.2160	0.2462
0.2008	0.1302
0.1702	0.1558
0.4634	0.4537
0.1580	0.1804
0.3931	0.4025
0.6618	0.6745
0.3137	0.3158
0.2679	0.1880
0.9000	0.8950

BIOGRAPHY

Aynur Zöngör was born in 1979 in Bingöl, Turkey. She was graduated from Uskudar Girl High School in 1997 and attended to Istanbul Technical University in the same year. She received B.Sc. degrees in Electronics and Communication Engineering in 2003. She received M.Sc. degree in Biomedical Engineering from Istanbul Technical University in 2007.