MAY 2022

M.Sc. – Electronics and Computer Engineering

FURKAN ÖZKAN

HASAN KALYONCU UNIVERSITY

GRADUATE SCHOOL OF

NATURAL & APPLIED SCIENCES

PREDICTION OF LONG-TERM STREAMFLOW BY USING ADAPTIVE NEURO-FUZZY INFERENCE SYSTEM (ANFIS)

ELECTRONICS AND COMPUTER ENGINEERING
M. Sc. THESIS

BY

FURKAN ÖZKAN

MAY 2022

PREDICTION OF LONG-TERM STREAMFLOW BY USING ADAPTIVE NEURO-FUZZY INFERENCE SYSTEM (ANFIS)

Hasan Kalyoncu University

Electronics and Computer Engineering

M. Sc. Thesis

Supervisor

Assist. Prof. Dr. Bülent HAZNEDAR

Furkan ÖZKAN

May 2022

GRADUATE SCHOOL OF NATURAL & APPLIED SCIENCES INSTITUTE
M.Sc. ACCEPTANCE AND APPROVAL FROM

Electronics-Computer Engineering M.Sc. (Master of Science) program student **Furkan ÖZKAN** prepared and submitted the thesis titled "**Prediction of long-term streamflow by using adaptive neuro-fuzzy inference system (ANFIS)**" defended successfully on the date of 20/05/2022 and accepted by the jury as a M.Sc. thesis.

Position	**Title, Name and Surname** **Department/University**	**Signature:**
M. Sc. Supervisor	Assist. Prof. Dr. Bülent HAZNEDAR Computer Engineering Department Gaziantep University	
Jury Head	Prof. Dr. M. Fatih HASOĞLU Computer Engineering Department Hasan Kalyoncu University	
Jury Member	Assist. Prof. Dr. Mustafa BIÇAKCI Computer Engineering Department Hasan Kalyoncu University	

This thesis is accepted by the jury members selected by institute management board and approved by institute management board.

Prof. Dr. İbrahim Halil GÜZELBEY
Director

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Furkan ÖZKAN

ABSTRACT

LONG-TERM STREAMFLOW FORECASTING USING ADAPTIVE NEURO-FUZZY INFERENCE SYSTEM (ANFIS)

ÖZKAN, Furkan
M.Sc. in Electronic - Computer Engineering
Supervisor: Asst. Prof. Dr. Bülent HAZNEDAR
May 2022
93 Pages

Water resources are one of the most basic needs of living life. In order to sustain human life without any problems, a rational planning is required for the protection and use of existing water resources. At the beginning of the plans to be made, the potential of the water source to be used in the future should be determined. Therefore, river flow estimation is necessary to provide basic information on a wide variety of problems associated with the functioning of river systems. In this study, the daily flow values of Zamanti River-Değirmenocağı, Zamanti River-Ergenuşağı and Eğlence River-Eğribük stations in the Seyhan Basin in Turkey were investigated. has been made. Artificial intelligence methods have been used instead of traditional methods for some time due to their success in modelling complex nonlinear problems. Within the scope of the thesis, the Adaptive Neuro-Fuzzy Inference System (ANFIS) model was trained using Simulated Annealing (SA), Back Propagation (BP) and Hybrid Learning (HB) algorithms in order to make forward flow rate estimation from past flow measurement values and the results obtained from all models were compared. Mean Absolute Error (MAE), Root Mean Square Error (RMSE), Determination Coefficient (R^2) and Mean Absolute Percentage Error (MAPE) evaluation criteria were used for comparison. After the analysis, it was concluded that HB and BP algorithms can be used more successfully and effectively than SA algorithm in training ANFIS parameters in nonlinear problems.

Key Words: Neuro-Fuzzy, ANFIS, Hydrology, Streamflow, Prediction

ÖZET

UYARLANABİLİR AĞ TABANLI BULANIK MANTIK ÇIKARIM SİSTEMİ (ANFIS) KULLANILARAK UZUN VADELİ NEHİR AKIM TAHMİNİ

ÖZKAN, Furkan
Yüksek Lisans, Elektronik Bilgisayar Mühendisliği
Tez Danışmanı: Dr. Öğr. Üyesi. Bülent HAZNEDAR
Mayıs 2022
93 Sayfa

Su kaynakları, canlı yaşamın en temel ihtiyaçlarından biridir. İnsan yaşamını sorunsuz bir şekilde sürdürebilmek için mevcut su kaynaklarının korunması ve kullanımına yönelik akılcı bir planlama gerekmektedir. Yapılacak planların başında kullanılacak su kaynağının gelecekte yaratacağı potansiyel belirlenmelidir. Bu nedenle, nehir sistemlerinin işleyişi ile ilgili çok çeşitli problemler hakkında temel bilgi sağlamak için nehir akış tahmini gereklidir. Bu çalışmada, Türkiye'de Seyhan Havzası'nda bulunan Zamanti Nehri-Değirmenocağı, Zamanti Nehri-Ergenuşağı ve Eğlence Deresi-Eğribük istasyonlarının günlük akım değerleri incelenmiştir.1988-2011 yılları arasında üç ayrı istasyon için ölçülmüş olan 8928 günlük akım verileri kullanılarak akış miktarı tahmini yapılmıştır. Yapay zekâ metotları karmaşık doğrusal olmayan problemleri modellemedeki başarımlarından kaynaklı bir süredir geleneksel yöntemlerin yerine kullanılmaktadır. Tez kapsamında geçmiş akım ölçüm değerlerinden ileriye dönük akış hızı tahmini yapılması amacıyla Uyarlamalı Ağ Tabanlı Bulanık Çıkarım Sistem (ANFIS) modeli Benzetimli Tavlama (SA), Geri Yayılım (BP) ve Hibrid Öğrenme (HB) algoritmaları kullanılarak eğitilmiş ve tüm modellerden elde edilen sonuçlar karşılaştırılmıştır. Kıyaslama için Ortalama Mutlak Hata (MAE), Kök Ortalama Kare Hata (RMSE), Belirleme Katsayısı (R^2) ve Ortalama Mutlak Yüzde Hata (MAPE) değerlendirme kriterleri kullanılmıştır. Yapılan analizler sonrasında doğrusal olmayan problemlerde ANFIS parametrelerini eğitmede HB ve BP algoritmalarının SA algoritmasından daha başarılı ve etkili şekilde kullanılabileceği sonucuna varılmıştır.

Anahtar Kelimeler: Nöro-Bulanık, ANFIS, Hidroloji, Nehir Akışı, Tahmin

To my Worthy Family

ACKNOWLEDGMENTS

I would like to thank my thesis advisor, Asst. Prof. Dr. Bülent HAZNEDAR, for his valuable contributions during our thesis work. I would like to express my gratitude to our department head, Prof. Dr. M. Fatih HASOĞLU, for his unwavering interest and support throughout the department, and to my beloved family who have always been by my side and supported me during the writing of the thesis.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS AND ABBREVIATIONS

ANN	Artificial Neural Network
ANFIS	Adaptive Neuro-Fuzzy Inference System
BP	Back Propagation Algorithm
FCM	Fuzzy C-Means
GP	Grid Partitioning
HB	Hybrid Learning Algorithm
MAE	Mean Absolute Error
MAPE	Mean Absolute Percentage Error
MF	Membership Function
RMSE	Root Mean Square Error
R^2	Determination Coefficient
SA	Simulated Annealing

CHAPTER 1

INTRODUCTION

Water resources are one of the most basic needs of living life. In order to maintain human life without any problems, a rational planning is required for the protection and use of existing water resources. At the beginning of the plans to be made, the potential created by the water source to be used and the potential to be made in the future is to be determined. Thanks to its current potential, water is a natural resource that meets the energy needs of humanity. So, river flow estimation is necessary to provide basic information on a wide variety of problems associated with the operation of river systems (Dibike et al., 2001). The availability of extended precipitation records and other climate data that can be used to derive streamflow data has prompted the practice of runoff forecasting modelling. Behaviour detection is possible with the observation values of the current measuring stations. Prospective estimation of river flow rates can be made with the help of time series.

The water loss situation that may occur for the world can be explained as there can be no life anymore. Because water is a resource that cannot be artificially produced, created or imitated. These limited water resources are very important for all living things to survive. For this reason, the value of water and water resources should be well known and acted accordingly. Although Turkey seems to be rich in terms of water resources, when the per capita water consumption is considered, it is well below the world average (Tombul, 2014).

In recent years, the estimation of river flow in hydrological processes has become very important to facilitate the sustainable use and effective planning and management of water resources (Firat et al., 2010). Detailed observational data of the river network is required to predict hydrological processes such as precipitation, flow and water level change with current methods.

Artificial Intelligence (AI) is a set of software and hardware systems used to encompass a wide variety of sub-fields dedicated to creating algorithms to perform complex behaviors such as sensing, learning, inferring, and decision-making that human intelligence is capable of (Mutasa et al.,2020). Artificial neural networks have been successfully applied in the modelling of non-linear systems, and because of this feature, successful results have been obtained in modelling hydrological events. Artificial Intelligence techniques have gained momentum in the last two decades, particularly for modelling and assessment of disruption of water distribution networks. Artificial intelligence techniques are frequently and successfully used in the estimation of stream flow values.

Artificial neural network (ANN) and Deep Learning (DL) models from Artificial Intelligence techniques are used as prediction methods. The estimation results provide appropriate planning for both the producer and the user for the production of water resources for living things, irrigation, hydroelectric power generation and transferring water to future generations. The flow in the rivers is determined by the measurement stations established by the relevant institutions at certain points of the river.

The precipitation-flow relationship has been modelled with different artificial neural network methods (Alp et al., 2010). They used the feedforward backpropagation method and the generalized regression neural network method. Modelling was done using daily flow values from the Juniata River in Pennsylvania, USA. It has been determined that the feedforward backpropagation method gives better results than the generalized regression neural network method.

In recent years, time series modelling and forecasting has become one of the most common research topics in academic research and engineering applications. (Wang et al., 2015). Time series modelling consists of a chronological series of observed data according to a time series that is usually sampled at invariant time intervals. Researchers often predict future changes and probabilities based on past data. For example, changes in the river flow rate according to the situation in the past or current period, an increase in precipitation and an increase or decrease in the number of living things living in the river are predicted in the future. Time series forecasting affects people's lives everywhere and plays an important role in every aspect of today's society, creating research prospects. For this reason, this situation is also of great importance in the field of computer application. River flow forecasting management has been created for the forecasting and analysis of the flow of rivers in selected regions.

Traditional methods usually give good results in solving small-sized problems. Conceptual or physically based models are important in understanding hydrological processes (Behzad, 2009). In fact, the statistical properties of time series data often violate the assumptions of traditional methods. Therefore, analysing time series data requires a unique set of tools and methods known as time series analysis (Altunkaynak et al., 2018).

In addition to traditional methods, after successful applications of artificial neural networks in water engineering, fuzzy logic models as an alternative method have also found application in water engineering. Successful results were obtained in the models made with fuzzy logic models. (Mahabir et al., 2003) used fuzzy logic modelling techniques in Canadian Lodge Creek and Middle Creek airfields for seasonal flow estimation and water supply. The results were compared with previous studies, and it was seen that the results of the Fuzzy Logic method were better. Seasonal flow estimates from fuzzy expert systems were found to be much more reliable than regression models, especially in terms of identifying low or average level flow years and predicting the appropriate flow region. Based on the modelling results in these two basins, it was concluded that fuzzy logic has significant and powerful potential to provide reliable water supply forecasts (Jiang et al., 2021).

(Bisht et al., 2011) developed ANFIS and Linear Multiple Regression (MLR) methods for flow estimation in rivers in this study. They observed that ANN models and Fuzzy Logic models are applicable for flow prediction in rivers. With the developed models, they trained and tested the data of the Gadavri river. When they compared the data, they observed that the developed ANFIS models gave better results than traditional models such as MLR.

The aim of this study is to investigate the applicability of artificial intelligence techniques such as ANFIS (Adaptive Neuro-Fuzzy Inference System) in the estimation of flow rate in the river system. In the study, it is planned to measure success in river flow rate problems by using the HB and BP algorithms, which are the optimization algorithms of ANFIS, and the iterative based SA algorithm. Performance analysis of ANFIS structures in nonlinear problems is made by comparing traditional derivative-based BP and HB algorithms with SA, which is Artificial Intelligence algorithm. Applications have been made on the estimation of the future flow value by using past river flows and the estimation of the measurement at the flow observation station, which cannot be measured. 24 years of data per station and 3 stations were used for the implementation of the study. ANFIS models with various input structures and membership functions are created, trained and tested to evaluate the efficiency of the models. Statistical indices such as MAPE, MAE, RMSE and R^2 are used to evaluate the performance of ANFIS models to predict river flow velocity. The values of the indices indicate that the ANFIS model can be used accurately and reliably to predict the flow rate in a river system.

The main purpose of this study is to investigate the applicability and capacity of ANFIS and ANN methods for daily river flow estimation. To validate the application of this approach, the Seyhan basin, located in the southern part of Turkey, was chosen as the case study area. Zamanti River-Değirmenocağı, Zamanti River-Ergenuşağı and Eğlence River-Eğribük are one of the most important water resources of the Seyhan basin in Turkey. The flow of rivers depends on various factors such as climatic variables of the basin, water use for agriculture and hydroelectric power generation. Models have been developed for modelling river flow with various input structures and applied to the estimation of river flows. Accurate estimates are obtained using a twenty-four-year dataset and specialized processing of input data.

This study has emerged in order to make a river flow estimation by using the river flow data of the stations effectively. In general, the content of this thesis is organized as follows: In the second chapter, titled "Literature Review", studies in the literature on the problem of river flow rate are mentioned and information about classical approaches is given. In the third chapter titled "Materials and Methods", detailed information is given about the data set and stations used. Information on how the algorithms used are trained and the parameter optimization results of the created models are examined. Afterwards, the evaluation methods used in the comparison of the models are mentioned. In the fourth chapter titled "Results and Discussion", the models we created as a result of the study are compared among themselves. The interpretations of the models were made according to the performance evaluation criteria. Performance evaluation criteria for the models we have created are presented with explanations. In the last section titled "Conclusion", the studies are summarized and information is given about the advantages and disadvantages of the models obtained as a result of the study. Ideas about future work in this area have been put forward.

CHAPTER 2

LITERATURE REVIEW

Use of Artificial Intelligence Methods in hydrology recently, many studies have been carried out with artificial intelligence techniques such as Fuzzy Logic, ANN and ANFIS (Sarma et al., 2016). When these studies are examined, both the convenience they provide in modelling and the compatible results obtained show that these methods can be applied. While modelling with these methods, only the current flow data is used without the need for the physical characteristics of the basin or the region. In addition, these methods are advantageous in solving nonlinear problems in terms of approach.

In (Topçu, 2021) it is aimed to make a drought analysis in the Middle Euphrates Section, which is one of the driest parts of Turkey, using the Mamdani Fuzzy Logic method and the Monthly Aydeniz method. Monthly average precipitation, monthly average temperature, monthly average humidity, and monthly total sunshine duration parameters obtained from 3 meteorological observation stations, Gaziantep, Adıyaman and Şanlıurfa, were used as input parameters in the analysis and drought analysis was carried out. According to the results obtained, it has been observed that the entire region has a mostly desert climate. The drought coefficient values obtained by the Monthly Aydeniz method and accepted as the observed values were tried to be estimated by creating 85 rule bases for the purpose of drought determination in the Fuzzy Logic Approach. The average correct guessing rate is over 83%. Thus, it is expected that the methods used in drought analysis in the region will guide the decision makers in the planning of water resources.

The reservoir control system for Yuvacık dam was modelled using the Fuzzy Logic System method, with four input parameters as losses, precipitation, inflow and outflow flow, and the lake water level as output parameter. A fuzzy logic system has been developed by using very low, low, medium, high and very high membership functions for reservoir losses in the Fuzzy Logic System and the same membership functions are used for all inputs and outputs (Bizimana et al., 2016). In this study, meteorological and hydrological variables such as precipitation, water losses, inflow, outflow, evaporation, and humidity, which are the most important parameters in water resources management, are discussed. Mamdani inference system, which is a fuzzy logic inference method, was used in the analysis of these parameters. The findings show that Fuzzy Logic System can be used successfully to regulate any reservoir level. The results of the study showed that the fuzzy theory is more flexible and reliable in reservoir control compared to other systems, since it also handles different hydrological uncertainties and the relationship between inputs and output can be interpreted in the rules.

It is (Mahabir et al., 2003) investigated the feasibility of water supply forecasting using fuzzy logic modelling techniques with the help of 15-year datasets from Lodge Creek and Middle Creek watersheds in southeastern Alberta. Applying fuzzy logic, a water resource estimate was created by classifying the potential flow as low, average and high. Findings from fuzzy expert systems were found to be far superior to regression models for predicting water supply. Based on the modelling results in these two basins, it was concluded that fuzzy logic has promising potential to provide reliable water supply predictions (Astel, 2007).

In (Gunathilake et al., 2021) hydrological modelling of the Seethawaka River Basin was performed with Artificial Neural Networks by simulating the flow and comparing the flow through data-driven techniques. Daily precipitation data was fed into the system as input and the Neural network was trained using stream flow data. Seventy percent of the time series data was used in the training process, while 15% of each was used in the validation and testing processes. Training algorithms such as Levenberg-Marquardt (LM), Bayesian regularization (BR) and scaled conjugate gradient (SCG) were used in the training process of the developed ANNs. Among the training algorithms, the best one was selected based on the R^2 and the MSE.

ANN was used using daily rainfall and runoff data from Aghbalou station in a basin in a semi-arid climate in Morocco (Riad et al., 2004). The data were taken by Rabat Hydraulic in a seven-year period between 1990-1996, the first 6 years for training and the last year for testing. In this study, a multilayer perceptron (MLP) neural network was preferred to model the precipitation flow relationship. During this time, 2550 precipitation and runoff pair values were effective in modelling. Based on the results, it clearly shows that the artificial neural network method can improve better than the classical regression model in such arid regions where precipitation and surface flow are very irregular. ANN approach emerges as a suitable tool for solving problems in water resource management in studies in hydrology.

Jimeno-Saez et al. (Jimeno-Sáez et al., 2018) compared models using the Soil and Water Assessment Tool (SWAT) and ANN, a machine learning technique, to find a way to improve flow prediction. In this study, the daily precipitation data and temperature values of the basins with two different climatic conditions in the Spanish Peninsula were modelled. For the input type and number of ANN models, four scenarios including daily precipitation, previous days precipitation, and previous days total precipitation were included, showing that they are important for estimating daily runoff flow. The fact that the ANN model does not use the current physical state of the basin has made modelling easier. The results were successful in estimating very low flow values with the SWAT model, but ANN was found to be superior because ANN better simulates high flows in all cases studied.

In (Dalkiliç et al., 2020) ANN, wavelet neural network (WNN) and ANFIS models were selected to predict daily flow at four different stations located near the Büyük Menderes River. The RMSE, R^2, Nash–Sutcliffe (NS) and RMSE-observation standard deviation ratio (RSR) evaluation indices were used to compare the outputs of the models. Average temperature, precipitation, relative humidity and daily flow data obtained from Muğla meteorology station are given as input to verify the performance of the models. 70% of the most recent data available was used in 1996–2007 training and 30% of the data 2008–2011 was used in testing. Looking at the results, the WNN model outweighed the ANN and ANFIS models.

It is aimed to apply a new hybrid model named ANFIS-PSO for monthly flow estimation at BP ghat and Dholai measuring stations located in the Barak River basin near India. A hybrid model was developed combining particle swarm optimization (PSO) algorithm and ANFIS (Samanataray et al., 2021). Various input units such as monthly precipitation, temperature, humidity have been applied in ANFIS models. The RMSE, MAE, R^2, and Nash–Sutcliffe Coefficient (ENS) evaluation indices were used to compare the outputs of the models. 70% of the data was used in 1980 – 2007 training and 30% of the data 2008 – 2019 was used in testing. It has been observed that the ANFIS-PSO model has the best performance compared to the independent ANFIS and ANN models.

(Kisi et al., 2012) analyzed the lake level changes of ANN, ANFIS and gene expression programming (GEP) approaches in their study. Daily lake level data of Iznik lake, which is the largest lake in the Marmara region of Turkey, was used. In all models, the first half of all data was applied in the training phase, the last half in the testing phase for 6 years, and the remaining 5 years used in the validation phase of the optimal models. Model capacities were calculated using statistical evaluation criterias; R^2, RMSE, distribution index (SI) and adjusted fit index (d1). It has been suggested that the GEP model gives better results than the ANFIS and ANN models in the lake level change problem. In addition to this analysis, all models were compared among themselves with autoregressive moving average (ARMA). Although the linear nature of the data prevented large differences in the results of AI models and ARMA, AI models produced better predictions. Likewise, (Buyukyıldız et al., 2014) monthly lake level changes were estimated based on the 29-year data of Beyşehir Lake, which ranks third among the lakes in Turkey. 80% of the data was used in the training process and the remaining 20% was used in the testing phase. In this study, the inflow, precipitation, evaporation and outlet values in the existing lake were analyzed with different combinations, and the change in water level was observed, and the monthly variation was calculated by taking the elevation differences compared to the previous month. Particle Swarm Optimization (PSO), Support Vector Regression (SVR), Multilayer Artificial Neural Networks (MLP), Radial Based Neural Networks (RBNN) techniques have been used for this purpose. The estimation results of the applied models were evaluated with the help of RMSE, mean square error (MSE), and R^2 criteria. When the results were analyzed, it was determined that ε-SVR was the model with the highest performance with an R^2 value of 0.9988, while ANFIS, MLP and RBNN methods also performed well.

In another study (Niu et al., 2021) using daily flow and precipitation data, artificial neural network, adaptive neural-based fuzzy inference system, extreme learning machine (ELM), Gaussian process regression (GPR) and support vector machine (SVM) artificial intelligence methods were used to examine the daily flow potential. In the study, Hongjiadu reservoir and Xinfengjiang reservoir in China were analyzed as two different areas. Daily flow series of hydroelectric reservoirs were estimated using 10- and 15-years data, respectively. In performance measurement, RMSE, mean absolute relative error (MARE), correlation coefficient (R), Nash-

Sutcliffe efficiency coefficient (CE) criteria were taken into consideration. All the artificial intelligence techniques used performed well in daily flow problems, and when the models were compared, SVM, GPR and ELM techniques found the best results when evaluated in the given criteria.

CHAPTER 3

MATERIALS AND METHOD

3.1 Materials (Study Area)

In this study, the daily flow values of Zamanti River-Değirmenocağı, Zamanti River- Ergenuşağı and Eğlence Stream-Eğribük stations in the Seyhan Basin in Turkey were investigated. The data used was provided by the DSİ (Hydraulic State Works). The upper part of the Seyhan Basin is located in Central Anatolia, the middle and lower part in the Mediterranean Region, in the north of Adana province, between 36º 30' and 39° 15' north latitudes and 34º 45' and 37º 00' east longitudes. It has a length of 560 km and a surface area of 22035 km² (TOB, 2019).

E18A025 Eğlence Stream-Eğribük Station is a tributary of the Seyhan River, has an altitude of 222 m and a forehead of 544.5 km² of precipitation. This current observation station is located between 35°11'35" East longitudes and 37°21'50" North latitudes in Adana province Karaisalı district. The 24-year flow data at the station between 1988-2011 are 8928 daily measured flow values. The position of the related station in the Seyhan basin on the map is given in Figure 3.1. In Figure 3.2, the current values of this station are given graphically.

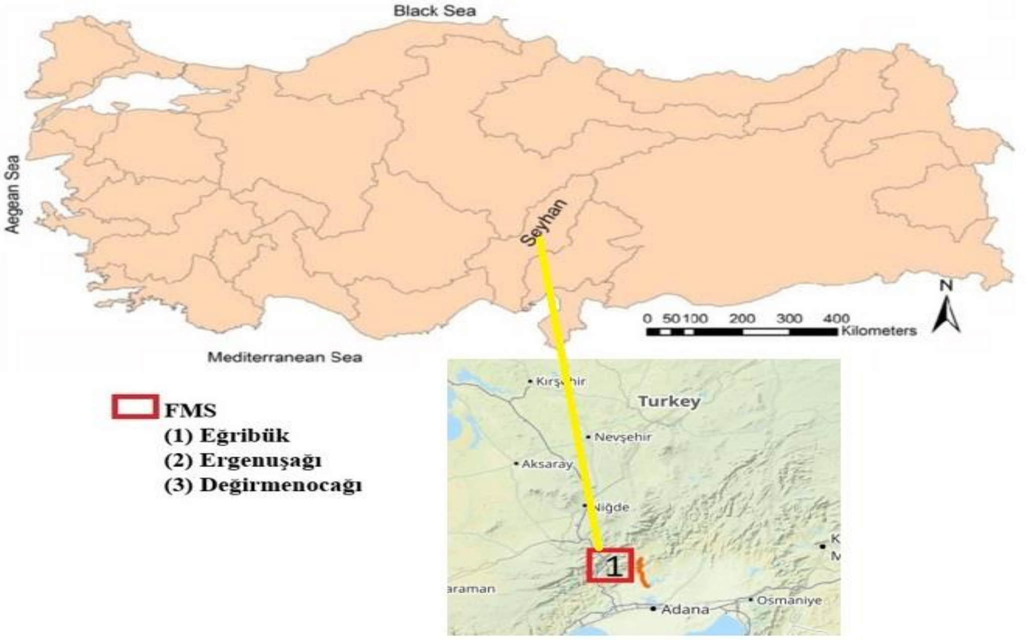


Figure 3.1: Eğribük (1825) station's location in the Seyhan river basin

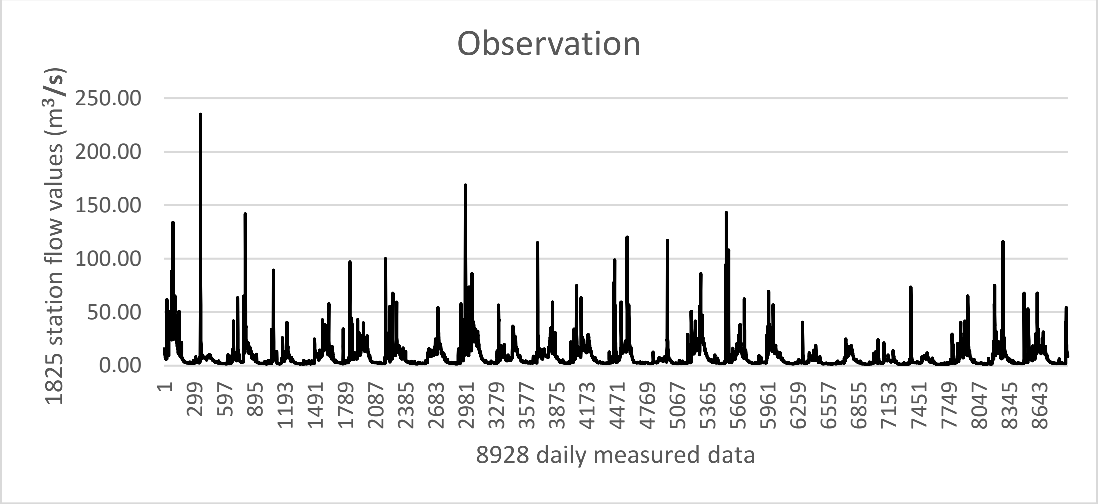


Figure 3.2: 1825 station's flow values

Zamanti River-Ergenuşağı Station, numbered E18A026, is a tributary of Seyhan River, has an altitude of 360 m and a forehead of 8698.1 km² precipitation. This current observation station is located in Kozan district of Adana province, between 35°34'47" East longitudes and 37°39'55" North latitudes. The 24-year flow data at the station between 1988-2011 are 8928 daily measured flow values. The positions of the Ergenuşağı and Değirmenocağı stations in Seyhan basin on the map are given in Figure 3.3. In Figure 3.4, the current values of this station are given graphically.

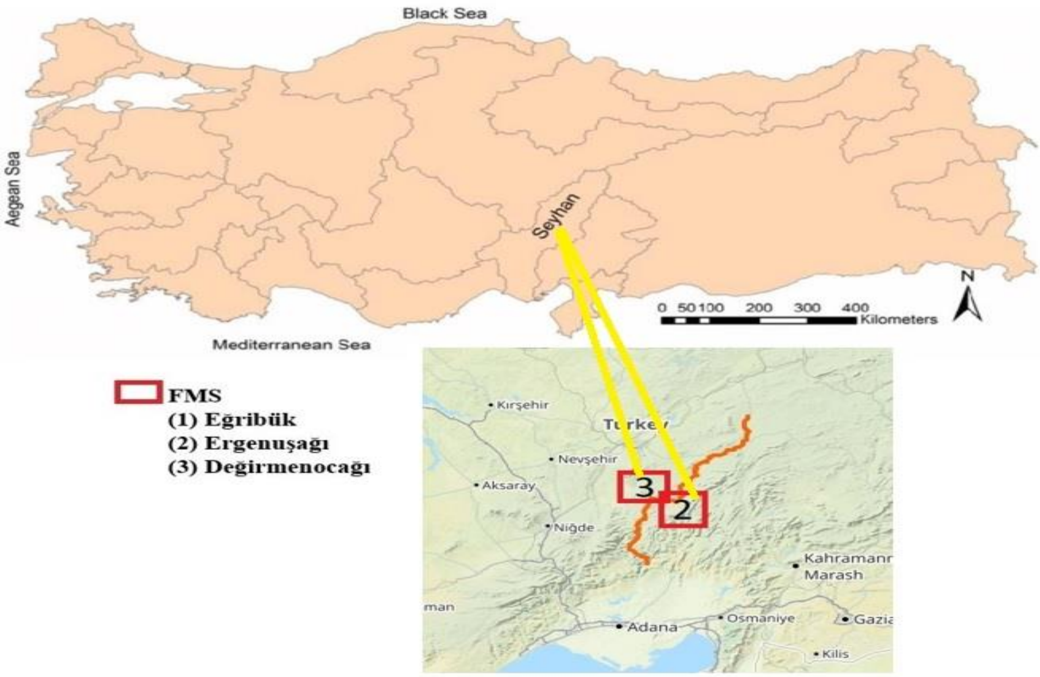


Figure 3.3: Locations of Ergenuşağı (1826) and Değirmenocağı (1827) stations

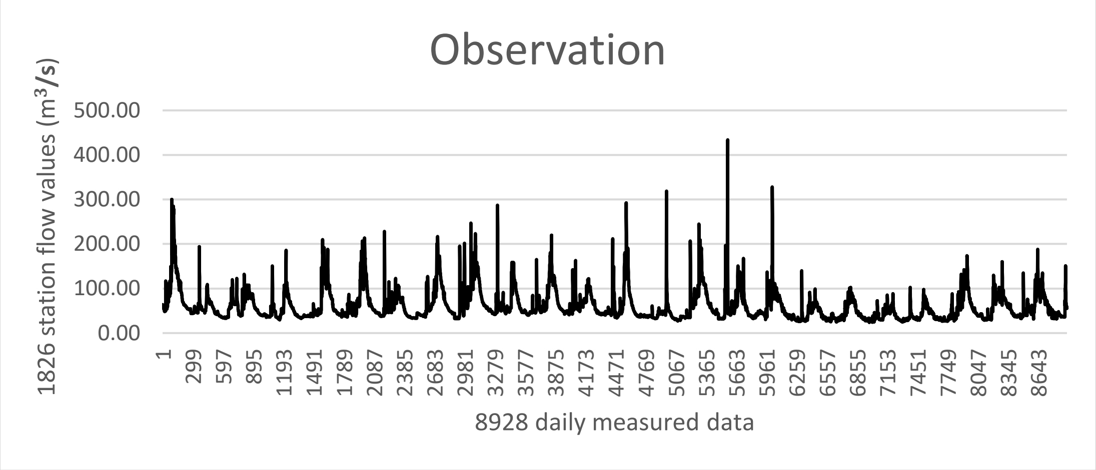


Figure 3.4: 1826 station's flow values

Zamanti River-Değirmenocağı Station, numbered E18A027, is a tributary of the Seyhan River, has an altitude of 740 m and a forehead of 7718 km² of precipitation. This current observation station is located in Yahyali district of Kayseri province, between 35°29'10" East longitudes and 37°51'18" North latitudes. The 24-year flow data at the station between 1988-2011 are 8928 daily measured flow values. In Figure 3.5, the current values of this station are given graphically.

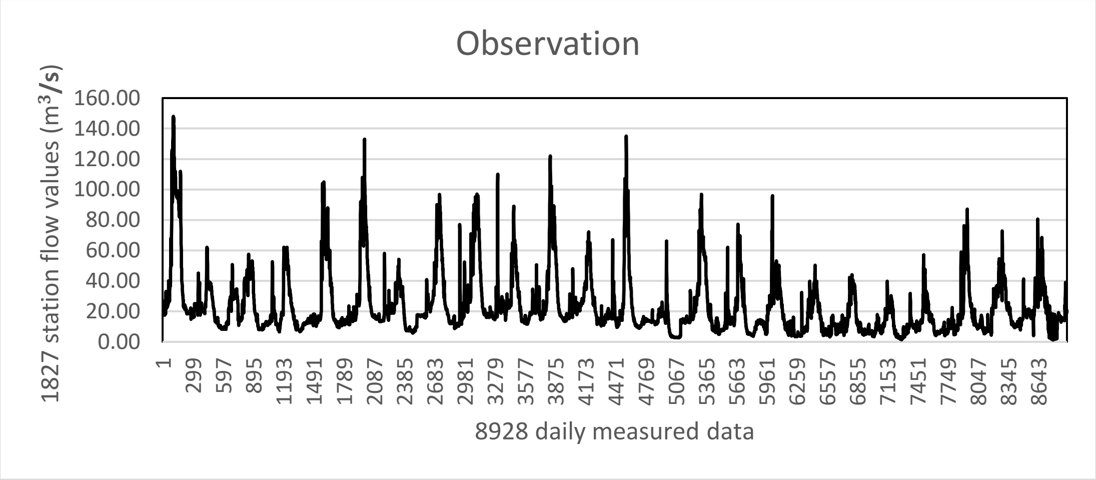


Figure 3.5: 1827 station's flow values

3.2 Methods

3.2.1 Fuzzy Logic and Fuzzy Model

The real world is complex. The difficulties encountered in nature have opened the door to the modelling of the uncertainties they create with new methods. Fuzzy logic is concerned with formal principles of precise and approximate reasoning, which is seen as a limiting case. The mathematical approach adapted to classical logic emerges as a method with sharp boundaries. In summary, in fuzzy logic, unlike classical logical systems, it is the modelling of imprecise reasoning methods to make rational decisions in an environment of uncertainty and uncertainty (Yuen et al, 2006). The concept of fuzzy logic, which is one of the leading areas of these methods, emerged contrary to Aristotle's logic. In Western culture (Boolean), Aristotle has a dual value logic. Fuzzy logic generates very valuable results by considering values between 0-1 and makes it possible to use variables whose sizes are suitable for verbal language, together with numerical data in solving problems.

Although the first information about the concepts of fuzzy logic was brought to the literature by Zadeh (Zadeh, 1965) in 1965 in the modern sense, this idea was not welcomed in the Western world at first. Although the first information about the concepts of fuzzy logic was brought to the literature by Zadeh in 1965 in the modern sense, this idea was not welcomed in the Western world at first. Fuzzy logic has been adopted more in Turkey and Japan than in the West. The reason for this is that the transition and continuity in fuzzy logic with the philosophical thought structure of the eastern world is appropriate. In classical logic there is 0 or 1. So it either exists or it does not. But fuzzy logic shows us that it has different expressions such as it can exist or be out of nothing. One of the most important reasons for the increase in the importance of fuzzy logic is the important technological developments that were made using fuzzy logic in Japan after 1970.

The curve that changes with the values of the set members is called the membership function. The membership function (MF) is a curve that describes how each point in the input space is mapped to a membership value (or degree of membership) between 0 and 1 (Mistry et al., 2014). Thus, the items collected under the membership function have a membership degree according to their importance. In general, fuzzy systems are systems that use fuzzy set principles to obtain output variables from input variables selected from the available data. Membership functions in different forms can be used while creating fuzzy models in studies with fuzzy logic. Although there are many membership functions, the most frequently used membership functions are triangular, trapezoidal, normal distribution (gaussian) and bell-shaped membership functions. These membership functions are shown in Figure 3.6.

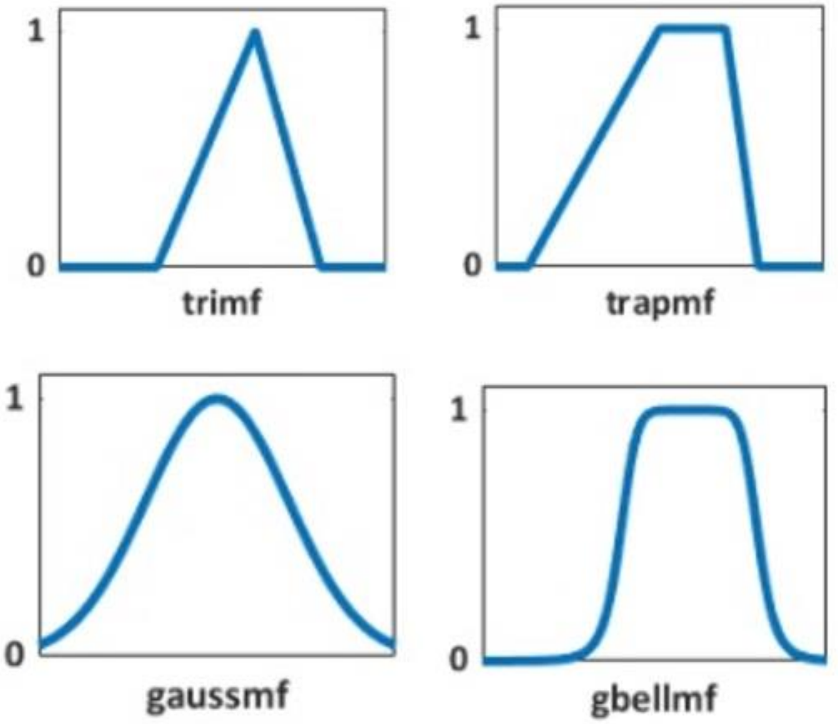


Figure 3.6: Commonly used membership functions

Triangle Membership Function

A triangular membership curve is functions of a vector defined by three parameters b_1, b_2, and b_3. With the Triangle membership function, the membership degrees are calculated by Equation 3.1.

$$\mu_{\boldsymbol{B}}(\boldsymbol{x}; \boldsymbol{b_1}, \boldsymbol{b_2}, \boldsymbol{b_3}) = \begin{cases} \boldsymbol{b_1 \leq x \leq b_2} & \boldsymbol{if} \quad \boldsymbol{(x - b_1)/(b_2 - b_1)} \\ \boldsymbol{b_2 \leq x \leq b_3} & \boldsymbol{if} \quad \boldsymbol{(b_3 - x)/(b_3 - b_2)} \\ \boldsymbol{x > b_3 \; or \; x < b_1} & \boldsymbol{if} \quad \boldsymbol{0} \end{cases} \tag{3.1}$$

Trapezoidal Membership Function

A trapezoidal membership curve is functions of a vector defined by four parameters b_1, b_2, b_3 and b_4. In fact, the triangular membership function is a special case of the trapezoidal membership function (Ubando et al., 2017). With the

Trapezoidal membership function, the membership degrees are calculated by Equation 3.2.

$$\mu_B(x; b_1, b_2, b_3) = \begin{cases} b_1 \leq x \leq b_2 & if \quad (x - b_1)/(b_2 - b_1) \\ b_2 \leq x \leq b_3 & if \quad 1 \\ b_3 \leq x \leq b_4 & if \quad (b_4 - x)/(b_4 - b_3) \\ x > b_4 \; or \; x < b_1 & if \quad 0 \end{cases}$$

(3.2)

In terms of computational efficiency of formulas, both triangular and trapezoidal membership functions are frequently used in various fuzzy logic applications (Aliev et al., 2018).

Gaussian Membership Function

This type of membership function is defined by the parameters c and σ. With the Gaussian membership function, the membership degrees are calculated by Equation 3.3.

$$\mu_B(x; c, \sigma) = exp\left\{\frac{-(x-c)^2)}{2\sigma^2}\right\} \tag{3.3}$$

In this function, c is the center of the function and σ is the width of the function. By changing the value of σ, we can change the form of the function. If σ is small, the membership function will be thinner, while the larger this value, the wider the membership function will be.

Bell-Shaped Membership Function

This type of membership function is defined with three parameters as b_1, b_2, b_3. With the Bell-Shaped membership function, the membership degrees are calculated by Equation 3.4.

$$\mu_A(x; b_1, b_2, b_3) = \left\{\frac{1}{1+\left|\frac{x-b_3}{b_1}\right|^{b_2}}\right\} \tag{3.4}$$

3.2.1.1 Mamdani Fuzzy Inference System

Mamdani fuzzy inference system, which is one of the frequently used fuzzy logic systems, was used for the first time for the control of the steam engine, and it is the system in which maximum minimum operations are used. İbrahim Mamdani (Mamdani et al., 1977) developed the Mamdani method based on the principles of fuzzy logic. It has become one of the most widely used fuzzy inference methods, thanks to its easy design, its compatibility with human behavior and its interpretable structure.

The working principle of the Mamdani fuzzy inference system is given in the Figure 3.7. In the Mamdani fuzzy inference system, membership functions are determined for all inputs and outputs of the system by using expert knowledge or sample data in the fuzzification process. All inputs and outputs of the system are blurred according to the predetermined number of clusters. After fuzzification, fuzzy rules are defined again based on expert knowledge or sample data or database. By determining the power of the fuzzy rules, the outputs of the rules are combined. Thus, blurry output is obtained, and clear output is obtained using the selected clarifier.

5 steps are followed when creating a Mamdani type fuzzy logic model.

a) Determination of membership degrees between 0 and 1 for input variables using fuzzy expressions
b) Setting the rules
c) Application And, or
d) Combining fuzzy sets representing the outputs
e) Defuzzification of fuzzy set results

Figure 3.7 shows how the C_i fuzzy set functions are calculated in a Mamdani type fuzzy inference system with two rules containing numerical variables such as x or y. Equations 3.5 and 3.6 list the two rules used in the calculation of this system.

Rule 1 : if $x = A_1$ and $y = B_1$ then $z = C_1$ (3.5)

Rule 2 : if $x = A_2$ and $y = B_2$ then $z = C_2$ (3.6)

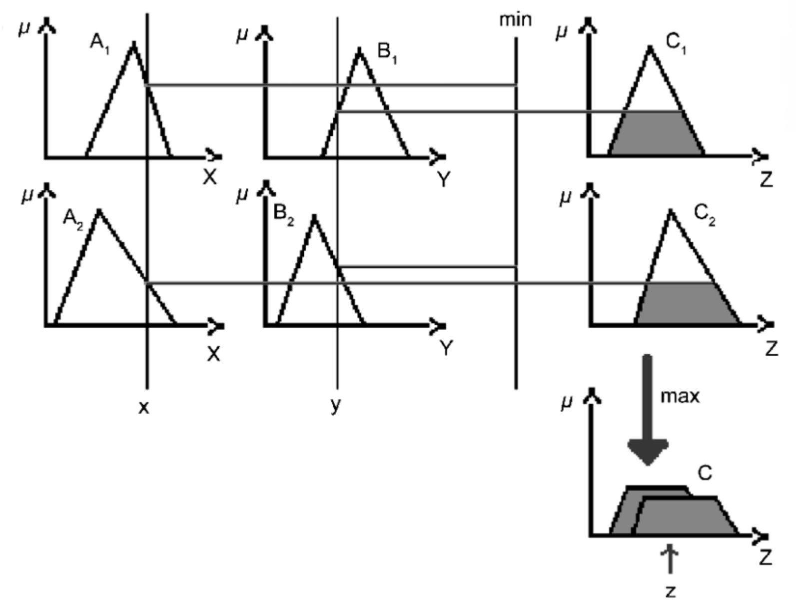


Figure 3.7: Mamdani type fuzzy inference system (Akyılmaz, 2005)

3.2.1.2 Sugeno (Takagi-Sugeno-Kang) Fuzzy Inference System

The Sugeno fuzzy model, proposed by Tagaki-Sugeno-Kang in 1985, is an adaptation of the Mamdani type model and was created by changing the structure of the rules in the Mamdani fuzzy inference system (Sivanandam et al., 2007). In the Sugeno fuzzy inference system, the antecedent part of the rules is still based on fuzzy sets (Zschorn, 2006). Therefore, it is understood that the input of each system should be blurred. The difference of this model compared to the Mamdani model is that the values of the output variable are not fuzzy. In Sugeno type fuzzy modelling, the output membership functions can take a fixed value or a server value of a linear function (Fırat, 2007). This model is called a zero-order model when the output membership functions are constant, and a first-order Sugeno fuzzy model when it is in the form of a first-order line equation. Accordingly, the writing of the rules also differs. Figure 3.8 shows the fuzzy inference mechanism of the first-order Sugeno model. In the system with P inputs and one output, the Sugeno type rules are as follows;

P^i = if Input$_1$ = x and Input$_2$=y then z = c (3.7)

or

P^i = if Input$_1$ = x and Input$_2$=y then z = ax + by+ c (3.8)

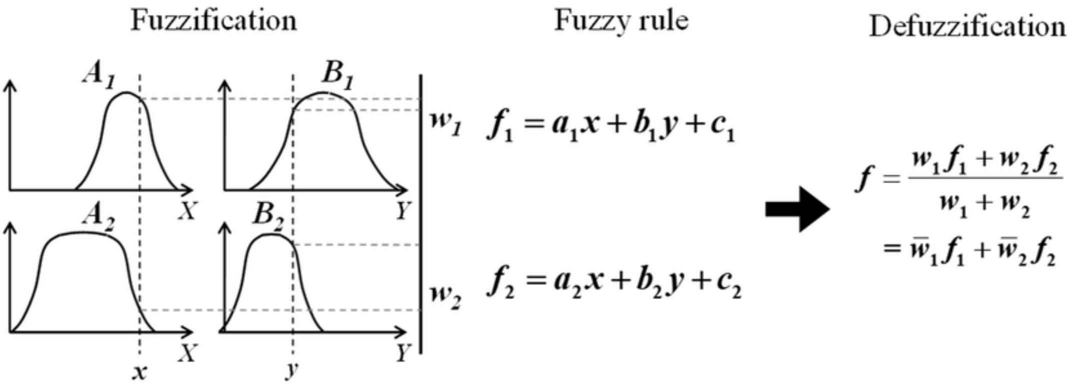


Figure 3.8: Sugeno type fuzzy inference system (Cho et al., 2020)

3.2.2 Adaptive Neuro-Fuzzy Inference System (ANFIS)

ANFIS consists of the representation of the Sugeno type fuzzy system as a network structure with neural learning capability (Özçalık et al., 2003). Both artificial neural networks and fuzzy logic are used in its structure (Avcı et al., 2002). The combination of neural network and fuzzy system is called fuzzy neural network. ANFIS is a fuzzy Sugeno model that uses adaptive systems to facilitate learning and adaptation processes. Such an approach makes fuzzy logic more systematic and less dependent on experience (Özgan et al, 2009). The main purpose of ANFIS is to optimize the parameters of the equivalent fuzzy logic system by means of a learning algorithm using input-output datasets. Parameter optimization is done in such a way that the error value between the actual output and the target output is minimum (Güney et al., 2008).

ANFIS consists of If-Then rules and input-output pairs of fuzzy inference system. However, ANN learning algorithms are used in system training and control to define the input and output relationship (Jang, 1993), (Franklin et al., 1990). If x and y are taken as input and z as output, the basic rule structure can be written as follows:

$$\text{If } x \in = A_i \text{ and } y \in = B_i \text{ Then } z_i = p_ix + q_iy + r_i \qquad (3.9)$$

Here, A_i and B_i are the labels of the sets that divide the variable space x and y in the antecedent part into fuzzy subspaces, respectively. p_i, q_i and r_i are the design parameters determined during the training process. z_i is the output value of that rule and is a function of the input variables. The resulting output value for any x, y input pair is the weighted average of the output value z_i of all rules (Özgan et al, 2009).

Based on the two fuzzy rules given below, the possible ANFIS architecture for the first order fuzzy Sugeno model is given in Figure 3.9. If we summarize the layers of the ANFIS model in general, the fuzzification is done by applying membership functions to the input data in the 1st layer. In the second layer, rules are created according to the fuzzy logic inference system. In the 3rd layer, a weighted average and normalization process is applied to each node coming from the rule layer. In the 4th layer, the fuzzy results are converted to numerical values, and finally, in the 5th layer, the output values of all nodes are collected to produce the single output value of the system (Cihan et al., 2021).

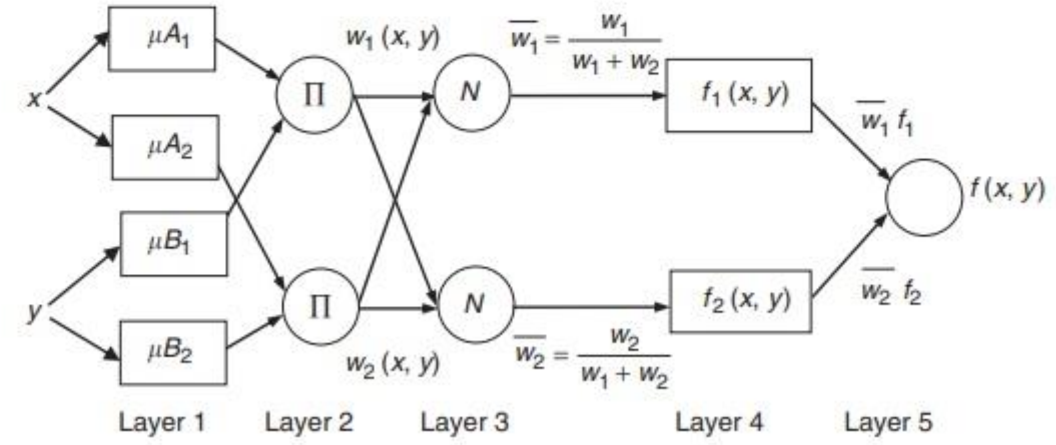


Figure 3.9: The scheme of the adaptive neural–fuzzy inference system

Rule 1 : If $x = A_1$ and $y = B_1$ then $f_1 = p_1 \times x + q_1 \times y + r_1$ (3.10)

Rule 2 : If $x = A_2$ and $y = B_2$ then $f_2 = p_2 \times x + q_2 \times y + r_2$ (3.11)

In these rules p_i, q_i and r_i are equation constants for each rule (Güney et al., 2008). As can be seen from the Figure 3.9, ANFIS has a 5-layer feed-forward neural network architecture. The main task of the learning algorithm for this architecture is to set all adaptive parameters to simulate the ANFIS output and the learning data (Haznedar et al., 2018). The training dataset is introduced to the neural network and the network is trained with the help of any training algorithm. It is aimed to determine the conditions where the error function between the model output and the learning data is minimum (Hímer et al., 2004).

The node functions of each layer in the ANFIS structure and the operation of the layers are as follows, respectively.

Layer 1

It is called the fuzzification layer. ANFIS model was used to separate the input values into fuzzy sets (Güneşer, 2019). The cells in this layer are adaptive cells and the number of cells is equal to the number of input variables. In this layer, the output of each node consists of membership degrees that depend on the input values and the membership function used. The outputs of the cells in this layer are given in O_{1i} Equation 3.12 and Equation 3.13 (Jang et al, 1995)

$$O_{1i} = \mu A_i (x) \qquad i=1,2 \qquad (3.12)$$

$$O_{1i} = \mu B_{i-2} (x) \qquad i=3,4 \qquad (3.13)$$

Here, A_i and B_i are any fuzzy set parameters, μA_i and μB_i are membership degrees for these set parameters (Haznedar et al., 2018). When the Gaussian membership function is used, μA_i is determined by Equation 3.14.

$$\mu A_i = \exp\left[-\left(\frac{x-c_i}{a_i}\right)^2\right] \qquad i=1,2 \qquad (3.14)$$

a_i is the sigma of the gaussian membership function, and c_i is its center. Parameters in this layer are called initial parameters.

Layer 2

It is the rule layer. Each node in this layer represents the number and rules created according to the Sugeno fuzzy logic inference system. The cells in this layer are fixed and the number of cells is equal to the number of rules (Haznedar et al., 2018). The cell inputs are the membership function values of the variables in the antecedent part of the rules, and the cell outputs (O_{2i}) give the weights (w_i) of the rules. The output of each rule node μ_i is determined by multiplying the membership degrees from the first layer as given in Equation 3.15.

$O_{2i} = w_i = \mu A_i(x).\mu B_i(y)$ i=1,2 (3.15)

Layer 3

It is the normalization layer. Each node in this layer accepts all nodes from the rule layer as input values and calculates the normalized ignition coefficient of each rule (Güneşer, 2019). The cells in this layer are also fixed cells, and their input is the degrees of weight they take from the previous layer. In this layer, the weight degrees are normalized in accordance with the expression given in Equation 3.16.

$$O_{3i} = \overline{w_i} = \frac{w_i}{\sum w_i}, \qquad i=1,2 \qquad (3.16)$$

Layer 4

It is the defuzzification layer. The weighted result values of a given rule are calculated at each node in the defuzzification layer. The cells in this layer are adaptive cells and the output of these cells is calculated as in Equation 3.17.

$$O_{4i} = \overline{w_i}.f_i = \overline{w_i}.(p_i x + q_i y + r_i) \qquad i=1,2 \qquad (3.17)$$

Layer 5

It is the total layer. There is only one node in this layer and is labeled with ∑. Here, the output value of each node in the 4th layer is summed, resulting in the actual value of the ANFIS system. The output value of the system is calculated according to the Equaiton 3.18.

$$O_{5i} = f = \sum \overline{w_i}.f_i = \frac{\sum w_i.f_i}{\sum w_i} \qquad i=1,2 \qquad (3.18)$$

3.3 ANFIS Model Identification Methods

In the implementation phase for problem solving, the creation and configuration of the ANFIS model is one of the main problems to be solved. All these processes, such as selecting the input variables, partitioning the input space, choosing the type and number of membership functions for the inputs, determining the number of rules by creating the rule layer, and determining the initial values of the membership functions parameters, include the creation of the ANFIS model (Tulun et al., 2021). Expert knowledge based on experience is generally used to construct the ANFIS model. However, in many cases, the knowledge of experts may be insufficient to determine the most appropriate criteria for the ANFIS model by making the right choices. The process of selecting the most suitable features by making different experiments takes time, but its suitability is also controversial. There are several methods that automate this process and significantly reduce the runtime while improving the performance and interpretability of the fuzzy model. Grid Partitioning (GP), Subtractive Clustering (SC) and Fuzzy C-Means Clustering (FCM) methods are some of these commonly used methods (Olatunji et al., 2022). The basic idea behind these methods is to predict fuzzy rules by learning the relationship between input-output sample data. Partitioning the input space effectively increases the speed and performance of the model both in the learning and implementation phases, by reducing the number of rules and membership functions of ANFIS in general. In the following sections, information about Fuzzy C-Means Clustering and Grid Partitioning are given.

3.3.1 Grid Partitioning (GP)

The most accurate way to construct the initial parameters of the MFs is to divide the input field into equal parts for the number of clusters selected (Setia et al., 2006). Each section presents a fuzzy if-then rule. The parameters of the membership functions are adjusted so that the centers of the membership functions are placed evenly across each data size range. After obtaining the membership functions for each input, the combinations of membership functions of all inputs are enumerated to form the rule set. The Equations 3.19 and 3.20 used for the calculation of the given c membership functions (clusters), cluster centers (v_i) and width values (σ_i) are given below.

$$v_i = U_{\min} + (i - 1)\frac{U_{\max} - U_{\min}}{c-1} \qquad 1 \leq i \leq c \qquad (3.19)$$

$$\sigma_i = \frac{U_{\max} - U_{\min}}{2(c-1)} \qquad 1 \leq i \leq c \qquad (3.20)$$

Here $U_{\min}$ and $U_{\max}$ are the maximum and minimum values of the data size.

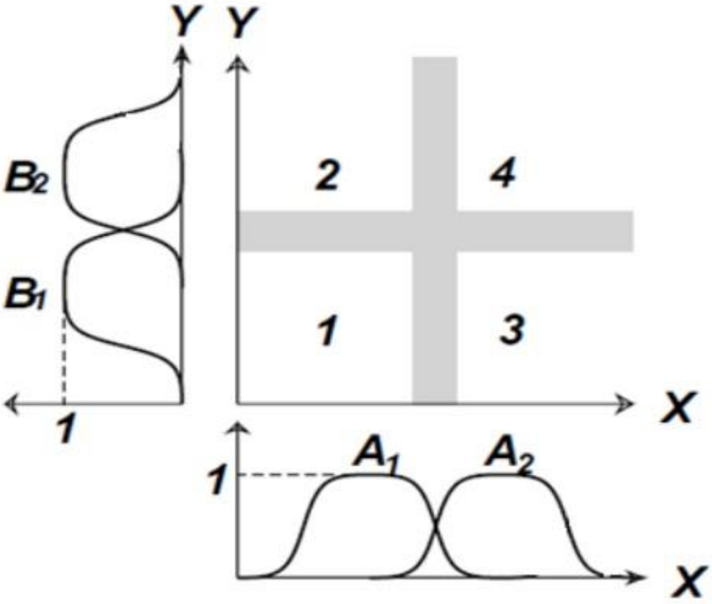


Figure 3.10: Grid partitioning model with 2 input variables (Sadouki et al., 2016)

At the very beginning of the training phase, which is a learning process of the model aimed to be developed, the grid partition method divides the available data set into rectangular sub-areas called grids according to the predetermined number and types of membership functions to be used (Buyukyildiz et al., 2017). For example, in an ANFIS model with two inputs and 2 membership functions applied to each input, a total of 2^2=4 If-Then fuzzy rules are formed. In Figure 3.10, as an example, partitioning into sub-domains is shown in the case where 2 input variables and 2 membership functions are defined for each input variable.

3.3.2 Fuzzy C-Means Clustering (FCM)

FCM algorithm is one of the best methods of fuzzy-based clustering methods and is frequently used. The FCM method is an iterative-based clustering algorithm that ensures that each data point belongs to a cluster according to the specified membership degree (Guzel et al., 2020). According to the fuzzy logic principle, each data belongs to each of the clusters with a membership value varying between [0,1]. The sum of the membership values of a data to all classes must be "1". Whichever cluster center the object is close to, the membership of that cluster will be larger than the membership of other clusters. This method was first introduced by Dunn in 1973 and developed by Bezdek in 1981 as an enhancement to previous clustering methods (Bezdek, 1981).

FCM is a supervised clustering method based on minimizing the following objective function.

$$J\ (U, V) = \sum_{i=1}^{n} \sum_{j=1}^{c} (\mu_{ij})^{m} (d_{ij})^{2} \quad (3.21)$$

Here:

$X = x_1, x_2, \ldots, x_n$: array of n data points,

$V = v_1, v_2, \ldots, v_c$: elements of cluster centers,

$\mu_{ij} \varepsilon [0,1]$: cluster center v_j with x_i degree of membership of the data point,

d_{ij}: The Euclidean distance between x_i and v_j, $d_{ij} = \| x_i - v_j \|$.

m: the fuzziness index is used to control the fuzziness of the membership degrees of each data point.

$U = (m_{ij})_n xc$: fuzzy partition matrix covers all membership degrees of all cluster centers from each data.

Fuzzy partitioning is done by updating membership degree μ_{ij} and cluster centers v_j to optimize the J function given below.

$$v_j = \frac{\sum_{i=1}^{n} (\mu_{ij})^{m} x_i}{\sum_{i=1}^{n} (\mu_{ij})^{m}} \quad (3.22)$$

$$\mu_{ij} = \frac{1}{\sum_{k=1}^{c} \frac{(d_{ij})}{d_{jk}}^{2/(m-1)}} \quad (3.23)$$

Using the two conditions given above, the algorithm runs iteratively until the stopping criteria are met. If the difference between the updated and previous objective function J is less than the predefined minimum threshold or the maximum number of iterations is reached, the stopping criteria are met (Raju et al., 2012). When using the FCM method, the user must also determine the number of clusters to be created. However, the cluster validity criterion is used to find the optimal number of clusters for the data set.

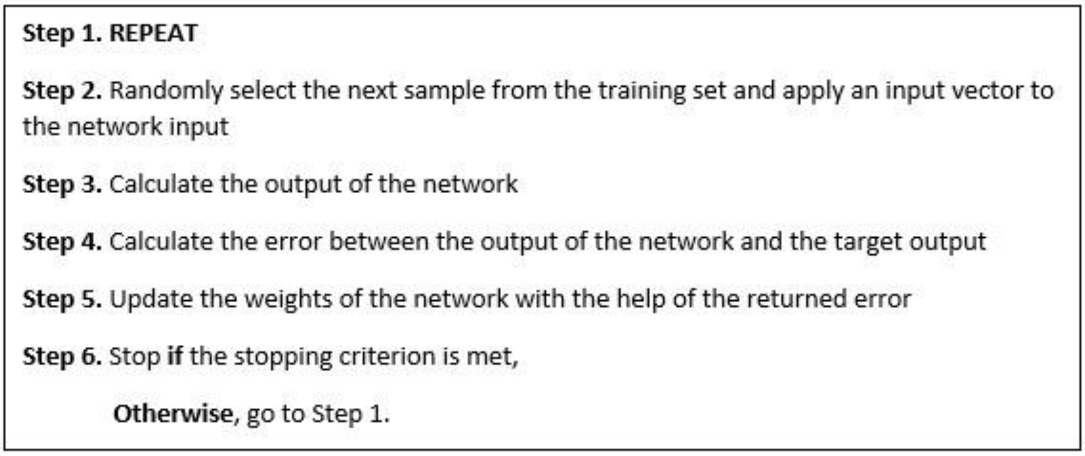
3.4. Training of ANFIS

ANFIS has two parameter types that need to be trained, initial and final parameters. Initial parameters belong to membership functions as $\{a_i, b_i, c_i\}$. Here, a_i is the variance of membership functions, c_i is the center of membership functions, and b_i is another training parameter. The total number of initial parameters is equal to the total number of parameters in all membership functions. The resulting parameters are the parameters of the output polynomial shown as {p_i, q_i, r_i} in Equation 3.17 and used in the defuzzification layer of the ANFIS network in Figure 3.9. Many methods have been proposed to update the initial and outcome parameters in the training process of ANFIS. ANFIS-BP, ANFIS-HB and ANFIS-SA models were tested to train the parameters of the ANFIS structure.

3.4.1 Back Propagation Algorithm (BP)

The BP learning model, first proposed by (Rumelhart et al., 1986) is one of the most widely used artificial neural network learning models. The discovery of the BP model has been one of the most important historical developments for artificial neural networks (Rumelhart et al., 1986). BP is one of the ANFIS parameter optimization options. Error back propagation networks are fully interconnected feed-forward networks consisting of many layers that basically use a supervised learning strategy that requires the desired output of the system to be presented to the network along with the input sample. In other words, every neuron in a layer connects to all neurons in the lower layer.

The BP learning model got this name because it tries to reduce the error backwards from the output according to the states of the values received from the output and the values expected to be estimated. It performs the parameter update process by calculating the slope drop information with the error value obtained from the existing data set. The basic operation steps of a BP algorithm are given in Figure 3.11.



Step 1. REPEAT

Step 2. Randomly select the next sample from the training set and apply an input vector to the network input

Step 3. Calculate the output of the network

Step 4. Calculate the error between the output of the network and the target output

Step 5. Update the weights of the network with the help of the returned error

Step 6. Stop **if** the stopping criterion is met,

Otherwise, go to Step 1.

Figure 3.11: Basic steps of back propagation algorithm

The error calculated at the network output in the learning rule is used to calculate the new values of the weights (Haznedar et al., 2018). The update formula for the input parameters in the reverse direction transition is as follows.

$$\Delta\alpha = -\eta \frac{\partial E}{\partial \alpha} \qquad (3.24)$$

Here α is any input parameter, η is the learning rate, and E is the error value at the output of the network (Öztürk, 2011).

3.4.2 Hybrid Learning Algorithm (HB)

In the hybrid learning algorithm used in ANFIS, there is a bidirectional processing process in each iteration, forward and backward (Akyılmaz et al., 2010). In the first step of the HB algorithm, the antecedent parameters are kept constant and the signals from the input data are advanced by calculating the output of a node until the matrices A and B in Equation 3.25 are obtained (Hussain et al., 2015). In the HB algorithm, the parameters of the ANFIS network structure are considered in two parts as initial and result parameters. Total parameter set; Expressed as $S= S_1+ S_2$, S_1 corresponds to the initial parameters and S_2 corresponds to the result parameters. The result parameters, S_2, are calculated by Equation 3.28. The functional signals then continue in the forward direction until the error criterion is calculated. In the reverse direction, the error rates are transferred from the output side to the input side in equations 3.26 and 3.28, and the initial parameters S_1 are updated by the slope reduction method given in Equation 3.24. The process flow of the hybrid algorithm is summarized in Table 3.1.

Table 3.1: Two-pass learning process of hybrid learning algorithm

	Forward Passing	**Backward Passing**
Initial parameters	Stable	Gradient descent
Result parameters	Least squares estimation	Stable
Signals	Node outputs	Error rates

$$A\theta = B \tag{3.25}$$

The vector θ in the matrix equation is the unknown vector consisting of the elements of the S_2 output parameters. This equation represents the standard linear least-squares problem, and the best solution for θ is $\| A\theta - B \|^2$ the minimum value of the equation is the least squares estimation (LSE), θ^* (Padmini et al., 2008).

$$\theta^* = (A^T A)^{-1} A^T B \tag{3.26}$$

Where A^T is the transpose of matrix A and if A^TA is non-singular, then $(A^TA)^{-1} A^T$ is the pseudo-inverse of matrix A. Alternatively, the LSE formula can be used iteratively and θ vector can be calculated.

$$\theta_{i+1} = \theta_i + S_{i+1}\, a_{i+1}\, (\boldsymbol{b}_{i+1}^{T} - \boldsymbol{a}_{i+1}^{T}\, \theta_{i}) \tag{3.27}$$

$$S_{i+1} = S_i - \frac{S_i\, a_{i+1} a_{i+1}^T S_i}{1 + a_{i+1}^T\, S_i\, a_{i+1}} \qquad i = 0,1,\ldots, \mathrm{P}-1 \tag{3.28}$$

The initial values for Equation 3.27 are $\theta_0 = 0$ and $S_0 = \gamma$ I, γ being a large positive number and I is the unit matrix of size MxM (Wang et al.,2007).

3.4.3 Simulated Annealing (SA)

SA is a heuristic optimization technique designed for the global optimum approximation of a given cost function, first proposed by Kirkpatrick et al (Kirkpatrick et al., 1983). The SA algorithm is based on heating solids and then slowly cooling them (Kirkpatrick et al., 1983). It is designed to find the smallest values of nonlinear functions with many local minimums when slowly cooling down (Asrul et al., 2020).

At a given temperature T, the probability distribution of the system energies is determined by the thermodynamic law in Equation 3.29.

$$\mathrm{P(E)} = \boldsymbol{e}^{-\boldsymbol{E}/(\boldsymbol{kT})} \tag{3.29}$$

Here E is the system energy and k is the Boltzmann constant.

The initial value of T will gradually decrease with each iteration. According to the rules of SA, the primary solution is completely random (Jafari et al., 2020). Therefore, the performance of SA is highly dependent on its initial value. If the quality of the initial value is poor, the result will be unsatisfactory (İlhan, 2020). Therefore, it is necessary to start the search with a sufficiently high temperature value. According to Equation 3.29, P(E) converges to 1 for all energy states at high temperatures. Even at low temperatures, it is unlikely that the system will have a high energy level. Therefore, the statistical distribution of energies allows the system to exit a local energy minimum. In the analogy between the combinatorial optimization problem and the annealing process, the states of the solid represent possible solutions to the optimization problem, and the energies of these states correspond to the objective function values calculated for the solutions (Hajipour et al., 2016). The minimum energy state represents the optimal solution to the problem, while the rapid cooling process can be seen as the local optimum.

3.5. Assessment Methods

The most important step in any machine learning model is to evaluate the accuracy of the related models to describe how well the model is performing in its predictions. MAE, RMSE, R^2 and MAPE evaluation criteria are used to evaluate the performance of the model in statistical analysis. Among these performance measures, R^2 is the accuracy rate decision coefficient of the model. A high value of this coefficient indicates a good estimation relationship (Atik, 2022). Since MSE, RMSE and MAE are error measures, low results are measures that show high performance inversely proportional to performance (Wang et al., 2004).

3.5.1 Mean Absolute Error (MAE)

The measure of difference between two continuous variables is called MAE. MAE is the average vertical distance between each true value and the line that best fits the data. This value is also the average horizontal distance between all data points and the best-fit line (Salmasi, 2021). It is a "MEAN" number divided by the sum of all data points in the data set divided by the total data point. For statistics, the forecast error is the difference between the true value of a time series and the predicted or predicted value. Since MAE value is an easily interpreted value, it is frequently used in regression and time series problems. MAE value takes a value between 0 and ∞. The lower its value, the better it performs (Celestin, 2020).

MAE is determined in Equation 3.30.

$$\text{MAE} = \frac{1}{n}\sum_{t=1}^{n}|X_i - Y_i| \tag{3.30}$$

Where X_i is the sum of the actual data, Y_i is the sum of the predicted data, and n is the total number of data points.

3.5.2 Mean Absolute Percentage Error (MAPE)

MAPE is a statistical measure of how accurate a forecasting system is. MAPE statistic eliminates the disadvantages that may arise when comparing models with different unit values (Khatib et al., 2011). MAPE is widely used because it is easy to interpret and explain. Another feature of MAPE is that it is a measure of error, so low results are measures of high performance. MAPE is often used in model evaluation as a loss function for regression and time series problems.

MAPE is determined in Equation 3.31.

$$\text{MAPE} = \frac{1}{n}\sum_{t=1}^{n}\left|\frac{X_i - Y_i}{X_i}\right| \times 100 \tag{3.31}$$

3.5.3 Root Mean Square Error (RMSE)

RMSE is a quadratic metric that measures the magnitude of error of a machine learning model, which is often used to find the distance between the predicted values of the estimator and the true values (Karahoca et al., 2013). RMSE is the standard deviation of the estimation errors (residues). That is, residuals are a measure of how far the regression line is from the data points; RMSE is a measure of how widespread these residues are. In other words, it tells you how dense the data is around the line that best fits the data. RMSE value can range from 0 to ∞.

RMSE is determined in Equation 3.32.

$$\text{RMSE} = \sqrt[2]{\frac{1}{n}\sum_{t=1}^{n}(X_i - Y_i)^2} \tag{3.32}$$

3.5.4 Coefficient of Determination (R^2)

In predictive analysis, the coefficient of determination method is used to predict and evaluate the future results of a model. This method serves as a guide to help measure the accuracy of the model (Gkountakou et al., 2020). The coefficient of determination method shows the level of variation in the given data set. R^2 is the model's accuracy rate decision coefficient. R^2 value can range from 0 to 1. The prediction success of the model increases as it gets closer to 1.

R^2 is determined in Equation 3.33.

$$R^2 = \mathbf{1 - \frac{\mathit{RSS}}{\mathit{TSS}}} \tag{3.33}$$

Where RSS is the residuals sum of squares, TSS is total sum of squares.

CHAPTER 4

RESULTS AND DISCUSSION

In order to compare the river flow performance with different methods, the ANFIS network is trained with BP, HB and SA algorithms to classify the problem under consideration (Haznedar et al., 2021). For this purpose, the learning coefficient for the BP algorithm used was 0.2 and the momentum ratio was chosen as 0.4. As a hybrid algorithm, a method consisting of the least squares estimation and the combination of the BP algorithm was used. In addition, the number of iterations for these algorithms was determined as 1000.

To demonstrate the accuracy and reliability of the results, 27 FCM and 48 GP simulation studies were conducted with each algorithm. In this study, many membership function numbers, gaussmf, gbellmf, trimf and trapmf membership function types and output type constant/linear have been tested one by one. In addition, the $RMSE_{AVG}$, $\mathrm{MAE}_{\mathrm{AVG}}$, $\mathrm{MAPE}_{\mathrm{AVG}}$ and $\mathrm{R}^2{}_{\mathrm{AVG}}$ values were found by taking the average of the MAE, MAPE, RMSE and R^2 values obtained for each model.

The training data learns the relationship between the network input-output data and adjusts the membership function parameters until it reaches the specified number of revolutions or the desired error level (Tulun et al., 2021). While the training error decreases during the training phase, the error of the control set may suddenly increase at a certain point. This point is called the “overfitting point”. The training process continues until the training error value is fixed. At one of these points, the training is terminated and the testing phase begins. The network is then tested with a test dataset that it has never seen before. It is seen with how much error the network predicts the test data. If this error rate is within acceptable limits, it means that the desired network structure has been reached. However, if the error rate is large, the parameters are changed by returning to the training phase of the network. When the desired error rate is reached, the prediction success of the model created is evaluated.

7141 daily data, which is 80% percent of the flow data of the station between 01.01.1988 and 11.03.2007, were used in the training phase of the models, and 1785 daily data, the 20% percent between 12.03.2007-31.12.2011, was used in the testing phase. While estimating the current flow value Q(t), 3 models were established. In the first model, Q(t) is estimated based on the current value Q(t–3) of the previous three days. In the second model, Q(t) is estimated based on the flow value Q(t–4) 4 days ago and the flow value Q(t–5) 5 days ago. It is set up similarly in other models. ANFIS model of all algorithms to predict river flow was constructed from 3 input variables (lag-3, lag-4 and lag-5). Based on the previous steps, we can conclude that Q(t–3) as lag-3, Q(t–4) as lag-4, and Q(t–5) as lag-5 are considered ANFIS inputs. According to the 5-time delay scenario, daily flow values starting from 5 days before and up to one day before are used as input in the 6th Day forecast. For the data set of the 4 time-shifted scenarios, the daily flow values starting from 4 days before and up to the day before are used as input in the 5th Day forecast. In the data set of the 3 shift scenarios, the 4th day flow values were estimated by using the flow values from 3 days before to 1 day ago. These lag values are given as input values to the 4 membership functions in GP clustering. Then the optimal inputs will be evaluated to determine the optimal number of membership functions. In the FCM clustering method, the parameters are planned by dividing into clusters between 2 and 10 for optimal results. Finally, all the findings obtained as a result of the simulation results are given in detail in the Tables 4.1-4.18. The most successful methods of the simulation models presented in the Table 4.1-4.18, corresponding training and test performances graphs are shown in the Figures 4.1-4.36. In the next part of the study, the results obtained from the models were compared at all lag levels. Tables 4.19, 4.20, and 4.21 categorize the error measures associated with each model according to their clustering method. The data in the tables consist of 3 different lag scenarios by taking the average of the test values of the model results obtained in two different clustering methods. Finally, in Tables 4.22, 4.23 and 4.24, all models were compared by taking the average results of the two clustering methods. In this way, the performances of the models were measured separately based on all methods.

Table 4.1: Training and Test Performances of 1825 Station's using ANFIS-BP Model with FCM Method

Models		Data	Scenario											
			LAG 3				LAG 4				LAG 5			
			RMSE	**MAE**	**MAPE**	**R²**	**RMSE**	**MAE**	**MAPE**	**R²**	**RMSE**	**MAE**	**MAPE**	**R²**
ANFIS-BP	FCM-2	Train	6.3716	1.8943	28.1113	0.6640	6.3821	1.8621	26.3997	0.6629	6.3678	1.8110	25.0717	0.6645
		Test	5.0136	1.7116	38.0975	0.6664	5.0117	1.6739	35.9331	0.6625	4.9883	1.6172	33.2281	0.6693
	FCM-3	Train	6.3098	1.6804	16.7042	0.6705	6.2902	1.7014	17.6479	0.6725	6.0692	1.7198	17.8711	0.6952
		Test	4.8687	1.3473	19.3159	0.6864	4.9053	1.3948	20.6882	0.6819	5.0508	1.4427	20.4824	0.6665
	FCM-4	Train	6.2536	1.6583	16.1305	0.6763	6.2530	1.6721	16.8268	0.6764	6.2418	1.6801	17.1816	0.6776
		Test	**4.7950**	1.3434	18.8686	0.6956	**4.8085**	1.3736	20.0422	0.6941	**4.8425**	1.3953	20.5538	0.6898
	FCM-5	Train	6.0519	1.6188	15.4438	0.6969	6.0702	1.6416	16.1031	0.6951	6.0698	1.6527	16.4166	0.6951
		Test	4.8665	1.3450	18.1172	0.6885	4.8776	1.3688	19.0475	0.6874	4.8820	1.3843	19.6457	0.6870
	FCM-6	Train	6.0356	1.6122	14.8903	0.6985	5.9782	1.5970	15.4669	0.7042	5.8824	1.5952	15.6926	0.7137
		Test	4.9159	**1.3360**	17.6765	0.6842	4.9353	1.3787	18.7652	0.6795	5.0768	1.4313	19.1037	0.6661
	FCM-7	Train	5.9639	1.5504	14.4863	0.7056	5.9404	1.5810	14.9424	0.7080	5.8381	1.5850	15.2461	0.7180
		Test	4.9097	1.3384	17.8591	0.6826	5.0590	1.3852	17.9546	0.6670	5.0579	1.3942	18.2902	0.6657
	FCM-8	Train	5.7821	1.5651	14.3385	**0.7233**	5.9255	1.5872	14.6680	0.7094	5.8093	1.5809	14.7989	**0.7208**
		Test	5.0766	1.3747	17.7585	0.6645	5.0662	1.3894	17.8135	0.6664	5.1062	1.3951	17.9336	0.6601
	FCM-9	Train	5.7705	1.5625	14.0014	0.7244	5.8493	1.5643	14.0830	0.7168	5.9611	1.5832	14.2734	0.7059
		Test	5.0675	1.3698	17.3362	0.6651	4.9939	**1.3521**	17.2850	0.6743	4.9202	**1.3439**	17.3943	0.6839
	FCM-10	Train	5.9688	1.5642	**13.7082**	0.7051	5.8136	1.5703	**14.0288**	**0.7203**	5.8454	1.5681	**14.1137**	0.7173
		Test	4.9075	1.3416	16.9264	0.6837	5.1417	1.3755	17.3852	0.6588	5.0968	1.3956	17.1425	0.6662

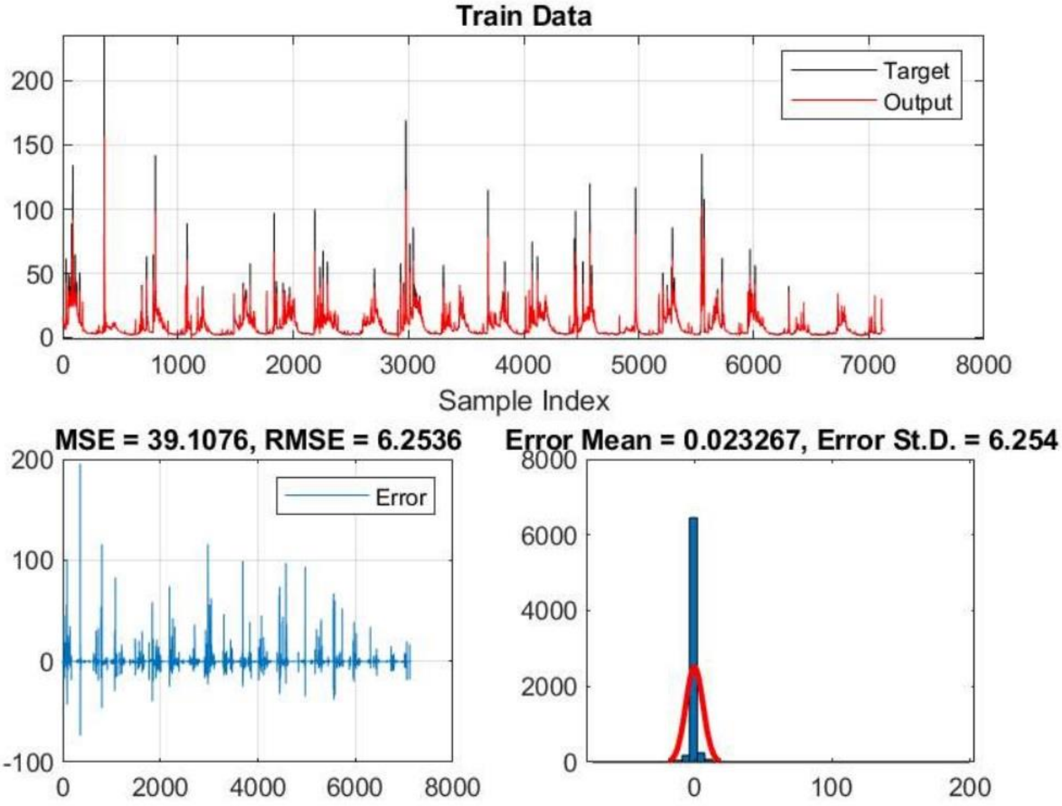


Figure 4.1: Training Performances of 1825 Station's using ANFIS-BP Model with FCM-4 Method (Lag 3)

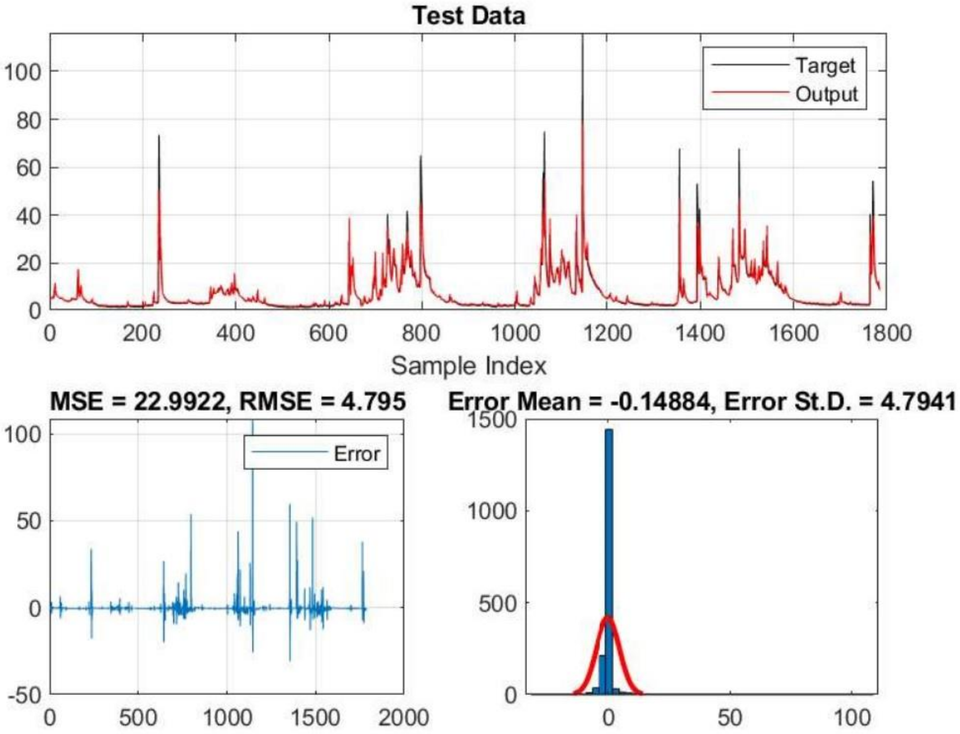


Figure 4.2: Test Performances of 1825 Station's using ANFIS-BP Model with FCM-4 Method (Lag 3)

Table 4.2: Training and Test Performances of 1825 Station’s using ANFIS-HB Model with FCM Method

			Scenario											
Models		Data	LAG 3				LAG 4				LAG 5			
			RMSE	**MAE**	**MAPE**	**R^2**	**RMSE**	**MAE**	**MAPE**	**R^2**	**RMSE**	**MAE**	**MAPE**	**R^2**
ANFIS-HB	FCM-2	Train	6.0475	1.6045	14.5087	0.6973	6.0574	1.6105	14.8443	0.6963	6.1406	1.6690	16.7703	0.6880
		Test	5.0260	1.3974	18.4634	0.6759	5.0300	1.4066	19.2155	0.6740	4.9543	1.4381	21.7712	0.6805
	FCM-3	Train	6.1309	1.6108	12.7545	0.6889	5.9933	1.6283	13.1695	0.7027	6.0102	1.6295	13.0415	0.7011
		Test	4.8848	1.3022	12.7608	0.6885	5.0466	1.3455	12.3609	0.6748	5.0371	1.3322	12.1558	0.6739
	FCM-4	Train	6.1033	1.5801	12.3760	0.6917	6.1433	1.6021	12.7828	0.6876	6.1539	1.6148	12.9582	0.6866
		Test	**4.8646**	**1.2793**	**11.4845**	0.6914	**4.8597**	**1.2930**	12.2079	0.6921	**4.8616**	1.2984	12.3192	0.6915
	FCM-5	Train	5.9283	1.5549	12.3145	0.7091	5.9574	1.5719	12.4833	0.7063	6.0031	1.5996	12.8932	0.7018
		Test	5.0034	1.3066	11.8329	0.6759	4.9489	1.2977	**11.5564**	0.6831	4.9315	1.3051	12.1109	0.6844
	FCM-6	Train	5.8874	1.5538	12.3984	0.7131	5.8639	1.5621	12.4925	0.7154	5.8590	1.5546	12.4720	0.7159
		Test	5.0959	1.3031	12.1766	0.6664	4.9989	1.3229	11.6825	0.6756	5.0341	1.3298	**11.5820**	0.6711
	FCM-7	Train	5.7986	1.5686	14.2083	0.7217	5.8480	1.5481	12.3553	0.7170	5.7787	1.5450	12.4922	0.7237
		Test	4.9931	1.3692	15.8734	0.6778	5.0817	1.3336	11.7747	0.6672	5.1593	1.3254	11.5940	0.6557
	FCM-8	Train	5.5918	1.5411	12.2849	0.7412	5.8071	1.6150	14.7449	0.7209	5.7068	1.5676	12.9111	0.7305
		Test	5.1280	1.3167	11.6415	0.6603	5.1007	1.4089	16.0335	0.6646	5.1873	1.3681	12.2406	0.6532
	FCM-9	Train	5.5049	1.5364	12.4723	0.7492	5.5512	1.6026	14.6237	0.7450	5.5442	1.5474	12.9045	**0.7450**
		Test	4.9687	1.2916	12.0679	0.6760	5.3587	1.4334	16.3863	0.6383	5.2750	1.3834	12.2057	0.6440
	FCM-10	Train	5.4802	1.5312	12.6801	**0.7514**	5.4924	1.5818	13.0160	**0.7503**	5.8635	1.5702	13.0260	0.7155
		Test	5.4418	1.3692	12.9465	0.6300	5.4539	1.3974	12.1138	0.6311	4.9861	**1.2944**	12.0222	0.6782

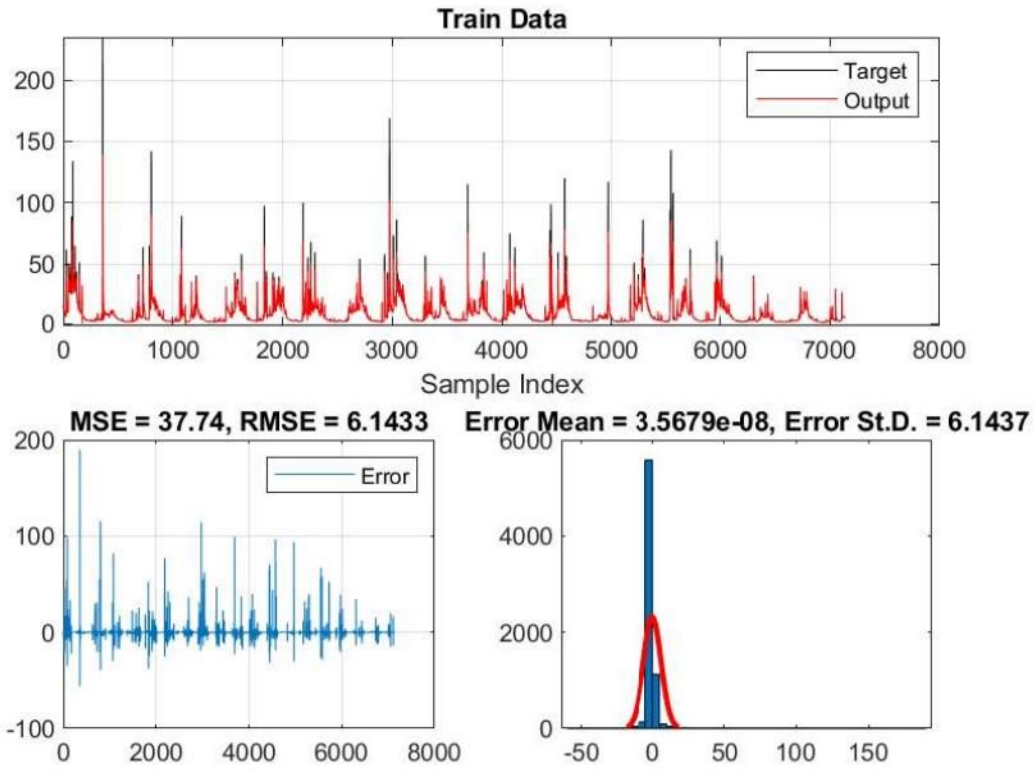


Figure 4.3: Training Performances of 1825 Station’s using ANFIS-HB Model with FCM-4 Method (Lag 3)

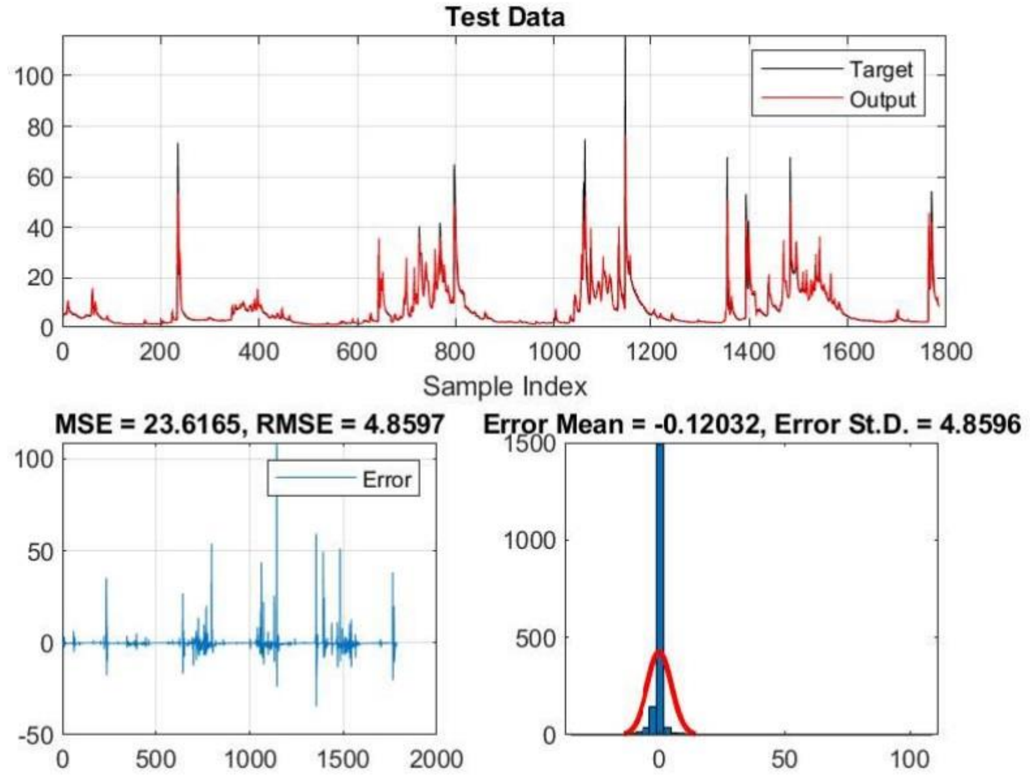


Figure 4.4: Test Performances of 1825 Station’s using ANFIS-HB Model with FCM-4 Method (Lag 3)

Table 2.3: Training and Test Performances of 1825 Station's using ANFIS-SA Model with FCM Method

			Scenario											
Models		Data	LAG 3				LAG 4				LAG 5			
			RMSE	**MAE**	**MAPE**	**R^2**	**RMSE**	**MAE**	**MAPE**	**R^2**	**RMSE**	**MAE**	**MAPE**	**R^2**
ANFIS-SA	FCM-2	Train	6.0436	1.5337	12.9531	0.6977	5.8983	1.6035	15.4283	0.7123	6.3576	1.8233	24.0833	0.6655
		Test	**4.8943**	**1.2972**	14.8357	0.6876	4.8946	1.3906	16.4967	0.6762	5.0883	1.6215	30.8833	0.6566
	FCM-3	Train	6.0708	1.7592	14.8667	0.6952	5.9629	1.5364	13.3713	0.7057	6.0044	1.6352	17.7324	0.7017
		Test	4.9868	1.4539	14.9740	0.6786	4.9693	1.3039	14.6121	0.6797	5.0882	1.4422	21.6349	0.6625
	FCM-4	Train	5.5555	1.5674	15.2865	0.7446	6.1018	**1.7293**	19.8504	0.6919	6.0915	1.6142	13.8615	0.6933
		Test	4.8969	1.3695	18.8203	0.6824	4.9993	1.4924	25.0031	0.6709	4.9217	1.3204	14.9074	0.6817
	FCM-5	Train	5.6364	1.5461	13.8796	0.7378	5.4262	1.5223	15.0415	0.7563	5.6678	1.6739	20.0197	0.7343
		Test	6.6074	1.4487	15.9413	0.5395	5.0113	1.3798	18.8906	0.6716	6.7663	1.6086	25.4107	0.5135
	FCM-6	Train	5.8894	1.5160	11.8838	0.7129	5.3668	1.4028	**12.1067**	0.7618	5.8958	1.5224	**13.4495**	0.7124
		Test	4.9627	1.2728	**11.6356**	0.6772	7.4917	1.4548	14.0704	0.4608	14.0629	1.6892	17.9368	0.2125
	FCM-7	Train	5.6790	1.5057	12.3034	0.7331	5.4802	1.4987	14.5747	0.7515	5.4084	1.5320	13.4847	0.7579
		Test	30.6227	2.1143	14.5313	0.0996	5.5306	1.3859	16.6253	0.6238	5.4411	1.3943	14.3378	0.6276
	FCM-8	Train	5.5464	1.6083	16.6841	0.7454	5.7347	1.4990	13.3217	0.7278	5.1882	1.5275	14.9722	**0.7774**
		Test	5.3813	1.4284	16.7525	0.6367	**4.9158**	1.3163	15.5868	0.6857	5.0520	1.3506	15.0341	0.6667
	FCM-9	Train	5.2138	1.4210	12.7262	**0.7750**	5.1068	1.4878	14.9636	**0.7842**	5.8533	1.5397	13.7163	0.7165
		Test	6.0613	1.4082	16.0209	0.5723	5.4265	1.4017	18.2522	0.6284	**4.8425**	**1.3192**	15.0064	0.6905
	FCM-10	Train	5.3284	1.5076	12.7835	0.7650	5.2846	1.4760	12.5254	0.7689	5.2386	1.5436	16.1801	0.7732
		Test	5.2008	1.3397	14.1499	0.6534	7.9038	1.5086	14.9108	0.4462	4.9746	1.4001	20.2734	0.6725

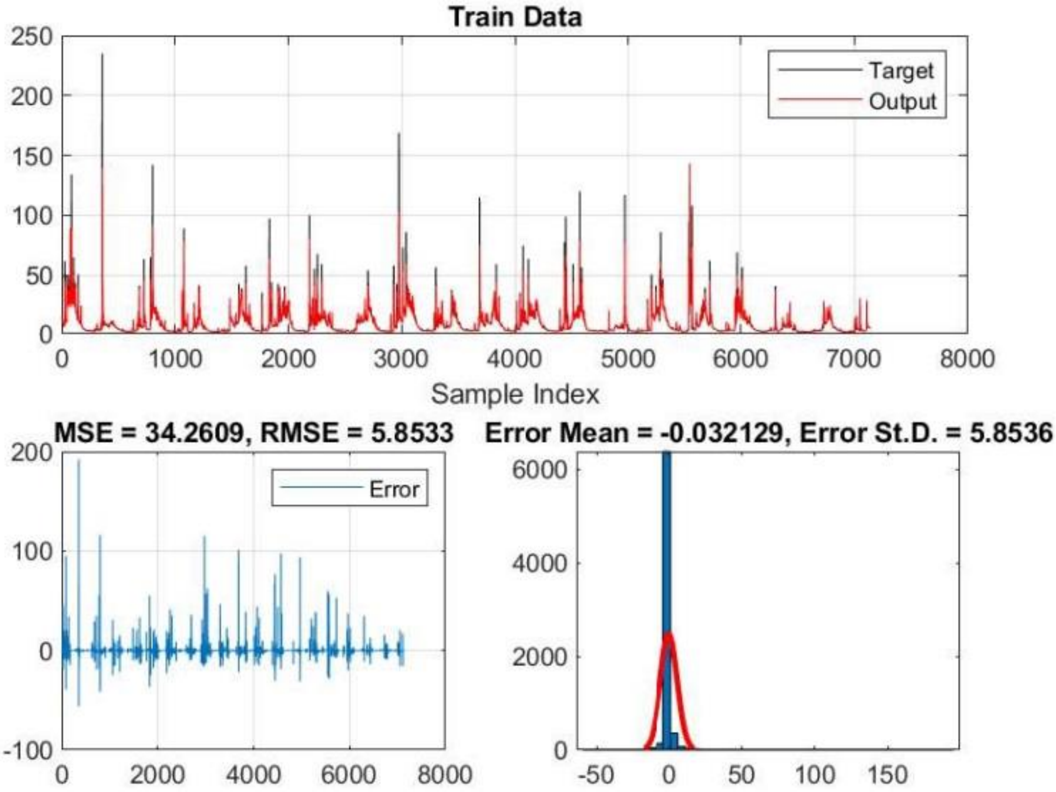


Figure 4.5: Training Performances of 1825 Station's using ANFIS-SA Model with FCM-9 Method (Lag 5)

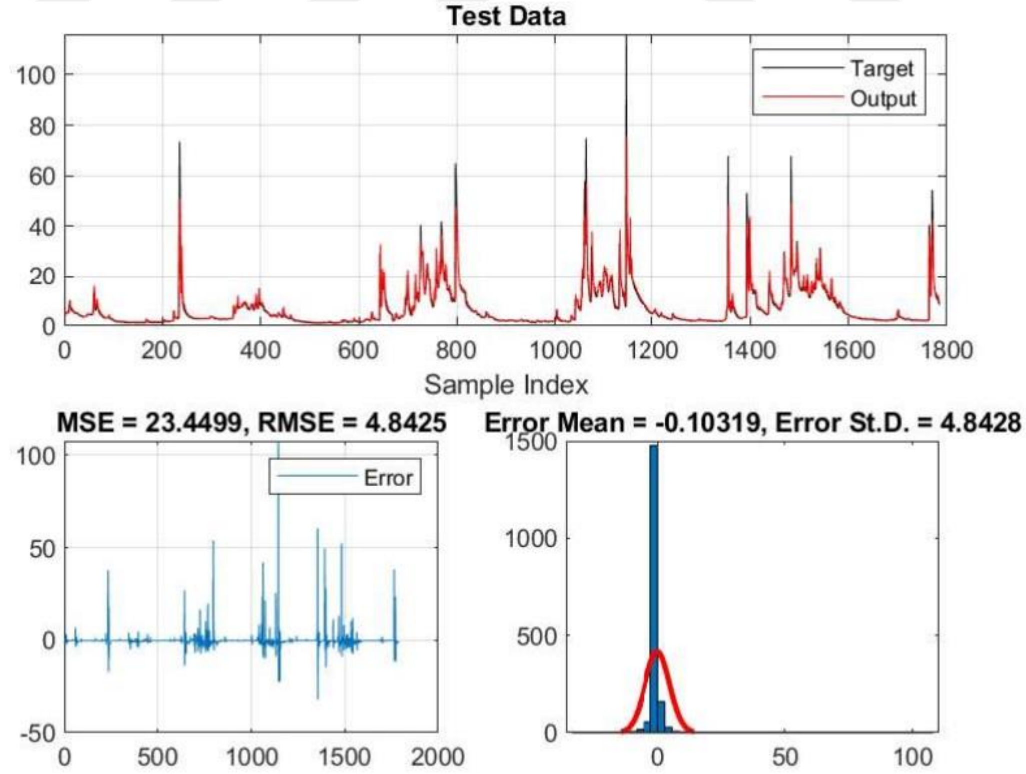


Figure 4.6: Test Performances of 1825 Station's using ANFIS-SA Model with FCM-9 Method (Lag 5)

Table 4.4: Training and Test Performances of 1826 Station's using ANFIS-BP Model with FCM Method

			Scenario											
Models		Data	LAG 3				LAG 4				LAG 5			
			RMSE	**MAE**	**MAPE**	**R^2**	**RMSE**	**MAE**	**MAPE**	**R^2**	**RMSE**	**MAE**	**MAPE**	**R^2**
ANFIS-BP	FCM-2	Train	10.6996	3.3785	4.3502	0.9005	10.6463	3.3997	4.3987	0.9014	10.6523	3.4223	4.4311	0.9013
		Test	**6.7849**	**3.2099**	5.4735	**0.9238**	**6.8225**	3.2438	5.5260	**0.9230**	**6.8422**	3.2532	5.5433	**0.9226**
	FCM-3	Train	10.5855	3.4193	4.4189	0.9024	10.5813	3.3993	4.3668	0.9025	10.5769	3.3996	4.3549	0.9026
		Test	6.9954	3.2749	5.6943	0.9198	6.8802	**3.2268**	5.5473	0.9222	6.8574	3.2180	5.4933	**0.9226**
	FCM-4	Train	10.4968	3.4122	4.3845	0.9040	10.4883	3.4020	4.3480	0.9042	10.4952	3.3974	4.3227	0.9041
		Test	6.8868	3.2867	5.6695	0.9226	6.9382	3.2555	5.5565	0.9213	6.8880	**3.2004**	5.4194	0.9223
	FCM-5	Train	10.4650	3.3686	4.2364	0.9046	10.4408	3.3842	4.2844	0.9051	10.4437	3.3828	4.2743	0.9050
		Test	6.9347	3.2216	5.4080	0.9219	6.9579	3.2425	5.4644	0.9211	6.9480	3.2259	5.4116	0.9213
	FCM-6	Train	10.3150	3.3598	4.2339	0.9073	10.2824	3.3708	4.2875	0.9080	10.2492	3.3831	4.2983	0.9086
		Test	7.0283	3.2772	5.4500	0.9202	7.0388	3.2954	5.5171	0.9197	7.0809	3.2894	5.4779	0.9188
	FCM-7	Train	10.3020	3.3384	4.1929	0.9076	10.2516	3.3601	4.2624	0.9085	10.2176	3.3691	4.2672	0.9091
		Test	7.0375	3.2424	5.3724	0.9202	7.0075	3.2595	5.4483	0.9204	6.9964	3.2443	5.4011	0.9205
	FCM-8	Train	10.2935	3.3276	4.1725	0.9077	10.2327	3.3467	4.2314	0.9088	10.2094	3.3561	4.2457	0.9093
		Test	7.0024	3.2203	5.3228	0.9209	6.9674	3.2347	5.3863	0.9213	6.9534	3.2329	5.3766	0.9214
	FCM-9	Train	10.2571	3.3191	**4.1620**	0.9084	10.2142	3.3474	4.2274	0.9092	10.2146	3.3594	4.2550	0.9092
		Test	7.0043	3.2275	5.3313	0.9211	6.9720	3.2423	5.3910	0.9215	6.9413	3.2359	5.3839	0.9219
	FCM-10	Train	10.2467	3.3222	4.1668	0.9086	10.2129	3.3431	**4.2175**	0.9092	10.2117	3.3539	**4.2437**	0.9092
		Test	6.9863	3.2259	5.3249	0.9215	6.9685	3.2338	5.3659	0.9216	6.9416	3.2359	5.3772	0.9219

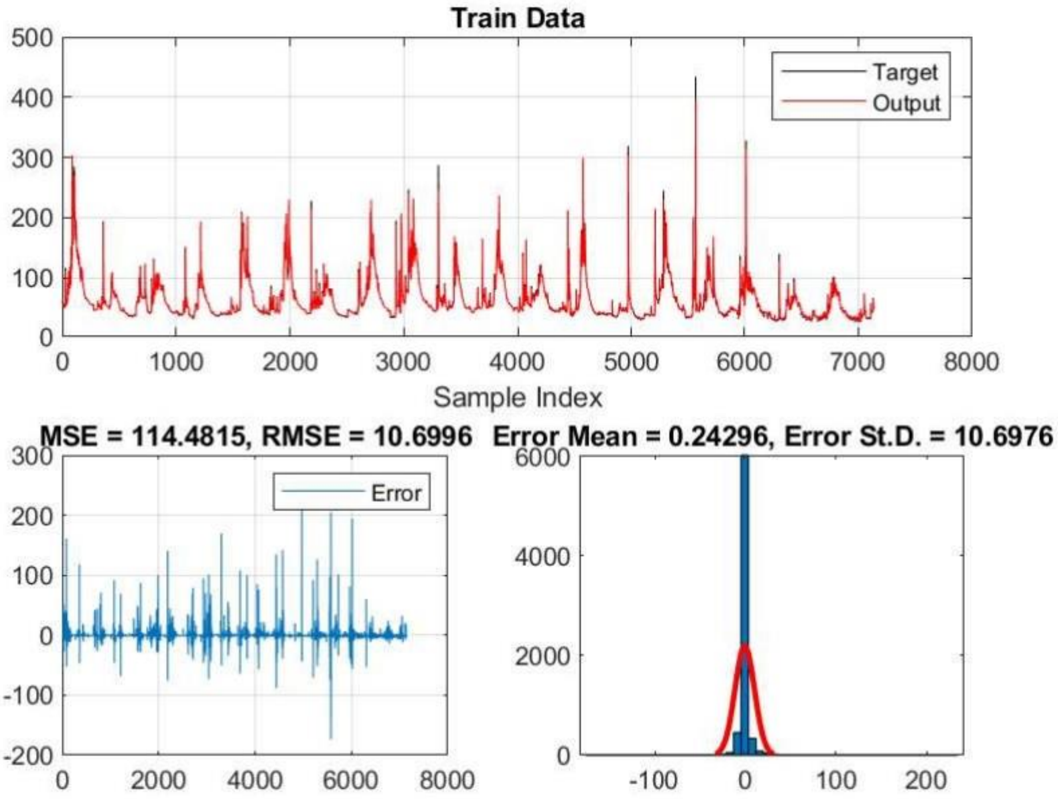


Figure 4.7: Training Performances of 1826 Station’s using ANFIS-BP Model with FCM-2 Method (Lag 3)

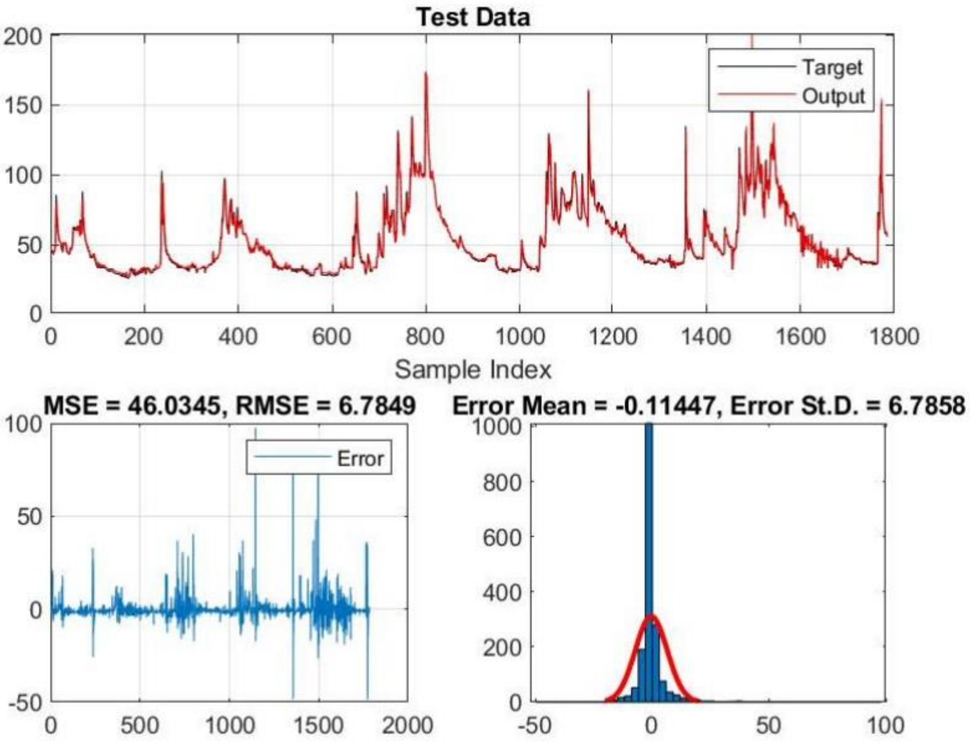


Figure 4.8: Test Performances of 1826 Station’s using ANFIS-BP Model with FCM-2 Method (Lag 3)

Table 4.5: Training and Test Performances of 1826 Station’s using ANFIS-HB Model with FCM Method

			Scenario											
Models		Data	LAG 3				LAG 4				LAG 5			
			RMSE	**MAE**	**MAPE**	**R²**	**RMSE**	**MAE**	**MAPE**	**R²**	**RMSE**	**MAE**	**MAPE**	**R²**
ANFIS-HB	FCM-2	Train	9.9609	3.1707	**4.0596**	0.9136	10.2560	3.3744	4.5146	0.9084	10.2569	3.3959	4.5647	0.9084
		Test	**6.7855**	**3.1306**	5.1768	**0.9254**	**6.8146**	3.3143	5.7521	0.9239	**6.8368**	3.3452	5.8184	0.9233
	FCM-3	Train	10.2916	3.4006	4.3467	0.9078	10.2740	3.3980	4.3376	0.9081	10.2525	3.3429	4.1804	0.9085
		Test	7.0605	3.3544	5.7108	0.9201	7.0390	3.3313	5.6674	0.9201	6.9910	**3.2259**	5.3453	0.9209
	FCM-4	Train	9.9521	3.3619	4.2275	0.9137	10.0054	3.3616	4.2404	0.9128	10.1398	3.3290	**4.1682**	0.9105
		Test	7.0967	3.2793	5.4142	0.9190	7.1041	3.3002	5.4217	0.9181	7.0304	3.2353	5.3472	0.9195
	FCM-5	Train	9.9729	3.2937	4.1239	0.9134	9.8779	3.3083	4.1699	0.9150	9.8701	3.3575	4.2707	0.9152
		Test	7.1846	3.2273	5.2731	0.9178	7.2493	3.2580	5.3136	0.9166	7.3651	3.3499	5.5422	0.9137
	FCM-6	Train	9.9325	3.2600	4.1477	0.9141	9.8336	3.2537	**4.1160**	0.9158	9.6580	3.3253	4.2525	0.9188
		Test	7.1993	3.2798	5.4698	0.9171	7.0498	**3.1882**	5.2784	0.9201	7.2754	3.3557	5.5436	0.9143
	FCM-7	Train	9.8335	3.2850	4.1450	0.9158	9.7691	3.2847	4.1835	0.9169	9.6201	3.3013	4.2599	0.9194
		Test	7.0721	3.2495	5.3156	0.9191	7.2119	3.2762	5.3852	0.9159	7.0829	3.3498	5.6153	0.9183
	FCM-8	Train	9.7053	3.2962	4.1958	0.9180	9.7164	3.3073	4.2504	0.9178	9.2864	3.2858	4.2775	**0.9249**
		Test	7.1286	3.2450	5.3413	0.9175	7.3863	3.3978	5.6669	0.9129	8.0072	3.6088	5.8656	0.9003
	FCM-9	Train	9.6471	3.2127	4.0618	0.9189	9.6088	3.2714	4.1520	0.9196	9.3942	3.2476	4.1813	0.9232
		Test	7.0280	3.2162	5.3098	0.9205	7.1158	3.2271	5.3149	0.9179	6.9458	3.2735	5.4174	0.9226
	FCM-10	Train	9.5460	3.2272	4.1221	0.9206	9.3098	3.2420	4.1620	**0.9245**	9.3425	3.2642	4.2314	0.9240
		Test	7.2771	3.2263	5.3150	0.9150	8.9072	3.4892	5.6788	0.8795	7.0432	3.3251	5.5184	0.9202

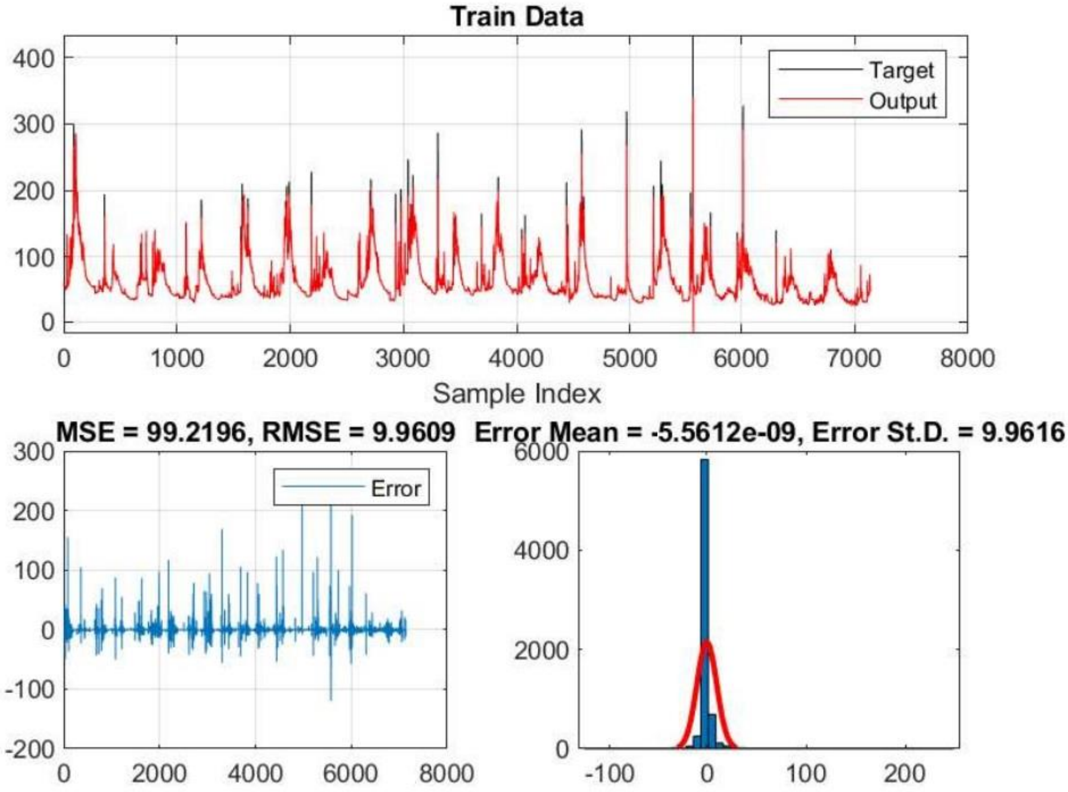


Figure 4.9: Training Performances of 1826 Station's using ANFIS-HB Model with FCM-2 Method (Lag 3)

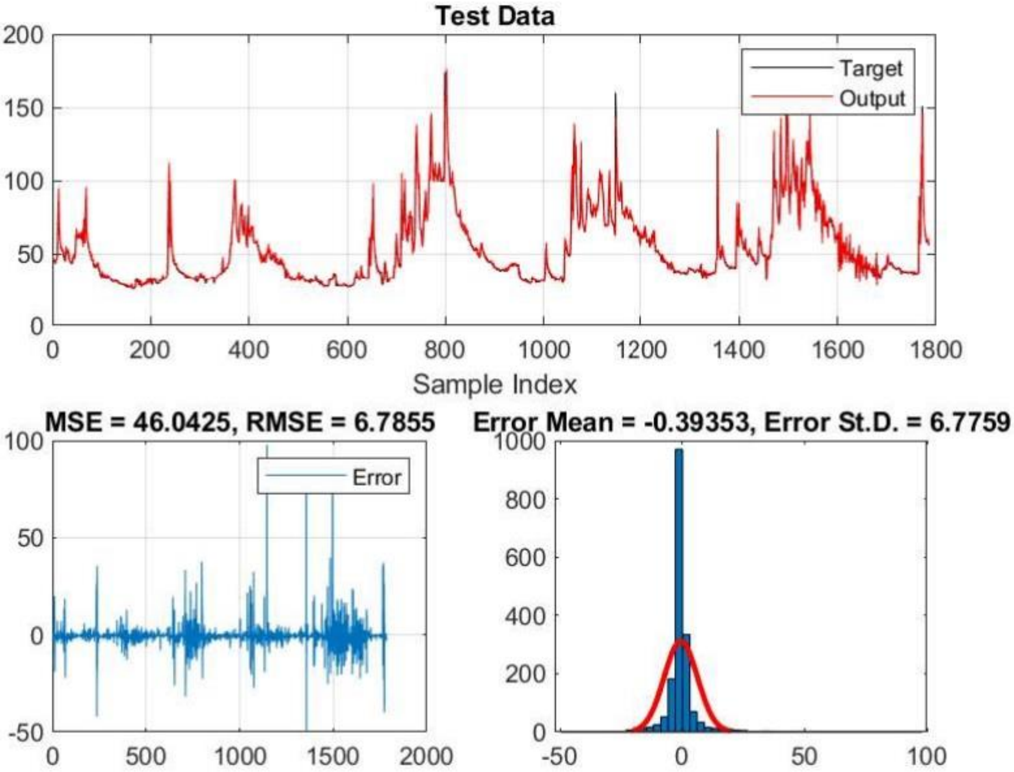


Figure 4.10: Test Performances of 1826 Station's using ANFIS-HB Model with FCM-2 Method (Lag 3)

Table 4.6: Training and Test Performances of 1826 Station's using ANFIS-SA Model with FCM Method

			Scenario											
Models		Data	LAG 3				LAG 4				LAG 5			
			RMSE	**MAE**	**MAPE**	**R²**	**RMSE**	**MAE**	**MAPE**	**R²**	**RMSE**	**MAE**	**MAPE**	**R²**
ANFIS-SA	FCM-2	Train	10.2209	3.1836	**3.9465**	0.9091	10.1966	3.4837	4.6961	0,9096	10.4364	3.3976	4.3375	0.9052
		Test	**6.7476**	3.1246	5.1840	**0.9264**	7.1604	3,3796	5,8774	0,9156	6.9132	3.2177	5.4150	0.9215
	FCM-3	Train	10.4431	3.5483	4.8188	0.9051	10.1002	3.3907	4.3585	0.9112	10.3980	3.4985	4.6717	0.9059
		Test	8.1130	3.4986	6.2663	0.8922	7.0744	3.2768	5.5001	0.9179	6.8224	3.3151	5.7708	0.9233
	FCM-4	Train	10.2379	3.3618	4.2450	0.9087	10.1624	3.3902	4.4162	0.9101	10.2185	3.4437	4.5471	0.9092
		Test	6.7719	3.1765	5.2361	0.9250	7.2991	3.2978	5.5559	0.9121	6.9197	3.3094	5.6802	0.9211
	FCM-5	Train	10.3217	3.4530	4.4516	0.9082	10.3449	3.3960	4.3282	0.9077	10.0131	3.2665	4.1307	0.9134
		Test	7.0032	3.2709	5.4810	0.9218	**6.9528**	3.2575	5.4295	**0.9223**	7.1305	3.2181	5.2720	0.9164
	FCM-6	Train	9.9331	3.2300	4.0531	0.9141	10.0224	3.4254	4.5529	0.9125	9.8640	3.3168	4.3794	0.9154
		Test	6.9853	3.2037	5.2370	0.9223	7.1850	3.3923	5.8078	0.9155	12.3153	3.5277	5.7726	0.7932
	FCM-7	Train	9.8571	3.1735	4.0191	0.9154	10.2590	3.2469	**4.0855**	0.9088	9.6885	3.1367	4.0378	0.9184
		Test	7.2175	3.2019	5.2556	0.9165	7.2760	3.2434	5.3469	0.9159	**6.7839**	3.1283	5.2758	**0.9251**
	FCM-8	Train	9.5494	3.2213	4.1335	0.9206	10.1068	3.2952	4.2185	0.9111	9.8594	3.5760	5.0170	0.9158
		Test	6.8518	3.1350	5.1600	0.9239	6.9750	3.2520	5.4754	0.9209	12.3470	3.8290	6.8089	0.7758
	FCM-9	Train	10.0664	3.2067	4.0835	0.9120	9.8701	3.2383	4.1181	0.9161	9.7911	3.2397	4.0806	0.9172
		Test	7.1203	3.2036	5.2984	0.9186	7.0601	**3.2024**	5.3069	0.9192	7.0690	3.1533	5.0700	0.9188
	FCM-10	Train	9.8512	3.1695	3.9591	0.9155	10.0333	3.2654	4.0902	0.9129	9.3434	**3.0869**	**3.8516**	0.9249
		Test	6.8057	**3.0877**	5.0902	0.9246	7.3774	3.2756	5.4086	0.9130	13.2169	3.2909	5.1419	0.7675

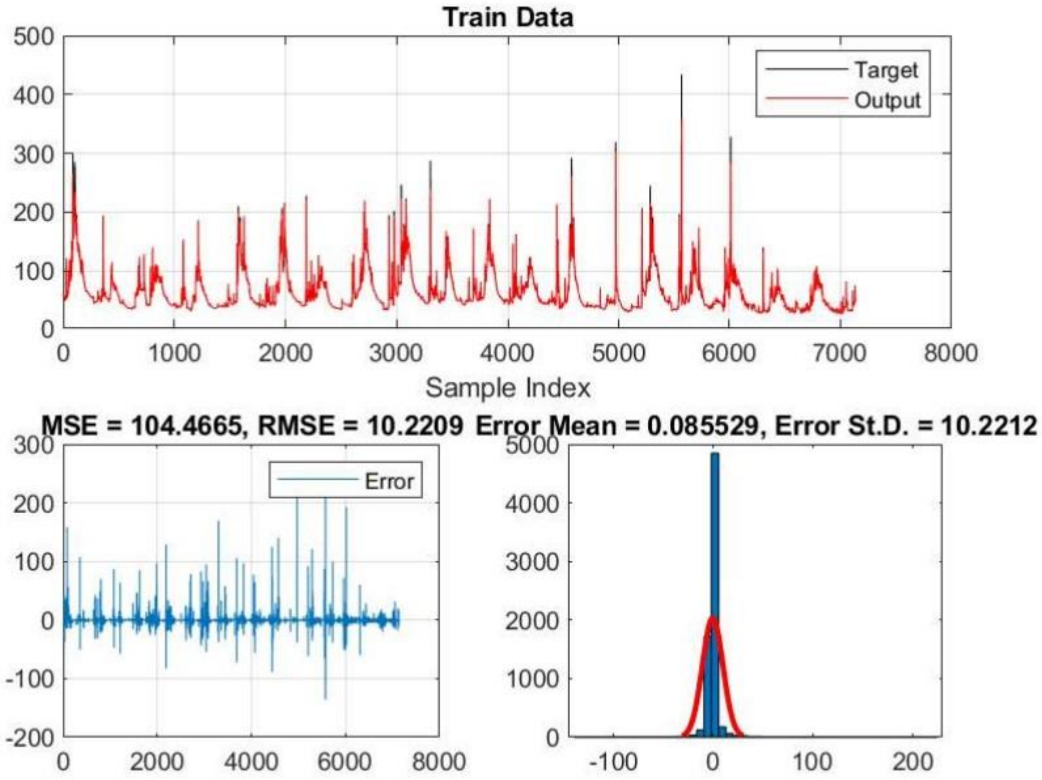


Figure 4.11: Training Performances of 1826 Station's using ANFIS-SA Model with FCM-2 Method (Lag 3)

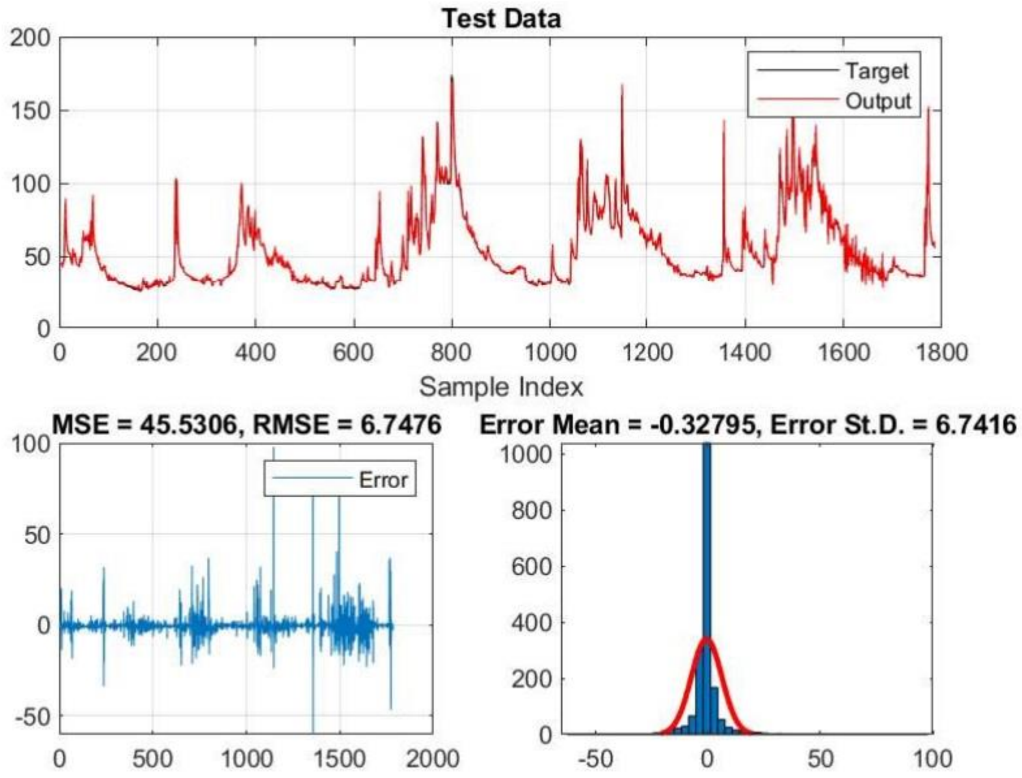


Figure 4.12: Test Performances of 1826 Station's using ANFIS-SA Model with FCM-2 Method (Lag 3)

Table 4.7: Training and Test Performances of 1827 Station's using ANFIS-BP Model with FCM Method

			Scenario											
Models		Data	LAG 3				LAG 4				LAG 5			
			RMSE	**MAE**	**MAPE**	**R^2**	**RMSE**	**MAE**	**MAPE**	**R^2**	**RMSE**	**MAE**	**MAPE**	**R^2**
ANFIS-BP	FCM-2	Train	2.9510	1.3144	6.0275	0.9802	2.9470	1.3172	6.0454	0.9802	2.9437	1.3196	6.0721	0.9803
		Test	3.5258	1.8709	17.1218	0.9323	3.5256	1.8727	17.1494	0.9323	3.5401	1.8802	17.2090	0.9317
	FCM-3	Train	2.9451	1.3107	5.9878	0.9802	2.9437	1.3146	6.0014	0.9803	2.9442	1.3175	6.0285	0.9803
		Test	3.5081	1.8559	16.9030	0.9330	3.4996	1.8537	16.9432	0.9333	3.5013	1.8564	16.9501	0.9332
	FCM-4	Train	2.9370	1.3129	5.9528	0.9804	2.9341	1.3151	5.9677	0.9804	2.9327	**1.3120**	5.9522	0.9804
		Test	3.5065	1.8555	16.7472	0.9332	3.5022	1.8553	16.8491	0.9333	3.4776	1.8392	16.6358	0.9342
	FCM-5	Train	2.9311	1.3125	5.9134	0.9804	2.9291	1.3170	5.9350	0.9805	2.9265	1.3210	5.9629	0.9805
		Test	3.4937	1.8489	16.6213	0.9339	3.4848	1.8481	16.6256	0.9342	3.5028	1.8600	16.7379	0.9335
	FCM-6	Train	2.9316	1.3141	5.9060	0.9804	2.9257	1.3170	5.9133	0.9805	2.9196	1.3207	5.9445	0.9806
		Test	3.5221	1.8633	16.7692	0.9329	3.5228	1.8634	16.7399	0.9329	3.5353	1.8687	16.7931	0.9324
	FCM-7	Train	2.9064	1.3111	**5.8800**	**0.9808**	2.9055	1.3153	5.8957	0.9808	2.9057	1.3203	5.9298	**0.9808**
		Test	3.4932	1.8500	16.5881	0.9339	3.4952	1.8521	16.5709	0.9338	3.5150	1.8585	16.6376	0.9331
	FCM-8	Train	2.9045	1.3108	5.8825	**0.9808**	2.9069	1.3147	5.8861	0.9808	2.9092	1.3188	**5.9186**	0.9807
		Test	3.4919	1.8496	16.5817	0.9339	3.4930	1.8486	16.5363	0.9339	3.4974	1.8514	16.5817	0.9337
	FCM-9	Train	2.9045	1.3105	5.8895	**0.9808**	2.9059	1.3139	5.8908	0.9808	2.9069	1.3184	5.9195	**0.9808**
		Test	3.5021	1.8529	16.5753	0.9336	3.4933	1.8508	16.5474	0.9338	3.5078	1.8551	16.6037	0.9333
	FCM-10	Train	**2.9037**	**1.3095**	5.8901	**0.9808**	**2.8983**	**1.3128**	**5.8707**	**0.9809**	**2.2909**	1.3202	5.9273	**0.9808**
		Test	3.4922	1.8495	16.5667	0.9339	3.4809	1.8444	16.3944	0.9343	3.5235	1.8562	16.5744	0.9327

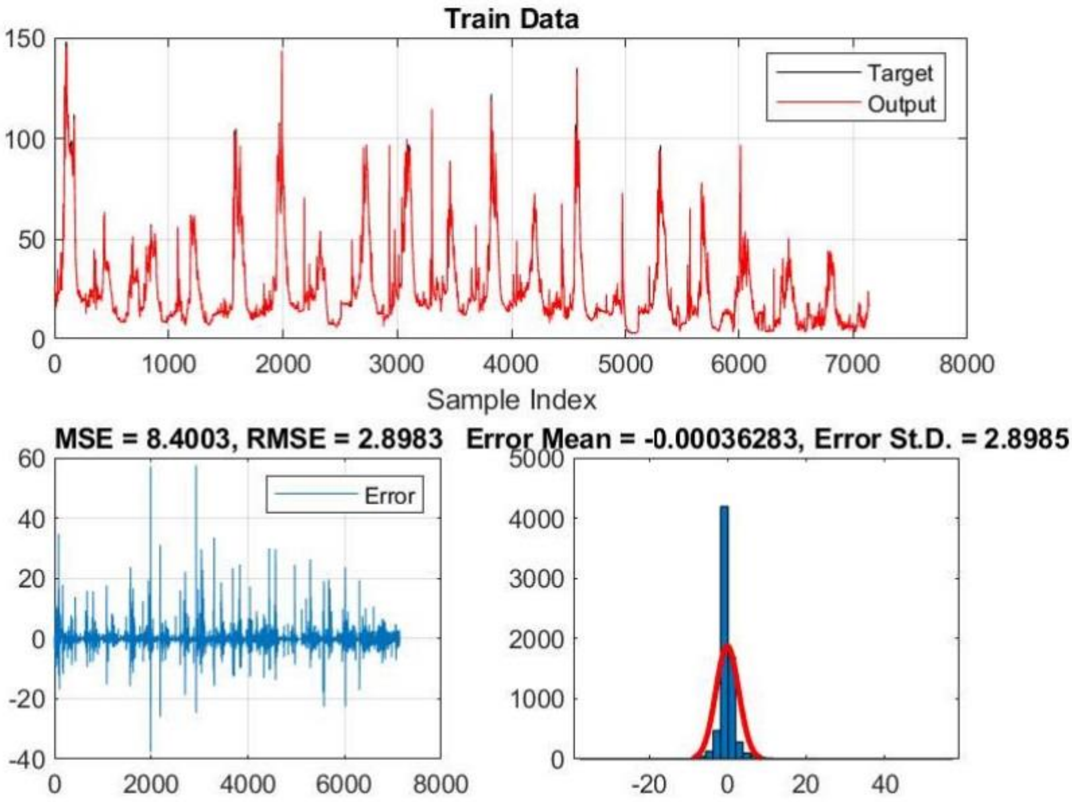


Figure 4.13: Training Performances of 1827 Station's using ANFIS-BP Model with FCM-10 Method (Lag 4)

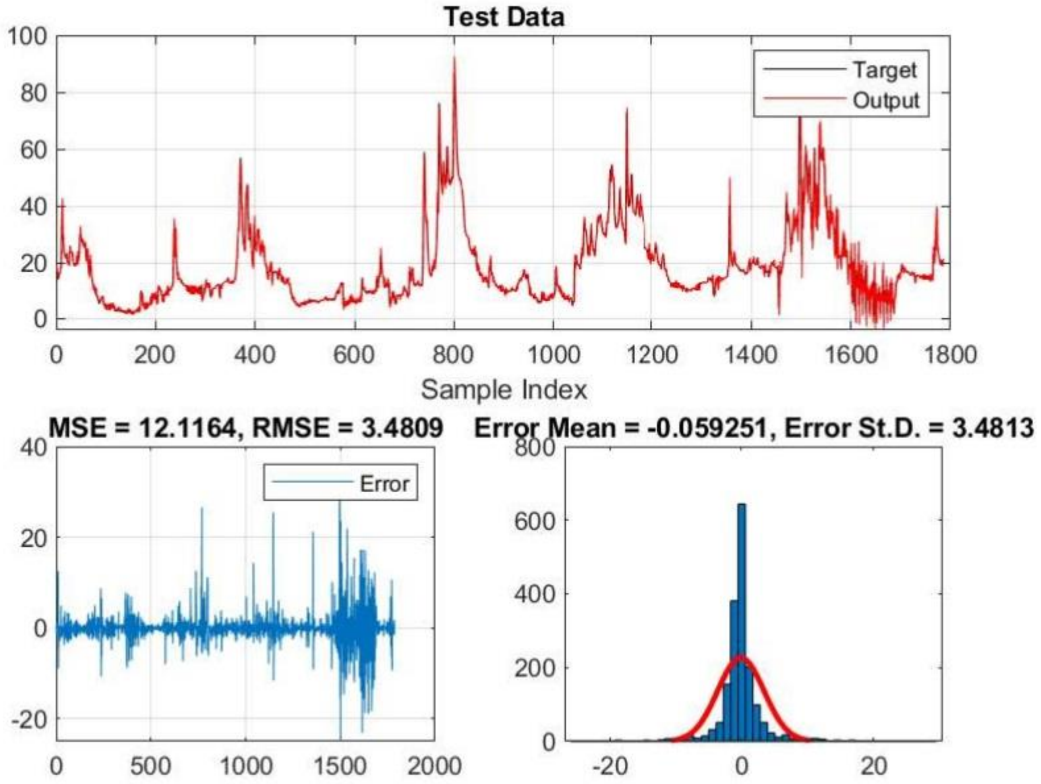


Figure 4.14: Test Performances of 1827 Station's using ANFIS-BP Model with FCM-10 Method (Lag 4)

Table 4.8: Training and Test Performances of 1827 Station's using ANFIS-HB Model with FCM Method

Models		Data	Scenario											
			LAG 3				LAG 4				LAG 5			
			RMSE	**MAE**	**MAPE**	**R^2**	**RMSE**	**MAE**	**MAPE**	**R^2**	**RMSE**	**MAE**	**MAPE**	**R^2**
ANFIS-HB	FCM-2	Train	2.8314	1.2843	5.9745	0.9817	2.8722	1.2916	5.9636	0.9812	2.9286	1.3321	6.1023	0.9805
		Test	3.5508	1.8717	16.7878	0.9317	3.5069	1.8450	16.4316	0.9330	3.5814	1.9095	17.4945	0.9304
	FCM-3	Train	2.8665	1.2719	5.6267	0.9813	2.9035	1.2904	5.7411	0.9808	2.9044	1.2897	5.7254	0.9808
		Test	3.5296	1.8035	15.9189	0.9326	3.4120	1.7838	15.4340	0.9367	3.4131	1.7928	15.6069	0.9367
	FCM-4	Train	2.9011	1.2927	5.7147	0.9808	2.8927	1.2990	5.7363	0.9809	2.8523	1.3044	5.9600	0.9815
		Test	3.3460	1.7576	15.2142	0.9389	3.4029	1.7935	15.6425	0.9371	3.3715	1.7970	15.2878	0.9384
	FCM-5	Train	2.8633	1.2886	5.6658	0.9813	2.8279	1.2934	5.7299	0.9818	2.8293	1.2851	**5.7199**	0.9818
		Test	3.4069	1.7639	15.2695	0.9369	3.4637	1.7978	15.3553	0.9350	3.3954	1.7698	15.1662	0.9371
	FCM-6	Train	2.8410	1.2891	5.7233	0.9816	2.7800	1.2763	5.7165	0.9824	2.8402	1.2977	5.7705	0.9816
		Test	3.4453	1.7786	15.4600	0.9355	3.4263	1.7734	15.4441	0.9361	3.4804	1.7943	15.3062	0.9338
	FCM-7	Train	2.7138	1.2583	5.6283	0.9832	2.7628	1.2756	5.7525	0.9826	2.7738	1.2949	5.7965	0.9825
		Test	105.8811	4.2951	61.4178	0.0101	3.4265	1.7599	15.1478	0.9359	3.4440	1.7864	15.3681	0.9352
	FCM-8	Train	**2.6556**	**1.2464**	**5.6103**	**0.9839**	2.7375	1.2749	**5.7142**	0.9829	2.7733	1.2941	5.8077	0.9825
		Test	3.4647	1.7730	15.0407	0.9348	3.4564	1.7819	15.1964	0.9351	3.4242	1.7587	15.0330	0.9361
	FCM-9	Train	2.7389	1.2620	5.6808	0.9829	2.7312	1.2762	5.7918	0.9830	2.7386	1.2849	5.8132	0.9829
		Test	3.4787	1.8032	15.8392	0.9344	3.4768	1.7768	15.1654	0.9345	3.5089	1.8046	15.0857	0.9330
	FCM-10	Train	2.7483	1.2584	5.6773	0.9829	**2.7147**	**1.2659**	5.7449	**0.9832**	**2.7267**	**1.2817**	5.8151	**0.9831**
		Test	3.4766	1.8043	15.7942	0.9344	3.6109	1.7950	15.2773	0.9291	3.5118	1.7950	15.0825	0.9329

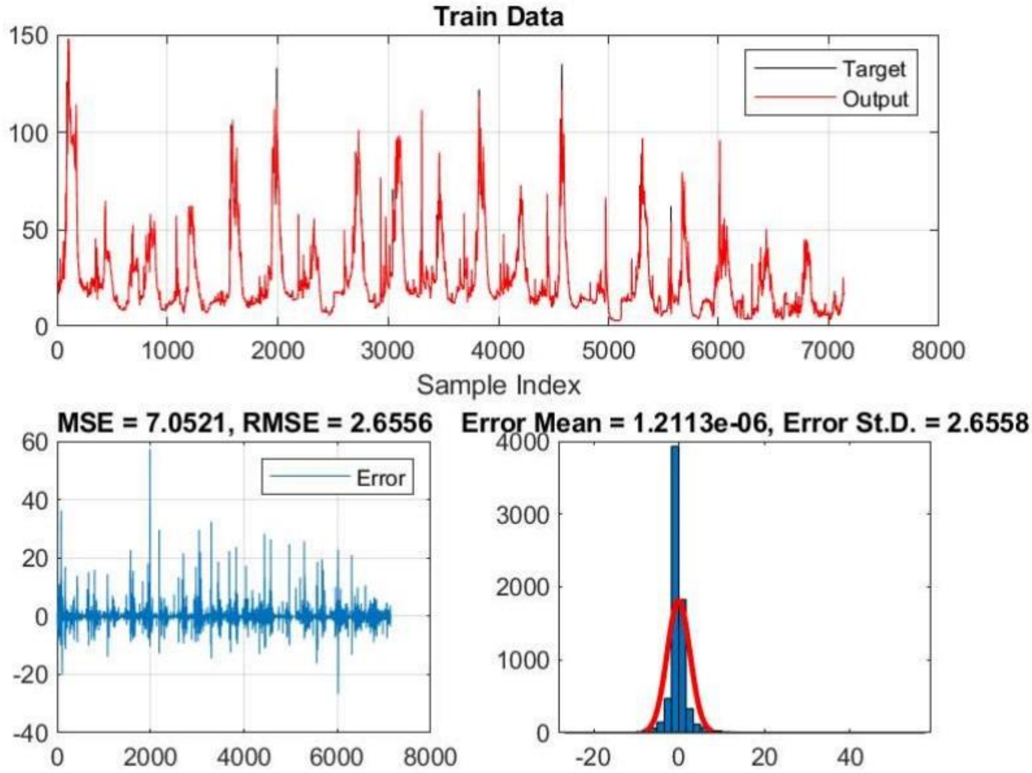


Figure 4.15: Training Performances of 1827 Station's using ANFIS-HB Model with FCM-8 Method (Lag 3)

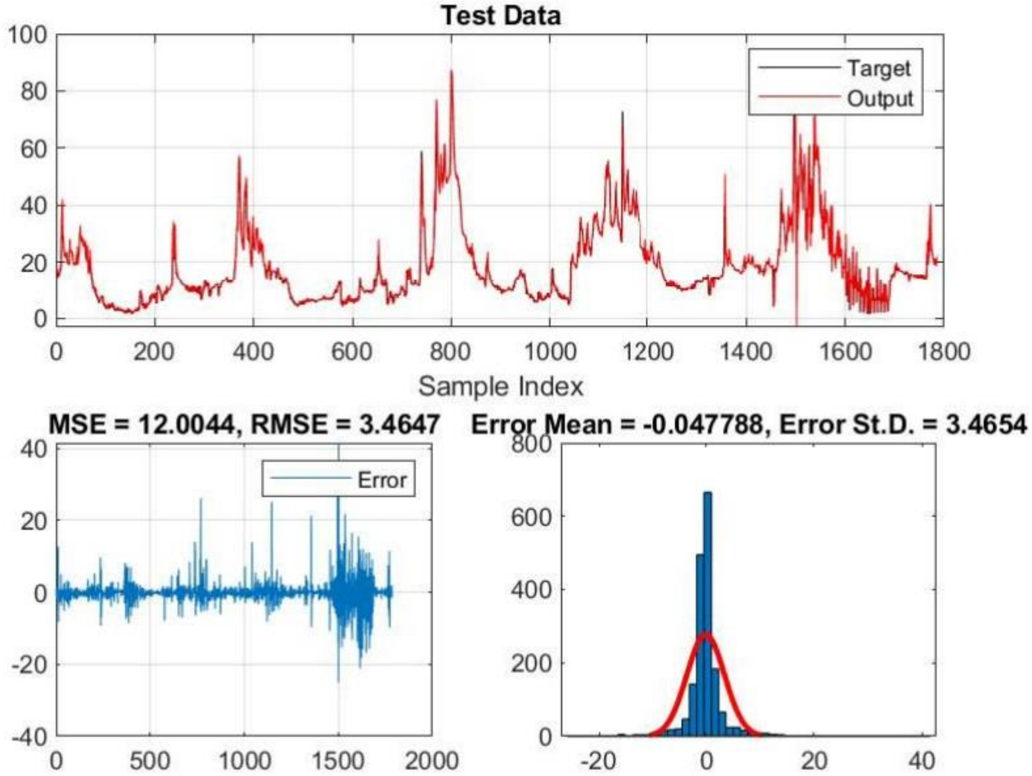


Figure 4.16: Test Performances of 1827 Station's using ANFIS-HB Model with FCM-8 Method (Lag 3)

Table 4.9: Training and Test Performances of 1827 Station's using ANFIS-SA Model with FCM Method

Models		Data	Scenario											
			LAG 3				LAG 4				LAG 5			
			RMSE	**MAE**	**MAPE**	**R^2**	**RMSE**	**MAE**	**MAPE**	**R^2**	**RMSE**	**MAE**	**MAPE**	**R^2**
ANFIS-SA	FCM-2	Train	2.9379	1.3199	6.1202	0.9803	2.9377	1.3076	5.9632	0.9803	2.9216	1.3264	6.1816	0.9806
		Test	5.4673	2.0168	17.5449	0.8591	3.5463	1.8742	17.1538	0.9316	3.5339	1.8812	17.3453	0.9320
	FCM-3	Train	2.8931	1.3038	5.9460	0.9809	2.8833	1.3096	6.0456	0.9811	2.8682	1.3152	6.1310	0.9814
		Test	3.6189	1.8906	17.2027	0.9292	3.5384	1.8781	17.2559	0.9318	5.1534	1.9524	18.0108	0.8610
	FCM-4	Train	2.8131	1.2945	5.9470	0.9820	2.9077	1.2975	5.9036	0.9808	2.8067	1.3118	6.1034	0.9821
		Test	3.4847	1.8499	16.3342	0.9340	10.5990	2.1048	17.4423	0.6322	3.5342	1.8792	17.2907	0.9320
	FCM-5	Train	2.8987	1.3177	6.1287	0.9809	2.8745	1.2846	**5.8310**	0.9812	2.8673	1.3044	6.0189	0.9813
		Test	3.4970	1.8811	17.2926	0.9335	38.9873	2.9414	17.4856	0.1724	3.5814	1.8886	17.2677	0.9302
	FCM-6	Train	2.8529	1.3128	6.0588	0.9815	2.8651	1.3050	5.9309	0.9813	2.8186	1.2980	5.9948	0.9819
		Test	3.7919	1.9158	17.2473	0.9227	11.4611	2.5706	31.9990	0.5408	3.5559	1.8823	17.1992	0.9312
	FCM-7	Train	2.8154	1.2985	6.0163	0.9820	2.8325	1.3094	6.0525	0.9818	**2.7981**	1.2962	5.9093	**0.9822**
		Test	3.5564	1.8846	17.2726	0.9313	3.5501	1.8805	17.2456	0.9315	3.5767	1.8721	16.9817	0.9305
	FCM-8	Train	**2.7891**	1.2875	6.0173	0.9823	**2.7088**	1.2929	5.9967	**0.9833**	2.8606	1.2938	5.9417	0.9814
		Test	7.6050	2.1231	19.2680	0.7402	3.5453	1.8884	17.2981	0.9317	306.3534	9.6828	28.5844	0.0188
	FCM-9	Train	2.8000	**1.2823**	**5.9264**	0.9821	2.7774	**1.2912**	6.0288	0.9824	2.8688	**1.2919**	**5.8884**	0.9813
		Test	4.8682	2.0044	17.4049	0.8836	211.8726	6.9670	23.6764	0.0285	3.6799	1.8954	17.1839	0.9269
	FCM-10	Train	2.7959	1.2953	6.0903	**0.9822**	2.7961	1.2958	6.0000	0.9822	2.8286	1.2993	6.1582	0.9818
		Test	7.0776	2.1319	18.4315	0.7787	3.8408	1.9007	17.1490	0.9207	4.5382	1.9432	17.5487	0.8922

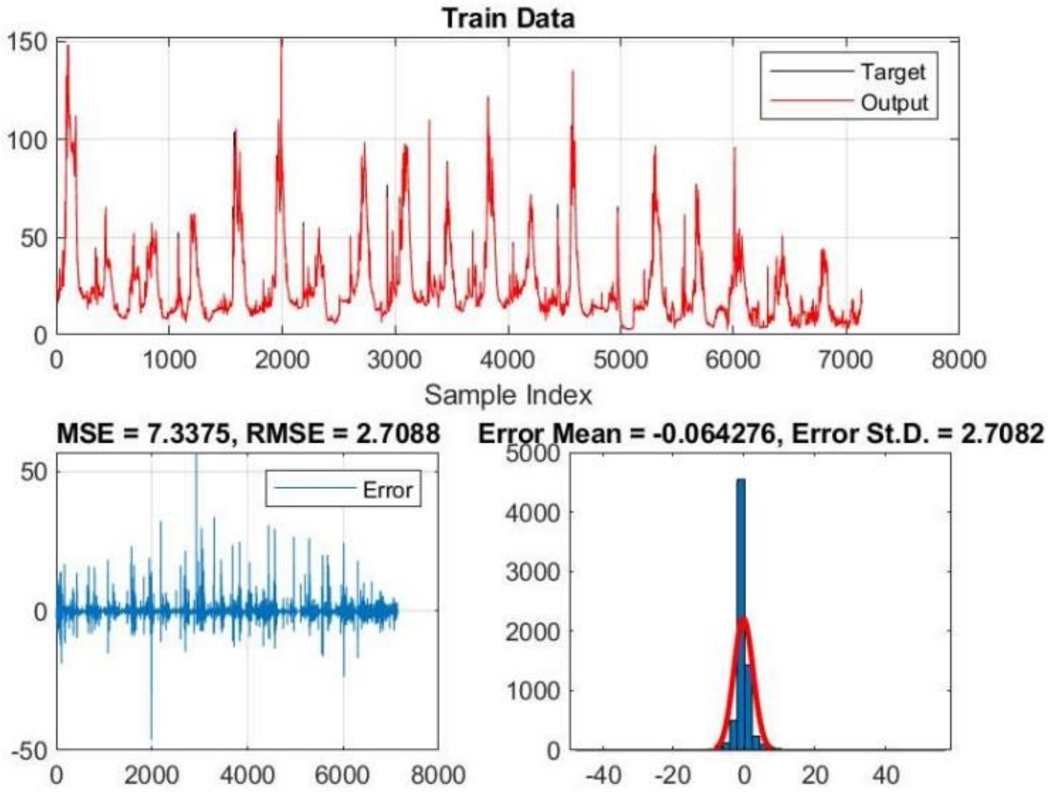


Figure 4.17: Training Performances of 1827 Station's using ANFIS-SA Model with FCM-8 Method (Lag 4)

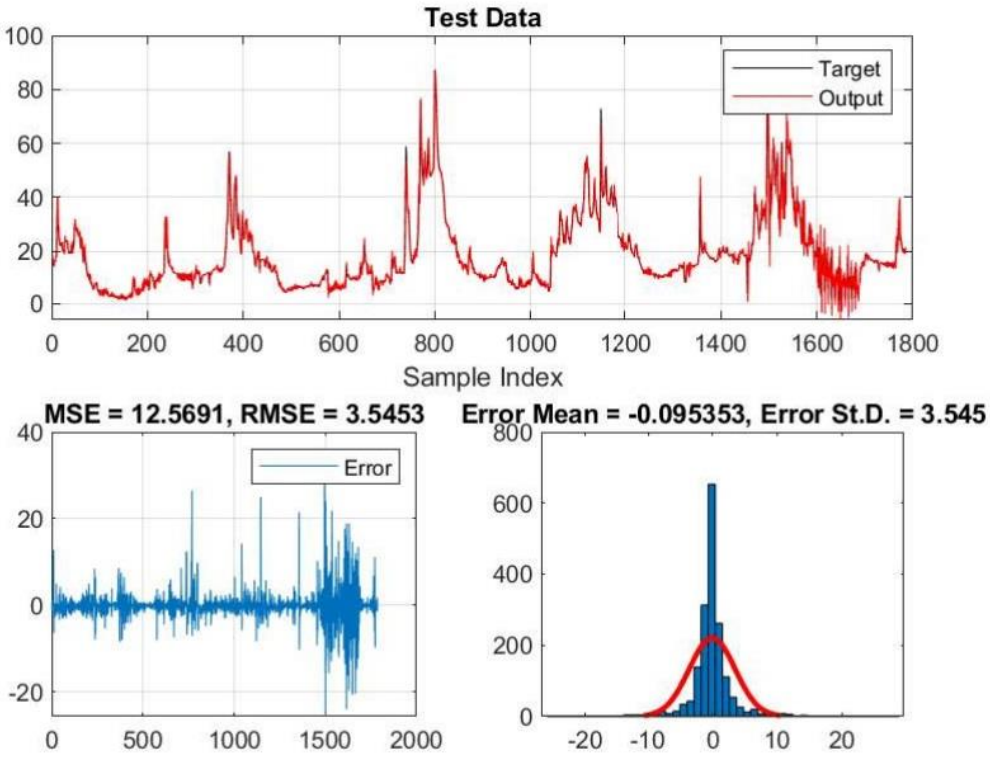


Figure 4.18: Test Performances of 1827 Station's using ANFIS-SA Model with FCM-8 Method (Lag 4)

Table 4.10: Training and Test Performances of 1825 Station's using ANFIS-BP Model with Grid Partition Method

				Scenario											
Models			Data	LAG 3				LAG 4				LAG 5			
				RMSE	**MAE**	**MAPE**	**R^2**	**RMSE**	**MAE**	**MAPE**	**R^2**	**RMSE**	**MAE**	**MAPE**	**R^2**
ANFIS-BP	Gaussmf	Constant(2)	Train	6.3198	1.7074	18.0790	0.6695	6.3443	1.6379	15.3216	0.6669	6.3711	1.6364	15.1006	0.6642
			Test	5.0197	1.4846	22.4005	0.6766	5.0174	1.3914	18.5344	0.6765	5.0265	1.3841	18.2207	0.6758
		Constant(3)	Train	6.4448	1.7468	19.0034	0.6563	6.5195	1.7954	20.2694	0.6483	6.5711	1.8081	20.5265	0.6427
			Test	4.9539	1.5110	24.5128	0.6812	4.9506	1.5366	26.3710	0.6812	4.9528	1.5420	26.7393	0.6807
		Linear(2)	Train	6.0647	1.5608	13.3608	0.6956	6.0732	1.5886	13.6025	0.6947	6.0720	1.5467	12.8640	0.6949
			Test	4.9082	1.2976	13.5225	0.6862	4.9134	1.3236	13.6704	0.6862	4.9082	1.2900	13.0555	0.6854
		Linear(3)	Train	5.8959	1.4971	11.5624	0.7123	5.8738	1.4794	11.7042	0.7145	5.8392	1.4800	11.9706	**0.7179**
			Test	4.8893	1.2515	11.0203	0.6896	4.8992	1.2396	**10.9568**	0.6879	4.8624	1.2349	11.2413	0.6907
	Gbellmf	Constant(2)	Train	8.2089	2.7926	35.7648	0.4469	8.2395	2.8607	36.8211	0.4407	8.2725	2.8330	35.5778	0.4366
			Test	6.0724	2.4999	53.0330	0.5101	6.0890	2.5608	54.3471	0.5088	6.0791	2.5182	52.4468	0.5098
		Constant(3)	Train	7.6513	2.1601	20.1872	0.5222	7.7431	2.2233	21.3999	0.5094	12.6584	6.2658	51.9418	0.4473
			Test	5.6048	1.8287	25.1803	0.5838	5.6517	1.9077	27.3052	0.5766	9.7837	4.9780	61.4029	0.4462
		Linear(2)	Train	6.0728	1.5681	14.5470	0.6948	6.0842	1.5688	13.9142	0.6936	6.0877	1.5532	13.3843	0.6933
			Test	4.9278	1.3264	15.7610	0.6836	4.9303	1.3184	14.7154	0.6833	4.9311	1.3085	14.0514	0.6828
		Linear(3)	Train	5.8920	1.4849	11.5255	**0.7127**	5.8706	1.4676	11.6314	**0.7148**	6.1722	1.4619	10.0710	0.6888
			Test	4.8863	**1.2403**	**10.7278**	0.6896	4.8907	**1.2342**	11.0365	0.6883	**4.8259**	1.1377	9.4539	0.6908

Table 4.10: (continued)

ANFIS-BP	Trimf	Constant(2)	Train	6.4617	1.6960	13.8462	0.6547	6.5007	1.7133	13.8258	0.6506	6.5534	1.7305	13.5673	0.6451
			Test	4.9884	1.4459	18.8079	0.6726	5.0000	1.4487	18.8556	0.6710	5.0074	1.4474	18.4480	0.6701
		Constant(3)	Train	6.5525	1.7705	15.2378	0.6446	6.6573	1.6805	11.3579	0.6335	6.7683	1.8576	16.0852	0.6209
			Test	5.0076	1.4890	20.3476	0.6760	5.0284	1.3704	15.7062	0.6720	5.0377	1.5213	21.1479	0.6706
		Linear(2)	Train	6.0249	1.5427	12.0157	0.6996	6.0245	1.5303	12.0753	0.6997	6.0157	1.5222	12.2582	0.7006
			Test	**4.8693**	1.2604	11.6119	0.6910	**4.8725**	1.2581	11.4289	0.6902	4.8705	1.2578	11.9999	0.6895
		Linear(3)	Train	5.9061	1.5472	12.5838	0.7113	5.8851	1.5103	12.2970	0.7134	6.1596	1.4588	10.0434	0.6893
			Test	4.9206	1.3013	12.5274	0.6864	4.9272	1.2661	11.7233	0.6846	4.8660	1.1540	9.6997	0.6874
	Trapmf	Constant(2)	Train	6.3872	1.6228	14.7888	0.6630	6.4229	1.6364	14.9487	0.6593	6.4614	1.6564	15.5458	0.6555
			Test	4.8941	1.3953	22.8562	0.6821	4.8954	1.3989	23.0942	0.6818	4.8957	1.4109	23.9551	0.6815
		Constant(3)	Train	6.4063	1.6202	12.2318	0.6605	6.5302	1.7509	14.2611	0.6472	12.2413	6.0336	59.9337	0.2241
			Test	4.9146	1.3566	16.3143	0.6851	4.9316	1.4591	19.5202	0.6820	9.4130	4.8954	76.1913	0.1974
		Linear(2)	Train	6.1420	1.6406	18.2640	0.6878	6.1368	1.6222	16.7799	0.6883	6.1372	1.5782	15.1698	0.6884
			Test	5.0110	1.4373	22.2419	0.6729	5.0102	1.4083	19.8576	0.6734	5.0021	1.3653	17.5910	0.6735
		Linear(3)	Train	5.9345	1.5104	12.8573	0.7085	5.9165	1.5065	12.8938	0.7103	6.1318	1.4153	9.0577	0.6920
			Test	4.8970	1.2692	12.9398	0.6880	4.8745	1.2669	13.0454	0.6894	4.8692	**1.1273**	**8.4550**	0.6858

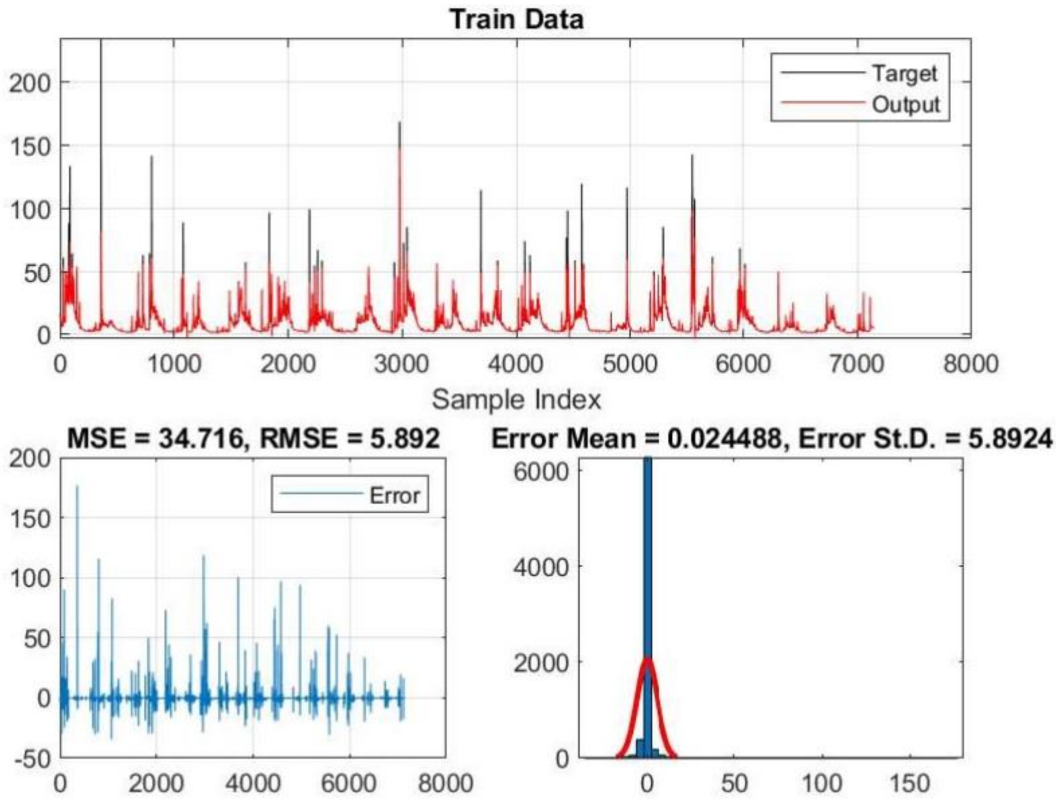


Figure 4.19: Training Performances of 1825 Station's using ANFIS-BP Model with Gbellmf (Linear 3 Lag 3)

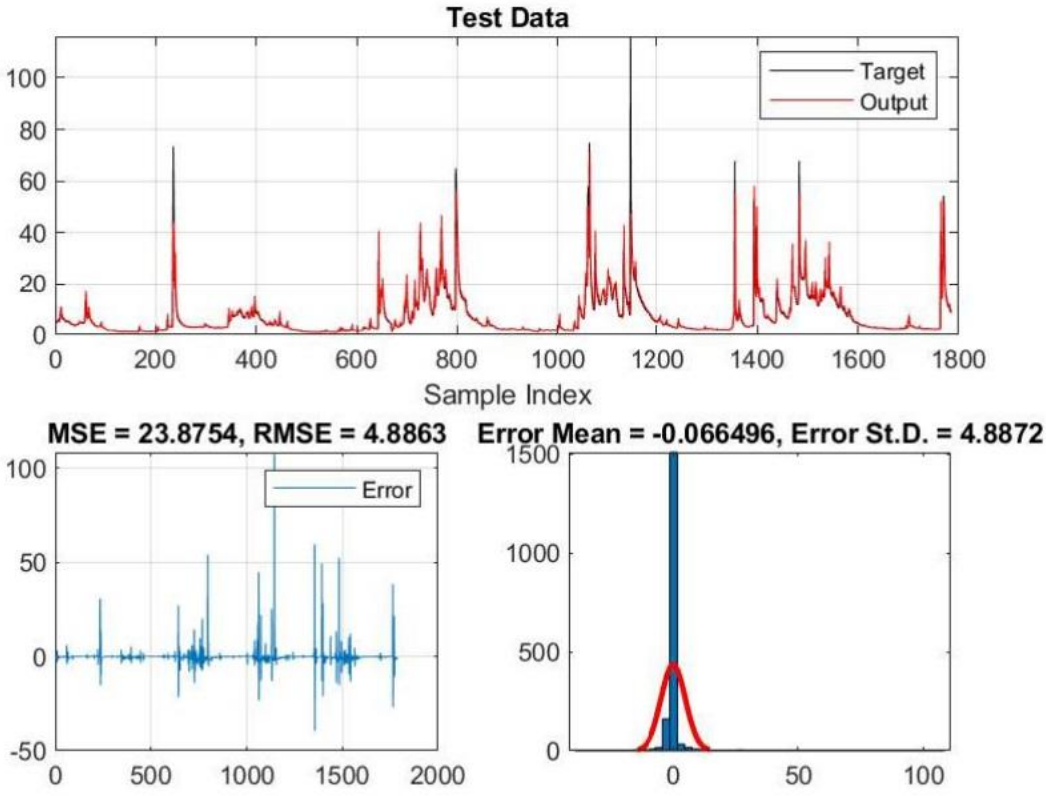


Figure 4.20: Test Performances of 1825 Station's using ANFIS-BP Model with Gbellmf (Linear 3 Lag 3)

Table 4.11: Training and Test Performances of 1825 Station's using ANFIS-HB Model with Grid Partition Method

Models			Data	Scenario											
				LAG 3				LAG 4				LAG 5			
				RMSE	**MAE**	**MAPE**	**R^2**	**RMSE**	**MAE**	**MAPE**	**R^2**	**RMSE**	**MAE**	**MAPE**	**R^2**
ANFIS-HB	Gaussmf	Constant(2)	Train	6.1434	1.5890	13.3374	0.6876	6.0890	1.5934	13.6996	0.6931	5.8368	1.6683	17.0060	0.7181
			Test	4.9273	1.3099	13.1895	0.6844	5.0513	1.3601	14.2189	0.6688	5.0656	1.5720	28.8592	0.6715
		Constant(3)	Train	5.8338	1.5144	12.5348	0.7183	5.5471	1.4985	12.8671	0.7453	5.3155	1.4482	12.8731	0.7662
			Test	5.0706	1.3365	13.5644	0.6716	8.6938	1.5517	12.4322	0.3243	5.3614	**1.3516**	12.6632	0.6351
		Linear(2)	Train	5.8167	1.4803	0.7200	0.6768	5.4374	1.4647	12.5508	**0.7553**	4.9943	1.3584	12.0361	**0.7936**
			Test	5.0056	1.2813	**11.0533**	0.6768	15.8805	1.7287	13.0803	0.0935	116.3164	4.9227	20.7351	0.0012
		Linear(3)	Train	5.7280	1.4585	11.9111	0.7284	5.8738	1.4794	11.7042	0.7145	-	-	-	-
			Test	6.0510	1.3764	11.7841	0.5905	4.8992	**1.2396**	**10.9568**	0.6879	-	-	-	-
	Gbellmf	Constant(2)	Train	6.1237	1.7347	16.9502	0.6896	6.1051	1.7174	16.9639	0.6915	6.0503	1.6859	16.5428	0.6971
			Test	4.8877	1.5107	23.3718	0.6899	4.8834	1.5055	23.8870	0.6902	**4.8533**	1.4753	23.4277	0.6925
		Constant(3)	Train	5.8976	1.5068	11.9631	0.7121	5.7807	1.4689	11.6301	0.7234	5.8915	1.6421	16.7536	0.7128
			Test	4.9131	**1.2777**	11.8157	0.6858	5.1278	1.2915	11.4546	0.6684	4.9661	1.5372	25.6443	0.6774
		Linear(2)	Train	5.8343	1.4907	12.0609	0.7183	5.7197	1.4545	12.1931	0.7292	5.5843	1.4217	**12.0122**	0.7419
			Test	5.0100	1.2997	12.7120	0.6771	5.4998	1.3880	15.7042	0.6347	5.8268	1.4536	13.3476	0.6104
		Linear(3)	Train	5.4275	1.4676	12.6517	**0.7562**	5.4497	1.4129	11.9143	0.7542	-	-	-	-
			Test	13.8654	1.9194	15.7022	0.2407	29.3676	2.5994	15.3676	0.0366	-	-	-	-

Table 4.11: (continued)

ANFIS-HB	Trimf	Constant(2)	Train	6.0700	1.5557	12.4043	0.6950	6.0542	1.5462	12.5552	0.6966	5.9631	1.5725	13.6453	0.7057
			Test	4.9097	1.3202	15.5448	0.6884	4.9035	1.3140	15.7657	0.6885	4.8938	1.3577	18.4139	0.6880
		Constant(3)	Train	5.9071	1.5186	11.9987	0.7112	5.8395	1.5133	12.6735	0.7178	5.9235	1.6071	14.4012	0.7096
			Test	5.0266	1.3154	13.0274	0.6731	4.9280	1.3481	16.8735	0.6842	5.2212	1.4864	18.9379	0.6489
		Linear(2)	Train	5.7433	1.5683	15.2262	0.7270	5.8202	1.5131	12.6457	0.7196	5.6116	1.4609	12.2524	0.7394
			Test	5.1620	1.4309	19.0169	0.6621	5.0732	1.3715	16.1121	0.6693	5.2917	1.4141	15.3954	0.6399
		Linear(3)	Train	5.7789	1.4781	12.1719	0.7236	5.7098	1.4597	12.0127	0.7302	-	-	-	-
			Test	4.9471	1.3269	15.7510	0.6820	5.0986	1.3149	12.6824	0.6703	-	-	-	-
	Trapmf	Constant(2)	Train	6.1201	1.7309	16.3347	0.6900	6.1187	1.7018	15.2583	0.6901	6.0806	1.7155	16.1533	0.6940
			Test	**4.8599**	1.5267	23.8019	0.6907	**4.8543**	1.4857	21.5143	0.6918	4.8648	1.5267	23.7623	0.6898
		Constant(3)	Train	6.0227	1.5622	11.8274	0.6998	5.9870	1.5581	11.9871	0.7033	8.9937	5.5992	119.8100	0.3306
			Test	4.9033	1.3289	12.5907	0.6884	4.9444	1.3393	12.1147	0.6821	7.8721	5.4314	167.9558	0.2143
		Linear(2)	Train	5.9150	1.5588	16.4807	0.7104	5.8821	1.5430	16.5524	0.7136	5.8381	1.5168	16.1564	0.7169
			Test	5.1034	1.4122	19.5997	0.6637	5.1074	1.4168	19.8251	0.6639	5.0308	1.3882	19.2431	0.6719
		Linear(3)	Train	5.7884	1.4770	12.6093	0.7227	5.7659	1.4645	13.0120	0.7248	-	-	-	-
			Test	5.4429	1.3477	13.1803	0.6405	12.3049	1.6676	14.7245	0.3008	-	-	-	-

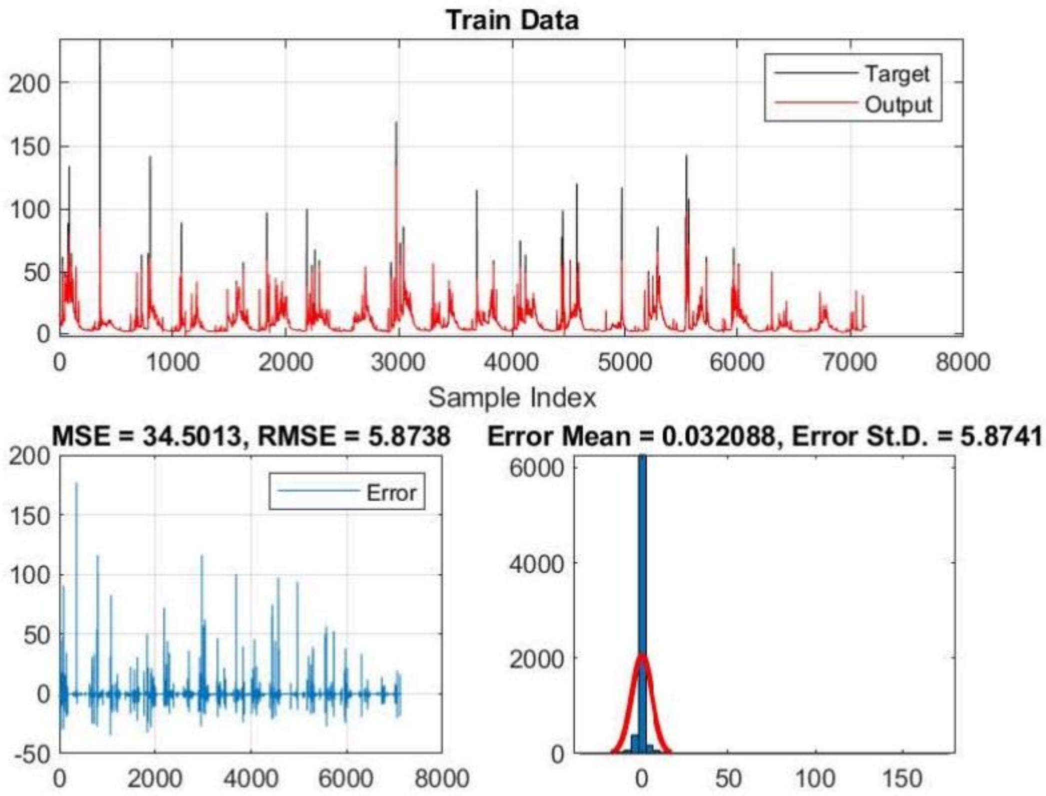


Figure 4.21: Training Performances of 1825 Station's using ANFIS-HB Model with Gaussmf (Linear 3 Lag 4)

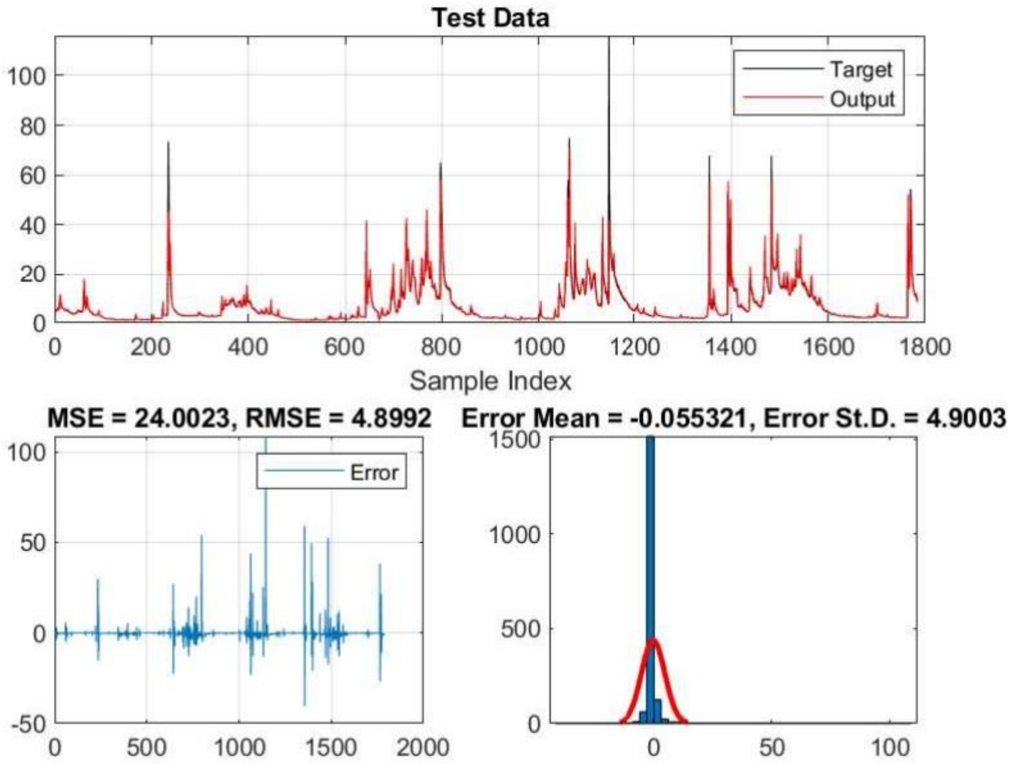


Figure 4.22: Test Performances of 1825 Station's using ANFIS-HB Model with Gaussmf (Linear 3 Lag 4)

Table 4.12: Training and Test Performances of 1825 Station's using ANFIS-SA Model with Grid Partition Method

				Scenario											
Models			Data	LAG 3				LAG 4				LAG 5			
				RMSE	**MAE**	**MAPE**	**R^2**	**RMSE**	**MAE**	**MAPE**	**R^2**	**RMSE**	**MAE**	**MAPE**	**R^2**
ANFIS-SA	Gaussmf	Constant(2)	Train	6.8705	2.7953	49.9317	0.6102	6.4530	2.1600	30.8075	0.6562	6.8190	2.7547	55.9771	0.6209
			Test	5.4090	2.6585	72.9619	0.6205	5.3977	2.0297	45.4226	0.6288	5.5108	2.7495	80.2102	**0.6121**
		Constant(3)	Train	6.2464	2.1816	37.0669	0.6785	6.2250	2.3715	50.0342	0.6846	5.9918	1.7935	23.5515	0.7032
			Test	5.2346	2.1141	54.9159	0.6426	5.2710	2.3705	72.1179	0.6435	5.1303	1.7080	34.0716	0.6573
		Linear(2)	Train	6.3192	1.7849	18.1008	0.6696	6.1332	1.7202	16.3953	0.6891	6.1595	1.7171	**17.0772**	0.6864
			Test	4.9523	1.4806	18.8729	0.6766	**4.8818**	**1.4347**	16.8397	0.6889	**4.8908**	1.4435	18.0599	0.6873
		Linear(3)	Train	6.0346	1.6645	**16.3099**	0.6988	6.8219	2.8057	51.7382	0.6158	-	-	-	-
			Test	**4.9341**	**1.4218**	17.1503	0.6844	5.4118	2.6978	75.5362	0.6186	-	-	-	-
	Gbellmf	Constant(2)	Train	6.6988	2.2181	21.7191	0.6286	6.2997	1.6940	16.0655	0.6716	6.1645	1.8291	22.4472	0.6859
			Test	5.2848	1.9151	27.7268	0.6311	4.9218	1.4904	20.0308	0.6815	5.1689	1.7295	33.6782	0.6626
		Constant(3)	Train	6.0022	1.7489	20.8577	0.7019	6.2342	1.7406	19.6032	0.6786	6.0927	**1.7042**	20.0821	0.6930
			Test	5.2396	1.7256	31.1660	0.6494	4.8902	1.5507	28.2496	0.6856	4.9022	1.5538	29.5218	0.6841
		Linear(2)	Train	5.9975	1.7381	20.2248	**0.7031**	6.1471	2.0384	28.3843	**0.6895**	6.3004	1.9200	27.5197	0.6734
			Test	4.9360	1.5216	22.2962	0.6816	5.0228	1.7968	30.6629	0.6710	4.9855	1.7432	39.0075	0.6747
		Linear(3)	Train	6.5461	1.8906	18.8867	0.6456	6.0540	1.6388	**15.6562**	0.6967	-	-	-	-
			Test	5.0998	1.6192	21.4806	0.6593	4.9771	1.4425	18.6774	0.6777	-	-	-	-

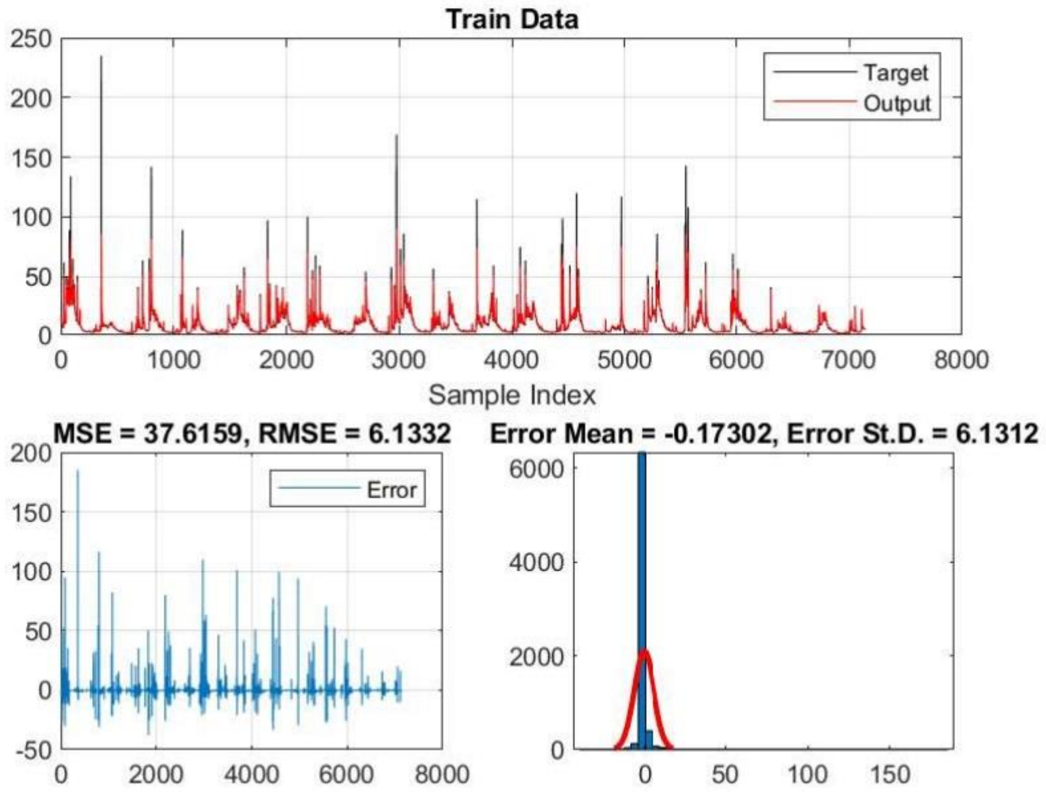


Figure 4.23: Training Performances of 1825 Station's using ANFIS-SA Model with Gaussmf (Linear 2 Lag 4)

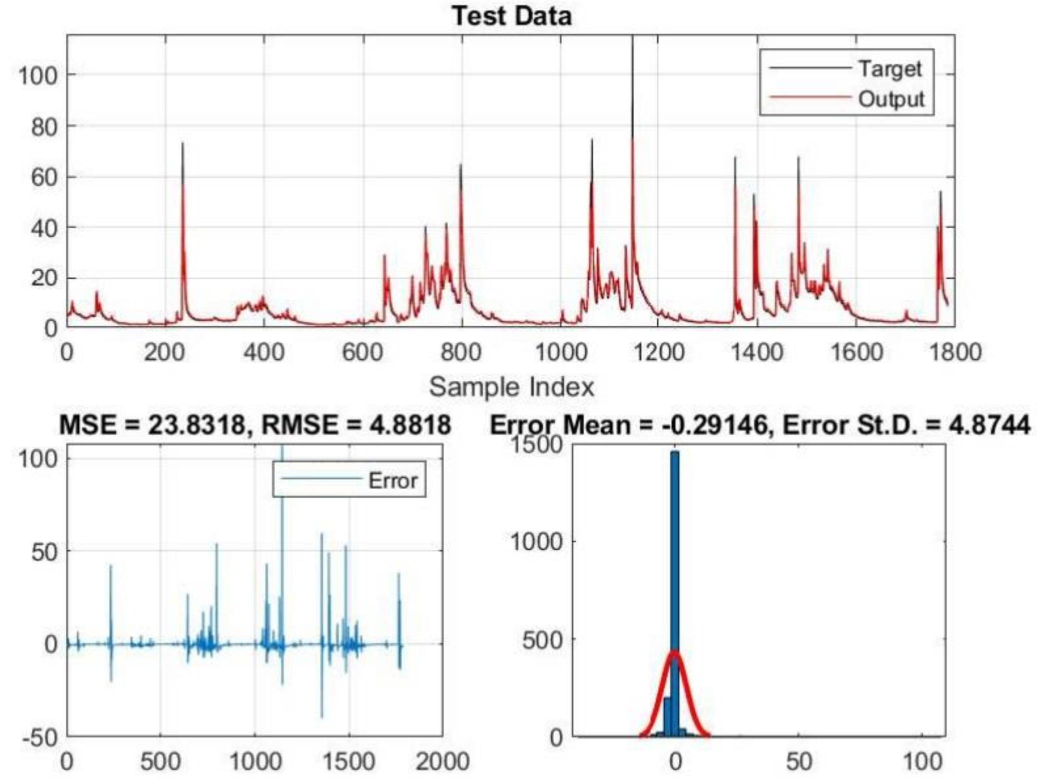


Figure 4.24: Test Performances of 1825 Station's using ANFIS-SA Model with Gaussmf (Linear 2 Lag 4)

Table 3.13: Training and Test Performances of 1826 Station's using ANFIS-BP Model with Grid Partition Method

				Scenario											
Models			Data	LAG 3				LAG 4				LAG 5			
				RMSE	**MAE**	**MAPE**	**R²**	**RMSE**	**MAE**	**MAPE**	**R²**	**RMSE**	**MAE**	**MAPE**	**R²**
ANFIS-BP	Gaussmf	Constant(2)	Train	10.6350	3.7307	5.0714	0.9015	10.6431	3.7149	5.0240	0.9014	10.6521	3.7175	5.0255	0.9012
			Test	7.1132	3.6992	6.7864	0.9168	7.1038	3.6769	6.7175	0.9171	7.1064	3.6793	6.7225	0.9170
		Constant(3)	Train	10.8363	3.6075	4.7086	0.8978	10.8335	3.7236	5.0290	0.8979	68.8598	59.5733	95.7696	0.7765
			Test	7.0182	3.5135	6.2267	0.9191	7.0517	3.6715	6.6880	0.9191	56.5388	50.6291	94.8973	0.8447
		Linear(2)	Train	10.0186	3.1754	3.9877	0.9126	10.0467	3.1725	3.9349	0.9121	10.0999	3.2154	**3.9745**	0.9112
			Test	6.8262	3.1396	5.1714	0.9247	6.8508	3.1454	5.1409	**0.9242**	**6.8502**	**3.1554**	5.1395	**0.9242**
		Linear(3)	Train	9.8946	3.2166	4.1396	0.9147	9.8769	3.1687	4.0696	0.9151	12.2256	5.3238	8.0331	0.8804
			Test	6.8187	3.1931	5.3523	**0.9249**	**6.8407**	3.1752	5.3179	0.9239	7.9113	4.6763	8.8409	0.9138
	Gbellmf	Constant(2)	Train	28.4511	15.4165	22.4477	0.3676	28.6218	15.4605	22.1947	0.3425	28.7600	15.3026	22.0030	0.3531
			Test	16.3021	12.2455	25.9894	0.6679	15.9006	11.8709	24.7737	0.6598	15.9038	11.8881	25.0482	0.6789
		Constant(3)	Train	36.4507	17.8188	22.4341	0.0362	39.9200	18.8513	22.5209	0.0014	68.5668	59.1857	94.9821	0.7751
			Test	19.4945	12.3168	23.3976	0.4597	21.3054	12.7332	23.3367	0.3947	56.1894	50.2372	94.0481	0.8015
		Linear(2)	Train	10.0698	3.1976	4.0167	0.9117	10.1058	3.2726	4.1130	0.9111	10.1656	3.3148	4.1340	0.9100
			Test	6.8439	3.1439	5.1690	0.9243	6.8797	3.1992	5.2367	0.9240	6.8904	3.2134	5.2299	0.9239
		Linear(3)	Train	9.8550	3.1465	4.0136	0.9154	9.8633	3.2784	4.3067	0.9153	12.6177	5.4059	7.8243	0.8707
			Test	**6.8146**	3.1478	5.2804	0.9245	6.8467	3.2303	5.4811	0.9239	8.0137	4.5664	8.3737	0.9077

Table 4.13: (continued)

ANFIS-BP	Trimf	Constant(2)	Train	11.1984	3.5199	4.1378	0.8909	11.6763	3.5928	4.1811	0.8814	12.0390	3.9319	4.9079	0.8738
			Test	7.0371	3.2078	5.1396	0.9189	7.0430	3.2055	5.1378	0.9188	7.0332	3.4083	5.7011	0.9197
		Constant(3)	Train	10.8004	3.5222	4.3377	0.8995	11.0684	3.7880	4.7895	0.8934	57.7893	40.0893	52.2985	0.7097
			Test	6.8824	3.2784	5.4652	0.9233	7.0517	3.4805	5.8931	0.9207	43.3761	30.0767	45.2555	0.8125
		Linear(2)	Train	10.0070	3.1834	3.9939	0.9128	10.0446	3.2190	4.0395	0.9122	10.1147	3.2489	4.0629	0.9109
			Test	6.8439	3.1694	5.2568	0.9245	6.8757	3.1904	5.2815	0.9236	6.8763	3.1856	5.2686	0.9233
		Linear(3)	Train	9.9512	3.1824	4.0477	0.9138	9.9262	3.2737	4.2701	0.9142	12.1523	4.7400	6.7909	0.8767
			Test	6.8478	3.1976	5.3559	0.9241	6.8785	3.2975	5.6004	0.9235	7.6595	4.3326	8.1180	0.9128
	Trapmf	Constant(2)	Train	10.4797	3.5627	4.5736	0.9044	10.5595	3.5906	4.6040	0.9029	10.6392	3.5151	4.4237	0.9015
			Test	6.8686	3.3354	5.6283	0.9230	6.8730	3.3416	5.6406	0.9230	6.8734	3.2918	5.5318	0.9229
		Constant(3)	Train	10.8542	3.6892	4.7100	0.8975	10.9915	3.7569	4.7636	0.8949	67.7798	58.1490	92.8824	0.6425
			Test	6.8385	3.3821	5.6987	0.9240	6.8887	3.3861	5.6596	0.9229	55.2539	49.1772	91.7347	0.5203
		Linear(2)	Train	10.1731	3.2540	4.0441	0.9099	10.2600	3.2036	**3.9244**	0.9084	10.4381	3.4049	4.2336	0.9051
			Test	6.8740	3.1564	5.1315	0.9242	6.8671	**3.1142**	5.0359	0.9238	6.8945	3.2106	5.2126	0.9238
		Linear(3)	Train	9.8861	3.1584	**3.9750**	0.9149	9.9454	3.2821	4.1987	0.9139	13.5928	7.7717	12.0153	0.8501
			Test	6.8188	**3.1290**	5.1961	0.9244	6.8557	3.2024	5.3176	0.9239	8.9097	6.3610	12.0701	0.9019

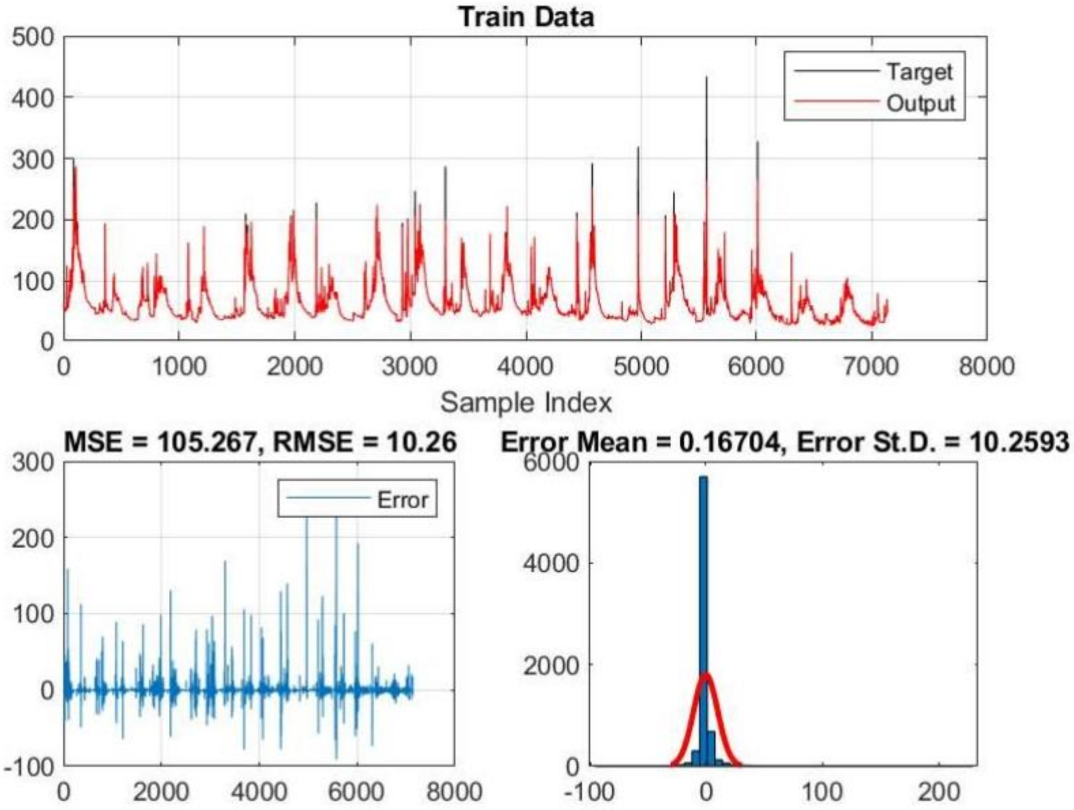


Figure 4.25: Training Performances of 1826 Station's using ANFIS-BP Model with Trapmf (Linear 2 Lag 4)

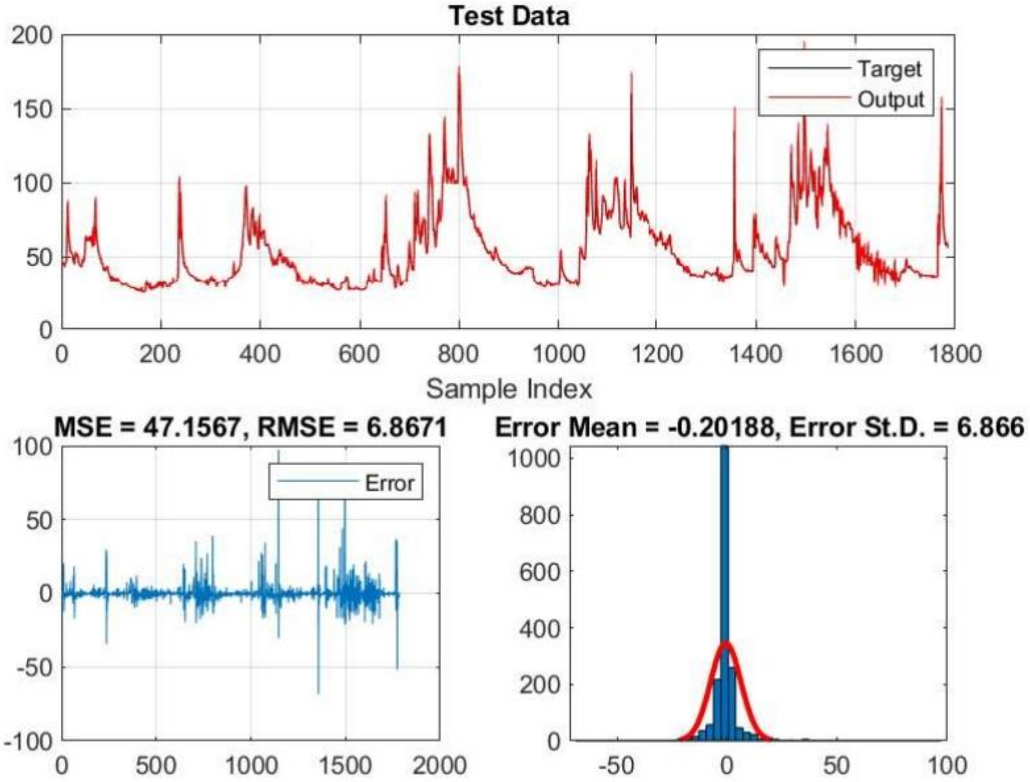


Figure 4.26: Test Performances of 1826 Station's using ANFIS-BP Model with Trapmf (Linear 2 Lag 4)

Table 4.14: Training and Test Performances of 1826 Station's using ANFIS-HB Model with Grid Partition Method

				Scenario											
Models			Data	LAG 3				LAG 4				LAG 5			
				RMSE	**MAE**	**MAPE**	**R^2**	**RMSE**	**MAE**	**MAPE**	**R^2**	**RMSE**	**MAE**	**MAPE**	**R^2**
ANFIS-HB	Gaussmf	Constant(2)	Train	10.0272	3.2091	4.0840	0.9124	9.8492	3.2290	4.1319	0.9155	9.7185	3.2394	4.1726	0.9178
			Test	**6.7681**	3.1216	5.1575	0.9257	6.8832	3.2147	5.4150	0.9155	6.8022	3.2109	5.3747	0.9251
		Constant(3)	Train	9.4978	3.1892	4.0521	0.9214	8.9274	3.0894	4.1202	0.9306	9.8401	3.2551	4.3177	0.9157
			Test	7.0117	3.1923	5.2868	0.9213	7.7309	3.3164	5.6187	0.9053	7.0282	3.2981	5.5487	0.9189
		Linear(2)	Train	9.3704	3.1472	4.1292	0.9235	8.8413	3.0340	4.0615	0.9319	8.9561	3.1333	4.3185	0.9302
			Test	7.2348	3.3318	5.6520	0.9169	7.1146	3.3191	5.6439	0.9179	7.1958	3.5212	6.1119	0.9162
		Linear(3)	Train	8.7499	2.9873	4.0778	**0.9333**	8.6109	2.8141	3.7683	0.9354	-	-	-	-
			Test	7.2214	3.3372	5.7363	0.9170	6.9186	3.1710	5.2571	0.9218	-	-	-	-
	Gbellmf	Constant(2)	Train	10.2858	3.3981	4.3739	0.9079	10.1358	3.3786	4.3967	0.9105	9.9460	3.2943	4.3122	0.9139
			Test	6.8418	3.3064	5.5776	0.9238	6.8723	3.3463	5.7084	0.9231	6.8328	3.2766	5.5666	0.9244
		Constant(3)	Train	9.6541	3.1587	3.9836	0.9188	9.3056	3.1033	3.9842	0.9246	9.4456	3.1067	4.0148	0.9223
			Test	6.8927	3.1284	5.1364	0.9235	6.9072	3.1461	5.2255	0.9224	6.8292	3.1600	5.2525	0.9240
		Linear(2)	Train	9.6713	3.1929	4.1855	0.9185	9.2727	3.1448	4.1452	0.9251	8.8729	3.0474	4.1389	**0.9315**
			Test	6.9712	3.3205	5.7118	0.9229	6.9326	3.2950	5.6111	0.9222	7.1601	3.4101	5.8149	0.9170
		Linear(3)	Train	8.8538	**2.9719**	3.9057	0.9317	8.5795	**2.8629**	**3.8880**	**0.9359**	-	-	-	-
			Test	6.8098	3.1694	5.2860	0.9246	7.0793	3.3019	5.5117	0.9187	-	-	-	-

Table 4.14: (continued)

ANFIS-HB	Trimf	Constant(2)	Train	10.0173	3.2234	4.1046	0.9126	9.8734	3.2334	4.1581	0.9151	9.8355	3.2088	4.1193	0.9158
			Test	6.8004	3.1731	5.3134	0.9252	6.8799	3.2579	5.5321	0.9236	6.8367	3.2280	5.4918	0.9244
		Constant(3)	Train	9.8084	3.2039	4.1444	0.9162	9.6772	3.2090	4.2375	0.9185	9.8316	3.1658	4.0772	0.9158
			Test	6.9014	3.2313	5.4385	0.9231	6.9322	3.3392	5.7763	0.9224	**6.8012**	3.1730	5.2164	0.9246
		Linear(2)	Train	9.8521	3.2036	4.1314	0.9155	9.5878	3.2008	4.1690	0.9200	9.3480	3.1544	4.1373	0.9239
			Test	6.9538	3.2755	5.5645	0.9224	7.0254	3.3271	5.7324	0.9205	6.8401	3.2917	5.5783	0.9240
		Linear(3)	Train	9.6471	3.0835	**4.0208**	0.9189	9.3560	3.1196	4.1193	0.9238	-	-	-	-
			Test	7.0031	3.2555	5.4974	0.9215	7.0166	3.2966	5.5683	0.9204	-	-	-	-
	Trapmf	Constant(2)	Train	10.1073	3.3699	4.3576	0.9110	10.0618	3.3146	4.2128	0.9118	9.8621	3.3189	4.3205	0.9153
			Test	6.9506	3.3459	5.6244	0.9219	**6.8692**	3.2556	5.3522	0.9238	6.8490	3.2884	5.5847	0.9240
		Constant(3)	Train	9.7756	3.2375	4.1659	0.9168	9.6857	3.4092	4.5847	0.9183	16.8435	11.6432	22.1898	0.7530
			Test	6.8802	3.1505	5.1611	0.9235	6.9133	3.3808	5.7238	0.9229	16.0066	13.0940	30.1003	0.5933
		Linear(2)	Train	9.7479	3.1578	4.0406	0.9172	9.7311	3.0955	3.9561	0.9175	9.6145	**3.0190**	**3.9076**	0.9195
			Test	6.9432	3.2062	5.3602	0.9221	6.9365	3.1536	5.1885	0.9223	7.0194	3.1973	5.2230	0.9206
		Linear(3)	Train	9.3403	3.1580	4.1600	0.9240	9.0622	3.1488	4.1990	0.9285	-	-	-	-
			Test	7.1059	3.3527	5.7038	0.9195	7.0711	3.4063	5.8063	0.9190	-	-	-	-

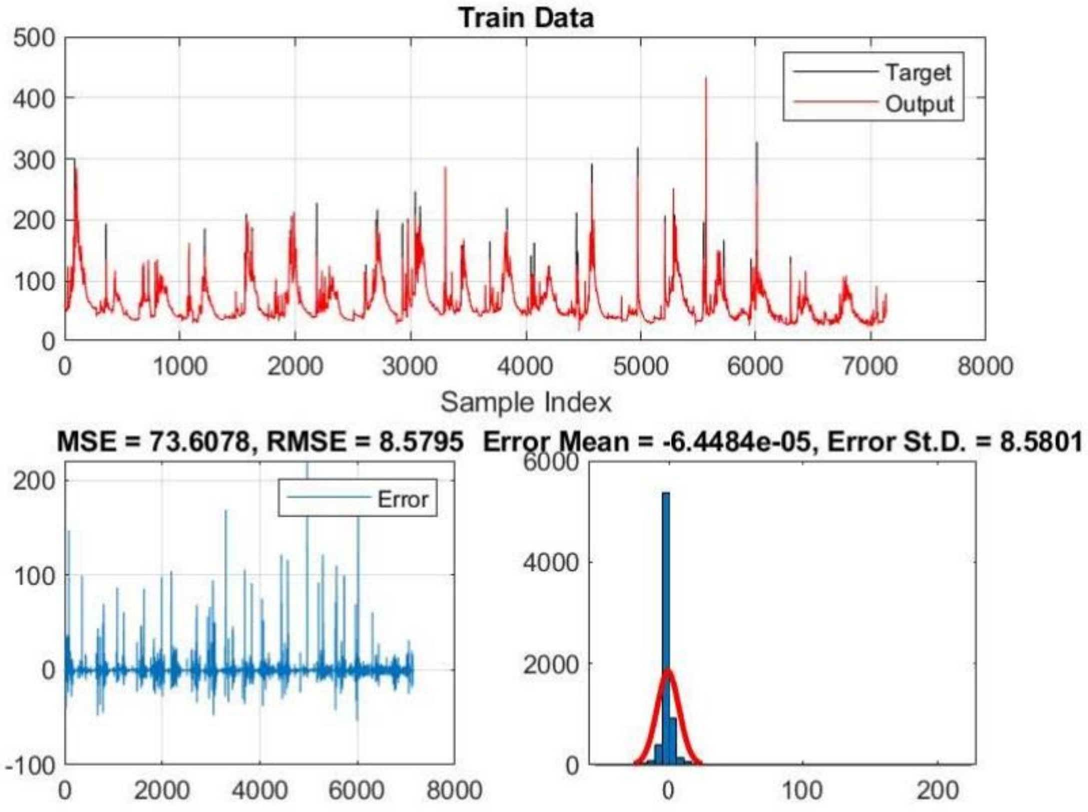


Figure 4.27: Training Performances of 1826 Station's using ANFIS-HB Model with Gbellmf (Linear 3 Lag 4)

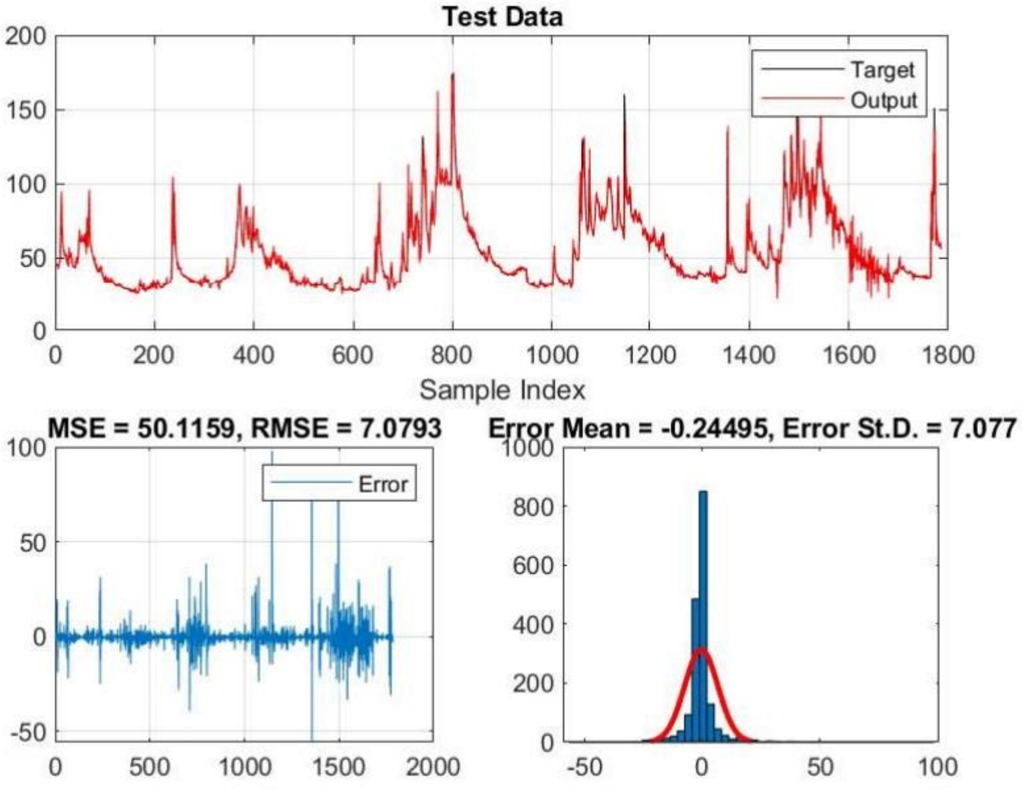


Figure 4.28: Test Performances of 1826 Station's using ANFIS-HB Model with Gbellmf (Linear 3 Lag 4)

Table 4.15: Training and Test Performances of 1826 Station's using ANFIS-SA Model with Grid Partition Method

Models			Data	Scenario											
				LAG 3				LAG 4				LAG 5			
				RMSE	**MAE**	**MAPE**	**R^2**	**RMSE**	**MAE**	**MAPE**	**R^2**	**RMSE**	**MAE**	**MAPE**	**R^2**
ANFIS-SA	Gaussmf	Constant(2)	Train	10.7896	3.9847	5.6522	0.8986	10.6752	3.8277	5.4806	0.9008	11.2273	4.2775	6.1332	0.8903
			Test	7.2834	3.9956	7.6353	0.9137	7.1335	3.8932	7.3289	0.9174	7.7439	4.2029	7.7213	0.9013
		Constant(3)	Train	10.8340	4.5047	6.7328	0.8979	10.5970	3.7209	5.1885	0.9023	10.3861	3.7042	4.9955	0.9061
			Test	7.7566	4.5549	9.0141	0.9041	7.2587	3.7771	6.8392	0.9138	7.1072	3.6361	6.5138	0.9179
		Linear(2)	Train	10.4040	3.5533	**4.6516**	0.9058	10.7945	3.7229	**4.8127**	0.8986	10.6046	3.7057	5.0291	0.9022
			Test	**6.7797**	**3.2815**	5.5320	**0.9247**	**7.0419**	**3.4336**	5.7305	0.9185	6.9920	3.5940	6.4737	0.9199
		Linear(3)	Train	12.1405	4.4320	5.7398	0.8717	13.7800	5.2818	7.2132	0.8349	11.0318	4.0612	5.2376	0.8941
			Test	7.6915	3.9874	6.6725	0.9040	8.8964	4.9219	8.9299	0.8721	7.5084	3.7241	6.1036	0.9067
	Gbellmf	Constant(2)	Train	11.1627	4.1261	5.5885	0.8915	10.9165	4.2682	6.3374	0.8963	10.8840	4.2411	6.1646	0.8971
			Test	7.6400	3.9446	6.8519	0.9045	7.6813	4.4346	8.6308	0.9045	7.4962	4.3076	8.3437	0.9099
		Constant(3)	Train	11.1199	4.2786	5.8411	0.8923	10.5570	3.7513	5.2915	0.9030	9.9875	3.5148	**4.5869**	0.9132
			Test	7.6242	3.9756	6.8480	0.9065	7.3318	3.8264	6.8961	0.9118	**6.9718**	**3.3823**	5.6060	**0.9219**
		Linear(2)	Train	11.3634	4.1845	5.9135	0.8876	11.0931	4.0480	5.4301	0.8940	15.2180	6.5486	9.6490	0.7987
			Test	7.5160	4.0963	7.6344	0.9080	7.0717	3.7406	6.6014	**0.9186**	10.4936	6.3731	12.2685	0.8230
		Linear(3)	Train	11.6472	5.2100	8.3228	0.8826	12.2685	4.9559	6.8565	0.8690	-	-	-	-
			Test	8.1683	5.3380	11.3890	0.8987	8.7529	4.8713	8.7303	0.8756	-	-	-	-

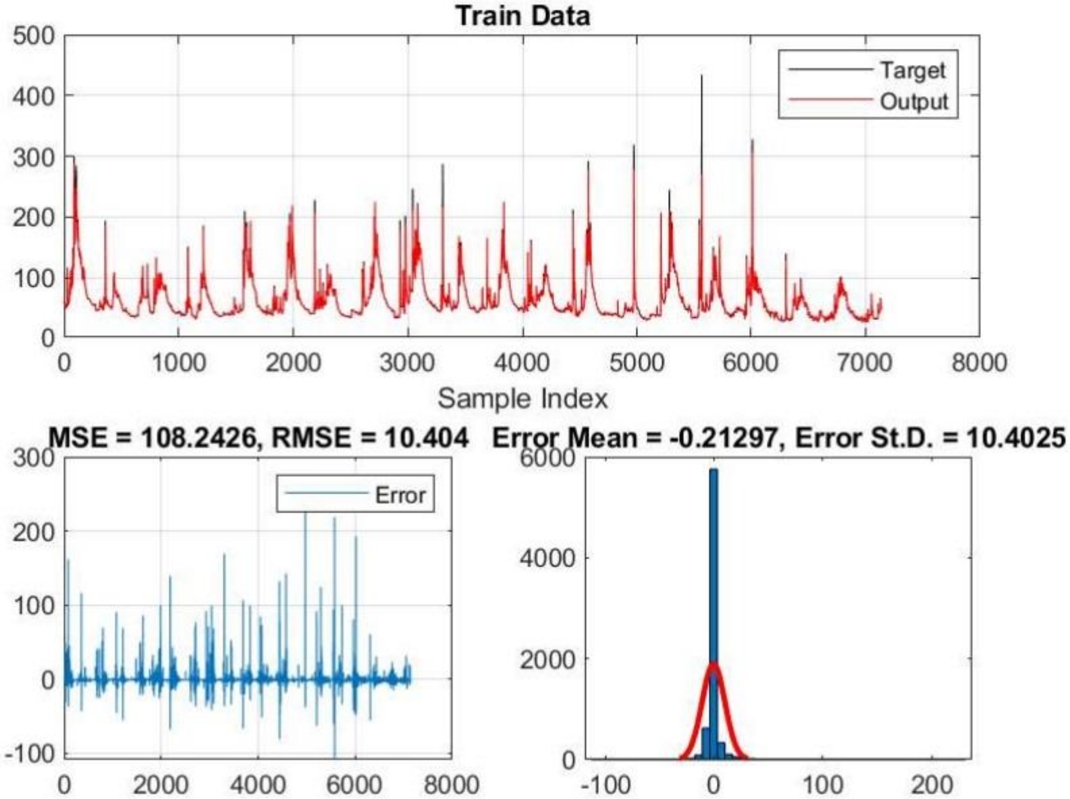


Figure 4.29: Training Performances of 1826 Station's using ANFIS-SA Model with Gaussmf (Linear 2 Lag 3)

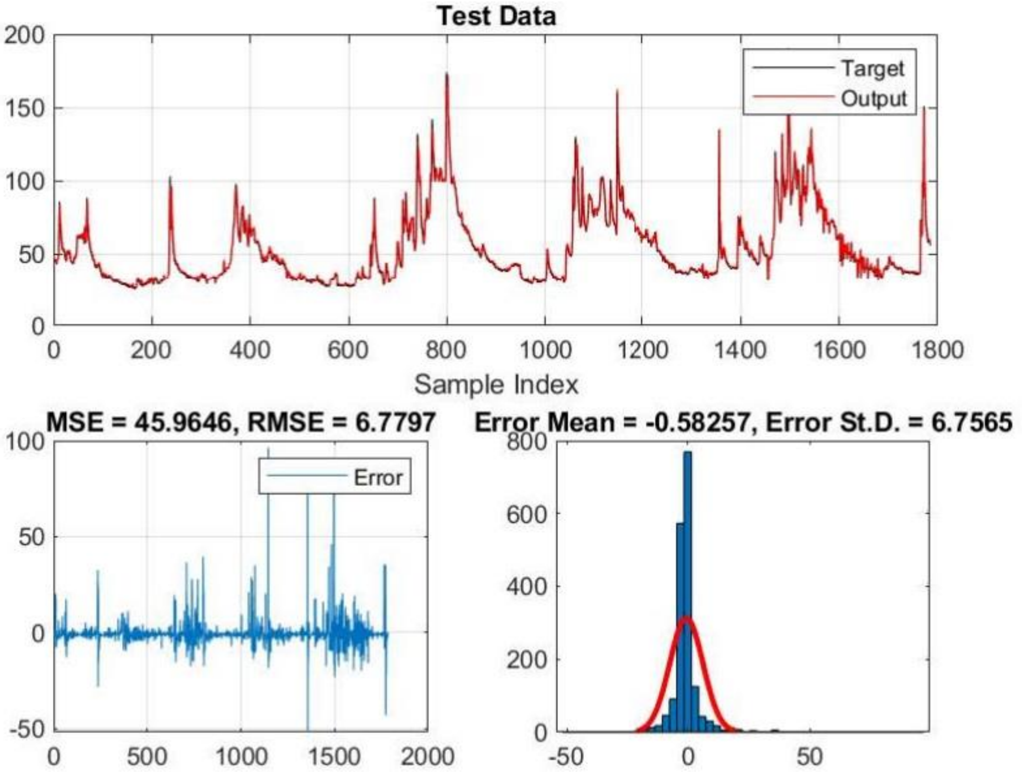


Figure 4.30: Test Performances of 1826 Station's using ANFIS-SA Model with Gaussmf (Linear 2 Lag 3)

Table 4.16: Training and Test Performances of 1827 Station's using ANFIS-BP Model with Grid Partition Method

Models			Data	Scenario											
				LAG 3				LAG 4				LAG 5			
				RMSE	**MAE**	**MAPE**	**R^2**	**RMSE**	**MAE**	**MAPE**	**R^2**	**RMSE**	**MAE**	**MAPE**	**R^2**
ANFIS-BP	Gaussmf	Constant(2)	Train	3.7138	1.7757	9.1472	0.9687	3.4370	1.6873	9.4389	0.9732	3.3982	1.6591	9.0284	0.9737
			Test	3.4837	1.9875	19.0317	0.9344	3.4945	2.0033	20.1676	0.9332	3.4831	1.9715	19.4107	0.9331
		Constant(3)	Train	3.3766	1.5849	7.2430	0.9741	3.3942	1.6084	7.5850	0.9738	31.5643	23.2222	88.1203	0.5494
			Test	3.3667	1.8336	14.6812	0.9375	3.3812	1.8602	15.4108	0.9370	21.8031	16.7535	83.3686	**0.7768**
		Linear(2)	Train	2.8883	1.2812	5.7183	0.9810	2.8658	1.2798	5.7469	**0.9813**	**2.8818**	**1.3001**	5.8372	0.9811
			Test	3.4689	1.8392	16.7589	0.9348	3.4170	1.8165	16.6574	0.9363	3.3904	1.8074	16.5768	0.9373
		Linear(3)	Train	**2.8579**	**1.2680**	**5.5985**	**0.9814**	2.8661	**1.2737**	**5.5988**	**0.9813**	4.8311	2.3239	9.5193	0.9487
			Test	3.3575	1.7564	15.1983	0.9385	3.3250	1.7377	15.0386	0.9395	4.3019	2.3224	16.4517	0.8991
	Gbellmf	Constant(2)	Train	4.2772	2.2444	12.1569	0.9589	4.1933	2.2359	12.3274	0.9604	6.4069	2.8082	12.7146	0.9086
			Test	3.8247	2.3460	21.3271	0.9195	3.8601	2.4021	22.2868	0.9178	5.1806	2.6647	24.3298	0.8519
		Constant(3)	Train	3.9726	1.8110	9.8202	0.9648	4.5865	1.8176	9.1957	0.9526	31.4054	22.9107	85.7227	0.5708
			Test	3.7580	2.0171	19.1646	0.9224	3.8426	2.0136	18.4431	0.9190	21.6235	16.4470	80.7734	0.7813
		Linear(2)	Train	2.8755	1.2906	5.8082	0.9812	**2.8655**	1.2799	5.7375	**0.9813**	2.8876	1.3087	5.8632	0.9810
			Test	3.4680	1.8473	16.9330	0.9349	3.4049	1.8115	16.6768	0.9367	3.3828	1.8062	16.6053	0.9376
		Linear(3)	Train	2.8718	1.2737	5.6243	0.9812	2.9016	1.2903	5.6525	0.9808	4.8796	2.3514	9.3380	0.9479
			Test	3.3729	1.7687	15.2681	0.9381	3.3498	1.7481	15.0197	0.9387	4.3143	2.3226	16.1328	0.8984

Table 4.16: (continued)

ANFIS-BP	Trimf	Constant(2)	Train	3.4737	1.6101	7.2188	0.9725	3.5594	1.5610	6.4790	0.9712	3.7083	1.6753	7.4435	0.9688
			Test	3.4455	1.8726	16.1685	0.9349	3.3499	1.8007	15.3775	0.9355	3.4953	1.8754	16.4712	0.9339
		Constant(3)	Train	3.5006	1.6365	7.9906	0.9722	3.7619	1.5020	6.5789	0.9680	31.6107	23.3543	88.8423	0.3808
			Test	3.4623	1.8941	17.2881	0.9340	3.3789	1.7897	15.0137	0.9372	21.8904	16.8813	84.1608	0.6799
		Linear(2)	Train	2.9262	1.3110	5.8700	0.9805	2.9054	1.3106	5.9069	0.9808	2.9206	1.3216	5.9457	0.9806
			Test	3.5227	1.8731	17.0923	0.9330	3.4842	1.8567	17.0649	0.9340	3.4656	1.8469	17.0356	0.9346
		Linear(3)	Train	2.8846	1.2808	5.6618	0.9810	2.8864	1.2873	5.6411	0.9810	4.7201	2.2582	9.6810	0.9504
			Test	3.3350	1.7525	15.4658	0.9393	3.3080	1.7419	15.3041	0.9402	4.3084	2.3530	16.9087	0.8976
	Trapmf	Constant(2)	Train	3.3964	1.5086	6.5645	0.9738	4.5910	1.8651	7.3925	0.9531	3.9058	1.6331	6.6429	0.9655
			Test	3.4613	1.8539	16.2900	0.9342	4.6098	2.1879	16.8792	0.8929	3.9216	1.9453	16.0729	0.9155
		Constant(3)	Train	5.2198	1.8501	8.3418	0.9387	7.5339	2.6317	11.2486	0.8786	30.8345	22.0085	79.5253	0.3926
			Test	3.5004	1.8824	16.9865	0.9327	6.4619	2.6371	20.2923	0.8097	21.0955	15.6294	74.6515	0.6110
		Linear(2)	Train	2.8872	1.2789	5.6595	0.9810	2.9031	1.2883	5.6233	0.9808	2.9425	1.3037	**5.6212**	0.9803
			Test	3.4430	1.8265	16.5063	0.9359	3.3838	1.7957	16.2359	0.9380	3.3635	1.7769	15.9806	0.9388
		Linear(3)	Train	2.9477	1.2965	5.6390	0.9802	3.0305	1.3355	5.6974	0.9791	5.9059	2.9649	10.3175	0.9222
			Test	3.2807	1.7463	14.8781	0.9415	3.2740	1.7347	14.5891	0.9413	5.0082	2.6466	16.5243	0.8624

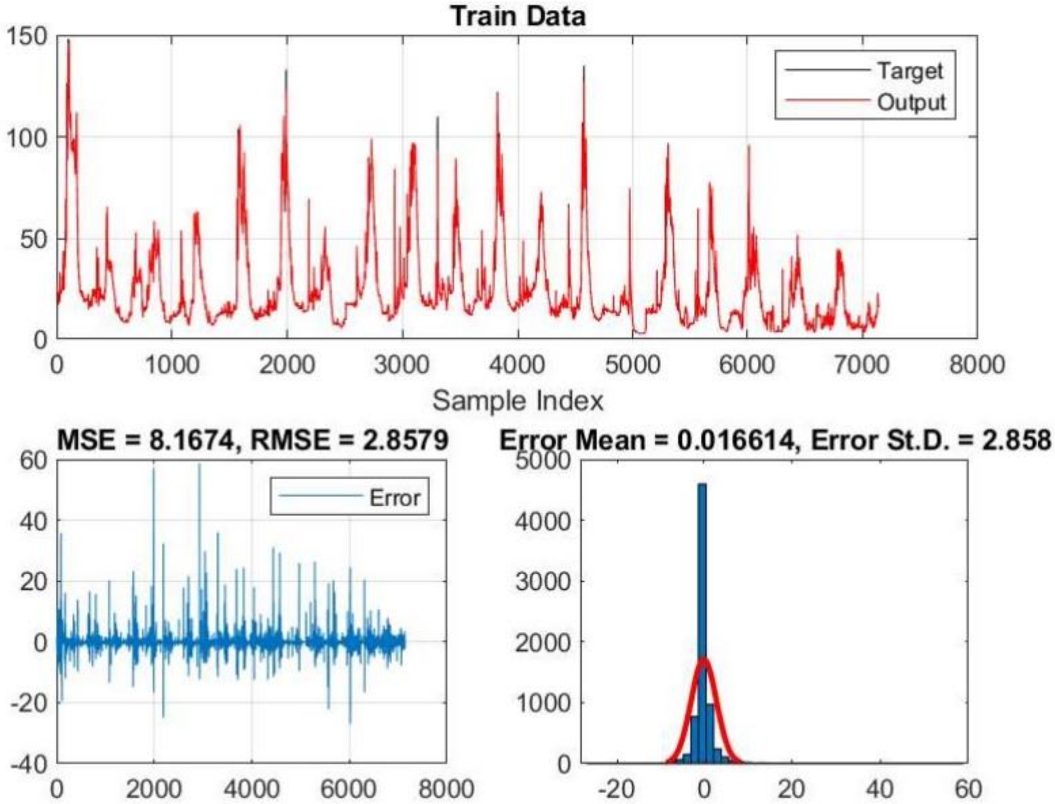


Figure 4.31: Training Performances of 1827 Station's using ANFIS-BP Model with Gaussmf (Linear 3 Lag 3)

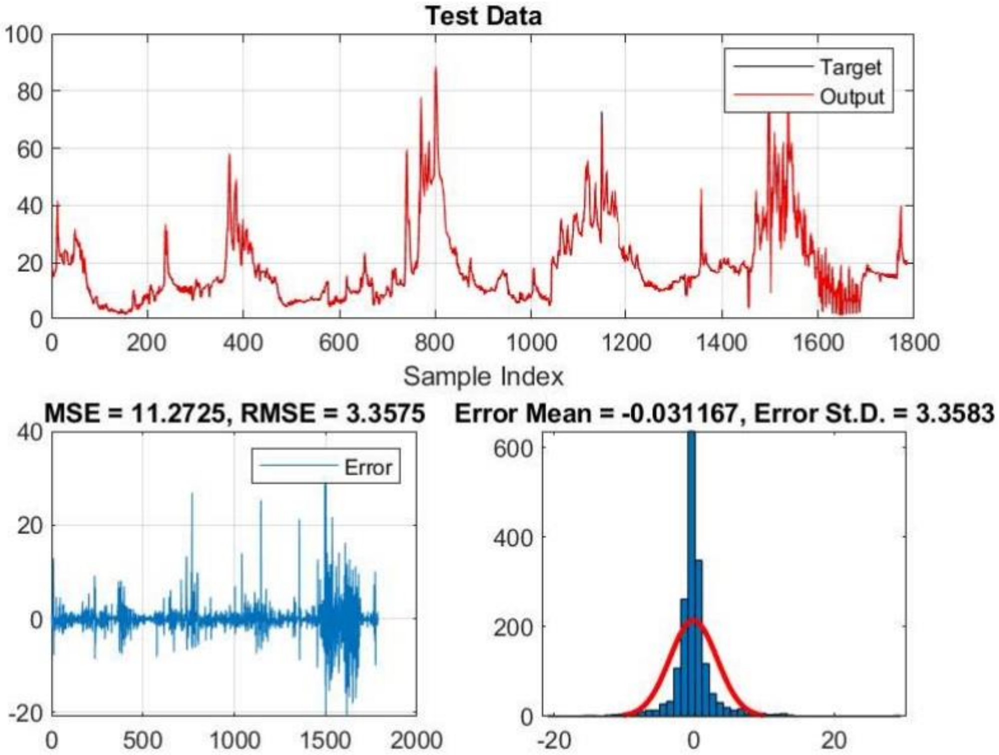


Figure 4.32: Test Performances of 1827 Station's using ANFIS-BP Model with Gaussmf (Linear 3 Lag 3)

Table 4.17: Training and Test Performances of 1827 Station's using ANFIS-HB Model with Grid Partition Method

Models			Data	Scenario											
				LAG 3				LAG 4				LAG 5			
				RMSE	**MAE**	**MAPE**	**R²**	**RMSE**	**MAE**	**MAPE**	**R²**	**RMSE**	**MAE**	**MAPE**	**R²**
ANFIS-HB	Gaussmf	Constant(2)	Train	2.8457	1.2782	5.6934	0.9816	2.8256	1.2750	5.7106	0.9818	2.7799	1.2782	5.7560	0.9824
			Test	3.4764	1.8027	15.9561	0.9341	3.4855	1.8243	16.1146	0.9340	4.0117	1.9076	16.6073	0.9154
		Constant(3)	Train	2.7452	1.2453	5.6435	0.9828	2.6826	1.2335	5.6492	0.9836	2.6616	1.2166	5.7284	0.9839
			Test	3.3767	1.7572	15.4978	0.9378	3.5159	1.7992	15.8974	0.9325	3.5451	1.7828	15.2505	0.9320
		Linear(2)	Train	2.7469	1.2580	5.7563	0.9828	2.6561	1.2280	5.6336	0.9839	2.5746	1.1936	**5.5785**	0.9849
			Test	3.3726	1.7669	15.4840	0.9379	3.6594	1.8553	15.8945	0.9267	5.0259	1.9643	16.4232	0.8715
		Linear(3)	Train	**2.6077**	**1.2126**	5.6693	**0.9845**	**2.4721**	1.1482	5.4729	**0.9861**	-	-	-	-
			Test	3.9308	1.9100	16.1152	0.9187	4.3130	1.9412	16.6890	0.9004	-	-	-	-
	Gbellmf	Constant(2)	Train	2.8925	1.2900	5.8165	0.9809	2.8303	1.2807	5.8217	0.9818	2.8097	1.2832	5.8132	0.9820
			Test	3.4497	1.8172	16.3804	0.9352	3.4202	1.8107	16.0564	0.9364	3.5526	1.8646	17.6399	0.9321
		Constant(3)	Train	2.8017	1.2764	5.9312	0.9821	2.7849	1.2648	5.7654	0.9823	2.7076	1.2409	5.8540	0.9833
			Test	3.4311	1.7861	15.8325	0.9360	3.5127	1.7524	14.9275	0.9328	3.6071	1.8033	15.3363	0.9296
		Linear(2)	Train	2.7545	1.2498	5.6619	0.9827	2.6686	1.2317	5.6595	0.9838	**2.5642**	**1.2001**	5.6358	**0.9850**
			Test	3.4999	1.8061	15.8590	0.9333	3.4973	1.8165	15.8476	0.9337	4.3858	1.9344	16.2082	0.8997
		Linear(3)	Train	2.6481	1.2151	5.9555	0.9840	2.4735	**1.1430**	**5.4655**	**0.9861**	-	-	-	-
			Test	3.6575	1.8455	16.1506	0.9289	6.3536	2.1568	17.2403	0.8079	-	-	-	-

Table 4.17: (continued)

ANFIS-HB	Trimf	Constant(2)	Train	2.8522	1.2882	5.7885	0.9815	2.8470	1.2930	5.9043	0.9815	2.8668	1.2820	5.7496	0.9813
			Test	3.5809	1.8604	16.5476	0.9305	3.5613	1.8691	16.7411	0.9314	3.4158	1.8013	16.2391	0.9366
		Constant(3)	Train	2.7782	1.2743	5.7693	0.9824	2.8031	1.2579	5.7451	0.9834	2.6537	1.2377	5.7134	0.9840
			Test	3.5518	1.8379	16.1850	0.9317	3.5501	1.8301	16.2143	0.9315	3.4676	1.8083	15.9272	0.9345
		Linear(2)	Train	2.8282	1.2837	5.8968	0.9818	2.7216	1.2654	5.8450	0.9831	2.6743	1.2533	5.8212	0.9837
			Test	7.6222	2.5555	35.8883	0.7349	3.3541	1.8620	16.1423	0.9304	4.1098	1.9999	17.5531	0.9102
		Linear(3)	Train	2.6851	1.2311	5.6574	0.9836	2.5496	1.1854	5.6167	0.9852	-	-	-	-
			Test	3.9221	1.9786	20.8894	0.9194	21.3012	4.7136	72.6897	0.2383	-	-	-	-
	Trapmf	Constant(2)	Train	2.8677	1.2992	5.8829	0.9813	2.8459	1.2997	5.7964	0.9816	2.8391	1.2898	5.7176	0.9816
			Test	3.6042	1.8741	16.7686	0.9295	3.3794	1.7881	15.4881	0.9379	3.3805	1.7707	15.1345	0.9381
		Constant(3)	Train	2.7689	1.2737	5.6910	0.9825	6.3651	4.5170	34.1078	0.9077	6.2458	4.3913	33.4955	0.9112
			Test	3.3993	1.7604	15.2031	0.9374	6.5338	4.8785	57.2904	0.7784	6.6445	4.7936	56.4543	0.7700
		Linear(2)	Train	2.7820	1.2817	5.9558	0.9824	2.7127	1.2534	5.8699	0.9832	2.6115	1.2168	5.8043	0.9845
			Test	3.6306	1.9057	17.5672	0.9285	3.5924	1.8893	17.3210	0.9297	4.1334	2.0103	17.6014	0.9112
		Linear(3)	Train	2.7142	1.2278	**5.6162**	0.9832	2.6774	1.2036	5.5846	0.9837	-	-	-	-
			Test	3.4472	1.8078	16.0594	0.9355	3.8251	1.8843	16.4368	0.9216	-	-	-	-

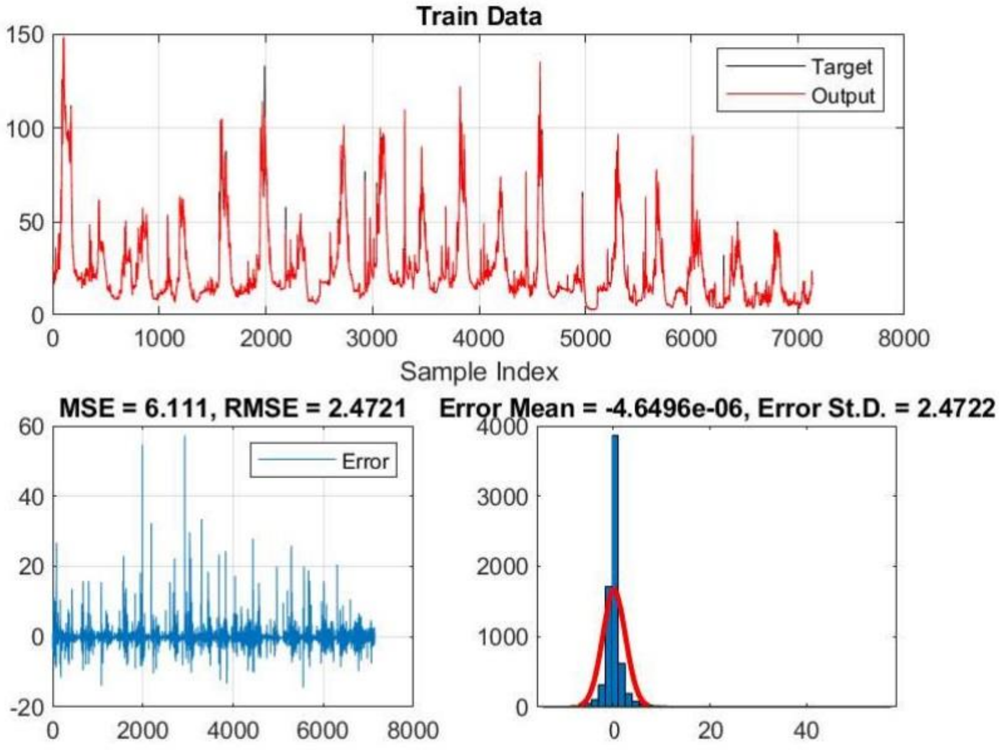


Figure 4.33: Training Performances of 1827 Station's using ANFIS-HB Model with Gaussmf (Linear 3 Lag 4)

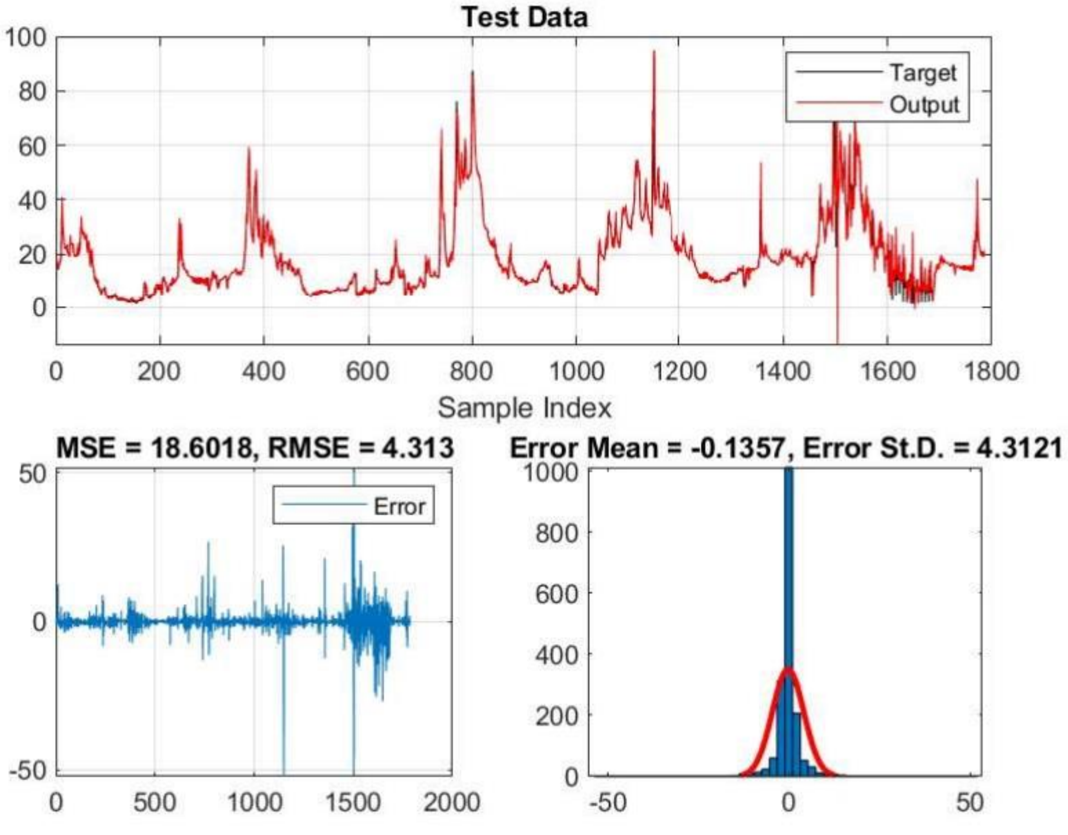


Figure 4.34: Test Performances of 1827 Station's using ANFIS-HB Model with Gaussmf (Linear 3 Lag 4)

Table 4.18: Training and Test Performances of 1827 Station's using ANFIS-SA Model with Grid Partition Method

Models			Data	Scenario											
				LAG 3				LAG 4				LAG 5			
				RMSE	**MAE**	**MAPE**	**R²**	**RMSE**	**MAE**	**MAPE**	**R²**	**RMSE**	**MAE**	**MAPE**	**R²**
ANFIS-SA	Gaussmf	Constant(2)	Train	3.7012	2.0905	15.7933	0.9692	3.6538	2.0012	11.5165	0.9697	3.7350	1.8293	9.9640	0.9683
			Test	3.8784	2.5716	30.6163	0.9213	3.7548	2.2976	21.6082	0.9226	3.5866	2.1003	19.7212	0.9298
		Constant(3)	Train	3.3746	1.7859	10.2532	0.9741	**3.1932**	1.5599	8.0831	**0.9768**	**3.3247**	1.7429	11.0719	**0.9749**
			Test	3.4984	2.1038	20.3464	0.9327	3.3750	1.9057	16.7923	0.9377	3.5593	2.1072	22.8747	0.9312
		Linear(2)	Train	3.3059	1.4916	6.8712	0.9751	3.6551	1.8743	9.9693	0.9696	3.3614	**1.5890**	**7.3602**	0.9743
			Test	3.3615	1.7987	15.2511	0.9380	3.6044	2.1392	18.5133	0.9288	4.1465	1.9544	18.4001	0.9069
		Linear(3)	Train	4.4586	2.1182	11.4508	0.9551	3.2072	**1.5373**	**6.9712**	0.9767	-	-	-	-
			Test	4.2362	2.3613	22.4224	0.9031	3.3764	1.8495	15.8121	0.9373	-	-	-	-
	Gbellmf	Constant(2)	Train	3.3207	1.5951	8.0065	0.9749	3.5625	1.8346	9.1976	0.9711	3.7717	1.8621	9.2381	0.9676
			Test	3.4675	1.9412	16.6116	0.9339	3.6376	2.1049	17.7220	0.9276	3.6051	2.0692	18.2401	0.9288
		Constant(3)	Train	**3.2526**	1.5771	8.0056	**0.9759**	3.3233	1.5633	7.8553	0.9749	3.5579	1.7629	8.6457	0.9712
			Test	3.3645	1.9193	16.4251	0.9380	3.3429	1.8548	17.8659	0.9386	3.7925	2.0960	18.2165	0.9215
		Linear(2)	Train	3.2820	**1.5142**	**6.8644**	0.9755	3.4978	1.7796	9.4190	0.9747	4.3546	2.1637	9.9313	0.9571
			Test	3.3721	1.8130	15.3792	0.9372	3.5097	2.0393	18.9453	0.9343	4.1634	2.3676	17.6924	0.9074
		Linear(3)	Train	4.5446	2.3245	10.2761	0.9532	4.8250	2.5389	11.5313	0.9472	-	-	-	-
			Test	4.4936	2.4891	18.0480	0.8889	34.4706	5.8830	89.3168	0.0723	-	-	-	-

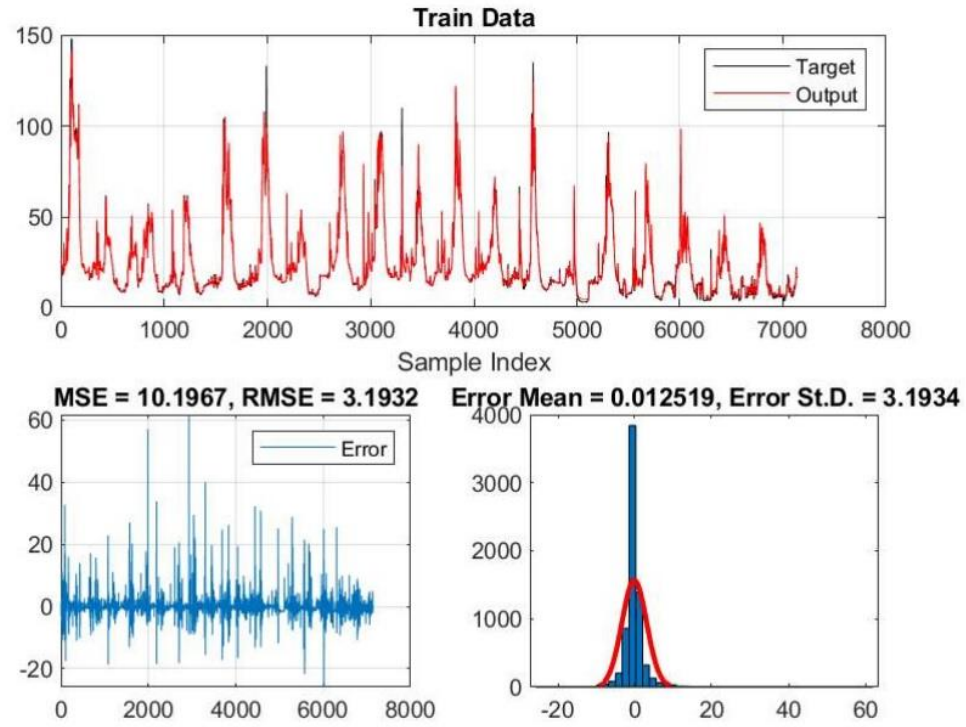


Figure 4.35: Training Performances of 1827 Station's using ANFIS-SA Model with Gaussmf (Constant 3 Lag 4)

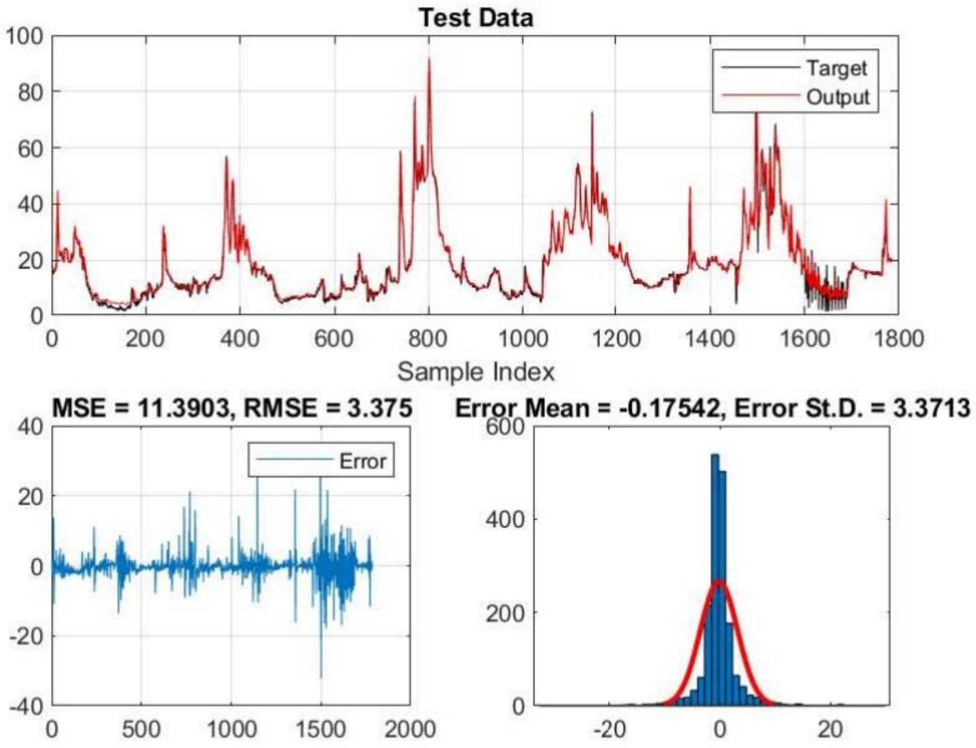


Figure 4.36: Test Performances of 1827 Station's using ANFIS-SA Model with Gaussmf (Constant 3 Lag 4)

Table 4.19: Comparison of 1825 Station's average results with clustering methods

Model	Method	Scenario											
		LAG 3				LAG 4				LAG 5			
		RMSE$_{(Avg)}$	**MAE**$_{(Avg)}$	**MAPE**$_{(Avg)}$	**R^2**$_{(Avg)}$	**RMSE**$_{(Avg)}$	**MAE**$_{(Avg)}$	**MAPE**$_{(Avg)}$	**R^2**$_{(Avg)}$	**RMSE**$_{(Avg)}$	**MAE**$_{(Avg)}$	**MAPE**$_{(Avg)}$	**R^2**$_{(Avg)}$
ANFIS-BP	FCM	**4.9357**	1.3898	20.2173	**0.6797**	**4.9777**	1.4102	20.5461	**0.6747**	**5.0024**	1.4227	20.4194	**0.6727**
ANFIS-BP	Grid Partition	5.0478	1.4622	19.6128	0,6659	5.0551	1.4618	19.3855	0.6646	5.5832	1.8483	24.6312	0.6261
ANFIS-HB	FCM	5.0451	**1.3261**	**13.2492**	0.6714	5.0977	**1.3599**	**13.7035**	0.6668	5.0474	**1.3417**	**13.1113**	0.6703
ANFIS-HB	Grid Partition	5.6304	1.3950	15.3566	0.6441	7.9136	1.4952	15.4196	0.5535	14.6303	2.0764	32.3655	0.5701
ANFIS-SA	FCM	8.1794	1.4592	15.2957	0.5808	5.6825	1.4038	17.1609	0.6159	6.2486	1.4607	19.4916	0.5982
ANFIS-SA	Grid Partition	5.1363	1.8071	33.3213	0.6557	5.0968	1.8516	38.4421	0.6620	5.0981	1.8213	39.0915	0.6630

Table 4.20: Comparison of 1826 Station's average results with clustering methods

Model	Method	Scenario											
		LAG 3				LAG 4				LAG 5			
		RMSE$_{(Avg)}$	**MAE**$_{(Avg)}$	**MAPE**$_{(Avg)}$	**R^2**$_{(Avg)}$	**RMSE**$_{(Avg)}$	**MAE**$_{(Avg)}$	**MAPE**$_{(Avg)}$	**R^2**$_{(Avg)}$	**RMSE**$_{(Avg)}$	**MAE**$_{(Avg)}$	**MAPE**$_{(Avg)}$	**R^2**$_{(Avg)}$
ANFIS-BP	FCM	6.9623	3.2429	5.4496	0.9213	**6.9503**	**3.2483**	**5.4670**	**0.9213**	**6.9388**	**3.2373**	**5.4316**	**0.9215**
ANFIS-BP	Grid Partition	8.2652	4.2660	7.8903	0.8780	8.3821	4.4326	7.8912	0.8729	19.2675	14.6931	26.6995	0,8593
ANFIS-HB	FCM	7.0925	3.2676	5.9602	0.9191	7.3198	3.3091	5.4977	0.9139	7.1753	3.3410	5.5570	0.9170
ANFIS-HB	Grid Partition	**6.9556**	3.2436	5.4505	**0.9222**	7.0052	3.2830	5.5420	0.9201	7.6834	4.0958	7.5720	0.8947
ANFIS-SA	FCM	7.0685	**3.2114**	**5.3565**	0.9190	7.1511	3.2864	5.5232	0.9169	8.8353	3.3322	5.5786	0.8736
ANFIS-SA	Grid Partition	7.5575	4.1467	7.6972	0.9080	7.6460	4.1123	7.4609	0.9040	7.7590	4.1743	7.5758	0.9001

Table 4.21: Comparison of 1827 Station's average results with clustering methods

Model	Method	Scenario											
		LAG 3				LAG 4				LAG 5			
		$\mathbf{RMSE}_{(Avg)}$	$\mathbf{MAE}_{(Avg)}$	$\mathbf{MAPE}_{(Avg)}$	$\mathbf{R^2}_{(Avg)}$	$\mathbf{RMSE}_{(Avg)}$	$\mathbf{MAE}_{(Avg)}$	$\mathbf{MAPE}_{(Avg)}$	$\mathbf{R^2}_{(Avg)}$	$\mathbf{RMSE}_{(Avg)}$	$\mathbf{MAE}_{(Avg)}$	$\mathbf{MAPE}_{(Avg)}$	$\mathbf{R^2}_{(Avg)}$
ANFIS-BP	FCM	3.5040	**1.8552**	**16.7194**	0.9334	3.4997	1.8543	16.7062	0.9335	3.5112	1.8584	16.7470	0.9331
ANFIS-BP	Grid Partition	**3.4720**	1.8811	16.8149	**0.9341**	3.7079	1.9336	16.9036	0.9242	**3.3768**	5.6906	33.2159	0.8618
ANFIS-HB	FCM	14.8422	2.0723	20.7491	0.8321	**3.4647**	**1.7897**	15.4549	**0.9347**	3.4590	**1.8009**	**15.4923**	**0.9348**
ANFIS-HB	Grid Partition	3.8096	1.8795	17.6094	0.9193	5.0534	2.2295	**11.2016**	0.8690	4.1067	2.1201	19.6979	0.9067
ANFIS-SA	FCM	4.7741	1.9665	17.5554	0.8791	32.3268	2.6673	19.6340	0.6690	37.5008	2.7641	18.6014	0.8172
ANFIS-SA	Grid Partition	3.7453	2.1248	19.3875	0.9241	7.3839	2.5093	27.0720	0.8249	3.8089	2.1158	19.1908	0.9209

Table 4.22: Comparison of 1825 Station's all average results

Model	Scenario											
	LAG 3				LAG 4				LAG 5			
	$\mathbf{RMSE}_{(Avg)}$	$\mathbf{MAE}_{(Avg)}$	$\mathbf{MAPE}_{(Avg)}$	$\mathbf{R^2}_{(Avg)}$	$\mathbf{RMSE}_{(Avg)}$	$\mathbf{MAE}_{(Avg)}$	$\mathbf{MAPE}_{(Avg)}$	$\mathbf{R^2}_{(Avg)}$	$\mathbf{RMSE}_{(Avg)}$	$\mathbf{MAE}_{(Avg)}$	$\mathbf{MAPE}_{(Avg)}$	$\mathbf{R^2}_{(Avg)}$
ANFIS-BP	**4.9918**	1.4260	19.9151	**0.6728**	**5.0164**	1.4360	19.9658	**0.6697**	**5.2928**	**1.6355**	**22.5253**	**0,6494**
ANFIS-HB	5.3378	**1.3606**	**14.3029**	0.6578	6.5057	**1.4276**	**14.5616**	0.6102	9.8389	1.7091	22.7384	0.6202
ANFIS-SA	6,6579	1.6332	24.3085	0.6183	5.3897	1.6277	27.8015	0.6390	5.6734	1.6410	29.2916	0.6306

Table 5 Comparison of 1826 Station's average results

	Scenario											
Model	LAG 3				LAG 4				LAG 5			
	RMSE$_{(Avg)}$	**MAE**$_{(Avg)}$	**MAPE**$_{(Avg)}$	**R^2**$_{(Avg)}$	**RMSE**$_{(Avg)}$	**MAE**$_{(Avg)}$	**MAPE**$_{(Avg)}$	**R^2**$_{(Avg)}$	**RMSE**$_{(Avg)}$	**MAE**$_{(Avg)}$	**MAPE**$_{(Avg)}$	**R^2**$_{(Avg)}$
ANFIS-BP	7.6138	3.7545	6.6670	0.8997	7.6662	3.8404	6.6791	0.8971	13.1032	8.9652	16.0655	0.8904
ANFIS-HB	**7.0241**	**3.2556**	**5.7054**	**0.9207**	**7.1625**	**3.2961**	**5.5199**	**0.9170**	**7.4294**	**3.7184**	**6.5645**	**0.9059**
ANFIS-SA	7.3130	3.6791	6.5269	0.9135	7.3986	3.6994	6.4921	0.9105	8.2972	3.7533	6.5772	0.8869

Table 4.24: Comparison of 1827 Station's average results

	Scenario											
Model	LAG 3				LAG 4				LAG 5			
	RMSE$_{(Avg)}$	**MAE**$_{(Avg)}$	**MAPE**$_{(Avg)}$	**R^2**$_{(Avg)}$	**RMSE**$_{(Avg)}$	**MAE**$_{(Avg)}$	**MAPE**$_{(Avg)}$	**R^2**$_{(Avg)}$	**RMSE**$_{(Avg)}$	**MAE**$_{(Avg)}$	**MAPE**$_{(Avg)}$	**R^2**$_{(Avg)}$
ANFIS-BP	**3.4880**	**1.8681**	**16.7672**	**0.9338**	**3.6038**	**1.8939**	16.8049	**0.9288**	5.9440	3.7745	24.9815	0.8975
ANFIS-HB	9.3259	1.9759	19.1793	0.8757	4.2591	2.0096	**13.3283**	0.9019	**3.7828**	**1.9605**	**17.5951**	**0.9208**
ANFIS-SA	4.2597	2.0457	18.4715	0.9016	19.8554	2.5883	23.3530	0.7470	20.6549	2.4400	18,8961	0.8691

While choosing the most suitable model, RMSE, MAE, MAPE and R² values were taken into consideration. In the trials for the models, during the training and testing stages, the model with the smallest RMSE, MAE, MAPE, and the largest R² value were selected as the most appropriate. All models used in the datasets were applied using FCM and GP clustering methods with Lag 3,4 and 5 values, respectively. In the FCM clustering method, input values between 2 and 10 were applied, while in the GP clustering method, 4 different input parameters belonging to 4 membership functions were applied 134 times for each station, 304 times in total, and measured with 1000 iterations under the same conditions. In the ANFIS-SA model, unlike other models, the results with 2 membership functions, gaussmf and gbellmf, are listed in Tables 4.12, 4.15 and 4.18 in the GP clustering method. The input parameters used in the ANFIS-SA model was not in harmony with the trimf and trapmf membership functions.

When the test values of all models were compared, when the FCM-4 clustering method of the ANFIS-BP model was applied, the best RMSE value was obtained with 4.795, while the Lag value was 3 at the first station, Eğribük (1825). The overlap of the values in the training and test data set and the predicted values of the ANFIS-BP model is shown graphically in Figures 4.1 and 4.2, respectively. “Black lines” represent the actual values of the training and test data, while the “red lines” represent the predicted values of the ANFIS model. The overlap of two symbols shows how effectively the established model reflects the decision-making logic. Accordingly, it can be said that the values of the training and test data and the ANFIS estimation values are quite close to each other. The best MAE and MAPE test values of the same station were obtained as 1.2403 and 10.7278, respectively, by running the Gbellmf membership function in the GP clustering method of the ANFIS-BP model with the Linear 3 parameter. Finally, the highest R^2 value was obtained by running the Gbellmf membership function of the ANFIS-HB model with the Constant 2 parameter, with a value of 0.6925. The best values for this station are given in Tables 4.1, 4.10 and 4.11. The lowest RMSE test value of the second station (Ergenuşağı-1826) was 6.7476, and the Lag 3 value of the ANFIS-SA model was reached with the FCM-2 clustering method. The overlap of the values for RMSE values in the training and test data set and the predicted values of the ANFIS-SA model is shown graphically in Figures 4.11 and 4.12, respectively. The lowest MAE value was 3.1216 with the ANFIS-HB model and the Constant 2 parameter of the gaussmf membership function. The lowest MAPE value of 5.0902 was found in the FCM-10 scenario of the ANFIS-SA model. Finally, in the FCM-2 scenario made with the ANFIS-SA model, the R^2 value was found to be 0.9264, which is very close to the value of 1. At the last station, Değirmenocağı (1827), when the ANFIS-BP model was run with the Trapmf membership function Linear 3 parameter, the lowest RMSE value of 3.2807, the lowest MAE with 1.7463 and the highest R^2 test value with 0.9415, which is most close to the value of 1, these values are given in Table 4.16. The overlap of the values for RMSE values in the training and test data set and the predicted values of the ANFIS-BP model is shown graphically in Figures 4.31 and 4.32, respectively. With the Gaussmf function Constant 3 parameter of the same model, the lowest MAPE value was obtained with the value of 14.6812.

Tables 4.19, 4.20, and 4.21 report the error measurements associated with each model. While creating the tables, the test values of the results obtained in two different clustering methods were averaged and the values were listed under their own Lag values. When the first station results were examined, the lowest test value of 4.9357 $RMSE_{(Avg)}$ and 0.6797 $R^2_{(Avg)}$ were calculated as the highest test average value in the FCM clustering method of model ANFIS-BP, and based on these statistics, model ANFIS-BP model can be selected by preferring ANFIS-HB and ANFIS-SA models. Based on $MAE_{(Avg)}$ or $MAPE_{(Avg)}$ statistics, the model with average test values of 1.3261 and 13.1113, respectively, can be selected by preferring other models in the ANFIS-HB model with FCM clustering method. Model ANFIS-SA appears to be relatively weak from all other models based on error statistics, but it has also been observed to find values close to the results of other models. When the second station results were examined, the lowest test value of 6.9503 $RMSE_{(Avg)}$ were calculated in model ANFIS-BP with FCM clustering method and 0.9222 $R^2_{(Avg)}$ were calculated as the highest test average value in the GP clustering method of model ANFIS-HB. Based on $MAE_{(Avg)}$ or $MAPE_{(Avg)}$ statistics, the model with average test values of 3.2114 and 5.3565, respectively, can be selected by preferring other models in the ANFIS-SA with FCM clustering method. Model ANFIS-SA looks relatively weak when evaluated using most of the error statistics but performs very well relative to MAE and MAPE. Considering all the RMSE, MAE, MAPE and R^2 statistics, the ANFIS-BP model will be chosen as a preference to the ANFIS-SA model. When the last station results were examined, the lowest test value of 3.3768 $RMSE_{(Avg)}$ were calculated in model ANFIS-

BP with GP clustering method and 0.9348 $R^2_{(Avg)}$ were calculated as the highest test average value in the FCM clustering method of model ANFIS-HB. Based on $MAE_{(Avg)}$ or $MAPE_{(Avg)}$ statistics, the model with average test values of 1.7897 and 11.2016, respectively, can be selected by preferring other models in the ANFIS-SA. The average MAPE value obtained with the ANFIS-HB model gave a great advantage to the other models. When we checked the other error statistics, it was concluded that the ANFIS-BP and ANFIS-SA models also made good measurements like the ANFIS-HB model. Algorithms such as SA are iterative-based and usually try to find a global optimum by improving a randomly selected candidate solution. This kind of algorithm can generally obtain the optimal solution in a shorter time interval than population-based algorithms. For this reason, the performance of the SA algorithm, which we applied within the scope of the thesis, fell behind the BP and HB algorithms.

CHAPTER 5

CONCLUSION

Water has vital importance for the survival of living things. From the smallest living organism to the largest living being, it is water that sustains all biological life and all human activities. In order to sustain human life without any problems, a rational planning is required for the protection and use of existing water resources. At the beginning of the plans to be made, the potential created by the water source to be used and the potential to be made in the future should be determined. Therefore, river flow estimation is necessary to provide basic information on a wide variety of problems associated with the functioning of river systems. Behavior detection is possible with the observation values of existing measuring stations. Prospective estimation of river flow rates can be made with the help of time series. In this study, the daily flow values of Zamanti River-Değirmenocağı, Zamanti River-Ergenuşağı and Eğlence River-Eğribük stations in the Seyhan Basin in Turkey have been investigated for the flow rate values. In our modelling approaches, the models were being trained such that 80% of the data was used in the training part and 20% in the test part. The aim of this study is to compare the river flow estimation of 3 FMS with ANFIS approaches. Taking long-term daily river flow data as input to the ANFIS model, the model was trained using BP, HB, and SA.

When the results were examined, low estimation errors measured the usability of Artificial Intelligence and Deep Learning methods in river flow problems. ANFIS models are not always easy to interpret because of the rules they contain, in some cases having too many inputs. Water Flow in rivers is a multivariate nonlinear phenomenon. Therefore, it is very difficult to accurately determine all the parameters of the flow in rivers and to make parametric modelling. Prospective river flow estimates were created by reducing this complexity to facilitate interpretation and comparison of models by taking only one input. The training process was tested in all models with different input parameters and existing membership functions, and the evaluation method values were recorded each time. In the method used in model evaluation, R^2, MAPE, MAE and RMSE were used as evaluation methods. The best goal was to achieve the lowest values of RMSE, RMSE, MAE and MAPE and the highest of R^2.

In this study, prospective flow estimation was made by using the time series of daily flow values. Traditional artificial neural networks, which are frequently used in different scientific fields within the scope of forecasting studies, have also been tried to be applied to hydrological data. The success of traditional methods has been compared with the SA method, which is an Artificial Intelligence algorithm. The HB, BP, and SA models are methods of comparison. Since the internal dependency is very strong in daily flow data, the values of 3 days, 4 days and 5 days ago were used to increase the prediction performance in all models. Average results demonstrating estimation success are given in Tables 4.22, 4.23 and 4.24. While creating the tables, the average of the test values of the results obtained in all models was taken and the values were listed under their own Lag values. When the first station results are examined, the ANFIS-BP model was calculated as the lowest value of 4.9918 $\text{RMSE}_{(\text{Avg})}$ and the highest value of 0.6728 $\text{R}^2_{(\text{Avg})}$ and based on these statistics, ANFIS-BP model can be selected by preferring ANFIS-HB and ANFIS-SA. According to $\text{MAE}_{(\text{Avg})}$ or $\text{MAPE}_{(\text{Avg})}$ statistics, the model with average test values of 1.3606 and 14.3029, respectively, ANFIS-HB model can be selected by preferring other models. The ANFIS-SA model seems to be relatively weak compared to all other models based on error statistics, but it has also been observed that it finds values close to the results of other models. When the results of the second station were examined, the lowest value of 7.0241 $\text{RMSE}_{(\text{Avg})}$, and 3.2556 $\text{MAE}_{(\text{Avg})}$, 5.5199 $\text{MAPE}_{(\text{Avg})}$ and the highest 0.9207 $\text{R}^2_{(\text{Avg})}$ were calculated in the ANFIS-HB model. Based on all its statistics, the ANFIS-HB model can be chosen by preferring other models. When other models were evaluated using most of the error statistics, it was observed that they found values close to the results of the ANFIS-HB model. When the last station results were examined, the lowest 3.4880 RMSE(Avg), 1.8681 MAE(Avg) and the highest 0.9338 R2(Avg) were calculated in the ANFIS-BP model. According to the MAPE(Avg) statistic, the lowest value was calculated as 13.3283 in the ANFIS-HB model. The average MAPE value obtained with the ANFIS-HB model provided a great advantage compared to other models. When we look at all the error statistics, it is concluded that the ANFIS-BP and ANFIS-HB models make better predictions than the ANFIS-SA model. The most successful prediction was made in the Hybrid algorithm within the scope of the modelling made by ANFIS. After the analysis, it was concluded that traditional derivative-based HB and BP algorithms can be used more successfully and effectively than SA model, which is Artificial Intelligence Algorithm, in training ANFIS parameters in nonlinear problems.

ANFIS will be used frequently in the future as improved methods for researchers working in the field of Hydrology. In addition, the comparison of the ANFIS models with different methods can be chosen as the subject of future studies. It should not be ignored that the results obtained in this study are specific to the data used. Performance measurements can be repeated by making short-term data studies in addition to the estimations made using long-term data to compare all models. In addition, Back Propagation and Hybrid Learning algorithms for training ANFIS were used in this study. In the future, studies will be conducted to train ANFIS with metaheuristic artificial intelligence algorithms and apply it to different problems.

REFERENCES

AKYILMAZ, O., and Ayan T. (2010). Applications of flexible computation methods in geodesy. JOURNAL OF İTÜ/d 5.1

Akyılmaz, O., Ayan, T., Özlüdemir, M. T., (2003). Geoid surface approximation by using Adaptive Network based Fuzzy Inference Systems. AVN, 8 (9): 22-27.

Akyılmaz, O., (2005). Applications of flexible computation methods in geodesy. PhD thesis, İ.T.Ü. Institute of Science and Technology, Istanbul.

Aliev, Rafik A., et al., eds. (2018). 13th International Conference on Theory and Application of Fuzzy Systems and Soft Computing—ICAFS-2018. Vol. 896. Springer.

ALP, Murat, and H. Kerem CIĞIZOĞLU. (2010). Modelling the precipitation-flow relationship with different artificial neural network methods. JOURNAL OF İTÜ d 3.1

ALTUNKAYNAK, A., and BAŞAKIN, E. A. (2018). River flow estimation using time series and comparison with different methods. Journal of Erzincan University Institute of Science and Technology 11.1 92-101.

Asrul S.R, Ikram M, Mohd R. Mohd A.A. (2020). Energy Management Strategy of HEV based on Simulated Annealing, Int. J. of Integrated Engineering, 12(2): 30-37

Astel, A. (2007). Chemometrics based on fuzzy logic principles in environmental studies. Talanta 72.1 1-12.

Atik, I. (2022). A New CNN-Based Method for Short-Term Forecasting of Electrical Energy Consumption in the Covid-19 Period: The Case of Turkey. IEEE Access 10 22586-22598.

Avcı, E., Akpolat, Z. H. (2002). Speed control of DC motors with an adaptive network-based fuzzy inference system. ELECO’2002 Electrical-Electronics-Computer Engineering Symposium, Bursa. 193-196.

Behzad, M., et al. (2009). Generalization performance of support vector machines and neural networks in runoff modelling. Expert Systems with applications 36.4 7624-7629.

Bezdek, J.C. (1981). Pattern Recognition with Fuzzy Objective Function Algorithms, Plenum Press, New York.

Bisht, D. CS. and Ashok J. (2011). Discharge modelling using adaptive neuro-fuzzy inference system. International Journal of Advanced Science and Technology 31.1 99-114.

Bizimana, H., Demir F., and Sönmez O. (2016). Modeling of Yuvacık Dam Water Level Changes with Fuzzy Logic. 4th International Symposium on Innovative Technologies in Engineering and Science (ISITES2016) 3-5 Nov 2016 Alanya/Antalya-Turkey.

Buyukyildiz, M., Tezel, G., and Yilmaz V. (2014). Estimation of the change in lake water level by artificial intelligence methods. Water resources management 28.13 4747-4763.

Buyukyildiz, M., and Tezel G. (2017). Utilization of PSO algorithm in estimation of water level change of Lake Beysehir. Theoretical and Applied Climatology 128.1 181-191.

Celestin Vd., 2020- Celestin, S., Qi, F., Li, R., Yu, T., Cheng, W. (2020) Evaluation of 32 Simple Equations against the Penman–Monteith Method to Estimate the Reference Evapotranspiration in the Hexi Corridor, Northwest China, Water Journal, 12(10), 1-22.

Cho, Hae-Chang, et al. (2020). Heating Temperature Prediction of Concrete Structure Damaged by Fire Using a Bayesian Approach." Sustainability 12.10 4225.

Cihan, P., Ozel, H., and Ozcan H. K. (2021). Modelling of atmospheric particulate matters via artificial intelligence methods. Environmental Monitoring and Assessment 193.5 1-15.

Civelekoğlu, G., (2006). Modeling of Treatment Processes with Artificial Intelligence and Multiple Statistical Methods, Environmental Engineering Department, PhD Thesis, Isparta.

Dalkiliç, H. Y., and Hashimi, S. A (2020). Prediction of daily streamflow using artificial neural networks (ANNs), wavelet neural networks (WNNs), and adaptive neuro-fuzzy inference system (ANFIS) models. Water Supply 20.4 1396-1408.

Dibike, Y. B., and Solomatine D. P. (2001). River flow forecasting using artificial neural networks. Physics and Chemistry of the Earth, Part B: Hydrology, Oceans and Atmosphere 26.1 1-7.

Firat, M., and Turan M. E. (2010). Monthly river flow forecasting by an adaptive neuro-fuzzy inference system. Water and environment journal 24.2 116-125.

Fırat, M. (2007) Basin modeling with neural fuzzy logic approach.

Fırat, M., Yurdusev, M., Mermer, M., (2008). Estimation of monthly water consumption with adaptive neural fuzzy approach. Journal of Gazi University Faculty of Engineering and Architecture, 23 (2): 449-457.

Franklin, G. F., Powell, J. D., Workman, M. L., (1990). Digital Control of Dynamic Systems. Addison Wesley, 132 pp.

Gkountakou, F., and Papadopoulos B. (2020). The use of fuzzy linear regression and ANFIS methods to predict the compressive strength of cement. Symmetry 12.8 1295.

Gunathilake, M. B., et al. (2021). Hydrological models and Artificial Neural Networks (ANNs) to simulate streamflow in a tropical catchment of Sri Lanka. Applied Computational Intelligence and Soft Computing 2021

Günay, E., Alçı, M., Parmaksızoğlu, S., (2006). Obtaining the piecewise-linear function characteristic of the DK-HSA circuit producing multiple chaotic attractors by fuzzy neural network control. URSI-Turkey 2006 Scientific Congress and National General Assembly Meeting, Hacettepe University, Ankara.

Güneşer, M. T. (2019). Artificial intelligence solution to extract the dielectric properties of materials at sub-THz frequencies. IET Science, Measurement & Technology 13.4 523-528.

Güney, K., Sarıkaya, N., (2008). Calculation of patch radius of circular microstrip antennas with adaptive networks based on fuzzy logic system optimized by various algorithms. ELECO'2008 Electrical-Electronics-Computer Engineering Symposium and Fair, Bursa.

Guzel, M., et al. (2020). ANFIS and Deep Learning based missing sensor data prediction in IoT. Concurrency and Computation: Practice and Experience 32.2 e5400.

Hajipour, V., et al. (2016). Multi-objective multi-layer congested facility location-allocation problem optimization with Pareto-based meta-heuristics. Applied Mathematical Modelling 40.7-8 4948-4969.

Haznedar, B., Kalinli, A., (2016). Detection of the relationship between thrombophilia disease with genetic disorders by adaptive neuro-fuzzy inference system (ANFIS), Sakarya University Journal Of Science, 20 (1): 13-21.

Haznedar, B., and Kalinli, Adem. (2018). Training ANFIS structure using simulated annealing algorithm for dynamic systems identification. Neurocomputing 302 66-74.

Haznedar, B., Arslan, M. T. and Kalınlı, A. (2018). Using adaptive neuro-fuzzy inference system for classification of microarray gene expression cancer profiles.

Haznedar, B, Arslan, M. T., and Kalınlı, A. (2021). Optimizing ANFIS using simulated annealing algorithm for classification of microarray gene expression cancer data. Medical & Biological Engineering & Computing 59.3 497-509.

Hímer, Z., Kovács, J., Benyó, I., Kortela, U., (2004). Neuro- fuzzy modelling and genetic algorithms optimization for flue gas oxygen control, 2nd IFAC Workshop on Advanced Fuzzy/Neural Control, University of Oulu.

Hocaoğlu, F., Kurban, M., (2005). Prediction of sunshine durations for Eskişehir region with adaptive network-based fuzzy logic inference system, 11th National Congress and Fair of Electrical-Electronics-Computer Engineering, Istanbul.

Hussain, K., and Salleh, M. N. M. (2015). Optimization of fuzzy neural network using APSO for predicting strength of Malaysian SMEs. 2015 10th Asian Control Conference (ASCC).

İlhan, İ. (2020). A population based simulated annealing algorithm for capacitated vehicle routing problem, , Turk J Elec Eng & Comp Sci, 28: 1217–1235.

Jafari, H., Ehsanifar, M., Sheykhan, A. (2020). Finding Optimum Facility’s Layout by Developed Simulated Annealing Algorithm, Int. J. Res. Ind. Eng., 9(2): 172–182.

Jang, J. S. R., (1993). ANFIS: Adaptive network-based fuzzy inference systems. IEEE Trans. Syst., Man. and Cybern, 23: 665-685.

Jang, J. S. R., Sun, C. T., (1995). Neuro-Fuzzy modelling and control. proceedings of the IEEE, 83: 3.

Jiang, J., et al. (2021). A hybrid wavelet-Lyapunov exponent model for river water quality forecast." Journal of Hydroinformatics 23.4 864-878.

Jimeno-Sáez, P., et al. (2018). A comparison of SWAT and ANN models for daily runoff simulation in different climatic zones of peninsular Spain. Water 10.2 192.

Karaboga, D., Kaya, E., (2016). An adaptive and hybrid artificial bee colony algorithm (aABC) for ANFIS training. Applied Soft Computing, 49: 423-436.

Karahoca, D., Karahoca, A. and Yavuz Ö. (2013). An early warning system approach for the identification of currency crises with data mining techniques. Neural Computing and Applications 23.7 2471-2479.

Khatib, T., et al. (2011). Modelling of daily solar energy on a horizontal surface for five main sites in Malaysia. International Journal of Green Energy 8.8 795-819.

Kirkpatrick, S., Gelatt, C.D., Vecchi, M.P., (1983). Optimization by Simulated Annealing, Science, 220: 671-680.

Kisi, O., Jalal S., and Bagher N. (2012). Forecasting daily lake levels using artificial intelligence approaches. Computers & Geosciences 41 169-180.

Mahabir, C., F. E. Hicks, and A. Robinson Fayek. (2003). Application of fuzzy logic to forecast seasonal runoff. Hydrological processes 17.18 3749-3762.

Mamdani, E. H. (1977). Application of fuzzy logic to approximate reasoning using linguistic synthesis. IEEE transactions on computers 26.12 1182-1191.

Ministry of Agriculture and Forestry, (2019). Asi and Seyhan Basins Flood Management Plan Preparation Project. Seyhan Basin Flood Management Plan Strategic Environmental Assessment Draft Scoping Report.

Mistry, D. A., et al. (2014). Power Quality Improvement Using PI and Fuzzy Logic Controllers Based Shunt Active Filter. International Journal of Electronics and Communication Engineering 8.4 668-677.

Mutasa, S., Sun, S., and Ha, R. (2020). Understanding artificial intelligence-based radiology studies: What is overfitting. Clinical Imaging.

Niu, W., and Feng, Z. (2021). Evaluating the performances of several artificial intelligence methods in forecasting daily streamflow time series for sustainable water resources management. Sustainable Cities and Society 64 102562.

Olatunji, O. M., et al. (2022). Application of hybrid ANFIS-based non-linear regression modelling to predict the% oil yield from grape peels: Effect of process parameters and FIS generation techniques. Cleaner Engineering and Technology 6 100371.

Özçalık, H., Uygur, A., (2003). Effective modeling of dynamic systems based on coherent neural-fuzzy network structure. KSU Journal of Science and Engineering, 6 (1): 36-46.

Özgan, E., Kap, T., Beycioğlu, A., Emiroğlu, M., (2009). Estimation of Marshall stability in asphalt concrete with adaptive neural fuzzy logic approach, International Advanced Technologies Symposium, Karabuk.

Öztürk, C., (2011). Training of Artificial Neural Networks with Artificial Bee Colony Algorithm, Erciyes University, Institute of Science and Technology, PhD Thesis, Kayseri.

Padmini, D., K. Ilamparuthi, and K. P. Sudheer. (2008). Ultimate bearing capacity prediction of shallow foundations on cohesionless soils using neurofuzzy models. Computers and Geotechnics 35.1 33-46.

Raju, B. VS., Ramakrishnam, and V. V. Kumari. (2012). Parameter-lite clustering algorithm based on MST and fuzzy similarity merging. Proceedings of the International Conference on Advances in Computing, Communications and Informatics.

Rumelhart, D., et al., (1986). Learning representations by backpropagation errors. Nature, 323 (9): 533-536.

Sadouki, L., and B. Haddad. (2016). Adaptive Neuro-Fuzzy Inference System for Echoes Classification in Radar Images. VISIGRAPP (4: VISAPP).

Samanataray, S., and A.Sahoo. (2021). A Comparative Study on Prediction of Monthly Streamflow Using Hybrid ANFIS-PSO Approaches. KSCE Journal of Civil Engineering 25.10 4032-4043.

Sarma, A. K., et al., eds. (2016). Urban hydrology, watershed management and socio-economic aspects. Basel, Switzerland: Springer International Publishing.

Setia, R., and G. S. May. (2006). A neuro-fuzzy approach to failure detection and diagnosis of excimer laser ablation in microvia formation. Photon Processing in Microelectronics and Photonics V. Vol. 6106. International Society for Optics and Photonics.

Sivanandam, S. N., S. Sumathi, and S. N. Deepa. (2007). Introduction to fuzzy logic using MATLAB. Vol. 1. Berlin: Springer.

Riad, S., et al. (2004). Rainfall-runoff model using an artificial neural network approach." Mathematical and Computer Modelling 40.7-8 839-846.

TOMBUL, F. (2014). Water management in Maritsa river basin within the frame of international agreements. Anadolu University Journal of Science and Technology A-Applied Sciences and Engineering 15.2 147-155.

TOPÇU, E. (2021). Drought Analysis of Turkey's Middle Euphrates Section Using Fuzzy Logic Approach and Monthly Aydeniz Method. European Journal of Science and Technology 23: 296-303.

Tulun, Ş., et al. (2021). Adaptive neuro-fuzzy interference system modelling for chlorpyrifos removal with walnut shell biochar. Arabian Journal of Chemistry 14.12 103443.

Ubando, A. T., et al. (2017). A Systematic Approach for the Optimal Design of an Off-Grid Polygeneration System using Fuzzy Linear Programming Model. Computer Aided Chemical Engineering. Vol. 40. Elsevier, 2191-2196.

Yuen, K. K. F., and H. C. W. Lau. (2006). A distributed fuzzy qualitative evaluation system. 2006 IEEE/WIC/ACM International Conference on Intelligent Agent Technology.

Zadeh, L. A. (1965). Information and control. Fuzzy sets 8.3 338-353.

Wang, J., and C. Ning. (2015). ANFIS based time series prediction method of bank cash flow optimized by adaptive population activity PSO algorithm. Information 6.3 300-313.

Zschorn, L. (2006). An extended model of ATP to increase flexibility of delivery. International Journal of Computer Integrated Manufacturing 19.5 434-442.

Wang, K., and S. Chen. (2007). Product form design using ANFIS-Kansei engineering model. 2007 IEEE International Conference on Systems, Man and Cybernetics.