2020-03-24

ANALYSIS AND PREDICTION OF WEATHER CONDITION FACTORS USING MACHINE LEARNING TECHNIQUES

TEBABAL, SISAY

MASTER THESIS

BY SISAY TEBABAL

ANALYSIS AND PREDICTION OF WEATHER CONDITION FACTORS USING MACHINE LEARNING TECHNIQUES.

BAHIR DAR UNIVERSITY INSTITUTE OF TECHNOLOGY

FACULTY OF COMPUTING

DEPARTMENT OF COMPUTER SCIENCE

November, 2016

Sisay Tebabal

ANALYSIS AND PREDICTION OF WEATHER CONDITION FACTORS USING MACHINE LEARNING TECHNIQUES

A Thesis in partial fulfillment of the requirements for the Degree of Master of Science in computer science presented to the Faculty of Computing, Institute of Technology, Bahir Dar University.

Supervised by Bhabani Shankar D.M.

Ethiopia

Bahir Dar

2016

DECLARATION

I, the undersigned, declare that this thesis comprises of my own work. In compliance with internationally accepted practices, I have duly acknowledged and refereed all materials used in this work. I understand that non-adherence to the principles of academic honesty and integrity, misrepresentation/fabrication of any idea/data/fact/source will constitute sufficient ground for disciplinary action by the university and can also evoke penal action from the sources which have not been properly cited or acknowledged.

Signature

Name of the student

Date

By: Sisay Tebabal

APPROVAL SHEET

Approved by Board of Examiners

Faculty Director	Signature	Date
Adviser	Signature	Date
External Examiner	Signature	Date
Internal Examiner	Signature	Date

DEDICATION

I would like to dedicate this thesis to my beloved family, brother, and dear friends, for their unreserved support and encouragement.

ABSTRACT

The problem of weather variation and their impacts on socio-economic development is more acute in developing countries like Ethiopia where alternating droughts have been persistent causes of severe economic hardships. Due to this, weather prediction has become a stimulating area of research for scientists and researchers.

Weather prediction is a spatio-temporal and time series based process. Predicting future weather condition is a very important issue in today's world as the precarious fields like air flights, tourism, agricultural and industrial sectors are largely dependent on the weather conditions.

Weather prediction involves a combination of computer models, observations, and knowledge of trends and patterns by using which reasonably accurate forecasts can be made up to a finite number of days in advance.

This study is aimed to investigate and model the existing weather data series to enable the future information of the weather condition to be forecasted accurately using machine learning techniques. The study presents a research on weather forecasting by using historical weather dataset. Since atmospheric pattern is a complex and non-linear system, traditional methods are seized to be effective and efficient in such situation. It is observed that Artificial Neural Networks, including MLP, GRNN, RBF and Elman recurrent networks, are influential methods for resolving such problems.

The criteria used for model selection include MSE, MAE, RMSE, correlation coefficient, and confusion matrix.

Among the used predictive models, MLP return acceptable result with the performance criteria MAE and correlation coefficient for temperature and rainfall variables prediction.

However, for relative humidity variable prediction MLP model return better result with the performance criteria MSE and correlation coefficient. Therefore, MLP model is sufficient for future weather data prediction for the three selected variables: temperature, rainfall and relative humidity.

Keywords: Weather, Prediction, Machine learning, Meteorology, MLP.

ACKNOWLEDGEMENT

In the first place, I would like to thank my almighty God as nothing would be possible without his divine support. I am honestly grateful to my advisor, Professor Bhabani Shankar for his supervision, advice and guidance while I was working this thesis.

I also express my sincere gratitude to my friend Nega Kessete for identifying the experimental dataset for this study and his general comments on the nature of the dataset as well as for his constructive comments on this study.

I would like to thank Elias Wondmagegn, my fellow classmate in PG study, for his friendly and cooperative approach with constructive comments while undertaking this thesis.

Lastly, I especially appreciate my dear friends Dejenie Aynalem and Mulat Walle, who encouraged and supported me in accomplishing this thesis by giving productive comments and ideas.

TABLE OF CONTENTS

Content page

LIST OF TABLES

LIST OF FIGURES

NOMENCLATURE

ANN	Artificial Neural Network
ARIMA	Auto regression Integrated Moving Average
BPN	Back Propagation Network
CFSR	Climate Forecast System Reanalysis
CSV	Comma Separated Value
GCM	Global climate models outputs
GRNN	general regression neural network
IDW	Inverse Distance Weighting
IPCC	Intergovernmental panel for climate change
IPCC DDC	Inter-governmental panel for climate change distribution data center
MAE	Mean absolute error
maxTemp	Maximum Temperature
MDN	mixture density network
minTemp	Minimum Temperature
ML	Machine Learning
MLP	multilayer perceptron
MSE	Mean square error
NASA	National Aeronautics and Space Administration.
NCEP	National Centers for Environmental Prediction
NLP	non-linear prediction
NMA	National Meteorology Agency
NOAA	National oceanic and atmospheric Administration
PNN	probabilistic neural networks
RBF	radial basis function
RBFN	Radial basis function networks
RCM	Regional Climate Model
RH	Relative Humidity
RMSE	Root mean square error
RNN	recurrent Neural network
SOM	self-organizing maps

CHAPTER ONE

1. INTRODUCTION

1.1. Background

Global weather variation and their impacts on economic development have been a worldwide concern during the past several decades. The problem is more acute in developing countries in Africa, Asia, and Latin America where alternating droughts and floods have been persistent causes of severe economic hardships. To help mitigate the impacts of extreme weather and climate fluctuations requires routine timely and accurate monitoring of the global monsoon to better understand the mechanisms for monsoon variability and to improve forecasts of monsoon at all-time scales [K. E. Trenberth et.al (2000), Mintewab Bezabih et.al (2014)].

Observed change of meteorological variables such as temperature, rainfall, relative humidity, sunshine or solar radiation, wind speed and pressure are the main features of climate change. Human activities are increasingly influencing the earth's atmosphere by burning fossil fuels, cutting down rainforests and farming livestock [NOAA, 2004-2005]. This adds vast amounts of greenhouse gases to those naturally occurring in the atmosphere, increasing the greenhouse effect and global warming. These activities have impacts on ecosystem, water resource management, agriculture, contaminant management, flood prevention, etc.

Since the impact of climate change is very serious in every aspect, it needs more attention in developing countries like Ethiopia which has complex topography and scares meteorological information. Because the base of economy in developing country, like Ethiopia, is mainly dependent on rain-fed agriculture, the failure or the goodness of seasonal rainfall is incredibly critical in the country’s socio economic functioning - in particular, food production [Kassa Fekadu, 2015].

As a benefit, Meteorological information can be used for efficient resource planning and management including disaster risk reduction such as floods, droughts and wind storm to minimize the loss of life and property. The other functions of meteorological information include agricultural planning, construction, tourism, health sector and other environmental sectors. Forest Risk factor can also be predicted by temperature, humidity, dryness of wood & other factors [B. Bekuretsion and T. Beshah, 2014).

Therefore, it is necessary to investigate a model that can analyze and correlate weather variable history and predict its future values using machine learning approaches.

The existing system of national meteorology agency of Ethiopia cannot forecasts more than the next three days and this forecasting has also a problem because sometimes their forecasting has no correct relation to the actual weather condition.

To solve this existing problem, sufficient studies have not been conducted in Ethiopia on the analysis of prediction on weather variables though some research works are reported. [Dejenie Aynalem, 2015] investigated an interpolation method using machine learning techniques to interpolate existing weather stations data and to predict temperature and rainfall of remote location in Ethiopia. [Kassa Fekadu, 2015] develop statistical seasonal rainfall forecasting model using Canonical Correlation Analysis for use in real-time rainfall forecasting.

To study the weather prediction or its future condition, several techniques have been stated in many literatures but the difficulty of selecting the best approach has become a challenge for the researchers.

In the recent past, a number of studies have been conducted using traditional prediction methods such as kriging, inverse distance weighting, etc. Since atmosphere pattern is a complex and non-linear system, these traditional methods are seized to be effective and efficient. Machine leaning techniques such as artificial neural networks are perceived to be influential methods for resolving such problems like prediction of future meteorological information. These approaches have a number of advantages such as improving the accuracy of prediction analysis for drought and flood related applications, agriculture and climate analysis studies, etc. [T. Paraskevas, et al, 2010].

Machine Learning (ML) is a branch of artificial intelligence which focuses on extracting knowledge from datasets. This knowledge is represented in the form of model which provides description of the given data and allows predictions for new data [N.

Khandelwal, et al, 2012]. Many ML algorithms have shown promising results in bioclimatic modeling including modeling and prediction of species distribution.

In ML, the model built is a result of learning process that extracts useful information about the data generation process of the system using the previous observations. The resulting model can be applied with independent data to predict new values. The resulting predictions can then be compared to desired output to measure the model accuracy.

In this study, a variety of existing machine learning techniques specifically artificial neural network such as multilayer perceptron, radial basis function, general regression neural network, and Elman recurrent neural network have been investigated.

1.2. Statement of the problem

The problem of weather variation and their impacts on economic development is more acute in developing countries like Ethiopia where alternating droughts have been persistent causes of severe economic hardships. Due to this, weather prediction has become a stimulating area of research for scientists and researchers [K E. Trenberth, et.al, 2000].

As mentioned in [R. Nayak, 2015], weather Prediction is the application of science and technology to predict the state of the atmosphere for a future time and a given location. Accurately forecasting the future weather variation is very important in today's world as agricultural and industrial sectors are highly dependent on the weather conditions.

According to the recent climate change impact assessment studies [Mintewab Bezabih, et.al (2014), L. Al-Matarneh, et.al (2014), and Zenebe Gebreegziabher, et.al (2011)], agriculture, vegetation, water resources and tourism are the sectors affected directly by temperature and rainfall changes. However, the analysis and study of the weather variables in developing countries, like Ethiopia, is always hindered by limited data on weather variables especially data of the possible scenario.

The dangerous weather events cause thousands of deaths and loss of billions of U.S. dollars each year all around the world [L. Al-Matarneh, et.al, 2014]. Any change in the climate may badly affect many social, economic and tourism activities, ranging from agriculture to transportation and water resource management.

Discerning weather conditions in advanced is imperative for both individuals as well as organizations because weather information is very important to various socio-economic activities related to planning disaster mitigation, water resources management, construction, environmental protection, transportation, recreation, tourism, and others. Accurate weather forecasts can guide an airport control tower regarding what information needs to be communicated to airplanes that are taking off or landing. It can also lead a farmer to the best time to cultivate various crops, and also to predict natural disasters [H. Tyagi, 2016].

According to "World of Earth Science", a huge percentage of the total annual crop loss is often occurred due to abnormal weather condition. Crop and animal diseases are also influenced by weather. Weather anomalies directly or indirectly account for about 75% of all annual losses in farm production.

In the past, significant efforts have been made towards effective rain prediction and weather forecasting techniques. Timely and accurate weather forecasts could significantly reduce the damage to crops with the use of effective adjustments [D. C. Corrales, et.al, 2015]. Such weather forecasts could support and provide guidelines for long range seasonal planning and selection of crops which are best suited according to varying climatic conditions.

The bottom line being accurate weather forecasting is imperative for effective sustainability of agricultural and industrial sector because weather is certainly one of the most crucial issues in determining the success or failure of agricultural enterprises.

Human beings have been looking for ways to forecast accurate weather conditions. Since the weather conditions for the earlier date is already known, it can be assumed that similar weather patterns would be encountered on the later date.

Many international and regional forecast models are utilized in different countries worldwide, each using weather satellites, radar systems and other observational systems for monitoring weather in real-time [X. Ziniu, et.al, 2012].

In the recent past, a number of studies have been conducted using traditional prediction methods such as kriging, inverse distance weighting, etc. Since atmosphere pattern is a complex and non-linear system, these traditional methods are not effective and efficient to tackle such problems.

Modern weather forecasting methods involve a combination of computer models, observation and knowledge of trends and patterns. Machine leaning techniques such as artificial neural networks are perceived to be influential methods for resolving such problems like prediction of future meteorological information. These approaches have a number of advantages such as improving the accuracy of prediction analysis for drought and flood related applications, agriculture and climate analysis studies, etc. [T. Paraskevas, et al, 2010].

Investigating the models and selecting the fitted models for predicting the weather data such as temperature, rainfall, and relative humidity are important in agricultural planning, flood frequency analysis, hazard mapping, disaster prevention, water resource assessment, climate change impacts and other environmental assessment. Temperature and rainfall data are the major input for water resources studies and water resource development planning works.

The national meteorology Agency of Ethiopia forecasting method has a problem to forecast more than three days and this forecasting has also a problem because sometimes their forecasting has no correct relation to the actual weather condition. Therefore, there is a need to investigate a model that can forecast weather variables accurately and help to provide warning services in order to prevent unexpected hazards (such as drought and floods that may cause losses of life and properties) caused by weather variation.

The research carried out due to the following major reasons.

- Lack of data records and poor meteorological information to study the future weather conditions.
- Traditional prediction methods are not efficient and effective to predict and non-linear weather patterns.
- Sufficient studies were not conducted in in Ethiopia.

1.3. Research questions

The research questions to be addressed in this study are:

- How machine learning methods effectively predict the future weather patterns?
- How should weather parameters be represented in order to predict the future weather patterns accurately?
- What is the correlation between the predicted values and the actual values in the weather patterns prediction? Has it acceptable accuracy?
- What will the weather be like in the next few days/months?

1.4. Objectives

1.4.1. General Objective

The objective of the study is to investigate for an optimal model to predict weather condition factors in Ethiopia using machine learning techniques.

1.4.2. Specific Objectives

To achieve the general objective, this study attempt to address the following specific objectives.

- To investigate the appropriate machine learning techniques for predicting weather variables.
- To develop the models with appropriate data and selecting the fitted model for predicting the future weather.
- Experiment and evaluate the model output.

1.5. Significance of the study

The study is expected to have the following benefits:

- Support to provide major inputs for water resources studies and water resource development planning.
- Helps to determine the budget for future water studies and other infrastructure construction.
- Provide weather variable data to environmentalists, natural resource managers and decision makers to map and understand the future weather condition of Ethiopia.

- Support to investigate the effects in order to take adaptation measures to improve livelihood, Rehabilitation of natural resources, Improve management of land resources, soil, water, forests and agriculture.
- Predict dangerous weather events (such flood and drought) and take proactive actions than reactive measures.

1.6. Scope and Limitation

The accuracy of weather datasets in the selected area and geographical properties of the weather have been the major limitation of this study. This problem hinders the full investigation and prediction. Due to this, we are forced to focus only on prediction of air temperature, rainfall, and relative humidity in the selected area where droughts occurred most frequently.

With this limitation, the scope of this work can be stated as follows:

- Study of the weather by using the three climate elements: temperature, rainfall, and relative humidity.
- Select the specific location in which drought occurred most frequently.

1.7. Document Organization

This thesis consists of five chapters and organized as described below. First chapter consists of the introduction about the thesis, problems, objective, research question, significant, scope and limitations.

Chapter two comprises of the reviews of related literatures that helps to understand basic concepts related to source of weather data, prediction techniques to study the weather variables and its variations using machine learning techniques.

The third chapter deals with data preparation, description of algorithms and performance measures employed for model building and evaluation. The fourth chapter consists of the experimentation. In this chapter, different experiments have been conducted using the selected algorithms and their corresponding interpretations together with the performance of the developed models are discussed.

Chapter four deals to experiment and discusses the result of the experiments. Finally, chapter five concludes and gives recommendations for future work based on the findings.

CHAPTER TWO

2. LITERATURE REVIEW

2.1 Introduction

In this chapter, related literature reviews have been conducted to assess the efforts that have been made in the previous weather variables prediction and its characteristics. We have reviewed the general fields of prediction that this thesis work is involved in, and describes some previous work on the application of machine learning and statistical techniques related to weather variables prediction as well as its impacts.

The concepts and functions of time series based prediction, the source of datasets to train and test the model and techniques that have been so far done are reviewed.

Focusing on the machine learning techniques for predicting future weather variables, we decided to review previous related works such as research papers, published conference papers, surveys, and other publications that share ideas on prediction methods and machine learning approach to model it.

The sources of literatures that have been reviewed are related research papers, conferences papers, books, and other publications.

2.2 Source of Climate Data

Currently large amounts of climate data of various types can be accessible for future research. There are different sources of climate data, ranging from single-site observations scattered across each country to climate model output. Each class of data has particular characteristics before it can be used or compared. A brief introduction to each type of data source was done in (C. Monteleoni, et al, 2013). Some of the sources of climate datasets are explained below:

2.2.1 In-Situ Observations data

In-situ measurements are raw (only minimally processed) measurements of diverse climate system properties that can include temperature, rainfall, wind, column ozone, cloud cover, radiation, etc., taken from specific locations. These locations are often at the surface from weather stations. The raw in-situ data should be preprocessed into appropriate format to be easier to work with it.

2.2.2 Satellite Retrievals data

Global and near-global observations of the Earth’s climate have been made from low-earth orbit and geostationary satellites. These observations are based either on passive radiances or by active scanning via lasers or radars. These satellites mainly operate by U.S. agencies (NOAA, NASA), the European Space Agency, and the Japanese program (JAXA), and the data are generally available in near-real-time.

2.2.3 Paleoclimate Proxies data

For a longer term perspective, climate information could be extracted from so-called “proxy” archives, such as ice cores, ocean mud, lake sediments, tree rings, pollen records, caves, or corals, which retain information that is sometimes highly correlated to specific climate variables or events.

2.2.4 Climate Model Output Data

As mentioned in (C. Monteleoni, et.al, 2013), there were two types of climate model output data; Global climate models (GCM) outputs and Regional Climate Model (RCM) outputs. GCM are physics-based simulations of the climate system, incorporating components for the atmosphere, ocean, sea ice, land surface, vegetation, ice sheets, atmospheric aerosols and chemistry, and carbon cycles.

In order to incorporate more details at the local level (particularly regional topography), output from the global models or the global reanalysis can be used to drive a higher-resolution, regional climate model. The large-scale fields can then be transformed to higher resolution using physical principles embedded in the RMC code.

2.3 Weather variations in Ethiopia

As stated in [NMA, 2014], weather reflects the short-term conditions of the Earth’s atmosphere with respect to temperature, humidity, precipitation, wind speed, wind direction, cloudiness, and other elements at a particular place.

As mentioned in [N. Khandelwal and R. DAVEY, 2012], change of weather parameters event is one of the natural events frequently generating serious impact on the world’s economy and social life of humans. The amount of sunlight striking an area, the geographic location of an area, the air pressure surrounding an area and the amount of water in the atmosphere all influence the local weather.

According to [Temesgen Enku and Assefa Melesse, 2014], Ethiopia is located about 3-15^0 N and 33-48^0E above the equator. There is high altitude difference in the country, which ranges from an elevation of about 116 meters below sea level to 4620 meters above sea level. Because of this high altitude difference, there is high spatial variability of temperature and high seasonal variability of rainfall, whereas the seasonal variability of temperature is relatively low.

In Ethiopia there are three seasons based on climatological means of rainfall and temperature. These seasons are locally known as Kiremt, Bega, and Belg [Kassa Fekadu, 2015]. Kiremt is the main rainy season, occurring from June to September. Bega is the dry and cool season, running from October to January. Belg is short rainy season from February to May. Belg is the small rainy season, lasting from February to May. It is described by varying dry and wet days.

[P. Camberlin, 1997] reported that the Indian monsoon activity is a major cause for summer rainfall variability in the East African Highlands. During the period from June to the end of August, Ethiopia enjoys its summer monsoon season, when rainfall is at its heaviest. From October to May, most part of Ethiopian has dry season. This is the time of year when most visitors choose to come to Ethiopia.

In the period from October to January, the overall climate in Ethiopia is a little cooler than during the rest of the year. From March to May, temperatures rise again. However, night time temperatures do fall quickly.

The rainfall season over the study area starts in June and ends in the half of September. The rest of the time, it remains relatively dry.

Even though weather has several variables, we only focus on the three weather factors prediction: temperature, rainfall, and relative humidity.

Temperature is a measure of the air's hotness or coldness during continuous time interval of 24 hours. Rainfall is the result of water vapor condensed. Rainfall information is an important input in the hydrological modeling, predicting extreme precipitation events such as droughts and floods, estimating quantity and quality of surface water and ground water [B.S. Majani, 2007].

Relative Humidity is a type of humidity that considers the ratio of the actual vapor pressure of the air to the saturation vapor pressure. It is usually expressed in percentage. Humidity is water vapour content of the air.

Humans are very sensitive to humidity, as the skin relies on the air to get rid of moisture. That is, relative humidity is used to determine human and animal comfort levels. Wind is motion air caused by differences in air pressure, which is produced by the uneven heating of the earth's surface.

2.4 Weather data prediction

Although weather prediction is a complex process and a challenging task, it is a stimulating area of research for scientists. The prediction of atmospheric parameters is essential for various applications such as drought detection, severe weather prediction, agriculture and production, planning in aviation industry, etc. Accurate prediction of weather parameters is a difficult task due to the dynamic nature of atmosphere [Y.Radhika and M.Shashi, 2009].

Weather forecasts for farming and agriculture can be grouped into three primary categories: (1) short range forecast, which is up to 40 hours, (2) medium range forecasts, which is from 3 to 10 days, and (3) long range forecasts, which is from one week to entire season. Each of which plays a significant role in farm operations and planning of agricultural activities.

Weather prediction is made by collecting quantitative data about the past and current state of the atmosphere at a given place.

According to [R. Nayak, 2015], Application of current technology and science helps to predict the state of the atmosphere for a future time in the given location. It involves a combination of computer models, observations, and knowledge of trends and patterns. By using these methods, reasonable accurate forecasts can be made up to N days in advance. This is essential to prepare for the best and the worst of the weather condition.

2.5 Prediction techniques

As mentioned in [S. Singh and J. Gill, 2014], time series based prediction is an important technique in which past observations are collected and analyzed to develop a model to discover the best patterns that describe the series as x_1, x_2, …, x_{t+1}. Time series based prediction takes an existing series of data x_{t-n}, x_{t-n-1}, …, x_{t-2}, x_{t-1}, x_t as input and forecasts the data values x_{1+1}, x_{t+2}, …, x_{t+m} ,where x_t is the value of a variable measured at time t. In such approach, the order (in time) in which an event occurs is crucial for the analysis.

According to [Chapman, 2000], the behavior patterns of time series can be grouped as seasonal, trend, cyclic and irregular fluctuation. Seasonal Variation is a cyclic variation that occurs at regular periods of time, a year, quarterly, monthly or weekly. Trend is the type of variation consisting of upward or downward behavior of the series. Cyclic variations occur at regular periods not related with the calendar. The last Irregular fluctuation is a random behavior that is not characterized by any of the previous variations.

As we have seen in the literature, many works were done related to time series based prediction. The most recent method in time series based forecast is applying machine learning or statistical techniques, or a combination of models to perform prediction. Some instances are efficient temperature prediction system with back propagation neural network [S. S Baboo and I.K Shereef, 2010], hybrid model for ongoing forest fire development forecasting by applying neural networks and ARIMA [T. Cheng and J. Wang, 2008], hybrid approach to model time series data of water consumption prediction using BPN and ARIMA (S. BuHamra, et.al, 2003].

Machine learning is very powerful approach to data analysis, modeling and visualization, and it is developing rapidly for application in different fields. That is, the goal of machine learning is to develop methods that allow computers to learn from training. There are several types of methods that were applied to develop a model for temporal data analysis and forecasting purpose. The well-known types are statistical and machine learning techniques.

2.5.1 Geostatistical prediction Methods

According to [B. S Majani, 2007], Geostatistics is a branch of statistics that focus on spatial or spatio-temporal datasets. This method is used to create spatial correlation between neighbouring observations to predict attribute values at target locations.

In Geostatistics, when the variable of interest is sparse or poorly correlated in space, the prediction of this variable over the whole study area may be improved by accounting for secondary information exhaustively sampled over the same study area. The secondary information can be incorporated using kriging. That is, Geostatistics addresses the problem of estimating unknown values of a variable of interest, Z, on certain geographical locations, based on a spatial data set $Z = (Z_1, Z_2, \ldots, Z_n)$, where Z_i is the value of the variable Z at location i. This idea was motivated by the first law of geography that is “everything is related to everything else but nearby things are more related than distant things”. The most commonly used traditional geostatistical methods are inverse distance and Kriging [J. Diggle and J. A. Tawn, 1998].

2.5.1.1 Inverse Distance Weights (IDW)

This method estimates the values of an attribute at target points using a linear combination of values at sampled points weighted by an inverse function of the distance from the point of interest to the sampled points. In this approach, the sampled points closer to the target points are more similar to it than those further away in their values. Weights decrease as the distance increases so nearby samples have a heavier weight and have more influence on the estimation [O. Babak and C.V. Deutsch, 2009].

2.5.1.2 Kriging

As mentioned in [B. S Majani, 2007], Kriging is a geostatistical estimation technique which uses a linear combination of sampled values to make predictions at unsampled locations. In kriging, to predict we need to know the weights applied to each surrounding sampled data. It allows deriving weights that result in optimal and unbiased estimates.

When predicting in the presence of spatial dependence, observations close to each other are more likely to be similar than those far apart. Temporal variation and spatial distribution of meteorological variables such as rainfall and temperature can be analyzed and predicted using this statistical tool.

2.5.2 Machine learning Methods

Before we go to specific machine learning algorithms, we decided to look its basic definition along with some of the different learning style.

According to [T. O. Ayodele, 2010], machine learning is a method of teaching computers to make and improve predictions or behaviors based on some examples or experiences. Its major focus is to automatically produce models, and a model is a pattern, plan, representation, or description designed to show the main working of a system, or concept for performing a mathematical operation and obtaining a certain result.

Since machine learning is a very powerful approach to data analysis, modeling and visualization, it's algorithms are very important to a range of technologies, including web search, recommendation systems, personalized Internet advertising, computer vision, and natural language processing.

As briefly described in [T. O. Ayodele, 2010], machine learning algorithms are organized into groups based on the desired outcome of the algorithms. The well-known algorithm types include:

Supervised learning: This algorithm uses known input with known desired output to create a model that describes their relationship. This model is prepared through a training process where it is required to make predictions. The training process continues until the model achieves a desired level of accuracy on the training data.

For each training sample, the results are computed with the current state of the network, and compared the actual output with the desired output. To adjust the network input-output behavior, the weights of the neurons need to be updated.

Supervised learning algorithm can be applied for classification and regression problems.

In this learning algorithm, a mean squared error is usually taken to measure its accuracy, although other measures can be used as well.

Unsupervised learning: This algorithm attempts to create models to relate input data without having any predefined data sets (label examples) to learn. The learning task is to find some kind of output based on the assumed structures and dependencies in the input space. The algorithm can be applied for clustering and dimensionality reduction problems.

Semi-supervised learning: This is a combination of supervised and unsupervised learning algorithm. This algorithm is required when there are both known and unknown data samples. This type of learning can be used with methods such as classification, regression and prediction.

Reinforcement learning: This algorithm learns a policy of how to act on the given an observation of the world. Every action has some impact in the environment, and the environment provides feedback that guides the learning algorithm.

Based on the literatures we have reviewed, artificial neural network and support vector machine are the most appropriate algorithms for environmental studies.

2.5.2.1 Artificial Neural Network (ANN)

An Artificial neural network is an example of a non-linear prediction (NLP) method, which have been extensively studied and applied to a variety of problems, including meteorological simulation and forecasting [Taye Tolu, 2011].

ANN is a strong data modeling tool that is able to represent complex relationship between input and output. One of the important characteristic of neural networks is its adaptive nature, where "learning by example replaces programming". This feature makes the ANN methods very attractive in application domains for solving highly non-linear problems. During the past, various complex problems like weather prediction, stock market prediction, etc. have been evidenced to be areas with abundant scope of application of this sophisticated mathematical tool.

An artificial neural network is information processing system that has certain performance characteristics in common with biological neural networks. It can perform intelligent tasks similar to those performed by the human brain [L. Fausett, 1994].

As mentioned in [S. S. Baboo and I.K Shereef, 2010], an artificial neural network resembles the human brain in the following two ways:

(1) A neural network acquires knowledge through learning, and

(2) A neural network's knowledge is stored within interneuron connection strengths known as synaptic weights.

The basic unit for information processing, as considered in biological neuroscience, is a neuron that are linked together based on specific network architecture. Each connection link has an associated weight, and each neuron applies an activation function, usually nonlinear, to its net input (sum of weighted input signals) to determine its output signal. The simple and basic model architecture of neural network is illustrated in the following figure:

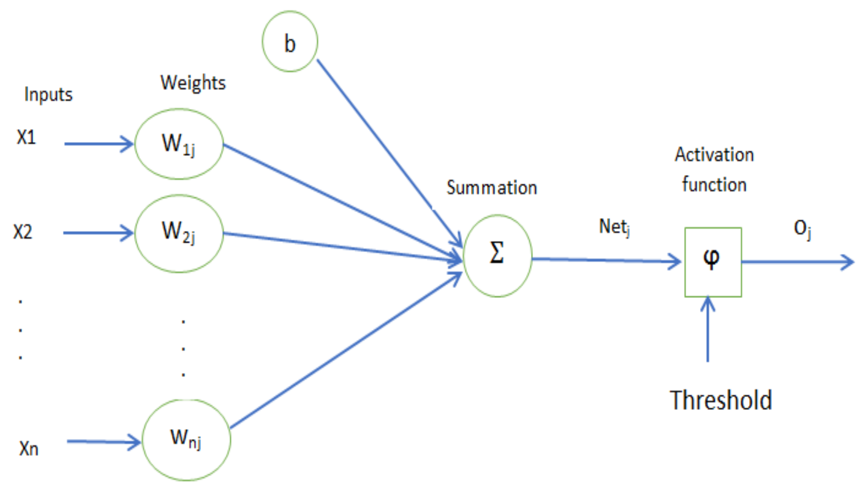


Figure 2.1 McCullogh-Pitts Model Artificial Neuron [R. Nkoana, 2011]

$net_j = \sum w_{ij} x_i + b$ and $o_j = f(\mathrm{net}_j)$

Where - w_{ij} is the vector of synaptic weights x_i is the input to the neurons, b is the bias parameters, f is non-linear activation function, and o_j is the network output.

The Neural Networks package supports different types of training or learning algorithms. Thus include multilayer perceptron (MLP), radial basis function (RBF), general regression neural network (GRNN), Elman recurrent neural network, probabilistic neural networks (PNN), mixture density network (MDN), self-organizing maps (SOM) and Kohonen networks, etc.

An ANN model can be adjusted, or trained using the appropriate datasets (training set). After successful training, the network will be able to perform classification, estimation, prediction, or simulation on new data from the same or similar data sets.

2.5.2.1.1 Multilayer perceptron Neural Network

According to [Tariku Debela, 2013], multilayer perceptron (MLP) is the most known and most frequently used type of neural network, which is a fully connected network of neurons organized in several layers. As mentioned in the above, the basic unit of the neural network is the neuron. The basic architecture of the MLP model consists of three types of neuron layers. These are input layer, hidden layers, and output layer.

According to [A. Kaur, et.al, 2011], multilayer perceptron network can learn with a supervised learning rule using the back propagation algorithm. The backward error propagation algorithm for ANN learning or training causes advancement in the application of multilayer perceptron. The back propagation algorithm gives rise to the iterative gradient algorithms designed to minimize the error measure between the actual output of the neural network and the desired output using a pre-computed error on the forward pass of information through the network.

As mentioned in [R. Nkoana, 2011], multilayer perceptron neural network can be grouped into two major categories: feed-forward network and feedback (recurrent) networks.

The first most commonly used MLP neural network is a layered network in which neurons are organized into layers with connections strictly in one direction from one layer to another. The signal flow is from input units to output units, directly in a feed-forward direction. The data processing can extend over multiple units, but no feedback connections are present.

The second one is the recurrent Neural network (RNN), in which the network has at least one feed-back connection, so that activation can flow round in a loop that enables the networks to do temporal processing and learn sequences. Thus RNN may have feedback connections between units of different layers from output to inputs. This implies that the output of the network not only depends on the external inputs, but also on the state of the network in the previous time step. RNN has several advantages over feed-forward neural network; the first is that RNN has the capability to retain values from previous cycles of processing, which can be used in current computations. This advantage allows RNN to produce complex, time varying outputs in response to simple static inputs.

There are two ways to configure the multilayer neural networks [Taye Tolu, 2011]. The first one is that setting the weights manually and training the neural network to learn certain patterns. The second approach is the most commonly used that is the network learn the pattern by itself and accordingly update its weights

2.5.2.1.2 Generalized regression neural network (GRNN)

According to [D. F. Specht, 1991], GRNN was introduced as a memory-based network that provides estimates of continuous variables. The algorithm provides smooth approximation of a target function even with sparse data in a multidimensional space.

A GRNN is a variation of the radial basis neural networks, which is based on kernel regression networks [B. Kim et.al (2004), H.K. Cigizoglu and M. Alp (2006)]. It does not require an iterative training procedure as back propagation networks, rather it approximates any arbitrary function between input and output vectors, drawing the function estimate directly from the training data. In addition, it is consistent as the training set size becomes large.

This network is a well-known and widely applied mathematical model for forecasting of nonlinear functions [Napagoda N.A.D.N., 2013]. It consists of four layers: input layer, pattern layer, summation layer and output layer as shown in Figure 2.3

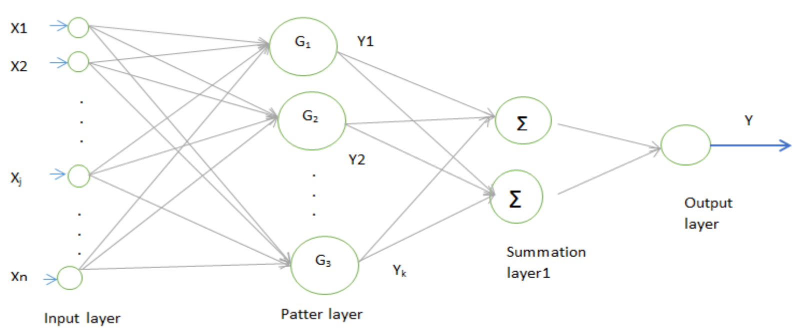


Figure 2.2 General structure of GRNN [B. Kim et al, 2004]

The number of input units in the first layer is equal to the total number of parameters. The first layer is fully connected to the second pattern layer, where each unit represents a training pattern and its output is a measure of the distance of the input from the stored pattern. Each pattern layer unit is connected to the two neurons in the summation layer.

This network can be treated as a normalized radial basis function network in which the hidden unit is centered at every training case.

2.5.2.1.3 Radial Basis Function (RBF)

Radial basis function is an alternative to the more widely used MLP network that consumes less computer time for network training [S. A. Hannan, et.al, 2010].

RBF consists of three layers: an input layer, a hidden (kernel) layer, and an output layer. The numbers of neurons within each layer are fully connected to the previous layer. The input variables are each assigned to the nodes in the input layer and they pass directly to the hidden layer without weights.

According to [L. Fausett, 1994], Radial basis function networks can be used for approximating functions and recognizing patterns. This function is feed-forward network consisting of a hidden layer of radial kernels and an output layer of linear neurons. The hidden layer performs a non-linear transformation of input space and the output layer performs linear regression to predict the desired targets.

In RBF, the activation function is the nonlinear radial basis function (kernel, or units) which usually is a radially symmetrical function [M. D. Buhmann, 1963].

It is a function of the distance $r = ||x - c||$ between points and some location c in the space which is called the center of the RBF, and can also possess some inner parameters, describing its characteristic width. Most commonly used RBFs are Gaussians.

2.6 Related works on weather prediction

Many works have been consummated related to prediction of weather variables using artificial neural network models. Some of them are presented below. This leads us to make better understanding of weather variables prediction. Some basic concepts, findings & facts of the prior works are explained to make some conventions for our work.

[S. Singh and J. Gill, 2014] proposed a Hybrid back propagation based genetic algorithm approach to train neural networks for weather prediction. This time series based hybrid technique could learn efficiently by combining the strengths of genetic algorithm with back propagation. The model was developed with multilayer feed forward architecture for prediction of the weather variables. Comparison for RMSE values has been done corresponding to different number of hidden neurons along with the population size and number of iterations to select the best model.

[Ch. J. Devi, et.al, 2012] Present two different ANN architectures to forecast temperature. In this work, a three layered neural network is designed and trained. Many parameters are taken, like temperature, humidity, dew point, atmospheric pressure, sea level, wind speed, wind direction, etc. The work was done to check two different ANN architectures. These were Back Propagation (BPN) feed forward network and Radial basis function network (RBN), in which BPN was selected as the best.

[S.S. Baboo and I.K. Shereef, 2010] used back propagation neural network techniques for doing prediction and tested as the best algorithms for training the ANN. The ANN was trained & tested using complete one year weather data including Temperature, Due Point, Humidity, Sea Level Pressure, wind speed, etc. ANN was trained with 200 data and tested for unseen data and then the result varied with 2.16% errors. This research paper has shown minimum RMSE error of 0.0079 and maximum RMSE Error of 1.2916. The epochs taken were 1000 to 5000. *Epoch* is number of iterations taken by a network to train the model with the training data.

[S.S. De, 2009] used ANN to forecast the maximum & minimum temperature for Monsoon month. The temperatures of June, July & August were predicted with the help of January to May temperature. The data of three months of 1901 to 2003 were used. The ANN model generated here was a single hidden layer model with 2 neurons at hidden layer. After 500 epochs the result was validates. The Maximum Error appeared was 5%.

[Y. Radhika and M. Shashi, 2009] present an application of Support Vector Machines for weather prediction. Time series data of daily maximum temperature at given location was studied to predict the maximum temperature of the next K days at that location based on the daily maximum temperatures for a span of previous N days referred to as order of the input. Performance of the system was observed for various spans of 2 to 10 days by using optimal values of the kernel.

[M. Hayati and Z. Mohebi, 2007] present ANN for one day ahead prediction of temperature. They used MLP to train & test ten years' meteorological data. For accuracy of prediction, they split data into four seasons and then for each seasons one network was presented. Two random unseen days in each season were selected to test the performance. In this research, the results shown that MLP network had the minimum forecasting error and MLP was considered as a good method to model the short-term temperature forecasting systems. The error in result varied from 0 to 2 for MSE.

[B.A. Smith et.al, 2007] focused on developing ANN models with reduced average prediction error by increasing the number of distinct observations used in training, adding additional input terms that described the date of an observation, increasing the duration of prior weather data included in each observation, and reexamining the number of hidden neurons used in the network. The Model was created to forecast air temperature at hourly intervals from 1 to 12 hours ahead. Each ANN model, having a network architecture and set of associated parameters, was evaluated by instantiating and training 30 networks and calculating the mean absolute error (MAE) of the resulting networks for some set of input patterns.

[Dejenie Aynalem, 2015] investigated an interpolation method based on ANN approach to interpolate existing weather station data and then to predict mean temperature and mean rainfall. This interpolation method was used to estimate the values of a given locations using observations of that variables made at another location. In the study three ANN models, such as MLP, RBF and GRNN, were used and the RBF model was selected as the optimal interpolation methods for a new data to interpolate or estimate for the sampled location weather station variables data based on the performance criteria MAE, MSE, RMSE and Correlation coefficient.

The proposed approach, in this thesis, is investigating the predictive model using machine learning techniques; specifically ANN. The data model in the proposed approach uses day wise weather data (maximum 366 days) over the duration of 25 years to train the model and can predict the next few days or months weather data (day wise). In this study, we considered the effects of other weather condition factors to predict the values of single weather variable. For instance, we considered rainfall, relative humidity, evapotranspiration, temperature of previous 25 years, and elevation and coordinates to predict the next few days, or month temperature data.

According to [T. R. Nkuna ,and J. O. Odiyo, 2016], the influence of temperature on rainfall has been incorporated in an indirect, or sometimes a direct way in a number of studies. Temperature influences rainfall in many ways; such that in some cases high temperatures may result in exceedingly high rates of potential evaporation and low precipitation. This results in an area being dominated by an arid or semi-arid landscape. In other cases, high temperatures lead to more evaporation and consequently increased condensation leading to high rainfall.

CHAPTER THREE

3 DATA PREPROCESSING AND MODELING

3.1 Dataset overview

In this chapter, we have preprocessed the dataset and developed a model using machine learning techniques to perform prediction. Datasets have been taken for the area of interest obtained from National Centers for Environmental Prediction (NCEP) Climate Forecast System Reanalysis (CFSR). These datasets are preprocessed into a format that is appropriate for training, validation and testing of the selected models. Transforming and filling missing data activities are assessed here.

3.2 Data analysis

3.2.1. Source of data

According to [Yihun Taddele and R. Srinivasan, 2014], data scarcity has been a huge problem in modeling weather variables and water resources. Several hydrological modeling studies have been carried out using satellite data and different statistical methods to improve the quality of conventional meteorological data in Ethiopia.

Most of the studies have used a daily weather generator from satellite to generate climate data or to fill the gaps in the measured records. Global reanalysis weather data have been used for various hydrological applications all over the world and yielded sound results. In data-scarce regions such as remote parts in Ethiopia, CFSR weather could be a valuable option for hydrological predictions where conventional gauges are not available.

Although the available datasets in Ethiopia are scares, there are some options to access the datasets related to Ethiopian weather variables. Thus options of weather data sources include:

- National Meteorology Agency (NMA): we can collect weather station climate data from Ethiopia National Meteorology Agency. However, this datasets have many missed values.
- Cape Town University. African countries climate data can obtain from Cape Town University, South Africa.

- Datasets from (NCEP CFSR): The National Centers for Environmental Prediction (NCEP) Climate Forecast System Reanalysis (CFSR).

The NCEP was established by the National Oceanic and Atmospheric Administration (NOAA) in 1958 as the National Meteorological Center. It is located in Camp Springs, Maryland. The NCEP's goal is to protect life, property, and enhance the nation's economy and growing need for environmental information by providing accurate forecasts and forecast guidance products to weather service field offices.

The NCEP delivers national and global weather, water, climate, and space weather guidance to a broad range of users and partners. These products and services respond to user needs to protect life and property, support and enhance the nation's economy and growing need for environmental information.

The source of datasets for this study is from National Centers for Environmental Prediction (NCEP) Climate Forecast System Reanalysis (CFSR), which has a daily weather data over 34-year period of 1979 through 2013. The CFSR is designed and executed as a global, high resolution, coupled atmosphere-ocean-land surface-sea ice system to provide the best estimate of the state of these coupled domains over this period.

NCEP has satellite station in each 38 kilometer apart from the whole world. These stations are used to collect daily weather data. The data include observation date, station's coordinate, station's elevation, maximum and minimum temperature, precipitation, wind speed, relative humidity and solar in csv file format for a given location and time period. NCEP has a complete daily record since 1979 except the year 1986, which has records only 23 days for the month December.

	A	B	C	D	E	F	G	H	I	J
1	Date	Longitude	Latitude	Elevation	Max Temperature	Min Temperature	Precipitation	Wind	Relative Humidity	Solar
2	1/1/1979	38.4375	12.64529991	1639	30.97	8.234	0	1.358714073	0.296999337	22.96653588
3	1/2/1979	38.4375	12.64529991	1639	29.884	9.395	0	1.386521613	0.343878664	23.10035868
4	1/3/1979	38.4375	12.64529991	1639	31.184	8.94	0	1.307172387	0.433636351	23.176404
5	1/4/1979	38.4375	12.64529991	1639	30.237	9.901	0	1.108226967	0.50456045	23.0592924
6	1/5/1979	38.4375	12.64529991	1639	28.142	10.304	0	1.48765177	0.479942971	22.5612576
7	1/6/1979	38.4375	12.64529991	1639	28.588	10.099	0	1.603014169	0.413480798	20.65742172
8	1/7/1979	38.4375	12.64529991	1639	29.413	7.484	0	1.548778199	0.339814687	23.08927608
9	1/8/1979	38.4375	12.64529991	1639	30.367	10.523	0	1.066165063	0.430287197	17.69636772
10	1/9/1979	38.4375	12.64529991	1639	31.363	12.112	0	1.359030594	0.41222444	19.96171632

. . .

12771	12/26/2013	38.4375	12.64529991	1639	31.788	8.737	0	0.770474542	0.396525121	21.58675581
12772	12/27/2013	38.4375	12.64529991	1639	31.879	10.137	0.001716613	0.828041083	0.427221441	21.04182354
12773	12/28/2013	38.4375	12.64529991	1639	32.136	9.947	0	0.688353352	0.338695919	22.95331449
12774	12/29/2013	38.4375	12.64529991	1639	32.397	8.161	0	0.607515703	0.321545158	23.01867508
12775	12/30/2013	38.4375	12.64529991	1639	31.851	8.382	0	0.683397763	0.256998952	23.10522016
12776	12/31/2013	38.4375	12.64529991	1639	31.861	8.24	0	0.734003309	0.258514908	23.09606859

Figure 3.1 Original data format of Ebinat weather station

We have used NCEP satellite weather station data for this study because most NMA's stations are recently installed (after 2000). Though some of the stations in capital cities and in some particular places are installed since 1980, they have more missing values and incomplete records.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1	Reg:	Amhara		Lat:	12.65										
2	Stn:	Guhala		Lon:	38.13										
3	Ele	Maximum Temperature													
4	Year	DATE	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec	
5	2012	1	xx	xx	xx	36.5	na	33	28	25	26	27.5	25.5	28.5	
6		2	xx	xx	xx	35.5	na	33	28.5	26.5	25	28	26	28.5	
7		3	xx	xx	xx	31	na	32	28	24.5	24	28.5	25.5	28.5	
8		4	xx	xx	xx	32.5	na	31.5	24.5	24	25	29	27	28.5	
9		5	xx	xx	xx	31.5	na	31.5	24.5	24	26.5	30	26.5	29.5	
10		6	xx	xx	xx	32	na	31.5	26.5	25	26.5	29	28	29	
11		7	xx	xx	xx	33.5	na	32.5	26	24	29	29.5	25.5		
12		8	xx	xx	xx	32.5	na	31	24.5	23	27	29.5	25.5	28.5	
13		9	xx	xx	xx	33.5	na	32.5	25.5	24	27.5	29.5	27.5	28	
14		10	xx	xx	xx	31	na	32.5	26	24.5	27.5	29	28	29	
15		11	xx	xx	xx	32.5	na	31.5	27.5	24.5	29	29.5	28	28	
16		12	xx	xx	xx	33	na	31.5	26	24.5	27.5	29.5	29.5	28.5	
17		13	xx	xx	xx	33.5	na	30.5	27	24.5	28	29	29.5	28	
18		14	xx	xx	xx	32.5	na	31.5	27	24.5	34	28.5	30.5	29	
19		15	xx	xx	xx	33.5	na	32.5	27	26	28	27	31	28	
20		16	xx	xx	xx	32.5	32.5	32.5	27	24.5	28	27	28.5	28.5	
21		17	xx	xx	xx	33.5	32.5	30	26	25	28	27.5	28.5	28	
22		18	xx	xx	xx	33.5	32.5	31	25	25.5	28	28	28.5	28	
23		19	xx	xx	xx	33	33.5	30	24.5	25.5	28.5	28.2	29.5	27	
24		20	xx	xx	xx	33.5	32.5	28.5	25.5	26	29	29.5	29.5	27.5	
25		21	xx	xx	xx	32	32.5	29	24	24.5	29	29.5	29.5	28	
26		22	xx	xx	xx	31.5	32.5	27	25.5	23	29	29	29	27	
27		23	xx	xx	xx	32.5	32	28.5	25.5	24.5	29	29.5	29	28.5	
28		24	xx	xx	xx	32	31	28	26.5	26	29.5	29.5	29	28.5	
29		25	xx	xx	xx	32	31	29	24	26.5	29.5	29.5	28.5	28	

Figure 3.2 Original data format of Guhala station from NMA

For this study, we took daily minimum and maximum temperature, rainfall, wind speed, and relative humidity from 18 satellite weather stations in the Northern part of Ethiopia as representative. These representative weather stations are taken from areas where droughts occurred most frequently.

From the given temperature, we have calculated and found out the evapotranspiration data using simple temperature method calculation [Temesgen Enku and Assefa Melesse, 2014] and used as a predictor variable to predict temperature, rainfall and humidity. Evapotranspiration is a calculated estimate of the water that evaporates from soil, water surfaces and plants. Evapotranspiration is one of the main components of the hydrologic cycle.

Some of the selected stations and their geographical variables (latitude, longitude and elevation) are displayed below.

Table 3.1 Sample selected stations and their geographical properties

No	Station name	Longitude (degree)	Latitude (degree)	Elevation (m)
1	Belesa	38.125	12.6453	1841
2	Dehana	38.75	12.333	2421
3	Ebinat	38.125	12.333	1742
4	Kemkem	37.8125	12.333	2031
5	Sekota	39.375	12.333	2555
6	Wegera	37.8125	12.6453	2032
7	Ziquala	38.75	12.6453	1470
8	Gonder zuria	37.5	12.333	1794

3.2.2. Data Preparation

Variety types of datasets can be generated or collected from real world, which may have missed or inconsistent values or it may highly susceptible to noise. Therefore, data preparation is the first activity for machine learning process. This process should be done before one can apply machine learning techniques to the dataset.

Data preprocessing activities help to produce quality data and to improve prediction output. This process includes data collection, data cleaning, data transformation, data integration, data reduction, and data discretization (S. Zhang et al, 2003).

Since we are going to use machine learning algorithms, the process of data transformation is required. This process include three basic steps: (1) selecting the data – considering what data is available, what data is missing and what data can be removed, (2) preprocessing the data - organizing the selected data by formatting, cleaning and sampling from it, and (3) transforming the data -transforming preprocessed data to be ready for machine learning process by using scaling, attribute decomposition and attribute aggregation.

Use of transformed data improves the speed, accuracy, efficiency, and performance of the model [Gustavo E. A. P. A. Batista and M.C. Monard, 2008].

The daily weather variables collected from NCEP weather stations have missing values of seven days for the month December in the year 1986 for all features.

3.2.2.1. Missing value filling techniques

As mentioned in [Tariku Debela, 2013], missing values might be occurred due to equipment malfunction, data not entered due to misunderstanding, certain data may not be considered important at the time of entry, inconsistency, no registered history or changes of the data.

The presence of missing values in a dataset can affect the performance of a model built when using that dataset as a training sample. Rates of less than 1% missing data are generally considered as insignificant and 1-5% manageable. However, 5-15% requires sophisticated methods to handle, and more than 15% may severely impact any kind of prediction/interpretation.

There are different techniques suggested in the literature for treating the missing values. Some of those techniques include: (1) Ignoring missed value: this method usually done when class label is missed (assuming the task in classification). (2) Filling in the missed value manually. This method is tedious, time consuming and infeasible. (3) Use of a global constant to fill in the missing value. This method replaces all missing attribute values by the same constant such as a label like Unknown. This can be done if a new class is unknown. (4) Use of the most probable value to fill in the missing value. (5) Use of the attributes' mean to fill in the missing value: Replacing the missed values with the attributes' mean or mode (for numeric or nominal attributes, respectively).

The last method is the most commonly used approach to handle missing values in a dataset [J. Han, et.al, 2012].

In weather dataset, missing values are mostly due to equipment malfunction or users error at the time of entry. Therefore, the appropriate method to fill the missing value in weather data uses the attributes' mean because all attributes in the weather dataset are numeric attributes and missing values of time series data has strong relation to its previous and next value in the series.

In this study, the missed value filled with the mean values of the same day with same month of the previous five years and the next five years because all attributes in the weather datasets are numeric attributes and missing values of time series data have strong relation to its previous and next value in the series.

3.2.2.2. Data transformation

After the collection of data and selection of the weather variables, the next issue is data transformation. Neural networks generally provide improved performance with normalized data. The use of original data to train the neural network may cause convergence problem. Therefore, we have to normalize the data at the beginning of modeling for improving the accuracy, speed, and efficiency of the network training step.

Among the data transformation methods, the most commonly used one is scaling, which can be done to have a specific range, such as the range between -1 and 1 or 0 and 1, for the entire data set of a given station.

Therefore, the data are scaled into a range between 0 and 1 before the training stage.

Here, the aim is that all the weather data sets are transformed into values between 0 and 1 through dividing the difference of actual and minimum values by the difference of maximum and minimum values. The equation for the scale range [0, 1] is given by:

$$v' = \frac{\mathrm{v-min}}{\mathrm{max-min}} \tag{3.1}$$

where - *v'* is the scaled value of a given station

- *v*: the actual value of a given station

- min: minimum value in the entire data set of a given station
- max: maximum value in the entire data set of a given station

This transformation process is done in order to normalize input variables and to prevent slow convergence in the training process [Juan P. Rigol, et.al, 2001]

3.2.2.3. Set up the data

Dividing the available sample dataset for training, testing and validation is the most common for prediction and classification in machine learning techniques. Therefore, the selected data sets were randomly divided into three groups: 70% for training, 15% for validation and 15% for testing after it had been first transformed into a certain range as explained in [Juan P. Rigol, et.al, 2001]. This random division of data set is more suitable for MLP and Elman network model.

3.2.2.4. Implementation approaches

In this study, the proposed approaches assessed different machine learning techniques that are more suitable to model existing time series based datasets, and then evaluated the performance using different metrics.

For research purpose, there are several types of applications software that have the latest version in commercially as well as open source. Therefore, to implement our model, we have used matlab R2014a tool that is appropriate to process the weather datasets. This tool is more suitable to build predictive models, to train it and to evaluate the output of the trained model.

3.2.3. Prediction approach

Prediction is a kind of dynamic filtering, in which past values of one or more time series are used to predict future values. A time series based dataset is a sequence of vectors, x(t), where t = 0,1,… n, which represents elapsed time, and x is a value which varies continuously with time t, such as temperature, rainfall, etc. In practice, for any given physical system, x will be sampled to give a series of discrete data points, equally spaced in time.

The work with artificial neural networks has focused on forecasting future developments of the time series from values of x_{t-p} up to the current time. This means obtaining an estimated value of x at time t + d, from the p time steps back from time t.

$$x_{t+d} = f(x_t, x_{t-1}, \ldots, x_{t-p+1}) \quad (3.2)$$

$$x_{t+d} = f(y_t) \quad (3.3)$$

Where y_t is p vector of lagged x values. Normally d is one, so that the function f will be forecasting the next value of x.

In neural network model, the usual and appropriate method for model training is to first partition the data sets into three disjoint sets: training set, validation set, and testing set. The network is trained with feed-forward back propagation algorithm, generalized regression, radial basis function, and Elman recurrent neural network directly on the training set. The network generalization ability is monitored on the validation set, and its ability to forecast is measured on the test set.

The MLP, GRNN, RBF and Elman models architecture along with the inputs required to feed to the network to perform prediction are shown in figure 3.4, 3.5, 3.6, and 3.7 respectively.

To conduct ANN weather prediction, the basic steps are explained as follow:

- The daily weather data is selected and considered to be portioned into training, validation and testing for the models.
- The initial network architecture is determined.
- The numbers of hidden layers and number of neurons in each layer are determined in order to improve the performance of the neural network.
- To test the trained neural network, performance metrics are determined to be mean absolute error (MSE), mean squared error (MAE), root mean squared error (RMSE), and the correlation coefficient (R).

The available datasets detail for this study, i.e., daily weather data from the selected location, has been discussed in the previous section.

Modeling the available datasets is often the first phase in weather prediction. The data model in the proposed approach uses day wise data (maximum 366 days) over the duration of 25 years for predicting the weather (day wise) of the 26th year. From the 34-years datasets, we have used 25 years data as inputs and the next one year data of a single feature are used as a desired output to build the predictive model.

As mentioned in [S. Singh et.al (2011), L. Al-Matarneh et.al (2014)], we have considered the effect of other weather parameters to predict the future values of single weather variables. That is, to predict the future temperature, the input variables include temperature, rainfall, humidity, evapotranspiration, elevation, latitude, and longitude; to predict future rainfall, the input variables include rainfall, temperature, evapotranspiration, wind speed, elevation, latitude, and longitude; to predict future humidity, the input variables include humidity, temperature, rainfall, elevation, latitude and longitude.

3.3 Model design

We already discussed about the data preparation in the previous section. In this section, we proceed to model building using machine learning techniques.

Machine Learning methods build models based on previous observations which can then be used to predict new values. The built model is a result of a learning process that extracts useful information about the data generation process of the system using the previous observations. The developed model can be applied with new data to predict new values as a result. The resulting predicted values can be then compared to desired values and find the significant variations.

In this study, the selected machine learning methods is artificial neural network which is a powerful data modeling tool that can simulate human thinking and learn from examples.

An ANN consists of many simple elements (artificial neurons), which is the basic unit for information processing. These neurons are linked together based on specific network architecture. Each neuron processes incoming information and may propagate information forward if warranted by its activation function. Each connection link has an associated weight, and each neuron applies an activation function (usually nonlinear) to its net input (sum of weighted input signals) to determine its output signal.

Designing ANN models follows a number of systemic procedures. That is, there are five basics steps: collecting data, preprocessing data, building the network, train the model, and test performance of model as shown in figure 3.3.

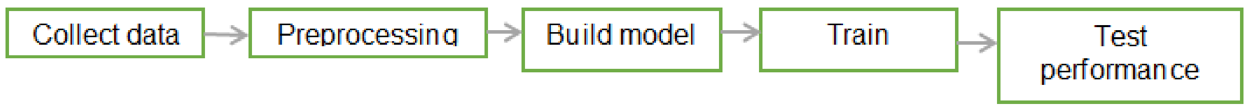


Figure 3.3 Basic flows for building artificial neural network model

3.3.1 Multilayer Perceptron (MLP)

ANN is characterized in principle by a network topology, a connection pattern, neural activation properties, train strategy and ability to process data. This type of network is a supervised network because it requires a desire output in order to learn.

Multilayer feed-forward with Back propagation learning algorithm is a common method of teaching artificial neural networks to perform a given task. Since it is a supervised learning method and a generalization of the delta rule, it requires a teacher that knows the desired output for any input in the training set.

In back propagation, the learning process is the adjustment of weights of a connection between neurons i and j, through which the inputs are linearly related to minimize the error between the network's prediction and the actual response. This procedure iterates to adjust the synaptic weights values when the network gaining extra knowledge after each iteration. Finally, the output is compared with the target output using an error function to find out the significant deviation between predicted values and desired outputs.

In this study, the first model we have used is the feed-forward network with a simple structure involving four layers, seven inputs in the input layer, six neurons in the first hidden layer, three neurons in the second hidden layer, and one neurons in the output layer (7-6-3-1) to perform prediction for the selected variables. The learning function is the back-propagation learning algorithm. The MLP model architecture along with the inputs required feed to the network to perform prediction and the outputs are shown in the figure 3.4.

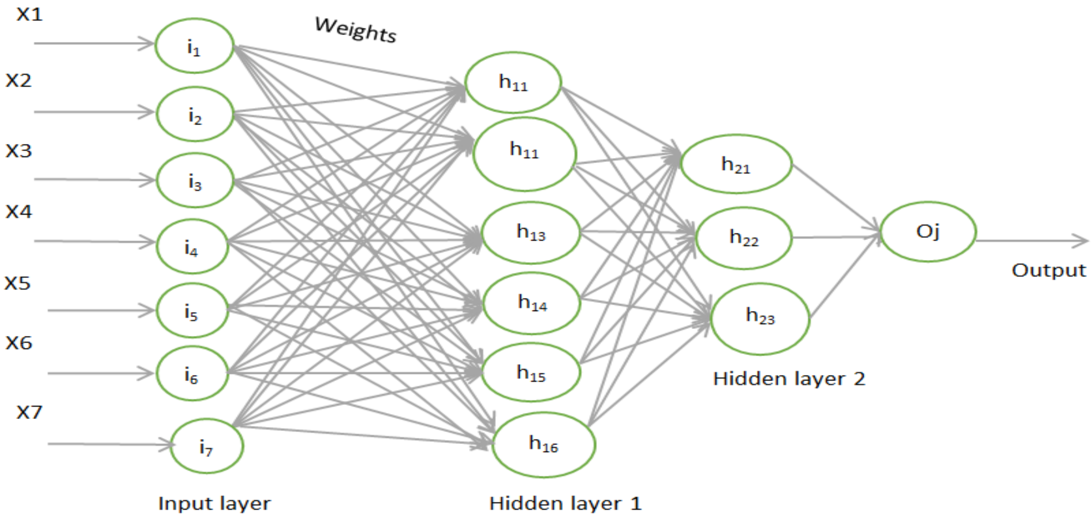


Figure 3.4 Multilayer feed-forward Network Architecture

X_k is weather variable (k=1, 2, 3, 4), X_5 is elevation, X_6 and X_7 are altitude and longitude respectively.

A trained neural network as a computational model can be represented with a simple formula for computing predictions based on learned/tuned weights and the inputs. From the single perceptron, it is possible to modify the formula for multilayer perceptron.

Consider the above network with two hidden layer h_{1i} and h_{2i}:

$$h_{1i}^{out1} = f(\sum w_{1j} x_i + b_{1j}) \tag{3.4}$$

$$h_{2i}^{out2} = f(\sum w_{2j} h_{1i}^{out1} + b_{2j}) \tag{3.5}$$

$$o_j = \Psi(\sum w_{3j} h_{2i}^{out2} + b_{3j}) \tag{3.6}$$

Where w_{1j} - the weight of the link from jth input layer to the first hidden layer,

w_{2j} - the weight of the link between the jth input of the first hidden layer and second hidden layer,

w_{3j} - the weight of the link between the jth input of the second hidden layer and output layer,

b_1, b_2, b_3 are the bias for first hidden layer, second hidden layer, and output layers respectively.

f - Activation function (non-linear)

ψ - Activation function (linear)

h_{1i}^{out1} - First hidden layer output

h_{2i}^{out2} - Second hidden layer output

o_j - Final output of the network

We used Matlab v.2014 for experimenting with the proposed machine learning methods. The functions used in the MLP procedure are *newff* (type of architecture, size and type of training algorithm) and *train*.

(1) *newff* - create a feed-forward back propagation network object and it also automatically initializes the network.

The function is given as:

net = newff (P, T, [sizeArray], { transferFunctionCellArray }, { trainingAlgorithm })

where P is the input matrix, T is desired output for each input and *sizeArray* is an array that contains size for each hidden layer.

transferFunctionCellArray is the Cell Array that contains strings representing the transfer functions for each layer (not including input layer). These functions include logarithmic-sigmoidal (logsig), tangential-sigmoidal (Tansig) and Linear (purelin).

t*rainingAlgorithm* denotes a string representing the training algorithm for the network. These include Resilient Back propagation (trainrp), Levenberg-Marquardt (trainlm), etc.

net signifies the neural network generated by *newff().*

(2*) train( )* is a function that used to train the network whenever it is called.

Training function: net1 = train (net, P, T)

where net - the initial MLP network to be trained, P: training data set, T: desired output for each input, default = zeros, and net1: new network object.

Training stops when any of the following conditions is met: (1) the maximum number of epochs is reached, (2) performance has been minimized to the goal, (3) the maximum amount of time has been exceeded, or (4) validation performance has been increased more than max_fail times.

3.3.2 Radial Basis Function (RBF)

In the previous chapter, we have discussed in some details about the concept of radial basis function and its application. Here the function will be modeled to train using the neural network strategies.

RBF is an alternative to the more widely used MLP network and consumes less computer time for network training [S. A. Hannan et.al, 2010]. In this method, a number of hidden nodes with RBF activation functions are connected in a feed forward parallel architecture. The parameters associated with the RBFs are optimized during the network training.

The function learns from the training set that contains elements which consist of paired values of the independent input variable and the dependent output variable.

There are different basis functions. The most widely used function is the Gaussian function:

$$\phi\left(\left\|x_i - x_j\right\|\right) = \exp\left(-\frac{\left\|x_i - x_j\right\|^2}{2\sigma^2}\right) \tag{3.7}$$

RBF consists of three layers: an input layer, a hidden (kernel) layer, and an output layer. The nodes within each layer are fully connected to the previous layer. The input variables are each assigned to the nodes in the input layer and the input layer distributes the inputs to the hidden layers. Each node in the hidden layer is the radial function and its dimension is same as of the input data.

In the network, the output is calculated by a linear combination.

$$f(\mathrm{x_i}) = \sum_{j=1}^{N} w_j \phi\left(\left\|x_i - c_j\right\|\right) + b \tag{3.8}$$

The RBF predictive model for this study is created as shown in the figure 3.5. The model can accept seven inputs and return one output.

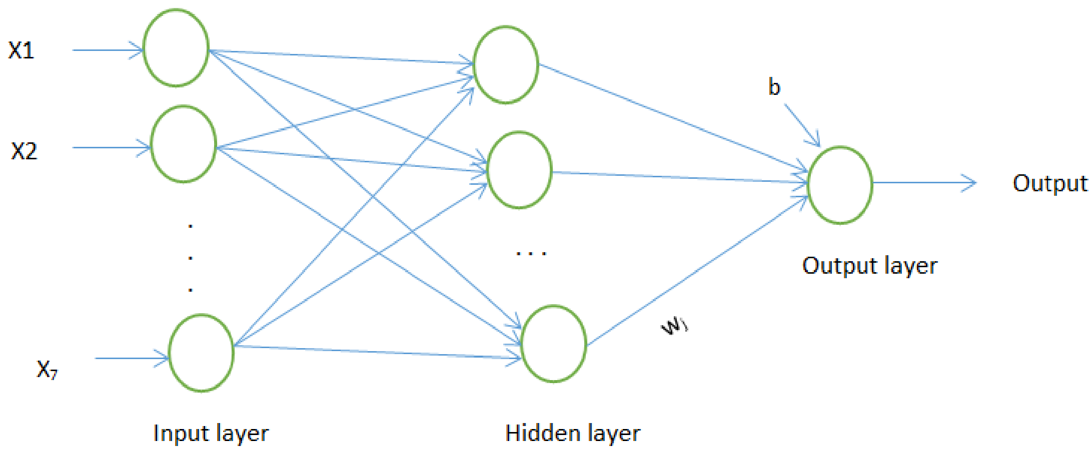


Figure 3.5 RBF Neural network architecture

X_m is weather variable (m=1, 2, 3, 4), X_5 is elevation, X_6 and X_7 are altitude and longitude respectively.

Again we used Matlab to evaluate the RBF algorithm for our purpose. Here the net input to the *radbas* transfer function in MatLab is the vector distance between its weight vector **w** and the input vector **x**.

The parameters of the RBF network include: the centers of the RBF activation functions, the spreads of the Gaussian RBF activation functions, and the weights from the hidden to the output layer.

RBF has two built-in functions in matlab: *newrbe* and *newrb*.

The *newrbe* function takes matrices of input vectors, desired output vectors, and a spread constant for the radial basis layer, and returns the net as output of the network.

The second function is *newrb*. This function iteratively creates a radial basis network one neuron at a time. The *newrb* adds neurons to the hidden layer of a radial basis network until it meets the specified mean squared error goal or a maximum number of neurons have been reached.

net = newrbe (Input, Target, Spread)

The *newrb* function takes matrices of input and desired output as vectors, and parameters of goal and spread, and returns the desired network.

The design method of *newrb* is similar to that of newrbe. The difference is, *newrb* creates neurons one at a time.

net = *newrb* (P, T, goal, spread, MN, NN) ,

where P is R-by-N matrix of N input vectors, T is S-by-N matrix of N desired output vectors, Goal is the Mean squared error goal (default = 0.0), MN is the Maximum number of neurons, and NN is the number of neurons to add between displays (default = 25).

We can call *newrb* function with different spreads to find the best value for a given problem. As the spread becomes loftier, the function approximation befits smother. And a lot of neurons are required to fit a fast-changing function. Too small a spread means many neurons are required to fit a smooth function, and the network might not generalize well.

3.3.3 Generalized regression neural network (GRNN)

Generalized Regression Neural Network is a kind of Radial Basis Function network with a one pass learning algorithm and highly parallel structure [G. Dudek, 2011].

GRNN provides estimates of continuous variables [Donald F. Specht, 1991]. The algorithm provides smooth approximation of a target function even with sparse data in a multidimensional space.

This model has four layers: input, pattern (radial basis layer), summation and output as shown in figure 3.6.

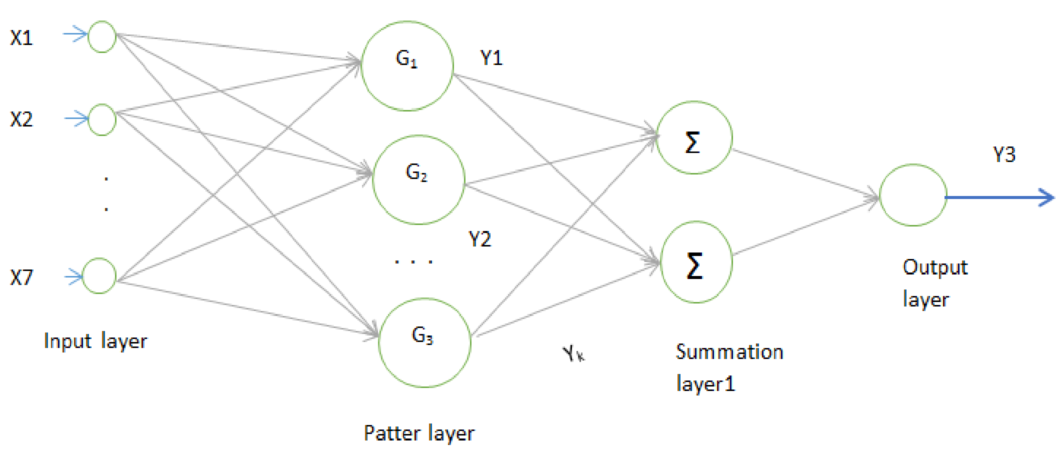


Figure 3.6 GRNN architecture (B. Kim et al, 2004]

The input layer transfers an input data into the next layer (pattern layer). The number of neurons in the pattern layer is equal to the number of training samples. For each input data, all pattern layer neurons compute the Euclidian distance between the input vector and the corresponding neuron location. These distances are assumed to be an activation function (a Gaussian kernel in this case). The summation layer consists of two neurons that calculate the nominator and denominator respectively. Each of these neurons computes a weighted sum of the output from a previous layer. The weights correspond to the links between the neurons. Each link from the pattern layer neuron to the nominator neuron is a target value associated with the corresponding neuron location.

Each neuron of the pattern layer uses a radial basis function as an activation function. This function is commonly taken to be Gaussian:

$$G_j(x) = \exp\left(-\frac{\|x - c_j\|^2}{s_j^2}\right) \tag{3.9}$$

where c_j is a center vector, s_j is a smoothing parameter or bandwidth and ||.|| is the Euclidean norm. Each training vector is represented by one pattern neuron with the center $c_j = x_j$, j = 1, 2, …, N, where N is a number of training points. The neuron output expresses the similarity between the input vector x and the j^{th} training vector. So the pattern layer maps the n-dimensional input space into N-dimensional space of similarity. The function of GRNN network in matlab software [M.H. Beale, et al, 2014]:

net = newgrnn(P, T, spread),

where P: RxN matrix of N input vectors, T: SxN matrix of N desired output vectors, Spread: spread of radial basis functions, default=1.0, and net: returns a new generalized regression neural network.

When the spread becomes large, the function approximation will be smoother. To fit data very closely, spread must be smaller than the typical distance between input vectors. To fit the data more smoothly, use a larger spread.

3.3.3.1 Elman recurrent neural Network

The base architecture of Elman networks are two-layer back propagation network with the addition of a feedback connection from the output of the hidden layer to its input. This feedback path allows network to learn to recognize and generate temporal patterns, as well as spatial patterns.

The activations can flow round in a loop that enables the networks to do temporal processing and learn sequences or perform temporal association or prediction.

Elman recurrent neural network architectures can have many different forms. One common type consists of a standard Multi-Layer Perceptron (MLP) plus added loops. These can exploit the powerful non-linear mapping capabilities of the MLP [R. Nkoana, 2011]. Unlike feed forward networks, recurrent networks have the ability to "remember" past events and they are generally regarded to be more effective for time series prediction problems.

The fourth used model is the Elman recurrent network with a simple structure involving four layers, which is exactly the same as MLP mode architecture with added feedback connection to the hidden layers. The learning function is the back-propagation learning algorithm. The Elman model architecture along with the inputs required feed to the network to perform prediction and the outputs are shown in the figure 3.7. That is, this model can accept seven external inputs, loop hidden layer output as input and the return one output.

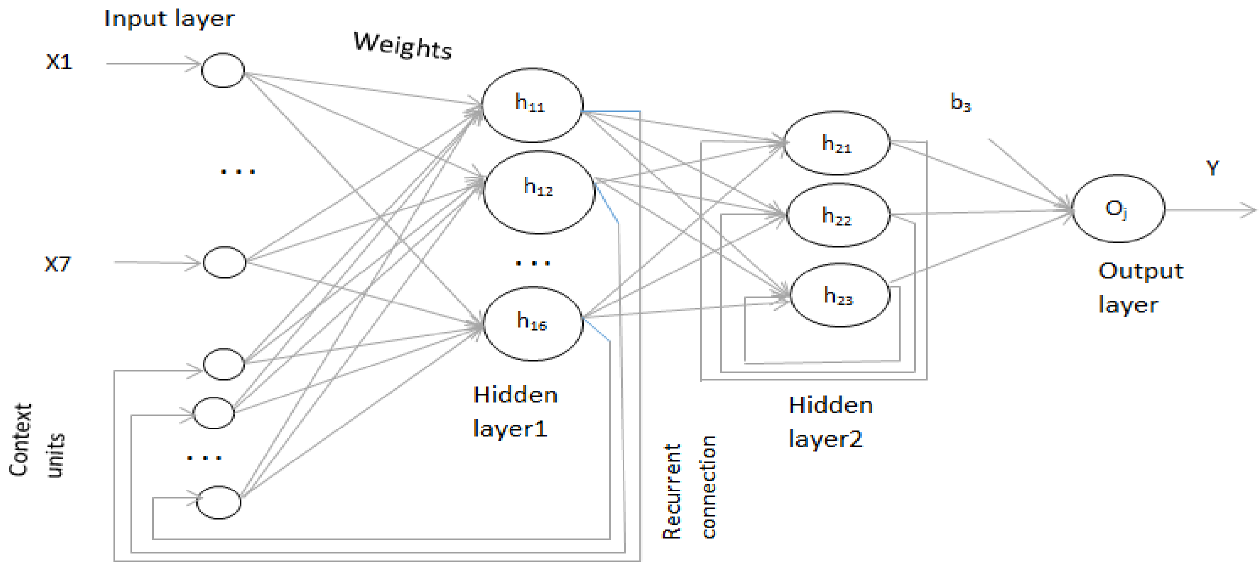


Figure 3.7 Architecture of Elman network

X_k is weather variable (k=1, 2, 3, 4), X_5 is elevation, X_6 and X_7 are altitude and longitude respectively.

A trained neural network as a computational model can be represented with a simple formula for computing predictions based on learned/tuned weights and the inputs. From the single layer model, it is possible to modify the formula for multilayer model. Consider the above network with two hidden layer h_{1i} and h_{2i}:

$$y_{1j}(t) = f(\sum x_i(t)v_{1j} + \sum y_h(t-1)\,u_{1h}) + b_1) \tag{3.10}$$

$$y_{2j}(t) = f(\sum y_{1j}(t)v_{2j} + f(\sum y_k(t-1)\,u_{2k}) + b_2) \tag{3.11}$$

$$o_j = \Psi(\sum w_{3j} y_{2j} + b_3) \tag{3.12}$$

Where x_i - the input variables and v_{1j} - the weight of the link from jth input layer to the first hidden layer.

u_{1h} - The weight of the link from hth input layer to the first hidden layer. It is the weight of the link from first hidden layer to input layer.

y_h - is the output of first hidden layer at time t-1.

v_{2j} - is the weight of the link from jth input layer to the second hidden layer.

u_{2k} - The weight of the link from kth input layer to the second hidden layer. It is the weight of the link from second hidden layer to input layer.

y_k - is the output of second hidden layer at time t-1.

w_{3j} - The weight of the link between the jth input of the second hidden layer and output layer.

b_1, b_2, b_3 are the bias for first hidden layer, second hidden layer, and output layers respectively.

ƒ - Activation function (non-linear)

ψ - Activation function (linear)

y_{1j} - First hidden layer output

y_{2j} - Second hidden layer output

o_j - Final output of the network

We used Matlab v.2014 for experimenting with the proposed machine learning methods. The functions used in the Elman recurrent network procedure are *newelm* (type of architecture, size and type of training algorithm) and *train*.

The Elman recurrent model is created with the function newelm() . The function is given as: net = newelm(P, T, [sizeArray], { transferFunctionCellArray }, { trainingAlgorithm })

where *net*: neural network generated by newelm(.),

P: the input matrix,

T: desired output for each input.

sizeArray: an array that contains size for each hidden layer.

transferFunctionCellArray: Cell Array that contains strings representing the transfer functions for each layer (not including input layer). These function include logarithmic-sigmoidal (logsig), tangential-sigmoidal (Tansig) and Linear (purelin).

trainingAlgorithm: A string representing the training algorithm for the network. These include Resilient Back propagation (trainrp), Levenberg-Marquardt (trainlm), etc.

The training function "*trainingAlgorithm*" can be any of the back propagation training functions such as TRAINGD, TRAINGDM, TRAINGDA, TRAINGDX, etc.

Algorithms, which take large step sizes, such as TRAINLM, and TRAINRP, etc., are not recommended for Elman networks. Because of the delays in Elman networks, the gradient of performance used by these algorithms is only approximated making learning difficult for large step algorithms.

The model results hold different parameters such as epoch, time, performance, learning rate while training in MLP and Elman recurrent network.

Epoch is number of iterations taken by a network to train data, *time* shows time interval taken by a network while training data and performance *is the* performance measure that shows the error value given by the network. If the value of this performance is small, it shows that the network works properly, otherwise more training will be required. And *gradient* **(**learning rate) is a constant used in error back-propagation learning that affects the speed of learning. The smaller the learning rate, the more steps it takes to get to the stopping criterion and training will be slow. If learning rate is large, then learning will be fast but may not give better results.

3.4 Evaluation metrics

The prediction performance can be measured with different performance metrics that are appropriate to evaluate the results of the models.

In this study, the models accuracy is measured with the performance criteria of mean square error, mean absolute error, root mean square error, correlation coefficient, and confusion matrix.

Mean Squared Error (MSE) is one of the most commonly used measures of success for numeric prediction. The value of MSE is computed by taking the average of the squared differences between each predicted value and its corresponding actual value.

MSE measure is defined by:

$$\text{MSE (T, P)} = \frac{1}{n}\sum_{i=1}^{n}(\text{T(i)} - P(i))^2, \qquad (3.13)$$

Mean Absolute Error (MAE): The MAE measures the average magnitude of the errors in a set of forecasts. It measures accuracy for continuous variables. The MAE is a linear score which means that all the individual differences are weighted equally in the average. MAE measure is defined by:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|\text{T(i)} - \text{P(i)}| \qquad (3.14)$$

Root Mean Squared Error (RMSE): The RMSE is a quadratic scoring rule which measures the average magnitude of the error. That is, the difference between forecast and corresponding observed values are each squared and then averaged over the sample. Finally, the square root of the average is taken. Since the errors are squared before they are averaged, the RMSE gives a relatively high weight to large errors. This means the RMSE is most useful when large errors are particularly undesirable.

RMSE is defined as:

$$\text{RMSE} = \left[\sqrt{\sum_{i=1}^{n}\frac{(T(i)-P(i))^2}{n}}\right] \qquad (3.15)$$

where T = (T(1), T(2), … ,T(n-1), T(n)) is the vector of actual values, P= (P(1), P(2), …, P(n-1), P(n)) is the vector of predicted values and n is the number of samples.

Correlation Coefficient: This criterion measures the statistical correlation between the predicted and actual values. It is unique in that it does not change with a scale in values for the test cases. A higher number means a better model, R= 1 meaning a perfect statistical correlation, and R= 0 meaning there is no correlation at all. A correlation coefficient greater than 0.8 is generally described as strong, whereas a correlation coefficient less than 0.5 is generally described as weak[R. Beaumont, 2012].

A confusion matrix is a table that is often used to describe the performance of a prediction or classification model on a set of test data for which the true values are known. The confusion matrix is represented by a matrix in which each row represents the instances in an actual class, while each column represents in a predicted class.

The confusion matrix shows the accuracy of the predictor or classifier as the percentage of correctly predicted or classified patterns in a given class divided by the total number of patterns in that class. The overall or average accuracy of the classifier is also evaluated by using the confusion matrix.

In the Confusion Matrix, for each cell in the matrix we have fields as True Positives (correctly predicting a label), False Positives (Falsely predicting a label), False Negatives (Missing and incoming label) and True Negatives (Correctly predicting the other label).

CHAPTER FOUR

4. EXPERIMENT AND DISCUSSION

4.1 Introduction

In this section, the results of the experiments have been explained and presented based on the model training simulations experimented. The objective of this study is to develop a model that is used to predict future values of temperature, rainfall, and relative humidity of a selected location in northern Ethiopia region in which drought occurred most frequently.

Experiments have been carried out to evaluate the performance of the presented method. These experiments were conducted using datasets from 143 weather stations in the northern part of Ethiopia.

The datasets were collected from National Centers for Environmental Prediction (NCEP) Climate Forecast System Reanalysis (CFSR).

The whole data set covers the period from January 01, 1979 to December 31, 2013, a total of 12988 daily satellite records.

Since atmosphere pattern is a complex and nonlinear system, traditional methods are not effective and efficient. Artificial Neural Networks, including MLP, GRNN, RBF and Elman recurrent networks, are influential methods for resolving such problems.

For each model, an initial acceptable architecture was selected to repeatedly modify the hidden layers and number of neurons in order to determine a model that can produce better performance.

The first model we used was multilayer feed-forward (MLP) with back propagation learning algorithm. The back propagation learning algorithm was used for accuracy correction by adjusting the weight.

We configure the network with seven inputs, two hidden layers and one output layer architecture, and produce one output. The numbers of neurons were six for the first hidden layer, three for second hidden layer, and one for output layer which results better performance. The transfer function used was sigmoid function in the hidden layers and linear function in the output layer.

The second used algorithm was general regression neural network. In GRNN, the input layer transfers an input data into the next layer (pattern layer). The number of neurons in the pattern layer is equal to the number of training samples, with each neuron corresponding to a training sample. For each input data, all pattern layer neurons compute the Euclidian distance between the input vector and the corresponding neuron location.

The third neural network model was Radial basis function (RBF), which is a special class of single hidden layer feed forward neural network with supervising learning. In this model, the input layer distributes inputs to the hidden layer. Each node in the hidden layer is a radial function, and its dimension is same as of the input data.

The last model was Elman recurrent network. This network is similar to two-layered back-propagation MLP with the addition of a feedback connection from the output of the hidden layer to its input. All layers except the last have recurrent weights.

The results of the models are compared using the performance metrics MAE, MSE, RMSE and Correlation coefficient (R).

After developing a model, we get their results in the form of graphs and values. Results are sub-divided according to the data sets used to check the above mentioned models. These are training sets, testing sets and validation sets.

4.2 Prediction

Prediction was conducted using the above mentioned four neural network approaches described.

Thus, the description and performance analysis of each network approach are given in detail as follows.

4.2.1 Temperature prediction

To predict the future temperature of the given location, the input variables (predictors) for the model in the input layer were daily temperature, rainfall, relative humidity, evapotranspiration, elevation, and coordinates.

Based on the result from the experiment, the better result was recorded with multilayer feed-forward with back propagation learning algorithm.

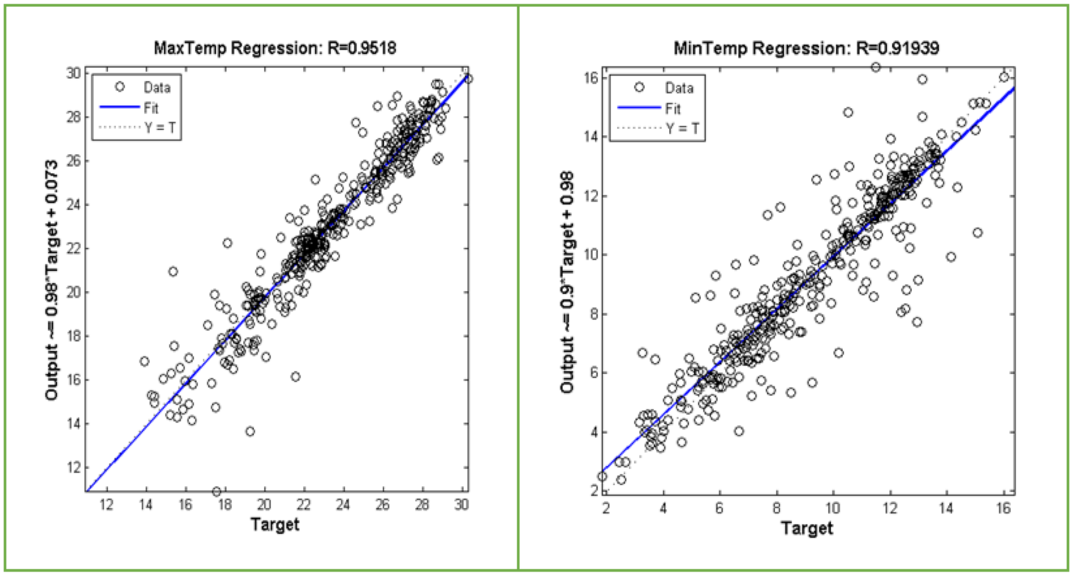


Figure 4.1 Regression plot displays the network outputs with respect to targets (Ebinat weather station temperature data)

In figure 4.1, the regression diagram shows that the network output with respect to desired outputs fall more near to the desired diagonal line with the correlation coefficient of R=0.95 in maximum temperature prediction and R=0.91 in minimum temperature prediction. This correlation coefficient measures the strength and direction of a relationship between the predicted and actual values. A higher correlation value means a better model, with a 1 meaning a perfect statistical correlation and 0 meaning there is no correlation at all. Therefore, this regression coefficient result for both variables indicates that the network outputs have strong correlation to the desired outputs.

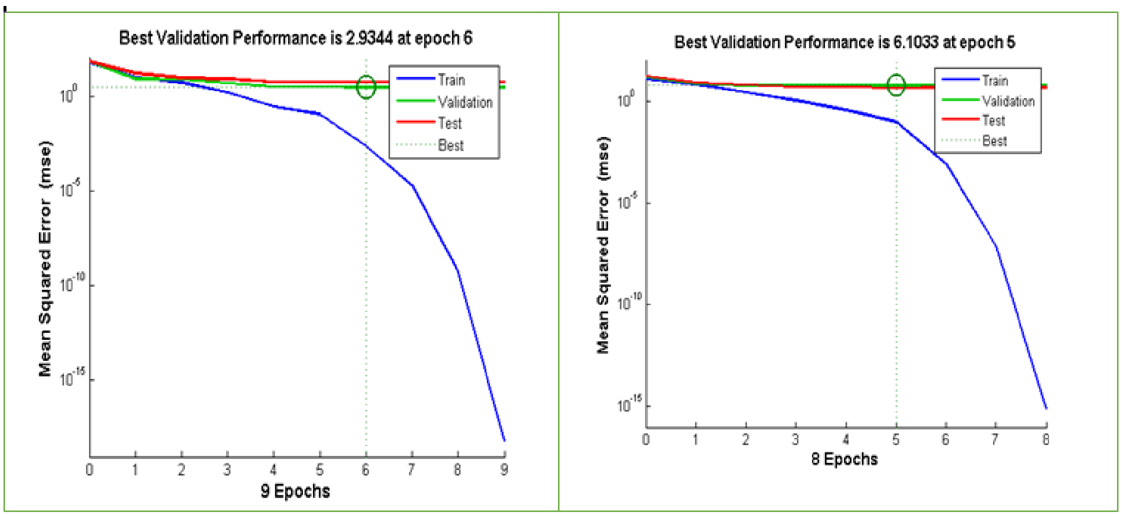


Figure 4.2 Best validations Performance

In figure 4.2, the performance diagram shows that the model best validation and testing performance is obtained at iteration 6 in maximum temperature and at iteration 5 in minimum temperature. This indicates that training error decreases near to 0 at epoch 9 and 8 for maximum and minimum temperature respectively.

First we have used 10 years datasets to train the predictive model, and then increase the training datasets up to 25 years step by step. When increased the datasets, the model performance error decrease. However, there was no more change on validation and testing error after best validation performance; but training error decreased to 0 after iteration 6 and 5 for maximum and minimum temperature respectively.

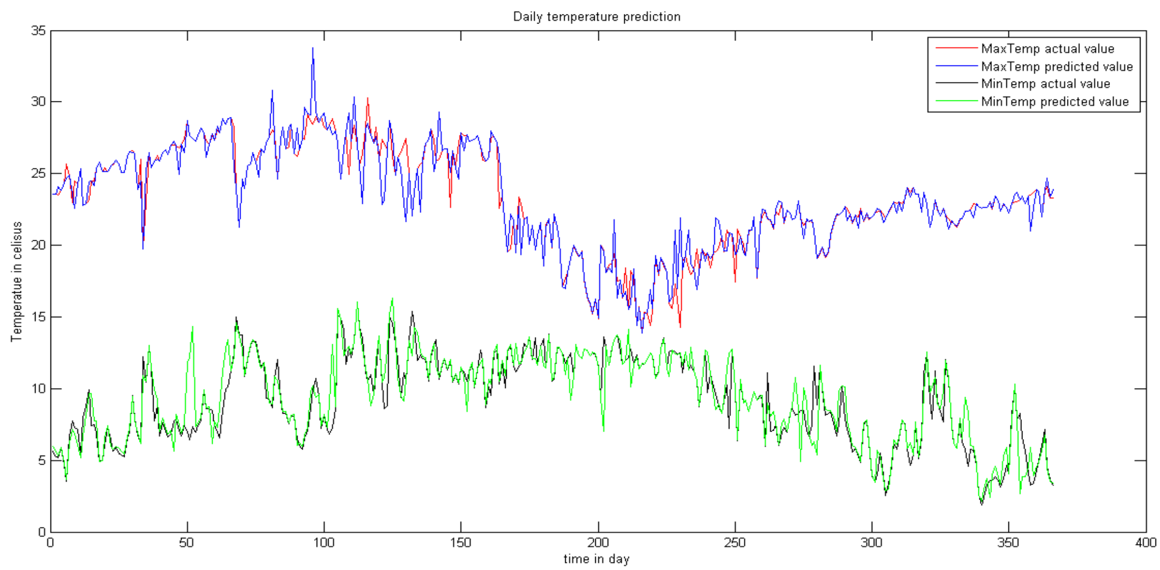


Figure 4.3 Observed versus predicted temperature (Ebinat weather station temperature data)

As shown in figure 4.3, the better prediction result has been obtained for daily maximum and minimum temperature using multilayer feed-forward network with back propagation learning algorithm. The red line indicates the actual value and the blue line is the predicted value in maximum temperature. The black line represents the actual value, whereas green line indicates the predicted value in minimum temperature.

Table 4.1 Model architecture and performance measures for temperature prediction

Location	Variable	Metrics	Models			
			MLP	GRNN	RBF	ELMAN
			7-6-3-1	7-366-1	7-366-1	7-6-3-1
		Mse	1.0751	4.704	7.8954	1.5246

Ebinat Station	maxTemp	Mae	**0.568**	1.5012	2.1191	0.9450
		Rmse	1.0368	2.169	2.8099	1.2347
		R	**0.95**	0.69	0.30	0.94
	MinTemp	Mse	1.5630	4.33	21.746	1.6658
		Mae	**0.7807**	1.6087	3.7706	0.8919
		Rmse	1.2502	2.081	4.6633	1.2907
		R	**0.91**	0.69	0.23	0.91

Table 4.1 shows the model architecture and performance measure in terms of Mse, Mae, Rmse, and correlation coefficient (R) for maximum and minimum temperature datasets.

It is clearly shown that all performance metrics have less value with MLP model for both maximum and minimum temperature. Among the metrics, MAE value is the lowest.

Therefore, the performance obtained with the above mentioned criteria indicates that the MLP model with network architecture 7-6-3-1 is suitable for maximum and minimum temperature prediction.

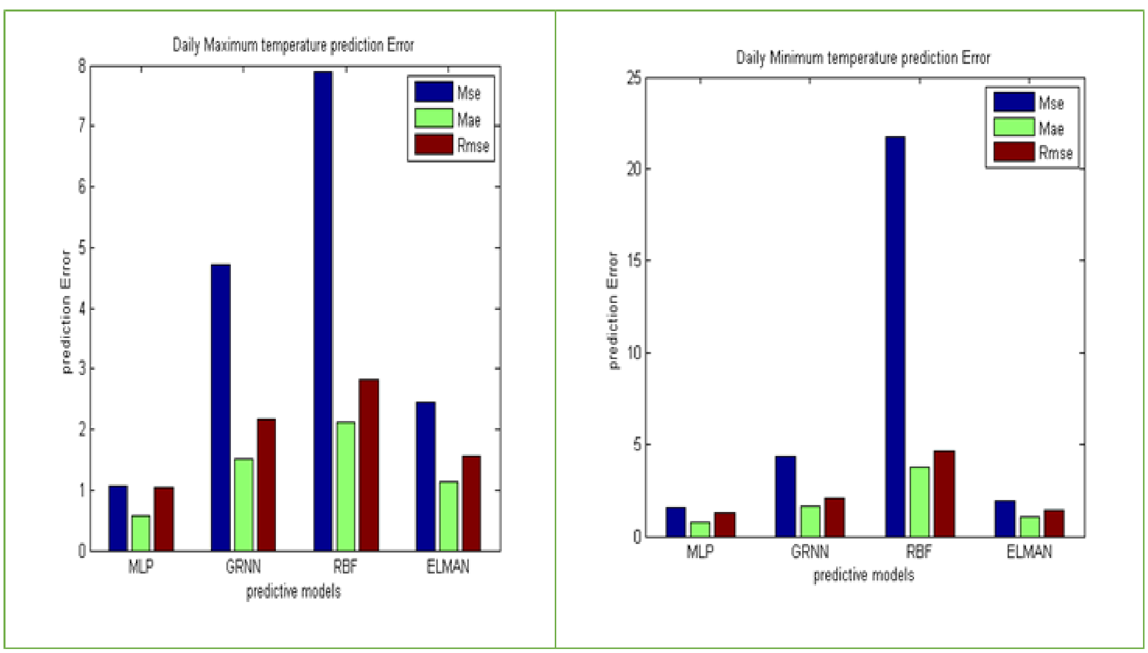


Figure 4.4 Error comparison for temperature prediction

As illustrated in the above bar graph, the error values corresponding to maximum and minimum temperature are small with MLP and high along with FBF models. So MLP model is a suitable model for both variables. The blue color indicates the MSE values, green color indicates the MAE values, and the brown one indicates the RMSE values.

4.2.2 Rainfall prediction

The nature of the rainfall dataset is different from the temperature dataset because its variation is high between each consecutive value. To predict the future rainfall of a given location, the input variables for the model in the input layer were daily rainfall, mean temperature, relative humidity, wind speed, elevation and coordinates.

An artificial neural network model is applied for weather datasets to predict rainfall and the result of the model performance are described below.

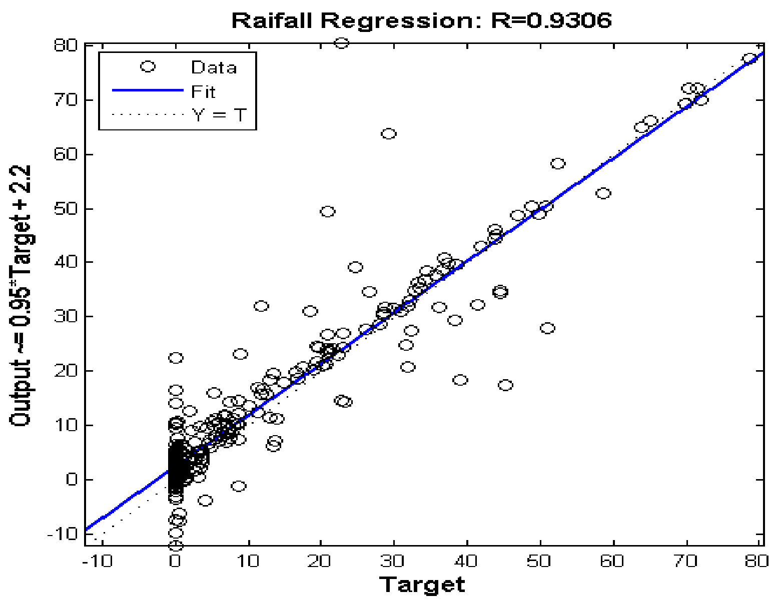


Figure 4.5 Regression plot displays the network outputs with respect to desired output (Belesa weather station rainfall data)

In figure 4.5, the regression result shows that the desired outputs verses network outputs have strong correlation for rainfall datasets with MLP model. This indicates that the model has strong fit for rainfall variables in terms of the regression coefficient value which is R= 0.93.

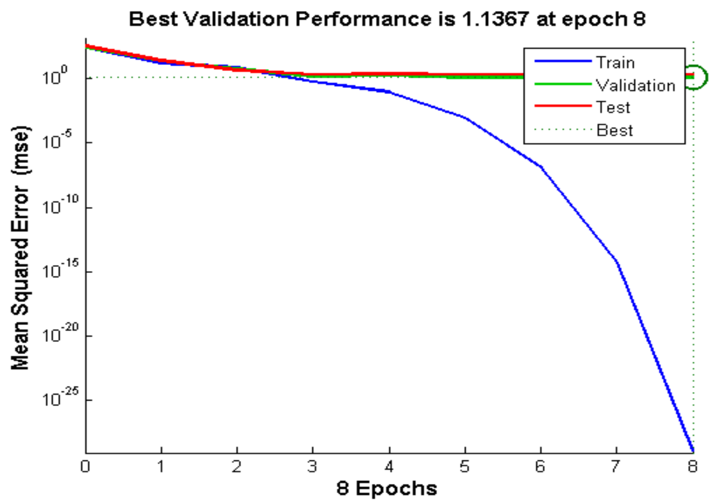


Figure 4.6 Best validations Performance

As shown in figure 4.6, the training performance decreases to 0 at epoch 8. Though the model best validation and testing performance is occurred at iteration 8, the validation and testing performance is not good at this iteration point. This may be due to higher variation of consecutive values of rainfall parameter.

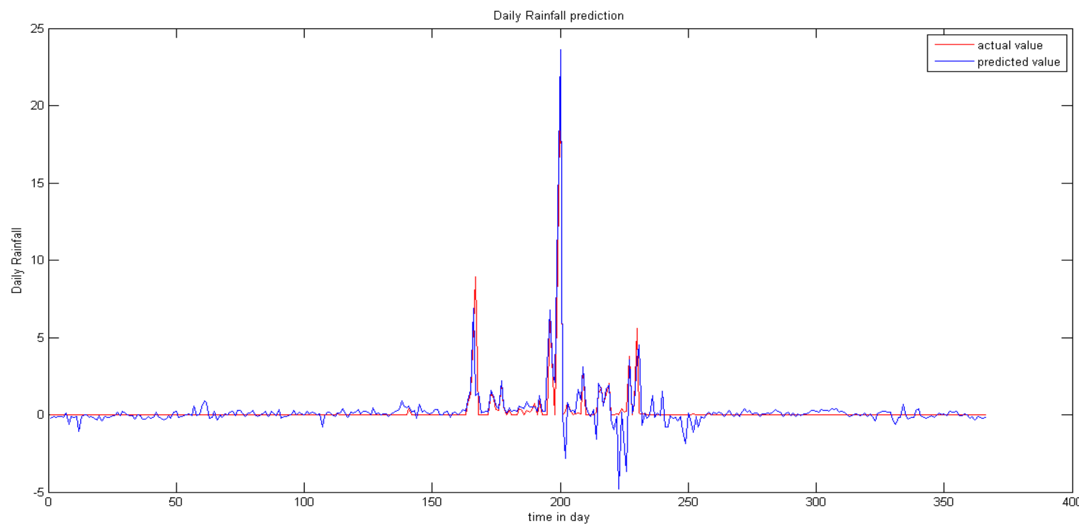


Figure 4.7 Observed versus predicted daily rainfall (Belesa weather station data)

The performance of the model for rainfall prediction has been evaluated with the performance metric MSE, MAE, RMSE, and R.

As illustrated in figure 4.7, the model performance for rainfall data is not much good; however, the better prediction has been carried out using MLP network with the model architecture 7-6-3-1.

Table 4.2 Model architecture and performance measures for rainfall prediction

Location	Variable	Metrics	Models			
			MLP	GRNN	RBF	ELMAN
			7-6-3-1	7-366-1	7-366-1	7-6-3-1
Belesa Station	Rainfall	Mse	1.2822	16.3777	42.2503	52.0288
		Mae	**1.251**	1.6212	2.2494	3.9099
		Rmse	1.4323	4.0469	6.5000	7.2131
		R	0.93	0.75	0.40	0.91

Table 4.2 displays the model architecture and performance measure in terms of Mse, Mae, Rmse, correlation coefficient R for rainfall datasets.

As shown in table 4.2, it is clear that all performance metrics have less value with MLP model for rainfall variable. However, the lowest value is obtained for MAE.

The model performance obtained indicates that the MLP model with network architecture 7-6-3-1 has acceptable result for rainfall variable prediction.

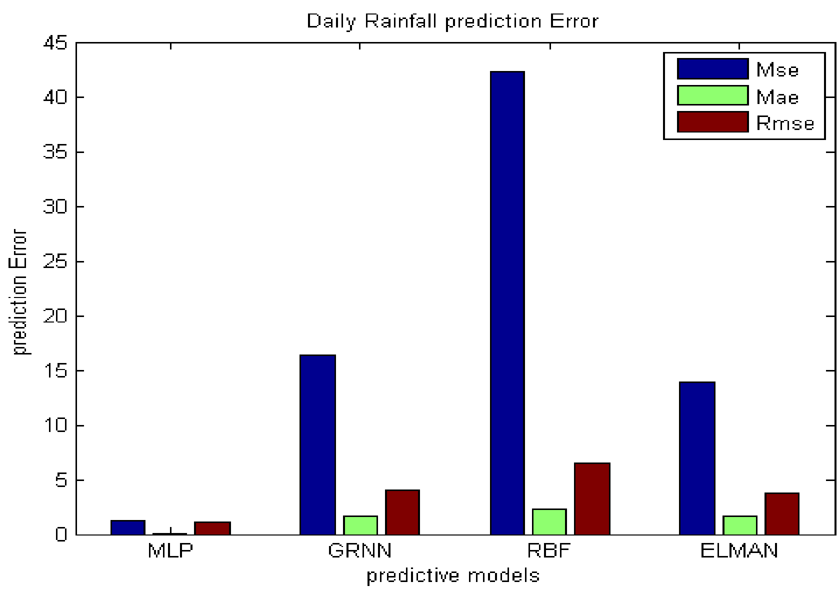


Figure 4.8 model Error comparison for rainfall dataset prediction

Figure 4.8 shows that the error values corresponding to rainfall datasets are small with MLP and high along with RBF model. Even though MLP model has acceptable result for rainfall variables prediction, it clearly indicates that the result is not better when comparing to the result for temperature variables. The blue color indicates the MSE values, green color indicates the MAE values, and the brown one indicates the RMSE values.

4.2.3 Relative humidity prediction

Relative humidity is a measure of the amount of water in air as compared to the maximum amount of water the air can absorb.

As explained in chapter two, relative humidity helps us to determine human and animal comfort levels. Without humidity, there would be no clouds, no precipitation, and no fog. To predict the future relative humidity of the given location, the input variables for the model in the input layer were daily relative humidity, mean temperature, rainfall, evapotranspiration, elevation, and coordinates.

The experimental result indicates that all used models except RBF are suitable for humidity prediction. The detailed experimental results are described below.

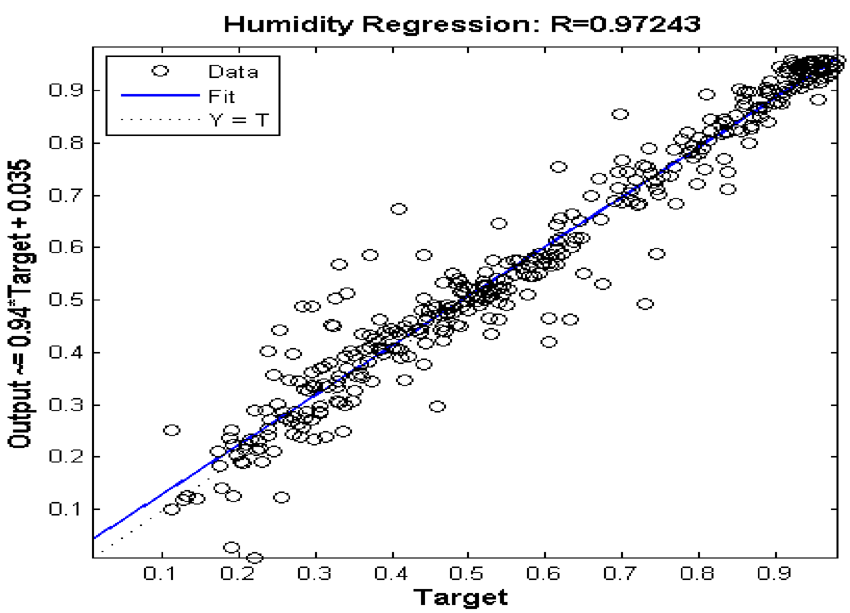


Figure 4.9 Regression plot displays the network outputs with respect to desired outputs (Ebinat station data)

Figure 4.9 regression diagram shows that the network output with respect to desired outputs becomes more close to diagonal line which indicates that the model output has strong correlation to the desired output with the correlation coefficient R=0.97 for humidity variable prediction.

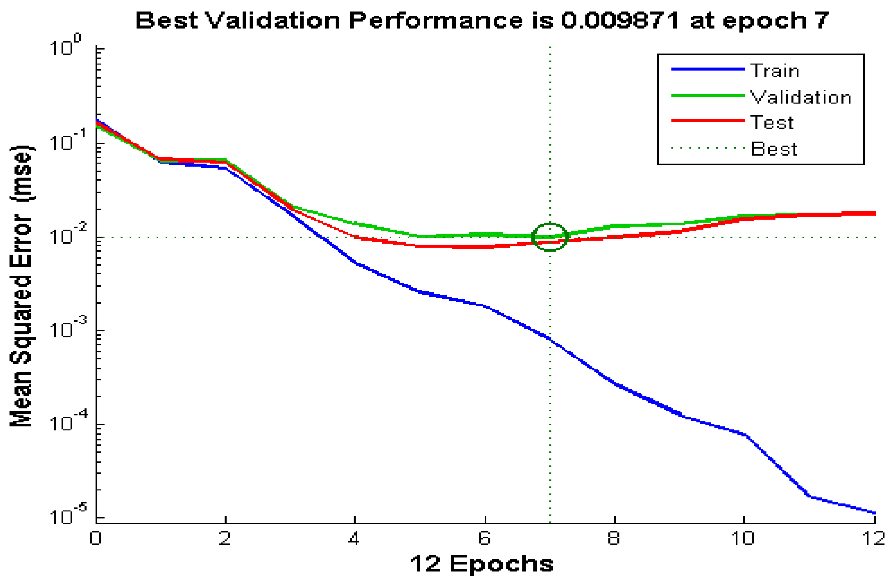


Figure 4.10 Best validations Performance

In figure 4.10, the diagram shows that the model best validation performance is 0.0098 at epoch 7. The training performance is decreased to 0 at iteration 12. The validation and testing performance also decrease up to iteration 7. However, validation and testing performance is far from training performance after iteration 7.

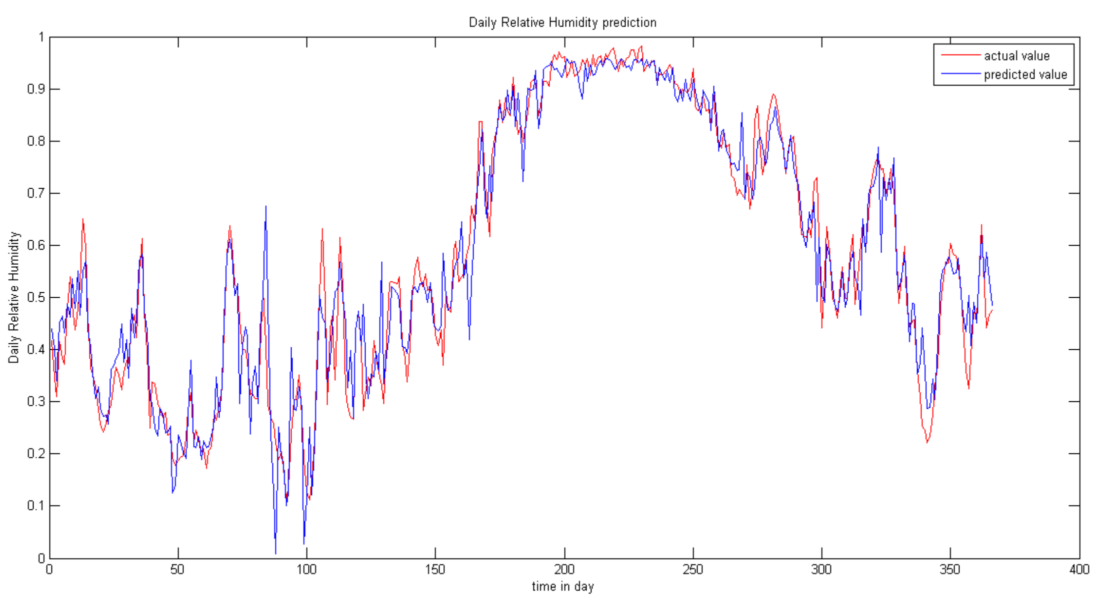


Figure 4.11 Observed versus predicted daily relative humidity (Ebinat station)

As shown in the figure 4.11, multilayer perceptron with back propagation learning algorithm returns good result for humidity variable prediction though all tested models are suitable for this variable prediction except RBF. Moreover, the best model performance was obtained by MLP in all performance criteria.

Table 4.3 Model architecture and performance measures for Humidity prediction

Location	Variable	Metrics	Models			
			MLP	GRNN	RBF	ELMAN
			7-6-3-1	7-366-1	7-366-1	7-6-3-1
Ebinat Station	Relative humidity	Mse	**0.0054**	0.01	30.7455	0.0056
		Mae	0.0507	0.0771	4.5949	0.0559
		Rmse	0.0735	0.10	5.5449	0.0749
		R	**0.97**	0.91	0.2	0.96

Table 4.3 shows the model architecture and performance measure in terms of Mse, Mae, Rmse, and correlation coefficient R for humidity datasets.

Table 4.2 shows that all performance metrics have less value with MLP model. Moreover, the Mse value is the lowest for all models except RBF. Therefore, the performance obtained with the given criteria indicates that the MLP model with network architecture 7-6-3-1 has better performance for humidity variables prediction.

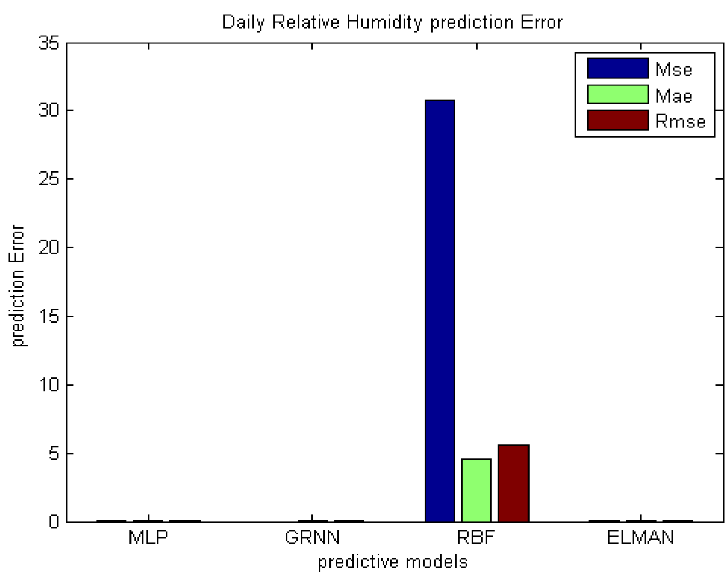


Figure 4.12 Error comparison for relative humidity prediction

As illustrated in figure 4.12, the error values are very small with MLP, GRNN, and ELMAN models. But error values are high along with RBF model. The smallest error value is returned by MLP model with in all performance metric. The blue color indicates the MSE values, green color indicates the MAE values, and the brown one indicates the RMSE values.

4.2.4 Overall model performance

Different types of machine learning techniques were investigated including their network architectures and variations of associated learning rules to determine the parameters that provide the optimal prediction method for the selected location weather station datasets. We have used the model MLP, GRNN, RBF, and Elman to predict the three selected weather elements.

The overall accuracy of prediction model based on multilayer feed-forward with back propagation learning algorithm is better to predict all the three weather variables than the general regression, radial basis function and Elman recurrent neural network model.

We have used different performance metrics to measure the result of all the models used. In this study, the performance measures used to evaluate each model include MSE, MAE, RMSE and R.

Among used performance criteria, MAE has the lowest value with MLP model for temperature and rainfall variables. But for humidity variable, MSE has the smallest value with the same model.

Table 4.4 MLP Model architecture and performance measures for overall prediction

Location	Variable	Metric	Model			
			MLP	GRNN	RBF	Elman
			7-6.3.1	7-366-1	7-366-1	7-6.3.1
Ebinat Station	MaxTemp	Mae	**0.5680**	1.5012	2.1191	0.945
		R	**0.95**	0.69	0.30	0.94
	minTemp	Mae	**0.7807**	1.6087	3.7706	0.8919
		R	**0.91**	0.69	0.23	0.91
Belesa station	Rainfall	Mae	**0.1251**	1.6212	2.2494	3.9099
		R	**0.93**	0.75	0.40	0.91
Ebinat station	Humidity	Mse	**0.0054**	0.10	30.7455	0.0056
		R	**0.97**	0.91	0.20	0.96

As shown in table 4.4, for all the three weather variables prediction, multilayer feed-forward neural network with back propagation learning algorithm is suitable to predict the future weather variables. Moreover, the multilayer feed-forward with back propagation model performance is better for temperature and humidity variables than rainfall variables in terms of the selected performance criteria. This may be due to high variation of rainfall consecutive values.

CHAPTER FIVE

5. CONCLUSION AND RECOMMENDATION

5.1 Conclusion

Climate information is essential for agricultural planning, economic, transportation, water resource management, and other environmental assessments. Knowing the coming weather information helps us to be safe from unexpected hazards such as drought, floods or windstorm that may cause losses of life and properties due to weather variation.

Weather prediction plays an important role in our daily activities by predicting what the weather will be tomorrow and it is reflected in a wide area of applications in our life, so we can prevent huge damages by forecasting the coming weather condition or get benefits from the forecasting activities.

Accurate weather forecasts can guide an airport control tower regarding what information needs to be communicated to airplanes that are taking off or landing. It can also lead a farmer to the best time to cultivate various crops, and also to predict natural disasters.

Artificial neural networks, including MLP, GRNN, RBF and Elman recurrent, are the influential data modeling tool that can solve complex and non-linear system. This type of network can create a model that correctly maps the input to the output using historical data so that the model can then be used to produce the output when the desired output is unknown.

Since weather datasets are complex and non-linear, we have investigated ANNs including their network architectures and variations of associated learning rules to select the parameters that provide the appropriate prediction method for the given location weather data.

We have used NCEP satellite station data for this study because most stations in NMA are recently installed, and have more missing values and incomplete records.

Before applying neural network model to process the weather data, we have transformed the original data into appropriate form that helps to improve the speed, accuracy, efficiency, and performance of the network training step. So the data were scaled into a range between 0 and 1 before the training stage.

We have used and trained the models MLP, GRNN, RBF and Elman recurrent network to predict the future values of the three weather variables such as temperature, rainfall, and humidity.

The experiments were conducted using maximum and minimum temperature, rainfall, wind speed, relative humidity, evapotranspiration, elevation and coordinate variables to predict the future value of temperature, rainfall and relative humidity in the selected location weather stations.

From 34-years weather datasets, we have used the 25 years data to predict the next one year data for those three variables. For instance, to predict the next year January 15th temperature value, we have used the previous 25 years January 15th known values of temperature, rainfall, humidity, evapotranspiration, elevation and coordinates of the given weather station.

To evaluate the performance of models, appropriate performance metrics such as mean square error, mean absolute error (MAE), root mean square error (RMSE) and the correlation coefficient (R) were used.

For all the three weather variables prediction, multilayer feed-forward neural network with back propagation learning algorithm is established to be more suitable as compared to other models. The prediction capability of this model is also measured with confusion matrix. With this performance metrics, overall 76.7%, 63.2% and 78.9% of the predictions are correct and 23.3%, 36.8.7%, and 21.1% are wrong predictions for maximum temperature, rainfall and relative humidity prediction respectively.

The multilayer feed-forward back propagation model performance is better for temperature and humidity variables than rainfall variables in terms of the selected performance criteria. The possible reason for this is high variation of rainfall data between consecutive values as compared to temperature and humidity datasets so that the model generates higher error for rainfall data than temperature and humidity data. However, the MLP Network is sufficient to apply for selected weather variables prediction.

Generally, we understood that the study approach and result can contribute towards the weather prediction tasks for early warning service to prevent hazards and for further study.

5.2 Recommendations

This study has provided an additional potential applicability of machine learning techniques to establish an approach that is used to predict the future weather conditions in the northern part of Ethiopia where drought occurs most frequently. With this, the main objective of this study was achieved because the network results have strong correlation with desired outputs for the selected weather variables.

Based on the findings of this study, it is recommended that the following issues need to be addressed in future:

1. We have investigated a model that can predict values of future weather variables. But the local weather variables can be influenced by a number of factors such as amount of sunlight striking an area, the geographic aspects and slopes, the air pressure surrounding an area. In the future, the effect of all these parameters would be included in the prediction.
2. The prediction that we have made for the three weather elements would be extended to other weather factors like wind speed, sunshine, etc. to evaluate the model accuracy.
3. Investigation of the appropriate missing handling methods to fill missed value in NMA data and apply the model with this data is, of course, recommended.
4. To improve the prediction, testing other machine learning techniques such as support vector machine, Hidden Markov Model and unsupervised learning could be considered and comparing the results with the models of this study is required.
5. Since weather variables are very complex, it needs further investigation that requires expertise domain knowledge in machine learning techniques as predictive methods to study weather variations and complexities across regions using this study is needed as the initial work.

6. REFERENCES

1. K. E. Trenberth (200), “EFFECTS OF CHANGING CLIMATE ON WEATHER AND HUMAN ACTIVITIES”, National Center for Atmospheric Research Boulder, Colorado.
2. Mintewab Bezabih, et.al (2014), “On the Impact of Weather Variability and Climate Change on Agriculture: Evidence from Ethiopia”, Environment for Development.
3. NOAA (2004-2005), “Climate change information resource New York Metropolitan Region”, National Oceanic and Atmospheric Administration (NOAA), National Climate Data Center, available from: http://ccir.ciesin.columbia.edu/nyc/pdf/q1a.pdf.
4. Kassa Fekadu (2015), “Ethiopian Seasonal Rainfall Variability and Prediction Using Canonical Correlation Analysis (CCA)”, Earth Sciences Vol. 4, No. 3.
5. B. Bekuretsion and T. Beshah (2014), “Prediction of Rainfall for Ethiopian Upper Blue Nile Basin Using Soft Computing Models”, HiLCoE Journal of Computer Science and Technology, Vol. 1, No. 2.
6. Dejenie Aynalem (2015), “Interpolation of existing weather stations data to predict temperature and precipitation of remote locations”, Master of Science in computer science, Bahir Dar University.
7. T. Paraskevas, et al (2010), “Use of Artificial Neural Network for Spatial Rainfall Analysis", Viewed on Jun, 2014, Available from: http://www.ias.ac.in/jess/forthcoming/JESS-D-13-00144.pdf.
8. N. Khandelwal and R. DAVEY (2012), “Climatic Assessment of Rajasthan’s Region for Drought with Concern of Data Mining Techniques”, International Journal of Engineering Research and Applications, Vol. 2, Issue 5.
9. R. Nayak (2015), “Artificial Neural Network Model for Weather Prediction”, Foundation of Computer Science FCS, New York, USA.
10. L. Al-Matarneh, et.al (2014), “Development of Temperature-based Weather Forecasting Models Using Neural Networks and Fuzzy Logic”, International Journal of Multimedia and Ubiquitous Engineering, Vol. 9, No. 12.
11. Z. Gebreegziabher, et.al (2011), “Climate Change and the Ethiopian Economy: A Computable General Equilibrium Analysis”, Environment for Development.

12. H. Tyagi (2016), “Weather - Temperature Pattern Prediction and Anomaly Identification using Artificial Neural Network”, International Journal of Computer Applications, Volume 140, No.3.
13. World of Earth Science – 2003, “Weather Forecasting Copyright”, the Gale Group, Inc.
14. D. C. Corrales, et.al (2015), “Towards detecting crop diseases and pest by supervised learning”, Ing. Univ., vol. 19, no. 1, pp. 207-228.
15. X. Ziniu, et.al (2012), “Progress in Climate Prediction and Weather Forecast Operations in China”, ADVANCES IN ATMOSPHERIC SCIENCES, VOL. 29, NO. 5.
16. C. Monteleoni, et.al (2013), “Climate Informatics”, chapter 4; Available from: http://www.ldeo.columbia.edu/~jsmerdon/papers/2013_cidasd_monteleonietal.pdf.
17. NMA, National meteorological agency of Ethiopia; Available from: http://www.ethiomet.gov.et.
18. Temesgen Enku and Assefa Melesse (2014), “A simple temperature method for the estimation of evapotranspiration”, Hydrological Processes. 28, 2945–2960.
19. P. Camberlin (1997), “Rainfall anomalies in the source region of the Nile and their connection with the Indian summer monsoon”, Journal of Climate, vol. 10.
20. B.S. Majani (2007), “Analysis of External Drift Kriging Algorithm with application to precipitation estimation in complex orography”, Master of Science in Geo-information Science and Earth Observation, Netherlands.
21. Y. Radhika and M. Shashi (2009), “Atmospheric Temperature Prediction using Support Vector Machines”, International Journal of Computer Theory and Engineering, Vol. 1, No. 1, 1793-8201.
22. S. Singh and J. Gill (2014), “Temporal Weather Prediction using Back Propagation based Genetic Algorithm Technique”, Intelligent Systems and Applications, 12, 55-61.
23. Chapman and C Hall/CRC (2000): “TIME-SERIES FORECASTING”, Department of Mathematical Sciences, University of Bath, UK.
24. S. S. Baboo and I.K Shereef (2010), “An Efficient Weather Forecasting System using Artificial Neural Network”, International Journal of Environmental Science and Development, Vol. 1, No. 4.

25. T. Cheng and J. Wang (2008), “Integrated Spatio-temporal Data Mining for Forest Fire Prediction”, Transactions in GIS, vol.12, No. 5, 591–611.
26. S. BuHamra et.al (2003), “The Box–Jenkins analysis and neural networks: prediction and time series modeling”, Applied Mathematical Modeling, 27, 805–815.
27. P.J. diggle, J. A. Tawn (1998), “Model- Based Geostatistics”, Appl.Statist , 47, No.3
28. O. Babak and C.V. Deutsch (2009), “Statistical Approach to Inverse Distance Interpolation”, Stoch Environ Res Risk Assess, DOI 10.1007/s00477-008-0226-6.
29. T. O. Ayodele (2010), “Machine Learning Overview, New Advances in Machine Learning”, Yagang Zhang (Ed.), Available from: http://www.intechopen.com/books/newadvances-in-machine-learning/machine-learning-overview.
30. Taye Tolu (2011), “Intelligent System Based Automatic Prediction of Drought Using Satellite Images”, Master of Science in Computer Engineering, Addis Ababa University.
31. L Fausett (1994), “Fundamentals of Neural Networks Architectures, Algorithms, and Applications”.
32. R. Nkoana (2011), “Artificial neural network modeling of flood prediction and early warning”, Master dissertation, university of the Free State, Bloemfontein, South Africa.
33. Tariku Debela (2013), “Developing A Predictive Model For Fertility Preference Of Women Of Reproductive Age Using Data Mining Techniques”, Master Of Science In Health Informatics, Addis Ababa University.
34. A. Kaur, et.al (2011), “Artificial neural networks in forecasting maximum and minimum relative humidity”, IJCSNS International Journal of Computer Science and Network Security, VOL.11, No.5.
35. Donald. F. Specht (1991), “A General Regression Neural Network”, IEEE Transactions on Neural Networks, vol.2, No.6.
36. B. Kim, et.al (2004), “Prediction of plasma etching using a randomized generalized regression neural network”, Vacuum 76, 37–43.
37. H.K. Cigizoglu and M. Alp (2006), “Generalized regression neural network in modelling river sediment yield”, Advances in Engineering Software 37, 63–68.

38. Napagoda N.A.D.N (2013), “Modeling of Soil Temperature by Generalized Regression Neural Network in Bathalagoda Area of Sri Lanka”, vol. 2, No.1, 07- 12.
39. S. A. Hannan et.al (2010), “Generalized Regression Neural Network and Radial Basis Function for Heart Disease Diagnosis”, International Journal of Computer Applications, Vol.7, No.13, 0975 – 8887.
40. M. D. Buhmann (1963), “Radial Basis functions: Theory and Implementations”, Cambridge University Press.
41. Ch. J. Devi, et.al (2012), “ANN Approach for Weather Prediction using Back Propagation”, International Journal of Engineering Trends and Technology-Volume3, Issue 1.
42. S. S. De (2009), “Artificial Neural Network Based Prediction of Maximum and Minimum Temperature in the Summer Monsoon Months over India”, Applied Physics Research, vol. 1, No.2.
43. M. Hayati and Z. Mohebi (2007), “Application of Artificial Neural Networks for Temperature Forecasting”, International Journal of Electrical, Computer, Energetic, Electronic and Communication Engineering Vol.1, No.4.
44. B. A. Smith, et.al (2007), “Improving Air Temperature Prediction with Artificial Neural Networks”, International Journal of Computer, Electrical, Automation, Control and Information Engineering Vol.1, No.10.
45. Yihun Taddele and R. Srinivasan (2014), “Evaluation of Cfsr Climate Data for Hydrologic Prediction in Data scarce Watersheds: An Application In The Blue Nile River Basin”, Journal of the American Water Resources Association, No. JAWRA-13-0074-P.
46. S. Zhang, et al (2003), “Data preparation for data mining”, Applied Artificial Intelligence, 17:375–381, DOI: 10.1080/08839510390219264.
47. Gustavo E. A. P. A. Batista and M. C. Monard (2014), “An Analysis of Four Missing Data Treatment Methods for Supervised Learning”, University of S˜ao Paulo – Institute of Mathematics and Computer Science.
48. J. Han, et.al (2012), “Data mining concepts and techniques”, 3rd ed.p. cm.
49. J. P. Rigol, et.al (2001), “Artificial neural networks as a tool for spatial interpolation”, International Journal of Geographical Information Science, vol.15, No.4.

50. S. Singh, et.al (2011), “Time Series based Temperature Prediction using Back Propagation with Genetic Algorithm Technique”, International Journal of Computer Science Issues, Vol. 8, Issue 5, No 3.
51. G. Dudek (2011),”Generalized Regression Neural Network for Forecasting Time Series with Multiple Seasonal Cycles”, available from: http://www.gdudek.el.pcz.pl/files/GRNN_14.pdf.
52. T. R. Nkuna ,and J. O. Odiyo (2016),”The relationship between temperature and rainfall variability in the Levubu sub-catchmentb, South Africa”, International Journal of Environmental Science, vol. 1, ISSN: 2367-8941,.
53. R. Beaumont (2012), Personal communication with Robin Beaumont, “An Introduction to statistics Correlation”, Available from: http://www.floppybunny.org/robin/web/virtualclassroom/stats/basics/part9.pdf.

7. APPENDICES

Description of used meteorological variables and their units

Table 7.1 Meteorological Variables

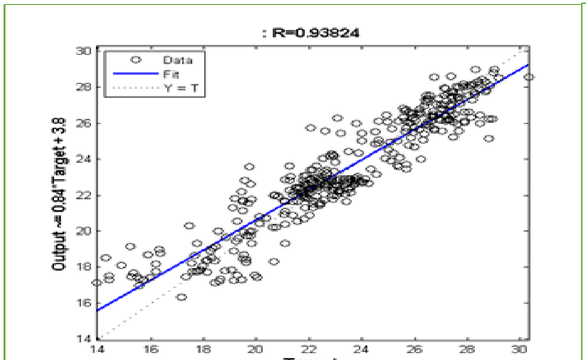
No	Variables	Unit
1	Temperature	Celsius (^{0}C)
2	Rainfall	Millimetres
3	Relative humidity	Percent (%)
4	Wind speed	Meter per second
5	Elevation	Meter
6	Latitude	degrees (°)
	Longitude	degrees (°)

The selected ANN model architecture and function description.

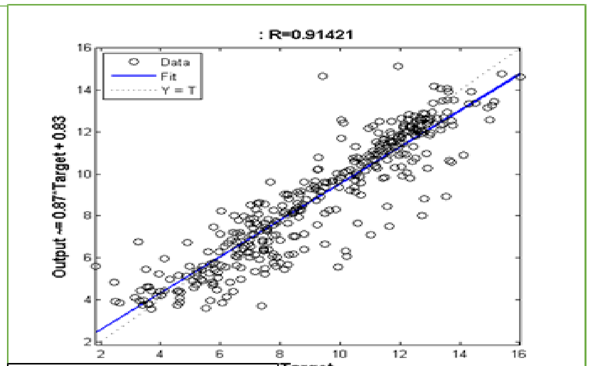
Table 7.2 MLP structure

Number of hidden layers	2
Number of hidden neuron	[6,3]
Number of epochs	1000
Activation function used in hidden layer	tansig
Activation function used in output layer	purelin
Training function	trainlm

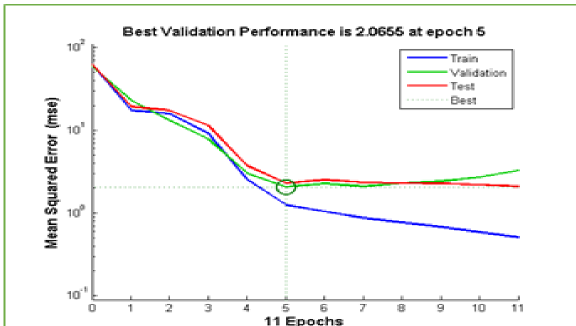
Sample output figure generated from Elman recurrent network with Ebinat weather station temperature, rainfall and relative humidity data.



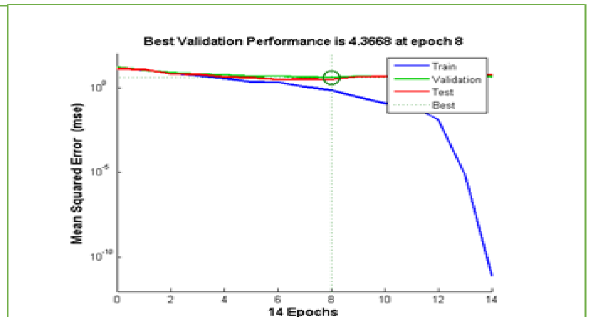
Maximum Temperature



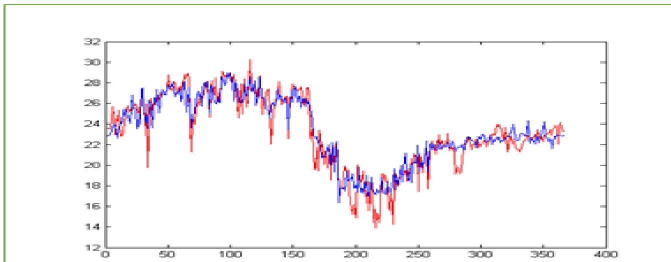
Minimum Temperature

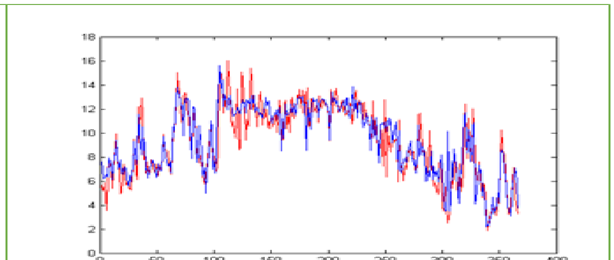


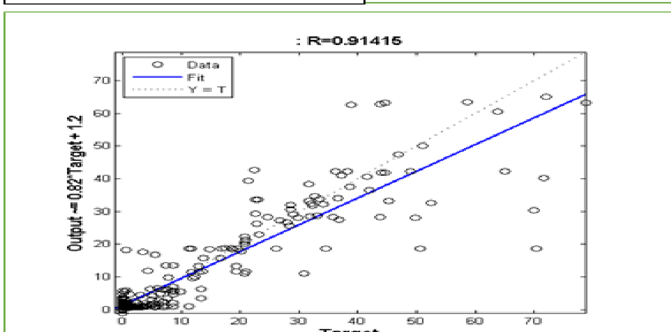
MaxTemp



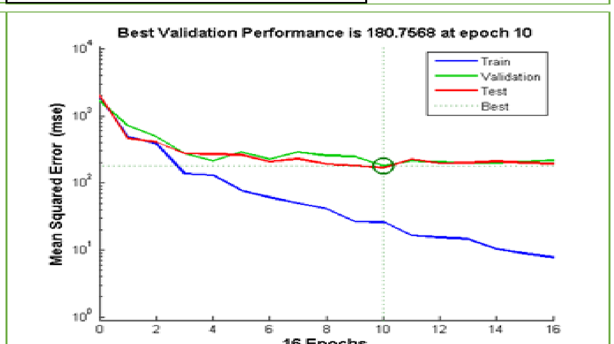
MinTemp performance


Maximum stochastic plot

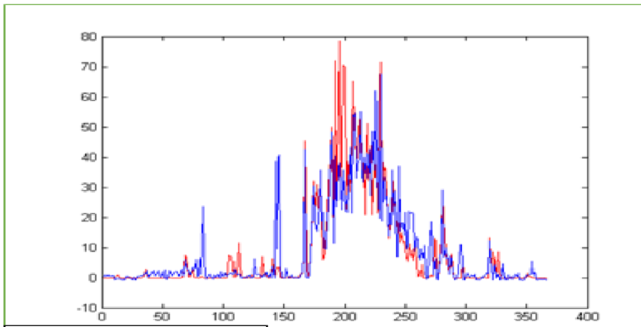

Minimum stochastic plot



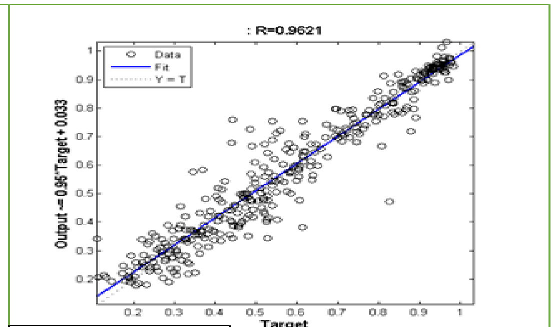
Rainfall



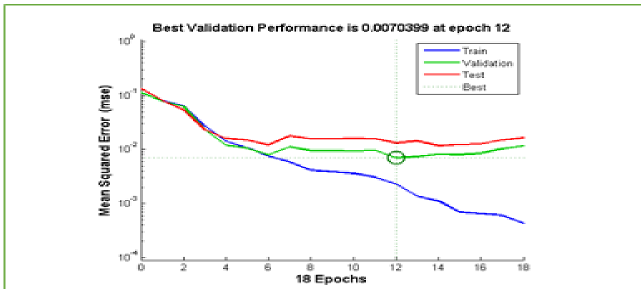
Rainfall performance



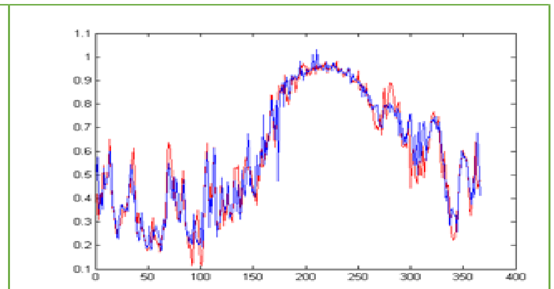
Rainfall stochastic plot



Relative Humidity



Humidity performance



Humidity stochastic plot

% Multilayer implementation in matlab

```
close all;
clc
clear all
%% create variable
NNDATA = [];

% Lists files in the current folder and  select the name of a file
filename = uigetfile('.txt');
delimiter = ' ';

%% Format string for each line of text:
% 10 columns: double (%f%f%f%f%f%f%f%f%f%f)
% For more information, see the TEXTSCAN documentation.
formatSpec = '%f%f%f%f%f%f%f%f%f%f%[^\n\r]';

%% Open the text file.
fileID = fopen(filename,'r');

%% Read columns of data according to format string.
% This call is based on the structure of the file used to generate this
% code. If an error occurs for a different file, try regenerating the code
% from the Import Tool.
dataArray = textscan(fileID, formatSpec, 'Delimiter', delimiter,
'MultipleDelimsAsOne', true, 'EmptyValue' ,NaN, 'ReturnOnError', false);

%% Close the text file.
fclose(fileID);

%% Create output variable
```

```
TEMPDATA = [dataArray{1:end-1}];

%% Display TEMPDATA values in Table
VARTEMP(9516, 10) = 0;
for i = 1: 9516
    VARTEMP(i, :) = TEMPDATA(i, :);
end

%% creating new column.
VTEMP = (VARTEMP(:, 4:10));

%% scaling the datasets
data=[VTEMP(1:9150,4);VTEMP(1:9150,6);VTEMP(:,7)];
scaledData = (data-min(data(:))) ./ (max(data(:)-min(data(:))));

%% Change number of columns and rows
newTemp=[VTEMP(1:366,1);VTEMP(1:366,2);VTEMP(1:366,3);VTEMP(1:9150,5);scaledD
ata];
day=[VARTEMP(1:366,1);VARTEMP(1:366,1);VARTEMP(1:366,1);VARTEMP(1:9150,1);VAR
TEMP(1:9150,1);VARTEMP(1:9150,1);VARTEMP(:,1)];
month=[VARTEMP(1:366,2);VARTEMP(1:366,2);VARTEMP(1:366,2);VARTEMP(1:9150,2);V
ARTEMP(1:9150,2);VARTEMP(1:9150,2);VARTEMP(:,2)];
year=[VARTEMP(1:366,3);VARTEMP(1:366,3);VARTEMP(1:366,3);VARTEMP(1:9150,3);VA
RTEMP(1:9150,3);VARTEMP(1:9150,3);VARTEMP(:,3)];

%% Merge columnes
TempEdit=[day,month,year,newTemp];
VARTEMP=TempEdit;
TIME = 1: 38064;
Xi = TIME;
Yi = interp1(TIME, newTemp, Xi, 'spline');
VARTEMP(:, 4) = Yi;

%% Display TEMPDATA values in Table
odatasize = length(NNDATA);
ndatasize = odatasize + 366;
NNDATA = resizem (NNDATA, [104 ndatasize]);
k = 1;
for i = 1: 104
    for j = odatasize+1 : ndatasize
        NNDATA (i , j) = VARTEMP (k, 4);
        k = k + 1;
    end
end

nninput = NNDATA(1:103, :);
nnoutput = NNDATA(104, :);

%% Display NNDATA values in Table
inputs = nninput;
targets = nnoutput;

% Create a Network architecture
```

```
 net =
newff(inputs,targets,[6,3],{'logsig','logsig','purelin'},'trainlm','learngdm'
);

% net.inputs{1}.processFcns = {'removeconstantrows','mapminmax'};
% net.outputs{2}.processFcns = {'removeconstantrows','mapminmax'};

% Setup Division of Data for Training, Validation, Testing
% data division functions
net.divideFcn = 'dividerand';  % Divide data randomly
net.divideMode = 'sample';  % Divide up every sample
net.divideParam.trainRatio = 70/100;
net.divideParam.valRatio = 15/100;
net.divideParam.testRatio = 15/100;

%  training functions:
% net.trainFcn = 'trainlm';  % Levenberg-Marquardt
net.trainParam.showWindow = true; % Show training GUI
net.trainParam.showCommandLine=false; % Generate command-line output
net.trainParam.epochs=1000; % Maximum number of epochs to train
net.trainParam.goal=0; % Performance goal.
net.trainParam.min_grad = 1e-7; % Minimum performance gradient
net.trainParam.max_fail = 6; % Maximum validation failures
net.trainParam.mu = 0.001; % Initial mu
net.trainParam.mu_dec = 0.1; % mu decrease factor
net.trainParam.mu_inc = 10; % mu increase factor
net.trainParam.mu_max =10000000000; % 1e10 Maximum mu

% Choose a Performance Function
net.performFcn = 'mse';  % Mean squared error

% Choose Plot Functions
% For a list of all plot functions type: help nnplot
net.plotFcns = {'plotperform','plottrainstate','ploterrhist', ...
  'plotregression', 'plotfit'};

% Train the Network
[net,tr] = train(net,inputs,targets);

% Test the Network
outputs = net(inputs);
errors = gsubtract(targets,outputs);
performance = perform(net,targets,outputs)

% Recalculate Training, Validation and Test Performance
trainTargets = targets .* tr.trainMask{1};
valTargets = targets  .* tr.valMask{1};
testTargets = targets  .* tr.testMask{1};
trainPerformance = perform(net,trainTargets,outputs)
valPerformance = perform(net,valTargets,outputs)
testPerformance = perform(net,testTargets,outputs)

% View the Network
view(net)
```

```
% plot
t=1:1:366;
figure,plot(t,targets,'r',t,outputs,'b');
 % predictedValue=outputs'
err=targets-outputs;
Mse=mse(err)
Mae=mae(err)
Rmse=sqrt(Mse)
% Uncomment these lines to enable various plots.
figure, plotperform(tr)
% figure, plottrainstate(tr)
% figure, plotfit(net,inputs,targets)
figure, plotregression(targets,outputs)
% figure, ploterrhist(errors)

% msgbox('Neural Network Training Complete !!!)
```