A

Major Project

On

RAINFALL PREDICTION USING MACHINE LEARNING ALGORITHMS

(Submitted in partial fulfilment of the requirements for the award of Degree)

BACHELOR OF TECHNOLOGY

In

COMPUTER SCIENCE AND ENGINEERING

By

PATHARLA RAJESH (187R1A0533)

Under the Guidance of

Dr. K. Srujan Raju

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CMR TECHNICAL CAMPUS

UGC AUTONOMOUS

(Accredited by NAAC, NBA, Permanently Affiliated to JNTUH, Approved by AICTE, New Delhi)

Recognised Under Section 2(f) & 12(B) of the UGCAct.1956, Kandlakoya (V),

Medchal Road, Hyderabad-501401.

2019-2023

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that the project entitled **"RAINFALL PREDICTION USING MACHINE LEARNING ALGORITHM"** being submitted by **PATHARLA RAJESH (187R1A0533)**in partial fulfilment of the requirements for the award of the degree of B.Tech in Computer Science and Engineering to the CMR Technical Campus Hyderabad, is a record of bonafide work carried out by them under our guidance and supervision during the year 2022-23. The results embodied in this thesis have not been submitted to any other University or Institute for the award of any degree or diploma.

Dr. K. Srujan Raju (HOD) **Dr. A. Raji Reddy** (DIRECTOR)

INTERNAL GUIDE

Dr. K. Srujan Raju (HOD) **EXTERNAL EXAMINER**

Submitted for viva voice Examination held on__________________________

ACKNOWLEDGEMENT

Apart from my efforts , the success of any project depends largely on the encouragement and guidelines of many others. We take this opportunity to express our gratitude to the people who have been instrumental in the successful completion of this project.

I took this opportunity to express my profound gratitude and deep regard to my guide **Dr. K. Srujan Raju**, Head of the Department for his exemplary guidance, monitoring and constant encouragement throughout the project work. The blessing, help and guidance given by him shall carry us a long way in the journey of life on which we are about to embark.

I also take this opportunity to express a deep sense of gratitude to the Project Review Committee (PRC) **Dr.Punyaban Patel, Ms.Shilpa, M.Subha Mastan Rao & J. Narasimharao** for their cordial support, valuable information and guidance, which helped us in completing this task through various stages.

I am also thankful to **Dr. K. Srujan Raju**, Head of the Department of Computer Science and Engineering for providing encouragement and support for completing this project successfully.

I am also thankful to **Dr. A. Raji Reddy**, Director for being cooperative throughout the course of this project. I also express our sincere gratitude for **Sri. Ch. Gopal Redd**y, Chairman for providing excellent infrastructure and a nice atmosphere throughout the course of this project.

The guidance and support received from all the members of **CMR Technical Campus** who contributed to the completion of the project. I'm grateful for their constant support and help.

Finally, I would like to take this opportunity to thank our family for their constant encouragement, without which this assignment would not be completed. I sincerely acknowledge and thank all those who gave support directly and indirectly in the completion of this project.

PATHARLA RAJESH (187R1A0533)

ABSTRACT

Rainfall is always a major issue across the world as it affects all the major factor on which the human being is depended. In current, Unpredictable and accurate rainfall prediction is a challenging task. We apply rainfall data of India to different machine learning algorithms and compare the accuracy of classifiers such as SVM, Navie Bayes, Logistic Regression, Random Forest and Multilayer Perceptron (MLP). Our motive is to get the optimised result and a better rainfall prediction. This studies are collected, trained and tested to achieve the sustainable results through ANFIS(Adaptive Neural fuzzy interference system) and ANN(Artificial neural network) models. The monthly rainfall predictions obtained after training and testing are then compared with actual data to ensure the accuracy of the model. The results of this study outline that the model is successful in predicting the monthly rainfall data with the particular parameters. The training and testing of data through Neural fuzzy model helped in not only minimising the errors and also maximising the reliability and durability of the predicted data. The results of the study highlight that the ANFIS model is most suitable among the artificial networks for the rainfall prediction. The outcome data with ANFIS system presented maximum accuracy with minimum error through the comparison between the actual data and predicted outcome data.

LIST OF FIGURES/TABLES

LIST OF SCREENSHOTS

TABLE OF CONTENTS

INTRODUCTION

INTRODUCTION

1.1 PROJECT SCOPE

Data analysis is done to achieve certainty of future result to be close so that prediction is valid and correctly interpreted. This certainty can be gained only after raw data is verified and checked for abnormality thus ensuring that the data was gathered without any errors. It also helps in finding the data which contains irrelevant features for prediction model. Data pre-processing is a data mining technique that converts raw and inconsistent data into useful understandable format for the model. Raw data is inconsistent and incomplete and contains missing features along with many errors. As per data exploration and analysis we have learned that raw data for our model contains many null values which must be replaced with their mean value.

1.2 PROJECT PURPOSE

The focus is to design a model and analyse the performance of various Machine Learning algorithms and predict the most accurate algorithm for the predicting of rainfall. This research was done using techniques of Logistic Regression, Random Forest, Decision Tree, K-Nearest Neighbour on the dataset. For the experimental purpose we have given the actual real time values of maximum and minimum temperature, relative humidity, wind speed etc. Dataset was segregated into training and testing data and after those models were trained and the accuracy score was noted and analysed before final prediction.

1.3 PROJECT FEATURES

Rainfall projection is utmost necessary all over world and it plays a key role in human life. It's cumbersome responsibility of meteorological department to analyse the frequency of rainfall with precariousness. It is difficult to forecast the rainfall precisely with varying atmospheric condition. It is conjectured to predict the rainfall for both summer and rainy seasons. This is the primary reason because of this there is necessity to analyse about the algorithms adaptable for rainfall prediction. One of such skilled and effective technologies is Machine Learning, “Machine

Learning is a way of manipulating and extraction of implicit, previously unknown and known and potential useful information about data". Machine Learning is colossal and deep field and its scope and implementation is increasing day by day.

SYSTEM ANALYSIS

SYSTEMANALYSIS

SYSTEMANALYSIS

Machine learning covers various classifiers of Supervised, Unsupervised and Ensemble Learning which are used to predict and find the accuracy of the given dataset. We can use that knowledge in our project of Rainfall Prediction System as it will help a lot of people. Various Machine Learning algorithms such as Logistic Regression, Decision Tree, K-Nearest Neighbour , Random Forest are compared to find the most accurate model. Here the rainfall dataset from the UCI repository is used. In this research a discussion and comparison of the existing classification techniques is made.

2.1 PROBLEM DEFINITION

The data sample is limited to monthly statistics only and does not provide the daily output predictions so the climatic change and the global warming effect may impact the accuracy of the expected output . The locations for the data processing used in this study are geographically different and distanced that could also impact the correlation efficient that will measure the performance of the ANFIS and ANN in this research. The system discussed in this particular study will operate with Matlab software (R2017).

2.2 EXISTING SYSTEM

The accurate and precise rainfall prediction is still lacking which could assist in diverse fields like agriculture, water reservation and flood prediction. The issue is to formulate the calculations for the rainfall prediction that would be based on the previous findings and similarities and will give the output predictions that are reliable and appropriate. The imprecise and inaccurate predictions are not only the waste of time but also the loss of resources and lead to inefficient management of crisis like poor agriculture, poor water reserves and poor management of floods. Therefore, the need is not to formulate only the rainfall predicting system but also a system that is more accurate and precise as compared to the existing rainfall predictors.

2.2.1 DISADVANTAGES OF EXISTING SYSTEM

Unpredictability of changes in ocean currents

Heavy rainfall can lead to numerous hazards

Forecasts are Never Completely Accurate

2.3 PROPOSED SYSTEM

Based on the LSTM and Stacked-LSTM Networks were adapted for the task of forecasting hourly rainfall using time-series data.A model based on Bidirectional-LSTM Networks was proposed for the task of forecasting rainfall on an hourly basis using time-series comparison of the performance of models based on LSTM-Networks, Stacked-LSTM Networks, Bidirectional-LSTM Networks, XGBoost, and the resulting model from AutoML was performed in the task of forecasting the amount of rainfall. First, research works that use Feed-forward Back-propagation Neural Networks and LSTM-Networks architectures to build rainfall prediction models are introduced.

2.3.1 ADVANTAGES OF THE PROPOSED SYSTEM

Water resources can be managed efficiently by using rainfall prediction system.

Regions can be evacuated if flood are expected.It helps in taking appropriate measures to efficiently

manage water resources, crop productivity and no wastage of any resources.

2.4 FEASIBILITY STUDY

The orographic rain is the form of rainfall that is formed by the moist air which usually can be observed above the mountains. The moist air above the mountains is evaporated or lifted upward direction. When the moist air is lifted and rises to a certain level it cools down; the orographic clouds are formed and then condenses and forms the precipitation. The orographic rainfall has tiny water drops that are condensed. These small water drops from clouds and then these small clouds come together to form bigger clouds. These clouds also turn into snow over some period of time.

2.4.1 TECHNICAL FEASIBILITY

The rainfall is a natural phenomenon that is measured in mm. The measuring instrument is 203mm in diameter. This is a funnel that gathers the rainfall into a cylinder and has the capacity of measuring 0 to 25mm of rainfall in 2004. There are two techniques for measuring rainfall (Ordinary Rain gauge) (Self- Recording Rain gauge).

2.4.2 BEHAVIOURAL FEASIBILITY

The cyclonic or the frontal rainfall is the last and third type of the rainfall. The cyclonic by name represents the tempesting and occurs when the air masses with distinct characteristics collide with one another. The collision of light air that is warm and the cold air that is heavy occurs; the cold air encourages the warm air because it is lighter to rise. The rising air cools down by forming the water vapours. The condensation process initiates and forms the clouds. The ordinary rain gauge measurement is a less effective and less accurate technique of measuring the rainfall. It has been observed that the ordinary gauge is the non-automatic observation and uses a glass to measure the rain at regular intervals. It has a shell, a storage bottle with a storage vessel and a glass for measuring the rain. It is not effective for the heavier and substantial rainfall like the cyclonic/frontal rainfall.

2.5 HARDWARE& SOFTWARE REQUIREMENTS

2.5.1 HARDWAREREQUIREMENTS:

System : iOs/ Windows

Hard Disk : 40Gb

Ram : 4Gb

2.5.2 SOFTWARE REQUIREMENTS:

Operating System : iOs/WindowsXp

Coding Language : Python

**Front-end : **Pycharm /, Jupyter

ARCHITECTURE

ARCHITECTURE

3.1 PROJECT ARCHITECTURE

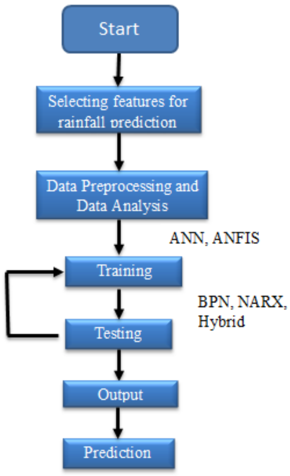


Figure 3.1: Project Architecture for Rainfall prediction using Machine learning

3.2 DESCRIPTION

This study is focusing on the back propagation to obtain accuracy for training, testing and validation of input and output data.

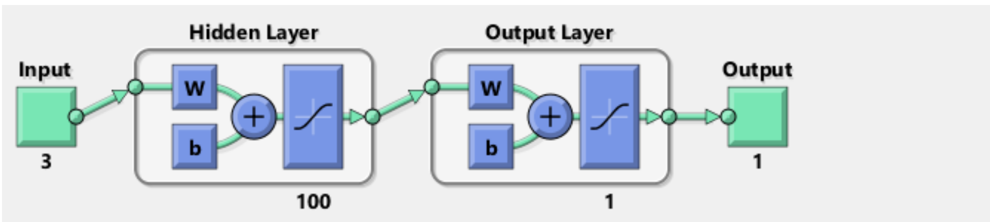


Figure 3.2: Back propagation and accuracy for Rainfall prediction

3.3 USE CASE DIAGRAM

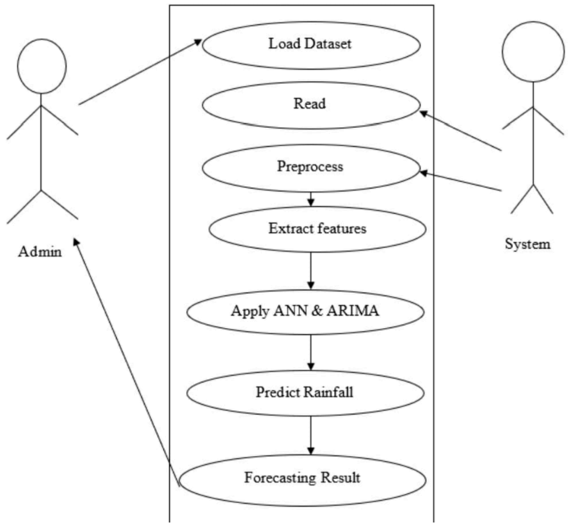


Figure 3.3: Use case diagram for Rainfall prediction using machine learning

3.4 CLASS DIAGRAM

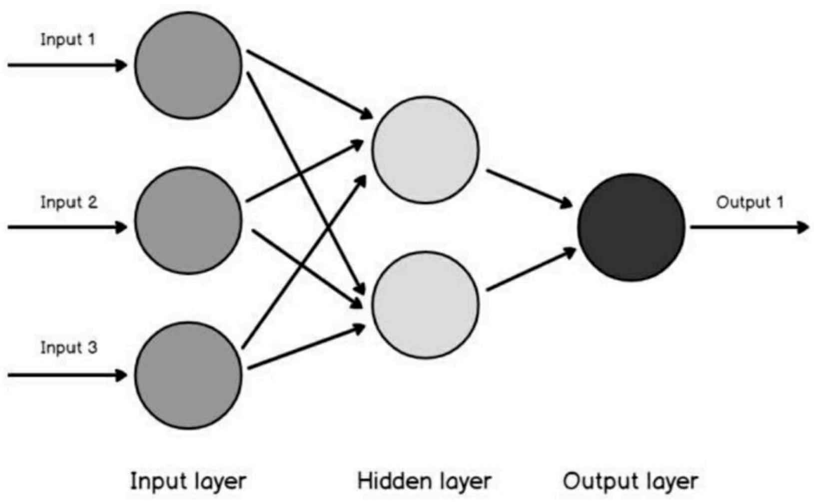


Figure 3.4: Class diagram for Rainfall Prediction

3.5 SEQUENCE DIAGRAM

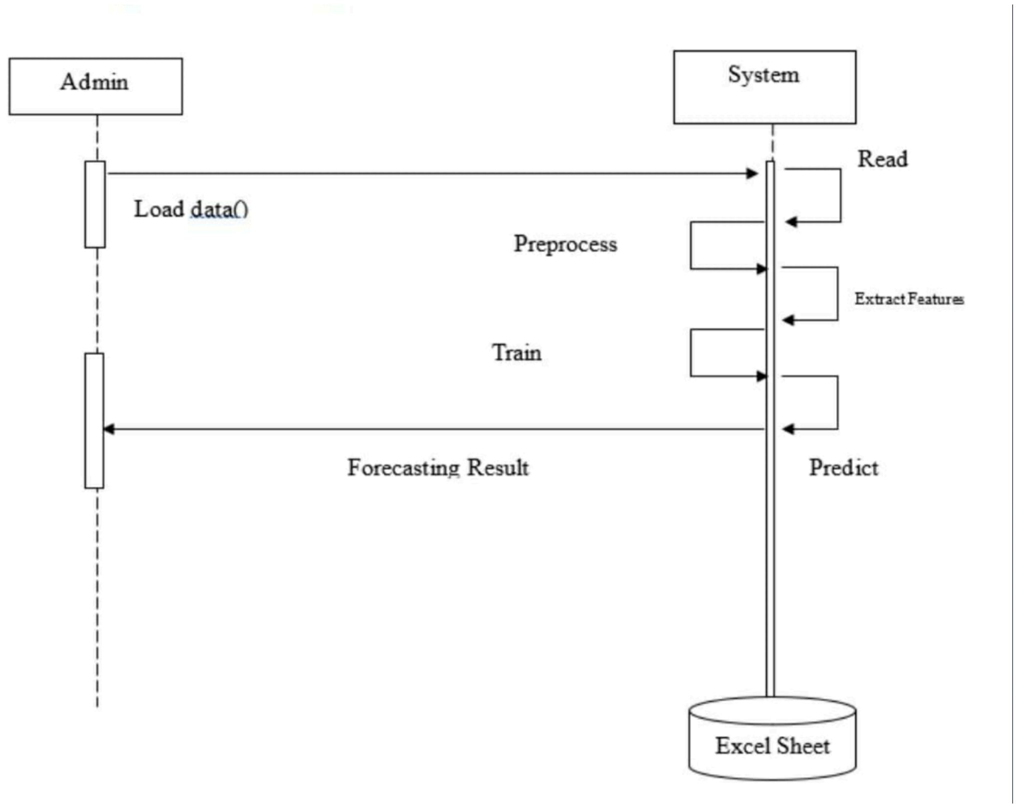


Figure 3.5: Sequence diagram for Rainfall prediction

3.6 ACTIVITY DIAGRAM

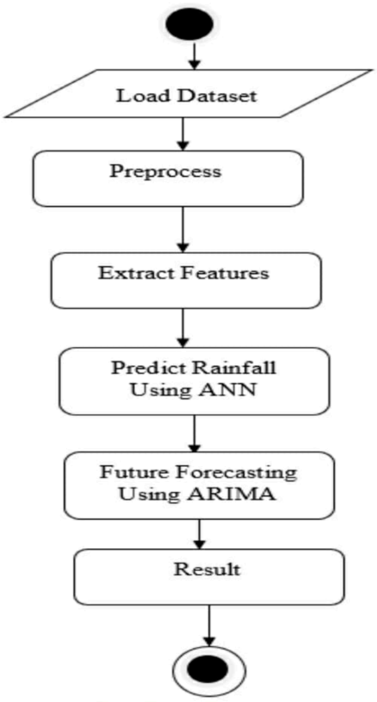


Figure 3.6: Activity Diagram for Rainfall prediction using machine learning

IMPLEMENTATION

IMPLEMENTAION

Execution code:

Try DataSpell — a dedicated IDE for data science,
with full support for local and remote notebooks

Try Datalore — an online environment
for Jupyter notebooks in the browser
Also read more about JetBrains Data Solutions on our website

Rainfall Prediction

```
import pandas as pd   #importing Libraries
import numpy as np
import matplotlib.pyplot as plt


ds=pd.read_csv("fall.csv")   #Reading Dataset


ds.head()     #Displaying first 5 records



ds=ds.drop(['month','day'], axis=1)   #Droping Unnecessary columns


ds.head()    #Displaying first 5 records



ds.isnull().sum()      #Checking for any missing values
year            0
tempavg         0
DPavg           0
humidity avg    0
SLPavg          0
visibilityavg   0
windavg         0
Rainfall        0
dtype: int64


x=ds.iloc[:,:7].values     #Assigning variables to the dependent and independent attiributes
y=ds.iloc[:,7].values


x
```

```
array([[2011,   21,   13, ..., 1009,    3,    5],
       [2011,   21,   18, ..., 1009,    3,    5],
       [2011,   22,   18, ..., 1010,    3,   10],
       ...,
       [2017,   23,   16, ..., 1014,    4,    5],
       [2017,   23,   17, ..., 1013,    4,    5],
       [2017,   24,   18, ..., 1012,    4,    2]], dtype=int64)


y
array([0., 0., 0., ..., 0., 0., 0.])


from sklearn.model_selection import train_test_split        #performing train-test split.
x_train, x_test, y_train, y_test = train_test_split(x, y, test_size = 0.20,random_state=0)


x_train
array([[2015,   30,   27, ..., 1004,    6,   14],
       [2015,   29,   26, ..., 1008,    6,   13],
       [2017,   25,   22, ..., 1012,    4,   14],
       ...,
       [2013,   28,   26, ..., 1003,    5,   10],
       [2013,   23,   22, ..., 1007,    5,   11],
       [2016,   18,   14, ..., 1017,    2,   13]], dtype=int64)


from sklearn.ensemble import RandomForestRegressor       #Model building using Random Forest
regressor = RandomForestRegressor(n_estimators=100, random_state = 0)
regressor.fit(x_train, y_train)

RandomForestRegressor(random_state=0)


ypred =regressor.predict(x_test)     #Predicting using testing data on trained data.


ypred
array([7.18700000e-01, 0.00000000e+00, 0.00000000e+00, 0.00000000e+00,
       0.00000000e+00, 0.00000000e+00, 0.00000000e+00, 0.00000000e+00,
       7.67110000e+00, 1.51170000e+00, 2.18380000e+00, 8.64940000e+00,
       2.54000000e-01, 0.00000000e+00, 6.47110000e+00, 5.10000000e-03,
       4.97400000e+00, 7.31500000e-01, 1.17830000e+00, 1.77600000e-01,
       0.00000000e+00, 4.55722000e+01, 6.93180000e+00, 1.01308000e+01,
       2.02310000e+00, 2.84500000e-01, 0.00000000e+00, 7.50000000e-03,
       1.22150000e+00, 0.00000000e+00, 6.98600000e-01, 1.26900000e-01,
       0.00000000e+00, 2.12630000e+00, 6.16950000e+00, 0.00000000e+00,
       3.81320000e+00, 0.00000000e+00, 1.51974000e+01, 0.00000000e+00,
```

4.14550000e+00, 0.00000000e+00, 5.60000000e-02, 0.00000000e+00,
0.00000000e+00, 0.00000000e+00, 1.01400000e-01, 6.14700000e-01,
9.55020000e+00, 5.05300000e-01, 2.03000000e-02, 9.43910000e+00,
6.08290000e+00, 3.09360000e+00, 1.33870000e+00, 2.94513000e+01,
0.00000000e+00, 9.95500000e-01, 6.55400000e-01, 8.57960000e+00,
3.60130000e+00, 1.42480000e+00, 2.77840000e+00, 0.00000000e+00,
1.46090000e+00, 1.60000000e-01, 0.00000000e+00, 0.00000000e+00,
2.47860000e+00, 2.23640000e+00, 3.98800000e-01, 3.45400000e-01,
2.31398000e+01, 3.27900000e+00, 2.15570000e+00, 3.19250000e+00,
0.00000000e+00, 2.37780000e+00, 3.07300000e-01, 0.00000000e+00,
1.84660000e+00, 0.00000000e+00, 0.00000000e+00, 1.53540000e+00,
4.67000000e-01, 1.87410000e+00, 2.28620000e+00, 4.37100000e-01,
1.43740000e+00, 3.68300000e-01, 0.00000000e+00, 1.31850000e+00,
7.22250000e+00, 2.35240000e+00, 1.49050500e+01, 5.89100000e-01,
0.00000000e+00, 2.51170000e+00, 2.31910000e+00, 0.00000000e+00,
5.39380000e+00, 3.38500000e+00, 1.31810000e+00, 3.16980000e+00,
3.46400000e+00, 8.36000000e-02, 0.00000000e+00, 0.00000000e+00,
2.50000000e-03, 5.05400000e-01, 1.62500000e-01, 0.00000000e+00,
1.39480000e+00, 9.85600000e-01, 1.04330000e+00, 1.67121000e+01,
9.21700000e-01, 7.64720000e+00, 0.00000000e+00, 1.49800000e-01,
3.25000000e-01, 8.78800000e-01, 6.35200000e-01, 4.36700000e-01,
0.00000000e+00, 1.43642000e+01, 0.00000000e+00, 1.64656000e+01,
6.42900000e-01, 7.87000000e-02, 1.02000000e-02, 1.19994000e+01,
1.59750000e+00, 4.24190000e+00, 1.57210000e+00, 5.80493333e+00,
1.52150000e+00, 3.49246000e+01, 0.00000000e+00, 0.00000000e+00,
3.62710000e+00, 7.64300000e-01, 2.32464000e+01, 7.38300000e+00,
0.00000000e+00, 2.18500000e+00, 3.49630000e+00, 1.31700000e-01,
8.96600000e+00, 3.19030000e+00, 0.00000000e+00, 3.84190000e+00,
1.24500000e-01, 3.34880000e+00, 1.00000000e-02, 0.00000000e+00,
7.92500000e-01, 0.00000000e+00, 3.32807500e+01, 8.63000000e-02,
0.00000000e+00, 3.16640000e+00, 0.00000000e+00, 1.45615000e+01,
0.00000000e+00, 2.35470000e+00, 1.02000000e-02, 5.71600000e-01,
6.10000000e-02, 7.62100000e-01, 3.03730000e+00, 0.00000000e+00,
1.26738000e+01, 2.03200000e-01, 1.12570000e+00, 1.56670000e+00,
4.09200000e-01, 3.26920000e+00, 5.97660000e+00, 5.43400000e-01,
6.27500000e-01, 1.22911000e+01, 0.00000000e+00, 2.13830000e+00,
6.40100000e-01, 2.46400000e-01, 4.27930000e+00, 9.71710000e+00,
1.12976000e+01, 1.01200000e+00, 0.00000000e+00, 7.28900000e-01,
1.49150000e+00, 1.63888000e+01, 1.73708000e+01, 0.00000000e+00,
0.00000000e+00, 0.00000000e+00, 9.11800000e-01, 0.00000000e+00,
0.00000000e+00, 1.07690000e+00, 0.00000000e+00, 0.00000000e+00,
0.00000000e+00, 0.00000000e+00, 3.66520000e+00, 2.87780000e+00,
0.00000000e+00, 1.05742000e+01, 4.84900000e-01, 1.64717000e+01,
6.83300000e-01, 4.04360000e+00, 9.09100000e-01, 0.00000000e+00,
1.52000000e-02, 0.00000000e+00, 0.00000000e+00, 1.71421000e+01,
6.37010000e+00, 1.50467000e+01, 4.02470000e+00, 0.00000000e+00,
1.56076000e+01, 0.00000000e+00, 8.48800000e-01, 1.16005000e+01,
3.40400000e-01, 5.66900000e-01, 6.81100000e-01, 1.00184000e+01,

0.00000000e+00, 7.44400000e-01, 1.02000000e-02, 0.00000000e+00,
3.45420000e+00, 0.00000000e+00, 9.15000000e-02, 5.48700000e-01,
2.99120000e+00, 1.41940000e+00, 0.00000000e+00, 2.28000000e-02,
2.35650000e+00, 0.00000000e+00, 2.00440000e+00, 8.43000000e-01,
2.83640000e+00, 3.48000000e-01, 0.00000000e+00, 8.63700000e-01,
0.00000000e+00, 6.12350000e+00, 1.85020000e+00, 1.29600000e-01,
2.03200000e-01, 3.70800000e-01, 0.00000000e+00, 0.00000000e+00,
1.47300000e-01, 2.03856000e+01, 1.83970000e+00, 0.00000000e+00,
3.79960000e+00, 2.12070000e+00, 4.31300000e-01, 1.27000000e-02,
0.00000000e+00, 0.00000000e+00, 1.06590000e+00, 1.98200000e-01,
2.30100000e+00, 7.60000000e-03, 1.91760000e+00, 3.43470000e+00,
1.88190000e+00, 1.53170000e+00, 1.06470000e+00, 9.85700000e-01,
3.06590000e+00, 1.69600000e-01, 0.00000000e+00, 9.22730000e+00,
2.53900000e-01, 2.06220000e+00, 8.81680000e+00, 4.41100000e+00,
2.28600000e-01, 0.00000000e+00, 2.74072000e+01, 1.26720000e+00,
6.85000000e-02, 1.06700000e-01, 0.00000000e+00, 8.84650000e+00,
2.03500000e+00, 1.90030000e+00, 1.17805000e+01, 1.21970000e+00,
2.30350000e+00, 1.43750000e+00, 4.65300000e-01, 0.00000000e+00,
0.00000000e+00, 0.00000000e+00, 6.49030000e+00, 0.00000000e+00,
4.03800000e-01, 0.00000000e+00, 0.00000000e+00, 3.30200000e-01,
2.94800000e-01, 3.28420000e+00, 2.84410000e+00, 2.90090000e+00,
2.68057100e+01, 5.07900000e-01, 6.33570000e+00, 3.71260000e+00,
3.54390000e+00, 2.72760000e+00, 3.06510000e+00, 9.92290000e+00,
3.70900000e-01, 1.04952500e+01, 0.00000000e+00, 9.52600000e-01,
0.00000000e+00, 8.07600000e-01, 3.18602500e+00, 2.19634000e+01,
0.00000000e+00, 4.11400000e-01, 5.00000000e-03, 4.97900000e-01,
2.19690000e+00, 2.42310000e+00, 1.34180000e+00, 0.00000000e+00,
2.04446000e+01, 2.50000000e-03, 8.39750000e+00, 0.00000000e+00,
0.00000000e+00, 1.04557000e+01, 1.19400000e-01, 1.17030000e+00,
1.24200000e+00, 2.28600000e-01, 2.04000000e-02, 0.00000000e+00,
2.29000000e-02, 2.01793000e+01, 0.00000000e+00, 1.52110000e+00,
0.00000000e+00, 5.31790000e+00, 0.00000000e+00, 8.84930000e+00,
0.00000000e+00, 1.27000000e-01, 1.00365000e+01, 5.59500000e+00,
0.00000000e+00, 4.78800000e-01, 1.41230000e+00, 6.42150000e+00,
0.00000000e+00, 0.00000000e+00, 0.00000000e+00, 2.14230000e+00,
0.00000000e+00, 1.08430000e+00, 9.06600000e-01, 0.00000000e+00,
4.49100000e+00, 1.44430000e+00, 1.09240000e+00, 3.06820000e+00,
1.32300000e-01, 0.00000000e+00, 9.57100000e-01, 4.45580000e+00,
1.01380000e+00, 1.00620000e+00, 2.02410000e+00, 1.29050000e+00,
7.87000000e-02, 1.92900000e-01, 0.00000000e+00, 0.00000000e+00,
2.26130000e+00, 1.39700000e-01, 1.16080000e+00, 0.00000000e+00,
2.05800000e-01, 1.09300000e-01, 1.15540000e+00, 2.25268000e+01,
5.08000000e-02, 3.61180000e+00, 0.00000000e+00, 1.23569000e+01,
4.34390000e+00, 3.21590000e+00, 2.50000000e-03, 1.08700000e+00,
1.55450000e+00, 0.00000000e+00, 1.17473000e+01, 1.88190000e+00,
3.60800000e-01, 1.93240000e+00, 1.57174000e+01, 2.31000000e-01,
1.50650000e+00, 1.17870000e+00, 7.28300000e-01, 2.46150000e+00,
1.85580000e+00, 1.83000000e-01, 0.00000000e+00, 2.46400000e-01,

```
3.71340000e+00, 0.00000000e+00, 0.00000000e+00, 1.42100000e-01,
3.76210000e+00, 0.00000000e+00, 7.38710000e+00, 0.00000000e+00,
0.00000000e+00, 7.72400000e-01, 8.80620000e+00, 0.00000000e+00,
9.53270000e+00, 7.06760000e+00, 1.47560000e+00, 1.02340000e+00,
0.00000000e+00, 1.31403000e+01, 0.00000000e+00, 4.37100000e-01,
0.00000000e+00, 2.32110000e+00, 1.87570000e+00, 2.03720000e+00,
0.00000000e+00, 0.00000000e+00, 3.68200000e-01, 1.60650000e+00,
1.78000000e-02, 0.00000000e+00, 1.82610000e+00, 6.79640000e+00,
1.32590000e+00, 2.59110000e+00, 2.18400000e-01, 1.26230000e+00,
2.04000000e-02, 3.58200000e-01, 8.43200000e-01, 1.33850000e+00,
1.87462500e+01, 0.00000000e+00, 7.72880000e+00, 7.24140000e+00,
4.02900000e-01])
```

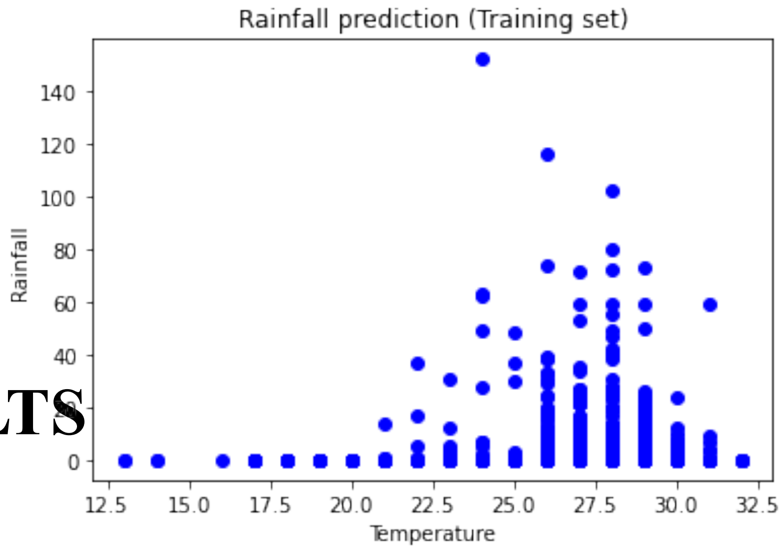
Front end Code:

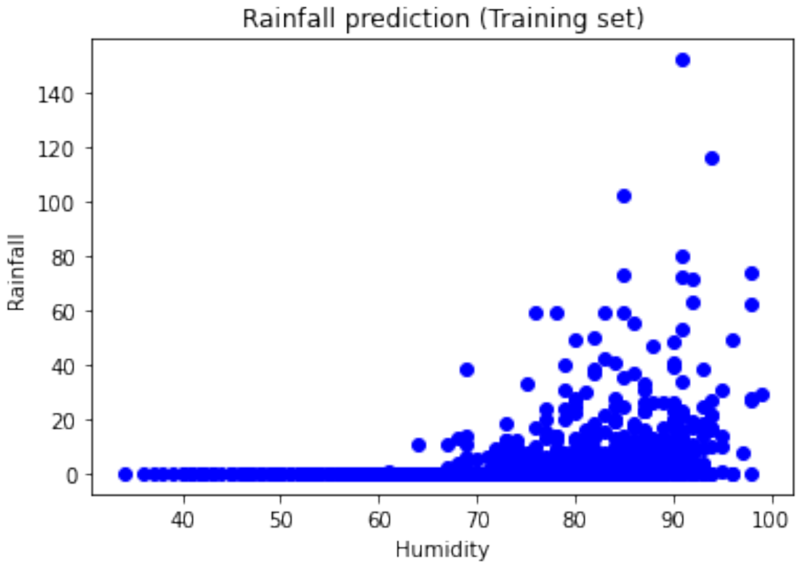
```
y_test.shape

from sklearn.metrics import r2_score #Testing accuracy
r2_score(y_test,ypred)
0.2728975259726836

x_train.shape

y_train.shape

plt.scatter(x_train[:,1],y_train,color='blue')      #Displaying relation b/w temp and rainfall
plt.title('Rainfall prediction (Training set)')
plt.xlabel('Temperature')
plt.ylabel('Rainfall')
plt.show()

plt.scatter(x_train[:,2],y_train,color='blue')      #Displaying relation b/w dp avg and rainfall
plt.title('Rainfall prediction (Training set)')
plt.xlabel('DP avg')
plt.ylabel('Rainfall')
plt.show()

plt.scatter(x_train[:,3],y_train,color='blue')      #Displaying relation b/w humidity and rainfall
plt.title('Rainfall prediction (Training set)')
plt.xlabel('Humidity')
plt.ylabel('Rainfall')
plt.show()

plt.scatter(x_train[:,4],y_train,color='blue')      #Displaying relation SLP and rainfall
plt.title('Rainfall prediction (Training set)')
plt.xlabel('SLP')
plt.ylabel('Rainfall')
plt.show()
plt.scatter(x_train[:,5],y_train,color='blue')      #Displaying relation b/w visibility and rainfall
```

```
plt.title('Rainfall prediction (Training set)')
plt.xlabel('visibility')
plt.ylabel('Rainfall')
plt.show()

plt.scatter(x_train[:,6],y_train,color='blue')     #Displaying relation b/w Wind and rainfall
plt.title('Rainfall prediction (Training set)')
plt.xlabel('Wind')
plt.ylabel('Rainfall')
plt.show()

ds.corr()     #Co-relation b/w attributes

import seaborn as sns      #importing seaborn Library


sns.heatmap(ds.corr(),annot=True)    #Displaying Co-relation b/w attributes using Heatmap
<matplotlib.axes._subplots.AxesSubplot at 0x219ec56fd30>
ypred1= regressor.predict([[2020,18,16,65,1013,6,8]])      #Predicting new data record.
ypred1     #prediction
```

Done

RESULTS


Rainfall prediction (Training set)
Rainfall
Temperature
0
20
40
60
80
100
120
140
12.5
15.0
17.5
20.0
22.5
25.0
27.5
30.0
32.5


Rainfall prediction (Training set)
Rainfall
Humidity
0
20
40
60
80
100
120
140
40
50
60
70
80
90
100

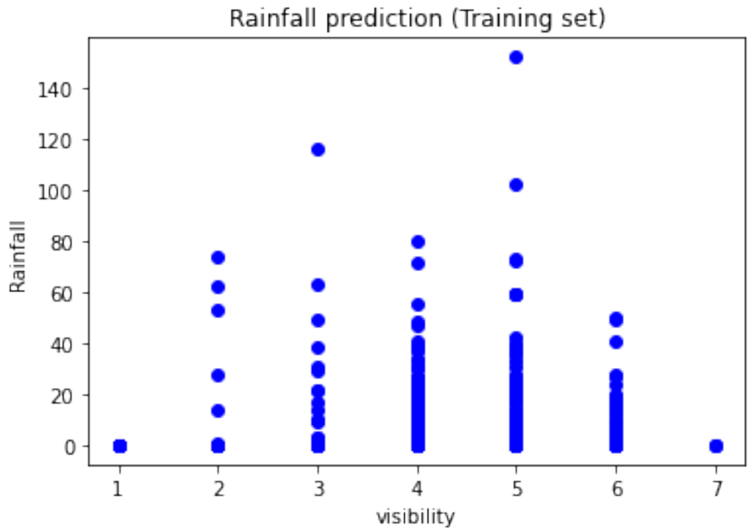
RESLUTS

Screenshot 5.1: Training set (Rainfall Vs Temperature)

(When temperature is 23-27 the Occurrence of Rainfall is Likely High)

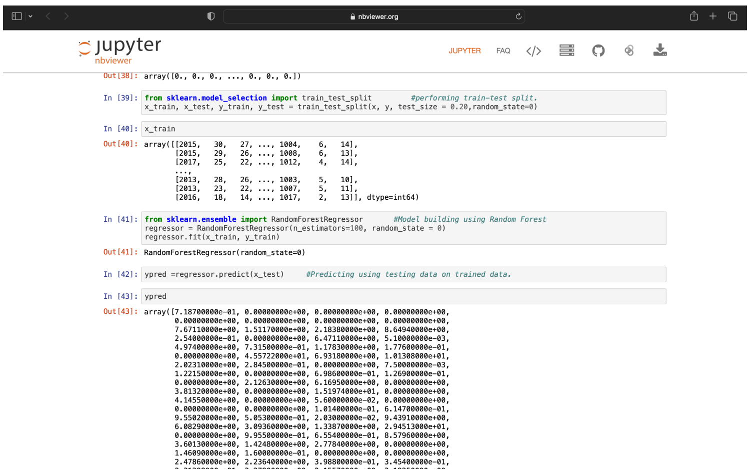
Screenshot 5.2: Training set (Rainfall Vs Humidity)

(When the humidity is 70-95 The Rainfall is likely High)



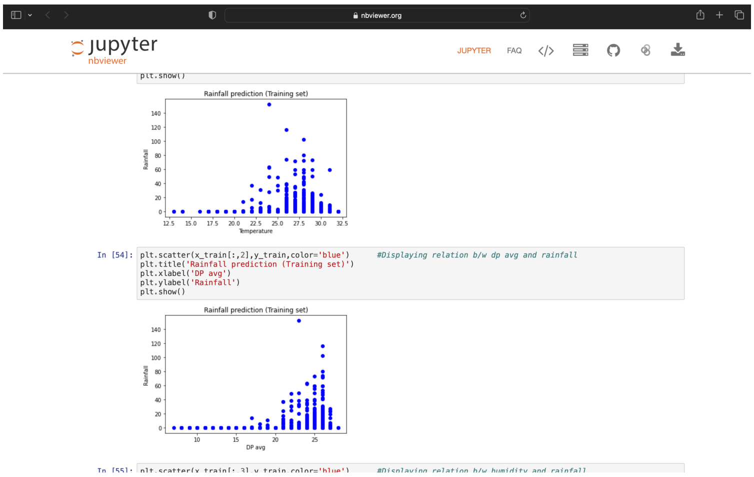
Screenshot 5.3: Training set (Rainfall Vs Visibility)

(When the Visibility On Radar is 3-6 The Rainfall is High)



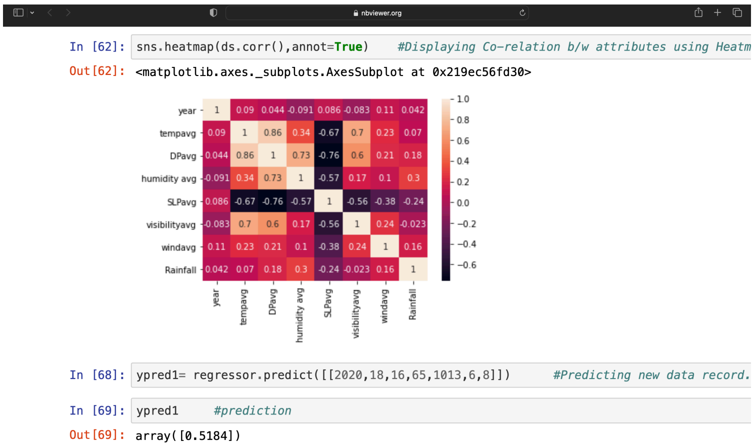
Screenshot 5.4: Compiling to obtain results based on

Previously recorded Rainfalls



Screenshot 5.5: Results afters Compiling obtained

(In the form of Graphs)



Screenshot 5.6: Output formed by Taking all Training sets Data into

Single Graphic Output

TESTING

TESTING

6.1 Feedforward Neural Network

The feed-forward neural network has multilayers for the processing of elements. Each layer processes the input data that it receives and forward the results obtained to the next layer. For this processing, each layer operates independently to generate the resulting that is forwarded to the next layer. The result obtained through processing of each layer is ultimately obtained from the output layer. Between input and output layer; there are hidden layers. The elements that process the input data work like the neuron's in the human brain, these are called artificial neuron's . The neuron's in the layers send messages or information to other neuron's through a channel called connections.

6.2 Back propagation Algorithm

The feed forward back propagation is used to detect the error and consequently highlight the performance of the network using the certain inputs, number of neuron's and to check the validity and accuracy of the output obtained. In the back-propagation model by the ANN(Artificial neural network); weights are decrypted and adjusted in the neural network. The system performs several cycles of back propagation with the input data to get the desired output (Y.H.Zweiri, J.F.Whidborne, & L.D.Seneviratne, 2002). The back propagation a very simple yet efficient algorithm, it consists of (n) number of processing elements with functions of input and output.

6.3 Adaptive Neuro Fizzy Inference System

The ANFIS(Adaptive neuro Fuzzy inference system) is an efficient machine learning and artificial intelligence network that is sometimes advantaged over the neural networks. The ANFIS aims at reducing the complexity of the operation and simplify it to get the desired results and output. It also uses the neuron's for processing the data, the neuron's work as nodes. The neuro-fuzzy system introduces a set of rules for each operation that also stores the data and information for the future operations. The rules introduced depend on the inputs and outputs. It has a domain knowledge which is commonly practised for obtaining the outputs. The concepts of adaptive networking are used with certain techniques to process the desired output. The output depends on the updating parameters and their collection. The node is the processing unit of the neuro-fuzzy.

6.3 Hybrid Learning Algorithm

In the ANFIS architecture there are five layers. The first and the fourth layer contain the parameters that can be updated time to time. But the first layer is nonlinear while the fourth is linear. Therefore, both the parameters need to be updated through the learning method which is capable of training linear and nonlinear simultaneously, hybrid system is the one introduced in 1993 by Jang, that train both layers at the same time (Faulina & Suhartono, 2013). The ANFIS network is trained through Hybrid learning algorithm. It uses descent gradient to denote the errors by forward pass and backward pass in order to train layer 1 and layer 4 at the same time.

CONCLUSION

CONCLUSION

The study aimed at building a predicting system using neural networks that could predict monthly rainfall accurately and efficiently with minimum error. The study incorporated different areas and used their rainfall data with different neural networks like ANFIS and ANN, through training the networks with these inputs and outputs. The trained data is tested and then validated by making a comparison between actual and predicted data. The system used feature extraction to deduce the output prediction that could be more precise and accurate. The neural networks with different algorithms and functions were trained with rainfall parameters and the previous rainfall data to predict the results in this study. After training and testing; the results were compared to check the efficiency of the system; the RMSE's were recorded to make sure that the system will operate not only to make the prediction but also the accurate data will be obtained. The study utilised back propagation, NARX and Hybrid algorithms to forecast the rainfall.

BIBLIOGRAPHY

BIBLIOGRAPHY

8.1 REFERENCES:

HTTPS://WWW.SKYMETWEATHER.COM/15-DAYS-RAINFALL-FORECAST-FOR-INDIA/

HTTPS://WWW.SCIENCEDIRECT.COM/SCIENCE/ARTICLE/PII/S266682702100102X

HTTPS://WWW.GEEKSFORGEEKS.ORG/RAINFALL-PREDICTION-USING-MACHINE-LEARNING-PYTHON/

HTTPS://WWW.IJRASET.COM/RESEARCH-PAPER/RAINFALL-PREDICTION-USING-ML

8.2 GITHUB LINK:

https://github.com/Rajeshpatharla/rainfallprediction.git

PAPER PUBLICATION

CERTIFICATES