NALAIYATHIRAN

PROJECT BASED EXPERIENTIAL LEARNING

EXPLORATORY ANALYSIS OF RAIN FALL DATA IN INDIA FOR AGRICULTURE

TEAM ID : PNT2022TMID17010

EXPLORATORY ANALYSIS OF RAIN FALL DATA IN INDIA FOR AGRICULTURE

A PROJECT REPORT ON

HX 8001 PROFESSIONAL READINESS FOR INNOVATION, EMPLOYABILITY AND ENTREPRENEURSHIP
(Naalaiya Thiran Program)

Submitted by

MANIKANDAN K	**ROLL.No - 6113191041053**
MARIYAPPAN K	**ROLL.No - 6113191041055**
MEIYAZHAGAN M	**ROLL.No - 6113191041056**
JEEVANANDHAN B	**ROLL.No - 6113191041034**

BACHELOR OF ENGINEERING

IN

ELECTRONICS AND COMMUNICATION ENGINEERING

MAHENDRA ENGINEERING COLLEGE

NOVEMBER 2022

INDEX

S.No.	TOPIC	PAGE No.
1	**INTRODUCTION**	1
1.1	Project Overview	1
1.2	Purpose	3
2	**LITERATURE SURVEY**	4
2.1	Existing problem	4
2.2	References	4
2.2	Problem Statement Definition	5
3	**IDEATION & PROPOSED SOLUTION**	6
3.1	Empathy Map Canvas	6
3.2	Ideation & Brainstorming	7
3.3	Proposed Solution	7
3.4	Problem Solution fit	9
4	**REQUIREMENT ANALYSIS**	10
4.1	Functional requirement	10
4.2	Non-Functional requirements	11
5	**PROJECT DESIGN**	12
5.1	Data Flow Diagrams	12
5.2	Solution & Technical Architecture	13
5.3	User Stories	16
6	**PROJECT PLANNING & SCHEDULING**	17
6.1	Sprint Planning & Estimation	17
6.2	Sprint Delivery Schedule	20
7	**CODING & SOLUTIONING**	22

7.1	Feature 1	**22**
7.2	Feature 2	**34**
8	**TESTING**	**36**
9	**RESULTS**	**38**
10	**ADVANTAGES & DISADVANTAGES**	**39**
11	**CONCLUSION**	**40**
12	**FUTURE SCOPE**	**41**
13	**REFERENCES**	**43**
	Source Code GitHub & Project Demo Link	**46**

CHAPTER – 1

INTRODUCTION

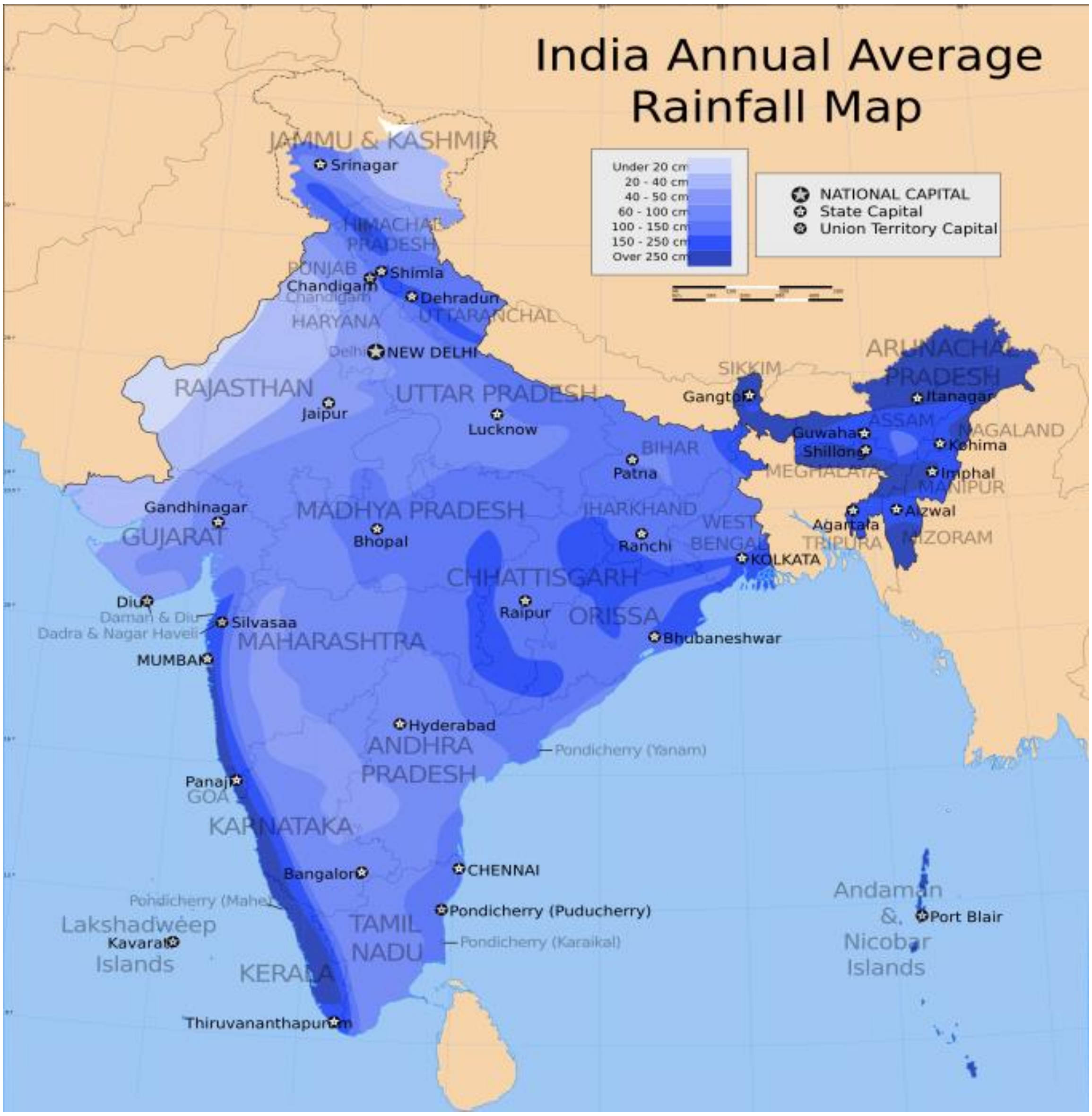
Rainfall has been a major concern these days. Weather conditions have been changing for time being. Rainfall forecasting is important otherwise, itmay lead to many disasters. Irregular heavy rainfall may lead to the destruction of crops, heavy floods that can cause harm to human life. It is important to exactly determine the rainfall for effective use of water resources, crop productivity, and pre-planning of water structures.

This comparative study is conducted concentrating on the following aspects: modeling inputs, Visualizing the data, modeling methods, and pre- processing techniques. The results provide a comparison of various evaluation metrics of these machine learning techniques and their reliabilityto predict rainfall by analyzing the weather data.

We will be using classification algorithms such as Decision tree, Random forest, KNN, and xgboost. We will train and test the data with these algorithms. From this best model is selected and saved in pkl format. Oncethe model is saved, we integrate it with flask application and also deploy the model in IBM.

1.1 Project Overview :

India is an agricultural country and secondary agro based market will be steady with a good monsoon. The economic growth of each year depends on the amount of duration of monsoon rain, bad monsoon can lead to destruction of some crops, which may result in scarcity of some agriculturalproducts which in turn can cause food inflation, insecurity and public unrest. In our analysis we are trying to understand the behavior of rainfall inIndia over the years, by months and different subdivisions.



The Indian summer monsoon typically lasts from June-September,with large areas of western and central India receiving more than 90% of their total annual precipitation during the period, and southern and northwestern India receiving 50%-75% of their total annual rainfall.Overall, monthly totals average 200-300 mm over the country as a whole, with the largest value observed during the heart of the monsoon season in July and August.

There was an early start to monsoon conditions during 1996, with monsoonal rains completely covering India by 30 June, 2 weeks earlier thannormal. There was also a late finish to the monsoon season, with a complete withdrawal of monsoonal rains from India not seen until 11 October. For the area as a whole, rainfall was above normal during June and July, near normal during August, slightly below normal during September, and then substantially above normal during October. In fact, average October totals nearly equalled those observed during September.

1.2 Purpose :

The main purpose of EDA is to help look at data before making any assumptions. It can help identify obvious errors, as well as better understand patterns within the data, detect outliers or anomalous events,find interesting relations among the variables

One of the primary purposes of exploratory data analysis is to identify issues early on to ensure that the data going into machine learning (ML) models is both accurate and consistent and will provide valid, unbiased results.

This analysis will provide useful information for water resources planner, farmers and urban engineers to assess the availability of water and create the storage accordingly. The mean, standard deviation and coefficient of variation of monthly and annual rainfall was calculated to check the rainfallvariability

CHAPTER – 2

LITERATURE SURVEY

2.1 Existing Problem

Weather conditions changes then and often. This can lead to Severe threats to all the living beings including human beings. So predicting weather, especially Irregular heavy rainfall can cause huge floods and economic losses. This also decreases crop productivity and may lead intoFood shortage. Collection of previous 10 years data may give us an idea about the pattern of Rainfall. Using all these Datas, Appropriate farming activities can be performed. These datas can help us in predicting Rainfall.

We will using lot of algorithms like KNN, XGboost, etc. We will train and test the data using these algorithms and predict the best one.

2.2 References

S.No	YEAR	AUTHOR	TITLE	CONTENT
1	2017	S. Cramer, M.Kampouridis, A. A. Freitas, and A. K. Alexandridis	An extensive evaluation of seven machinelearning methods	Rainfall prediction in weather derivatives.
2	2016	S. Zhang, L. Lu, J. Yu, and H. Zhou	Short-term waterlevel prediction using different artificial intelligent models	Geoinformatics, AgroGeoinformati cs
3	2016	S. Zainudin , D. S. Jasim ,and A. A. Bakar	Comparative Analysis of Data Mining Techniques	Rainfall Prediction
4	2013	R. Venkata Ramana, B. Krishna, S. R. Kumar, and N. G. Pandey	Monthly Rainfall Prediction Using Wavelet Neural Network Analysis	Wavelet Neural Network Analysis, Water Resource
5	2013	D. Nayak, A. Mahapatra, and P. Mishra	A Survey on Rainfall Prediction	Artificial Neural Network

2.3 Problem Statement:

Rainfall analyzing is very important because heavy and irregular rainfall can have many impacts like the destruction of crops and farms, damage of property so a better forecasting model is required for an early warning that can reduce the risks to life and property and also helps to manage the agricultural farms in a better way. Heavy rainfall is a cause for natural disasters like floods and drought that square measure encountered by individuals across theworld each year. Many models are developed to evaluate the rainfall and for predicting the likeliness of rain. These models are based on both supervised and unsupervised machine learning algorithms. Taking into consideration of overall rainfall will not help us to know if it rains in specific conditions.Accuracy is the major concern in machine learning.We are going to understand the dataand then train the model accordingly to predict whether if it rains under givenconditions or not.

CHAPTER – 3

1IDEATION & PROPOSED SOLUTION

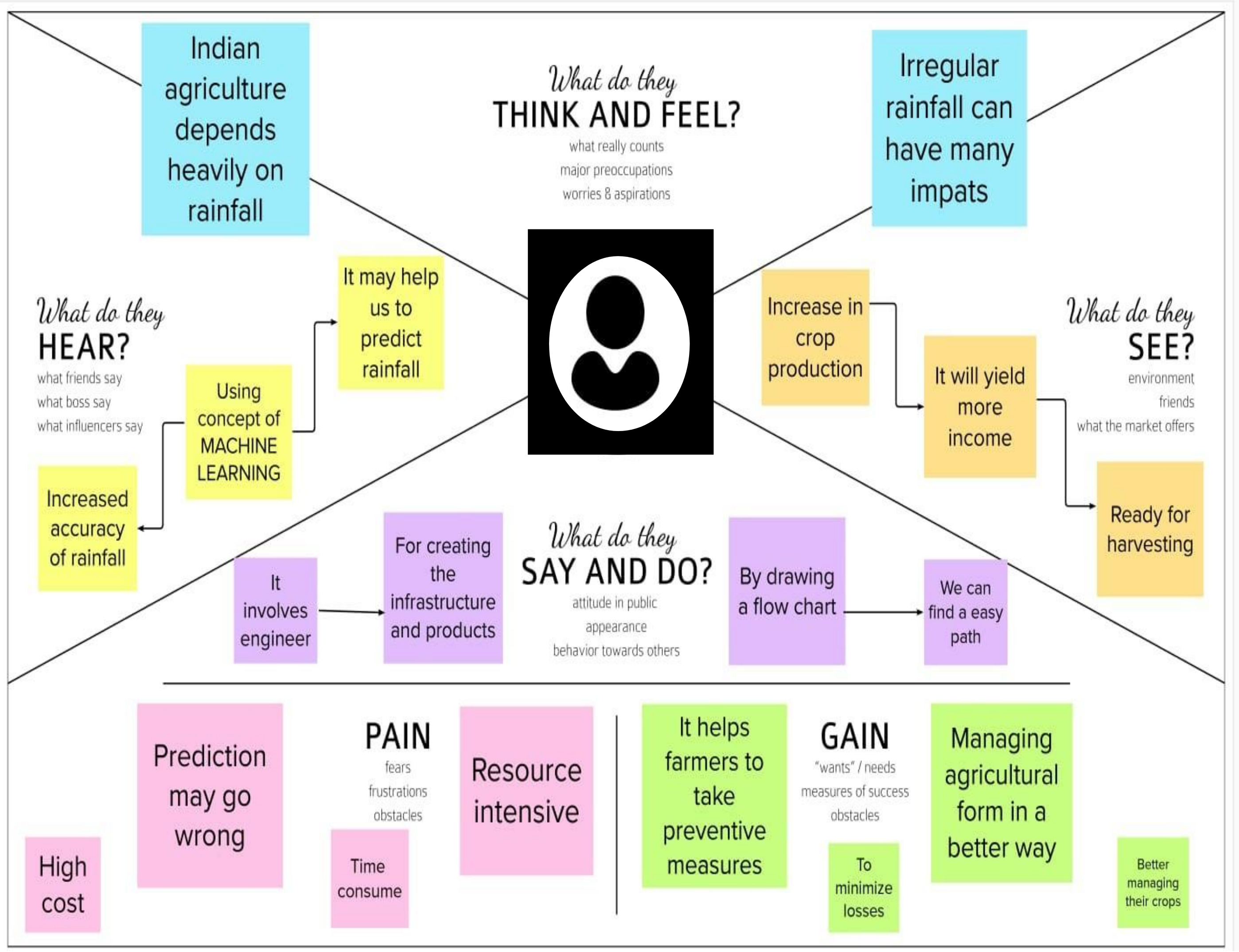
3.1 Empathy Map & Canvas

Definition:

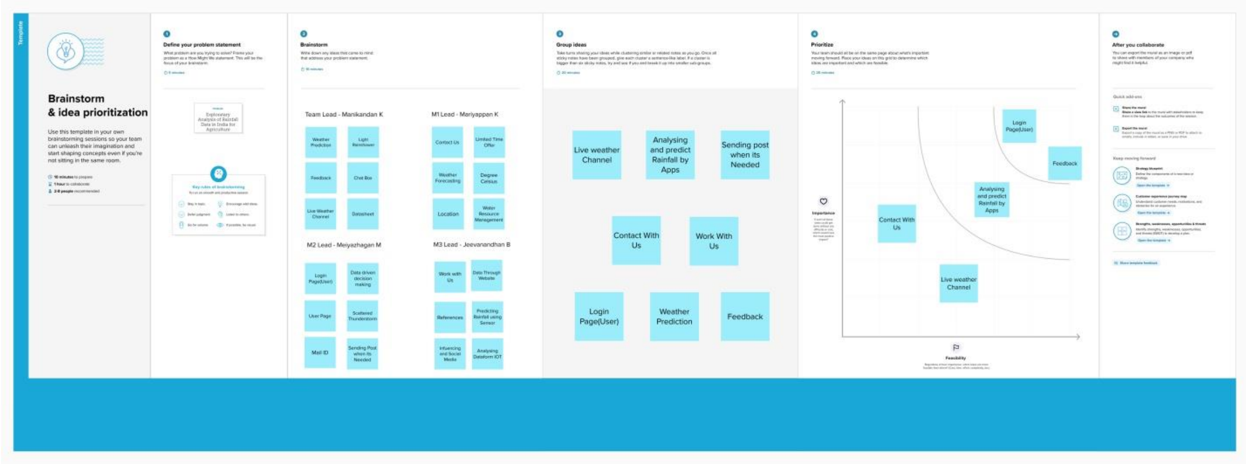
An empathy map is a collaborative visualization used to articulate what weknow about a particular type of user. It externalizes knowledge about usersin order to

1) create a shared understanding of user needs.

2) aid in decision making.



3.2 Ideation & Brainstorming



3.3 Proposed Solution

Proposed Solution Template:

S.No.	Parameter	Description
1	Problem Statement (Problem to be solved)	• Climate is a important aspect of human life. So, the Prediction should accurate as much as possible. In this paper we try to deal with the prediction of the rainfall which is also a major aspect of human life and which provide the major resource of human life which is FreshWater. • Now climate change is the biggest issue all over the world. Peoples are working on to detect the patterns in climate change as it affects the economy in production to infrastructure
2	Idea / Solution description	• In rainfall also making prediction of rainfall is a challenging task with a good accuracy rate. Making prediction on rainfall cannotbe done by the traditional way, so scientist is using machine learning and deeplearning to find out the pattern for rainfallprediction.

3	Novelty / Uniqueness	• This application is useful for the beginners in agriculture. • Seed maturity selection features are available.
4	Social Impact / Customer Satisfaction	• Different types of crops can be planted for good health. • Helps in producing healthy crops and good fields.
5	Business Model (RevenueModel)	• This comparative study is conducted concentrating on the following aspects: modeling inputs, Visualizing the data, modeling methods, and pre-processing techniques. • The results provide a comparison of various evaluation metrics of these machine learning techniques and their reliability to predict rainfall by analyzingthe weather data. • We will be using classification algorithms such as Decision tree, Random forest, KNN, and xgboost.
6	Scalability of the Solution	• When we predict rainfall correctly,it helps growth of crop and yielding will be better.

3.4 Problem – Solution Fit:

The Problem-Solution Fit simply means that you have found a problem with your customer and that the solution you have realized for it actually solves the customer’s problem. It helps entrepreneurs, marketers and corporate innovators identify behavioral patterns and recognize what would work and why.

Purpose:

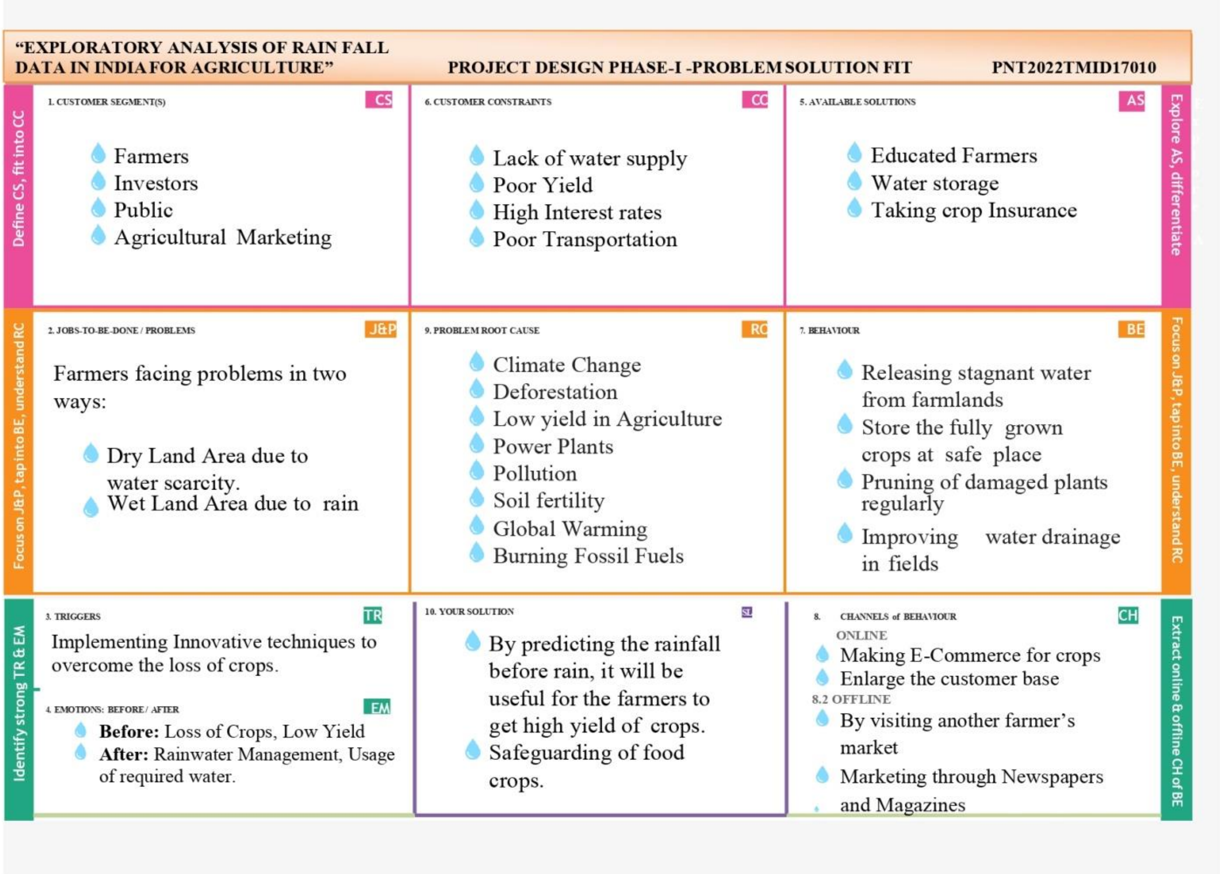
❑Solve complex problems in a way that fits the state of your customers.

❑Succeed faster and increase your solution adoption by tapping into existing mediums and channels of behavior.

❑ Sharpen your communication and marketing strategy with the right triggers and messaging.

❑Increase touch-points with your company by finding the right problembehavior fit and building trust by solving frequent annoyances, or urgent or costly problems.

❑Understand the existing situation in order to improve it for your target group.



CHAPTER – 4
REQUIREMENT ANALYSIS

4.1 Functional Requirements:

Following are the functional requirements of our proposed solution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	**Download and load the dataset**	Download and load the appropriate dataset.
FR-2	**Pre-processing of data**	Preparation of raw data and make it suitable for building of machine learning model.

FR-3	**Building machine learning model**	✓ Exploring the data and choose the suitable algorithm. ✓ Prepare and clean the dataset. ✓ Split the prepared dataset and make cross validation. ✓ Perform machine learning optimisation. ✓ Deploy the model.
FR-4	**Train the data**	Train the model using training set.
FR-5	**Test the data**	At last, test the model for evaluation of final model.

4.2 Non-functional Requirements:

Following are the non-functional requirements of our proposed solution.

FR No.	Non-Functional Requirement	Description

NFR-1	**Usability**	Local presence/traceability of WIS source in the farming community.
NFR-2	**Security**	Providing secure system networks then determine authenticity, originality and security.
NFR-3	**Reliability**	System will operate without failure for a specific period of time.
NFR-4	**Performance**	Our model predictions are same as the true values. So, the performance is higher.
NFR-5	**Availability**	Available to different groups of farmers including women, older persons, etc.
NFR-6	**Scalability**	In our model, Prediction of data will be faultless.

CHAPTER – 5
PROJECT DESIGN

5.1 Data Flow Diagrams:

A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It shows how data enters and leaves the system, what changes the information, and where data is stored.

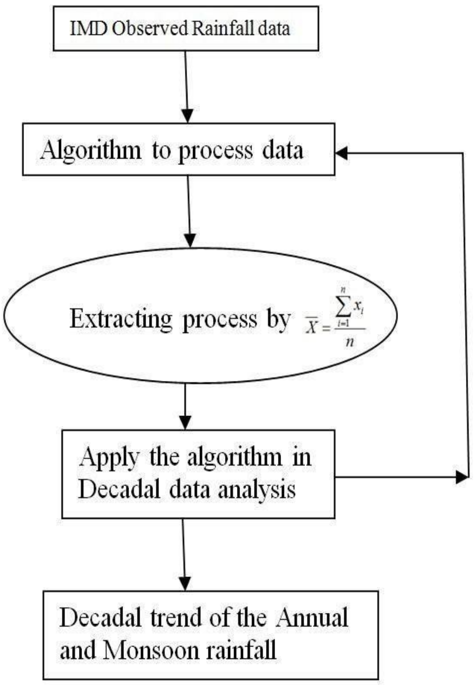


Fig: Data flow diagram to Compute the rainfall data in decadal basis over India

1.First Data's are collected

2.Using algorithms to process the data

3.Extract the process using some formulas

4.Apply that algorithms in decade basis data analysis5.Finally decadal trend of the annual rainfall

5.2 Solution &Technical Architecture:

The Deliverable shall include the architectural diagram as below and the information as per the table1 & table 2 Technology architecture associates application components from application architecture with technology components representing software and hardware components. Its components are generally acquired in the marketplace and can be assembled and configured to constitute the enterprise's technological infrastructure

Table-1 : Components & Technologies:

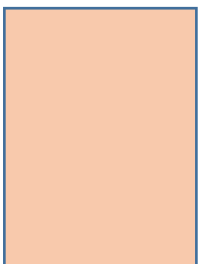
S.No	COMPONENTS	DESCRIPTION	TECHNOLOGY
1.	User interface	To anticipate the data for rainfall, the user engages with the prediction model via a website.	HTML, CSS, JavaScript
2.	Cloud Database	The model receives information froman IBM cloud database.	IBM Cloud DB, ibm_db(python package)
3.	APL	used to expandservice to additional applications	Flask Application
4.	JWT&Sessions	Is employed to extend service tomore applications	PyJWT, Flask Application
5.	Machine LearningModel	This model wascreated to forecast rainfallusing machine learning	Sklearn, Algorithms - DT& MLR
6.	Data processing	preprocessing ofthe data is followed by prediction	Pandas, Numpy, Matplotlib

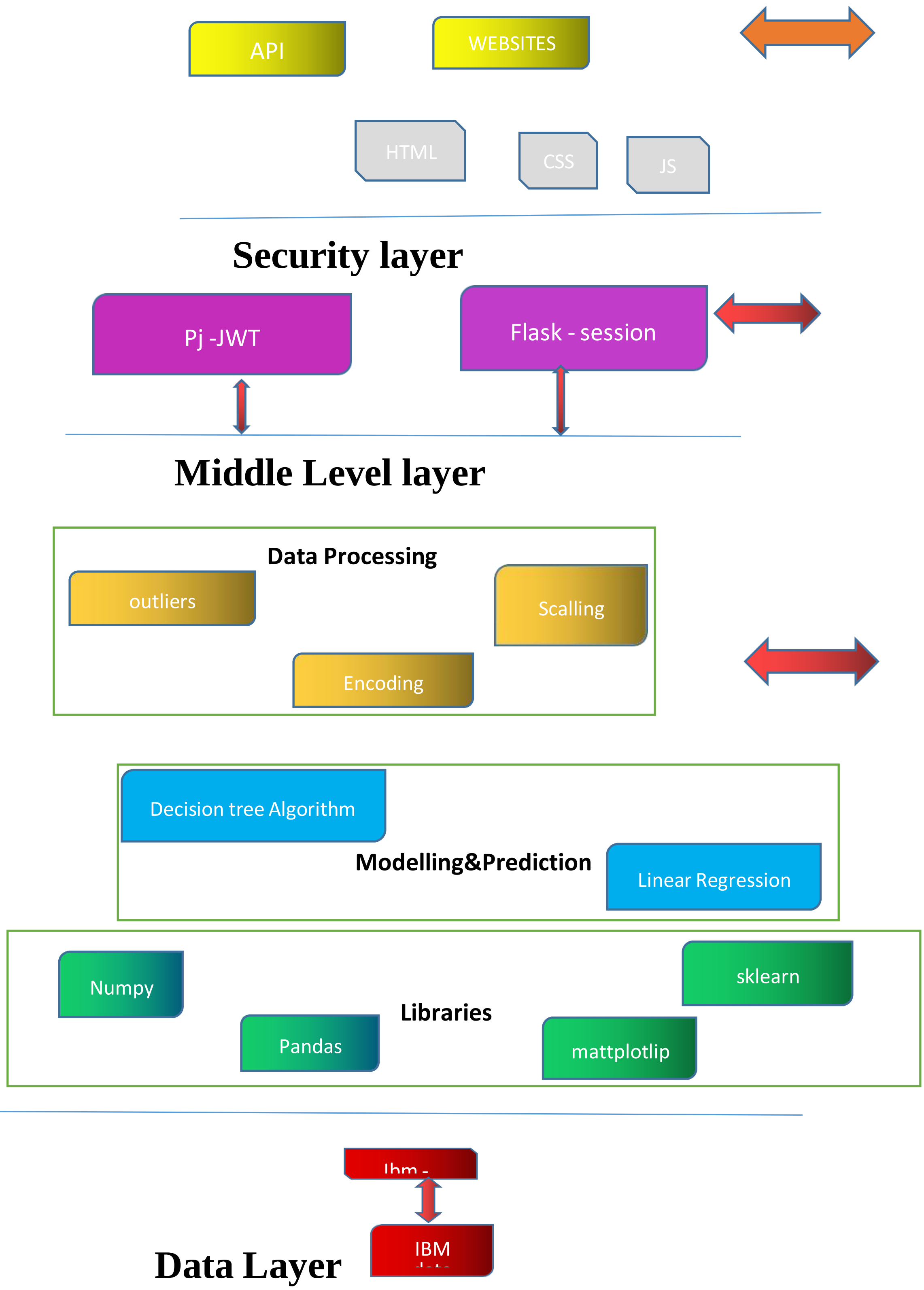
Table-2: Application Characteristics:

S.No	CHARCTERITICS	DESCRIPTION	TECHNOLOGY
1.	Open-Source Frameworks	Backend Framework, CSSStyling framework, Relational Database	PyJWT, Flask, IBM Cloud DB
2.	Security Implementations	Request authentication using JWT Tokens	HS-256, Encryptions,SSL Certs
3.	Scalable Architecture	Support for Multiple Sample prediction usingExcel File	File Pandas, Numpy
4.	Availability	Availability is increased by Distributed Servers in CloudVPS	IBM Cloud Hosting
5.	Performance	The applicationis expected to handle multiplepredictions per second	Load Balancers, Distributed ServerS

Technical Architecture:

Presentation Layer




API
WEBSITES
HTML
CSS
JS
Security layer
Pj -JWT
Flask - session
Middle Level layer
Data Processing
outliers
Scalling
Encoding
Decision tree Algorithm
Modelling&Prediction
Linear Regression
Numpy
sklearn
Libraries
Pandas
mattplotlip
Ibm -
IBM
Data Layer

5.3 User Stories

Use the below template to list all the user stories for the product.

User Type	Functioal Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Mobile user)	Registration	USN-1	As a user, I can register for the application by Filling the form	I can receive confirmation via OTP	High	Sprint - 1
		USN-2	As a user, I will receive confirmation email once I haveregistered for the application	I can receive confirmation email & click confirm	High	Sprint - 1

CHAPTER – 6
PROJECT PLANNING & SCHEDULING

6.1 Sprint Delivery Schedule:

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Rainfall Prediction ML Model (Dataset)	USN-1	Weather Dataset Collection, Data pre-processing, Data Visualization.	5	High	Manikandan, Mariyappan, Meiyazhagan, Jeevanandhan
Sprint-1		USN-2	Train Model using Different machine learning Algorithms	5	High	Manikandan, Mariyappan, Meiyazhagan, Jeevanandhan
Sprint-1		USN-3	Test the model and give best	10	High	Manikandan, Mariyappan, Meiyazhagan, Jeevanandhan
Sprint-2	Registration	USN-4	As a user, they can register for the application through Gmail. Password is set up.	5	Medium	Manikandan, Mariyappan, Meiyazhagan, Jeevanandhan
Sprint-2	Login	USN-5	As a user, they can log into the application byentering email & password	5	Medium	Manikandan, Mariyappan, Meiyazhagan, Jeevanandhan
Sprint-2		USN-6	Credentials should be used for multiple systems and verified	4	Medium	Manikandan, Mariyappan, Meiyazhagan, Jeevanandhan

Sprint-2	Dashboard	USN-7	Attractive dashboard forecasting live weather	6	Low	Manikandan, Mariyappan, Meiyazhagan, Jeevanandhan
Sprint-3	Rainfall Prediction	USN-8	User enter the location, temperature, humidity	10	High	Manikandan, Mariyappan, Meiyazhagan, Jeevanandhan
Sprint-3		USN-9	Predict the rainfall and display the result	10	High	Manikandan, Mariyappan, Meiyazhagan, Jeevanandhan
Sprint-4	Testing	USN-10	Test the application	10	High	Manikandan, Mariyappan, Meiyazhagan, Jeevanandhan
Sprint-4	Deploy Model	USN-11	Deploy the model in IBM cloud to make userfriendly application	10	High	Manikandan, Mariyappan, Meiyazhagan, Jeevanandhan

Project Tracker, Velocity & Burndown Chart:

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	20	6 Days	24 Oct 2022	29 Oct 2022	20	29 Oct 2022
Sprint-2	20	6 Days	31 Oct 2022	05 Nov 2022	20	05 Nov 2022
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022	20	12 Nov 2022
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022	20	19 Nov 2022

Velocity:

Imagine we have a 5-day sprint duration, and the velocity of the team is 20 (points per sprint). Let's calculate the team's average velocity (AV) periteration unit (story points per day)

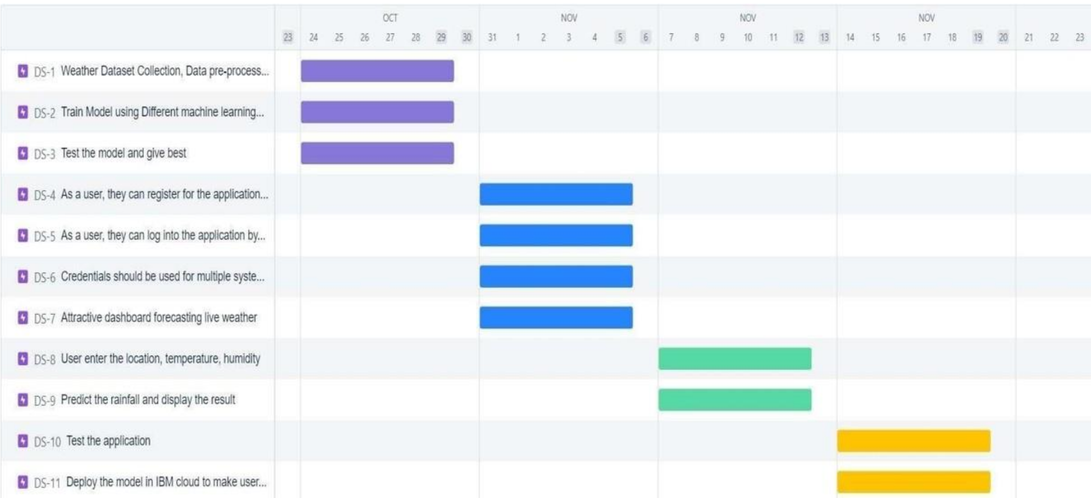
AV= Sprint duration/ Velocity = 20/5 =4

Total Average Velocity= 4

Burndown Chart:

A burn down chart is a graphical representation of work left to do versus time. It is often used in agile software development methodologies such as Scrum. However, burn down charts canbe applied to any project containing measurable progress over time.

Tool : Jira Software



6.2 Sprint Planning & Estimation:

TITLE	DESCRIPTION	DATE
Literature survey & information gathering	Collect the relevant information on project use case, refer the existing solutions, technical papers, research publications etc.	12 SEPTEMBER 2022
Prepare empathy map	Prepare Empathy Map Canvas and List ofproblem statements	14 SEPTEMBER 2022
Ideation	List the ideas by organizing the brainstorming session and prioritize the top 3 ideas based on the feasibility & importance	22 SEPTEMBER 2022
Proposed solution	Prepare the proposed solution document, which includes the novelty, feasibility of idea, business model, social impact, scalability of solution, etc.	24 SEPTEMBER 2022
Problem solution fit	Prepare problem - solution fit document & Solution Architecture	**19 OCTOBER 2022 (** RESUBMITTED)

Solution Architecture	Prepare Solution Architecture document	19 OCTOBER 2022 (RESUBMITTED)
Customer journey	Prepare the customerjourney maps to understand the user interactions & experiences with the application	19 OCTOBER 2022
Solution requirement	Prepare the Functional Requirement Document	19 OCTOBER 2022
Data flow diagrams	Prepare the Data Flow Diagrams	19 OCTOBER 2022
Technology architecture	Prepare Technology Architecture of the solution	19 OCTOBER 2022
Prepare Milestone & activity list	Prepare the Milestone& activity list of the project	26 OCTOBER 2022

CHAPTER – 7
CODING & SOLUTIONING

(Explain the features added in the project along with code)

7.1Feature 1

```
import pandas as pd
import numpy as np
import seaborn as sns
import  matplotlib.pyplot as plt
from sklearn import preprocessing
from sklearn import model_selection
from sklearn import metrics
from sklearn import linear_model
from sklearn import ensemble
from sklearn import tree
from sklearn import svm
import xgboost
import sklearn

data = pd.read_csv("/content/weatherAUS.csv - weatherAUS.csv.csv")
data.head()

         Date Location  MinTemp  MaxTemp  Rainfall  Evaporation
Sunshine  \
0  2008-12-01   Albury     13.4     22.9       0.6          NaN
NaN
1  2008-12-02   Albury      7.4     25.1       0.0          NaN
NaN
2  2008-12-03   Albury     12.9     25.7       0.0          NaN
NaN
3  2008-12-04   Albury      9.2     28.0       0.0          NaN
NaN
4  2008-12-05   Albury     17.5     32.3       1.0          NaN
NaN

  WindGustDir  WindGustSpeed WindDir9am  ... Humidity9am  Humidity3pm
\
0           W           44.0          W  ...        71.0         22.0

1         WNW           44.0        NNW  ...        44.0         25.0

2         WSW           46.0          W  ...        38.0         30.0

3          NE           24.0         SE  ...        45.0         16.0

4           W           41.0        ENE  ...        82.0         33.0


   Pressure9am  Pressure3pm  Cloud9am  Cloud3pm  Temp9am  Temp3pm
RainToday  \
0       1007.7       1007.1       8.0       NaN     16.9     21.8
No
1       1010.6       1007.8       NaN       NaN     17.2     24.3
No
2       1007.6       1008.7       NaN       2.0     21.0     23.2
```

```
No
3       1017.6       1012.8       NaN       NaN      18.1      26.5
No
4       1010.8       1006.0       7.0       8.0      17.8      29.7
No

   RainTomorrow
0            No
1            No
2            No
3            No
4            No

[5 rows x 23 columns]
```

```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 145460 entries, 0 to 145459
Data columns (total 23 columns):
 #   Column         Non-Null Count   Dtype
---  ------         --------------   -----
 0   Date           145460 non-null  object
 1   Location       145460 non-null  object
 2   MinTemp        143975 non-null  float64
 3   MaxTemp        144199 non-null  float64
 4   Rainfall       142199 non-null  float64
 5   Evaporation    82670 non-null   float64
 6   Sunshine       75625 non-null   float64
 7   WindGustDir    135134 non-null  object
 8   WindGustSpeed  135197 non-null  float64
 9   WindDir9am     134894 non-null  object
 10  WindDir3pm     141232 non-null  object
 11  WindSpeed9am   143693 non-null  float64
 12  WindSpeed3pm   142398 non-null  float64
 13  Humidity9am    142806 non-null  float64
 14  Humidity3pm    140953 non-null  float64
 15  Pressure9am    130395 non-null  float64
 16  Pressure3pm    130432 non-null  float64
 17  Cloud9am       89572 non-null   float64
 18  Cloud3pm       86102 non-null   float64
 19  Temp9am        143693 non-null  float64
 20  Temp3pm        141851 non-null  float64
 21  RainToday      142199 non-null  object
 22  RainTomorrow   142193 non-null  object
dtypes: float64(16), object(7)
memory usage: 25.5+ MB
```

```
data.shape
```

```
(145460, 23)
```

```
print('\nUnique Values: ',data.nunique())
```

```
Unique Values:  Date             3436
Location          49
MinTemp          389
MaxTemp          505
Rainfall         681
Evaporation      358
Sunshine         145
WindGustDir       16
WindGustSpeed     67
WindDir9am        16
WindDir3pm        16
WindSpeed9am      43
WindSpeed3pm      44
Humidity9am      101
Humidity3pm      101
Pressure9am      546
Pressure3pm      549
Cloud9am          10
Cloud3pm          10
Temp9am          441
Temp3pm          502
RainToday          2
RainTomorrow       2
dtype: int64
```

```
print('\nMissing Values: ',data.isna().sum())
```

```
Missing Values:  Date                0
Location            0
MinTemp          1485
MaxTemp          1261
Rainfall         3261
Evaporation     62790
Sunshine        69835
WindGustDir     10326
WindGustSpeed   10263
WindDir9am      10566
WindDir3pm       4228
WindSpeed9am     1767
WindSpeed3pm     3062
Humidity9am      2654
Humidity3pm      4507
Pressure9am     15065
Pressure3pm     15028
Cloud9am        55888
Cloud3pm        59358
Temp9am          1767
```

```
Temp3pm          3609
RainToday        3261
RainTomorrow     3267
dtype: int64
```

```
data.describe()
```

```
             MinTemp        MaxTemp       Rainfall   Evaporation  \
count  143975.000000  144199.000000  142199.000000  82670.000000
mean       12.194034      23.221348       2.360918      5.468232
std         6.398495       7.119049       8.478060      4.193704
min        -8.500000      -4.800000       0.000000      0.000000
25%         7.600000      17.900000       0.000000      2.600000
50%        12.000000      22.600000       0.000000      4.800000
75%        16.900000      28.200000       0.800000      7.400000
max        33.900000      48.100000     371.000000    145.000000

           Sunshine  WindGustSpeed   WindSpeed9am   WindSpeed3pm  \
count  75625.000000  135197.000000  143693.000000  142398.000000
mean       7.611178      40.035230      14.043426      18.662657
std        3.785483      13.607062       8.915375       8.809800
min        0.000000       6.000000       0.000000       0.000000
25%        4.800000      31.000000       7.000000      13.000000
50%        8.400000      39.000000      13.000000      19.000000
75%       10.600000      48.000000      19.000000      24.000000
max       14.500000     135.000000     130.000000      87.000000

         Humidity9am    Humidity3pm   Pressure9am    Pressure3pm  \
count  142806.000000  140953.000000  130395.00000  130432.000000
mean       68.880831      51.539116    1017.64994    1015.255889
std        19.029164      20.795902       7.10653       7.037414
min         0.000000       0.000000     980.50000     977.100000
25%        57.000000      37.000000    1012.90000    1010.400000
50%        70.000000      52.000000    1017.60000    1015.200000
75%        83.000000      66.000000    1022.40000    1020.000000
max       100.000000     100.000000    1041.00000    1039.600000

           Cloud9am      Cloud3pm        Temp9am       Temp3pm
count  89572.000000  86102.000000  143693.000000  141851.00000
mean       4.447461      4.509930      16.990631      21.68339
std        2.887159      2.720357       6.488753       6.93665
min        0.000000      0.000000      -7.200000      -5.40000
25%        1.000000      2.000000      12.300000      16.60000
50%        5.000000      5.000000      16.700000      21.10000
75%        7.000000      7.000000      21.600000      26.40000
max        9.000000      9.000000      40.200000      46.70000
```

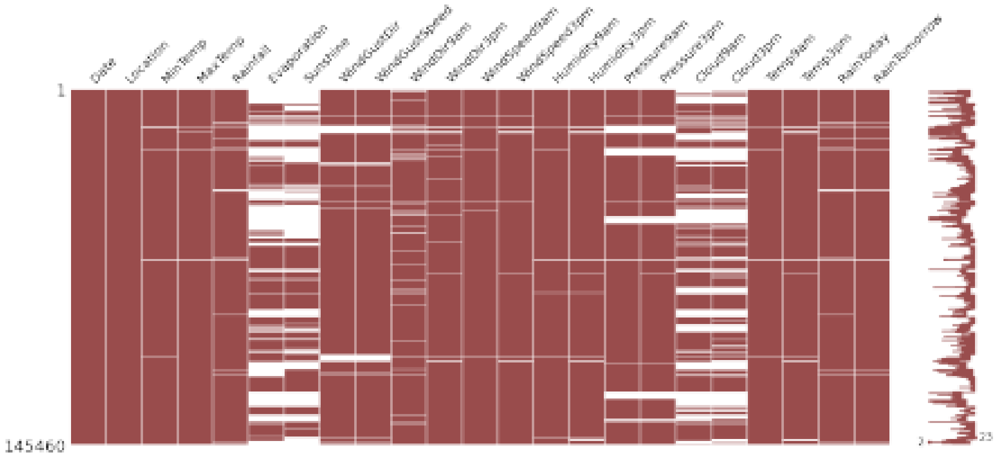
```
data.isnull().sum()
```

```
Date                0
Location            0
```

```
MinTemp            1485
MaxTemp            1261
Rainfall           3261
Evaporation       62790
Sunshine          69835
WindGustDir       10326
WindGustSpeed     10263
WindDir9am        10566
WindDir3pm         4228
WindSpeed9am       1767
WindSpeed3pm       3062
Humidity9am        2654
Humidity3pm        4507
Pressure9am       15065
Pressure3pm       15028
Cloud9am          55888
Cloud3pm          59358
Temp9am            1767
Temp3pm            3609
RainToday          3261
RainTomorrow       3267
dtype: int64
```

```
import missingno as msno
msno.matrix(data,color=(0.60,0.300,0.300),fontsize=20)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7ff0c1783bd0>
```




```
data_cat = data[['RainToday', 'WindGustDir', 'WindDir9am',
'WindDir3pm']]
data.drop(columns=['Evaporation', 'Sunshine', 'Cloud9am',
'Cloud3pm'],axis=1,inplace=True)
data.drop(columns=['RainToday', 'WindGustDir', 'WindDir9am',
'WindDir3pm'],axis=1,inplace=True)
```

```
data['MinTemp'].fillna(data['MinTemp'].mean(), inplace=True)

data['MaxTemp'].fillna (data['MaxTemp'].mean(), inplace=True)

data['Rainfall'].fillna (data['Rainfall'].mean(), inplace=True)

data['WindGustSpeed'].fillna (data['WindGustSpeed'].mean(),
inplace=True)

data['WindSpeed9am'].fillna (data['WindSpeed9am'].mean(),
inplace=True)

data['WindSpeed3pm'].fillna (data['WindSpeed3pm'].mean(),
inplace=True)

data['Humidity9am'].fillna (data[ 'Humidity9am'].mean(), inplace=True)

data['Humidity3pm'].fillna (data['Humidity3pm'].mean(), inplace=True)

data['Pressure9am'].fillna (data[ 'Pressure9am'].mean(), inplace=True)

data['Pressure3pm'].fillna (data['Pressure3pm'].mean(), inplace=True)

data['Temp9am'].fillna (data['Temp9am'].mean(),inplace=True)

data['Temp3pm'].fillna(data['Temp3pm'].mean(),inplace=True)


cat_names=data_cat.columns

import numpy as np


from sklearn.impute import SimpleImputer
imp_mode= SimpleImputer (missing_values=np.nan, strategy =
'most_frequent')

data_cat= imp_mode.fit_transform(data_cat)

data_cat = pd.DataFrame(data_cat,columns=cat_names)

data = pd.concat([data, data_cat],axis=1)

data.corr()
```

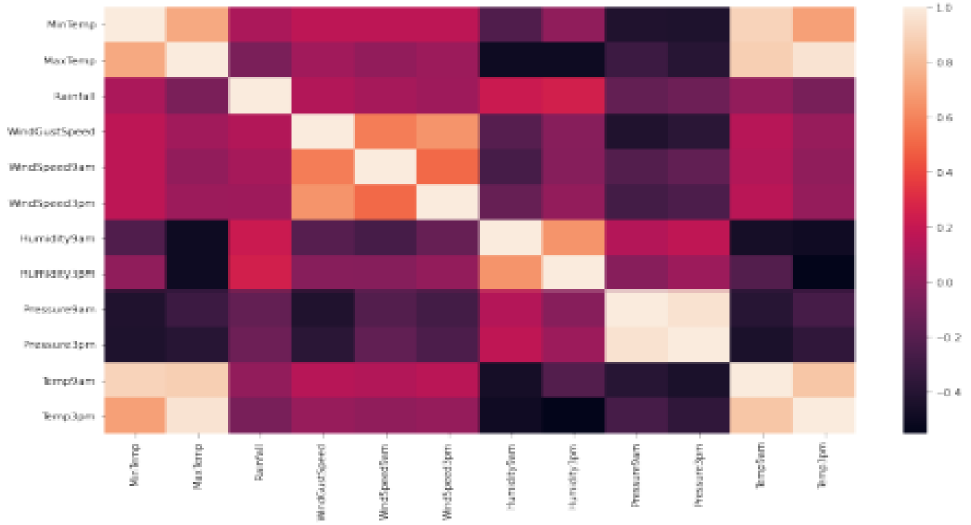
```
                MinTemp   MaxTemp  Rainfall  WindGustSpeed
WindSpeed9am  \
MinTemp        1.000000  0.733400  0.102706       0.172553
0.173404
MaxTemp        0.733400  1.000000 -0.074040       0.065895
```

```
0.014294
Rainfall        0.102706 -0.074040  1.000000       0.126446
0.085925
WindGustSpeed   0.172553  0.065895  0.126446       1.000000
0.577319
WindSpeed9am    0.173404  0.014294  0.085925       0.577319
1.000000
WindSpeed3pm    0.173058  0.049717  0.056527       0.657243
0.512427
Humidity9am    -0.230970 -0.497927  0.221380      -0.207964      -
0.268271
Humidity3pm     0.005995 -0.498760  0.248905      -0.025355      -
0.030887
Pressure9am    -0.423584 -0.308309 -0.159055      -0.425760      -
0.215339
Pressure3pm    -0.433147 -0.396622 -0.119541      -0.383938      -
0.165388
Temp9am         0.897692  0.879170  0.011069       0.145904
0.127592
Temp3pm         0.699211  0.968713 -0.077684       0.031884
0.004476

               WindSpeed3pm  Humidity9am  Humidity3pm  Pressure9am  \
MinTemp            0.173058    -0.230970     0.005995    -0.423584
MaxTemp            0.049717    -0.497927    -0.498760    -0.308309
Rainfall           0.056527     0.221380     0.248905    -0.159055
WindGustSpeed      0.657243    -0.207964    -0.025355    -0.425760
WindSpeed9am       0.512427    -0.268271    -0.030887    -0.215339
WindSpeed3pm       1.000000    -0.143458     0.016275    -0.277604
Humidity9am       -0.143458     1.000000     0.659072     0.131503
Humidity3pm        0.016275     0.659072     1.000000    -0.025848
Pressure9am       -0.277604     0.131503    -0.025848     1.000000
Pressure3pm       -0.239659     0.176009     0.048695     0.959662
Temp9am            0.161060    -0.469641    -0.216964    -0.397131
Temp3pm            0.027587    -0.490709    -0.555608    -0.265532

               Pressure3pm   Temp9am   Temp3pm
MinTemp          -0.433147  0.897692  0.699211
MaxTemp          -0.396622  0.879170  0.968713
Rainfall         -0.119541  0.011069 -0.077684
WindGustSpeed    -0.383938  0.145904  0.031884
WindSpeed9am     -0.165388  0.127592  0.004476
WindSpeed3pm     -0.239659  0.161060  0.027587
Humidity9am       0.176009 -0.469641 -0.490709
Humidity3pm       0.048695 -0.216964 -0.555608
Pressure9am       0.959662 -0.397131 -0.265532
Pressure3pm       1.000000 -0.441459 -0.360707
Temp9am          -0.441459  1.000000  0.846141
Temp3pm          -0.360707  0.846141  1.000000
```

```
cor=data.corr()
```

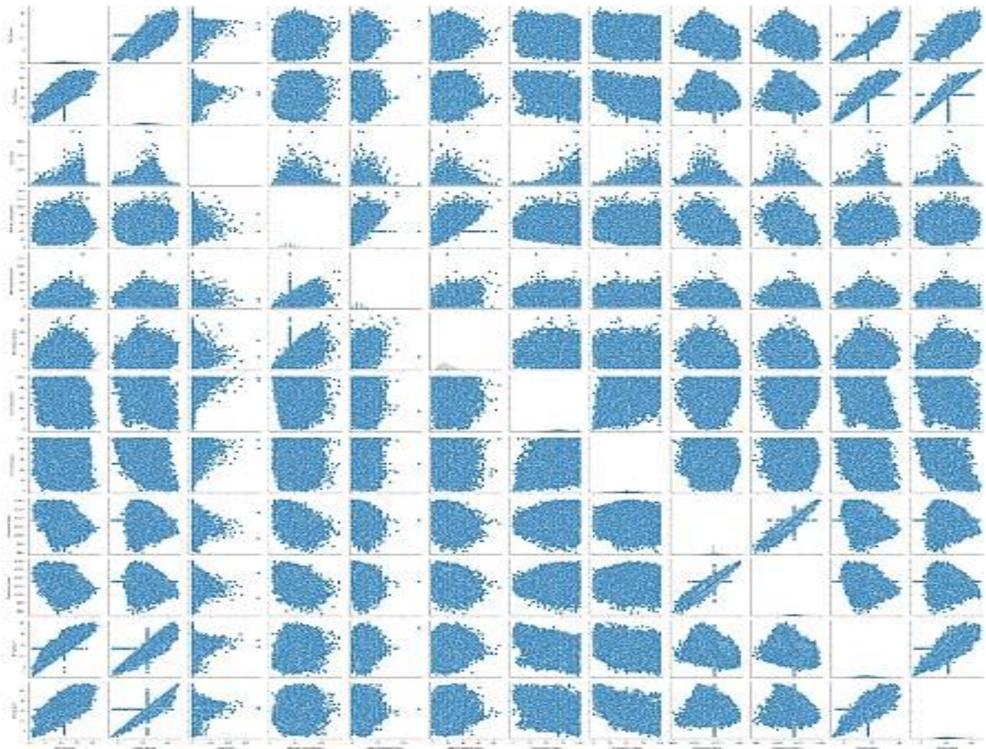
```
plt.figure(figsize=(15,8))
sns.heatmap(data=cor,xticklabels=cor.columns.values,yticklabels=cor.co
lumns.values)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fb321e2bc10>
```




```
sns.pairplot(data)
```

```
<seaborn.axisgrid.PairGrid at 0x7fb31479d610>
```



```
plt.figure(figsize=(15,8))
data.boxplot()
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fb30f56ec50>
```

```
0         W         WNW
1       NNW         WSW
2         W         WSW
3        SE           E
4       ENE          NW
```

```
df.shape
```

```
(142193, 19)
```

```
x=df.drop('RainTomorrow',axis=1)
y=df['RainTomorrow']
```

```
x.head()
```

```
         Date Location  MinTemp  MaxTemp  Rainfall  WindGustSpeed  \
0  2008-12-01   Albury     13.4     22.9       0.6           44.0
1  2008-12-02   Albury      7.4     25.1       0.0           44.0
2  2008-12-03   Albury     12.9     25.7       0.0           46.0
3  2008-12-04   Albury      9.2     28.0       0.0           24.0
4  2008-12-05   Albury     17.5     32.3       1.0           41.0

   WindSpeed9am  WindSpeed3pm  Humidity9am  Humidity3pm
Pressure9am  \
0          20.0          24.0         71.0         22.0       1007.7

1           4.0          22.0         44.0         25.0       1010.6

2          19.0          26.0         38.0         30.0       1007.6

3          11.0           9.0         45.0         16.0       1017.6

4           7.0          20.0         82.0         33.0       1010.8


   Pressure3pm  Temp9am  Temp3pm RainToday WindGustDir WindDir9am
WindDir3pm
0       1007.1     16.9     21.8        No           W          W
WNW
1       1007.8     17.2     24.3        No         WNW        NNW
WSW
2       1008.7     21.0     23.2        No         WSW          W
WSW
3       1012.8     18.1     26.5        No          NE         SE
E
4       1006.0     17.8     29.7        No           W        ENE
NW
```

```
x_main=x.drop(['Date','Location','WindGustDir','WindDir9am','WindDir3p
m'],axis=1)
x_main.head()
```

```
4          7.0          20.0          82.0          33.0        1010.8


   Pressure3pm  Temp9am  Temp3pm RainTomorrow RainToday WindGustDir  \
0       1007.1     16.9     21.8           No        No           W
1       1007.8     17.2     24.3           No        No         WNW
2       1008.7     21.0     23.2           No        No         WSW
3       1012.8     18.1     26.5           No        No          NE
4       1006.0     17.8     29.7           No        No           W

  WindDir9am WindDir3pm
0          W        WNW
1        NNW        WSW
2          W        WSW
3         SE          E
4        ENE         NW

df=data.dropna()
df.head()

         Date Location  MinTemp  MaxTemp  Rainfall  WindGustSpeed  \
0  2008-12-01   Albury     13.4     22.9       0.6           44.0
1  2008-12-02   Albury      7.4     25.1       0.0           44.0
2  2008-12-03   Albury     12.9     25.7       0.0           46.0
3  2008-12-04   Albury      9.2     28.0       0.0           24.0
4  2008-12-05   Albury     17.5     32.3       1.0           41.0

   WindSpeed9am  WindSpeed3pm  Humidity9am  Humidity3pm
Pressure9am  \
0         20.0          24.0          71.0          22.0        1007.7

1          4.0          22.0          44.0          25.0        1010.6

2         19.0          26.0          38.0          30.0        1007.6

3         11.0           9.0          45.0          16.0        1017.6

4          7.0          20.0          82.0          33.0        1010.8


   Pressure3pm  Temp9am  Temp3pm RainTomorrow RainToday WindGustDir  \
0       1007.1     16.9     21.8           No        No           W
1       1007.8     17.2     24.3           No        No         WNW
2       1008.7     21.0     23.2           No        No         WSW
3       1012.8     18.1     26.5           No        No          NE
4       1006.0     17.8     29.7           No        No           W

  WindDir9am WindDir3pm
```

```
   MinTemp  MaxTemp  Rainfall  WindGustSpeed  WindSpeed9am
WindSpeed3pm  \
0     13.4     22.9       0.6           44.0          20.0
24.0
1      7.4     25.1       0.0           44.0           4.0
22.0
2     12.9     25.7       0.0           46.0          19.0
26.0
3      9.2     28.0       0.0           24.0          11.0
9.0
4     17.5     32.3       1.0           41.0           7.0
20.0

   Humidity9am  Humidity3pm  Pressure9am  Pressure3pm  Temp9am
Temp3pm  \
0         71.0         22.0       1007.7       1007.1     16.9
21.8
1         44.0         25.0       1010.6       1007.8     17.2
24.3
2         38.0         30.0       1007.6       1008.7     21.0
23.2
3         45.0         16.0       1017.6       1012.8     18.1
26.5
4         82.0         33.0       1010.8       1006.0     17.8
29.7

  RainToday
0        No
1        No
2        No
3        No
4        No
```

```
x_p=pd.get_dummies(x_main,columns=['RainToday'])
x_p.head()
```

```
   MinTemp  MaxTemp  Rainfall  WindGustSpeed  WindSpeed9am
WindSpeed3pm  \
0     13.4     22.9       0.6           44.0          20.0
24.0
1      7.4     25.1       0.0           44.0           4.0
22.0
2     12.9     25.7       0.0           46.0          19.0
26.0
3      9.2     28.0       0.0           24.0          11.0
9.0
4     17.5     32.3       1.0           41.0           7.0
20.0

   Humidity9am  Humidity3pm  Pressure9am  Pressure3pm  Temp9am
```

```
Temp3pm  \
0         71.0         22.0        1007.7        1007.1       16.9
21.8
1         44.0         25.0        1010.6        1007.8       17.2
24.3
2         38.0         30.0        1007.6        1008.7       21.0
23.2
3         45.0         16.0        1017.6        1012.8       18.1
26.5
4         82.0         33.0        1010.8        1006.0       17.8
29.7

   RainToday_No  RainToday_Yes
0             1              0
1             1              0
2             1              0
3             1              0
4             1              0
```

```
from sklearn.preprocessing import LabelEncoder
lb=LabelEncoder()
y_main=pd.DataFrame(lb.fit_transform(y),columns=['RainTomorrow'])
y_main.head()
```

```
   RainTomorrow
0             0
1             0
2             0
3             0
4             0
```

```
from sklearn.preprocessing import StandardScaler

names = x.columns

names
```

```
Index(['Date', 'Location', 'MinTemp', 'MaxTemp', 'Rainfall',
'WindGustSpeed',
       'WindSpeed9am', 'WindSpeed3pm', 'Humidity9am', 'Humidity3pm',
       'Pressure9am', 'Pressure3pm', 'Temp9am', 'Temp3pm',
'RainToday',
       'WindGustDir', 'WindDir9am', 'WindDir3pm'],
      dtype='object')
```

```
sc=StandardScaler()

x_scaled=pd.DataFrame(sc.fit_transform(x_p),columns=x_p.columns)
x_scaled.head()
```

7.2 Feature 2:

```
    MinTemp   MaxTemp  Rainfall  WindGustSpeed  WindSpeed9am
WindSpeed3pm  \
0  0.189949 -0.045963 -0.207770       0.305395      0.677617
0.614796
1 -0.749180  0.263481 -0.279002       0.305395     -1.130078
0.385479
2  0.111688  0.347875 -0.279002       0.457621      0.564636
0.844114
3 -0.467441  0.671385 -0.279002      -1.216867     -0.339212     -
1.105087
4  0.831687  1.276207 -0.160282       0.077056     -0.791135
0.156161

   Humidity9am  Humidity3pm  Pressure9am  Pressure3pm   Temp9am
Temp3pm  \
0     0.113867    -1.436005    -1.475400    -1.220931 -0.013524
0.016423
1    -1.312289    -1.289891    -1.045530    -1.116169  0.032829
0.380285
2    -1.629213    -1.046369    -1.490223    -0.981474  0.619960
0.220185
3    -1.259469    -1.728231    -0.007913    -0.367863  0.171886
0.700483
4     0.694893    -0.900255    -1.015884    -1.385559  0.125534
1.166225

   RainToday_No  RainToday_Yes
0      0.532962      -0.532962
1      0.532962      -0.532962
2      0.532962      -0.532962
3      0.532962      -0.532962
4      0.532962      -0.532962
```

```
from sklearn.model_selection import train_test_split

x_train,x_test,y_train,y_test =
train_test_split(x_scaled,y_main,test_size=0.2,random_state=0)
```

MODEL BULIDING

Training And Testing The Model

```
XGBoost=xgboost.XGBRFClassifier()
Rand_forest=sklearn.ensemble.RandomForestClassifier()
svm=sklearn.svm.SVC()
Dtree=sklearn.tree.DecisionTreeClassifier()
GBM=sklearn.ensemble.GradientBoostingClassifier()
log=sklearn.linear_model.LogisticRegression()

# Training the every model with Train data
model1=XGBoost.fit(x_train,y_train)
model2=Rand_forest.fit(x_train,y_train)
model3=svm.fit(x_train,y_train)
model4=Dtree.fit(x_train,y_train)
model5=GBM.fit(x_train,y_train)
model6=log.fit(x_train,y_train)
```

```
/usr/local/lib/python3.7/dist-packages/sklearn/preprocessing/
_label.py:98: DataConversionWarning: A column-vector y was passed when
a 1d array was expected. Please change the shape of y to
(n_samples, ), for example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.7/dist-packages/sklearn/preprocessing/_label.py
:133: DataConversionWarning: A column-vector y was passed when a 1d
array was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:3:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples,), for
example using ravel().
  This is separate from the ipykernel package so we can avoid doing
imports until
/usr/local/lib/python3.7/dist-packages/sklearn/utils/validation.py:993
: DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.7/dist-packages/sklearn/ensemble/_gb.py:494:
DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.7/dist-packages/sklearn/utils/validation.py:993
: DataConversionWarning: A column-vector y was passed when a 1d array
was expected. Please change the shape of y to (n_samples, ), for
```

CHAPTER – 8

TESTING

TESTING REPORT

Testing of an individual software component or module is termed asUnit Testing. It is typically done by the programmer and not by testers, as it requires detailed knowledge of the internal program design and code.

The Code was developed in 3 separate parts-

1. AI Model developed using Jupyter Notebook
2. Web Front end was developed using VS Code
3. Backend Database was developed using MongoDB

PROJECT NAME	Exploratory Analysis of RainFall Data in India for Agriculture
PROJECT TYPE	APPLIED DATA SCIENCE
DEVELOPER	Manikandan, Mariyappan, Meiyazhagan, Jeevanandhan
LANGUAGE	PYTHON,HTML,CSS,JAVA SCRIPT
TOTAL NUMBER OF TEST CASES	50
NUMBER OF TEST CASES EXCUTED	49
NUMBER OF TEST CASES PASSED	45
NUMBER OF TEST CASES FAILED	4-DUE TO TECHNICAL ISSUES

UNIT TESTING:

Unit testing is carried out screen-wise, each screen being identified as an object. Attention is diverted to individual modules, independently to one another to locate errors .Thishas enabled the detection of errorsin coding and logic .Thisisthe first level of testing.In this, codes are written such thatfromone module ,we can move on to the next module according to the choice weenter.

SYSTEM TESTING:

In this, the entire system was tested as a whole with all forms, code, modules and class modules .System testing is the stage ofimplementation, which is aimed at ensuring that the system works accurately and efficiently before live operation commences.

It is a series of different tests that verifies that allsystem elements have been properly integrated and perform allocated functions.

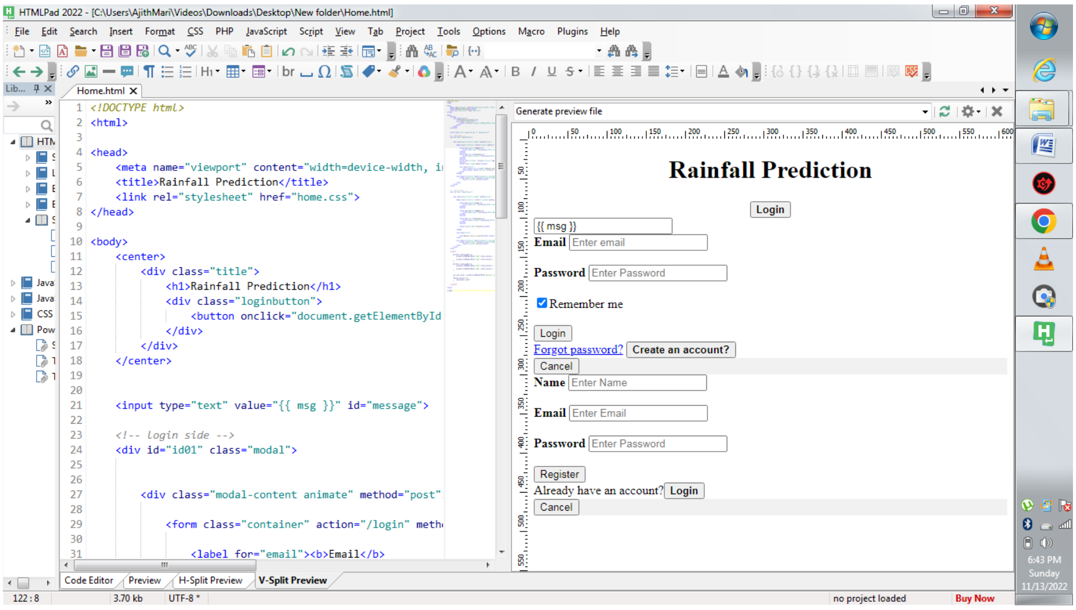
System testing makes logical assumptions that if all partsof the system are correct, the goal will be successfully achieved. Testing is the process of executing the program with the intent of finding errors.

Testing cannot show the absence of defects, itcan only show that software errors are present

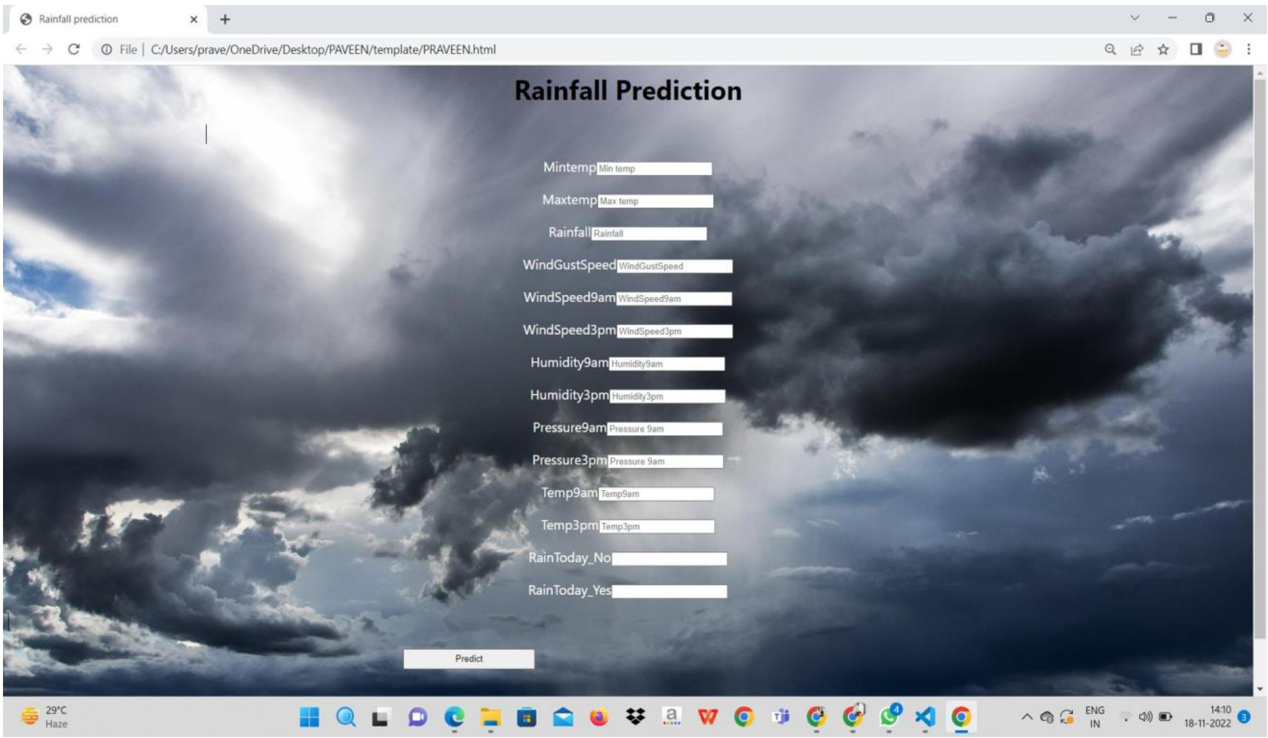
CHAPTER – 9

RESULTS

Model – 1



Model -2



CHAPTER – 10

ADVANTAGES & DISADVANTAGES OF EDA

Advantages of EDA:

- It gives us valuable insights into the data.
- It helps us with feature selection (i.e using PCA)
- Visualization is an effective way of detecting outliers.

Disadvantages of EDA:

- If not perform properly EDA can misguide a problem.
- EDA does not effective when we deal with high-dimensional data.

Applications of Exploratory Data Analysis:

- Let's analyze the applications of Exploratory Data Analysis with a use case of univariate analysis where we will seek the measurement of the central tendency of the data:
- Measurement of central tendency gives us an overview of the univariate variable. Central tendency is the measurement of Mean,Median, and Mode.
- Mean is the simple average where the median is the 50% percentileand Mode is the most frequently occurring value. Suppose we want the get the knowledge about the salary of a data scientist.
- Also, suppose we have carefully collected data of the data scientist with similar expertise and experience range.

Now if we want to get the average it is simply the total salary of all the datascientists of the sample divided by the number of data scientists in the sample or population. But if you think carefully the average salary is not a proper term because in the presence of some extreme values the result will be skewed. Suppose for maximum cases the salary is between 8-10 LPA and for one or two cases it is 32 LPA. Now adding all these the average will be skewed. Median is more suitable for such situations, it is more robust tooutliers.

CHAPTER – 11

CONCLUSION

In this article, we have discussed the various methodologies involved in exploratory data analysis, the applications, advantages, and disadvantagesit. We also walked through the sample codes to generate the plots in python using seaborn and Matplotlib libraries. EDA is the art part of data science literature which helps to get valuable insights and visualize the data.

12. SCOPE FOR FUTURE WORK:

Acceptance and use of weather information based farm advisories is likely to occur gradually. Farmers need time to try out new information, experience the benefits, and accept the results. Technology is changing rapidly whereas the mindset of the farmers changes slowly.

Experiencing accurate information and beneficial outcomes leads to trust building which certainly will encourage educated farmers to adopt the advisories. The following points may be taken into consideration while planning the future studies.

- Need to make these impact studies an integral part of the Agro advisory services of the country. Need to develop AAS service baseddecision support system for managing weather variability in reducingthe negative impacts on yield.
- Improving package of practices for major crops keeping in view the weather sensitive crop stages and weather sensitive farm operations for reducing cost of cultivation and improving yield and increasing netreturns. Need to improve the forecast quality during the sowing operations of kharif crops.
- Studies may be undertaken to quantify the value of medium range weather forecast in Nitrogen fertilizer management in arable crops. The N fertilizer advice may be tested through determining the uptake efficiency. The changes in N leaching, de-nitrification and crop N uptake due to the forecast quality needs to be assessed.
- Yield and gross profit changes may then be linked to N uptake. 99 Need to integrate Medium Range Weather Forecast with extended range forecast for better planning of the field operations particularly for sowing and mid-season corrections incase of drought
- The impact studies should be replicated in other crops of the region. Similar studies are also needed in other AAS units in India. The successful implementation of the scientific agro-meteorological forecasts need blending with local technologies like traditional methods so that farmers can readily adopt and be benefited from these scientific forecast..

- There is need to deliver district level weather based advisories through an automated dissemination system. In addition to the agriculture sector there is need to carry out similar studies in other weather sensitive sectors of economy as systematic and reliable data on the scope and dimensions of the relationship of weather and various user sectors is lacking.
- Better understanding of use and value of weather forecast may help substantially reduce the risks to life and property. For example, if there is knowledge about how many people and how much property is actually at risk to floods, one may be able to develop better strategies to reduce that undefined risk.
- In addition to the general lack of knowledge of the societal context ofweather events, there is also limited understanding of how decision makers could and actually use weather information. The significance of this study seems to call for a wide range of interests to support thesimilar efforts on other sectors such as aviation, power etc.
- The power firms like the Power Grid Corporation of India (PGCIL) require location specific quantitative forecast of Maximum/ MinimumTemperatures, Rainfall, Clouds, Wind Speed/direction four days in advance to run their Load Forecast models and the Power Distribution models.
- PGCIL estimates about 5-12 % saving on power equivalent to Rs 110crore per month through use of weather forecast of higher accuracy (>70%) Therefore to undertake work in such spheres, there is urgentneed to form a cohesive group of meteorologists (forecasters and

  researchers), users, and representatives from related fields (economics, policy makers, etc.).
- Although the entire meteorological community ought to be concerned with the outcome of that decision-making process, one should not try to do this in meteorological terms only. Public policy-makers must make difficult economic decisions that include issues of human safety, as well as purely economic factors.
- Decision making in weather sensitive sectors of economy must be made with knowledge of the economic impacts of weather forecasts, rather than without that quantitative information.

13. REFERENCES:

- Das Gupta, M. and Rizvi, S. R. H. (2001): An Overview of Meteorological Data Decoding and Pre-processing Operational at NCMRWF, NCMRWF Tech. Rep. No: NMRF/TR/1/2001, 33pp.

- Kumar A, Parvinder Maini, Rathore,L.S. and S.V.Singh, 2000: An operational medium range local weather forecasting system developed in India Int. J. Climatol., (a journal of Royal MeteorologicalSociety), 20: 73-87.

- Maini Parvinder, 2006: Development of Statistical-Dynamical Models for Location Specific Weather Forecast. (PhD Thesis), Andhra University, 242pp.

Annexure:

Annexure-I

Following are the verification scores that have been used for verifying the rainfall and temperature forecasts disseminated to the AAS units on a bi-weekly basis

(a) Measures of obtaining skill of Yes/No rainfall

In the following 2×2 contingency table, if Y stands for occurrence of rain and N stands for non-occurrence then

Forecast (Rain)	Observed (Rain)	
	Yes	No
Yes	*YY*	*YN*
No	*NY*	*NN*

The total number of cases (*M*) is given by:

$M = YY+YN+NY+NN$

i. Ratio Score

Ratio Score (*RS*), also known as the Hit Rate or Percentage Correct, measures the proportion of correct forecasts. The *RS* varies from 0 to 100 with 100 indicating perfect forecasts.

$$RS = \frac{correct\ forecasts}{total\ forecasts} = \frac{(YY + NN)}{M} \times 100$$

ii. Hanssen and Kuipers' Score

Hanssen and Kuipers' Score (*HKS*) (Woodcock, 1976, 1981) is the ratio of economic saving over climatology due to the forecast to that of a set of perfect forecasts. In *HKS* the reference hit rate in the denominator is for random forecasts that are constrained to be unbiased.

$$HKS = \frac{correct\ forecast - (correct\ forecast)_{\text{random}}}{M - (correct\ forecast)_{\text{random, unbiased}}}$$

$$HKS = \frac{(YY * NN - YN * NY)}{(YY + NY)(YN + NN)}$$

That is, the imagined random reference forecasts in the denominator have a marginal distribution that is equal to the (sample) climatology (Wilks, 1995).The value of *HKS* varies from –1 to +1. If all forecast are wrong (i.e. $YY = NN = 0$) then it is –1, and if all forecast are perfect (i.e. $YN = NY = 0$) then it is +1, and random forecasts receive a score of 0.

(b) Criteria for obtaining usability of Quantitative Precipitation (QP)

Error Structure for verification of Quantitative Precipitation

	Observed rainfall ≤10mm	Observed rainfall > 10mm
Correct	*Diff ≤ 0.2 mm*	*Diff ≤ 2% of obs*
Usable	*0.2 mm< Diff ≤ 2.0mm*	*2% of obs < Diff ≤ 20% of obs*
Unusable	*Diff >2.0 mm*	*Diff > 20% of obs*

where Diff stands for Absolute difference of observed and forecasted in mm and obs stands for observed rainfall in mm

(c) Measures of obtaining skill of temperature

Correlation Coefficient (r) and Root Mean Square Error (*RMSE*) are calculated for obtaining the skill of the model in forecasting maximum and minimum temperatures.

(i) Correlation coefficient can be defined as

$$r(f_i, o_i) = \frac{\sum (f_i - \bar{f})(o_i - \bar{o})}{\left[\sum (f_i - \bar{f})^2 \sum (o_i - \bar{o})^2\right]^{1/2}}$$

(ii) Root Mean Square Error (*RMSE*)**:** The *RMSE* is the square root of Mean Square Error (*MSE*) which measures the degree of correspondence between the forecasts and observations in terms of the average squared difference between f_i and o_i.

Where $$RMSE = \left(\frac{1}{n}\sum (f_i - o_i)^2\right)^{1/2}$$

f_i = *forecast value*

$\bar{f}$ = *mean forecast value*

o_i = *observed value*

$\bar{o}$ = *mean observed alue*

n = *total no : of observations / forecast*

(d) Criteria for obtaining usability of Temperature forecast

Error Structure for verification of Temperature Forecast

Correct	$Diff \leq 1^0C$
Usable	$1^0C < Diff \leq 2^0C$
Unusable	$Diff > 2^0C$

where Diff stands for Absolute difference of observed and forecasted temperatures in ^0C

GITHUB LINK : https://github.com/IBM-EPBL/IBM-Project-5771-1658814926