A PROJECT ON

"RAINFALL PREDICTION IN AUSTRALIA"

SUBMITTED IN
PARTIAL FULFILLMENT OF THE REQUIREMENT
FOR THE COURSE OF
DIPLOMA IN BIG DATA ANALYSIS

SUNBEAM INSTITUTE OF INFORMATION TECHNOLOGY, PUNE

Submitted By:

Prajwal Charde (83795)

Ajay Biradar (83994)

Mr.Nitin Kudale
Centre Coordinator

Mrs.Manisha Hingne
Course Coordinator

CERTIFICATE

This is to certify that the project work under the title “RAINFALL PREDICTION IN AUSTRALIA ” is done by Dattatray Hake & Udit Deshmukh in partial fulfillment of the requirement for award of Diploma in Big Data Analysis Course.

Mr. Aniket P
Project Guide

Mrs. Manisha Hingne
Course Coordinator

Date:

ACKNOWLEDGEMENT

A project usually falls short of its expectation unless aided and guided by the right persons at the right time. We avail this opportunity to express our deep sense of gratitude towards Mr. Nitin Kudale (Center Coordinator, SIIT, Pune) and Mrs. Manisha Hingne(Course Coordinator, SIIT ,Pune) and Project Guide Mr. Aniket P.

We are deeply indebted and grateful to them for their guidance, encouragementand deep concern for our project. Without their critical evaluation and suggestions at every stage of the project, this project could never have reached its present form.

Last but not the least we thank the entire faculty and the staff members of SunbeamInstitute of Information Technology, Pune for their support.

Prajwal Charde
DBDA March 2024 Batch,
SIIT Pune

Ajay Biradar
DBDA March 2024
Batch,SIIT Pune

TABLE OF CONTENTS

1. Introduction

1.1. Introduction And Objectives

1.2. Why this problem needs To be Solved?

1.3. Dataset Information

2. Problem Definition and Algorithm

2.1 Problem Definition

2.2 Algorithm Definition

3. Experimental Evaluation

3.1 Methodology/Model

3.2 Exploratory Data Analysis

4. Results And Discussion

5. GUI

6. GitHub link

7. Conclusion

1. Introduction

1.1 Introduction And Objectives:

The rainfall prediction project aims to develop a system that can accurately forecast rainfall amounts in a specific area or region. The primary objective of this project is to design and implement a machine learning model that can predict the amount of rainfall in advance, using historical weather data and other relevant factors. This project is crucial for various applications such as agriculture, water resource management, transportation, and disaster preparedness.

The project's objectives are to:

-Develop a machine learning model that can accurately predict rainfall amounts.
-Analyze the performance of the model using various evaluation metrics.
-Identify the most important factors that influence rainfall prediction.
-Provide insights and recommendations for improving rainfall prediction systems.

The project's significance lies in its potential to improve the accuracy of rainfall predictions, which can have a significant impact on various aspects of society, including agriculture, water resource management

1.2 Why this problem needs To be Solved?

Rainfall prediction is a complex problem that affects various aspects of our lives, including agriculture, transportation, and urban planning. Accurate rainfall prediction is crucial for:

Agricultural Planning: Farmers need to plan their crop cycles, reducing the risk of crop failure and improving yields.

Water Resource Management: Rainfall prediction helps manage water resources more efficiently, ensuring that water is allocated effectively for various uses.

1.3 Dataset Information.

Description of every column :-

Date: The date of observation

Location: The common name of the location of the weather station

MinTemp: The minimum temperature in degrees Celsius

MaxTemp: The maximum temperature in degrees Celsius

Rainfall: The amount of rainfall recorded for the day in mm

Evaporation: The so-called Class A pan evaporation (mm) in the 24 hours to 9am

Sunshine: The number of hours of bright sunshine in the day.

WindGustDir: The direction of the strongest wind gust in the 24 hours to midnight

WindGustSpeed: The speed (km/h) of the strongest wind gust in the 24 hours midnight

WindDir9am: Direction of the wind at 9am

WindDir3pm: Direction of the wind at 3pm

WindSpeed9am: Wind speed (km/hr) averaged over 10 minutes prior to 9am

WindSpeed3pm: Wind speed (km/hr) averaged over 10 minutes prior to 3pm

Humidity9am: Humidity (percent) at 9am

Humidity3pm: Humidity (percent) at 3pm

Pressure9am: Atmospheric pressure (hpa) reduced to mean sea level at 9am

Pressure3pm: Atmospheric pressure (hpa) reduced to mean sea level at 3pm

Cloud9am: Fraction of sky obscured by cloud at 9am. This is measured in "oktas", which are a unit of eigths. It records how

Cloud3pm: Fraction of sky obscured by cloud (in "oktas": eighths) at 3pm. See for a

description of the values

Temp9am: Temperature (degrees C) at 9am

Temp3pm: Temperature (degrees C) at 3pm

RainToday: Boolean 1 if precipitation (mm) in the 24 hours to 9am exceeds 1mm, otherwise 0

RainTomorrow: The amount of next day rain in mm. Used to create response variable RainTomorrow. A kind of measure of the "risk".

Train.csv

It has seven columns.

Following are the columns that has been Dropped:

1) RainTomorrow
2) WindDir9am
3) WindSpeed9am
4) Humidity9am
5) Pressure9am
6) Cloud9am
7) Temp9am

Test.csv: is same as train.csv except it does not have 'RainTomorrow' Column.

1. Problem Definition and Algorithm:

1.1 Problem Definition

Rainfall prediction involves predicting the amount of rainfall at a specific location and time. The goal is to develop a model that can accurately predict rainfall patterns, enabling decision-makers to take proactive measures to mitigate the impacts of heavy rainfall events, optimize water resource management, and improve agricultural product ivity.

1.2 Algorithm Definition

Logistic regression :It is a widely used statistical model for binary classification tasks, where the goal is to predict the probability of an outcome that can take one of two values, typically 0 or 1. Unlike linear regression, which predicts continuous outcomes, logistic regression uses a logistic function (also known as the sigmoid function) to map the predicted values to a probability between 0 and 1.

Random forest: is a Supervised Machine Learning Algorithm that is used widely in Classification and Regression problems. It builds decision trees on different samples and takes their majority vote for classification and average in case of regression.

One of the most important features of the Random Forest Algorithm is that it can handle the data set containing continuous variables as in the case of regression and categorical variables as in the case of classification. It performs better results for classification problems.

k-Nearest Neighbors (k-NN) : Algorithm is a simple, instance-based learning method used for classification and regression tasks. It works by finding the "k" closest data points (neighbors) to a given input point and then predicting the label based on the majority class (in classification) or the average value (in regression) of these neighbors.

The distance between points is usually measured using metrics like Euclidean distance. k-NN is intuitive and easy to implement but can be computationally expensive, especially with large datasets, as it requires storing all data points and calculating distances for each prediction.

Decision Tree: algorithm belongs to the family of supervised learning algorithms. Unlike other supervised learning algorithms, the decision tree algorithm can be used for solving regression and classification problems too.

The goal of using a Decision Tree is to create a training model that can use to predict the class or value of the target variable by learning simple decision rules inferred from prior data(training data).In Decision Trees, for predicting a class label for a record we start from the root of the tree. We compare the valuesof the root attribute with the record's attribute. On the basis of comparison, we follow the branch corresponding to that value and jump to the next node.

Naive Bayes: It is a simple, yet powerful, supervised machine learning algorithm used primarily for classification tasks. It is based on Bayes' Theorem, which describes the probability of an event occurring given prior knowledge of conditions related to the event. The "naive" part refers to the assumption that all features are independent of each other, which rarely holds true in real life, but the model still performs well in practice. Naive Bayes is particularly useful for text classification, such as spam detection, due to its efficiency and ability to handle large datasets.

Support Vector Machine (SVM): It is a supervised learning algorithm used mainly for classification. It works by finding the optimal hyperplane that best separates data points into different classes. This hyperplane is chosen to maximize the margin, the distance between it and the nearest data points from each class, called support vectors. SVM can handle non-linear data by applying kernel functions, which transform the data into a higher-dimensional space where separation is easier. It's effective for high-dimensional data but may require careful parameter tuning for optimal performance.

CatBoost: It is a high-performance gradient boosting algorithm developed by Yandex, designed for handling categorical data more effectively than many other machine learning models. It builds an ensemble of decision trees, combining the predictions of many models to improve accuracy. CatBoost stands out for its ability to automatically handle categorical features without requiring extensive preprocessing, such as one-hot encoding. It also reduces overfitting through techniques like ordered boosting and has built-in support for handling missing values. CatBoost is known for its ease of use, fast training, and strong performance in various machine learning tasks, including ranking, classification, and regression.

XGBoost: or extreme gradient boosting is one of the well-known gradient boosting techniques (ensemble) having enhanced performance and speed in tree-based (sequential decision trees) machine learning algorithms. XGBoost was created by Tianqi Chen and initially maintained by the Distributed (Deep) Machine Learning Community (DMLC) group. It is the most common algorithm used for applied machine learning in competitions and has gained popularity through winning solutions in structured and tabular data. It is open-source software. Earlier only python and R packages were built for XGBoostbut now it has extended to Java, Scala, Julia and other languages as well.

AdaBoost: It short for Adaptive Boosting, is an ensemble learning algorithm that combines multiple weak learners, typically decision trees with a single split (stumps), to create a strong classifier. The key idea behind AdaBoost is to focus on the data points that are hardest to classify correctly. During each iteration, it assigns higher weights to misclassified data points so that the next learner focuses more on those challenging cases. The final model is a weighted sum of the weak learners, where more accurate learners have greater influence. AdaBoost is effective in improving the accuracy of weak models but can be sensitive to noisy data and outliers.

2. Experimental Evaluation:

2.1 Methodology:

Rainfall prediction modeling involves a combination of computer models, observation and knowledge of trends and patterns. Using these methods, reasonably accurate forecasts can be made up. Several recent research studies have developed rainfall prediction using different weather and climate forecasting methods.

Loading in raw data and Importing Libraries

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns


df = pd.read_csv('./weatherAUS.csv')

df=df.set_index("Date")

df.head()

df.describe
```

Preprocessing:

Recoginzing Datatypes of each column: df.info()

We have to Deal with two types of values :

1) **Numerical [Fill with mean values]**
2) **Categorical Values [Fill with mode values]**

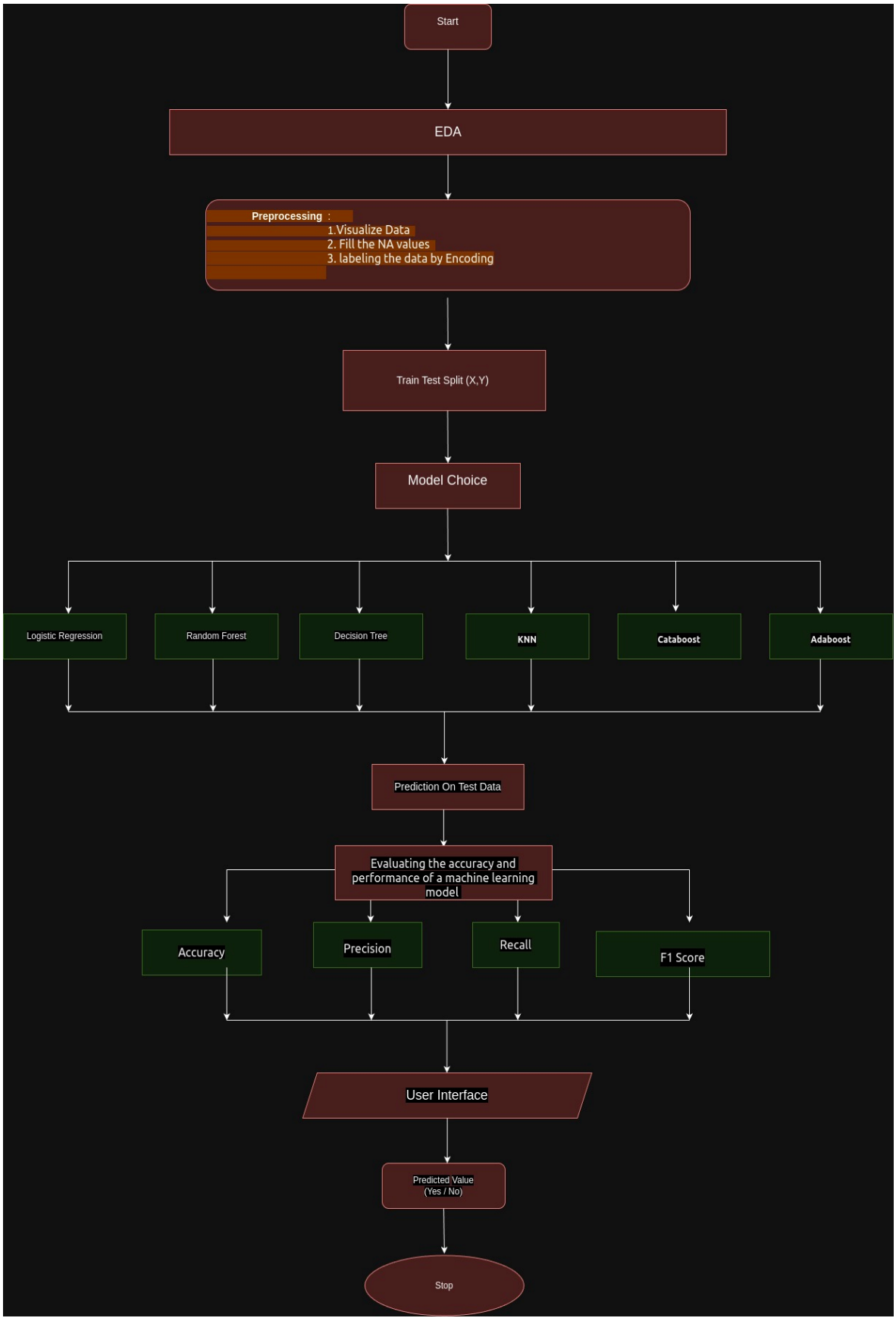
Numerical [Fill with mean values]

```
df['Evaporation'].fillna(df['Evaporation'].mean(),inplace=True)
df['Sunshine'].fillna(df['Sunshine'].mean(),inplace=True)
df['MinTemp'].fillna(df['MinTemp'].mean(),inplace=True)
df['MaxTemp'].fillna(df['MaxTemp'].mean(),inplace=True)
df['Rainfall'].fillna(df['Rainfall'].mean(),inplace=True)
df['WindGustSpeed'].fillna(df['WindGustSpeed'].mean(),inplace=True)
df['WindSpeed9am'].fillna(df['WindSpeed9am'].mean(),inplace=True)
df['WindSpeed3pm'].fillna(df['WindSpeed3pm'].mean(),inplace=True)
df['Humidity9am'].fillna(df['Humidity9am'].mean(),inplace=True)
df['Humidity3pm'].fillna(df['Humidity3pm'].mean(),inplace=True)
df['Pressure9am'].fillna(df['Pressure9am'].mean(),inplace=True)
df['Pressure3pm'].fillna(df['Pressure3pm'].mean(),inplace=True)
df['Cloud9am'].fillna(df['Cloud9am'].mean(),inplace=True)
df['Cloud3pm'].fillna(df['Cloud3pm'].mean(),inplace=True)
df['Temp9am'].fillna(df['Temp9am'].mean(),inplace=True)
df['Temp3pm'].fillna(df['Temp3pm'].mean(),inplace=True)
```

Categorical Values [Fill with mode values]

```
RainToday_mode = df['RainToday'].mode()[0]
df['RainToday'].fillna(RainToday_mode, inplace=True)
RainTomorrow_mode = df['RainTomorrow'].mode()[0]
df['RainTomorrow'].fillna(RainTomorrow_mode, inplace=True)
WindGustDir_mode = df['WindGustDir'].mode()[0]
df['WindGustDir'].fillna(WindGustDir_mode, inplace=True)
WindDir9am_mode = df['WindDir9am'].mode()[0]
df['WindDir9am'].fillna(WindDir9am_mode, inplace=True)
WindDir3pm_mode = df['WindDir3pm'].mode()[0]
df['WindDir3pm'].fillna(WindDir3pm_mode, inplace=True)
```

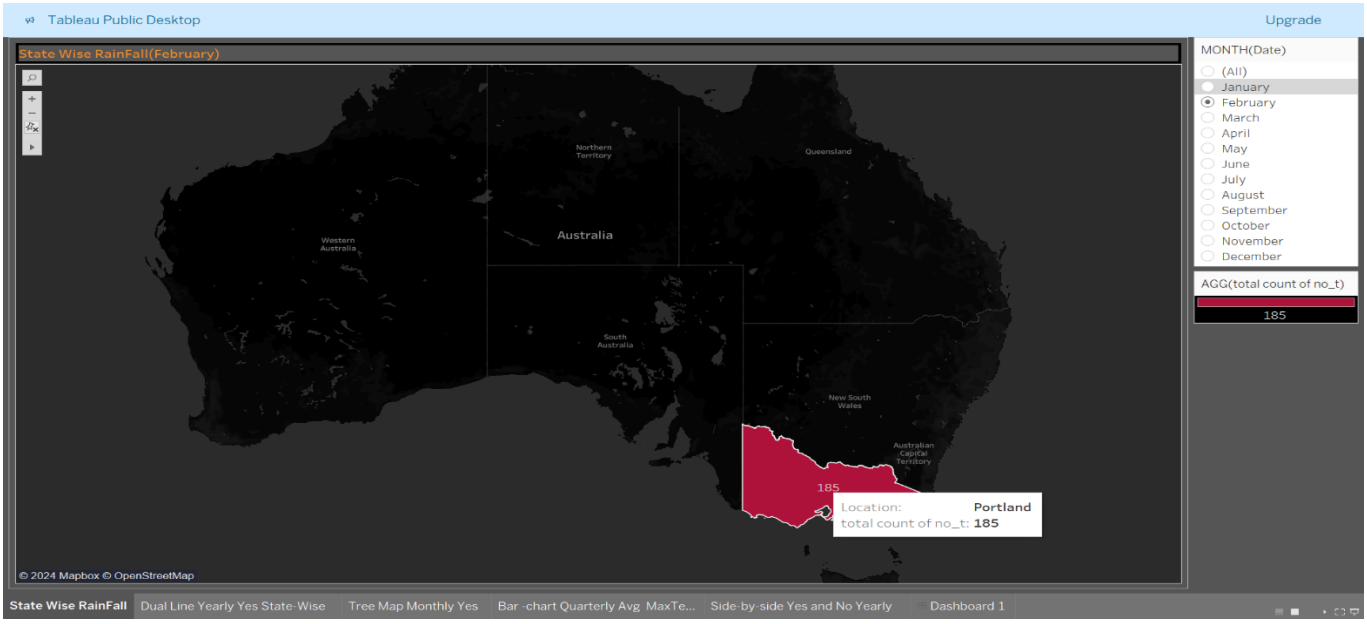
Flow Diagram :



2.2 Exploratory Data Analysis

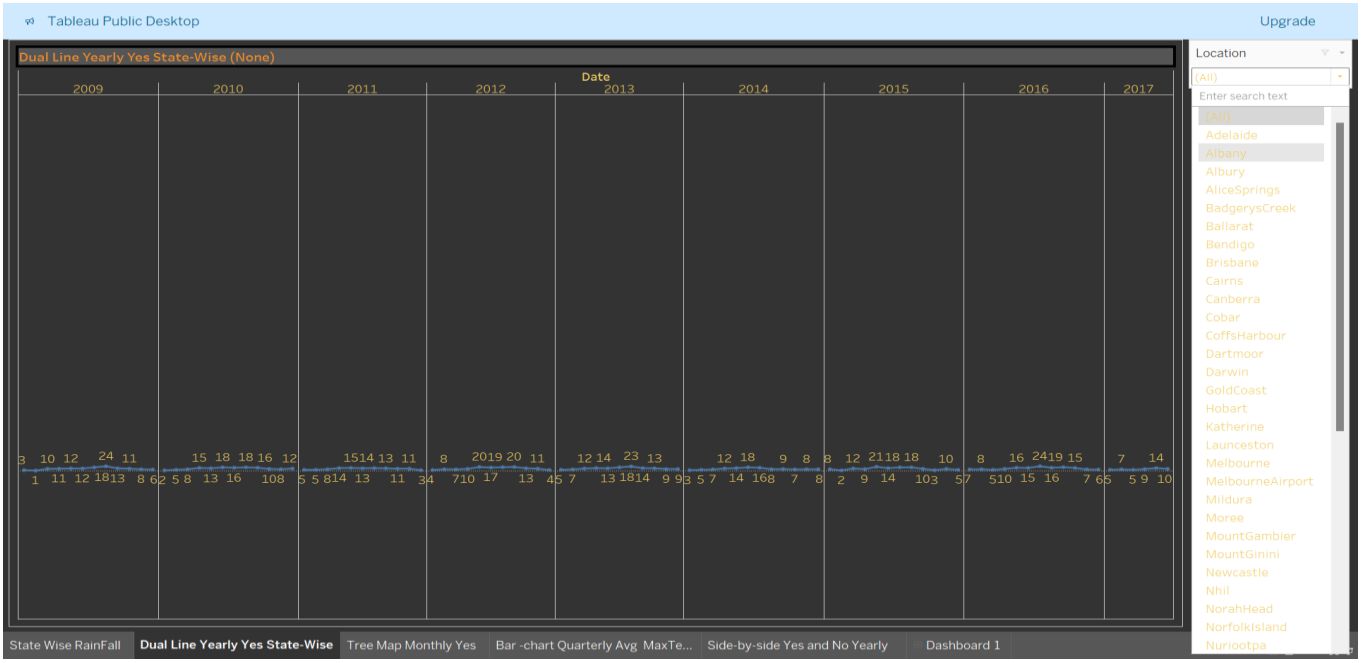
1) Geo graphical graph

To check the rain fall of no count location wise monthly filter.

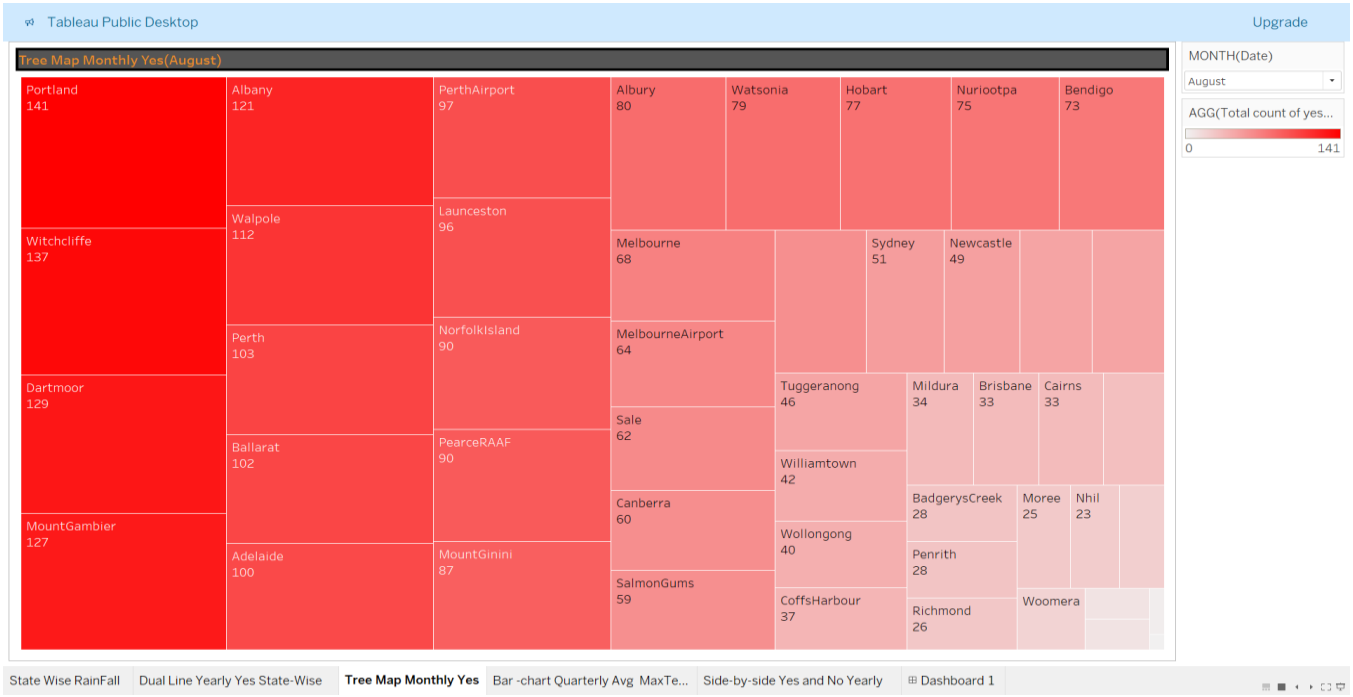


2) Dual Line Graph

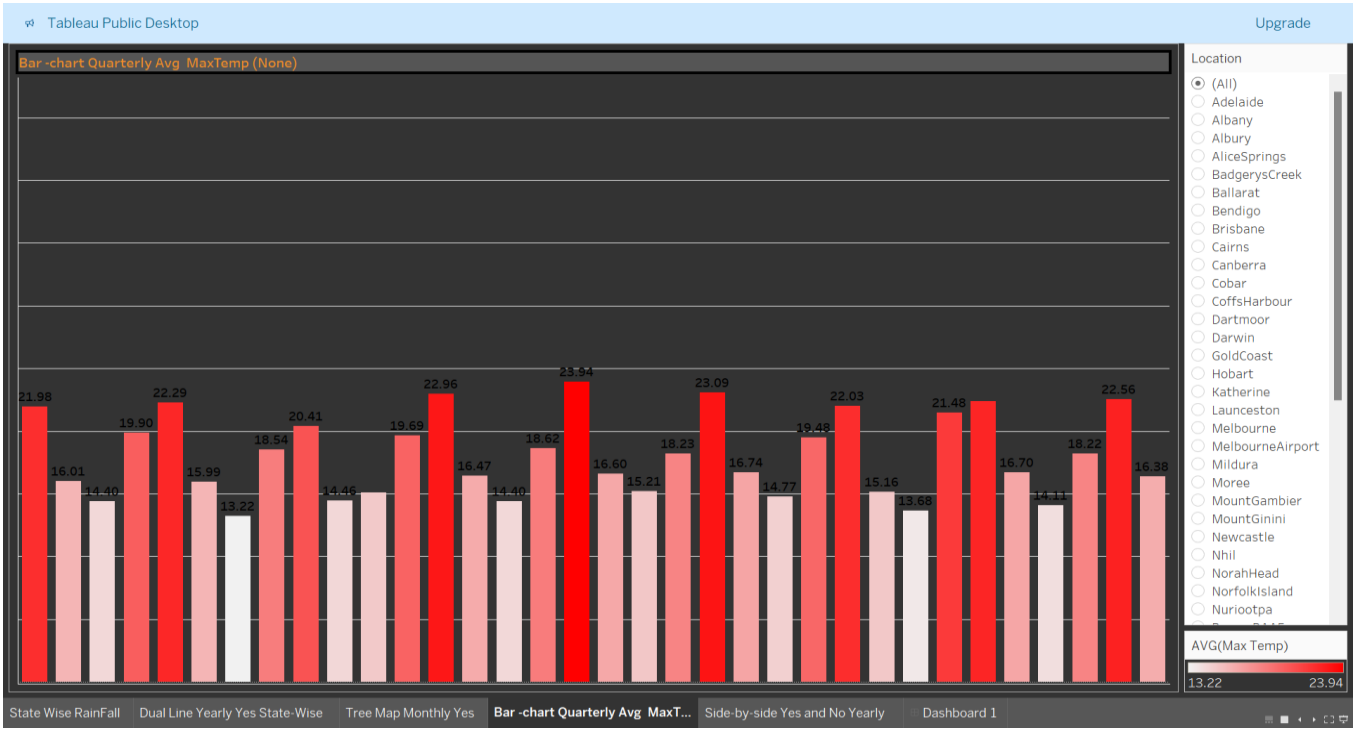
To check the count of Yes yearwise by using filter location.



3)Tree-Map Graph
To check the Total Count of yes Statewise by Filtering Monthly

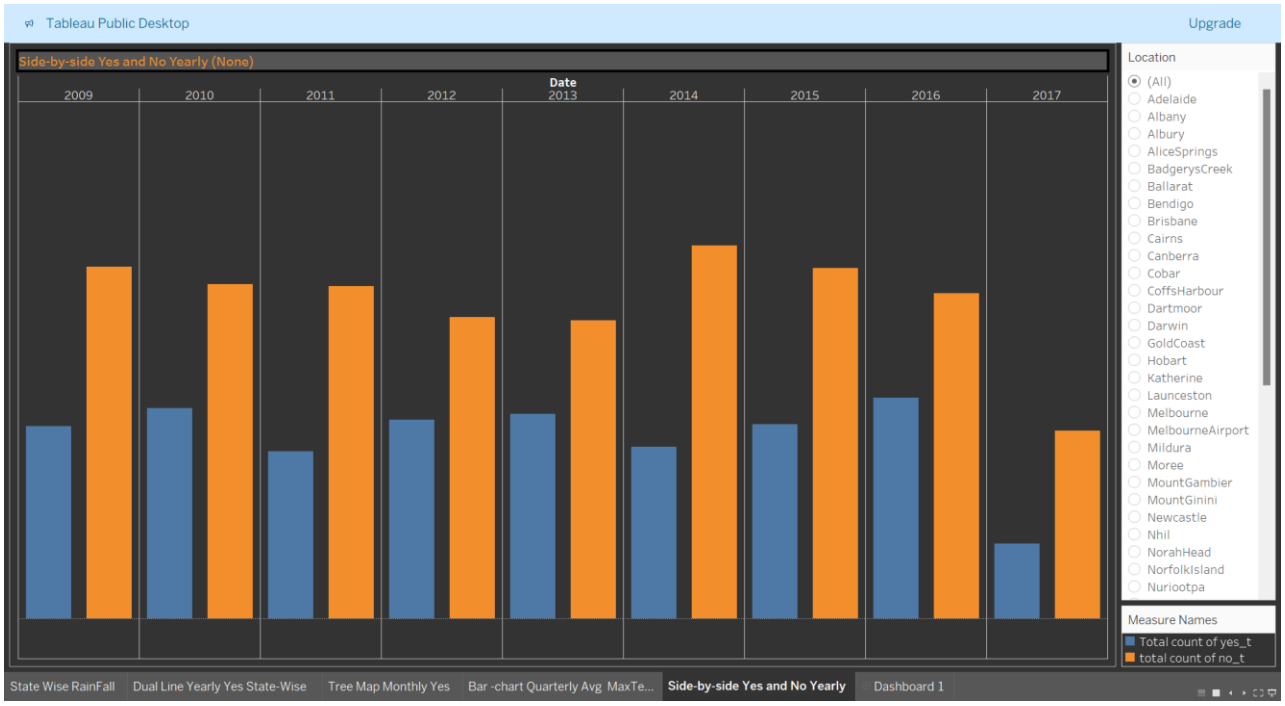


4)Bar-Chart Graph
To check the Avg-Max Temp of quarterly By filtering Location



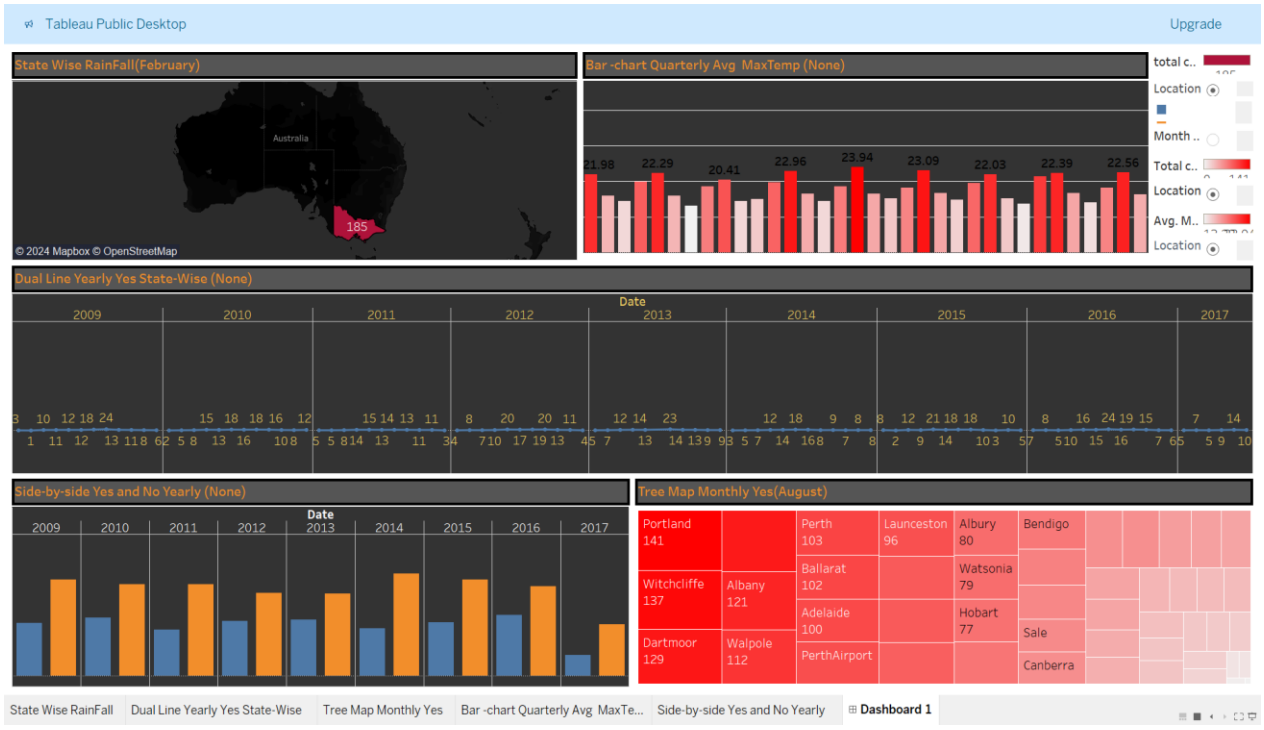
5)Side By side Chart

To compare between Count of Yes/No rainfall yearwise By using Location as filter



6) Dynamic Dashboard

Tree map as input source sheet and other Sheet are dynamically handle as we change in tree map other graphs change.



Results and discussion:

Linear regression, Decision Tree , random forest, decisiontree and gradient boosting machine algorithm were used to predict the Rainfall in Australia. Among the given algorithms CataBoost Machine algorithm was the best performing one as it provided the highest F2score of 62.97

```
def train_model_catboost():
    from catboost import CatBoostClassifier
    model=CatBoostClassifier()
    model.fit(x_train,y_train)
    return model,"catboost"

models=[train_model_catboost()]

def evaluate_model(model,model_name):
        from sklearn.metrics
    import   accuracy_score,precision_score,recall_score,f1_score
    y_true=y_test
    y_predict=model.predict(x_test)
    a=accuracy_score(y_true,y_predict)*100
    p=precision_score(y_true,y_predict)*100
    r=recall_score(y_true,y_predict)*100
    f1=f1_score(y_true,y_predict)*100
    return model_name,a,p,r,f1

rows=[]
for (model,model_name) in models:
    rows.append(evaluate_model(model,model_name))
```

Algorithm	accuracy_score	precision_score	recall_score	f1_score
Catboost	86.014024	75.039029	54.258121	62.9786

5 GUI:

GUI is made using Flask framework. **Flask** is a micro web framework written in Python. It is classified as a microframework because it does not require particular tools or libraries. It has no database abstraction layer, form validation, or any other components where pre-existing third-party libraries provide common functions. However, Flask supports extensions that can add application features as if they were implemented in Flask itself. Extensions exist for object-relational mappers, form validation, upload handling, various open authentication technologies and several common framework related tools.

6.GitHubLink:

https://github.com/B1-Prajwal-83795/CDAC-Project.git

7 Conclusion:

- On year 2016 Rainfall is fallen more as compared the range of years 2007 to 2017.

- It is mostly seen that in Portland Rain is fallen more by visualizing as compared to other location by looking at tree map.

- And one important think that had seen that on month of July more rainfall takes place.

- By using flak and Ui client can also get to know whether rain will takes places or not by giving certain parameters.