RAINFALL PREDICTION

# A MINI PROJECT REPORT

***Submitted by***

# LEENA JESSICA U -921321104109

**MEENAMBAL S V-921321104126**

**MALATHEESWARI E-921321104116**

**JANANI R-921321104074**

**HARSHINI M-921321104067**

**JAYATHENDRAL R K-921321104075**

**BACHELOR OF ENGINEERING**

***in***

# COMPUTER SCIENCE AND ENGINEERING

**PSNA COLLEGE OF ENGINEERING AND TECHNOLOGY,**

## (An Autonomous Institution Affiliated to Anna University, Chennai)

**DINDIGUL - 624622**

APRIL 2023

**PSNA COLLEGE OF ENGINEERING AND TECHNOLOGY,**

## (An Autonomous Institution Affiliated to Anna University, Chennai)

**DINDIGUL - 624622**

**BONAFIDE CERTIFICATE**

Certified that this project report “**RAINFALL DETECTION**” is the bonafide work of **“LEENA JESSICA U (921321104109), MEENAMBAL S V (921321104126), MALATHEESWARI E (921321104116), JANANI R (921321104074), HARSHINI M (921321104067) , JAYATHENDRAL R K (921321104075)”** who

carried out the project work under my supervision.

### SIGNATURE

**Dharshana Deepthi,ME., Supervisor,**

**Department of CSE,**

**PSNA College of Engg&Tech., Dindigul-624622.**

**ABSTRACT**

The aim of this project is to make a machine learn about the prediction of rainfall. Rainfall prediction is one of the challenging tasks in weather forecasting process. Rainfall also depends on geographic locations hence is an arduous task to predict. Machine learning techniques can predict rainfall by extracting hidden patterns from historical weather data.The proposed framework uses four widely used supervised machine learning techniques, i.e., decision tree, Naïve Bayes, K-nearest neighbors , and support vector machines. For agriculture, Rainfall is important but during these days’ rainfall prediction has become a major challenging problem. Good prediction of rainfall provides knowledge and know in advance to take precautions and have better strategy about theirs crops.  Because of its air is getting warmer and level of ocean is rising, leads to flood and cultivated field is changing into drought.  To predict Rainfall is one of the best techniques to know about rainfall and climate. The main aim of this study revolves around providing correct climate description to the clients from various perspectives like agriculture, researchers, generation of power etc. to grasp the need of transformation in climate and its parameters like temperature, humidity, precipitation, wind speed that eventually directs to projection of rainfall. Rainfall also depends on geographic locations hence is an arduous task to predict. Machine Learning is the evolving subset of an AI, that helps in predicting the rainfall. This study is to develop the rainfall prediction system and predict the rainfall with better accuracy with the use of Machine Learning classification algorithms.

# CONTENTS

## TITLE PAGE NO.

## SCOPE OF THE PROJECT 03

## THE EXISTING FEATURES OF A SYSTEM 04

## PROPOSAL OF THE SYSTEM 05

## SYSTEM ARCHITECTURE 07

## DESCRIPTION OF THE MODEL 08

## SYSTEM TESTING AND IMPLEMENTATION 09

## SOURCE CODE 15

## SCREENSHOTS 17

## CONCLUSION 20

## REFERENCES 21

**SCOPE OF THE PROJECT**

There is a general and increasing interest on weather information, since every day we habitually give an ear to weather forecast news for local and large-scale longterm or short-term weather predictions. Leading weather research institutions and companies have been developing weather prediction systems capable of detecting, predicting and forecasting weather phenomena and hazards by utilizing state-of-thescience technologies. Thus weather prediction utilization fields and prediction accuracy increases monotonically by the time.

The scope of the project also includes the development of a training dataset for the machine learning algorithm. The dataset will consist of the data about rainfall . The dataset will be used to train and validate the machine learning algorithm, ensuring its accuracy and reliability.

Overall, the scope of the rainfall detection ML project is to develop a comprehensive and user-friendly approach for detecting and predicting rainfall that can contribute to the existing features of a system sustainability of meterological and weather analysis.

# THE EXISTING FEATURES OF A SYSTEM

The existing features of a system for rainfall prediction using AI and ML techniques could include the following:

1.Image processing:Generally for weather forecasting and rain forecasting we use the satellite images and satellite techniques.Image processing and analysis can be defined as the “act of examining images for the purpose of identifying objects and judging their significance”. This is a physical process used to convert an image signal into a physical image.

2.Weather datasets:The climatic features included in the weather datasets are first described.The rationale for developing a prediction model that forecasts 8-hours of rainfall lies not only in the impact of precipitation on washing and deposition of different air pollutants but in providing prompt estimates to support decision making to diminish eventualities in several human-related activities.

3.Dataset pre-processing:As with any Machine Learning approach, a processing procedure is required to prepare raw data for use in model training and testing processes.The pre-processing process of the weather dataset results in five datasets with a 43-dimensional feature vector structure. Each dataset is integrated by all previously described weather measurements, the geographical coordinates, and by common weather codes across the five datasets such as Rain ID 500, Drizzle ID 300, and Mist ID 701.

4.Rainfall prediction approach:This study seeks to investigate the suitability of three LSTM-Networks architectures in the task of predicting 8-hours of rainfall volume using time-series data from five major UK cities. As mentioned earlier, the performance of algorithms is directly affected by a plethora of design decisions. These design decisions include the choice of the values of the different parameters and hyperparameters involved in each algorithm.

5. Accurate and reliability: Indeed, the forecast services are around 80% accurate for the first three days and tailoff to just under 70% by day nine, and for all days, the forecasts are better than persistence. You might be surprised, though, that the persistence forecasts are 60-70% accurate as well.

Overall, these features can help to improve the accuracy and reliability of the system for rainfall prediction and management.

# PROPOSAL OF THE SYSTEM

Introduction:

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 analyze 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. The objective of this research paper is to predict the Rainfall of a location based on input parameters that will be provided by the user. The parameters include date, location, maximum temperature, minimum temperature, humidity, wind direction, evaporation etc.

ADVANTAGES:

1. Water resources can be managed efficiently by using rainfall prediction system. 2. Regions can be evacuated if flood are expected.

3. It helps in taking appropriate measures to efficiently manage water resources, crop productivity and no wastage of any resources.

Metrology:

Precipitation, in short periods of time, is a phenomenon associated with high levels of uncertainty and variability. Given its nature, traditional forecasting techniques are expensive and computationally demanding. The model were developed with time series from ten agriculturally relevant regions in Brazil, these places are the ones with the longest available weather time series and and more deficient in accurate climate predictions, it was available 60 years of daily mean air temperature and accumulated precipitation which were used to estimate the potential evapotranspiration and water balance; these were the variables used as inputs for the ANNs models.

The models have peak performance in well defined seasons, but looses its accuracy in transitional seasons and places under influence of macro-climatic and mesoclimatic effects, which indicates that this technique can be used to indicate the eminence of rainfall with some limitations.

Conclusion:

The overall aim is to define various ML techniques that are useful in predicting rainfall. The goal of this research is to design accurate and efficient model by applying lesser number of attributes and tests. Taking into consideration the limitations of this study, there is a need to build more complex and combination of models to get higher accuracy for rainfall prediction system. Study can also be formulated using greater articulate monitoring for particular area and create this kind of model for enormous dataset so that calculation rate can be increased with better precision and with more accuracy.

# SYSTEM ARCHITECTURE

Dataset Upload

Processing and analysis of dataset

Result

Accuracy of rainfall Dataset

Dataset View

**DESCRIPTION OF THE MODEL**

The model for the rainfall prediction ML project can be based on the combination of dataset analysis and machine learning algorithms. Here's a brief description of the model:

Data analysis: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. We can handle the missing values either by deleting irrelevant column or row.

Data processing:Feature selection is also the part of pre-processing in which we select only those features which contributes to our rainfall prediction model thus helps in reducing training time and increases accuracy of the model. Feature scaling is the final stage in pre-processing in independent variables are brought into specific range so that no any variable dominates the other variable.

Machine learning algorithm :This research proposes a novel real-time rainfall prediction system for smart cities using a machine learning fusion technique. The proposed framework uses four widely used supervised machine learning techniques, i.e., decision tree, Naïve Bayes, K-nearest neighbors, and support vector machines.

Output :The result of the study revealed that the Extreme Gradient Boosting machine learning algorithm performed better than others.

Overall , the model for the rainfall prediction ML project is designed to be scalable,

accurate, and efficient, using a combination of data analysis and processing techniques and machine

learning algorithms to predict the rainfall.

|  |  |
| --- | --- |
|  |  |

**SYSTEM TESTING AND IMPLEMENTATION**

**SYSTEM TESTING**

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, Sub-assemblies, assemblies and\or a finished product It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of test type addresses a specific testing requirement.

**TESTING STEPS**

* Unit Testing
* Integration Testing
* Functional Testing
* System Testing
* White Box Testing
* Black Box Testing
* Output Testing
* User Acceptance Testing

**UNIT TESTING**

Unit testing is a type of software testing that focuses on individual units or components of a software system. The purpose of unit testing is to validate that each unit of the software works as intended and meets the requirements. Unit testing is typically performed by developers, and it is performed early in the development process before the code is integrated and tested as a whole system.

Unit tests are automated and are run each time the code is changed to ensure that new code does not break existing functionality. Unit tests are designed to validate the smallest possible unit of code, such as a function or a method, and test it in isolation from the rest of the system. This allows developers to quickly identify and fix any issues early in the development process, improving the overall quality of the software and reducing the time required for later testing.

**INTEGRATION TESTING**

Integration Testing (or I&T: Integration & Testing) is a type of software testing in which various units, modules, and components of the software are integrated and tested as a cohesive unit. With Integration Testing, testers want to find defects that surface due to code conflicts between software components when they work together. This level of testing is higher than unit testing, which only validates if software components function well as a single unit.

Integration testing occurs after unit testing, ideally after functional testing, and before system testing. The modules that have passed unit testing will be grouped together. After that, they create a test strategy, perform testing, and make recommendations where necessary.

**FUNCTIONAL TESTING**

        Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is centered on the following items:

Valid Input              : identified classes of valid input must be accepted.

Invalid Input            : identified classes of invalid input must be rejected.

Functions                 : identified functions must be exercised.

Output                      : Classes of application outputs must be exercised.

Systems/Procedures: interfacing systems or procedures must be invoked.

     Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

**WHITE BOX TESTING**

White box testing techniques analyze the internal structures the used data structures, internal design, code structure, and the working of the software rather than just the functionality as in black box testing. It is also called glass box testing or clear box testing or structural testing. White Box Testing is also known as transparent testing, open box testing.

White box testing is a software testing technique that involves testing the internal structure and workings of a software application. The tester has access to the source code and uses this knowledge to design test cases that can verify the correctness of the software at the code level.

**BLACK BOX TESTING**

Black box testing is a technique of software testing which examines the functionality of software without peering into its internal structure or coding. The primary source of black box testing is a specification of requirements that is stated by the customer.

In this method, tester selects a function and gives input value to examine its functionality, and checks whether the function is giving expected output or not. If the function produces correct output, then it is passed in testing, otherwise failed. The test team reports the result to the development team and then tests the next function. After completing testing of all functions if there are severe problems, then it is given back to the development team for correction.

**USER ACCEPTING TESTING**

User Acceptance Testing (UAT) is a type of testing performed by the end user or the client to verify/accept the software system before moving the software application to the production environment. UAT is done in the final phase of testing after functional, integration and system testing is done.

The main Purpose of UAT is to validate end to end business flow. It does not focus on cosmetic errors, spelling mistakes or system testing. User Acceptance Testing is carried out in a separate testing environment with production-like data setup. It is kind of black box testing where two or more end-users will be involved.

**OUTPUT TESTING**

After performing the next step is output of testing of the proposed system since no system could be useful if it does not produce the required output in the specific format. The output generated or displayed by the system under consideration is tested asking the users about the format required by then. Here, the output is considered into two ways: one is on the screen and the others print format.The output format on the screen is found to be correct as the format designed according to the user needs. For the hard copy also; the outcome comes as specified by the user. Hence output testing doesn’t result in any connection in the system.

# IMPLEMENTATION

1. Collecting Data:As you know, machines initially learn from the data that you give them. It is of the utmost importance to collect reliable data so that your machine learning model can find the correct patterns. The quality of the data that you feed to the machine will determine how accurate your model is. If you have incorrect or outdated data, you will have wrong outcomes or predictions which are not relevant. Make sure you use data from a reliable source, as it will directly affect the outcome of your model. Good data is relevant, contains very few missing and repeated values, and has a good representation of the various subcategories/classes present.

2.Preparing the data:

1. Putting together all the data you have and randomizing it. This helps make sure that data is evenly distributed, and the ordering does not affect the learning process.
2. Cleaning the data to remove unwanted data, missing values, rows, and columns, duplicate values, data type conversion, etc. You might even have to restructure the dataset and change the rows and columns or index of rows and columns.
3. Visualize the data to understand how it is structured and understand the relationship between various variables and classes present.
4. Splitting the cleaned data into two sets - a training set and a testing set. The training set is the set your model learns from. A testing set is used to check the accuracy of your model after training.

3.Choosing a Model:A machine learning model determines the output you get after running a machine learning algorithm on

the collected data. It is important to choose a model which is relevant to the task at hand. Over the

years, scientists and engineers developed various models suited for different tasks like speech

recognition, image recognition, prediction, etc. Apart from this, you also have to see if your model is

suited for numerical or categorical data and choose accordingly.

4.Training the Model: Training is the most important step in machine learning. In training, you pass the prepared data to your machine learning model to find patterns and make predictions. It results in the model learning from the data so that it can accomplish the task set. Over time, with training, the model gets better at predicting.

5.Evaluating the Model:After training your model, you have to check to see how it’s performing. This is done by testing the performance of the model on previously unseen data. The unseen data used is the testing set that you split our data into earlier. If testing was done on the same data which is used for training, you will not get an accurate measure, as the model is already used to the data, and finds the same patterns in it, as it previously did. This will give you disproportionately high accuracy. When used on testing data, you get an accurate measure of how your model will perform and its speed.

6.Parameter Tuning:Once you have created and evaluated your model, see if its accuracy can be improved in any way. This is done by tuning the parameters present in your model. Parameters are the variables in the model that the programmer generally decides. At a particular value of your parameter, the accuracy will be the maximum. Parameter tuning refers to finding these values.

7.Making Predictions:In the end, you can use your model on unseen data to make predictions accurately.

# SOURCE CODE

import numpy as np

import pandas as pd

import matplotlib.pyplot as plt

import seaborn as sb

from sklearn.model\_selection import train\_test\_split

from sklearn.preprocessing import StandardScaler

from sklearn import metrics

from sklearn.svm import SVC

from xgboost import XGBClassifier

from sklearn.linear\_model import LogisticRegression

from imblearn.over\_sampling import RandomOverSampler

import warnings

warnings.filterwarnings('ignore')

df = pd.read\_csv('Rainfall.csv')

df.head()

df.shape

df.info()

df.describe().T

df.isnull().sum()

df.columns

df.rename(str.strip,axis='columns',inplace=True)

df.columns

for col in df.columns:

  # Checking if the column contains

  # any null values

  if df[col].isnull().sum() > 0:

    val = df[col].mean()

    df[col] = df[col].fillna(val)

df.isnull().sum().sum()

plt.pie(df['rainfall'].value\_counts().values,

        labels = df['rainfall'].value\_counts().index,

        autopct='%1.1f%%')

plt.show()

plt.subplots(figsize=(15,8))

for i, col in enumerate(features):

  plt.subplot(3,4, i + 1)

  sb.distplot(df[col])

plt.tight\_layout()

plt.show()

df.drop(['maxtemp', 'mintemp'], axis=1, inplace=True)

features = df.drop(['day', 'rainfall'], axis=1)

target = df.rainfall

X\_train, X\_val, Y\_train, Y\_val = train\_test\_split(features,target,test\_size = 0.2,stratify = target,random\_state=2)

# As the data was highly imbalanced we will

# balance it by adding repetitive rows of minority class.

ros = RandomOverSampler(sampling\_strategy='minority',

                        random\_state=22)

X, Y = ros.fit\_resample(X\_train,Y\_train)

# Normalizing the features for stable and fast training.

scaler = StandardScaler()

X = scaler.fit\_transform(X)

X\_val = scaler.transform(X\_val)

models =[LogisticRegression(), XGBClassifier(), SVC(kernel='rbf', probability=True)]

for i in range(3):

  models[i].fit(X, Y)

  print(f'{models[i]} : ')

  train\_preds = models[i].predict\_proba(X)

  print('Training Accuracy : ', metrics.roc\_auc\_score(Y, train\_preds[:,1]))

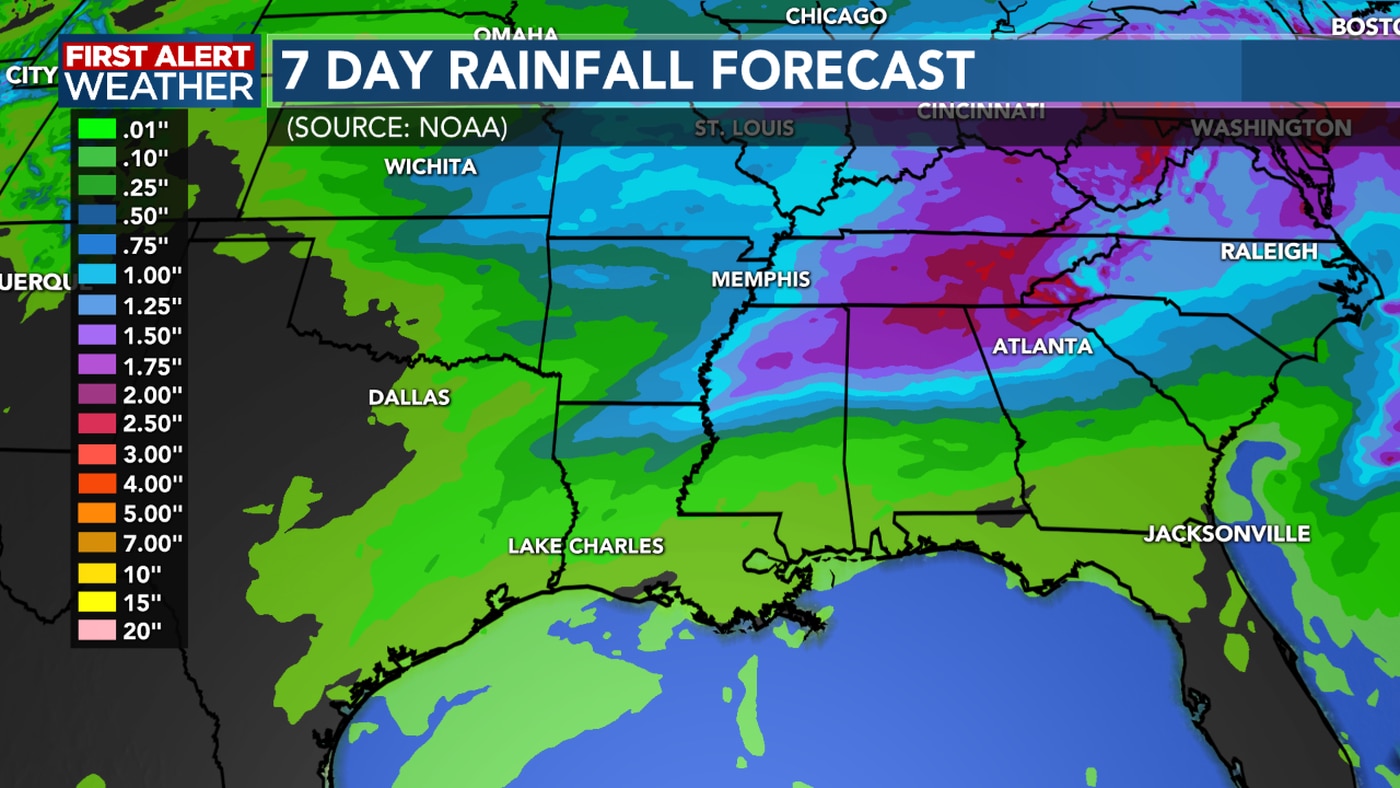
  val\_preds = models[i].predict\_proba(X\_val)

  print('Validation Accuracy : ', metrics.roc\_auc\_score(Y\_val, val\_preds[:,1]))

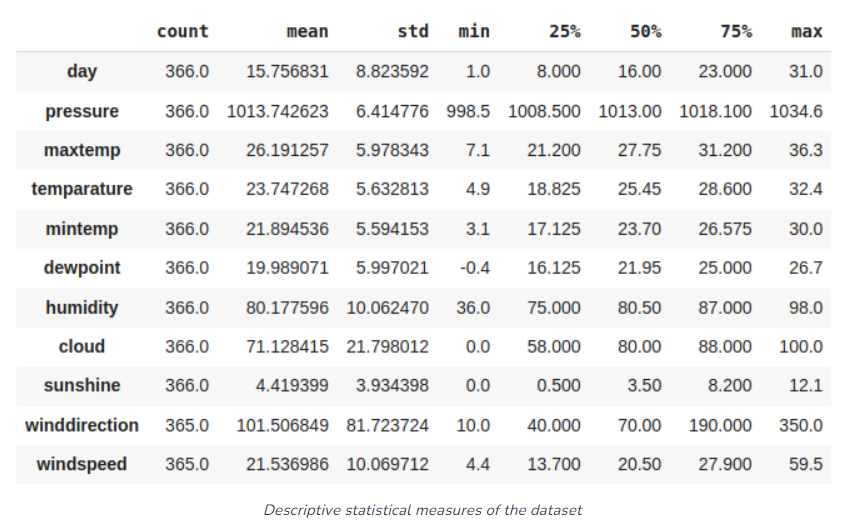
  print()

print(metrics.classification\_report(Y\_val,models[2].predict(X\_val)))

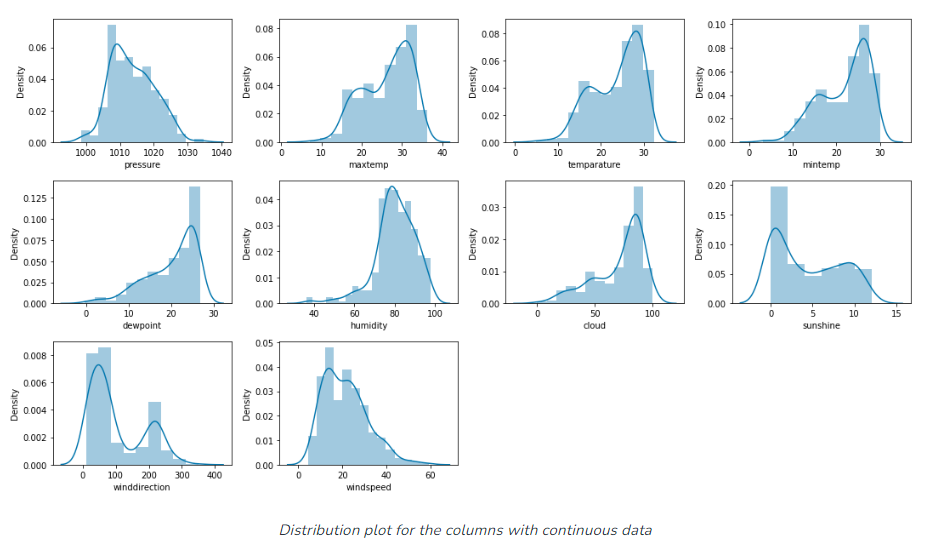
# SCREENSHOTS

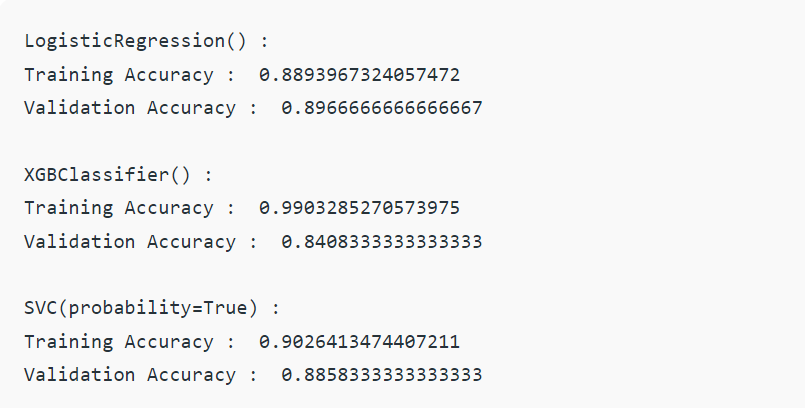


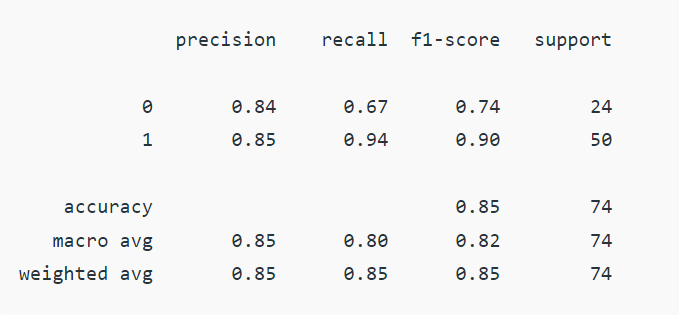
**RAINFALL DATASET**



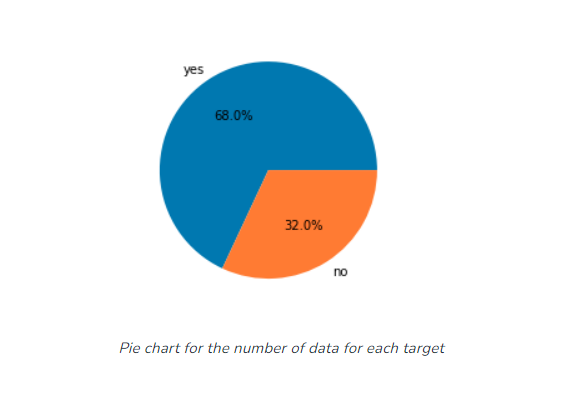
**RESULT**







**PIE CHART**



**CONCLUSION:**

The overall aim is to define various ML techniques that are useful in predicting rainfall. The goal of this research is to design accurate and efficient model by applying lesser number of attributes and tests. Firstly, the data is pre-processed and then it is used in the model. K-Nearest Neighbor with 87% and Random Forest classifier with approximately 88% are the most efficient classification algorithms. However, Decision Tree classifier gives the least accuracy with 73%. We can further expand this research covering other ML techniques such as time series, clustering and association rules and other ensemble techniques. Taking into consideration the limitations of this study, there is a need to build more complex and combination of models to get higher accuracy for rainfall prediction system. Study can also be formulated using greater articulate monitoring for particular area and create this kind of model for enormous dataset so that calculation rate can be increased with better precision and accuracy.

# REFRENCES:

1. "Analytics: Business Intelligence, Algorithms and Statistical Analysis (Predictive Analytics, Data Visualization, Data Analytics, Business Analytics, Decision Analysis, Big Data, Statistical Analysis)" ,Todd J Blatt.
2. “Rain Prediction Based on Machine Learning”,Ye Zhao,Hanqi Shi,Mengyan He,Haotian Deng,Zhou Tong,Yifei Ma.
3. “[Bad Data Handbook: Cleaning Up The Data So You Can Get Back To Work](https://amzn.to/3b5yutA)” , [Q. Ethan Mccallum](https://qethanm.cc/) (2012).
4. “[Best Practices in Data Cleaning: A Complete Guide to Everything You Need to Do Before and After Collecting Your Data](https://amzn.to/35wjsvx)”,[Jason Osborne](https://www.linkedin.com/in/jasonwosborne/) (2012).
5. “Machine Learning Techniques for Space Weather”,Enrico Camporeale,Simon Wing,Jay R.Johnson(2018).
6. “The Weather Observer’s Handbook”,Stephen Burt(2012)
7. “Smart weather prediction using machine learning”,Suvendra Kumar Jaysingh,Sipali Pradhan(2022).
8. “A Guide to Automated Deep Learning for Natural Language Processing”,crown iconRendyk(2021).
9. “Fundamentals of Machine Learning for Predictive Data Analysis”,John D.Kelleher,Brian Mac Namee,Aoife D’Arcy(2015).
10. “Machine Learning for Data Streams”,Albert Bifet,Ricard Gavalda,Geoffery Holmes,Bernhard Pfahringer(2023).

These references provide a comprehensive overview of the various techniques, algorithms, and approaches used in rainfall prediction in ML.