**PROJECT TITLE:**

**WEATHER FORECASTING ABSTRACT**

**Weather forecasting** is the practice of predicting the state of the atmosphere for a given location based on different weather parameters using the application of science and technology. Weather forecasts are made by gathering data about the current state of the atmosphere. Accurate weather forecasting has proven to be a challenging task for meteorologists and researchers. **Weather information is essential in every facet of life like agriculture, tourism, airport system, mining industry, and power generation**. Weather warnings are a special kind of short-range forecast carried out for the protection of human life. Weather warnings are issued by the governments throughout the world for all kinds of threatening weather events including tropical storms and cyclones depending upon the location. The forecast may be short-range or long-range. Weather forecasting has now entered the **era of Big Data** due to the advancement of climate observing systems like satellite meteorological observation and because of the fast boom in the volume of weather data.

This project mainly focuses on prediction of weather conditions at some future time and the weather conditions that may be expected. The weather condition parameters based on **factors like temperature, wind speed, humidity, cloudiness and rainfall**. To forecast the status of the weather of the next month, the algorithms that we use in this project are **Decision Tree Regression, Linear Regression, Random Forest Regression, Naïve Bayes and K Nearest Neighbor**. This project is **to increase the accuracy and predict weather in the future** at least one month priorly using machine learning techniques.

**Methodologies:**

We follow a structured methodology for our projects which starts from designing the solution to the implementation phase. Well planned Project reduces the time to deliver the project, hence we dedicate majority of our time understanding the weather changes and gather requirements. This ground up approach helps us deliver not only the solution to the uses but also match the actual weather conditions to the predicted -

* Analog forecasting – A method of forecasting where a current event is compared with similar events from the past in order to predict the future.

* Persistence forecasting – A method of forecasting where the forecast for a particular time period is the same as the forecast for the previous time period.

* Climatology – A method of forecasting where the forecast is based on the longterm climate statistics for a particular location.

* Numerical weather prediction – A method of forecasting where mathematical models are used to predict the weather.

Each of these methodologies has its own strengths and weaknesses, and forecasters will often use a combination of them in order to produce the most accurate forecast possible.

**Learning Objectives/Internship Objectives**

* Internships are generally thought of to be reserved for college students looking to gain experience in a particular field. However, a wide array of people can benefit from Training Internships in order to receive real world experience and develop their skills.

* An objective for this position should emphasize the skills you already possess in the area and your interest in learning more

* Internships are utilized in a number of different career fields, including architecture, engineering, healthcare, economics, advertising and many more.

* Some internship is used to allow individuals to perform scientific research while others are specifically designed to allow people to gain first-hand experience working.

* Utilizing internships is a great way to build your resume and develop skills that can be emphasized in your resume for future jobs. When you are applying for a Training Internship, make sure to highlight any special skills or talents that can make you stand apart from the rest of the applicants so that you have an improved chance of landing the position.

**INDEX**

**S.No. CONTENTS Page No.**

1. Introduction 15-16
2. Advantages and Disadvantages 17
3. System Requirements 18
   1. System Configuration 18
   2. Software Requirements 18
   3. Hardware Requirements 18
4. Technology Used 19-24
   1. Python 19
   2. Machine Learning 20
   3. Algorithms Used 21-24

#### 4.3.1 Random Forest 21

4.3.2 Decision Tree 22

4.3.3 Linear Regression 23

4.3.4 Naïve bayes 24

1. Software Used 25-28
   1. Anaconda 25
   2. Jupyter Notebook 25
   3. Python 26-28
2. Motivation 29
3. Literature Review 30
4. System Analysis 31
   1. Existing System 31
   2. Proposed System 31
5. Keywords and Definitions 32-33
6. Implementation and Architecture 34-36
   1. Implementation 34
   2. Architecture 35
   3. Data Flow 36
7. Challenges faced 37-38
8. Code 39-58
9. Result 59
10. Testing 60-61
11. Conclusion 62
12. Bibliography 63

**WEEKLY OVERVIEW OF INTERNSHIP ACTIVITIES:**

|  |  |  |  |
| --- | --- | --- | --- |
| **1**  **st WEEK** | **DATE** | **DAY** | **NAME OF THE TOPIC/MODULE COMPLETED** |
| 04-03-24 | Monday | Introduce the Topic & the Problem Statement |
| 05-03-24 | Tuesday | Introduce the Topic & the Problem Statement |
| 06-03-24 | Wednesday | Abstract Building |
| 07-03-24 | Thursday | Abstract Building |
| 08-03-24 | Friday | Abstract Building |

|  |  |  |  |
| --- | --- | --- | --- |
| **2**  **nd**    **WEEK** | **DATE** | **DAY** | **NAME OF THE TOPIC/MODULE COMPLETED** |
| 11-03-24 | Monday | Explain your approach to solving problem |
| 12-03-24 | Tuesday | Explain your approach to solving problem |
| 13-03-24 | Wednesday | Explain structure of project |
| 14-03-24 | Thursday | Explain structure of project |
| 15-03-24 | Friday | Explain structure of project |

|  |  |  |  |
| --- | --- | --- | --- |
| **3**  **rd**    **WEEK** | **DATE** | **DAY** | **NAME OF THE TOPIC/MODULE COMPLETED** |
| 18-03-24 | Monday | Data Preprocessing |
| 19-03-24 | Tuesday | Data Preprocessing |
| 20-03-24 | Wednesday | Data Preprocessing |
| 21-03-24 | Thursday | Perform analysis |
| 22-03-24 | Friday | Perform analysis |

|  |  |  |  |
| --- | --- | --- | --- |
| **4**  **th**    **WEEK** | **DATE** | **DAY** | **NAME OF THE TOPIC/MODULE COMPLETED** |
| 25-03-24 | Monday | PPT Presentation |
| 26-03-24 | Tuesday | Building and applying algorithm |
| 27-03-24 | Wednesday | Building and applying algorithm |
| 28-03-24 | Thursday | Building and applying algorithm |
| 29-03-24 | Friday | Building and applying algorithm |

**1. INTRODUCTION**

Weather forecasting can be defined as an attempt to predict the future weather conditions based on previously collected data. Sometimes extreme weather conditions can cause heavy losses. If we can predict the future weather conditions properly such losses can be minimized. Weather forecasting is important to each and every one, whether it is a student who decides to carry an umbrella or not by knowing the weather, or a government organization which helps people to vacate a location by knowing that it is going to rain heavily in that region. Existing systems give information about weather in terms of a wide ranged value. For example, the temperature is going to be ranging from 21 degrees to 29 degrees Celsius. Such systems are very confusing and are not very helpful to the people. There are various fields which need weather prediction data. People who consume weather prediction results may include farmers, pilots, power generation stations which depend on solar energy and wind energy and many more. There is a need for systems which can predict the weather conditions very accurately at a specific time and location. Machine Learning models can be used to build systems that can predict the weather with high accuracy. The most important part of any machine learning models is the data. The prediction accuracy of a machine learning model is completely dependent on the data that it is trained with.

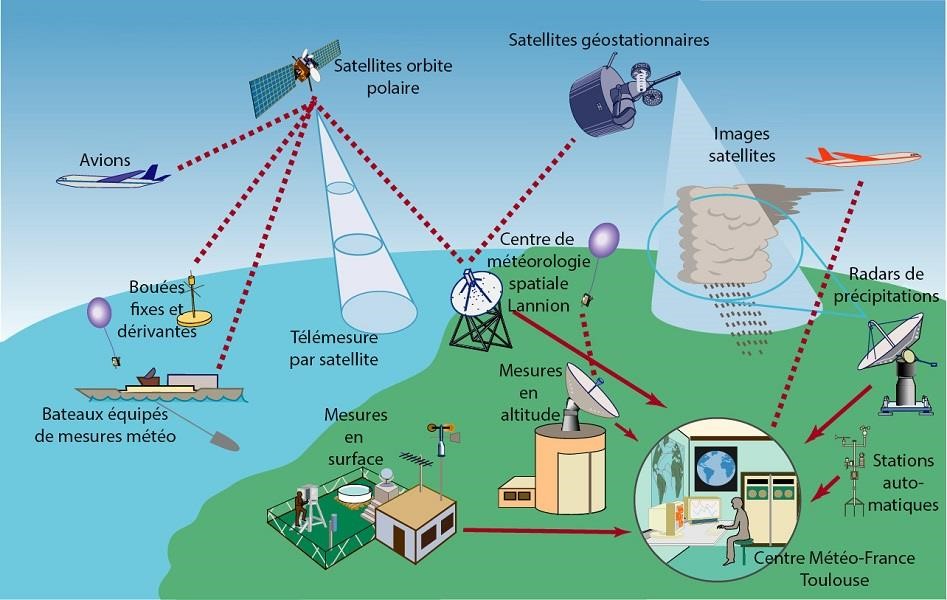
The remarkable improvement in the quality of weather forecasts is one of the great successes of environmental science in the 20th century, which continues at a sustained pace at the beginning of the 21st century. This is due to the progress of numerical prediction systems and the increasing number and variety of observations of the state of the atmosphere and related media (ocean, soils, vegetation, cryosphere), including observations from Earth observation satellites. The rapid development of supercomputers has been one of the keys to this success, which has also required significant scientific work.

Each country in the world has a National Meteorological Service (NMS), whose mission is to make regular observations of the atmosphere and to issue forecasts for government, industry and the public. But only the most advanced countries have

Numerical Weather Prediction (NWP) centres, whose products are also distributed to other countries, in exchange for their observations, within the framework of the World Meteorological Organization.

Among the main NWP centres outside Europe are those in the United States, Canada, Japan, Korea, China, Russia, Australia, India, Morocco, South Africa and Brazil. In Europe, only France, the United Kingdom and Germany make numerical forecasts for the entire globe, while the other countries have NWP centres covering only regional areas. The European countries have also come together in a “super-centre”, which is responsible for providing them with medium-range numerical forecasts.

`



**2. ADVANTAGES AND DISADVANTAGES:**

**Advantages:**

* Farmers can know when to plant or harvest their crops
* People can choose where and when to take their holidays to take advantages of good weather
* Surfers known when large waves are expected
* Regions can be evacuated if hurricanes or floods are expected
* Aircraft and shipping rely heavily on accurate weather forecasting

**Disadvantages:**

* Weather is extremely difficult to forecast correctly
* It is expensive to monitor-so many variables from so many sources
* The computers needed to perform the millions of calculations necessary are expensive
* The weather forecasters get blamed if the weather is different from the forecast

**3. SYSTEM REQUIREMENTS**

**System configurations**

The software requirement specification can produce at the culmination of the analysis task. The function and performance allocated to software as part of system engineering are refined by established a complete information description, a detailed functional description, a representation of system behavior, and indication of performance and design constrain, appropriate validate criteria, and other information pertinent to requirements.

**3.1 Software requirements:**

* **Operating System:** Windows, Linux.
* **Language:** Python
* **Python IDE:** Jupyter Notebook, Python 3.
* **Python 3.7:** Python is an interpreted, high level, general programming language. Its formatting is visually uncluttered, and it often uses English keywords where other languages use punctuation. It provides a vast library for data mining and predictions.
* **Jupiter Notebook/ Spider/ PyCharm:** It is an open-source cross-

platform integrated development environment (IDE) for scientific programming in the Python language. Spyder integrates with a number of prominent packages as well as another open-source software.

* **NumPy:** NumPy was used for building the front-end part of the system.
* **Pandas:** Pandas was used for the data preprocessing and statistical analysis of data.
* **Matplotlib**: Matplotlib was used for the graphical representation of our prediction.

**3.2 Hardware Requirements:**

* **Processor:** 64-bit, quad-core, 2.5 GHz minimum per core
* **Ram:** 4 GB or more
* **Hard disk:** 20 GB of available space or more.
* **Display:** Dual XGA (1024 x 768) or higher resolution monitors

**4. TECHNOLOGIES USED**

**4.1 PYTHON:**

* Python is a programming language that is created in 1991 by Guido van Rossum.
* It is commonly used in the web development, software development and system scripting. It is highly level authenticated, interpreted, interactive and object oriented
* Python is designed to be easily readable. It uses the keywords in English language where as other languages use punctuation, and it has fewer syntactical constructions than other languages.
* Python is a widely used dynamic programming language compared to other languages such as Java, Perl, PHP, and Ruby. It is often termed as a scripting language.
* It provides support for automatic memory management, multiple programming paradigms, and implements the basic concepts of object-oriented programming (OOP).
* Python is a procedural language which is of strongly typed form with support for a huge and broad standard library. The Python library supports many Internet protocols such as FTP, and IMAP.

**Features of python:**

* Easy to Learn
* Free and Open Source
* High-level Language
* Portable
* Interpreted
* Extensible
* Embeddable

**Advantages of python:**

* Python code is easy to read and can execute a lot of complex functionalities with ease,
* It is Object Oriented and Programming Driven and it supports many systems and platforms.
* With the innovation of Raspberry Pi, a card sized microcomputer, it has expanded its reach to unprecedented heights. Developers can build cameras, radios and games.
* Python has a framework that make web programming very flexible. Django is one of the most widely used Python framework for web development.
* A large number of resources is available in Python. It offers a built-in testing framework to set debugging time and enable fastest workflows.

**Disadvantages of python:**

* Python is slow and it is not a very good language for mobile development.
* Python is not a good choice for memory intensive tasks and has limitations with database  access.
* It's near impossible to build a high-graphic 3D game using Python**.**

**4.2 MACHINE LEARNING:**

Machine learning is an application of artificial intelligence (AI) that provides systems the ability to automatically learn and improve from experience without being explicitly programmed. Machine learning focuses on the development of computer programs that can access data and use it learn for themselves.

The process of learning begins with observations or data, such as examples, direct experience, or instruction, in order to look for patterns in data and make better decisions in the future. The primary aim is to allow the computers learn automatically without human intervention or assistance and adjust actions accordingly.

Machine learning algorithms are often categorized as supervised or unsupervised.

* **Supervised machine learning algorithms** can apply what has been learned in the past to new data using labelled examples to predict future events. Starting from the analysis of a known training dataset, the learning algorithm produces an inferred function to make predictions about the output values. The system is able to provide targets for any new input after sufficient training. The learning algorithm can also compare its output with the correct, intended output and find errors in order to modify the model accordingly.

* In contrast, **unsupervised machine learning algorithms** are used when the information used to train is neither classified nor labelled. Unsupervised learning studies how systems can infer a function to describe a hidden structure from un labeled data. The system doesn’t figure out the right output, but it explores the data and can draw inferences from datasets to describe hidden structures from un labeled data.

**4.3 ALGORITHMS USED:**

**4.3.1 Random Forest Algorithm:**

Random forest is a supervised learning algorithm which is used for both classification as well as regression. But however, it is mainly used for classification problems. As we know that a forest is made up of trees and more trees means more robust forest. Similarly, random forest algorithm creates decision trees on data samples and then gets the prediction from each of them and finally selects the best solution by means of voting. The algorithm works by constructing a number of decision trees at training time and outputting the class that is the mode of the classes (classification) or mean prediction (regression) of the individual trees. It is an ensemble method which is better than a single decision tree because it reduces the overfitting by averaging the result.

**Advantages:**

* + - * Powerful and accurate
      * Random forests are extremely flexible and have very high accuracy.
      * Good performance on many problems including non-linear.
      * Random forests also have less variance than a single decision tree.
      * Can be used for both regression and classification tasks.
      * Can handle large datasets and complex problems.
      * Is not sensitive to outliers and can detect them.
      * Has a good accuracy **Disadvantages:**
      * Can be slow to train.
      * Can be difficult to interpret.
      * The main disadvantage of Random forests is their complexity. They are much harder and time-consuming to construct than decision trees.
      * Overfitting can easily occur
      * In addition, the prediction process using random forests is time-consuming than other algorithms.
      * So basically, Random Forest is used when you are just looking for high performance with less need for interpretation

**4.3.2 Decision Tree Algorithm:**

A decision tree is a supervised learning algorithm that can be used for both classification and regression tasks. The goal of a decision tree is to create a model that predicts the value of a target variable based on other variables in the data. Decision trees are created by splitting the data into smaller groups based on an attribute value. The tree is then created by selecting the attribute that results in the largest information gain, which is the difference in entropy before and after the split. The process is then repeated on each of the child nodes, until the tree is complete. Once the tree is complete, it can be used to make predictions by traversing the tree from the root to the leaves. The prediction is made by taking the majority vote of the values at the leaves.

There are several advantages and disadvantages to using decision trees in machine learning-

One advantage is that decision trees are very efficient when it comes to both training data and prediction time. They are also very accurate, which makes them a popular choice for many machine learning applications.

However, decision trees can be overfit to training data if they are not pruned properly. This can lead to poor performance on test data. Decision trees can also be unstable, meaning that small changes in the training data can lead to large changes in the tree structure.

* Decision Tree is a Supervised learning technique that can be used for both classification and Regression problems, but mostly it is preferred for solving Classification problems. It is a tree-structured classifier, where internal nodes represent the features of a dataset, branches represent the decision rules and each leaf node represents the outcome.
* In a Decision tree, there are two nodes, which are the Decision Node and Leaf Node. Decision nodes are used to make any decision and have multiple branches, whereas Leaf nodes are the output of those decisions and do not contain any further branches.
* The decisions or the test are performed on the basis of features of the given dataset.
* It is a graphical representation for getting all the possible solutions to a problem/decision based on given conditions.
* It is called a decision tree because, similar to a tree, it starts with the root node, which expands on further branches and constructs a tree-like structure.
* In order to build a tree, we use the CART algorithm, which stands for Classification and Regression Tree algorithm.  A decision tree simply asks a question, and based on the answer (Yes/No), it further split the tree into subtrees.

#### 4.3.3 Linear Regression:

Linear regression is a supervised machine learning algorithm that is used to predict a continuous dependent variable given a set of independent variables. It is a linear approach to modeling the relationship between a dependent variable and one or more independent variables. Linear regression is widely used in many different fields, such as economics, finance, and engineering.

There are several advantages to using linear regression in machine learning:

* Linear regression is a relatively simple and straightforward technique that is easy to understand and implement.

* Linear regression is relatively efficient and can be applied to large datasets.

* Linear regression can be used to predict continuous values (such as prices, temperatures, etc.) and can also be used to predict binary outcomes (such as whether an individual will default on a loan).

However, linear regression also has a few disadvantages:

* Linear regression makes strong assumptions about the data, such as that the data is linear, homoscedastic, and free of outliers. Violating these assumptions can lead to inaccurate results.

* Linear regression can be sensitive to the inclusion of irrelevant features in the model. This can lead to overfitting, which results in poor generalization performance on out-of-sample data.

* Linear regression does not directly provide information about feature importance or which features are most predictive of the target variable. This can make it difficult to interpret the results of a linear regression model.

**4.3.4 Naive Bayes:**

Naïve Bayes is a machine learning algorithm that is used for classification tasks. The algorithm is based on the Bayesian theorem, which states that the probability of an event occurring is equal to the product of the prior probability of the event and the likelihood of the event. Naive Bayes is a simple and effective machine learning algorithm that can be used for a variety of tasks, including text classification, spam filtering, and document classification. Naive Bayes is a simple but powerful technique for building predictive models.

It has several advantages over more sophisticated methods:

* + Naive Bayes is easy to understand and implement.

* + Naive Bayes is often very accurate, especially when the underlying assumptions are true.

* + Naive Bayes is relatively efficient, both in terms of training time and prediction time.

* + Naive Bayes can be extended to handle more complicated data structures and relationships.

However, naive Bayes also has some disadvantages:

* + Naive Bayes makes strong independence assumptions. If the assumption is violated, the predictions will be inaccurate.

* + Naive Bayes can be sensitive to noisy data.

* + Naive Bayes can struggle with high-dimensional data.

* + Naive Bayes can be slow to converge if the data is very large.

# 5. SOFTWARE USED

**5.1 ANACONDA:**

Anaconda is a complete, open-source data science package with a neighbor hood of over 6 million customers and it includes greater than 250 popular data science applications suitable for Windows, Linux, and MacOS. It is a free and open-source distribution of the Python and R programming languages for data science and machine learning associated applications (large-scale data processing, predictive analytics, scientific computing), that objectives to simplify package deal administration and deployment. The distribution comes with extra than 1000 statistics packages as well as the Conda package deal and virtual environment manager, so it eliminates the need to learn to installation each library independently. As Anaconda’s website says,” The Python and R conda packages in the Anaconda Repository are curate and complied in our secure surroundings so you get optimized binaries that ‘just work’ on your system.” Here we used, Anaconda Navigator, a computer graphical consumer interface (GUI) system that includes hyperlink to all the applications included with the distribution inclusive of RStudio, iPython, Jupyter, Notebook, JupyterLab, Spyder, and Orange.

The default environment is Python 3.10 and Python 3.9 or R.

## 5.2 JUPYTER NOTEBOOK:

The Jupyter Notebook is an open-source net application that permits you to form and share documents that contain live code, equations, visualizations and narrative text. Uses include: information cleanup and transformation, numerical simulation, applied math modeling, information mental image, machine learning, and far additional.

* Language of choice: The Notebook has support for over forty programming languages, together with Python, R, Julia, and Scala.
* Share notebooks: Notebooks are often shared with others victimization email, Drop box, Git Hub and therefore the Jupyter Notebook Viewer.
* Interactive output: Your code will manufacture made, interactive output: markup language, images, videos, LaTeX, and custom MIME varieties.
* Massive data integration: Leverage big data tools, like Apache Spark, from Python, R and Scala. Explore that very same data with pandas, scikit-learn, ggplot2, TensorFlow.

Unique features:

* Supports all imports and exports
* Great for sharing and collaborative work
* Supports variety of data types within the same window such as text, code, graphs, videos, pictures
* Great for Visualizations
* Enables parallel computing
* Presentation feature in the Jupyter and ipython notebook where you can make the presentation directly from your notebook. One such extension is “RISE”.

## 5.3 PYTHON PACKAGES:

* NumPy:

The NumPy library is a fundamental library for scientific computing in Python. It provides a high-performance multidimensional array object, and tools for working with these arrays.

NumPy is an essential library for machine learning in Python. It provides a powerful array object and tools for working with arrays. NumPy arrays are efficient data structures for working with numerical data. They are also convenient for working with data in various formats, such as images and matrices.

NumPy is a key library for scientific computing in Python. It provides a high-performance multidimensional array object, and tools for working with these arrays. NumPy is an essential library for machine learning in Python. It provides a powerful array object and tools for working with arrays. NumPy arrays are efficient data structures for working with numerical data. They are also convenient for working with data in various formats, such as images and matrices.

* Pandas:

Pandas is a powerful tool for data analysis in Python. It can be used for data cleaning and manipulation, feature engineering, and even machine learning. Pandas is particularly well-suited for machine learning because it provides a number of features that are very useful for working with data:

* + Pandas can handle missing data gracefully.
  + Pandas can easily convert data from one format to another (e.g., from a CSV file to a SQL database).
  + Pandas provides many built-in functions for data analysis, which can be used for preprocessing data for machine learning.
  + Pandas is fast and efficient, which is important for working with large datasets.
  + Pandas is easy to use, which makes it a good choice for beginners.

* Seaborn:

Seaborn is a Python data visualization library based on matplotlib. It provides a high-level interface for drawing attractive and informative statistical graphics. Seaborn is particularly useful for exploring linear relationships between multiple variables. Some of the most useful functions in Seaborn for machine learning are the pairplot() and jointplot() functions. These functions create scatter plots between pairs of variables, and can be used to quickly visualize relationships between many variables.

The Seaborn library is also great for creating visualizations of categorical data. The countplot() function creates a barplot of counts for each category. This is a useful way to understand the distribution of a categorical variable.

The Seaborn library is also great for creating visualizations of time series data. The lineplot () function creates a line plot of a time series. This is a useful way to understand trends over time. Finally, the Seaborn library is also great for creating visualizations of statistical models. The lmplot() function creates a scatter plot of a linear model. This is a useful way to understand the relationship between variables in a linear model

* Sklearn:

Sklearn is a library in machine learning that provides a range of tools for data mining and data analysis. It includes tools for preprocessing data, dimensionality reduction, supervised and unsupervised learning, model selection, and evaluation.

* Matplotlib:

Matplotlib is often used in machine learning to visualize data. For example, it can be used to plot the data points in a dataset, or to visualize the results of a machine learning algorithm.

Matplotlib.pyplot is a collection of command style functions that make matplotlib work like MATLAB. Each pyplot function makes some change to a figure: e.g., creates a figure, creates a plotting area in a figure, plots some lines in a plotting area, decorates the plot with labels, etc. In matplotlib pyplot various states are preserved across function calls, so that it keeps track of things like the current figure and plotting area, and the plotting functions are directed to the current axes (please note that “axes” here and in most places in the documentation refers to the axes part of a figure and not the strict mathematical term for more than one axis). One of the most popular uses for Python is data analysis. Naturally, data scientists want a way to visualize their data. Either they are wanting to see it for themselves to get a better grasp of the data, or they want to display the data to convey their results to someone. With Matplotlib, arguably the most popular graphing and data visualization module for Python, this is very simplistic to do. In this tutorial, I will be covering all of what I consider to be the basic necessities for Matplotlib.

**6. MOTIVATION:**

The U.S. arguably has the world’s worst weather: hurricanes, tornados, large hail, blizzards, droughts, heat waves, etc. Hazardous weather is the reason the National Weather Service (NWS) was founded. The mission of the NWS is to reduce the loss of life and the loss of property associated with weather related hazards, and to mitigate the economic impact of disruptive weather. Weather forecasters look at current state of the weather and forecast maps and add their personal experience to come up with a forecast and to issue warnings.

The motivation of weather forecast is to allow people to better prepare for upcoming weather conditions. By knowing what the weather will be like in advance, people can take steps to protect themselves and their property from the effects of severe weather

The motivation for weather forecasting is to provide information about current and future conditions to help people plan their activities. Weather forecasts are made by collecting data about the current state of the atmosphere and using numerical models to predict how it will change.

# 7. LITERATURE REVIEW:

Prediction of Rainfall was done by N. Singh et al with the help of weather parameters such as Temperature, Humidity and Pressure. Random Forest Classification algorithm was used and 87.90% of accuracy was achieved by this technique.

A. Mahabub had predicted Rainfall, Humidity, Wind Speed, High Temperature and

Low Temperature using weather parameters such as Wind Speed, Humidity,

Temperature and Rainfall. It used several ML Techniques like Support Vector Regression (SVR), Linear Regression, Bayesian Ridge, Gradient Boosting (GB), Extreme Gradient Boosting (XGBoost), Category Boosting (CatBoost), Adaptive Boosting (AdaBoost), k-Nearest Neighbors (KNN) and Decision Tree Regressor (DTR). It was found that, ML-based models are more accurate than conventional methods. However, it can be seen that DTR and CatBoost methods were almost equivalent but adaptability of DTR was more for nonlinear data.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Reference | Parameters | Technique | Result | Prediction |
| N. Singh et al. | Temperature,  Humidity and  Pressure | Random Forest  Classification | Accuracy of 87.90% was achieved. | Rainfall Prediction |
| A. Mahabub | Wind Speed,  Humidity, Temperature and Rainfall | Support Vector  Regression  (SVR), Linear  Regression,  Bayesian Ridge,  Gradient Boosting (GB), Extreme  Gradient Boosting  (XGBoost),  Category  Boosting  (CatBoost),  Adaptive boosting  (AdaBoost), k-  Nearest  Neighbors (KNN) and Decision Tree  Regressor | ML-based models are more accurate than conventional methods. It can be said that DTR and CatBoost methods were almost equivalent but  adaptability of DTR was more  for nonlinear data. | Rainfall  Prediction,  Humidity  Prediction,  Wind Speed  Prediction,  High  Temperature  prediction,  Low  Temperature prediction. |

# 8. SYSTM ANALYSIS:

**8.1 Existing System:**

The existing systems used to forecast weather require a lot of data about the current state of the weather and based on understanding of the atmospheric weather processes to predict how the weather evolves in the future. Some very sophisticated and complex methods which use satellite for prediction of weather exist.

Other than such systems, other systems which make use of machine learning techniques such as Linear regression and statistical methods also exist. The systems which have been proposed are having certain limitations- o In some of the models, data of Weather is taken from the surroundings of the target place., More the surrounding data, less the efficiency as target weather does not always depend on its surrounding places.

* Over prediction is seen in many models. o The predicted readings are 10% (on average) higher than the original values.
* Cost is high when we go for Satellite image processing which indeed gives high accuracy

### 8.2 Proposed System:

User will enter current temperature; humidity and wind, System will take this parameter and will predict weather from previous data in database. The role of the admin is to add previous weather data in database, so that system will calculate weather based on these data. Weather forecasting system takes parameters such as temperature, humidity, and wind and will forecast weather based on previous record therefore this prediction will prove reliable

This is not only used to predict the weather conditions for the near future at a specific location with high accuracy and efficiency but also reduce the reliability on expensive instruments and methods presently used to predict the weather. The predicted weather conditions should match the actual weather conditions. And help daily commuters, farmers and other people by informing the weather situations beforehand so that they can plan their schedule accordingly. To overcome the limitations and to reduce complexity, we have proposed machine learning models

# 9. KEYWORDS AND DEFINITIONS:

* **Machine Learning:**

Machine learning is a subfield of artificial intelligence, which is broadly defined as the capability of a machine to imitate intelligent human behavior.

* **Artificial Intelligence:**

Artificial intelligence is the simulation of human intelligence processes by machines, especially computer systems.

* **Linear Regression:**

Linear regression analysis is used to predict the value of a variable (dependent variable) based on the value of another variable (independent variable).

* **Decision Tree:**

Decision Tree is a technique which uses labelled input and output datasets to train models, i.e., it is a graph that uses a branching method to illustrate every possible output for a specific input

* **Random Forest:**

Random Forest is the combination of large number of relatively uncorrelated models (trees) (i.e., large number of individual decision trees) operating as a committee that will outperform any of the individual constituent models.

* **Naïve Bayes:** 
  1. naive Bayes classifier is an algorithm that uses Bayes' theorem to classify objects that assumes strong, or naive, independence between attributes of data points.

* **K-Nearest Neighbor:**
  1. k-nearest-neighbor algorithm, often abbreviated k-nn, is an approach to data classification that estimates how likely a data point is to be a member of one group or the other depending on what group the data points nearest to it are in.

* **BeautifulSoup:**

BeautifulSoup is a Python library for pulling data out of HTML and XML files.

* **Pandas:**

Pandas is used to load the dataset into the project

* **NumPy:**

NumPy is a Python library used for working with arrays.

* **Matplotlib:**

Matplotlib is used to visualize the data using plots, histograms, boxplots

* **Seaborn:**

Seaborn is a Python data visualization library based on matplotlib. It provides a high-level interface for drawing attractive and informative statistical graphics.

* **Sklearn:**

Scikit-learn is a Python library that consists of many supervised and unsupervised learning algorithms

# 10. IMPLEMENTATION AND ARCHITECTURE:

**10.1 Implementation:**

We basically apply Machine Learning approaches and algorithms to predict the weather of individual based on their physical measures.

The methodology includes several steps so we need to follow a well-defined sequence in order to get more accuracy in Predictions.

1. **Get the dataset**: load the dataset using pandas

1. **Preprocess the dataset**: we need to preprocess the data before we training the model such as finding null values in the dataset and filling with either MEAN,

MEDIAN, MODE

1. **Visualize the data**: we need visualize the data and results, so in this step outliers are detected and also useful to find correlation between every feature in the dataset

1. **Split the dataset**: we need split our entire dataset to training and testing data

1. **Model Selection**: we need to select a machine learning model based on our requirement and outcome and also, we can apply various algorithms and may choose Model that have high accuracy.

1. **Model training**: after model selection we need train the model using our training data

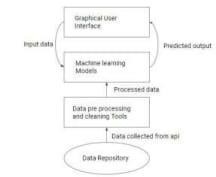
1. **Model testing**: after model training we need to test the Model using testing data.

1. **Accuracy of the model**: after testing the model we need to measure the prediction Capability of the model The model accuracy greater than 80% is said to be efficient Model.

* 1. **Architecture:**

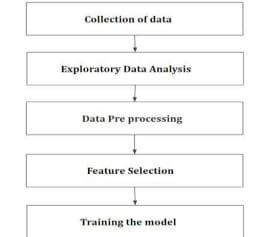
The complete system consists of a Graphical User Interface. The GUI takes the current day’s weather input. These inputs are given to the trained models in order to predict the weather conditions of the near future.

The architecture of the system consists of a data repository consisting of the historical weather data. This data is used for training the models. Then it consists of data preprocessing tools in order to select the required data and to clean the data to get efficient results. On top of that are the machine learning models which take in the preprocessed data and training the model to predict the future weather conditions. And the topmost layer is the Graphical User Interface which helps user to provide inputs for the day and get accurate predictions.



* 1. **Data Flow:**

Since data is the key component in machine learning models, it is very important to clean the raw data and extract only the required data and feed the data to the model in a specific format. The raw data obtained from the API has to be put through a series of steps for processing before it is used for training. The first step is collection of historical data from a trusted source which is of high quality and precision. EDA is the process of conducting initial analysis of data in order to search for any existing helpful patterns, to test assumptions and hypothesis on the data and to find if any anomalies exist in the data. This is usually done with some visual and graphical representation. Some data mining tools are used to transform the raw data into the desired format. Techniques such as data cleaning, data transformation, data reduction are used here to get the desired data. Not all the existing features are necessary for training the model. Only those features which contribute most to the accuracy of the model are selected here. After selecting the required features, the data is divided into a training set and testing set. The training set is used to train the machine learning model.



**11. THE CHALLENGES AND COMPLEXITIES OF WEATHER FORECASTING:**

There are several challenges associated with weather forecast using machine learning.

* Firstly, the data used for training the machine learning models can be quite noisy and incomplete. This can make it difficult for the models to learn the underlying patterns in the data.
* Secondly, the weather is a highly complex phenomenon and even the most sophisticated machine learning models may not be able to capture all the nuances of the weather.
* Finally, the accuracy of weather forecasts can vary significantly depending on the location and time of year, making it difficult to create a generalizable model. **Range/Scope of Weather Forecasting:**

Weather forecasting is a complex and often challenging skill that involves observing and processing vast amounts of data. Weather systems can range from small, shortlived thunderstorms only a few miles in diameter that last a couple hours to large scale rain and snow storms up to a thousand miles in diameter and lasting for days.

Forecasting ultimately is a three-step process. These include:

* 1. Observing
  2. Forecasting
  3. Communicating.

* The weather can change very quickly, making it difficult to predict exactly what the weather will be like at a specific time and place.

* The Earth's atmosphere is a chaotic system. This means that it is impossible to predict the exact state of the atmosphere at a future time. However, it is possible to make predictions about the general behavior of the atmosphere.

* the atmosphere is constantly changing. This means that forecasters must constantly update their models and forecasts.

* weather forecasting is also challenged by the fact that the atmosphere is a three-dimensional system. This means that forecasters must take into account the effects of the atmosphere on different parts of the globe.

* One of the biggest challenges in weather forecasting is predicting the weather several days in advance. This is difficult because the atmosphere is a complex system, and small changes in the initial conditions can lead to large changes in the weather.

* weather patterns can change over time, making it difficult to long-term predictions. For example, a region that typically experiences mild winters may suddenly have a cold snap, or a hurricane that veers off course can cause major damage.

* Another challenge is predicting the timing and intensity of storms. Storms can form quickly and intensify rapidly, making them difficult to forecast.

Finally, the weather is a complex phenomenon that is constantly changing. This makes it difficult to create a detailed and accurate model of the atmosphere.

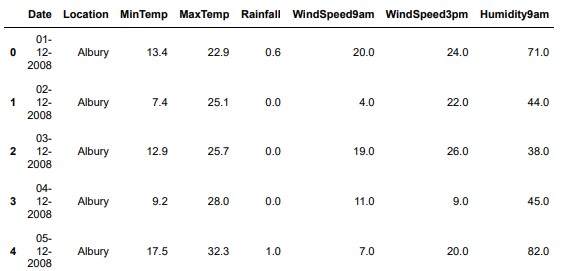
**12. CODING AND IMPLEMENTATION:**

**#Importing libraries**

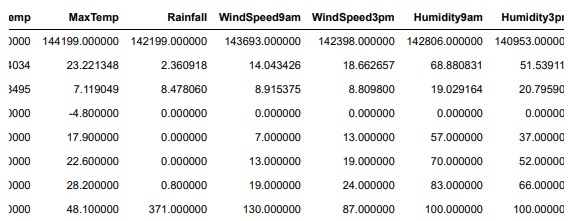
import pandas as pd import numpy as np import matplotlib.pyplot as plt import seaborn as sns plt.style.use('dark\_background')

**#Loading dataset**

df = pd. read\_csv('weatherAU.csv') df.head()



df**.**describe**()**

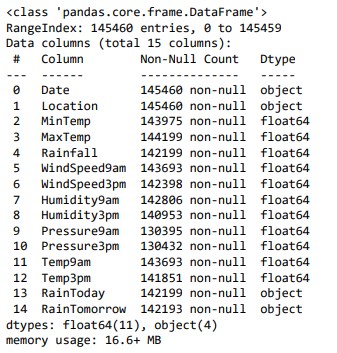


**#Understanding dataset**

df.shape

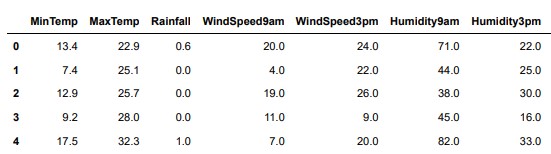


df.info()



**#preprocessing**

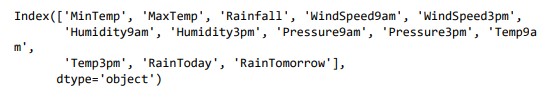
df = df.drop(["Location", "Date"], axis =1) df.head()



df = df.dropna(axis = 0) df.shape



df.columns

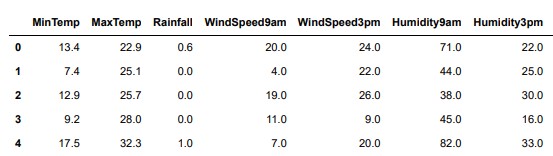


from sklearn.preprocessing import LabelEncoder le = LabelEncoder() df['RainToday'] = le.fit\_transform(df['RainToday'])

df['RainTomorrow'] = le.fit\_transform(df['RainTomorrow'])

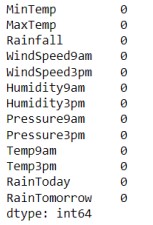
x = df.drop(['RainTomorrow'], axis = 1) y = df['RainTomorrow']

x.head()



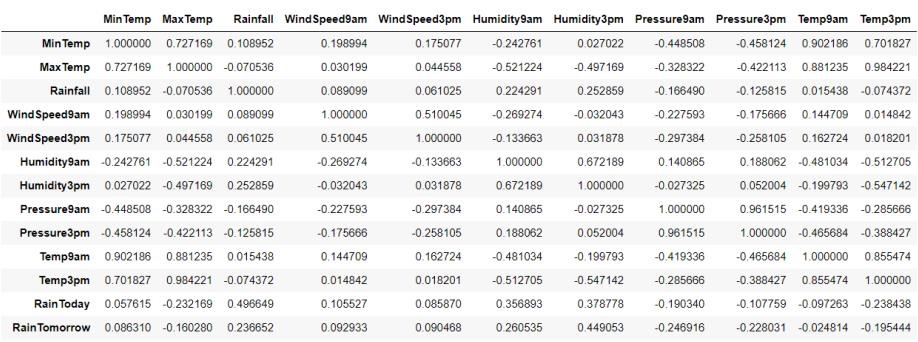
**#Checking null values**

df.isna().sum()



**# Correlation between the features**

df\_corr=df.corr() df\_corr

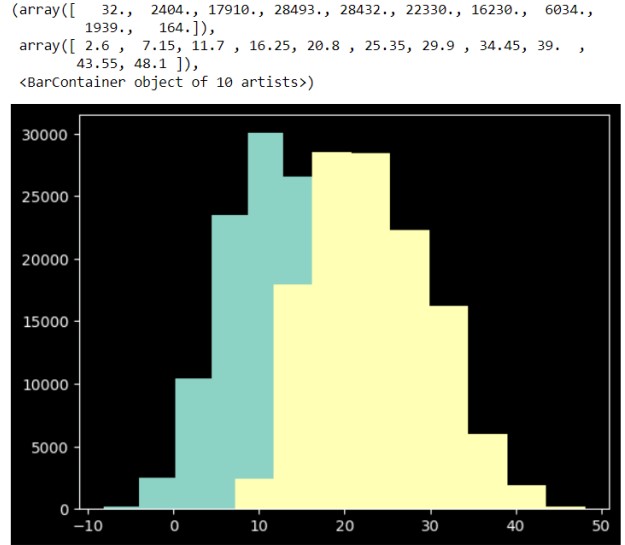


**#Visualization**

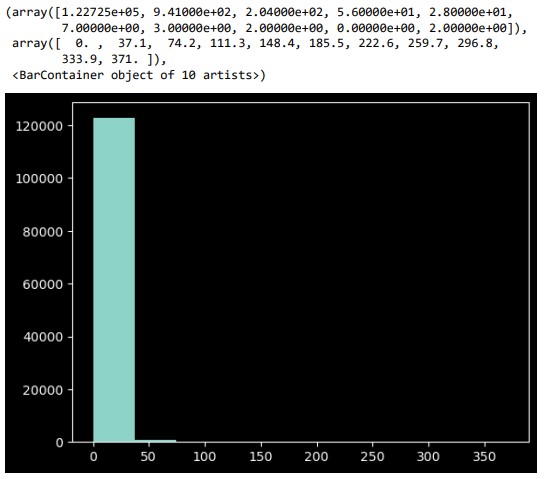
**#Histograms**

plt.hist(df['MinTemp'])

plt.hist(df['MaxTemp'])

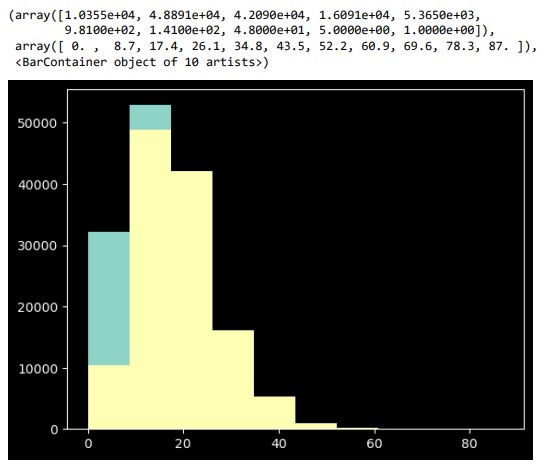


plt.hist(df['Rainfall'])



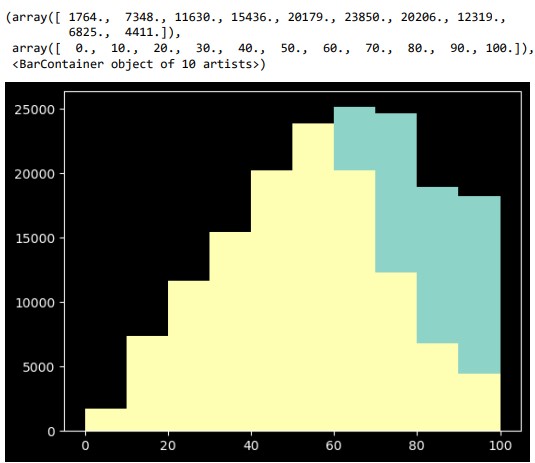
plt.hist(df['WindSpeed9am'])

plt.hist(df['WindSpeed3pm'])



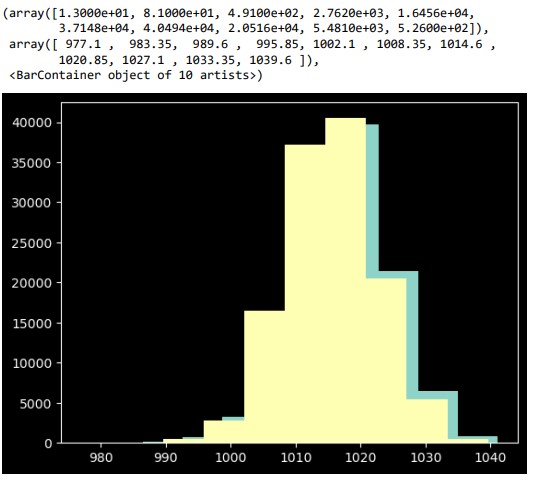
plt.hist(df['Humidity9am'])

plt.hist(df['Humidity3pm'])

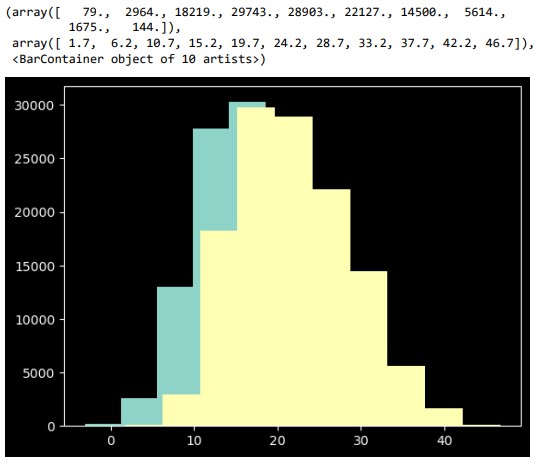


plt.hist(df['Pressure9am'])

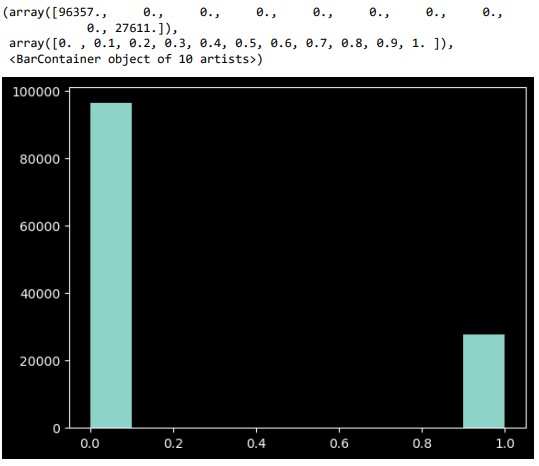
plt.hist(df['Pressure3pm'])



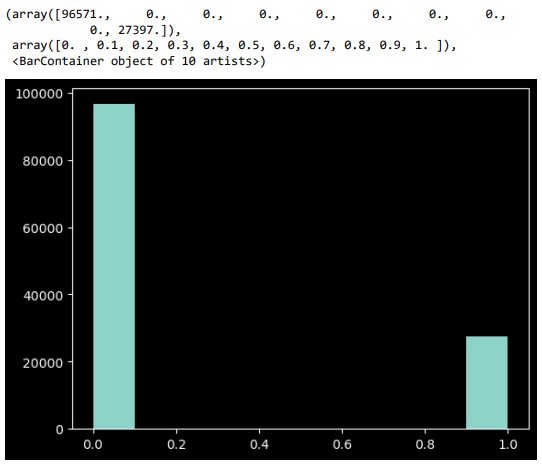
plt.hist(df['Temp9am']) plt.hist(df['Temp3pm'])



plt.hist(df['RainToday'])



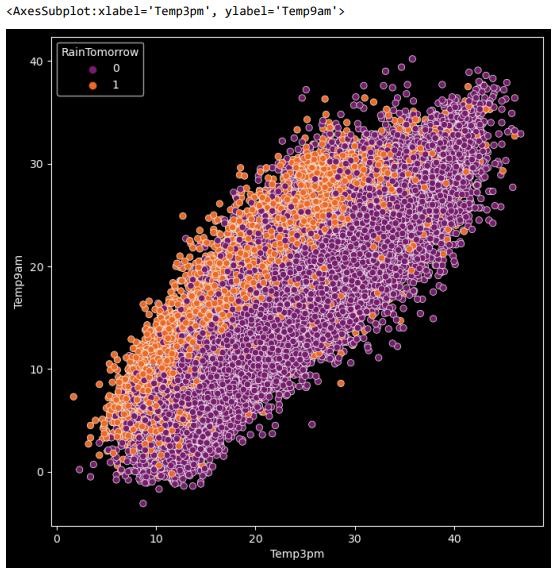
plt.hist(df['RainTomorrow'])



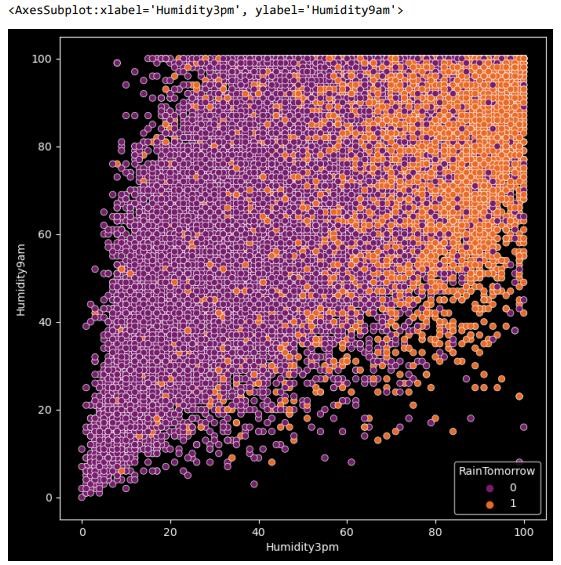
**#Scatterplots**

plt.figure(figsize = (8,8))

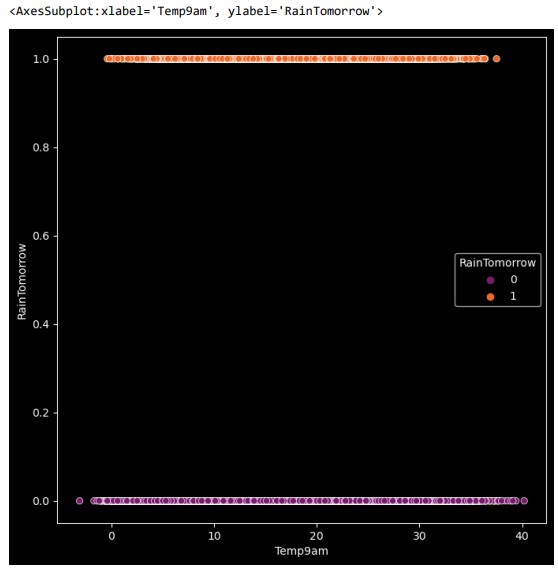
sns.scatterplot(x = 'Temp3pm', y = 'Temp9am', hue = 'RainTomorrow' , palette = 'inferno',data = df)



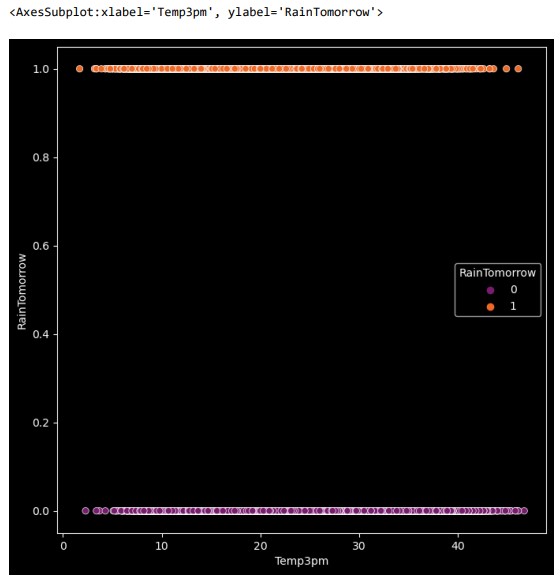
sns.scatterplot(x = 'Humidity3pm', y = 'Humidity9am', hue = 'RainTomorrow' , palette = 'inferno',data = df)



sns.scatterplot(x = 'Temp9am', y = 'RainTomorrow', hue = 'RainTomorrow' , palette = 'inferno',data = df)



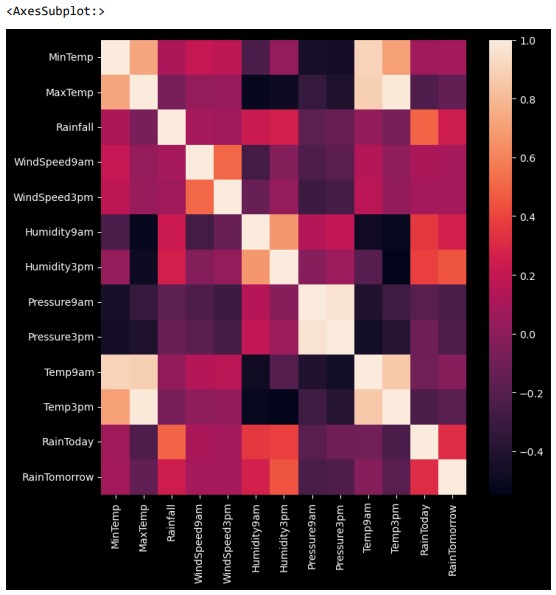
sns.scatterplot(x = 'Temp3pm', y = 'RainTomorrow', hue = 'RainTomorrow' , palette = 'inferno',data = df)



**#Heatmap**

plt.figure(figsize = (8,8))

sns.heatmap(df.corr())



**#splitting data**

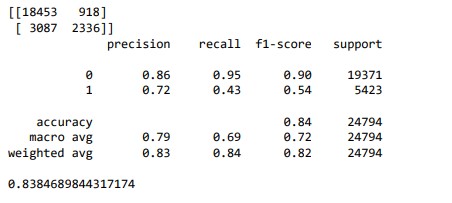
from sklearn.model\_selection import train\_test\_split

x\_train, x\_test, y\_train, y\_test = train\_test\_split(x, y, test\_size = 0.2)

**from** sklearn.metrics **import** classification\_report, confusion\_matrix, accuracy\_score

**#Linear Regression**

from sklearn.linear\_model import LogisticRegression lr = LogisticRegression() lr.fit(x\_train,y\_train) predictions = lr.predict(x\_test) print(confusion\_matrix(y\_test, predictions)) print(classification\_report(y\_test, predictions)) print(accuracy\_score(y\_test, predictions))

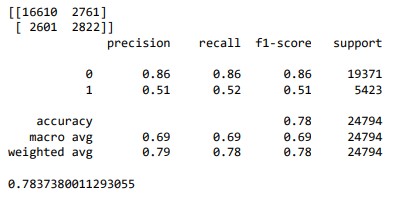


y\_pred=lr.predict(x\_test) y\_pred



**#Decision Tree**

from sklearn.tree import DecisionTreeClassifier dt=DecisionTreeClassifier() dt.fit(x\_train,y\_train) predictions=dt.predict(x\_test) print(confusion\_matrix(y\_test,predictions)) print(classification\_report(y\_test,predictions)) print(accuracy\_score(y\_test,predictions))

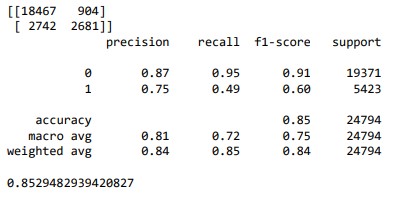


y\_pred=dt.predict(x\_test) y\_pred



**#Random Forest**

from sklearn.ensemble import RandomForestClassifier rf=RandomForestClassifier() rf.fit(x\_train,y\_train) predictions=rf.predict(x\_test) print(confusion\_matrix(y\_test,predictions)) print(classification\_report(y\_test,predictions)) print(accuracy\_score(y\_test,predictions))



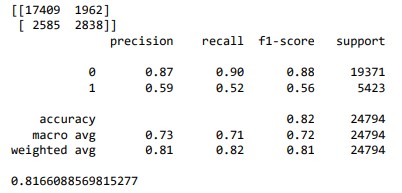
y\_pred=rf.predict(x\_test) y\_pred



**#Naive Bayes**

from sklearn.naive\_bayes import GaussianNB nb=GaussianNB()

nb.fit(x\_train,y\_train) predictions=nb.predict(x\_test) print(confusion\_matrix(y\_test,predictions)) print(classification\_report(y\_test,predictions)) print(accuracy\_score(y\_test,predictions))



y\_pred=rf.predict(x\_test) y\_pred



**13. RESULT:**

After applying above four Machine Learning algorithms – Linear Regression, Decision Tree, Random Forest and Naïve Bayes algorithms, we see that:

* Linear Regression: 0.8384689844317174

* Decision Tree : 0.7837380011293055

* Random Forest : 0 .8529482939420827

* Naïve Bayes : 0.8166088569815277

By observing the above results, we can conclude that ‘RANDOM FOREST

ALGORITHM’ has high accuracy than the other three algorithm.

**So, we conclude that Random Forest algorithm is best suited for Weather Forecasting with Machine Learning using Python**

**14. TESTING:**

Software testing is an investigation conducted to provide stakeholders with information about the quality of the product or service under test. Software testing can also provide an objective, independent view of the software to allow the business to appreciate and understand the risks of software implementation. Test techniques include, but are not limited to, the process of executing a program or application with the intent of finding software bugs (errors or other defects). Software testing can provide objective, independent information about the quality of software and risk of its failure to users and/or sponsors.

**Testing technologies:**

Testing is the process of detection errors. Testing performs a quality role for assurance and for ensuring the ability of software. The results of testing are used later on during maintenance also.

**13.1 Testing objectives:**

The main objective of testing is to uncover a host error, systematically, the minimum effort and time starting formally, we can say

* + - Test is the process of executing a program with the intent of finding an error.
    - A successful test is one that uncovers and yet undiscovered error.
    - A good test case is one that has a high probability of finding errors, if it exits.

**13.2 White box testing:**

* + - This is unit testing method where the unit will be taken at a time and tested thoroughly at a statement level to find the maximum level errors.
    - We have tested step wise every piece of code, taking care of every statement in the code.
      * Is executed at least once.
      * The white box testing is also called glass box testing.

* 1. **Black box testing:**

This testing method models a single unit and checks the unit at interface and communication with other models rather getting into detail levels. Here the model will be treated as a black box that take input and generates the output. Output of given input combinations are forwarded to other models.

* 1. **Unit testing:**

Unit testing focuses verification effort on the smallest unit of software that is the model using the detailed design and the process specifications testing is done to uncover errors within the boundary of the model all models must be successful in the unit test before the start of the integration testing. In our project unit testing involves checking each future specified in the component a component performs only small part of the functionality on the system and relies on cooperating with other part of the system.

* + - * 1. **CONCLUSION:**

With the help of more advanced techniques of machine learning, we can only try to forecast the weather conditions. But we are not sure of the results matching exactly the same as the actual values, this is because weather also depend on the increase in the number of buildings and concrete structure, changes in the vegetation, an increase in the number of vehicles and pollution level. But these factors are a machine learning problem in their own domain. Integrating all these factors would be a challenging task. Utilization of machine learning models in prediction of weather conditions in short periods of time can run on less resource-intensive machines. Prediction of variables such as rainfall possibilities, temperature, humidity etc. Evaluation of the proposed techniques and comparison of several machine learning

models in the prediction of future

* + - * 1. **BIBLIOGRAPHY:**

**REFERENCES:**

1. Introduction to Machine Learning for Atmospheric Science (Chapman &

Hall/CRC Data Mining and Knowledge Discovery Series) by Weiqiang Wang

1. Machine Learning Techniques for Weather Forecasting by Rong-Qing Zhang

1. Neural Networks and Related Methods for Weather Forecasting by Angelo Basile

1. Support Vector Machines for Time Series Forecasting by Garry Kasparov

1. Bayesian Methods for Time Series Forecasting by Christian Robert

1. Time Series Forecasting with Neural Networks by Daniele Fantini

1. Forecasting with Dynamic Regression Models by Michael P. McCracken

1. State Space Models for Time Series Analysis by James D. Hamilton

1. Time Series Analysis and Forecasting by Theodoros Lathourakis

1. Time Series Analysis: Forecasting and Control (4th Edition) by George E.P. Box

**WEB RESOURCES:**

https://www.kaggle.com/datasets/jsphyg/weather-dataset-rattle-package