**Weather Prediction Model**

Machine Learning Project Report

Antoine Ashraf 211001666

Dr. Mohamed Mahmoud El Sayeh

CBIO313

Contents

[1 Aim & Introduction: 3](#_Toc168017447)

[2 Dataset Description: 3](#_Toc168017448)

[3 Method: 3](#_Toc168017449)

[3.1 Importing Libraries and Loading Data: 3](#_Toc168017450)

[3.2 Data Preprocessing: (cleaning, handling missing values): 4](#_Toc168017451)

[3.3 Data Visualization & Analysis: 7](#_Toc168017452)

[3.4 Machine Learning Algorithms: 11](#_Toc168017453)

[3.5 Model deployment and running using streamlit: 13](#_Toc168017454)

[4 Results: 14](#_Toc168017455)

[5 Conclusion: 15](#_Toc168017456)

[6 Github Link: 15](#_Toc168017457)

[7 Video Link: 15](#_Toc168017458)

# 1 Aim & Introduction:

Weather forecasting is essential for many elements of daily life, such as agriculture, transportation, and disaster preparedness. Accurate weather forecasting enables risk-reduction strategies and proactive decision-making. Developing a predictive model that can forecast weather patterns based on significant meteorological elements is the aim of this project. Through the utilization of a comprehensive dataset containing variables such as wind speed, pressure, maximum and lowest temperatures, humidity, and rainfall, my objective is to enhance the precision of weather forecasts and provide valuable insights into seasonal variations.

# 2 Dataset Description:

Having daily recordings ranging from 2015 to 2023, the weather forecast dataset offers a thorough picture of the weather. Every entry includes the date, the minimum and maximum temperatures for the day, the amount of rainfall, and the temperatures at 9 AM and 3 PM. It also provides information on evaporation and the total number of hours of sunshine, as well as whether or not it rained that day. The dataset captures wind direction and speed at 9 AM and 3 PM, in addition to the direction and speed of the largest wind gust. Atmosphere pressure, humidity, and cloud cover values are given for the hours of 9 AM and 3 PM. This detailed dataset is valuable for analyzing past weather patterns and predicting future conditions based on historical data.

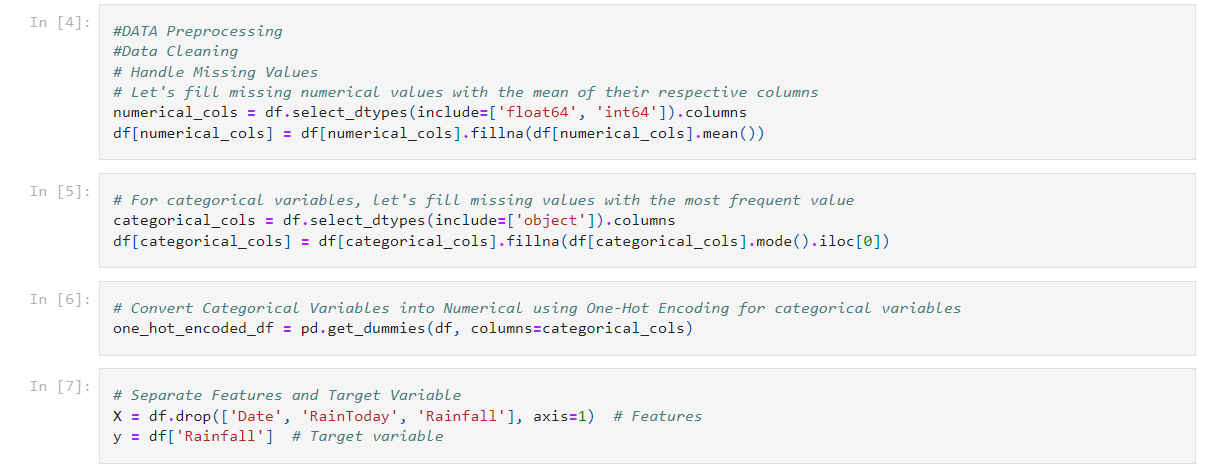
# 3 Method:

## 3.1 Importing Libraries and Loading Data:

To start the weather prediction analysis, we suppress unnecessary warnings and import essential libraries for data manipulation, visualization, and machine learning, such as Pandas, NumPy, Seaborn, Matplotlib, and Scikit-learn tools. The dataset is loaded into a PandasDataFrame. We then clean and preprocess the data by filling in missing values with SimpleImputer, encoding categorical variables using LabelEncoder, and standardizing numerical features with StandardScaler. Various machine learning models, including Logistic Regression, Decision Trees, K-Nearest Neighbors, and Support Vector Machines, are trained on the data, with GridSearchCV optimizing hyperparameters. For regression tasks, Linear Regression and Support Vector Regressor are used, while classification tasks may involve RandomForestClassifier or a Voting Classifier to enhance accuracy. Model performance is evaluated using metrics like accuracy, F1 score, mean squared error, and confusion matrix. Visualization tools such as Seaborn, Matplotlib, and Plotly are employed to create insightful plots and graphs, providing a clear and comprehensive view of the data and model performance, ensuring a robust and reliable weather prediction analysis.

## 3.2 Data Preprocessing: (cleaning, handling missing values):

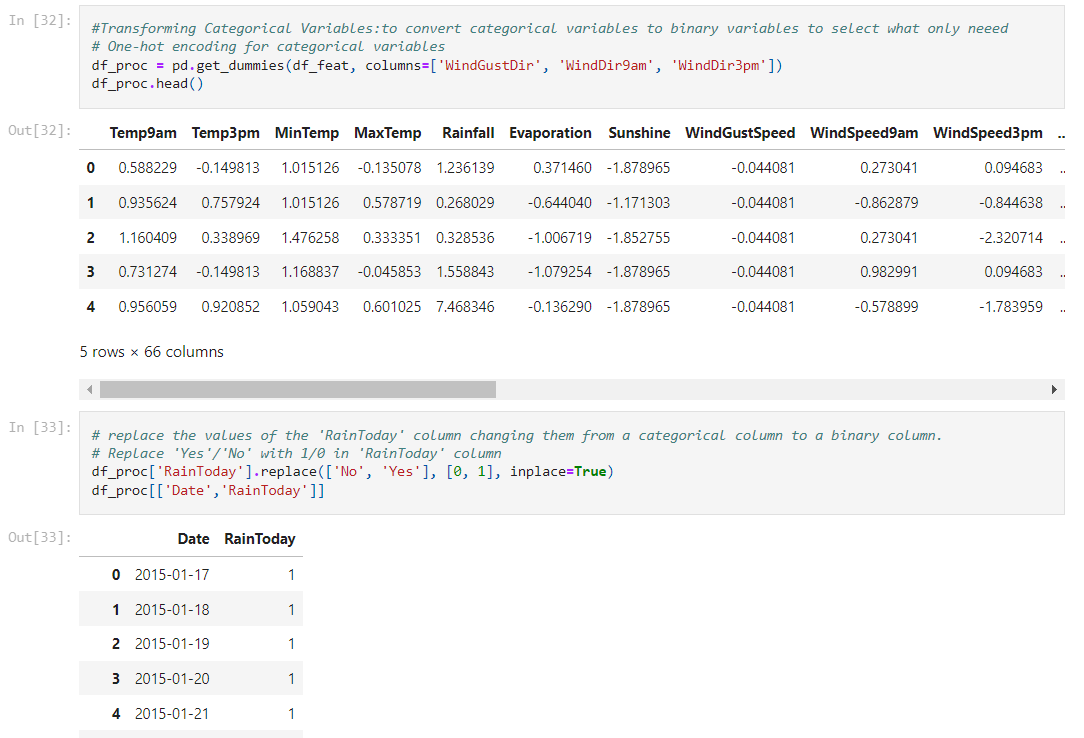
The initial step in preprocessing the weather prediction dataset is to deal with missing values. To ensure that there are no gaps in the data, fill in missing values for numerical columns with the mean of those columns. In order to ensure consistency for categorical variables, I substituted the most frequent value in each column for any missing values. After that, I used One-Hot Encoding to transform categorical variables into numerical representation. This produces binary columns for every category and prepares the data for machine learning techniques. I divided the dataset into features and the target variable after encoding. All columns are included in the features, with the exception of "Date," "RainToday," and "Rainfall," which are removed. "Rainfall" is the variable that will be predicted as the target. Preprocessing makes that the data is uniform, clean, and prepared for modelling.



In the data preprocessing phase, I first select the columns to be scaled, which include various weather features such as temperature, rainfall, wind speed, humidity, and pressure. These columns are standardized using StandardScaler to ensure uniformity in their scales, making them suitable for machine learning algorithms.



Next, we transform categorical variables into binary variables using one-hot encoding, and we replace the values in the 'RainToday' column with binary values (0 and 1) to indicate the absence or presence of rain.



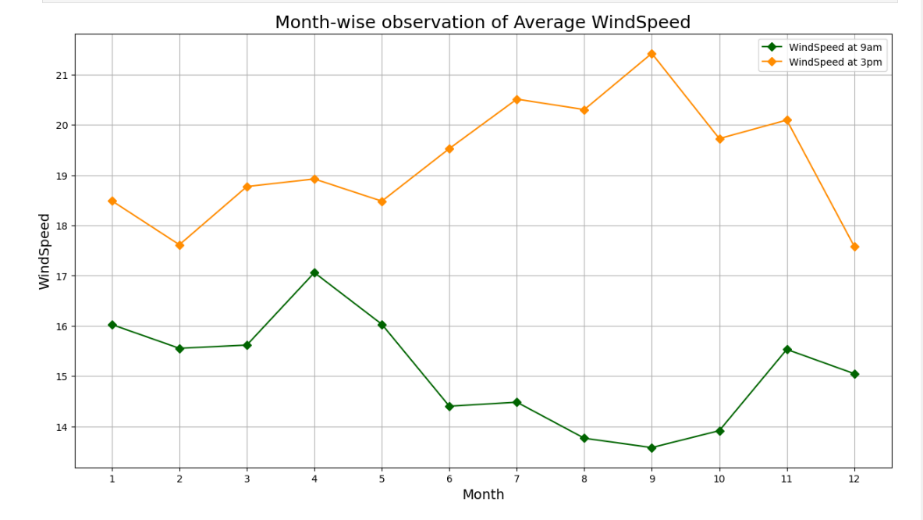
After dropping the 'Date' column, the data is converted to float type for compatibility with machine learning models.

## 3.3 Data Visualization & Analysis:

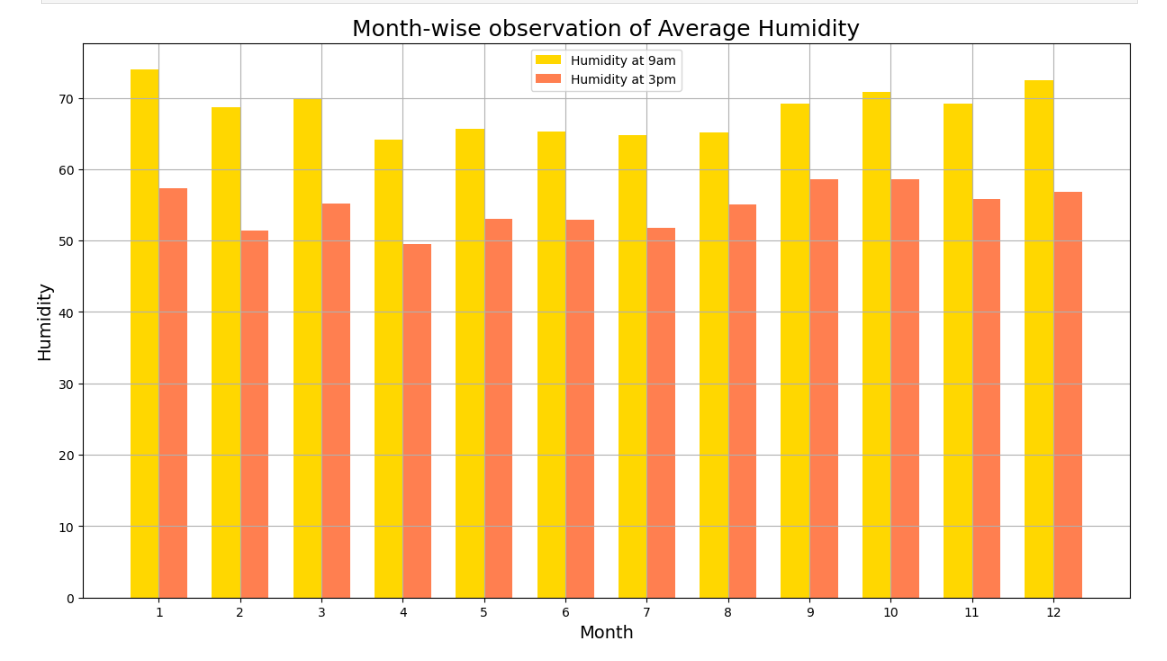
Then analyzed the feature distribution of the weather dataset, by visualizing the distribution of some statistics inside the data as:

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

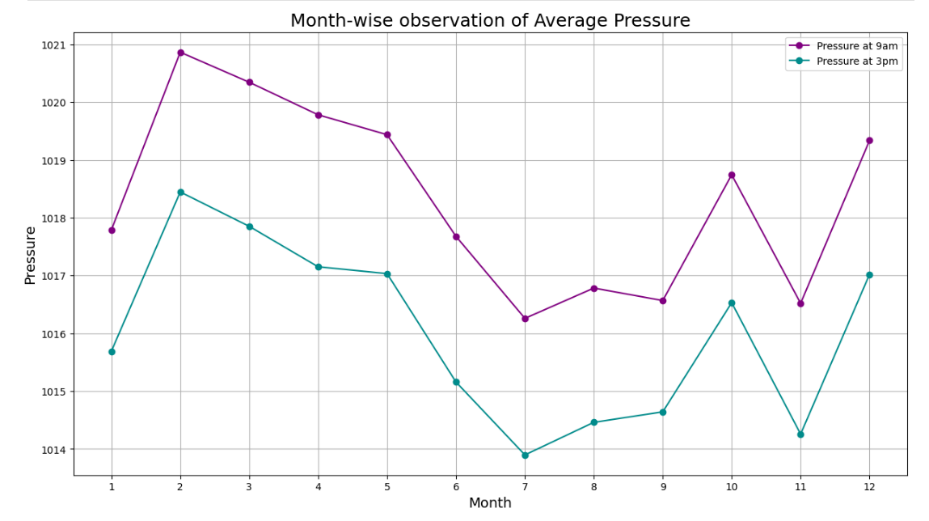
In the exploratory data analysis (EDA) phase of my weather dataset analysis, I investigated several key features. First, I analyzed the average wind speed over months, plotting observations for both 9 AM and 3 PM using a line plot to visualize any patterns or trends.



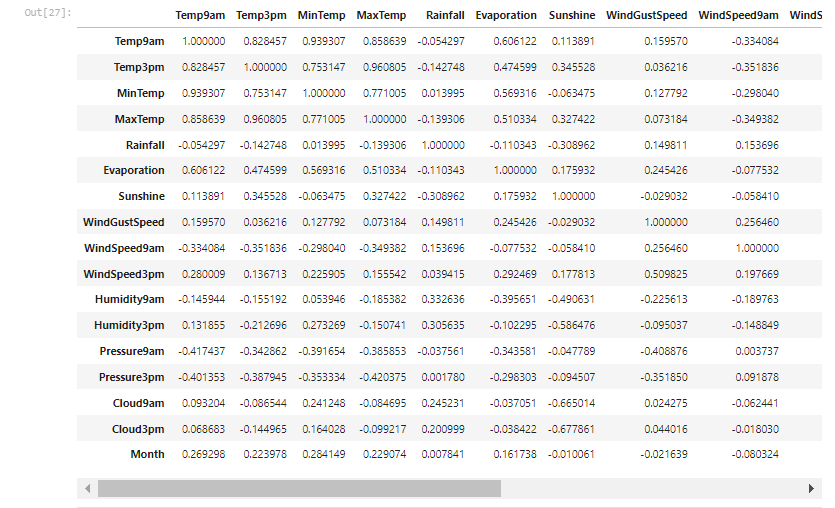
Following this, I examined the average humidity levels, plotting observations for 9 AM and 3 PM as stacked bar plots to compare humidity variations throughout the months.



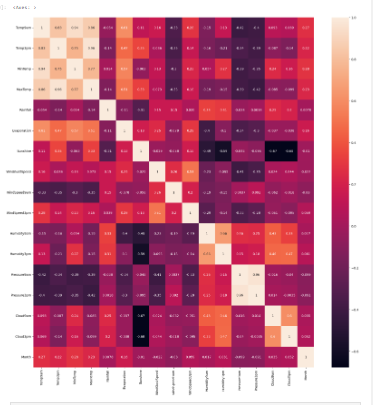
Additionally, I analyzed the average pressure over months, plotting observations for 9 AM and 3 PM using a line plot to observe any trends.



These visualizations offer insights into the seasonal variations of wind speed, humidity, and pressure. Finally, I conducted a correlation analysis.



dropping non-numeric columns and utilizing a heatmap to visualize the correlation between different features, aiding in identifying relationships between variables.



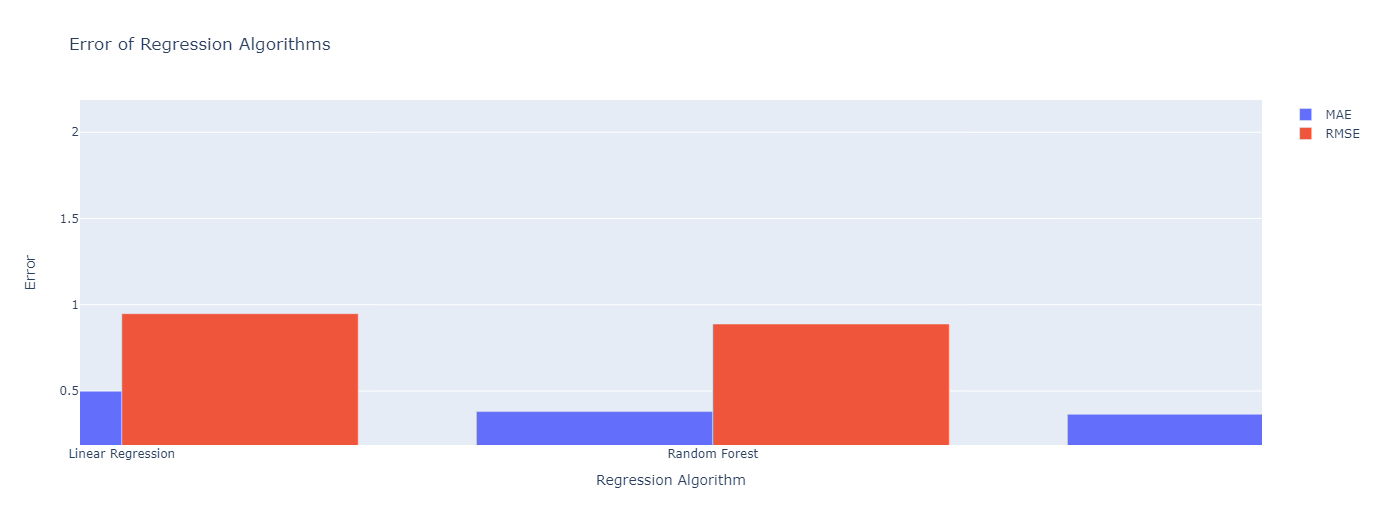
These EDA steps help in understanding the dataset's characteristics and preparing for further analysis.

## 3.4 Machine Learning Algorithms:

For rain occurrence prediction, I split the data into training and testing sets and then train several classification models, including Logistic Regression, K-Nearest Neighbors (KNN), Decision Tree, Support Vector Machine (SVM), and Random Forest. Each model is fitted with the training data and evaluated using accuracy and F1 scores on the testing set. These scores provide insights into the performance of each model in predicting rain occurrence. Finally, I visualized the accuracy scores of the classification algorithms using a bar plot to compare their performance visually. This comprehensive analysis aids in selecting the most suitable model for rain occurrence prediction based on its accuracy and F1-score.



For rainfall amount prediction, Features are selected while excluding the target variable 'Rainfall', and the data is split into training and testing sets using a test size of 35% and a random state of 101 to ensure reproducibility. I then trained three regression models: Linear Regression, Random Forest Regression, and Support Vector Machine (SVM) Regression. These models are fitted with the training data and evaluated using mean absolute error (MAE) and root mean squared error (RMSE) on the testing set. These metrics provide insights into the accuracy of rainfall amount predictions for each model. For the Linear Regression model, I calculated the MAE and RMSE, indicating the average magnitude and spread of errors in rainfall predictions. Similarly, for the Random Forest Regression and SVM Regression models. Finally, I visualized the errors of the regression algorithms using a bar plot to compare their performance visually.

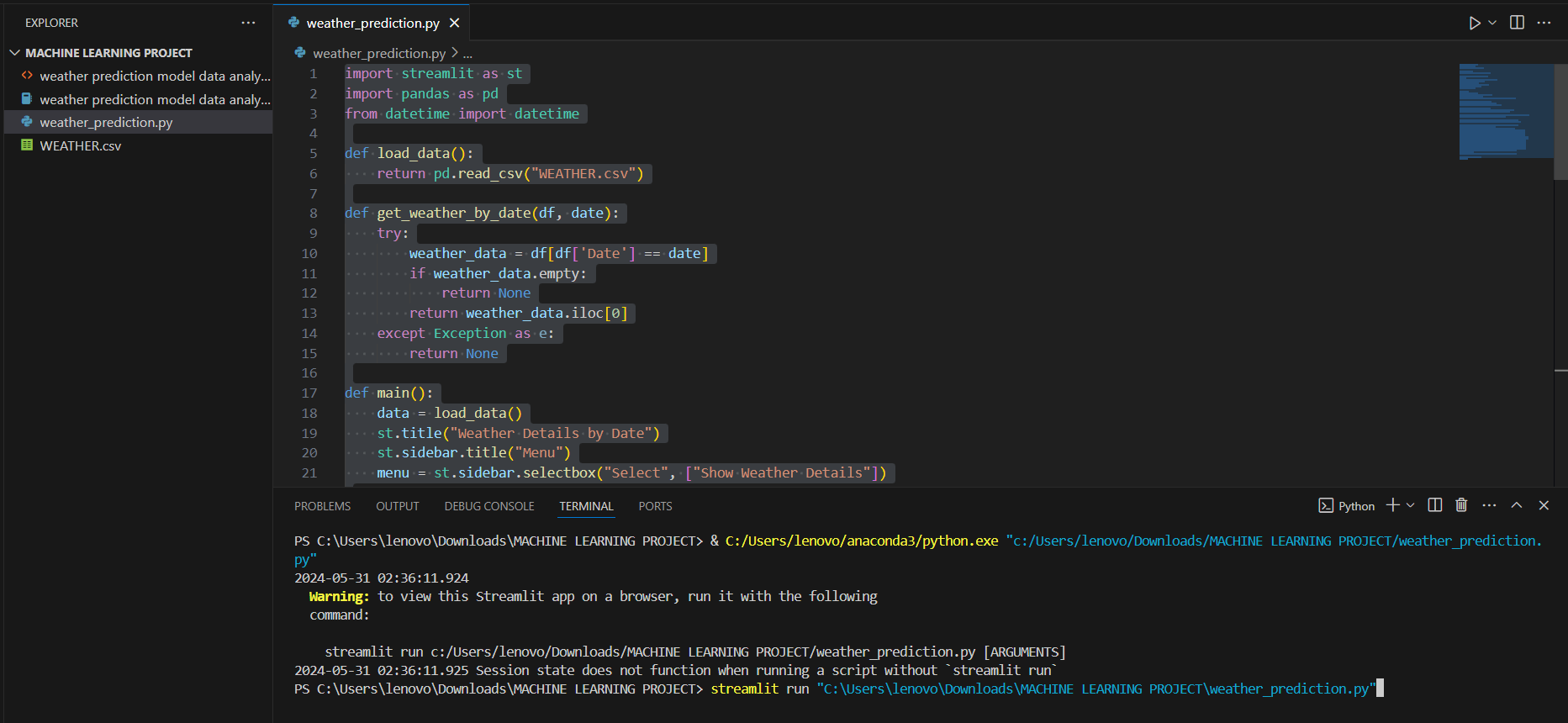


Same thing was made for the temperature, by calculating the average temperature of a day using the mean of 'Temp9am', 'Temp3pm', 'MinTemp', and 'MaxTemp' columns and creating a new feature 'AvgTemp'. training three regression models. evaluating the performance of each model. calculating the MAE and RMSE, representing the average magnitude and spread of errors in predicting the average temperature of a day. Finally, visualizing the errors of the regression algorithms using a bar plot to compare their performance visually. This comprehensive analysis aids in selecting the most suitable regression model for predicting the average temperature of a day based on its MAE and RMSE scores.



## 3.5 Model deployment and running using streamlit:

In the Streamlit app for weather details by date, I first load the weather data from a CSV file, and importing necessary libraries. Then, the get\_weather\_by\_date(df, date) function retrieves weather details for a specified date from the DataFrame. This function checks if data exists for the given date and returns the details if found, otherwise returning None.



# 4 Results:

Users can input a date using a date picker widget. Clicking the "Show Details" button triggers a function that retrieves weather details for the selected date. The date input is converted to a string in the 'MM/DD/YYYY' format and compared with the 'Date' column in the DataFrame, which has been converted to datetime objects for comparison. If weather details are found for the selected date, they are displayed, including temperatures at 9 AM and 3 PM, minimum and maximum temperatures, wind speeds, humidity levels, pressure readings, cloud cover, and rainfall. If no data is available for the selected date, a message indicating this is displayed. This app provides a simple and user-friendly interface for accessing weather details for specific dates.

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

# 5 Conclusion:

In conclusion, this project aimed to improve weather forecasting by developing predictive models based on a comprehensive dataset spanning from 2015 to 2023. By analyzing significant meteorological elements such as wind speed, pressure, temperature, humidity, and rainfall. Through extensive data preprocessing, exploratory data analysis, and the implementation of various machine learning algorithms for both classification and regression tasks, I have gained insights into weather patterns and built models capable of predicting rain occurrence, rainfall amount, and average daily temperature. The results of my analysis demonstrate the feasibility of using machine learning techniques to improve weather forecasting accuracy. Additionally, the deployment of a user-friendly Streamlit app allows for easy access to weather details for specific dates.

# 6 Github Link:

[**https://github.com/AntoineAsh/Weather-Prediction-Model**](https://github.com/AntoineAsh/Weather-Prediction-Model)

# 7 Video Link:

[Machine Learning Project-20240531\_142519-Meeting Recording.mp4](https://nileuniversity-my.sharepoint.com/:v:/g/personal/a_ashraf2166_nu_edu_eg/EU23btKzIedNrqiiUXSyBgYB_rk-Od6VwL0MAT4JI3jp7Q?e=fQsMZc&nav=eyJyZWZlcnJhbEluZm8iOnsicmVmZXJyYWxBcHAiOiJTdHJlYW1XZWJBcHAiLCJyZWZlcnJhbFZpZXciOiJTaGFyZURpYWxvZy1MaW5rIiwicmVmZXJyYWxBcHBQbGF0Zm9ybSI6IldlYiIsInJlZmVycmFsTW9kZSI6InZpZXcifX0%3D)