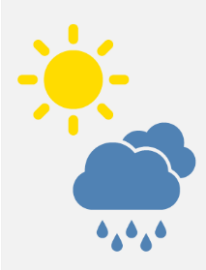
Bahria University
Discovering Knowledge

Fall 2022
Department of Computer Science
Bahria University, Karachi Campus

INTRODUCTION TO DATA SCIENCE LAB [CSL-487]



Karachi Weather Data Analysis (Rainfall Prediction)

SEMESTER PROJECT

Maximum Marks: 30

Submission Due Date: 17th January, 2023

Sr.no	Name	Enrolment	Semester
01	Ibraaheem Saeed Akbar	02-134201-100	**6-B**
02	Muhammad Zaheeruddin Halepoto	02-134201-066	**6-B**
03	Danial Zubair	02-134201-026	**6-B**

Name	Designation
Soomal Fatima	**Course Instructor**
Salas Akbar	**Lab Engineer**

Acknowledgement

We would like to express our utmost gratitude to both our instructors, Miss Soomal Fatima and Miss Salas Akbar for their mentorship and assistance for the continuous 16 weeks of our classes and labs. They had always been supportive with any queries we had and offered their help in any situation related to our work. We are also grateful that they gave us the opportunity to work on this project. Working on our project provided us a way to implement the skills and concepts we learned in our lectures and labs.

Contents

1. Chapter 1

1.1. Problem Statement:

- The Pakistan Meteorological Department has provided us a dataset of the weather conditions in Karachi for the year 2022. The dataset includes various weather parameters such as temperature, wind speed, air pressure, humidity, and rainfall.

- The goal of this project is to analyze the provided dataset and use machine learning techniques to make predictions about future weather patterns in Karachi, specifically with regards to rainfall.

- The project aims to identify patterns in the data and use them to make accurate predictions about future weather conditions in the city, which can help in making important decisions related to agriculture, construction, and disaster management. Additionally, the project will also explore the relationship between different weather parameters and how they affect rainfall in the city.

2. Chapter 2

2.1. Literature Review

1. **Machine Learning Applied to Weather Forecasting** (Mark Holmstrom, Dylan Liu, Christopher Vo Stanford University, Dated: December 15, 2016)

   Key Takeaways:
   This paper explores the application of machine learning techniques to weather forecasting to potentially generate more accurate weather forecasts for large periods of time. The scope of the paper is restricted to forecasting the maximum temperature and the minimum temperature for seven days, given weather data for the past two days. A linear regression model and a variation of a functional regression model were used, with the latter able to capture trends in the weather. Both models were outperformed by professional weather forecasting services, although the discrepancy between the models and the professional ones diminished rapidly for forecasts of later days. The linear regression model outperformed the functional regression model, suggesting that two days were too short for the latter to capture significant weather trends.

2. **Forecasting Severe Weather with Random Forests** (Aaron J. Hill, Gregory R. Herman, and Russ S. Schumacher, Department of Atmospheric Science, Colorado State University, Fort Collins, Colorado, Manuscript received 15 October 2019, in final form 10 March 2020)

   Key Takeaways:
   This paper focuses on using random forest (RF) models to make probabilistic predictions of severe weather across the contiguous United States (CONUS) at Days 1-3, with separate models for tornado, hail, and severe wind prediction at Day 1. The RFs are trained using nine years of historical forecasts from NOAA's GEFS/R ensemble and input predictors include fields associated with severe weather prediction, such as CAPE, CIN, and wind shear. The RFs are found to produce calibrated probabilistic forecasts that slightly underperform the Storm Prediction Center's (SPC) outlooks at Day 1, but significantly outperform them at Days 2 and 3.

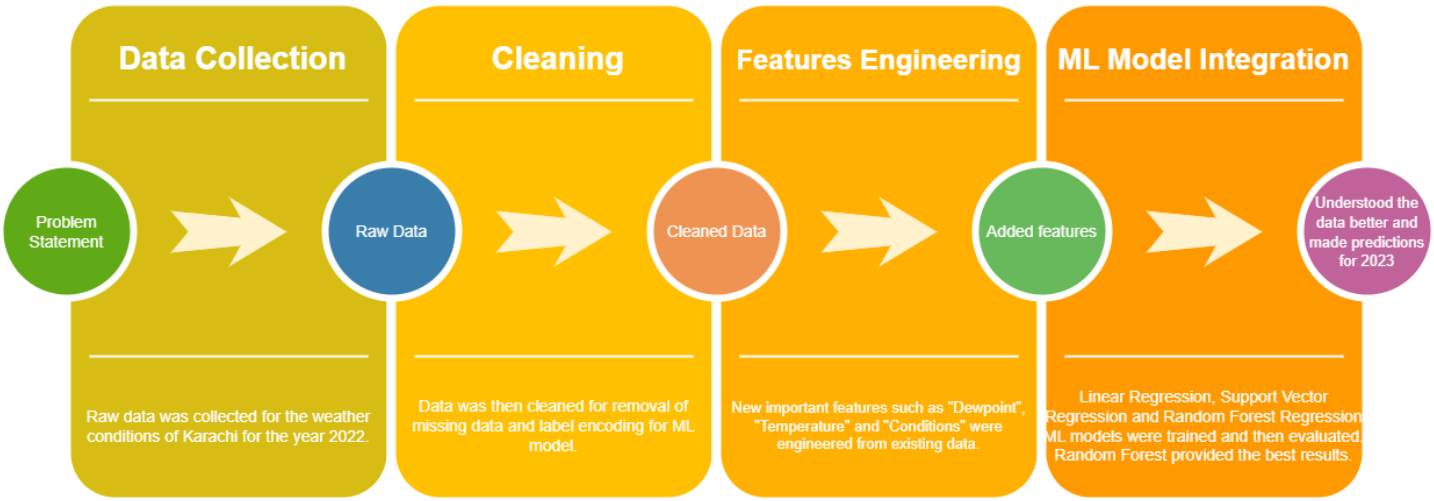
3. **Viable Forecasting Monthly Weather Data Using Time Series Method** (Ramzan Soomro1,*, Saghir Pervaiz Ghauri2, Azhar Ali Marri3, Sergij Vambol4,*, Hoang Thi Dung5, Nazish Manzoor6, Shella Bano7, Sana Shahid8, Asadullah9, Ahmed Farooq9, Yurii Lutsenko101 Quaid-e-Millat Government Degree College Liaqatabad Karachi, Pakistan2 Faculty of Business Administration, Commerce & Economics, Jinnah University for Women, Karachi, Pakistan 3 Department of Statistics, University of Baluchistan, Quetta, Pakistan4Life Safety Department, State Biotechnological University, Kharkiv, Ukraine5 Vietnam National University of Forestry, Xuan Mai, Ha Noi, Vietnam6 Kohat University of Science and Technology, Pakistan 7 Department of Geology, University of Karachi, Pakistan8 Media and Communication Studies Department, Sindh Madressatul Islam University, Karachi 74000, Pakistan 9 Department of Environmental Sciences, Sindh Madressatul Islam University, Karachi 74000, Pakistan 10 Institute of Public Administration and Research in Civil Protection, Kyiv, Ukraine)

   Key Takeaways:

   This paper aims to assess the forecast values of weather parameters using three-time series methods such as Decomposition of time series, Autoregressive (AR) model with seasonal dummies, and Autoregressive moving average (ARMA) /Autoregressive Integrated moving average (ARIMA) model. Stationarity is measured through the Augmented Dickey-Fuller test, and the reliability of the forecast results is examined through the goodness of fit test. The best fit model is chosen based on performance measures such as Root Mean Squire Error, Mean Absolute Error, and Mean Absolute Percentage Error. The research is focused on assessing the impact of climate change on weather in Pakistan and finding appropriate models to forecast it.

3. Chapter 3

3.1. Methodology:



The data science process was used to conduct this project. The process includes the following steps:

1. **Problem Definition:** The problem statement was to analyze the weather data of Karachi in 2022 and make predictions for the year 2023.
2. **Data Collection:** The weather data of Karachi in 2022 was obtained from the Pakistan Meteorological Department.
3. **Data Cleaning:** The data was cleaned by removing any missing values and converting non-numeric values to numeric values.
4. **Data Exploration:** The data was explored by creating visualizations to better understand the relationships between the different variables.
5. **Feature Engineering:** New features were created from the existing data, such as Dewpoint and Conditions.
6. **Model Selection:** Machine learning models such as Linear Regression and Random Forest Regressor and Support Vector Regression were used for the analysis.
7. **Model Evaluation:** The performance of the models was evaluated using metrics such as accuracy, mean squared error, and R-squared.
8. **Model Optimization:** The models were optimized by tuning the hyperparameters.
9. **Results:** The results of the analysis were reported and used to make predictions for the year 2023.

10. **Conclusion:** The project concluded that the weather in Karachi in 2022 was primarily sunny, with occasional rain, wind, and storms. The random forest regressor model was found to be the best model for predicting rainfall in 2023.

4. Chapter 4

4.1. Code Snippet

Training/Testing Code:

Linear Regression Model

```
from sklearn.linear_model import LinearRegression
from sklearn.model_selection import train_test_split
from sklearn.metrics import mean_absolute_error

# define the predictor and target variables
X = df[['Min_temp', 'Max_temp', 'Wind_speed', 'Wind_direction', 'Air Pr
essure', 'Humidity', 'Clouds', 'Dew_point', 'Conditions']]
y = df['Rainfall']

# split the data into training and test sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2
, random_state=31)

# fit a linear regression model to the training data
lin_reg = LinearRegression()
lin_reg.fit(X_train, y_train)

# make predictions on the test data
y_pred = lin_reg.predict(X_test)

# calculate the mean absolute error of the model
mae = mean_absolute_error(y_test, y_pred)
print("Mean Absolute Error: {:.2f}".format(mae))
```

Random Forest Regressor Model:

```
from sklearn.ensemble import RandomForestRegressor
from sklearn.model_selection import train_test_split
from sklearn.metrics import mean_squared_error, r2_score
# define the predictor and target variables
X = df[['Min_temp', 'Max_temp', 'Wind_speed', 'Wind_direction', 'Air Pr
essure', 'Humidity', 'Clouds', 'Dew_point', 'Conditions']]
y = df['Rainfall']


# split the data into training and test sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2
, random_state=32)

# fit a random forest regressor model to the training data
rf_reg = RandomForestRegressor(n_estimators=100)
rf_reg.fit(X_train, y_train)

# make predictions on the test data
y_pred = rf_reg.predict(X_test)

# calculate the mean absolute error of the model
mae = mean_absolute_error(y_test, y_pred)
print("Mean Absolute Error: {:.2f}".format(mae))
```

Support Vector Regressor Model:

```
from sklearn.svm import SVR
from sklearn.model_selection import train_test_split
from sklearn.metrics import mean_squared_error

# define the predictor and target variables
X = df[['Min_temp', 'Max_temp', 'Wind_speed', 'Wind_direction', 'Air Pr
essure', 'Humidity', 'Clouds', 'Dew_point', 'Conditions']]
y = df['Rainfall']

# split the data into training and test sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2
, random_state=42)

# fit a SVM model to the training data
svm = SVR(kernel='poly', degree=2)
svm.fit(X_train, y_train)

# make predictions on the test data
y_pred = svm.predict(X_test)

# calculate the mean squared error of the model
mae = mean_absolute_error(y_test, y_pred)
print("Mean Absolute Error: ", mae)
```

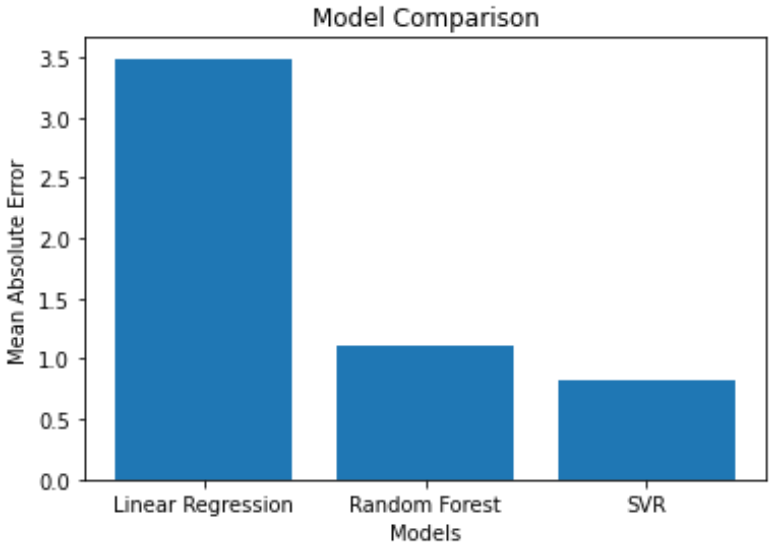


Fig. Models performance comparison chart

4.2. Features Engineering:

From our dataset, we were able to engineer new features such as:

- Dewpoint
  - Dewpoint is calculated by the Magnus Formula, which takes into account the temperature, humidity and air pressure.
  - We have created a function, which calculates dewpoint based on these parameters, and created a new column as Dew_point in our dataset
  - We also only had min and max temperatures, so we calculated the average temperature for each day and used it for dewpoint calculation.
  - The numbers in the equations are constant, **RH**(Relative Humidity) and **T**(Average Temperature) are used to calculate it.
  - Formula for dewpoint is as follows:

$$\text{Dew Point} = \frac{243.12 \times \left\{ \ln\left(\frac{RH}{100}\right) + \frac{17.62 \times T}{243.12 + T} \right\}}{17.62 - \left\{ \ln\left(\frac{RH}{100}\right) + \frac{17.62 \times T}{243.12 + T} \right\}}$$

Coding Implementation:

```
from numpy import log, e

def calculate_dewpoint(temperature, humidity):
    dewpoint = (243.12 * ((log((humidity/100) * (6.112 * (e ** ((17.62
* temperature) / (243.04 + temperature)))))))) / (17.62 - log((humidity
/100) * (6.112 * (e ** ((17.62 * temperature) / (243.04 + temperature))
))))
    return dewpoint

#calculate average temperature because we have min and max
df["Temperature"] = (df["Min_temp"] + df["Max_temp"])/2

#calculate dewpoint
df["Dew_point"] = calculate_dewpoint(df["Temperature"], df["Humidity"])
```

- Temperature
  - We calculated the average temperature for each day by our Min_temp and Max_temp columns.

- Conditions
  - We are going to classify each day in our dataset into 5 different weather conditions.
  - The weather conditions include "Sunny", "Windy", "Rainy", "Cloudy" and "Stormy".
  - We are classifying our days based on the values of "Rainfall", "Wind_speed", "Clouds" and "Air Pressure".

Coding Implementation:

```
# Create a new column 'Conditions'
df['Conditions'] = ''

# Classify each day as one of the conditions
for i in range(len(df)):
    if (df['Rainfall'][i] > 0):
        df.at[i, 'Conditions'] = 'Rainy'
    elif (df['Wind_speed'][i] > 15):
        df.at[i, 'Conditions'] = 'Windy'
    elif (df['Clouds'][i] > 6):
        df.at[i, 'Conditions'] = 'Cloudy'
    elif (df['Air Pressure'][i] < 1000):
        df.at[i, 'Conditions'] = 'Stormy'
    else:
        df.at[i, 'Conditions'] = 'Sunny'
```

4.3. Observations:

1.

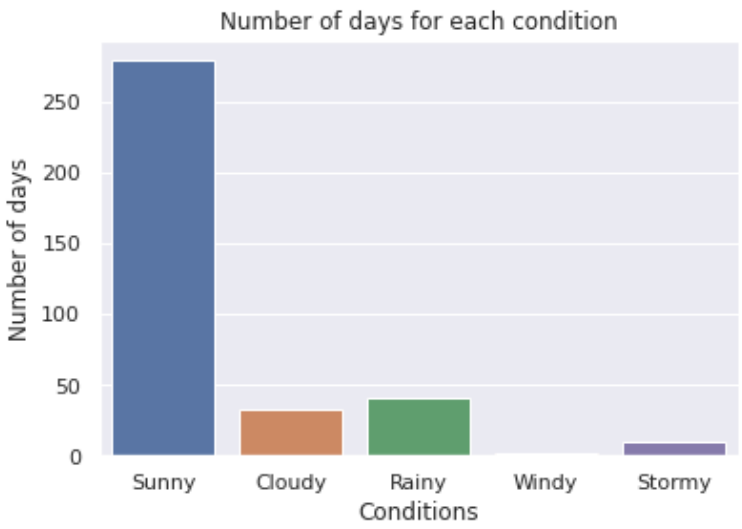


Fig 1. Number of days for each weather condition

- In the year 2022, majority of the days were sunny.

2.

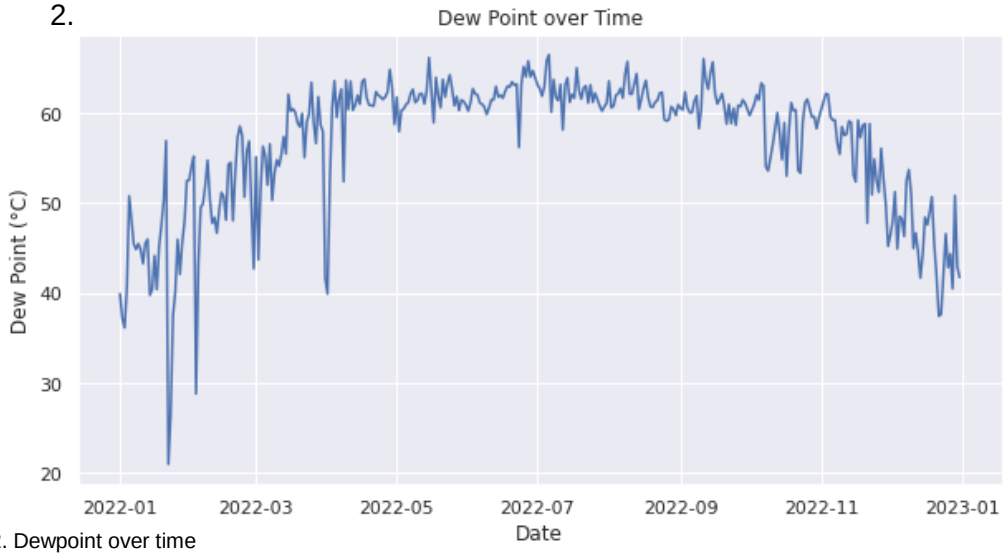


Fig 2. Dewpoint over time

- The sudden drop in dew point readings between January and March, and between March and May that we are observing in our graph could indicate a change in weather patterns during those periods.
- Dew point temperature is a measure of the amount of moisture in the air, so a drop-in dew point temperature typically indicates that the air is becoming drier.

3.

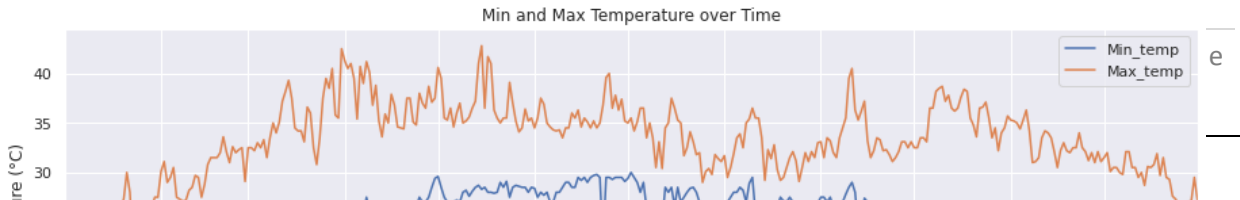


Fig 3. Min and Max temperatures throughout 2022:

- April and June were the warmest months of 2022

4.

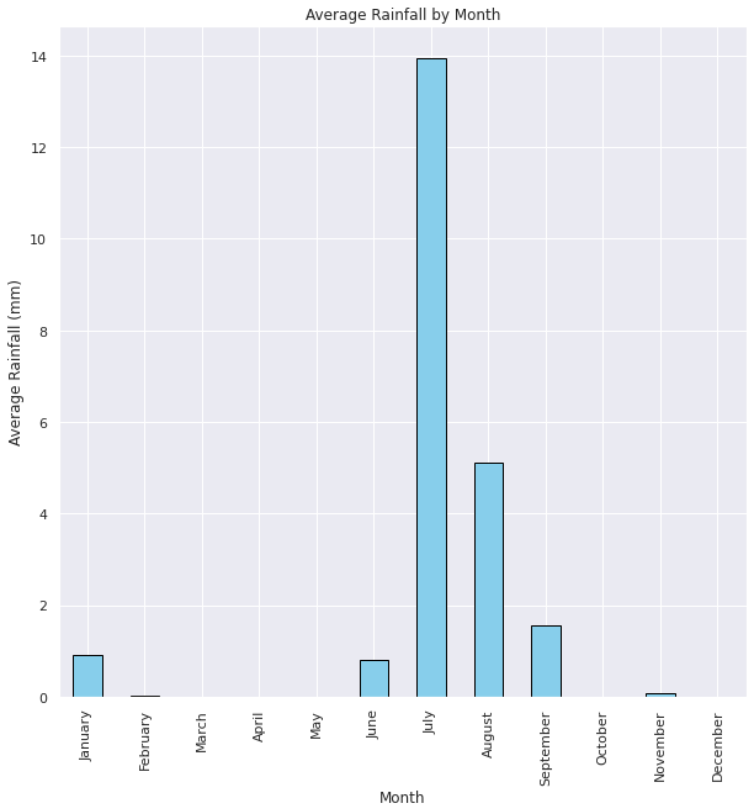


Fig 4. Average rainfall by each month in 2022

5.

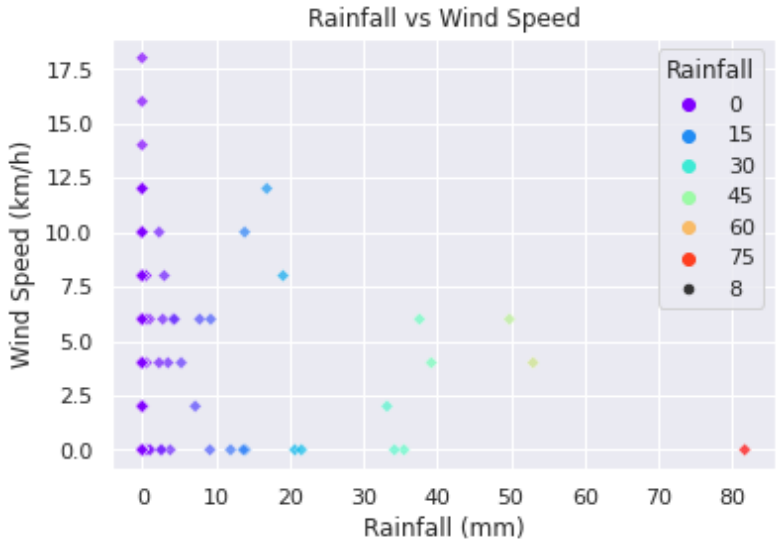


Fig 5. Rainfall and windspeed comparison

- As rainfall increases, windspeed decreases. Hence, we can say that wind speed is inversely proportional to rainfall.

5. Chapter 5

5.1. Prediction Model/Conclusion:

Using our support vector regressor model that we trained on the 2022 data, we were able to create a prediction model which could predict amount of rainfall in the month of July 2023.

In our code, we are able to change variables and check the predicted amount of rainfall in July 2023.

Code:

```
#support vector regressor object
svm = SVR(kernel='poly', degree=2)


#fit the model with our previous data
svm.fit(X_train, y_train)

#create a new dataset for July 2023
july_2023 = pd.DataFrame({'Min_temp': [30], 'Max_temp': [40], 'Wind_spe
ed': [15],
'Wind_direction': [270], 'Air Pressure': [1009], 'Humidity': [60],
'Clouds': [4], 'Dew_point': [15], 'Conditions': [1]})


#using the trained model to make predictions
july_2023_pred = rf_reg.predict(july_2023)

print("With these variables, the rainfall in july 2023 will be",july_20
23_pred,"(mm)")
```

Output:

```
With these variables, the rainfall in july 2023 will be [7.529] (mm)
```

5.2. Future Work

We have a few ideas as to what could be some future work for this project:

1. We could try incorporating more weather variables, such as precipitation, visibility, and UV index, to improve the accuracy of the models.
2. We could experiment using more advanced machine learning techniques, such as neural networks, to make predictions.
3. We could collect more weather data from multiple cities in Pakistan to build a more comprehensive model.
4. We could collect more historical data to improve accuracy of our predictions.

5.3. References

1. **Youtube**: Weather Prediction With Python And Machine Learning [W/Code]
2. **Website**: Regional Meteorological Center Karachi
3. **Youtube**: Seaborn and Matplotlib tutorial
6. **Paper** I: **Machine Learning Applied to Weather Forecasting** (Mark Holmstrom, Dylan Liu, Christopher Vo Stanford University, Dated: December 15, 2016)
7. **Paper II**: **Forecasting Severe Weather with Random Forests** (Aaron J. Hill, Gregory R. Herman, and Russ S. Schumacher, Department of Atmospheric Science, Colorado State University, Fort Collins, Colorado, Manuscript received 15 October 2019, in final form 10 March 2020)
4. **Paper III**: **Viable Forecasting Monthly Weather Data Using Time Series Method** (Ramzan Soomro1,*, Saghir Pervaiz Ghauri2, Azhar Ali Marri3, Sergij Vambol4,*, Hoang Thi Dung5, Nazish Manzoor6, Shella Bano7, Sana Shahid8, Asadullah9, Ahmed Farooq9, Yurii Lutsenko101 Quaid-e-Millat Government Degree College Liaqatabad Karachi, Pakistan2 Faculty of Business Administration, Commerce & Economics, Jinnah University for Women, Karachi, Pakistan 3 Department of Statistics, University of Baluchistan, Quetta, Pakistan4Life Safety Department, State Biotechnological University, Kharkiv, Ukraine5 Vietnam National University of Forestry, Xuan Mai, Ha Noi, Vietnam6 Kohat University of Science and Technology, Pakistan 7 Department of Geology, University of Karachi, Pakistan8 Media and Communication Studies Department, Sindh Madressatul Islam University, Karachi 74000, Pakistan 9 Department of Environmental Sciences, Sindh Madressatul Islam University, Karachi 74000, Pakistan 10 Institute of Public Administration and Research in Civil Protection, Kyiv, Ukraine)