

# **IMPLEMENTATION OF POLYNOMIAL REGRESSION USING WEATHER DATASET**

**A PROJECT REPORT FOR INDUSTRIAL INTERNSHIP**

**Submitted By**

**SOUMALYA BHATTACHARYYA**

**RAHUL CHAKRABORTY**

**SOUMYADEEP BANERJEE**

**RITAM KOLEY**

**DEVPARNO GHOSAL**

**SUBHOJIT BAKSHI**

**in partial fulfillment for the award of the degree**

*of*

**BACHELOR OF TECHNOLOGY**

**IN**

**ELECTRONICS AND COMMUNICATION ENGINEERING**

**TECHNO MAIN SALLAKE**

**Under the Guidance of**

**TATHAGATA CHATTERJEE**

**Project Carried Out in association with**

**ARDENT COMPUTECH PVT. LTD.**




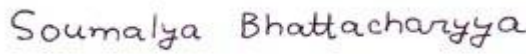
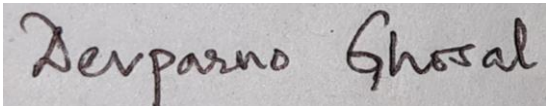

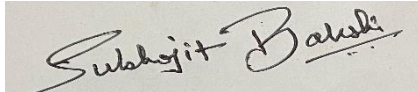
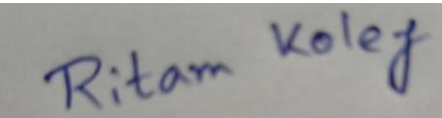
**1. Title of Project:** Implementation of Polynomial Regression using weather dataset

**2. Project Members:** Soumalya Bhattacharyya, Subhojit Bakshi, Soumyadeep Banerjee, Ritam Koley, Devparno Ghosal and Rahul Chakraborty

**3. Name and Address of the Guide:** Tathagata Chatterjee, Agarpara-Kolkata

**4. Educational Qualification of the Guide:** M.Tech, B.Tech

**5. Working / Training experience of the Guide:** 3 Years

1. 
2. 
3. 
4. 
- 5) 
- 6) 

Signatures of Team Members

Signature of Approval

Date: 05/07/2023

Date:

## PROJECT RESPONSIBILITY FORM

GROUP NO.	SL.NO.	NAME OF MEMBER	RESPONSIBILITY
1	1	Rahul Chakraborty	Model Implementation
	2	Soumalya Bhattacharyya	Data Collection
	3	Subhojit Bakshi	Data Cleaning
	4	Devparno Ghosal	Model Implementation
	5	Soumyadeep Banerjee	Data Collection
	6	Ritam Koley	Data Cleaning

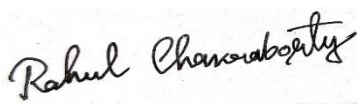
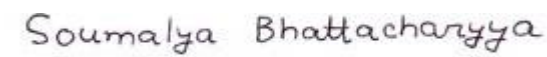
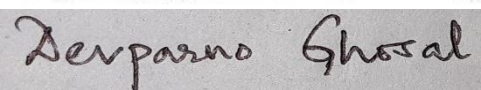

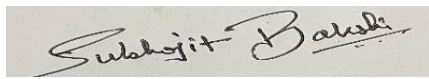
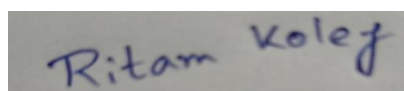
Each group member must participate in project development and developing the ideas for the required elements. Individual group members will be responsible for completing tasks which help to finalize the project and the performance. All group members must be assigned a task.

Date: 05/27/2023

Name of the Students

- 1.Devparno Ghosal
- 2.Rahul Chakraborty
- 3.Subhojit Bakshi
- 4.Soumalya Bhattacharyya
- 5.Soumyadeep Banerjee
- 6.Ritam Koley

Signatures of the students

- a. 
- b. 
- c. 
- d. 
- e. 
- f. 

## DECLARATION


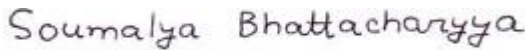
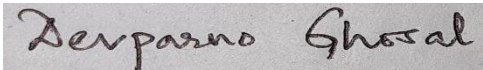

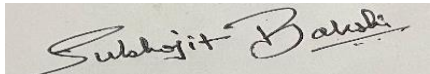
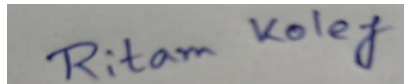
We hereby declare that the project work being presented in the project proposal entitled **“Implementation of Polynomial Regression using weather dataset ”** in partial fulfilment of the requirements for the award of the degree of **BACHELOR OF TECHNOLOGY IN ELECTRONICS AND COMMUNICATION ENGINEERING** at **Ardent Computech Pvt Ltd** is an authentic work carried out under the guidance of **MR. TATHAGATA CHATTERJEE**. The matter embodied in this project work has not been submitted elsewhere for the award of any degree of our knowledge and belief.

Date: 05/07/2023

Name of the Students

- 1.Soumyadeep Banerjee
- 2.Ritam Koley
- 2.Devparno Ghosal
- 3.Subhojit Bakshi
- 4.Soumalya Bhattacharyya
- 5.Rahul Chakraborty

Signature of the students

- a. 
- b. 
- c. 
- d. 
- e. 
- f. 

## **CERTIFICATE**

This is to certify that this proposal of minor project entitled “**Implementation of Polynomial Regression using weather dataset**” is a record of bona fide work, carried out by 1)SoumalyaBhattacharyya 2)Subhojit Bakshi 3)Soumyadeep Banerjee 4)Ritam Koley 5)Devparno Ghosal and 6)Rahul Chakraborty under my guidance at **ARDENT COMPUTECH PVT LTD**. In my opinion, the report in its present form is in partial fulfilment of the requirements for the award of the degree of **BACHELOR OF TECHNOLOGY IN ELECTRONICS AND COMMUNICATION ENGINEERING** and as per regulations of the . To the best of my knowledge, the results embodied in this report, are original in nature and worthy of incorporation in the present version of the report.

**Guide / Supervisor**

---

**Mr. Tathagata Chatterjee**

---

## **PROJECT OBJECTIVE**

The main objective of this project is to develop a machine learning model using polynomial regression to predict the maximum temperature based on historical weather data. By analyzing the relationship between various weather features such as humidity, pressure, wind speed, temperature, and cloud cover at different times, the model aims to accurately forecast the maximum temperature for future time periods. The project seeks to leverage machine learning algorithms to analyze large volumes of weather data, identify patterns, and establish predictive models that can aid in weather forecasting.

# **INDEX**

- 1. Libraries used in Python**
- 2. Brief description of library functions**
- 3. Project Discussion**
- 4. Project Steps**
- 5. Result**
- 6. Conclusion**

# **LIBRARIES IN PYTHON**

## **What is a Library?**

- A library is a collection of pre-combined codes that can be used iteratively to reduce the time required to code.
- They are particularly useful for accessing the pre-written frequently used codes instead of writing them from scratch every single time.
- Similar to physical libraries, these are a collection of reusable resources, which means every library has a root source.
- This is the foundation behind the numerous open-source libraries available in Python.

## **What is a Python Library?**

- Python library is a collection of modules that contain functions and classes that can be used by other programs to perform various tasks.

## **What are the uses of Libraries in Python ?**

- As we write large-size programs in Python, we want to maintain the code's modularity. For the easy maintenance of the code, we split the code into different parts and we can use that code later ever we need it.
- In Python, *modules* play that part. Instead of using the same code in different programs and making the code complex, we define mostly used functions in modules and we can just simply import them in a program wherever there is a requirement.
- We don't need to write that code but still, we can use its functionality by importing its module. Multiple interrelated modules are stored in a library. And whenever we need to use a module, we import it from its library. In Python, it's a very simple job to do due to its easy syntax. We just need to use import.



## **LIBRARIES USED IN THIS PROJECT**

**pandas:** pandas is a powerful library for data manipulation and analysis. It provides data structures and functions to efficiently work with structured data, such as loading datasets, handling missing values, filtering, and transforming data.

**numpy:** numpy is a fundamental library for numerical computations in Python. It provides efficient data structures, such as arrays and matrices, and a wide range of mathematical functions. numpy is commonly used for handling numerical operations and data manipulation in machine learning projects.

**sklearn.preprocessing:** This module from scikit-learn (sklearn) library provides various preprocessing techniques for data preparation. In this project, the PolynomialFeatures class from this module is used to generate polynomial features from the existing features. It transforms the original features into a set of new features by raising them to different powers, allowing us to capture non-linear relationships between the features and the target variable.

**sklearn.linear\_model:** The LinearRegression class from the sklearn.linear\_model module is used to perform linear regression. It fits a linear model to the given data by minimizing the sum of squared residuals between the predicted and actual target values. In this project, the linear regression model is used as the baseline model before applying polynomial regression.

**sklearn.model\_selection:** This module provides functions to split datasets into train and test sets. The train\_test\_split function from this module is used

to split the dataset into a training set and a testing set. It ensures that the temporal order of the data is maintained, which is crucial for time-series data like weather data.

**sklearn.metrics:** The sklearn.metrics module provides various metrics to evaluate the performance of machine learning models. In this project, the mean\_squared\_error and r2\_score functions from this module are used. mean\_squared\_error calculates the mean squared error between the predicted and actual target values, while r2\_score computes the coefficient of determination, which measures the proportion of the variance in the target variable that is predictable from the independent variables.

These libraries are widely used in machine learning projects for data manipulation, model building, and evaluation, making them valuable tools for developing a weather prediction model based on machine learning techniques.

## **PROJECT DISCUSSION**

- Weather prediction plays a crucial role in various fields such as agriculture, transportation, tourism, and disaster management. Machine learning techniques offer a promising approach to analyzing large amounts of historical weather data and extracting meaningful patterns to make accurate predictions. In this project, we will focus on predicting the maximum temperature as it is a key factor influencing daily weather conditions.
- To achieve our goal, we will gather a comprehensive dataset comprising historical weather data from reliable sources such as Weather APIs, databases, or specialized weather data providers. This dataset will include relevant

features such as humidity, pressure, wind speed, temperature, and cloud cover at different time intervals.

- The dataset will be preprocessed to handle missing values, outliers, and perform feature scaling or normalization if required. We will then split the data into training and testing sets, ensuring that the temporal order of the data is maintained.
- Next, we will employ polynomial regression as our machine learning algorithm. Polynomial regression can capture non-linear relationships between the weather features and the target variable (maximum temperature). We will train the model using the training data, tune hyperparameters if necessary, and evaluate its performance using appropriate metrics such as mean squared error or R-squared.
- Once the model is trained and validated, we will use it to predict the maximum temperature for future time periods by providing the corresponding weather feature values as input. The accuracy of the predictions will be assessed by comparing them with the actual observed maximum temperatures.

## **PROJECT STEPS**

1. Gather historical weather data from reliable sources, including features such as humidity, pressure, wind speed, temperature, and cloud cover at different time intervals.
2. Preprocess the dataset by handling missing values, outliers, and performing feature scaling or normalization.
3. Split the dataset into training and testing sets, preserving the temporal order of the data.

4. Apply polynomial regression as the machine learning algorithm for predicting the maximum temperature.
5. Train the model using the training data and fine-tune hyperparameters if necessary.
6. Evaluate the model's performance using appropriate metrics such as mean squared error or R-squared.
7. Use the trained model to predict the maximum temperature for future time periods based on the corresponding weather feature values.
8. Assess the accuracy of the predictions by comparing them with the actual observed maximum temperatures.

## Step1:

```
[2] #Polynomial Regression
```

```
[15] # Importing all necessary Libraries
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
```

```
[16] from sklearn.preprocessing import PolynomialFeatures
from sklearn.linear_model import LinearRegression
from sklearn.model_selection import train_test_split
from sklearn.metrics import mean_squared_error, r2_score
from sklearn import metrics
```

## Step 2:

```
#Importing our dataset and displaying it
df=pd.read_csv('weather.csv')
print(df)
```

[17]

```
...
      MinTemp  MaxTemp  Rainfall  Evaporation  Sunshine  WindGustDir  \
0           8.0    24.3        0.0          3.4         6.3          NW
1          14.0    26.9        3.6          4.4         9.7          ENE
2          13.7    23.4        3.6          5.8         3.3          NW
3          13.3    15.5       39.8          7.2         9.1          NW
4           7.6    16.1        2.8          5.6        10.6          SSE
..         ...      ...      ...      ...      ...      ...
361         9.0    30.7        0.0          7.6        12.1         NNW
362         7.1    28.4        0.0         11.6        12.7          N
363        12.5    19.9        0.0          8.4         5.3          ESE
364        12.5    26.9        0.0          5.0         7.1          NW
365        12.3    30.2        0.0          6.0        12.6          NW

      WindGustSpeed  WindDir9am  WindDir3pm  WindSpeed9am  ...  Humidity3pm  \
0              30.0          SW          NW           6.0  ...         29
1              39.0           E           W           4.0  ...         36
2              85.0           N          NNE           6.0  ...         69
3              54.0         WNW           W          30.0  ...         56
4              50.0         SSE          ESE          20.0  ...         49
..         ...      ...      ...      ...      ...      ...
361             76.0         SSE          NW           7.0  ...         15
362             48.0         NNW         NNW           2.0  ...         22
363             43.0         ENE          ENE          11.0  ...         47
364             46.0         SSW         WNW           6.0  ...         39
365             78.0          NW          WNW          31.0  ...         13

...
364              No          0.0          No
365              No          0.0          No

[366 rows x 22 columns]
```

## Step 3:

```
#Filling the null values if present in our dataset
df['Humidity9am'].fillna(60,inplace=True)
df['Pressure9am'].fillna(1000,inplace=True)
df['WindSpeed9am'].fillna(6,inplace=True)
df['Temp9am'].fillna(10,inplace=True)
df['Cloud9am'].fillna(2,inplace=True)
df['Humidity3pm'].fillna(40,inplace=True)
df['Pressure3pm'].fillna(1000,inplace=True)
df['WindSpeed3pm'].fillna(10,inplace=True)
df['Temp3pm'].fillna(15,inplace=True)
df['Cloud3pm'].fillna(4,inplace=True)
df['MaxTemp'].fillna(17,inplace=True)
```

## Step 4:

```
# Splitting our data into independent variable and dependent variables
X = df[['Humidity9am', 'Pressure9am', 'WindSpeed9am', 'Temp9am', 'Cloud9am', 'Humidity3pm', 'Pressure3pm', 'WindSpeed3pm', 'Temp3pm', 'Cloud3pm']].values
y = df['MaxTemp'].values
```

[19]

```
# Further splitting our data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

[20]

## Step 5:

```
[21] # Creating polynomial features
poly_features = PolynomialFeatures(degree=2)
X_train_poly = poly_features.fit_transform(X_train)
X_test_poly = poly_features.transform(X_test)
```

```
[22] # Creating and training the polynomial regression model
model = LinearRegression()
model.fit(X_train_poly, y_train)
```

```
... LinearRegression()
```

```
[23] # Making predictions on the test set and storing it
y_pred = model.predict(X_test_poly)
```

## Step 6:

```
[25] # Evaluating our regression model
print('Mean absolute error is:', metrics.mean_absolute_error(y_test,y_pred))
print('Mean squared error is:', metrics.mean_squared_error(y_test,y_pred))
print('Root mean squared error is:', np.sqrt(metrics.mean_squared_error(y_test,y_pred)))
```

```
... Mean absolute error is: 1.989196761878761
Mean squared error is: 7.688050939765337
Root mean squared error is: 2.7727334779537207
```

## Step 7:

```
[29] #Displaying actual and predicted values of different parameters
print()
df= pd.DataFrame({'Actual': y_test.flatten(), 'Predicted': y_pred.flatten()})
print(df.describe())
```

```
...
      Actual  Predicted
count  74.000000  74.000000
mean    21.286486   21.404513
std      6.550506    6.704385
min      7.600000    4.314337
25%     16.175000   18.220199
50%     21.050000   21.599418
75%     26.100000   26.267306
max     35.700000   39.558559
```

## Step 8:

Here we display our final prediction on the variable “*Maxtemp*” (referring to Maximum Temperature) based on different other variables that we took into consideration from the concerned dataset.

```
#Displaying our final prediction in comparison to actual values
print(df)
```

```
[32]
```

```
...
   Actual Predicted
0    19.2   20.644326
1    26.5   26.037237
2    32.1   30.097865
3    16.1   18.050429
4    28.3   27.832327
..     ...      ...
69   22.5   26.382161
70   29.6   29.280479
71   17.4   20.941310
72   12.1   11.432063
73   18.0   18.915279

[74 rows x 2 columns]
```

## RESULT

The result of this project will be a machine learning model capable of accurately predicting the maximum temperature based on historical weather data. The model will undergo rigorous evaluation using suitable metrics such as mean squared error or R-squared to assess its predictive performance. The accuracy of the predictions will be determined by comparing them with the actual observed maximum temperatures. The project aims to achieve a model with a high level of accuracy and reliability in forecasting the maximum temperature.

## CONCLUSION

In conclusion, this project demonstrates the potential of machine learning algorithms, specifically polynomial regression, in weather prediction. By analyzing large volumes of historical weather data, extracting patterns, and establishing predictive models, it becomes possible to forecast the maximum temperature accurately. The developed machine learning model provides valuable insights for various applications in sectors

such as agriculture, transportation, tourism, and disaster management, which heavily rely on accurate weather predictions. The project highlights the importance of leveraging machine learning techniques to enhance traditional weather forecasting methods and improve decision-making processes based on reliable weather predictions.