```python
import numpy as np;
import pandas as pd
import matplotlib.pyplot as plt;
import seaborn as sn

#load the dataset
dataset=pd.read_csv("c:\\Users\\lsrin\\Downloads\\TS-2\\Adavance_ML\\
climate.csv")
print(dataset)
```

```
          STATION                 DATE REPORT_TYPE  SOURCE
BackupElements  \
0     72518014735  2015-01-01T23:59:00       SOD         6
PRECIP
1     72518014735  2015-01-02T23:59:00       SOD         6
PRECIP
2     72518014735  2015-01-03T23:59:00       SOD         6
PRECIP
3     72518014735  2015-01-04T23:59:00       SOD         6
PRECIP
4     72518014735  2015-01-05T23:59:00       SOD         6
PRECIP
...           ...                  ...         ...     ...
...
2663  72518014735  2022-05-27T23:59:00       SOD         6
PRECIP
2664  72518014735  2022-05-28T23:59:00       SOD         6
PRECIP
2665  72518014735  2022-05-29T23:59:00       SOD         6
PRECIP
2666  72518014735  2022-05-30T23:59:00       SOD         6
PRECIP
2667  72518014735  2022-05-31T23:59:00       SOD         6
PRECIP

      BackupElevation BackupEquipment  BackupLatitude  BackupLongitude
\
0                 260         PLASTIC         42.6918        -73.83109

1                 260         PLASTIC         42.6918        -73.83109

2                 260         PLASTIC         42.6918        -73.83109

3                 260         PLASTIC         42.6918        -73.83109

4                 260         PLASTIC         42.6918        -73.83109

...               ...             ...             ...              ...

2663              260         PLASTIC         42.6812        -73.81650
```

```
2664              260         PLASTIC         42.6812        -73.81650

2665              260         PLASTIC         42.6812        -73.81650

2666              260         PLASTIC         42.6812        -73.81650

2667              260         PLASTIC         42.6812        -73.81650


          BackupName  ...  DailyPeakWindDirection  DailyPeakWindSpeed
\
0     NWS ALBANY, NY  ...                   190.0                26.0

1     NWS ALBANY, NY  ...                   250.0                30.0

2     NWS ALBANY, NY  ...                   170.0                21.0

3     NWS ALBANY, NY  ...                   290.0                33.0

4     NWS ALBANY, NY  ...                   280.0                42.0

...              ...  ...                     ...                 ...

2663  NWS ALBANY, NY  ...                   160.0                28.0

2664  NWS ALBANY, NY  ...                   310.0                26.0

2665  NWS ALBANY, NY  ...                    90.0                13.0

2666  NWS ALBANY, NY  ...                   200.0                15.0

2667  NWS ALBANY, NY  ...                   250.0                29.0


      DailyPrecipitation  DailySnowDepth  DailySnowfall  \
0                   0.00             0.0            0.0
1                      T             0.0              T
2                   0.57             0.0            1.6
3                   0.22             1.0            0.0
4                      T             0.0              T
...                  ...             ...            ...
2663                0.00               0            0.0
2664                0.04               0            0.0
2665                0.00               0            0.0
2666                0.00               0            0.0
2667                0.00               0            0.0

      DailySustainedWindDirection  DailySustainedWindSpeed  Sunrise
Sunset  \
0                           190.0                     20.0    726.0
```

```
1632.0
1                           310.0                      23.0    726.0
1633.0
2                           160.0                      15.0    726.0
1634.0
3                           290.0                      24.0    726.0
1635.0
4                           290.0                      32.0    726.0
1636.0
...                           ...                       ...      ...
...
2663                        160.0                      21.0    423.0
1922.0
2664                        310.0                      22.0    422.0
1923.0
2665                        180.0                       9.0    421.0
1924.0
2666                        190.0                      12.0    421.0
1925.0
2667                        250.0                      21.0    420.0
1926.0

      WindEquipmentChangeDate
0                  2006-09-08
1                  2006-09-08
2                  2006-09-08
3                  2006-09-08
4                  2006-09-08
...                       ...
2663               2006-09-08
2664               2006-09-08
2665               2006-09-08
2666               2006-09-08
2667               2006-09-08

[2668 rows x 32 columns]
```

```
#DATA PROCESSSING
dataset.head(10)
dataset.tail(8)
dataset.sample(7)
print(dataset.columns)
print(dataset.dtypes)
print(dataset.shape[0]) #column
print(dataset.shape[1]) #rows
print(dataset.size)
```

```
Index(['STATION', 'DATE', 'REPORT_TYPE', 'SOURCE', 'BackupElements',
       'BackupElevation', 'BackupEquipment', 'BackupLatitude',
       'BackupLongitude', 'BackupName',
'DailyAverageDewPointTemperature',
       'DailyAverageDryBulbTemperature',
'DailyAverageRelativeHumidity',
       'DailyAverageSeaLevelPressure', 'DailyAverageStationPressure',
       'DailyAverageWetBulbTemperature', 'DailyAverageWindSpeed',
       'DailyCoolingDegreeDays',
'DailyDepartureFromNormalAverageTemperature',
       'DailyHeatingDegreeDays', 'DailyMaximumDryBulbTemperature',
       'DailyMinimumDryBulbTemperature', 'DailyPeakWindDirection',
       'DailyPeakWindSpeed', 'DailyPrecipitation', 'DailySnowDepth',
       'DailySnowfall', 'DailySustainedWindDirection',
       'DailySustainedWindSpeed', 'Sunrise', 'Sunset',
       'WindEquipmentChangeDate'],
      dtype='object')
STATION                                          int64
DATE                                            object
REPORT_TYPE                                     object
SOURCE                                           int64
BackupElements                                  object
BackupElevation                                  int64
BackupEquipment                                 object
BackupLatitude                                 float64
BackupLongitude                                float64
BackupName                                      object
DailyAverageDewPointTemperature                float64
DailyAverageDryBulbTemperature                 float64
DailyAverageRelativeHumidity                   float64
DailyAverageSeaLevelPressure                   float64
DailyAverageStationPressure                    float64
DailyAverageWetBulbTemperature                 float64
DailyAverageWindSpeed                          float64
DailyCoolingDegreeDays                         float64
DailyDepartureFromNormalAverageTemperature    float64
DailyHeatingDegreeDays                         float64
DailyMaximumDryBulbTemperature                 float64
DailyMinimumDryBulbTemperature                 float64
DailyPeakWindDirection                         float64
DailyPeakWindSpeed                             float64
DailyPrecipitation                              object
DailySnowDepth                                  object
DailySnowfall                                   object
DailySustainedWindDirection                    float64
DailySustainedWindSpeed                        float64
Sunrise                                        float64
Sunset                                         float64
WindEquipmentChangeDate                         object
dtype: object
2668
```

```
32
85376
```

```
#do some statistical for int64,float64
dataNumerical=dataset.select_dtypes(include=['int64','float64'])
#print(dataNumerical.columns)
print(dataNumerical.dtypes)
```

```
STATION                                        int64
SOURCE                                         int64
BackupElevation                                int64
BackupLatitude                               float64
BackupLongitude                              float64
DailyAverageDewPointTemperature              float64
DailyAverageDryBulbTemperature               float64
DailyAverageRelativeHumidity                 float64
DailyAverageSeaLevelPressure                 float64
DailyAverageStationPressure                  float64
DailyAverageWetBulbTemperature               float64
DailyAverageWindSpeed                        float64
DailyCoolingDegreeDays                       float64
DailyDepartureFromNormalAverageTemperature    float64
DailyHeatingDegreeDays                       float64
DailyMaximumDryBulbTemperature               float64
DailyMinimumDryBulbTemperature               float64
DailyPeakWindDirection                       float64
DailyPeakWindSpeed                           float64
DailySustainedWindDirection                  float64
DailySustainedWindSpeed                      float64
Sunrise                                      float64
Sunset                                       float64
dtype: object
```

```
print(dataNumerical.describe())
```

```
            STATION  SOURCE  BackupElevation  BackupLatitude
BackupLongitude  \
count  2.668000e+03  2668.0           2668.0     2668.000000
2668.000000
mean   7.251801e+10     6.0            260.0       42.689750       -
73.828268
std    0.000000e+00     0.0              0.0        0.004187
0.005764
min    7.251801e+10     6.0            260.0       42.681200       -
73.831090
25%    7.251801e+10     6.0            260.0       42.691800       -
73.831090
50%    7.251801e+10     6.0            260.0       42.691800       -
73.831090
75%    7.251801e+10     6.0            260.0       42.691800       -
```

```
73.831090
max    7.251801e+10     6.0            260.0       42.691800       -
73.816500

       DailyAverageDewPointTemperature  DailyAverageDryBulbTemperature
\
count                      2668.000000                     2668.000000

mean                         38.217766                       50.107571

std                          19.116250                       18.747310

min                         -19.000000                       -3.000000

25%                          24.000000                       35.000000

50%                          38.000000                       51.000000

75%                          55.000000                       67.000000

max                          73.000000                       87.000000


       DailyAverageRelativeHumidity  DailyAverageSeaLevelPressure  \
count                   2668.000000                   2668.000000
mean                      66.085082                     30.031945
std                       13.401359                      0.223771
min                       24.000000                     29.240000
25%                       57.000000                     29.880000
50%                       66.000000                     30.020000
75%                       76.000000                     30.180000
max                      100.000000                     30.740000

       DailyAverageStationPressure  ...  \
count                  2668.000000  ...
mean                     29.709059  ...
std                       0.220846  ...
min                      28.890000  ...
25%                      29.570000  ...
50%                      29.700000  ...
75%                      29.850000  ...
max                      30.420000  ...

       DailyDepartureFromNormalAverageTemperature
DailyHeatingDegreeDays  \
count                                 2668.000000
2668.000000
mean                                     2.155660
17.040480
std                                      8.202932
```

```
16.134205
min                                    -28.700000
0.000000
25%                                     -3.200000
0.000000
50%                                      1.900000
14.000000
75%                                      7.200000
30.000000
max                                     34.700000
68.000000

       DailyMaximumDryBulbTemperature  DailyMinimumDryBulbTemperature
\
count                     2668.000000                     2668.000000

mean                        59.418666                       40.299100

std                         20.003706                       18.122395

min                          5.000000                      -13.000000

25%                         42.000000                       27.000000

50%                         60.000000                       40.000000

75%                         77.000000                       55.250000

max                         97.000000                       77.000000


       DailyPeakWindDirection  DailyPeakWindSpeed  \
count             2668.000000         2668.000000
mean               222.387556           25.513493
std                 90.828564            9.436276
min                 10.000000            6.000000
25%                170.000000           19.000000
50%                260.000000           24.000000
75%                290.000000           31.000000
max                360.000000           70.000000

       DailySustainedWindDirection  DailySustainedWindSpeed
Sunrise  \
count                  2668.000000              2668.000000
2668.000000
mean                    223.924288                19.023238
563.145427
std                      90.846564                 6.942113
108.536855
min                      10.000000                 5.000000
```

```
416.000000
25%                     170.000000                14.000000
447.000000
50%                     270.000000                18.000000
547.000000
75%                     290.000000                23.000000
650.000000
max                     360.000000                67.000000
726.000000

            Sunset
count  2668.000000
mean   1783.491004
std     111.230222
min    1621.000000
25%    1658.000000
50%    1805.000000
75%    1905.000000
max    1938.000000

[8 rows x 23 columns]
```

```
# # print(dataNumerical.info())
# dataNumerical.isnull().sum()
# print(dataNumerical.columns) #drop the
Station,source,'BackupElevation',
'BackupLatitude','BackupLongitude',Sunrise', 'Sunset

# dataFront=dataNumerical.iloc[:,0:5]
# dataBack=dataNumerical.iloc[:,-2:]
# print(dataFront.columns)
# print(dataBack.columns)
# # Combine the column names from dataFront and dataBack
# # columns_to_drop = list(dataFront.columns) + list(dataBack.columns)
# # #now drop those variable from NumericalDate
# # dataNumerical=dataNumerical.drop(columns=columns_to_drop)
# # print("The Final dataset to work:",dataNumerical.columns)

dataNumerical.isnull().sum()
```

```
STATION                                       0
SOURCE                                        0
BackupElevation                               0
BackupLatitude                                0
BackupLongitude                               0
DailyAverageDewPointTemperature               0
DailyAverageDryBulbTemperature                0
DailyAverageRelativeHumidity                  0
DailyAverageSeaLevelPressure                  0
DailyAverageStationPressure                   0
```

```
DailyAverageWetBulbTemperature                0
DailyAverageWindSpeed                         0
DailyCoolingDegreeDays                        0
DailyDepartureFromNormalAverageTemperature    0
DailyHeatingDegreeDays                        0
DailyMaximumDryBulbTemperature                0
DailyMinimumDryBulbTemperature                0
DailyPeakWindDirection                        0
DailyPeakWindSpeed                            0
DailySustainedWindDirection                   0
DailySustainedWindSpeed                       0
Sunrise                                       0
Sunset                                        0
dtype: int64

# Print the column names of the DataFrame
print(dataNumerical.columns)

# Drop the specified columns from the DataFrame
dataNumerical = dataNumerical.drop(columns=[
    'STATION', 'SOURCE', 'BackupElevation', 'BackupLatitude',
    'BackupLongitude', 'Sunrise', 'Sunset'
])

Index(['STATION', 'SOURCE', 'BackupElevation', 'BackupLatitude',
       'BackupLongitude', 'DailyAverageDewPointTemperature',
       'DailyAverageDryBulbTemperature',
'DailyAverageRelativeHumidity',
       'DailyAverageSeaLevelPressure', 'DailyAverageStationPressure',
       'DailyAverageWetBulbTemperature', 'DailyAverageWindSpeed',
       'DailyCoolingDegreeDays',
'DailyDepartureFromNormalAverageTemperature',
       'DailyHeatingDegreeDays', 'DailyMaximumDryBulbTemperature',
       'DailyMinimumDryBulbTemperature', 'DailyPeakWindDirection',
       'DailyPeakWindSpeed', 'DailySustainedWindDirection',
       'DailySustainedWindSpeed', 'Sunrise', 'Sunset'],
      dtype='object')
      DailyAverageDewPointTemperature  DailyAverageDryBulbTemperature
\
0                                 7.0                            25.0

1                                17.0                            32.0

2                                18.0                            27.0

3                                35.0                            39.0

4                                11.0                            27.0

...                               ...                             ...
```

```
2663                             63.0                            70.0
2664                             59.0                            68.0
2665                             50.0                            66.0
2666                             59.0                            72.0
2667                             62.0                            79.0

      DailyAverageRelativeHumidity  DailyAverageSeaLevelPressure  \
0                             46.0                         29.97   
1                             57.0                         30.18   
2                             74.0                         30.46   
3                             86.0                         29.76   
4                             59.0                         30.12   
...                            ...                           ...   
2663                          76.0                         29.83   
2664                          72.0                         29.74   
2665                          57.0                         30.03   
2666                          62.0                         30.03   
2667                          55.0                         29.93   

      DailyAverageStationPressure  DailyAverageWetBulbTemperature  \
0                           29.65                            21.0   
1                           29.80                            26.0   
2                           30.16                            23.0   
3                           29.47                            38.0   
4                           29.73                            20.0   
...                           ...                             ...   
2663                        29.54                            66.0   
2664                        29.41                            63.0   
2665                        29.70                            58.0   
2666                        29.73                            64.0   
2667                        29.62                            68.0   

      DailyAverageWindSpeed  DailyCoolingDegreeDays  \
0                       8.8                     0.0   
1                       9.5                     0.0   
2                       4.3                     0.0   
3                      10.0                     0.0   
4                      16.8                     0.0   
...                     ...                     ...   
2663                   12.7                     5.0   
2664                    7.6                     3.0   
2665                    2.2                     1.0   
2666                    3.4                     7.0   
2667                    8.4                    14.0   
```

```
      DailyDepartureFromNormalAverageTemperature
DailyHeatingDegreeDays  \
0                                            1.4
40.0
1                                            8.6
33.0
2                                            3.7
38.0
3                                           15.9
26.0
4                                            4.1
38.0
...                                          ...                     .
..
2663                                         8.8
0.0
2664                                         6.5
0.0
2665                                         4.2
0.0
2666                                         9.9
0.0
2667                                        16.5
0.0

      DailyMaximumDryBulbTemperature
DailyMinimumDryBulbTemperature  \
0                               32.0                            18.0

1                               37.0                            26.0

2                               33.0                            20.0

3                               45.0                            33.0

4                               41.0                            13.0

...                              ...                             ...

2663                            77.0                            63.0

2664                            80.0                            56.0

2665                            80.0                            51.0

2666                            88.0                            55.0

2667                            92.0                            65.0
```

```
      DailyPeakWindDirection  DailyPeakWindSpeed
DailySustainedWindDirection  \
0                      190.0                26.0
190.0
1                      250.0                30.0
310.0
2                      170.0                21.0
160.0
3                      290.0                33.0
290.0
4                      280.0                42.0
290.0
...                      ...                 ...
...
2663                   160.0                28.0
160.0
2664                   310.0                26.0
310.0
2665                    90.0                13.0
180.0
2666                   200.0                15.0
190.0
2667                   250.0                29.0
250.0

      DailySustainedWindSpeed
0                        20.0
1                        23.0
2                        15.0
3                        24.0
4                        32.0
...                       ...
2663                     21.0
2664                     22.0
2665                      9.0
2666                     12.0
2667                     21.0

[2668 rows x 16 columns]
```

```
# Display the updated DataFrame
print(dataNumerical.columns)
```

```
Index(['DailyAverageDewPointTemperature',
'DailyAverageDryBulbTemperature',
       'DailyAverageRelativeHumidity', 'DailyAverageSeaLevelPressure',
       'DailyAverageStationPressure',
'DailyAverageWetBulbTemperature',
       'DailyAverageWindSpeed', 'DailyCoolingDegreeDays',
       'DailyDepartureFromNormalAverageTemperature',
'DailyHeatingDegreeDays',
       'DailyMaximumDryBulbTemperature',
'DailyMinimumDryBulbTemperature',
       'DailyPeakWindDirection', 'DailyPeakWindSpeed',
       'DailySustainedWindDirection', 'DailySustainedWindSpeed'],
      dtype='object')
```

using linearRegression

''' **Predict Precipitation based on Temperature:** - **X (Feature):** `DailyAverageDryBulbTemperature` - **Y (Target):** `DailyPrecipitation` '''

using Multi-linear Regression

# X all except "DailyAverageDryBulbTemperature"

using multiple (ploynomial Regression)

```
# Using Linear Regress

# Features matrix (X) with the temperature column
X = dataNumerical[['DailyAverageDryBulbTemperature']]

# Ensure 'DailyPrecipitation' is numeric and fill NaN values with 0
dataNumerical['DailyPrecipitation'] = 
pd.to_numeric(dataNumerical['DailyPrecipitation'], errors='coerce')
dataNumerical['DailyPrecipitation'].fillna(0, inplace=True)

# Target vector (y) with precipitation values
y = dataNumerical['DailyPrecipitation']

# Print X and y to verify
print(X)
print(y)
```

```
      DailyAverageDryBulbTemperature
0                               25.0
1                               32.0
2                               27.0
3                               39.0
4                               27.0
...                              ...
2663                            70.0
2664                            68.0
2665                            66.0
```

```
2666                            72.0
2667                            79.0

[2668 rows x 1 columns]
0       0.00
1       0.00
2       0.57
3       0.22
4       0.00
        ...
2663    0.00
2664    0.04
2665    0.00
2666    0.00
2667    0.00
Name: DailyPrecipitation, Length: 2668, dtype: float64

C:\Users\lsrin\AppData\Local\Temp\ipykernel_13048\1432735839.py:6:
FutureWarning: A value is trying to be set on a copy of a DataFrame or
Series through chained assignment using an inplace method.
The behavior will change in pandas 3.0. This inplace method will never
work because the intermediate object on which we are setting values
always behaves as a copy.

For example, when doing 'df[col].method(value, inplace=True)', try
using 'df.method({col: value}, inplace=True)' or df[col] =
df[col].method(value) instead, to perform the operation inplace on the
original object.


  dataNumerical['DailyPrecipitation'].fillna(0, inplace=True)
```

```python
from sklearn.model_selection import train_test_split

# Split the data into training and testing sets (80% train, 20% test)
X_train, X_test, y_train, y_test = train_test_split(X, y,
test_size=0.2, random_state=32)

# Print the number of rows in X_train
print(X_train.shape[0])
```

```
2134
```

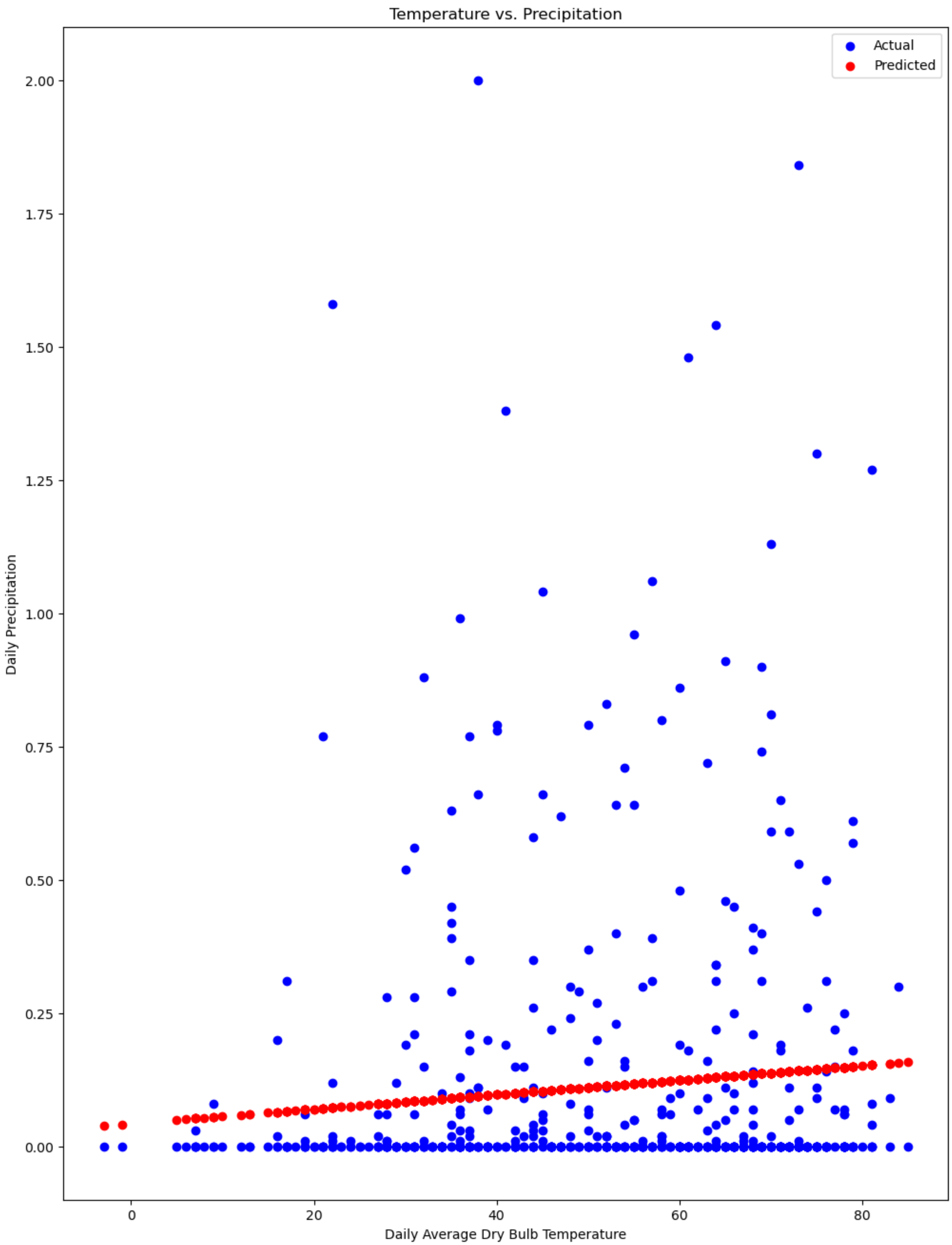
```python
#train the model
from sklearn.linear_model import LinearRegression
# Initialize and train the linear regression model
model = LinearRegression()
model.fit(X_train, y_train)
```

```
LinearRegression()
```

```python
# Make predictions
y_pred = model.predict(X_test)
#y_pred

# Evaluate the model
from sklearn.metrics import mean_squared_error, r2_score
mse = mean_squared_error(y_test, y_pred)
r2 = r2_score(y_test, y_pred)

print(f'Mean Squared Error: {mse}')
print(f'R-squared: {r2}')
```

```
Mean Squared Error: 0.0729113990218066
R-squared: 0.0025889789634933047
```

```python
#Plot the true vs. predicted values.
import matplotlib.pyplot as plt
plt.figure(figsize=(12, 16))
plt.scatter(X_test, y_test, color='blue', label='Actual')
plt.scatter(X_test, y_pred, color='red', label='Predicted')
plt.xlabel('Daily Average Dry Bulb Temperature')
plt.ylabel('Daily Precipitation')
plt.title('Temperature vs. Precipitation')
plt.legend()
plt.show()
```



```
dataNumerical.columns
```

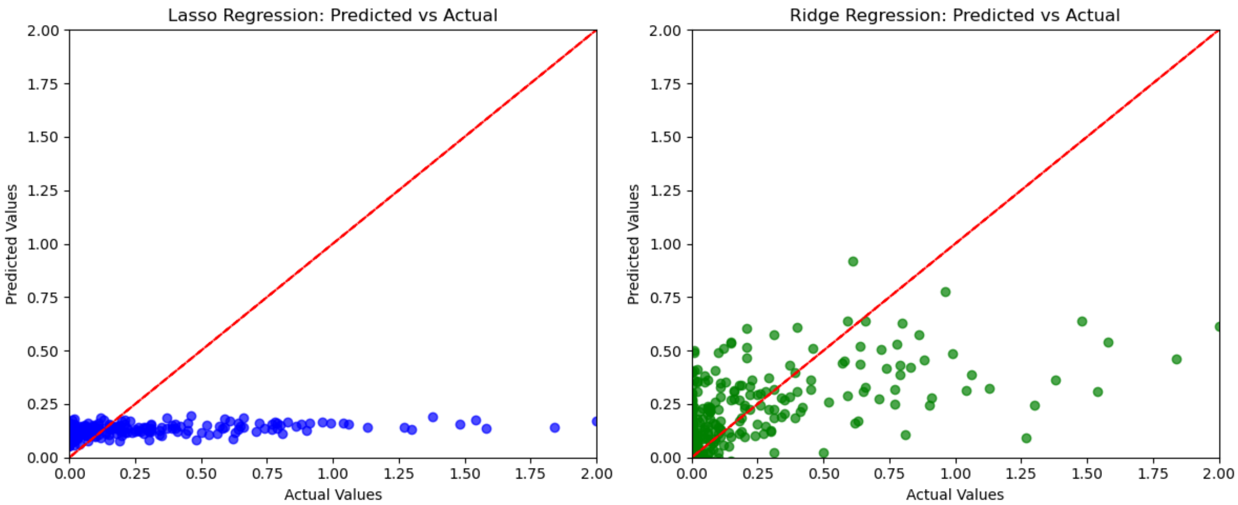
```
Index(['DailyAverageDewPointTemperature',
'DailyAverageDryBulbTemperature',
       'DailyAverageRelativeHumidity', 'DailyAverageSeaLevelPressure',
       'DailyAverageStationPressure',
'DailyAverageWetBulbTemperature',
       'DailyAverageWindSpeed', 'DailyCoolingDegreeDays',
       'DailyDepartureFromNormalAverageTemperature',
'DailyHeatingDegreeDays',
       'DailyMaximumDryBulbTemperature',
'DailyMinimumDryBulbTemperature',
       'DailyPeakWindDirection', 'DailyPeakWindSpeed',
       'DailySustainedWindDirection', 'DailySustainedWindSpeed',
       'DailyPrecipitation'],
      dtype='object')
```

```python
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import PolynomialFeatures, StandardScaler
from sklearn.linear_model import Lasso, Ridge
from sklearn.metrics import mean_squared_error, r2_score

# Step 1: Data Preparation
X = data.drop(columns=['DailyPrecipitation'])  # Features
y = data['DailyPrecipitation']  # Target
X_train, X_test, y_train, y_test = train_test_split(X, y,
test_size=0.2, random_state=32)

# Step 2: Polynomial Transformation (Degree 2)
poly = PolynomialFeatures(degree=2, include_bias=False)
X_train_poly = poly.fit_transform(X_train)
X_test_poly = poly.transform(X_test)

# Step 3: Feature Scaling
scaler = StandardScaler()
X_train_scaled = scaler.fit_transform(X_train_poly)
X_test_scaled = scaler.transform(X_test_poly)

# Lasso Regression with increased max_iter
lasso = Lasso(alpha=0.1, max_iter=10000)
lasso.fit(X_train_scaled, y_train)
y_pred_lasso = lasso.predict(X_test_scaled)

# Ridge Regression with increased max_iter
ridge = Ridge(alpha=0.1, max_iter=10000)
ridge.fit(X_train_scaled, y_train)
y_pred_ridge = ridge.predict(X_test_scaled)

# Step 4: Evaluate Lasso Model
mse_lasso = mean_squared_error(y_test, y_pred_lasso)
r2_lasso = r2_score(y_test, y_pred_lasso)
```

```python
print(f'Lasso Regression - MSE: {mse_lasso}, R²: {r2_lasso}')

# Step 4: Evaluate Ridge Model
mse_ridge = mean_squared_error(y_test, y_pred_ridge)
r2_ridge = r2_score(y_test, y_pred_ridge)
print(f'Ridge Regression - MSE: {mse_ridge}, R²: {r2_ridge}')
```

```
Lasso Regression - MSE: 0.06687719302293675, R²: 0.08513551691536181
Ridge Regression - MSE: 0.048396740676427255, R²: 0.33794381700895426
```

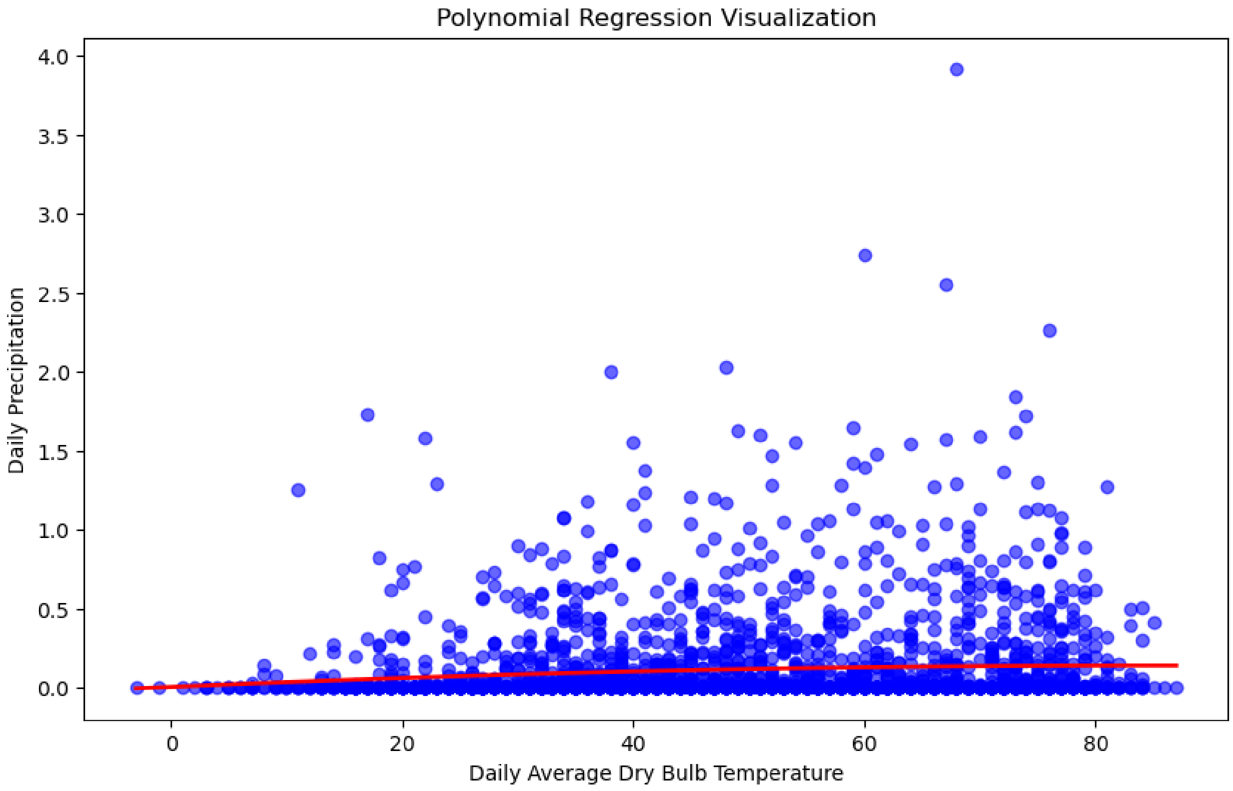
```python
#actual vs predication
import matplotlib.pyplot as plt
import numpy as np

# Scatter plot of predicted vs actual for Lasso
plt.figure(figsize=(12, 5))

# Lasso Regression
plt.subplot(1, 2, 1)
plt.scatter(y_test, y_pred_lasso, color='blue', alpha=0.7)
plt.plot(y_test, y_test, color='red', linestyle='--')  # Perfect prediction line
plt.title('Lasso Regression: Predicted vs Actual')
plt.xlabel('Actual Values')
plt.ylabel('Predicted Values')
plt.xlim([0, max(y_test)])
plt.ylim([0, max(y_test)])

# Ridge Regression
plt.subplot(1, 2, 2)
plt.scatter(y_test, y_pred_ridge, color='green', alpha=0.7)
plt.plot(y_test, y_test, color='red', linestyle='--')  # Perfect prediction line
plt.title('Ridge Regression: Predicted vs Actual')
plt.xlabel('Actual Values')
plt.ylabel('Predicted Values')
plt.xlim([0, max(y_test)])
plt.ylim([0, max(y_test)])

plt.tight_layout()
plt.show()
```



```
import numpy as np
import matplotlib.pyplot as plt
from sklearn.preprocessing import PolynomialFeatures
from sklearn.linear_model import LinearRegression

# Use one feature for visualization (e.g.,
DailyAverageDryBulbTemperature)
X_single_feature = data[['DailyAverageDryBulbTemperature']]
y = data['DailyPrecipitation']

# Split the data
X_train_single, X_test_single, y_train_single, y_test_single =
train_test_split(X_single_feature, y, test_size=0.2, random_state=32)

# Create polynomial features
poly = PolynomialFeatures(degree=2)  # Change the degree as needed
X_train_poly = poly.fit_transform(X_train_single)
X_test_poly = poly.transform(X_test_single)

# Fit a Linear Regression model
model = LinearRegression()
model.fit(X_train_poly, y_train_single)

# Create a range of values for plotting the polynomial curve
X_plot = np.linspace(X_single_feature.min(), X_single_feature.max(),
100).reshape(-1, 1)
X_plot_poly = poly.transform(X_plot)
y_plot = model.predict(X_plot_poly)

# Plot the actual data and the polynomial regression curve
plt.figure(figsize=(10, 6))
plt.scatter(X_single_feature, y, color='blue', label='Actual Data',
alpha=0.6)
plt.plot(X_plot, y_plot, color='red', label='Polynomial Regression
Curve', linewidth=2)
plt.title('Polynomial Regression Visualization')
plt.xlabel('Daily Average Dry Bulb Temperature')
plt.ylabel('Daily Precipitation')
# plt.legend()
# plt.grid()
# plt.show()
```

```
C:\Users\lsrin\anaconda3\Lib\site-packages\sklearn\base.py:493:
UserWarning: X does not have valid feature names, but
PolynomialFeatures was fitted with feature names
  warnings.warn(
```

```
Text(0, 0.5, 'Daily Precipitation')
```


Polynomial Regression Visualization
Daily Precipitation
Daily Average Dry Bulb Temperature