airtrainaf

May 1, 2025

[1]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

[2]:
```
df=pd.read_csv('AirQuality.csv',sep=';')
```

[3]:
```
df.head()
```

[3]:

	Date	Time	CO(GT)	PT08.S1(CO)	NMHC(GT)	C6H6(GT)	PT08.S2(NMHC)	\
0	10/03/2004	18.00.00	2,6	1360.0	150.0	11,9	1046.0	
1	10/03/2004	19.00.00	2	1292.0	112.0	9,4	955.0	
2	10/03/2004	20.00.00	2,2	1402.0	88.0	9,0	939.0	
3	10/03/2004	21.00.00	2,2	1376.0	80.0	9,2	948.0	
4	10/03/2004	22.00.00	1,6	1272.0	51.0	6,5	836.0	

	NOx(GT)	PT08.S3(NOx)	NO2(GT)	PT08.S4(NO2)	PT08.S5(O3)	T	RH	\
0	166.0	1056.0	113.0	1692.0	1268.0	13,6	48,9	
1	103.0	1174.0	92.0	1559.0	972.0	13,3	47,7	
2	131.0	1140.0	114.0	1555.0	1074.0	11,9	54,0	
3	172.0	1092.0	122.0	1584.0	1203.0	11,0	60,0	
4	131.0	1205.0	116.0	1490.0	1110.0	11,2	59,6	

	AH	Unnamed: 15	Unnamed: 16
0	0,7578	NaN	NaN
1	0,7255	NaN	NaN
2	0,7502	NaN	NaN
3	0,7867	NaN	NaN
4	0,7888	NaN	NaN

[4]:
```
df=df.drop(['Unnamed: 15','Unnamed: 16'],axis=1)
```

[5]:
```
df.head()
```

[5]:

	Date	Time	CO(GT)	PT08.S1(CO)	NMHC(GT)	C6H6(GT)	PT08.S2(NMHC)	\
0	10/03/2004	18.00.00	2,6	1360.0	150.0	11,9	1046.0	
1	10/03/2004	19.00.00	2	1292.0	112.0	9,4	955.0	

```
2  10/03/2004  20.00.00    2,2       1402.0      88.0      9,0          939.0
3  10/03/2004  21.00.00    2,2       1376.0      80.0      9,2          948.0
4  10/03/2004  22.00.00    1,6       1272.0      51.0      6,5          836.0

   NOx(GT)  PT08.S3(NOx)  NO2(GT)  PT08.S4(NO2)  PT08.S5(O3)     T    RH  \
0    166.0        1056.0    113.0        1692.0       1268.0  13,6  48,9
1    103.0        1174.0     92.0        1559.0        972.0  13,3  47,7
2    131.0        1140.0    114.0        1555.0       1074.0  11,9  54,0
3    172.0        1092.0    122.0        1584.0       1203.0  11,0  60,0
4    131.0        1205.0    116.0        1490.0       1110.0  11,2  59,6

       AH
0  0,7578
1  0,7255
2  0,7502
3  0,7867
4  0,7888
```

[6]:
```
df=df.rename(columns={'T':'Temperature','RH':'Relative Humidity','AH':'Absolute␣
↪Humidity'})
```

[7]:
```
df.head()
```

[7]:
```
         Date      Time CO(GT)  PT08.S1(CO)  NMHC(GT) C6H6(GT)  PT08.S2(NMHC)  \
0  10/03/2004  18.00.00    2,6       1360.0     150.0     11,9         1046.0
1  10/03/2004  19.00.00      2       1292.0     112.0      9,4          955.0
2  10/03/2004  20.00.00    2,2       1402.0      88.0      9,0          939.0
3  10/03/2004  21.00.00    2,2       1376.0      80.0      9,2          948.0
4  10/03/2004  22.00.00    1,6       1272.0      51.0      6,5          836.0

   NOx(GT)  PT08.S3(NOx)  NO2(GT)  PT08.S4(NO2)  PT08.S5(O3) Temperature  \
0    166.0        1056.0    113.0        1692.0       1268.0        13,6
1    103.0        1174.0     92.0        1559.0        972.0        13,3
2    131.0        1140.0    114.0        1555.0       1074.0        11,9
3    172.0        1092.0    122.0        1584.0       1203.0        11,0
4    131.0        1205.0    116.0        1490.0       1110.0        11,2

  Relative Humidity Absolute Humidity
0              48,9            0,7578
1              47,7            0,7255
2              54,0            0,7502
3              60,0            0,7867
4              59,6            0,7888
```

[8]:
```
df=df.replace(',','.',regex=True)
df
```

[8]:
```
            Date      Time CO(GT)  PT08.S1(CO)  NMHC(GT) C6H6(GT)  \
0     10/03/2004  18.00.00    2.6       1360.0     150.0     11.9
1     10/03/2004  19.00.00      2       1292.0     112.0      9.4
2     10/03/2004  20.00.00    2.2       1402.0      88.0      9.0
3     10/03/2004  21.00.00    2.2       1376.0      80.0      9.2
4     10/03/2004  22.00.00    1.6       1272.0      51.0      6.5
…            …       …    …           …        …      …
9466         NaN       NaN    NaN          NaN       NaN      NaN
9467         NaN       NaN    NaN          NaN       NaN      NaN
9468         NaN       NaN    NaN          NaN       NaN      NaN
9469         NaN       NaN    NaN          NaN       NaN      NaN
9470         NaN       NaN    NaN          NaN       NaN      NaN

      PT08.S2(NMHC)  NOx(GT)  PT08.S3(NOx)  NO2(GT)  PT08.S4(NO2)  \
0            1046.0    166.0        1056.0    113.0        1692.0
1             955.0    103.0        1174.0     92.0        1559.0
2             939.0    131.0        1140.0    114.0        1555.0
3             948.0    172.0        1092.0    122.0        1584.0
4             836.0    131.0        1205.0    116.0        1490.0
…               …       …            …       …            …
9466            NaN      NaN           NaN      NaN           NaN
9467            NaN      NaN           NaN      NaN           NaN
9468            NaN      NaN           NaN      NaN           NaN
9469            NaN      NaN           NaN      NaN           NaN
9470            NaN      NaN           NaN      NaN           NaN

      PT08.S5(O3) Temperature Relative Humidity Absolute Humidity
0          1268.0        13.6              48.9            0.7578
1           972.0        13.3              47.7            0.7255
2          1074.0        11.9              54.0            0.7502
3          1203.0        11.0              60.0            0.7867
4          1110.0        11.2              59.6            0.7888
…           …          …                 …                …
9466          NaN         NaN               NaN              NaN
9467          NaN         NaN               NaN              NaN
9468          NaN         NaN               NaN              NaN
9469          NaN         NaN               NaN              NaN
9470          NaN         NaN               NaN              NaN

[9471 rows x 15 columns]
```

[9]:
```
float_col=['CO(GT)','C6H6(GT)','Temperature','Relative Humidity','Absolute␣
 ↪Humidity']
for col in float_col:
    df[float_col]=df[float_col].astype(float)
```

[10]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 9471 entries, 0 to 9470
Data columns (total 15 columns):
 #   Column             Non-Null Count  Dtype
---  ------             --------------  -----
 0   Date               9357 non-null   object
 1   Time               9357 non-null   object
 2   CO(GT)             9357 non-null   float64
 3   PT08.S1(CO)        9357 non-null   float64
 4   NMHC(GT)           9357 non-null   float64
 5   C6H6(GT)           9357 non-null   float64
 6   PT08.S2(NMHC)      9357 non-null   float64
 7   NOx(GT)            9357 non-null   float64
 8   PT08.S3(NOx)       9357 non-null   float64
 9   NO2(GT)            9357 non-null   float64
 10  PT08.S4(NO2)       9357 non-null   float64
 11  PT08.S5(O3)        9357 non-null   float64
 12  Temperature        9357 non-null   float64
 13  Relative Humidity  9357 non-null   float64
 14  Absolute Humidity  9357 non-null   float64
dtypes: float64(13), object(2)
memory usage: 1.1+ MB
```

[11]:
```
df=df.drop_duplicates()
```

[12]:
```
df=df.drop(['Date','Time'],axis=1)
```

[13]:
```
df.isna().sum()
```

[13]:
```
CO(GT)               1
PT08.S1(CO)          1
NMHC(GT)             1
C6H6(GT)             1
PT08.S2(NMHC)        1
NOx(GT)              1
PT08.S3(NOx)         1
NO2(GT)              1
PT08.S4(NO2)         1
PT08.S5(O3)          1
Temperature          1
Relative Humidity    1
Absolute Humidity    1
dtype: int64
```

[14]:
```
df=df.fillna(df.mean(numeric_only=True))
```

[15]:
```
df=df.dropna()
```

[16]:
```
df.isna().sum()
```

[16]:
```
CO(GT)                0
PT08.S1(CO)           0
NMHC(GT)              0
C6H6(GT)              0
PT08.S2(NMHC)         0
NOx(GT)               0
PT08.S3(NOx)          0
NO2(GT)               0
PT08.S4(NO2)          0
PT08.S5(O3)           0
Temperature           0
Relative Humidity     0
Absolute Humidity     0
dtype: int64
```

[17]:
```
# Data Integretion
common_col=['CO(GT)','NO2(GT)']
df1=df[common_col+['C6H6(GT)','NOx(GT)']]
df2=df[common_col+["PT08.S1(CO)","NMHC(GT)","C6H6(GT)","PT08.
 ↪S2(NMHC)","NOx(GT)","PT08.S3(NOx)","PT08.S4(NO2)","PT08.
 ↪S5(O3)","Temperature","Relative Humidity","Absolute Humidity"]]
```

[18]:
```
df1_s=df1.head(100)
df2_s=df2.head(100)
```

[19]:
```
inner_merged=pd.merge(df1_s,df2_s,on=common_col,how='inner')
inner_merged.head()
# inner_merged=pd.merge(df1_s,df2_s,on=['CO(GT)','NO2(GT)'],how='inner')
# inner_merged.head()
```

[19]:

	CO(GT)	NO2(GT)	C6H6(GT)_x	NOx(GT)_x	PT08.S1(CO)	NMHC(GT)	C6H6(GT)_y	\
0	2.6	113.0	11.9	166.0	1360.0	150.0	11.9	
1	2.0	92.0	9.4	103.0	1292.0	112.0	9.4	
2	2.2	114.0	9.0	131.0	1402.0	88.0	9.0	
3	2.2	122.0	9.2	172.0	1376.0	80.0	9.2	
4	1.6	116.0	6.5	131.0	1272.0	51.0	6.5	

	PT08.S2(NMHC)	NOx(GT)_y	PT08.S3(NOx)	PT08.S4(NO2)	PT08.S5(O3)	\
0	1046.0	166.0	1056.0	1692.0	1268.0	
1	955.0	103.0	1174.0	1559.0	972.0	
2	939.0	131.0	1140.0	1555.0	1074.0	
3	948.0	172.0	1092.0	1584.0	1203.0	
4	836.0	131.0	1205.0	1490.0	1110.0	

	Temperature	Relative Humidity	Absolute Humidity

```
0         13.6                48.9              0.7578
1         13.3                47.7              0.7255
2         11.9                54.0              0.7502
3         11.0                60.0              0.7867
4         11.2                59.6              0.7888
```

[20]:
```
right_merged=pd.merge(df1_s,df2_s,on=common_col,how='right')
right_merged.head()
```

[20]:
```
   CO(GT)  NO2(GT)  C6H6(GT)_x  NOx(GT)_x  PT08.S1(CO)  NMHC(GT)  C6H6(GT)_y  \
0     2.6    113.0         11.9      166.0       1360.0     150.0         11.9
1     2.0     92.0          9.4      103.0       1292.0     112.0          9.4
2     2.2    114.0          9.0      131.0       1402.0      88.0          9.0
3     2.2    122.0          9.2      172.0       1376.0      80.0          9.2
4     1.6    116.0          6.5      131.0       1272.0      51.0          6.5

   PT08.S2(NMHC)  NOx(GT)_y  PT08.S3(NOx)  PT08.S4(NO2)  PT08.S5(O3)  \
0         1046.0      166.0        1056.0        1692.0       1268.0
1          955.0      103.0        1174.0        1559.0        972.0
2          939.0      131.0        1140.0        1555.0       1074.0
3          948.0      172.0        1092.0        1584.0       1203.0
4          836.0      131.0        1205.0        1490.0       1110.0

   Temperature  Relative Humidity  Absolute Humidity
0         13.6               48.9             0.7578
1         13.3               47.7             0.7255
2         11.9               54.0             0.7502
3         11.0               60.0             0.7867
4         11.2               59.6             0.7888
```

[21]:
```
left_merged=pd.merge(df1_s,df2_s,on=common_col,how='left')
left_merged.head()
```

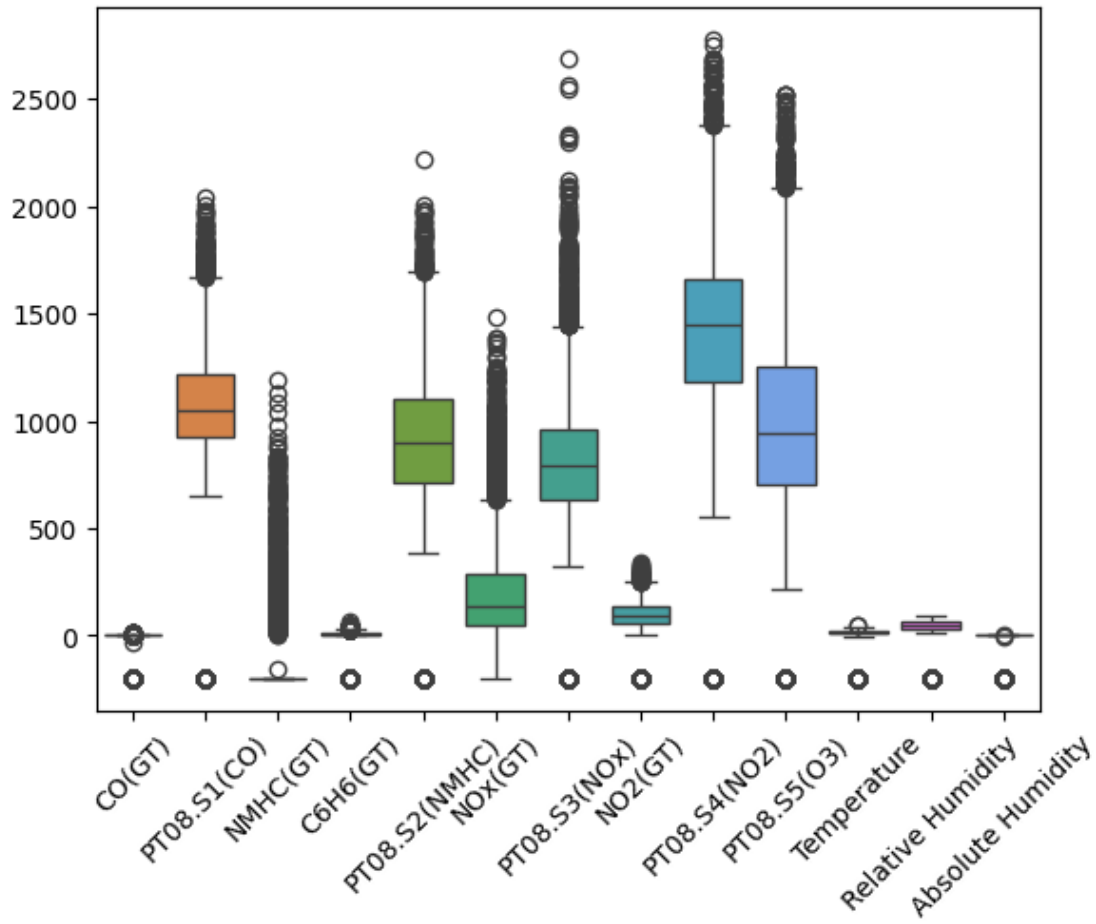
[21]:
```
   CO(GT)  NO2(GT)  C6H6(GT)_x  NOx(GT)_x  PT08.S1(CO)  NMHC(GT)  C6H6(GT)_y  \
0     2.6    113.0         11.9      166.0       1360.0     150.0         11.9
1     2.0     92.0          9.4      103.0       1292.0     112.0          9.4
2     2.2    114.0          9.0      131.0       1402.0      88.0          9.0
3     2.2    122.0          9.2      172.0       1376.0      80.0          9.2
4     1.6    116.0          6.5      131.0       1272.0      51.0          6.5

   PT08.S2(NMHC)  NOx(GT)_y  PT08.S3(NOx)  PT08.S4(NO2)  PT08.S5(O3)  \
0         1046.0      166.0        1056.0        1692.0       1268.0
1          955.0      103.0        1174.0        1559.0        972.0
2          939.0      131.0        1140.0        1555.0       1074.0
3          948.0      172.0        1092.0        1584.0       1203.0
4          836.0      131.0        1205.0        1490.0       1110.0
```

```
   Temperature  Relative Humidity  Absolute Humidity
0         13.6               48.9             0.7578
1         13.3               47.7             0.7255
2         11.9               54.0             0.7502
3         11.0               60.0             0.7867
4         11.2               59.6             0.7888
```

[22]:
```
sns.boxplot(df)
plt.xticks(rotation=45)
```

[22]:
```
([0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12],
 [Text(0, 0, 'CO(GT)'),
  Text(1, 0, 'PT08.S1(CO)'),
  Text(2, 0, 'NMHC(GT)'),
  Text(3, 0, 'C6H6(GT)'),
  Text(4, 0, 'PT08.S2(NMHC)'),
  Text(5, 0, 'NOx(GT)'),
  Text(6, 0, 'PT08.S3(NOx)'),
  Text(7, 0, 'NO2(GT)'),
  Text(8, 0, 'PT08.S4(NO2)'),
  Text(9, 0, 'PT08.S5(O3)'),
  Text(10, 0, 'Temperature'),
  Text(11, 0, 'Relative Humidity'),
  Text(12, 0, 'Absolute Humidity')])
```



[23]:
```
# Error Correcting
def remove_outliers(col):
    Q1=col.quantile(0.25)
    Q3=col.quantile(0.75)
    IQR=Q3-Q1
    lower=Q1-1.5*IQR
    upper=Q3+1.5*IQR
    outlier_mask=(col<lower)|(col>upper)
    return col[~outlier_mask]
```

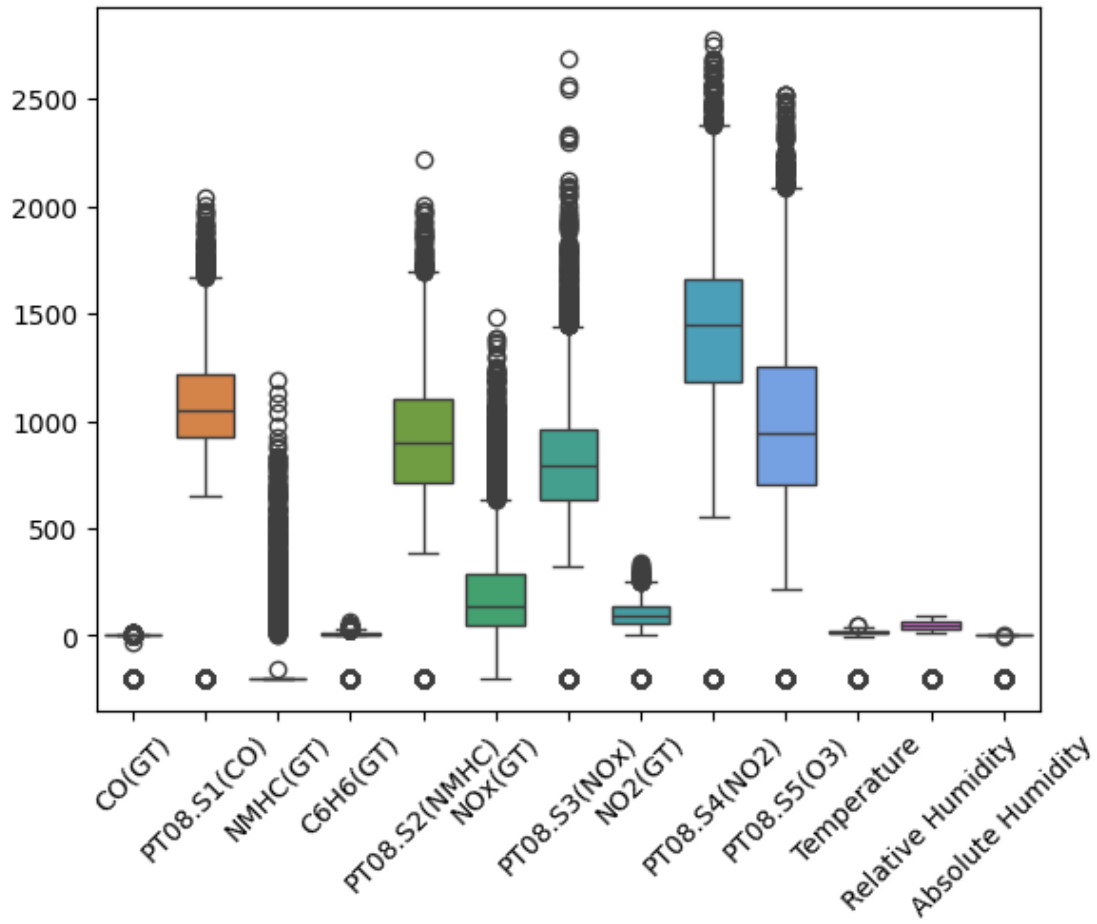
[24]:
```
df.columns
```

[24]:
```
Index(['CO(GT)', 'PT08.S1(CO)', 'NMHC(GT)', 'C6H6(GT)', 'PT08.S2(NMHC)',
       'NOx(GT)', 'PT08.S3(NOx)', 'NO2(GT)', 'PT08.S4(NO2)', 'PT08.S5(O3)',
       'Temperature', 'Relative Humidity', 'Absolute Humidity'],
      dtype='object')
```

[25]:
```
numeric_col=['CO(GT)', 'PT08.S1(CO)', 'NMHC(GT)', 'C6H6(GT)','PT08.S2(NMHC)',␣
 ↪'NOx(GT)', 'PT08.S3(NOx)', 'NO2(GT)', 'PT08.S4(NO2)','PT08.S5(O3)',␣
 ↪'Temperature', 'Relative Humidity', 'Absolute Humidity']
for col in numeric_col:
    cleaned_col=remove_outliers(df[col])
    df.loc[cleaned_col.index,col]=cleaned_col
```

[26]:
```
sns.boxplot(df)
plt.xticks(rotation=45)
```

[26]:
```
([0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12],
 [Text(0, 0, 'CO(GT)'),
  Text(1, 0, 'PT08.S1(CO)'),
  Text(2, 0, 'NMHC(GT)'),
  Text(3, 0, 'C6H6(GT)'),
  Text(4, 0, 'PT08.S2(NMHC)'),
  Text(5, 0, 'NOx(GT)'),
  Text(6, 0, 'PT08.S3(NOx)'),
  Text(7, 0, 'NO2(GT)'),
  Text(8, 0, 'PT08.S4(NO2)'),
  Text(9, 0, 'PT08.S5(O3)'),
  Text(10, 0, 'Temperature'),
  Text(11, 0, 'Relative Humidity'),
  Text(12, 0, 'Absolute Humidity')])
```



[27]:
```
from sklearn.preprocessing import MinMaxScaler
scaler=MinMaxScaler()
df[numeric_col]=scaler.fit_transform(df[numeric_col])
```

[28]:
```
df.head()
```

[28]:

	CO(GT)	PT08.S1(CO)	NMHC(GT)	C6H6(GT)	PT08.S2(NMHC)	NOx(GT)	\
0	0.956111	0.696429	0.251980	0.803565	0.516156	0.217987	
1	0.953280	0.666071	0.224622	0.794084	0.478459	0.180465	
2	0.954224	0.715179	0.207343	0.792567	0.471831	0.197141	
3	0.954224	0.703571	0.201584	0.793326	0.475559	0.221560	
4	0.951392	0.657143	0.180706	0.783087	0.429163	0.197141	

	PT08.S3(NOx)	NO2(GT)	PT08.S4(NO2)	PT08.S5(O3)	Temperature	\
0	0.435657	0.579630	0.635966	0.539111	0.873262	
1	0.476587	0.540741	0.591261	0.430408	0.872036	
2	0.464794	0.581481	0.589916	0.467866	0.866312	
3	0.448144	0.596296	0.599664	0.515241	0.862633	

```
4      0.487340  0.585185      0.568067     0.481087     0.863451

   Relative Humidity  Absolute Humidity
0           0.862141           0.992715
1           0.857984           0.992556
2           0.879806           0.992678
3           0.900589           0.992858
4           0.899203           0.992869
```

[29]:
```
y=df['Temperature']
X=df.drop('Temperature',axis=1)
```

[30]:
```
from sklearn.model_selection import train_test_split
from sklearn.metrics import r2_score,mean_squared_error,mean_absolute_error

X_train,X_test,y_train,y_test=train_test_split(X,y,test_size=0.
 ↪2,random_state=42)
```

[31]:
```
from sklearn.linear_model import LinearRegression
model=LinearRegression()
model.fit(X_train,y_train)
y_pred=model.predict(X_test)

print(f'R2_Score:',r2_score(y_test,y_pred))
print(f'MSE:',mean_squared_error(y_test,y_pred))
print(f'MAE:',mean_absolute_error(y_test,y_pred))
print(f"Accuracy:",r2_score(y_test,y_pred)*100)
```

```
R2_Score: 0.9950183148394404
MSE: 0.00015687880147062074
MAE: 0.009805649230813743
Accuracy: 99.50183148394403
```

[32]:
```
from sklearn.tree import DecisionTreeRegressor
model1= DecisionTreeRegressor()
model1.fit(X_train,y_train)
y_pred=model1.predict(X_test)

print(f'R2_Score:',r2_score(y_test,y_pred))
print(f'MSE:',mean_squared_error(y_test,y_pred))
print(f'MAE:',mean_absolute_error(y_test,y_pred))
print(f"Accuracy:",r2_score(y_test,y_pred)*100)
```

```
R2_Score: 0.9998768106823122
MSE: 3.879368504828717e-06
MAE: 0.0012406877441628033
Accuracy: 99.98768106823121
```

[33]:
```
from sklearn.ensemble import RandomForestRegressor
model3 = RandomForestRegressor()
```

[34]:
```
from sklearn.model_selection import cross_val_score
scores = cross_val_score(model3, X, y, cv=5, scoring='r2')
print("Mean R2 from Cross-validation:", scores.mean())
```

```
Mean R2 from Cross-validation: 0.9996287049356042
```

[35]:
```
new_data = [[0.95, 0.66, 0.22, 0.79, 0.47, 0.18, 0.47, 0.54, 0.59,0.43, 0.86, 0.
↪99]]
prediction = model1.predict(new_data)
print("Predicted Temperature for new data:", prediction[0])
```

```
Predicted Temperature for new data: 0.812346688470973
```

```
C:\Users\AMOL\AppData\Local\Programs\Python\Python313\Lib\site-
packages\sklearn\utils\validation.py:2739: UserWarning: X does not have valid
feature names, but DecisionTreeRegressor was fitted with feature names
  warnings.warn(
```

[ ]:
```
```