SMARTBRIDGE
Let's Bridge the Gap
a Veranda Enterprise

Data Collection and Preprocessing Phase

Date	07 JULY 2024
Team ID	739850
Project Title	Air Quality Index Analyzer using machine learning
Maximum Marks	6 Marks

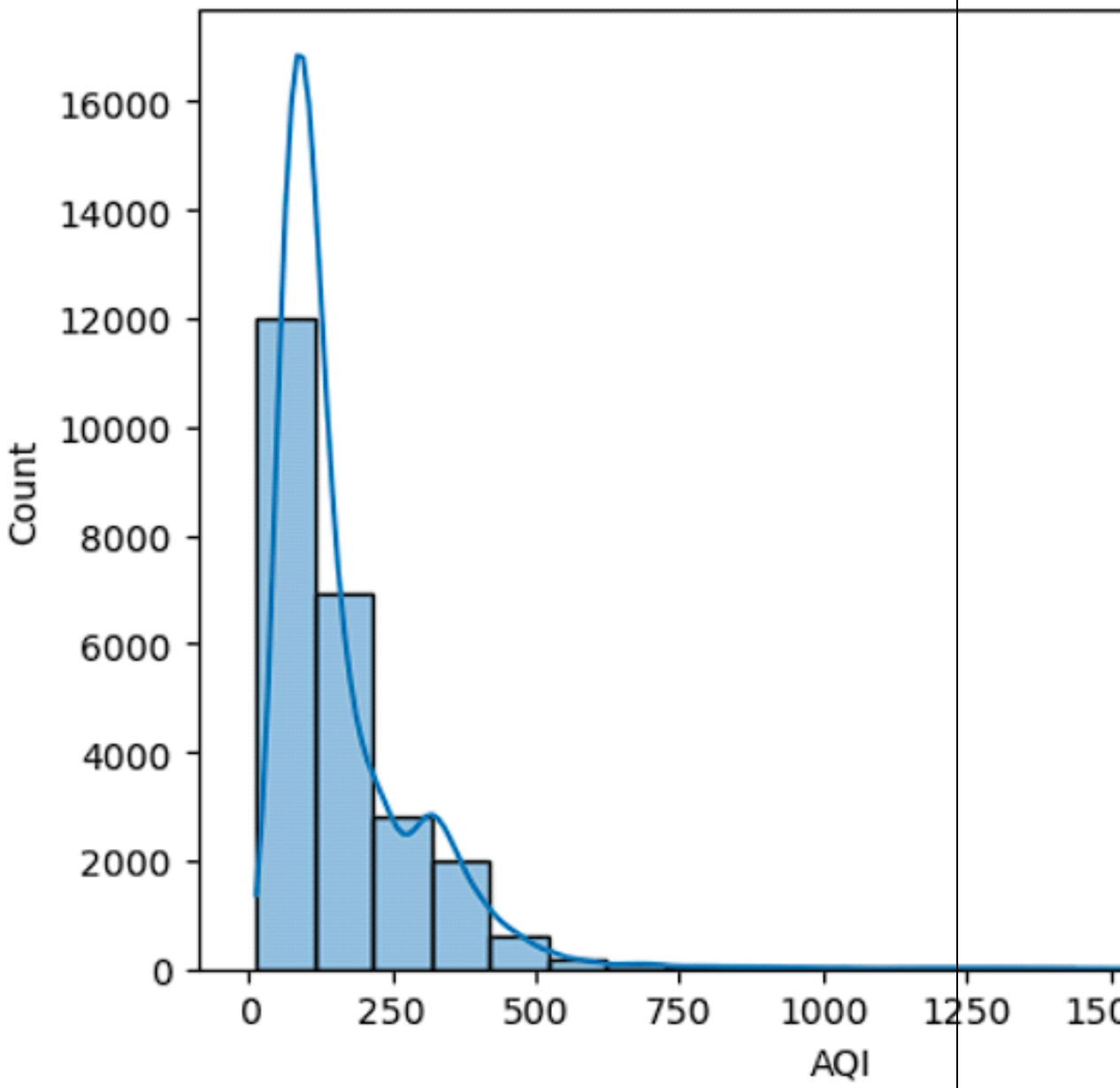
Data Exploration and Preprocessing Report

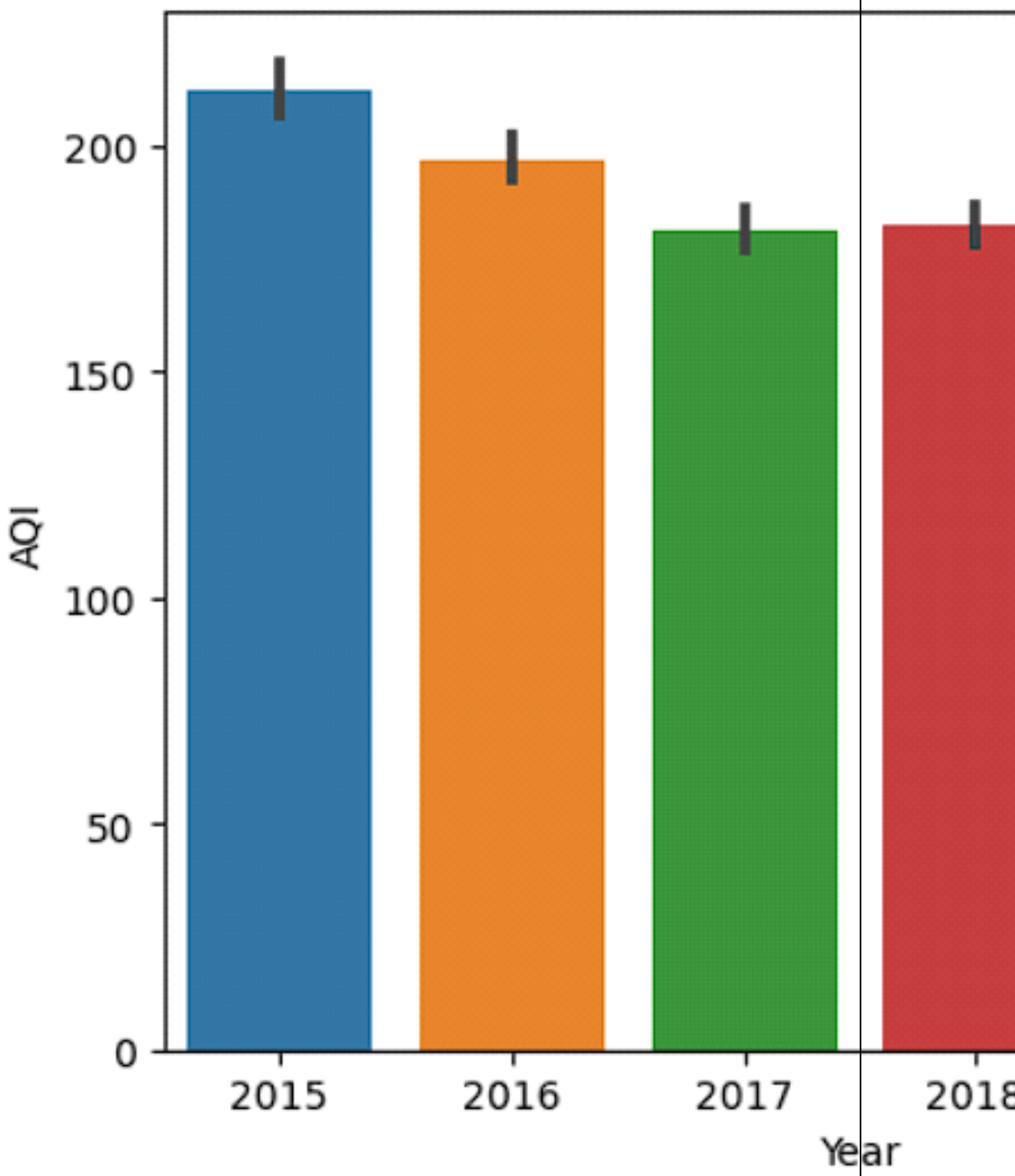
Identifies data sources, assesses quality issues like missing values and duplicates, and implements resolution plans to ensure accurate and reliable analysis

Section	Description
Data Overview	Basic statistics, dimensions, and structure of the data.

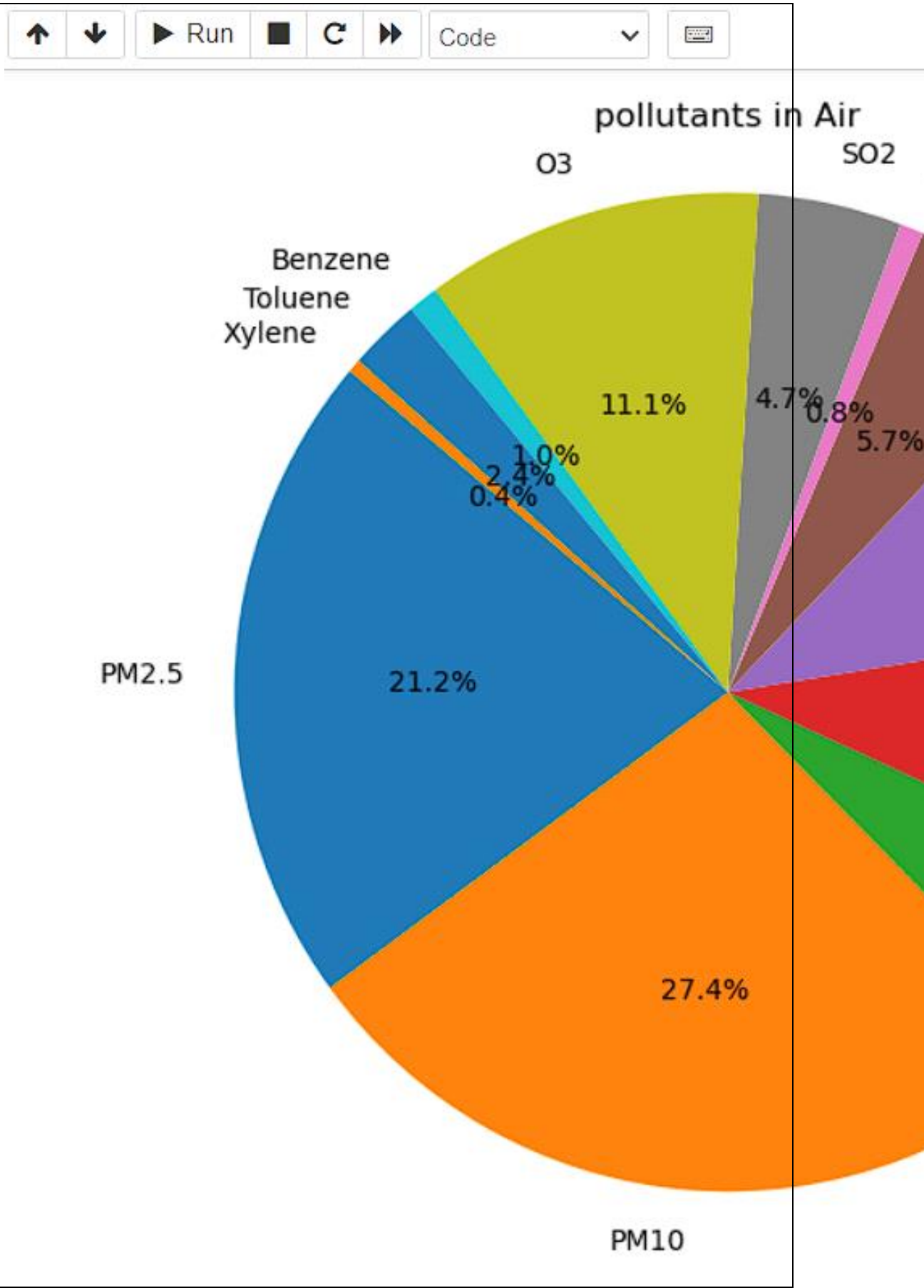
```
data_city.describe()
```

	PM2.5	PM10	NO	NO2	NOx	NH3	CO	SO2	O3	Benzene
count	24933.000000	18391.000000	25949.000000	25946.000000	25346.000000	19203.000000	27472.000000	25677.000000	25509.000000	23908.000000
mean	67.450578	118.127103	17.574730	28.560659	32.309123	23.483476	2.248598	14.531977	34.491430	3.280840
std	64.661449	90.605110	22.785846	24.474746	31.646011	25.684275	6.962884	18.133775	21.694928	15.811136
min	0.040000	0.010000	0.020000	0.010000	0.000000	0.010000	0.000000	0.010000	0.010000	0.000000
25%	28.820000	56.255000	5.630000	11.750000	12.820000	8.580000	0.510000	5.670000	18.860000	0.120000
50%	48.570000	95.680000	9.890000	21.690000	23.520000	15.850000	0.890000	9.160000	30.840000	1.070000
75%	80.590000	149.745000	19.950000	37.620000	40.127500	30.020000	1.450000	15.220000	45.570000	3.080000
max	949.990000	1000.000000	390.680000	362.210000	467.630000	352.890000	175.810000	193.860000	257.730000	455.030000

Univariate Analysis	Exploration of individual variables (mean, median, mode, etc.). 
Bivariate Analysis	Relationships between two variables (correlation, scatter plots).

	
Multivariate Analysis	Patterns and relationships involving multiple variables.

s


Run
Code
pollutants in Air
O3
SO2
Benzene
Toluene
Xylene
11.1%
4.7%
0.8%
5.7%
1.0%
2.4%
0.4%
PM2.5
21.2%
27.4%
PM10

Outliers and Anomalies	Identification and treatment of outliers.

```
26]: import pandas as pd
     import numpy as np
     import matplotlib.pyplot as plt

     def handle_outliers(df):
         # Plot boxplots before handling outliers
         plt.figure(figsize=(15, 10))
         df.boxplot(rot=90)
         plt.title('Boxplot Before Handling Outliers')
         plt.show()

         for column in df.columns:
             if pd.api.types.is_numeric_dtype(df[column]):
                 Q1 = df[column].quantile(0.25)
                 Q3 = df[column].quantile(0.75)
                 IQR = Q3 - Q1
                 lower_bound = Q1 - 1.5 * IQR
                 upper_bound = Q3 + 1.5 * IQR

                 # Cap the outliers
                 df[column] = np.where(df[column] < lower_bound,
                                       np.where(df[column] > uppe

         # Plot boxplots after handling outliers
         plt.figure(figsize=(15, 10))
         df.boxplot(rot=90)
         plt.title('Boxplot After Handling Outliers')
         plt.show()

         return df
```

Data Preprocessing Code Screenshots

Loading Data	Code to load the dataset into the preferred environment (e.g., Python, R).

In [48]:
```
X=data_city.drop('AQI',axis=1)
y=data_city['AQI']
```

In [49]:
```
X
```

Out[49]:

	City	PM2.5	PM10	NO	NO2	NOx
0	0	34.515	154.750	0.92	18.22	17.15
1	0	25.830	226.235	0.97	15.69	16.46
2	0	36.205	72.125	17.40	19.30	29.70
3	0	25.830	226.235	1.70	18.48	17.97
4	0	54.440	72.125	22.10	21.42	37.76
...	...	...	...	...	...	...
29526	25	15.020	50.940	7.68	25.06	19.54
29527	25	24.380	74.090	3.42	26.06	16.53
29528	25	22.910	65.730	3.45	29.53	18.33
29529	25	16.640	49.970	4.05	29.26	18.80
29530	25	15.000	66.000	0.40	26.85	14.05

29531 rows × 15 columns

Smart Internz

Handling Missing Data	Code for identifying and handling missing values. **Handling Null Values** 2]: ``` data_city.isna().sum() ``` 2]: ``` City 0 Date 0 PM2.5 4598 PM10 11140 NO 3582 NO2 3585 NOx 4185 NH3 10328 CO 2059 SO2 3854 O3 4022 Benzene 5623 Toluene 8041 Xylene 18109 AQI 4681 AQI_Bucket 4681 dtype: int64 ``` Data	Code for transforming variables (scaling, normalization).

Transformation	
Feature Engineering	Code for creating new features or modifying existing ones.

```
In [48]: X=data_city.drop('AQI',axis=1)
         y=data_city['AQI']
```

```
In [49]: X
```

Out[49]:

	City	PM2.5	PM10	NO	NO2	NOx	NH3	CO	SO2	O3	Benzene	To
0	0	34.515	154.750	0.92	18.22	17.15	8.975	0.92	27.640	85.635	0.00	
1	0	25.830	226.235	0.97	15.69	16.46	9.095	0.97	24.550	34.060	3.68	
2	0	36.205	72.125	17.40	19.30	29.70	6.880	2.86	29.070	30.700	6.80	
3	0	25.830	226.235	1.70	18.48	17.97	9.085	1.70	18.590	36.080	4.43	
4	0	54.440	72.125	22.10	21.42	37.76	7.915	2.86	29.545	39.310	7.01	
...	...	...	...	...	...	...	...	...	...	...	...	
29526	25	15.020	50.940	7.68	25.06	19.54	12.470	0.47	8.550	23.300	2.24	
29527	25	24.380	74.090	3.42	26.06	16.53	11.990	0.52	12.720	30.140	0.74	
29528	25	22.910	65.730	3.45	29.53	18.33	10.710	0.48	8.420	30.960	0.01	
29529	25	16.640	49.970	4.05	29.26	18.80	10.030	0.52	9.840	28.300	0.00	
29530	25	15.000	66.000	0.40	26.85	14.05	5.200	0.59	2.100	17.050	0.00	

29531 rows × 15 columns