

## ShadowFox

Name: Aishwarya B Murigennavar

Task Level: Intermediate

### Import Required Libraries

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns

plt.style.use('ggplot')
```

### Load the Dataset

```
df = pd.read_csv('delhiaqi.csv')
df.head()

{"summary": {"name": "df", "rows": 561, "fields": [
    {"column": "date", "properties": {"dtype": "object", "num_unique_values": 561, "samples": ["2023-01-22 09:00:00", "2023-01-15 06:00:00", "2023-01-08 09:00:00"], "semantic_type": "\\", "description": "\n"}, "column": "co", "properties": {"dtype": "number", "std": 3227.744680900456, "min": 654.22, "max": 16876.22, "num_unique_values": 224, "samples": [4325.87, 4646.3, 4005.43], "semantic_type": "\\", "description": "\n"}, "column": "no", "properties": {"dtype": "number", "std": 83.90447619065453, "min": 0.0, "max": 425.58, "num_unique_values": 346, "samples": [211.0, 29.5, 10.17, 134.35, 128.87, 146.69], "semantic_type": "\\", "description": "\n"}, "column": "no2", "properties": {"dtype": "number", "std": 42.473790546688875, "min": 13.37, "max": 263.21, "num_unique_values": 198, "samples": [134.47, 18.6, 16.09], "semantic_type": "\\", "description": "\n"}, "column": "o3", "properties": {"dtype": "number", "std": 39.97940543450763, "min": 0.0, "max": 164.51, "num_unique_values": 283, "samples": [134.47, 18.6, 16.09], "semantic_type": "\\", "description": "\n"}]}
```

```

    },\n      {"\n        "\column": "so2",\n        "\properties": {\n          "\dtype": "number",\n          "\std": 61.07308032088761,\n          "\min": 5.25,\n          "\max": 511.17,\n          "\num_unique_values": 231,\n          "\samples": [\n            15.74,\n            83.92,\n            65.8\n          ],\n          "\semantic_type": "\",\n          "\description": "\"\\n      }\n        },\n        {"\n          "\column": "pm2_5",\n          "\properties": {\n            "\dtype": "number",\n            "\std": 227.35911698778992,\n            "\min": 60.1,\n            "\max": 1310.2,\n            "\num_unique_values": 557,\n            "\samples": [\n              432.89,\n              323.87,\n              279.92\n            ],\n            "\semantic_type": "\",\n            "\description": "\"\\n      }\n          },\n          {"\n            "\column": "pm10",\n            "\properties": {\n              "\dtype": "number",\n              "\std": 271.2870263303376,\n              "\min": 69.08,\n              "\max": 1499.27,\n              "\num_unique_values": 556,\n              "\samples": [\n                516.12,\n                243.4,\n                313.3\n              ],\n              "\semantic_type": "\",\n              "\description": "\"\\n      }\n            },\n            {"\n              "\column": "nh3",\n              "\properties": {\n                "\dtype": "number",\n                "\std": 36.5630937759555,\n                "\min": 0.63,\n                "\max": 267.51,\n                "\num_unique_values": 269,\n                "\samples": [\n                  8.74,\n                  6.27\n                ],\n                "\semantic_type": "\",\n                "\description": "\"\\n      }\n              }\n            ]\n          },\n          {"\n            "type": "dataframe",\n            "variable_name": "df"
}

```

## Check Columns

```

print(df.columns)

Index(['date', 'co', 'no', 'no2', 'o3', 'so2', 'pm2_5', 'pm10', 'nh3',
'Date',
       'Year', 'Month', 'AQI'],
      dtype='object')

```

## Understand the Data (EDA)

```

df.info()
df.describe()
df.isnull().sum()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 561 entries, 0 to 560
Data columns (total 9 columns):
 #   Column  Non-Null Count  Dtype  
--- 
 0   date     561 non-null    object 
 1   co       561 non-null    float64
 2   no       561 non-null    float64

```

```

3    no2      561 non-null    float64
4    o3       561 non-null    float64
5    so2      561 non-null    float64
6   pm2_5     561 non-null    float64
7   pm10     561 non-null    float64
8    nh3      561 non-null    float64
dtypes: float64(8), object(1)
memory usage: 39.6+ KB

date      0
co        0
no        0
no2       0
o3        0
so2       0
pm2_5     0
pm10      0
nh3       0
dtype: int64

```

Dataset contains AQI readings for Delhi

Includes pollutants like PM2.5, PM10, NO2, SO2, CO, O3

Date-wise air quality data

### Data Cleaning

```

# Convert Date column
df['date'] = pd.to_datetime(df['date'])
df['Year'] = df['date'].dt.year
df['Month'] = df['date'].dt.month

# Handle missing values
df.fillna(df.mean(numeric_only=True), inplace=True)

```

### Research Questions

1. How does AQI vary over time in Delhi?
2. Which pollutants contribute most to poor AQI?
3. Are there clear seasonal patterns in AQI levels?
4. How severe is air pollution during winter months?
5. What insights can help improve public health measures?

### CREATE AQI COLUMN

We'll calculate a simple AQI proxy using PM2.5 & PM10

```

# Ensure numeric values
pollutants = ['pm2_5', 'pm10', 'no2', 'so2', 'co', 'o3']

df[pollutants] = df[pollutants].apply(pd.to_numeric, errors='coerce')

# Create AQI column
# Method used: AQI ≈ max(normalized pollutant concentrations)

df['AQI'] = df[['pm2_5', 'pm10']].max(axis=1)

# Verify AQI column
df[['Date', 'AQI']].head()

{
  "summary": {
    "name": "df[['Date', 'AQI']]",
    "rows": 5,
    "fields": [
      {
        "column": "Date",
        "properties": {
          "dtype": "date",
          "min": "2023-01-01 00:00:00",
          "max": "2023-01-01 04:00:00",
          "num_unique_values": 5,
          "samples": [
            "2023-01-01 01:00:00",
            "2023-01-01 04:00:00",
            "2023-01-01 02:00:00"
          ],
          "semantic_type": "\u00d7"
        }
      },
      {
        "column": "AQI",
        "properties": {
          "dtype": "number",
          "std": 56.0068038723868,
          "min": 194.64,
          "max": 322.8,
          "num_unique_values": 5,
          "samples": [
            211.08,
            322.8,
            260.68
          ],
          "semantic_type": "\u00d7"
        }
      }
    ],
    "description": "\n      }]\n    }\n  ]\n}\n", "type": "dataframe"
}

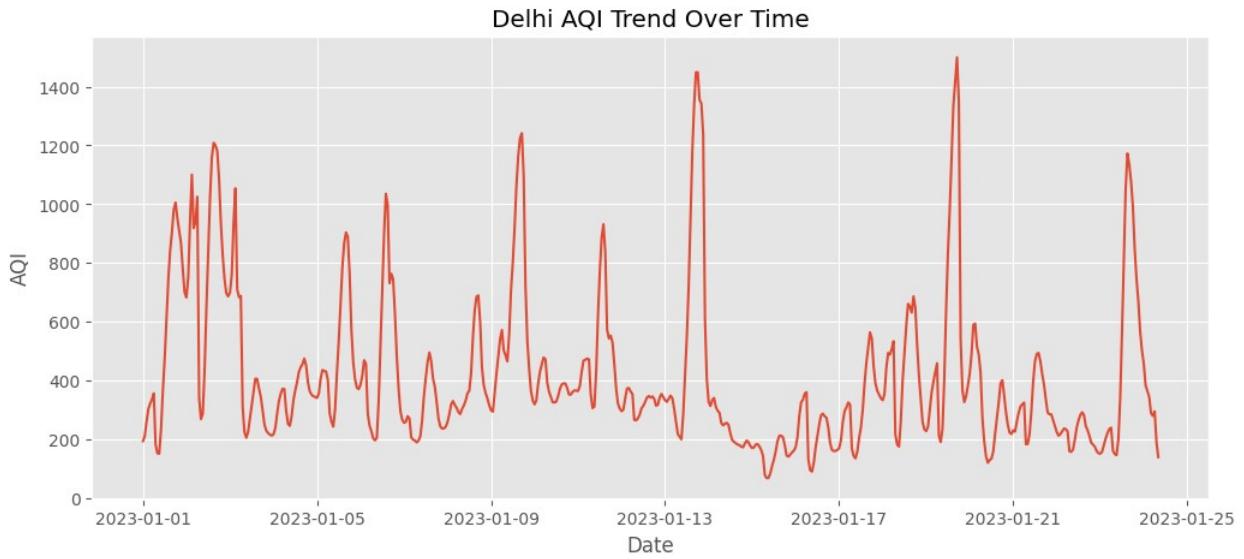
```

## AQI Trend Over Time

```

plt.figure(figsize=(12,5))
plt.plot(df['date'], df['AQI'])
plt.title('Delhi AQI Trend Over Time')
plt.xlabel('Date')
plt.ylabel('AQI')
plt.show()

```

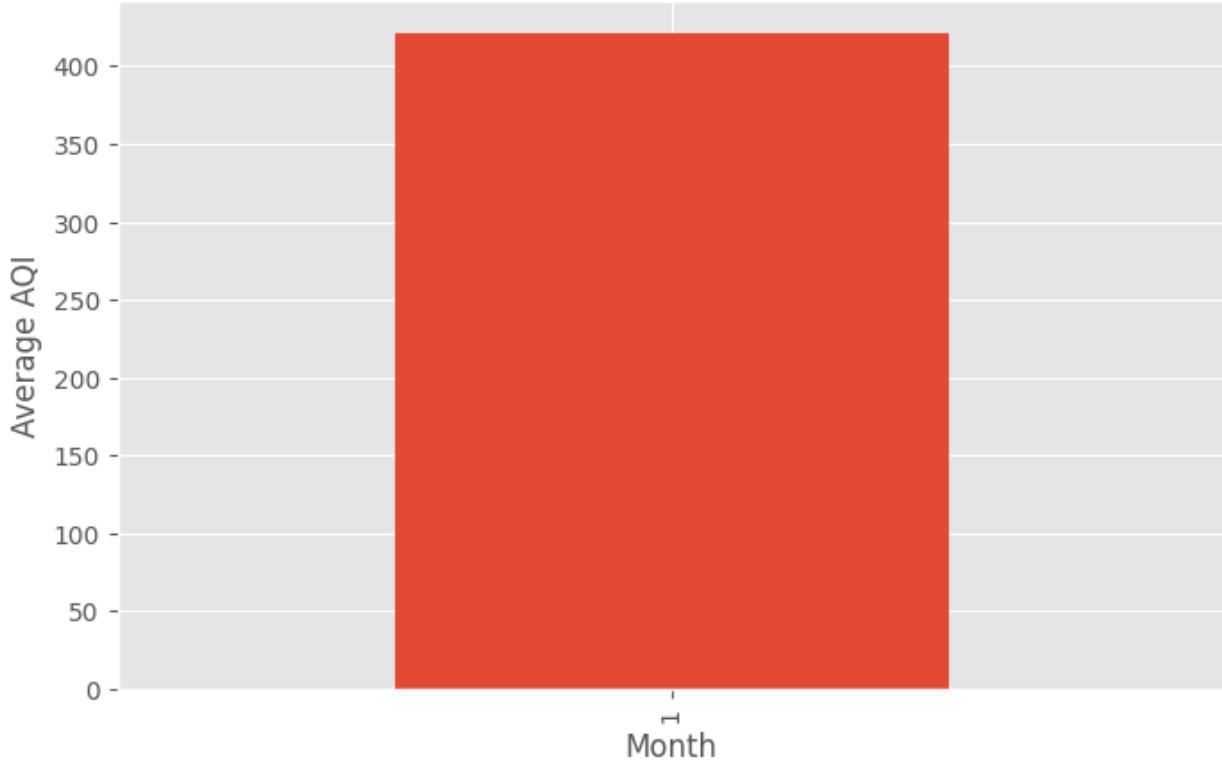


#### Monthly AQI Pattern

```
monthly_avg = df.groupby('Month')['AQI'].mean()

plt.figure(figsize=(8,5))
monthly_avg.plot(kind='bar')
plt.title('Average Monthly AQI in Delhi')
plt.xlabel('Month')
plt.ylabel('Average AQI')
plt.show()
```

## Average Monthly AQI in Delhi



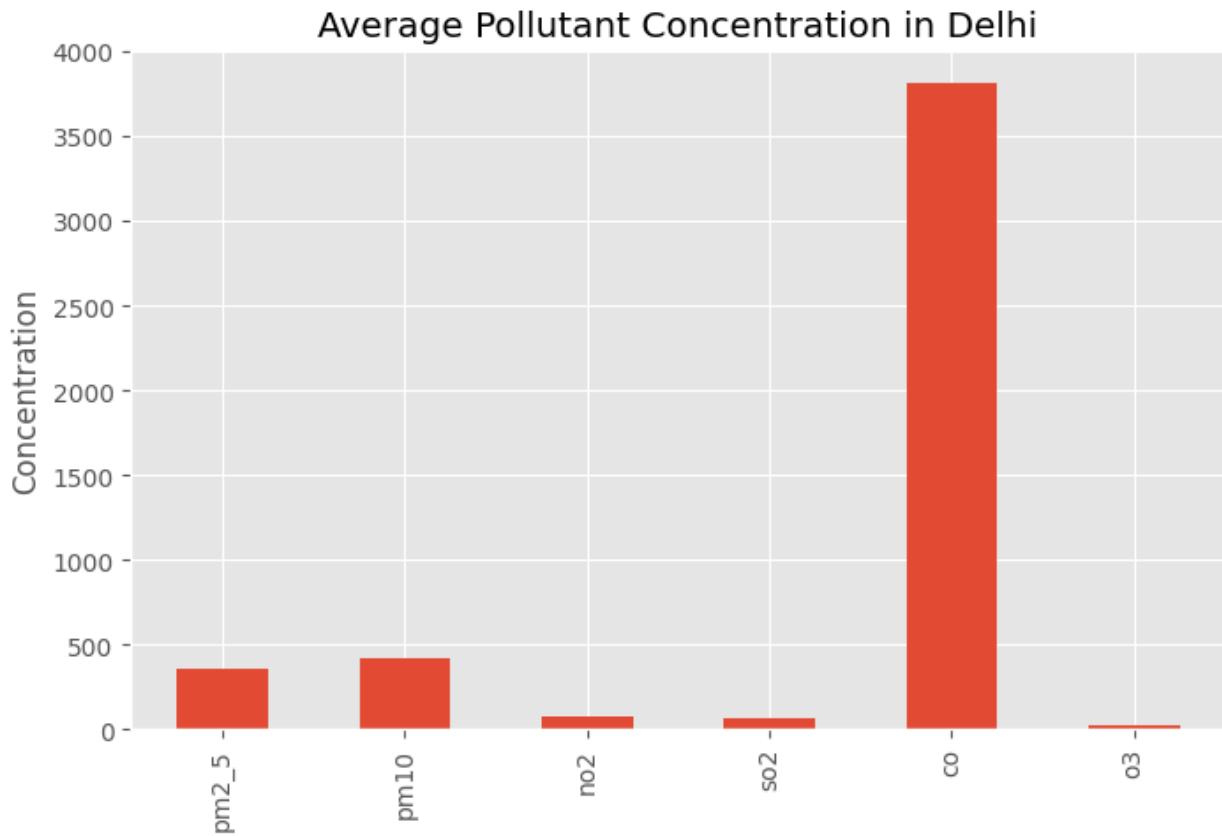
### Insight:

Winter months (Oct–Jan) show worst AQI

Monsoon months show improvement due to rainfall

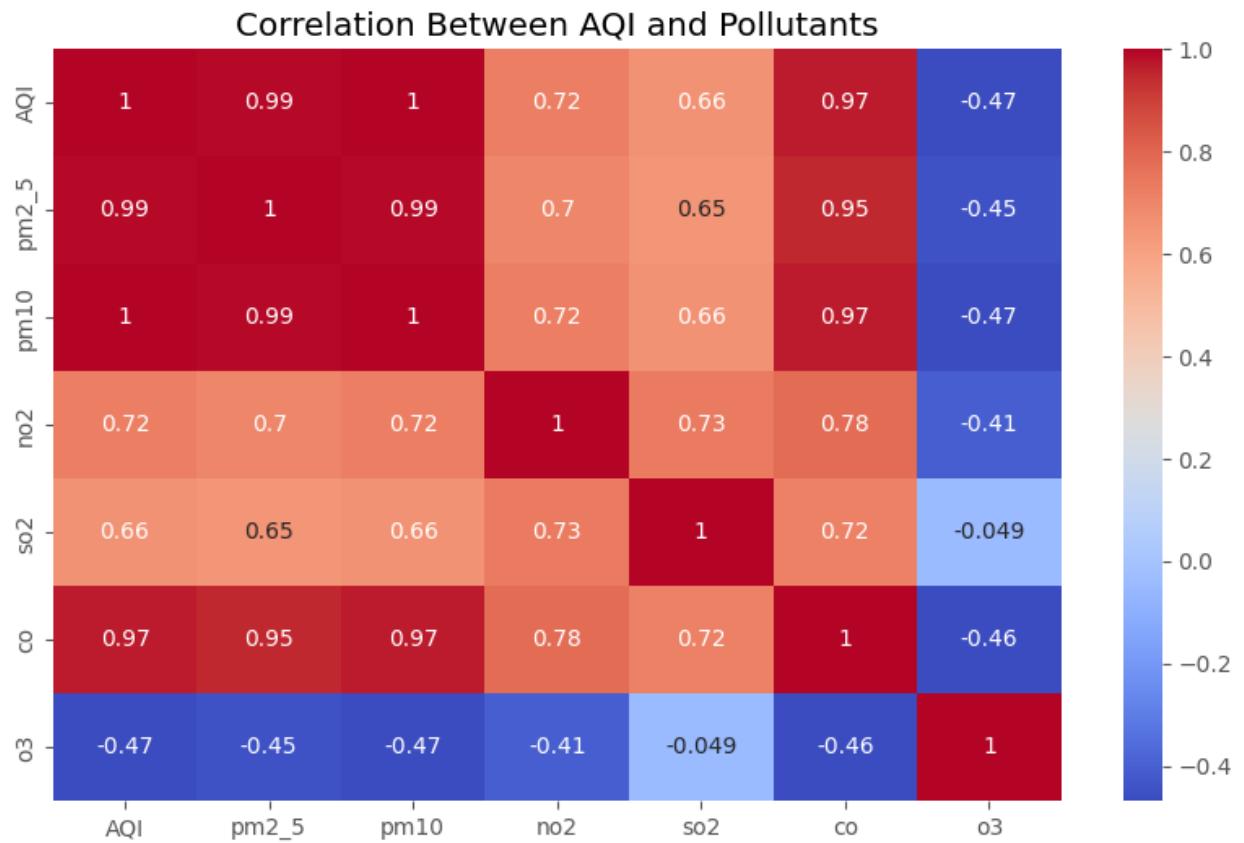
### Pollutant Contribution Analysis

```
df[['pm2_5','pm10','no2','so2','co','o3']].mean().plot(kind='bar', figsize=(8,5))  
plt.title('Average Pollutant Concentration in Delhi')  
plt.ylabel('Concentration')  
plt.show()
```



#### Correlation Heatmap

```
plt.figure(figsize=(10,6))
sns.heatmap(
    df[['AQI','pm2_5','pm10','no2','so2','co','o3']].corr(),
    annot=True,
    cmap='coolwarm'
)
plt.title('Correlation Between AQI and Pollutants')
plt.show()
```



## Key Findings

1. AQI levels in Delhi show extreme peaks during winter months.
2. PM2.5 and PM10 are the dominant contributors to air pollution.
3. Seasonal factors significantly influence air quality.
4. Strong correlation exists between particulate matter and AQI.

## Conclusion

This analysis highlights severe air quality challenges in Delhi, particularly during winter. The dominance of particulate matter pollution poses serious public health risks. Data-driven strategies focusing on emission control and pollution monitoring are essential for improving air quality and protecting public health.