Assignment 7

Data Loading, Storage and File Formats

Problem Statement: Analyzing Sales Data from Multiple File Formats Dataset:

Sales data in multiple file formats (e.g., CSV, Excel, JSON)

Hardware Requirement:

- 6 GB free disk space.
- 2 GB of RAM, plus additional RAM for virtual machines.
- 6 GB disk space for the host, plus the required disk space for the virtual machine(s).Virtualization is available with the KVM hypervisor
- Intel 64 andAMD64 architecture.

Software Requirement:

Jyupter Notebook/Ubuntu

Theory:

Analyzing Sales Data from Multiple File Formats:

Analyzing sales data from various file formats is a common challenge in data analytics and business intelligence. Sales data can be collected and stored in different formats, such as CSV, Excel, JSON, or even databases. The primary goal of this analysis is to consolidate, clean, and analyze sales data to derive valuable insights for informed decision-making.

Data Integration and Cleaning:

One of the initial tasks is to integrate data from various formats into a unified dataset. This may involve converting Excel files, parsing JSON, and loading CSV data. Cleaning the data is crucial to address issues like missing values, duplicates, and inconsistent formatting.

Sales Performance Analysis:

Once the data is harmonized and cleaned, it can be used to perform sales performance analysis. This involves tasks like calculating revenue, profit margins, and identifying topselling products or regions. Visualization techniques are often employed to present these insights effectively.

Business Decision Support:

Analyzing sales data from multiple file formats serves as a foundation for informed business decisions. Retailers can use these insights to improve inventory management, optimize marketing strategies, and enhance overall profitability. The ability to work with diverse data

formats equips analysts and data scientists with a versatile skill set, ensuring they can handle data from a wide range of sources.

Description: The goal is to load and analyze sales data from different file formats, including CSV, Excel, and JSON, and perform data cleaning, transformation, and analysis on the dataset.

Tasks to Perform:

1. Obtain sales data files in various formats, such as CSV, Excel, and JSON.
2. Load the sales data from each file format into the appropriate data structures or dataframes.
3. Explore the structure and content of the loaded data, identifying any inconsistencies, missing values, or data quality issues.
4. Perform data cleaning operations, such as handling missing values, removing duplicates, or correcting inconsistencies.
5. Convert the data into a unified format, such as a common dataframe or data structure, to enable seamless analysis.
6. Perform data transformation tasks, such as merging multiple datasets, splitting columns, or deriving new variables.
7. Analyze the sales data by performing descriptive statistics, aggregating data by specific variables, or calculating metrics such as total sales, average order value, or product category distribution.
8. Create visualizations, such as bar plots, pie charts, or box plots, to represent the sales data and gain insights into sales trends, customer behavior, or product performance.

Conclusion:

In this lab, we addressed the challenge of analyzing sales data stored in multiple file formats, including CSV, Excel, and JSON. Our journey began by obtaining these data files and loading them into appropriate data structures, followed by a meticulous exploration of their structure and content to ensure data quality.To ensure data consistency, we performed cleaning and transformation operations, which included handling missing values, removing duplicates, and converting the data into a unified format. This prepared the dataset for analysis, where we calculated metrics, aggregated data, and visualized insights using techniques such as bar plots, pie charts, and box plots.

The ability to work with diverse data formats and extract valuable insights equips data analysts and scientists with versatile skills, enabling them to make informed business decisions, optimize marketing strategies, and enhance overall profitability.

Assignment 8

Interacting with Web APIs

Problem Statement: Analyzing Weather Data from OpenWeatherMap API

Dataset: Weather data retrieved from OpenWeatherMap API

Hardware Requirement:

6 GB free disk space.

2 GB of RAM, plus additional RAM for virtual machines.

6 GB disk space for the host, plus the required disk space for the virtual machine(s).Virtualization is available with the KVM hypervisor Intel 64 andAMD64 architecture.

Software Requirement:

Jyupter Notebook/Ubuntu

Theory:

OpenWeatherMap API
The OpenWeatherMap API is a robust service that offers access to comprehensive weather data worldwide. Users can retrieve information such as current weather conditions, weather forecasts, historical weather data, and weather maps. To access this data, users need to register and obtain an API key, which is used to authenticate requests to the API.

One of the notable strengths of the OpenWeatherMap API is its global coverage. It provides weather information for locations all over the world, making it a valuable resource for a wide range of applications, from local weather apps to global weather analysis.The API is accessible through simple HTTP requests, with endpoints for current weather, forecasts, historical data, and more. Responses are typically delivered in JSON format, making it easy to parse and integrate into various applications and projects.

OpenWeatherMap offers both free and paid subscription plans, with the free tier having some limitations on the number of requests and available features. This flexibility allows developers and data scientists to choose a plan that suits their specific needs.

Customization is another advantage of this API, as it permits users to tailor their data requests according to the parameters of interest, enabling more focused and relevant data retrieval.

The goal is to interact with the OpenWeatherMap API to retrieve weather data for a specific location and perform data modeling and visualization to analyze weather patterns over time.

Tasks to Perform:

1.Register and obtain API key from OpenWeatherMap.

2.Interact with the OpenWeatherMap API using the API key to retrieve weather data for a specific location.

3.Extract relevant weather attributes such as temperature, humidity, wind speed, and precipitation from the API response.Clean and preprocess the retrieved data, handling missing values or inconsistent formats.

4.Perform data modeling to analyze weather patterns, such as calculating average temperature, maximum/minimum values, or trends over time.

5.Visualize the weather data using appropriate plots, such as line charts, bar plots, or scatter plots, to represent temperature changes, precipitation levels, or wind speed variations.

6.Apply data aggregation techniques to summarize weather statistics by specific time periods (e.g., daily, monthly, seasonal).

7.Incorporate geographical information, if available, to create maps or geospatial visualizations representing weather patterns across different locations.

8.Explore and visualize relationships between weather attributes, such as temperature and humidity, using correlation plots or heatmaps.

Conclusion:

In this lab, we worked with the OpenWeatherMap API to extract, analyze, and visualize weather data. We obtained an API key, accessed weather information, and processed it by cleaning and modeling the data. Our visualizations, including line charts and scatter plots, provided insights into temperature fluctuations, precipitation, and wind speed. Data aggregation techniques summarized weather statistics over time, and geographical data allowed us to map weather patterns. Through correlation plots, we explored relationships between weather attributes. This lab's skills and knowledge can be applied in various fields, such as weather forecasting and climate analysis, offering a deeper understanding of our environment.

Assignment 9

Data Cleaning and Preparation

Problem Statement: Analyzing Customer Churn in a Telecommunications Company

Dataset: "Telecom_Customer_Churn.csv" Hardware

Requirement:

- 6 GB free disk space.
- 2 GB RAM.
- 2 GB of RAM, plus additional RAM for virtual machines.
- 6 GB disk space for the host, plus the required disk space for the virtual machine(s).
- Virtualization is available with the KVM hypervisor
- Intel 64 and AMD64 architectures

Software Requirement:

Juypter Notebook/Ubuntu

Theory:

Customer churn is the rate at which customers discontinue services, and reducing it is crucial for businesses. This is achieved by understanding customer needs, improving services, and using predictive analytics to identify potential churn and take preventive measures.

Causes of Customer Churn:

1. Dissatisfaction with services or customer support.
2. Attracted by better deals or offers from competitors.
3. Changing needs or preferences.
4. Expiry of contracts.
5. Geographical relocation.
6. Economic factors and cost-cutting.

Reducing Churn:

1. Enhance customer service and satisfaction.
2. Personalize offers and incentives.
3. Ensure network quality.
4. Implement loyalty programs.

5. Utilize data-driven insights for proactive retention efforts.

Description of Dataset:

The dataset contains information about customers of a telecommunications company and whether they have churned (i.e., discontinued their services). The dataset includes various attributes of the customers, such as their demographics, usage patterns, and account information. The goal is to perform data cleaning and preparation to gain insights into the factors that contribute to customer churn.

Tasks to Perform:

1. Import the "Telecom_Customer_Churn.csv" dataset.
2. Explore the dataset to understand its structure and content.
3. Handle missing values in the dataset, deciding on an appropriate strategy.
4. Remove any duplicate records from the dataset.
5. Check for inconsistent data, such as inconsistent formatting or spelling variations, and standardize it.
6. Convert columns to the correct data types as needed.
7. Identify and handle outliers in the data.

Conclusion:

In this lab, we conducted data cleaning and preparation for analyzing customer churn in a telecommunications company using the "Telecom_Customer_Churn.csv" dataset. Our tasks included handling missing values, removing duplicates, ensuring data consistency, converting data types as needed, and managing outliers. These steps have readied the dataset for advanced analysis, enabling us to identify and address factors contributing to customer churn, a vital aspect of improving customer retention and business performance in the telecommunications sector.

Assignment 10

Data Wrangling

Aim: Data Wrangling

Problem Statement: Data Wrangling on Real Estate Market

Dataset: "RealEstate_Prices.csv"

.

Hardware Requirement:

- 6 GB free disk space.
- 2 GB RAM.
- 2 GB of RAM, plus additional RAM for virtual machines.
- 6 GB disk space for the host, plus the required disk space for the virtual machine(s).
- Virtualization is available with the KVM hypervisor
- Intel 64 and AMD64 architectures

Software Requirement:

Juypter Notebook/Ubuntu

Theory:

Data Wrangling:

Data wrangling, also known as data munging, is a fundamental process in the field of data science and analytics. It involves the transformation, cleaning, and restructuring of raw, messy data into a more organized and usable format for analysis. The primary goals of data wrangling are to enhance data quality, ensure data consistency, and prepare the data for various analytical tasks.

The process of data wrangling typically begins with data acquisition, where data is collected from diverse sources, including databases, web services, spreadsheets, or even text files. This data can be in various formats, such as CSV, JSON, or XML, and may contain missing values, duplicate entries, inconsistent formatting, and outliers. Data wrangling aims to address these issues by performing tasks such as imputing missing values, removing duplicates, standardizing formats, and dealing with outliers.

Once the data is cleaned and preprocessed, it can be transformed to extract relevant features or variables. Data wrangling often involves creating new features, aggregating data, or

reshaping data to fit the specific requirements of an analysis or modeling task. This process is essential for making the data more suitable for machine learning, statistical analysis, and visualization, ultimately leading to more meaningful insights and informed decision-making.

Description of dataset: The dataset contains information about housing prices in a specific real estate market. It includes various attributes such as property characteristics, location, sale prices, and other relevant features. The goal is to perform data wrangling to gain insights into the factors influencing housing prices and prepare the dataset for further analysis or modeling.

Tasks to Perform:

1. Import the "RealEstate_Prices.csv" dataset. Clean column names by removing spaces, special characters, or renaming them for clarity.
2. Handle missing values in the dataset, deciding on an appropriate strategy (e.g., imputation or removal).
3. Perform data merging if additional datasets with relevant information are available (e.g., neighborhood demographics or nearby amenities).
4. Filter and subset the data based on specific criteria, such as a particular time period, property type, or location.
5. Handle categorical variables by encoding them appropriately (e.g., one-hot encoding or label encoding) for further analysis.
6. Aggregate the data to calculate summary statistics or derived metrics such as average sale prices by neighborhood or property type.
7. Identify and handle outliers or extreme values in the data that may affect the analysis or modeling process.

Conclusion:

In this data wrangling lab focused on the real estate market dataset, "RealEstate_Prices.csv," we undertook a series of crucial data preparation tasks. We began by cleaning column names for clarity and handled missing values using suitable strategies. If additional datasets were available, we merged them for enrichment. Data filtering and subsetting were performed to focus our analysis, while categorical variables were adequately encoded. Aggregation provided summary statistics, such as average sale prices, by neighborhood or property type. The identification and handling of outliers ensured data quality. These steps collectively prepared the dataset for in-depth analysis, allowing us to gain insights into the factors influencing housing prices and make informed decisions in the real estate market.

Assignment 11

Data Visualization using matplotlib

Problem Statement: Analyzing Air Quality Index (AQI) Trends in a City Dataset:

"City_Air_Quality.csv"

Hardware Requirement:

- 6 GB free disk space.
- 2 GB RAM.
- 2 GB of RAM, plus additional RAM for virtual machines.
- 6 GB disk space for the host, plus the required disk space for the virtual machine(s).
- Virtualization is available with the KVM hypervisor
- Intel 64 and AMD64 architectures

Software Requirement:

Juypter Notebook/Ubuntu

Theory:

Air Quality Index (AQI) Analysis:

The Air Quality Index (AQI) is a vital tool used to assess and communicate the quality of the air in a specific location. It measures the concentration of major air pollutants, including ground-level ozone, particulate matter (PM2.5 and PM10), carbon monoxide (CO), sulfur dioxide (SO2), and nitrogen dioxide (NO2). Each pollutant is assigned a sub-index, and the overall AQI value is determined by the highest sub-index, reflecting the most critical pollutant. AQI values typically range from 0 to 500, with lower values indicating better air quality and higher values indicating more pollution.

Key Factors and Interpretation:

The AQI serves as a critical tool for governments, environmental agencies, and the public. It informs individuals about the safety of the air they breathe and helps them make decisions to protect their health. The AQI provides clear categories to interpret air quality, ranging from "Good" (0-50) to "Hazardous" (301-500), making it easy for the public to understand the potential health risks. Pollution sources, weather conditions, and geographical factors can impact AQI values. Analyzing AQI data can reveal air quality trends over time and provide insights into the effectiveness of pollution control measures.

Applications and Impact:

Analyzing AQI data is essential for various applications, including environmental policy development, public health, urban planning, and climate change studies. It helps in identifying areas with air quality problems and guiding pollution control initiatives. Moreover, it aids in raising public awareness about air pollution and its health consequences. As air quality continues to be a global concern, AQI analysis plays a pivotal role in safeguarding public health, reducing environmental impacts, and creating sustainable, cleaner living environments.

Description of dataset : The dataset contains information about air quality measurements in a specific city over a period of time. It includes attributes such as date, time, pollutant levels (e.g., PM2.5, PM10, CO), and the Air Quality Index (AQI) values. The goal is to use the matplotlib library to create visualizations that effectively represent the AQI trends and patterns for different pollutants in the city.

Tasks to Perform:

1. Import the "City_Air_Quality.csv" dataset.
2. Explore the dataset to understand its structure and content.
3. Identify the relevant variables for visualizing AQI trends, such as date, pollutant levels, and AQI values.
4. Create line plots or time series plots to visualize the overall AQI trend over time.
5. Plot individual pollutant levels (e.g., PM2.5, PM10, CO) on separate line plots to visualize their trends over time.
6. Use bar plots or stacked bar plots to compare the AQI values across different dates or time periods.
7. Create box plots or violin plots to analyze the distribution of AQI values for different pollutant categories.
8. Use scatter plots or bubble charts to explore the relationship between AQI values and pollutant levels.
9. Customize the visualizations by adding labels, titles, legends, and appropriate color schemes.

Conclusion:

In this lab, we utilized the matplotlib library to visualize Air Quality Index (AQI) trends in a specific city using the "City_Air_Quality.csv" dataset. Our tasks included creating line plots to depict the overall AQI trend over time, separate line plots for individual pollutant levels, bar plots for comparing AQI values across dates, and box plots to understand the distribution of AQI values for different pollutant categories. These visualizations provided essential insights into air quality, offering a basis for informed decision-making in environmental policy and public health.

By leveraging data visualization techniques, this lab facilitated a comprehensive understanding of AQI trends, helping assess air quality and its impact on public health and the environment in the studied city.

Assignment 12

Data Aggregation

Problem Statement: Analyzing Sales Performance by Region in a Retail Company Dataset:

"Retail_Sales_Data.csv"

Hardware Requirement:

- 6 GB free disk space.
- 2 GB RAM.
- 2 GB of RAM, plus additional RAM for virtual machines.
- 6 GB disk space for the host, plus the required disk space for the virtual machine(s).
- Virtualization is available with the KVM hypervisor
- Intel 64 and AMD64 architectures

Software Requirement:

Juypter Notebook/Ubuntu

Theory:

Analyzing Sales Performance by Region:

Analyzing sales performance by region is a crucial aspect of retail company operations. This analysis involves evaluating sales data from different geographical areas to identify trends, strengths, and areas for improvement. By segmenting sales data by region, retailers gain insights into which locations are thriving and which might require targeted strategies for growth.

Key Insights and Benefits:

This analysis provides several key benefits. It helps retailers allocate resources efficiently, optimize inventory management, and tailor marketing campaigns to suit regional preferences. Additionally, understanding sales performance by region aids in demand forecasting, enabling retailers to anticipate customer needs and react proactively to market dynamics.
Data Sources and Metrics:

Retailers typically rely on point-of-sale (POS) systems and customer databases to collect sales data. Metrics such as sales revenue, product performance, customer demographics, and market trends are analyzed. This information is instrumental in developing strategies to

enhance sales performance, increase customer satisfaction, and drive profitability across various regions.

Description of dataset: The dataset contains information about sales transactions in a retail company. It includes attributes such as transaction date, product category, quantity sold, and sales amount. The goal is to perform data aggregation to analyze the sales performance by region and identify the top-performing regions.

Tasks to Perform:

1. Import the "Retail_Sales_Data.csv" dataset.
2. Explore the dataset to understand its structure and content.
3. Identify the relevant variables for aggregating sales data, such as region, sales amount, and product category.
4. Group the sales data by region and calculate the total sales amount for each region.
5. Create bar plots or pie charts to visualize the sales distribution by region.
6. Identify the top-performing regions based on the highest sales amount.
7. Group the sales data by region and product category to calculate the total sales amount for each combination.
8. Create stacked bar plots or grouped bar plots to compare the sales amounts across different regions and product categories.

Conclusion:

In this lab, we focused on analyzing sales performance by region in a retail company using the "Retail_Sales_Data.csv" dataset. We started by importing the dataset and identifying key variables such as region and sales amount. By grouping and aggregating the data, we calculated total sales amounts by region and visualized the distribution. We also pinpointed the top-performing regions and compared sales across regions and product categories. This analysis equips retailers with valuable insights for resource allocation, marketing strategies, and optimizing profitability. Understanding regional sales dynamics is pivotal in ensuring retail success and customer satisfaction.

No.1

	ORDERNUMBER	QUANTITYORDERED	PRICEEACH	ORDERLINENUMBER	SALES	ORDERDATE	STATUS	QTR_ID	MONTH_ID	YEAR_ID	...	ADDRESSL
2818	10350	20	100.00	15	2244.40	12/2/2004 0:00	Shipped	4	12	2004	...	C/ Moralz
2819	10373	29	100.00	1	3978.51	1/31/2005 0:00	Shipped	1	1	2005	...	Torika
2820	10386	43	100.00	4	5417.57	3/1/2005 0:00	Resolved	1	3	2005	...	C/ Moralz
2821	10397	34	62.24	1	2116.16	3/28/2005 0:00	Shipped	1	3	2005	...	1 rue Al Lor
2822	10414	47	65.52	9	3079.44	5/6/2005 0:00	On Hold	2	5	2005	...	8616 Spini

	ORDERNUMBER	QUANTITYORDERED	PRICEEACH	ORDERLINENUMBER	SALES	QTR_ID	MONTH_ID	YEAR_ID	MSRP
count	2823.000000	2823.000000	2823.000000	2823.000000	2823.000000	2823.000000	2823.000000	2823.00000	2823.000000
mean	10258.725115	35.092809	83.658544	6.466171	3553.889072	2.717676	7.092455	2003.81509	100.715551
std	92.085478	9.741443	20.174277	4.225841	1841.865106	1.203878	3.656633	0.69967	40.187912
min	10100.000000	6.000000	26.880000	1.000000	482.130000	1.000000	1.000000	2003.00000	33.000000
25%	10180.000000	27.000000	68.860000	3.000000	2203.430000	2.000000	4.000000	2003.00000	68.000000
50%	10262.000000	35.000000	95.700000	6.000000	3184.800000	3.000000	8.000000	2004.00000	99.000000
75%	10333.500000	43.000000	100.000000	9.000000	4508.000000	4.000000	11.000000	2004.00000	124.000000
max	10425.000000	97.000000	100.000000	18.000000	14082.800000	4.000000	12.000000	2005.00000	214.000000

```
ed.head()
```

	Postcode	Sales_Rep_ID	Sales_Rep_Name	Year	Value
0	2121	456	Jane	2011	84219.497311
1	2092	789	Ashish	2012	28322.192268
2	2128	456	Jane	2013	81878.997241
3	2073	123	John	2011	44491.142121
4	2134	789	Ashish	2012	71837.720959

```
ed.tail()
```

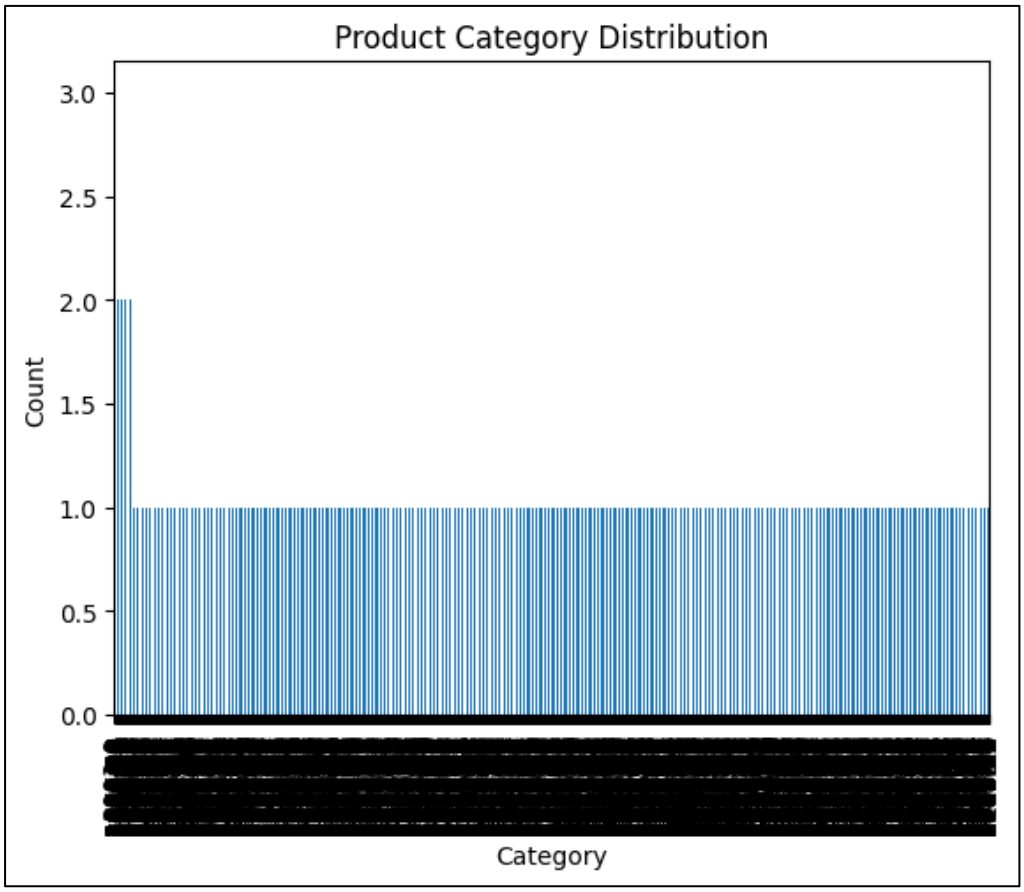
	Postcode	Sales_Rep_ID	Sales_Rep_Name	Year	Value
385	2164	123	John	2012	88884.535217
386	2193	456	Jane	2013	79440.290813
387	2031	123	John	2011	65643.689454
388	2130	456	Jane	2012	66247.874869
389	2116	456	Jane	2013	3195.699054

```
ed.info()
```

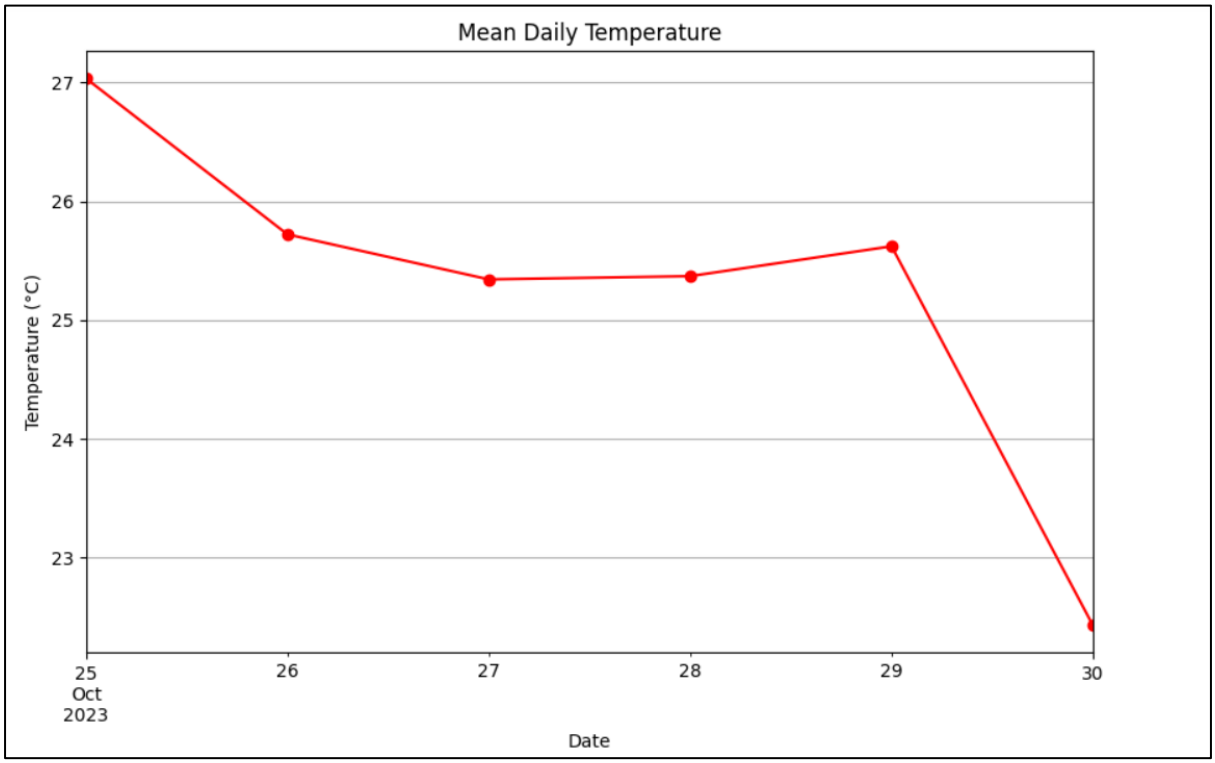
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 390 entries, 0 to 389
Data columns (total 5 columns):
 #   Column          Non-Null Count  Dtype
---  ------          --------------  -----
 0   Postcode        390 non-null    int64
 1   Sales_Rep_ID    390 non-null    int64
 2   Sales_Rep_Name  390 non-null    object
 3   Year            390 non-null    int64
 4   Value           390 non-null    float64
dtypes: float64(1), int64(3), object(1)
memory usage: 15.4+ KB
```

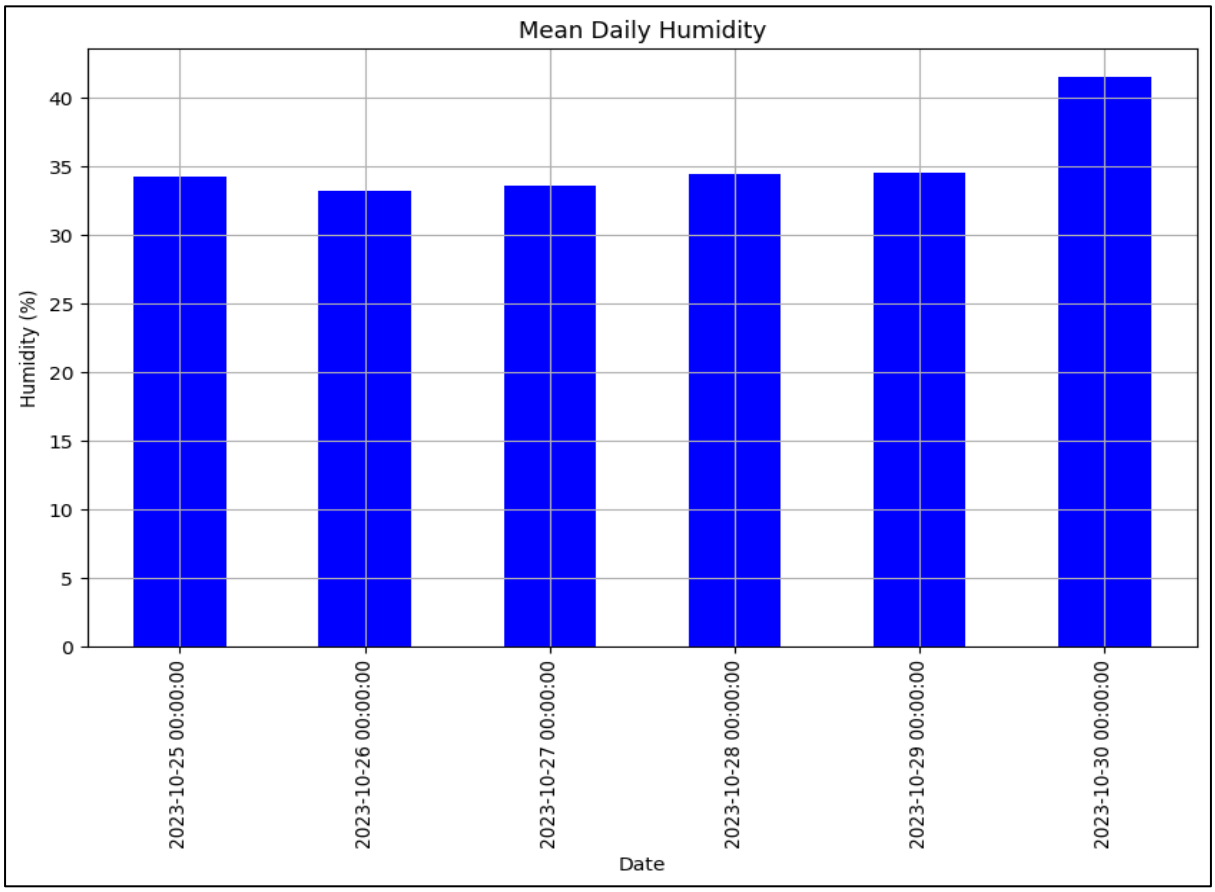
```
ed.describe()
```

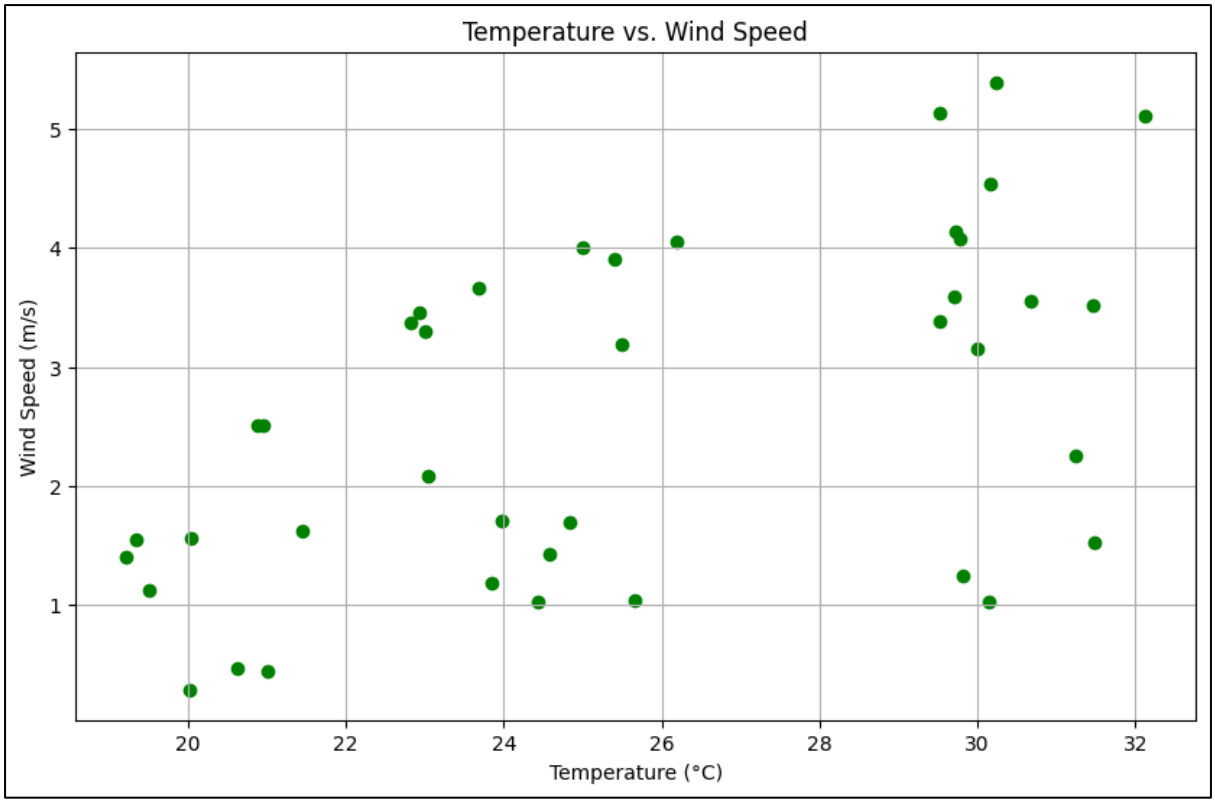
	Postcode	Sales_Rep_ID	Year	Value
count	390.000000	390.000000	390.000000	390.000000
mean	2098.430769	456.000000	2012.000000	49229.388305
std	58.652206	272.242614	0.817545	28251.271309
min	2000.000000	123.000000	2011.000000	106.360599
25%	2044.000000	123.000000	2011.000000	26101.507357
50%	2097.500000	456.000000	2012.000000	47447.363750
75%	2142.000000	789.000000	2013.000000	72277.800608
max	2206.000000	789.000000	2013.000000	99878.489209

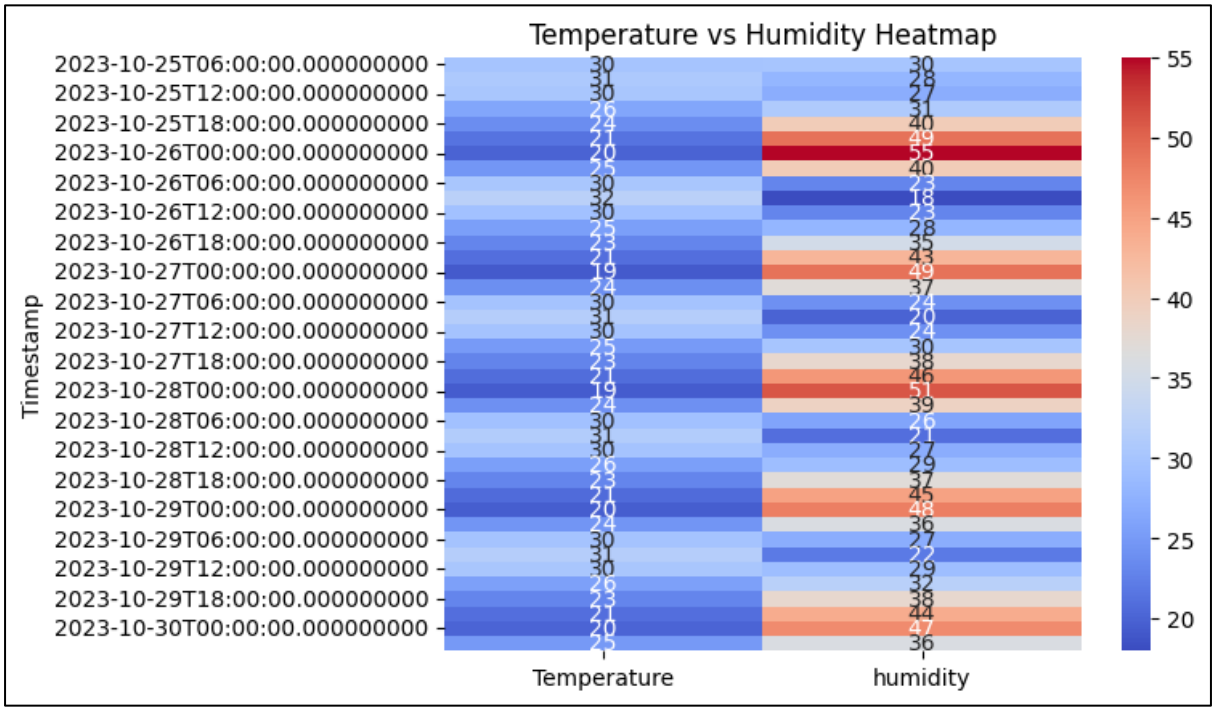


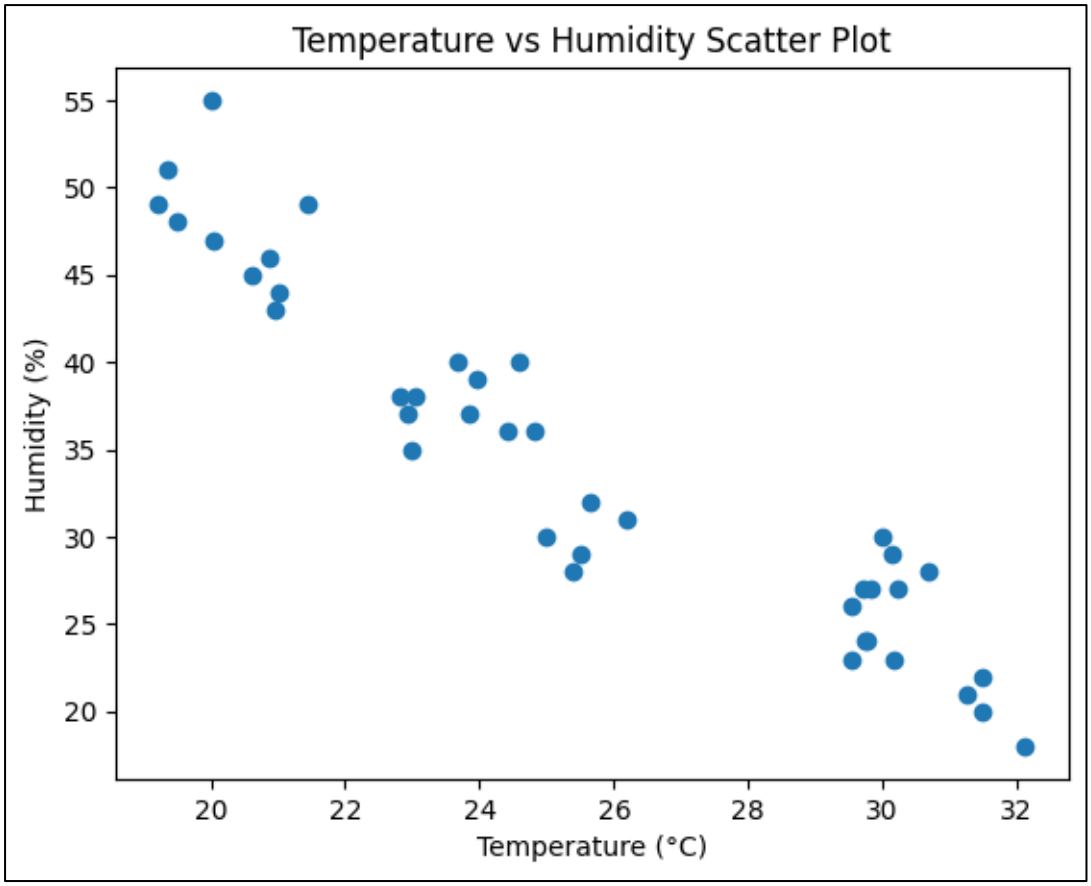
No.2


Mean Daily Temperature
Temperature (°C)
27
26
25
24
23
25
Oct
2023
26
27
28
29
30
Date

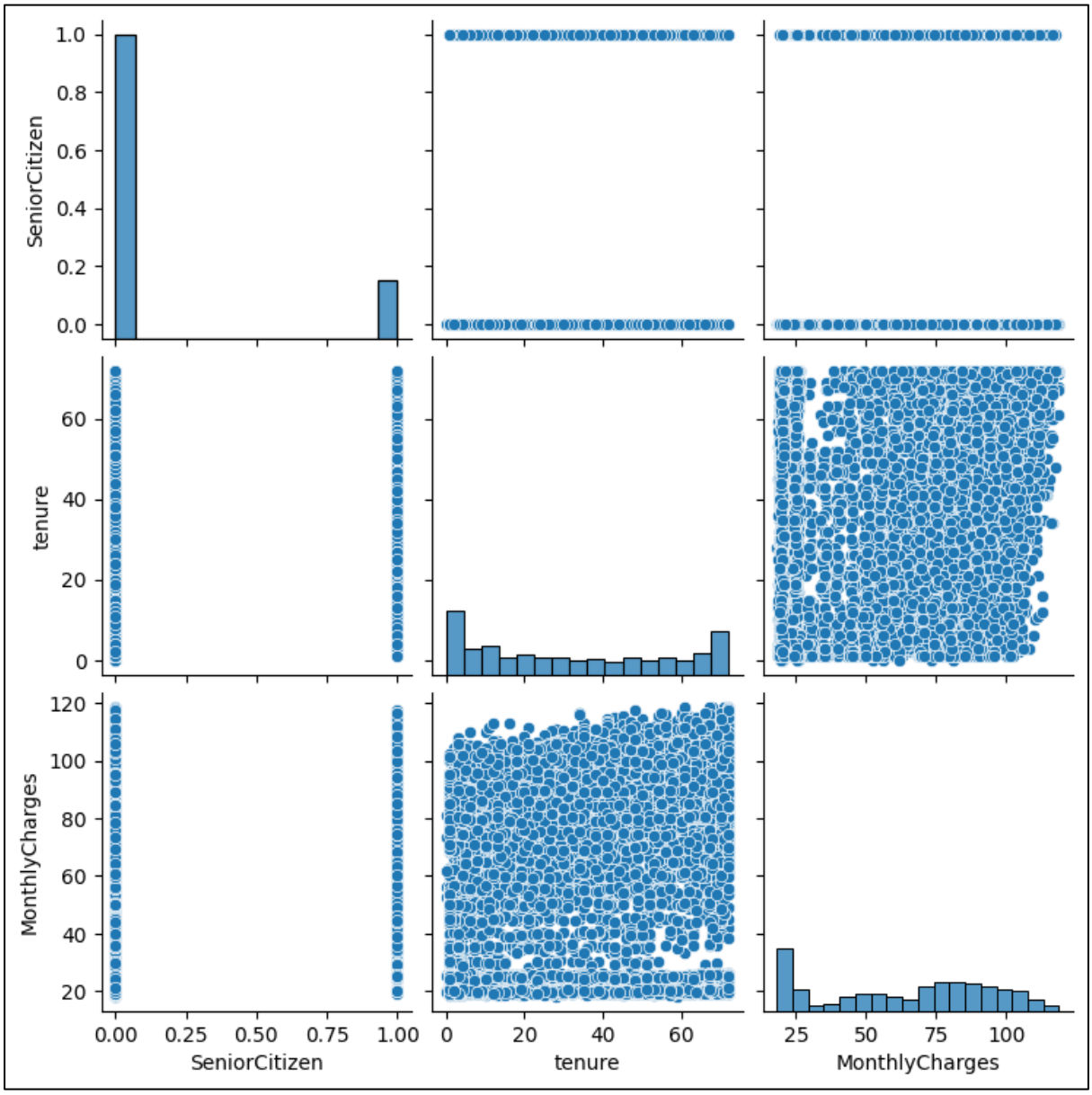

Mean Daily Humidity
Humidity (%)
40
35
30
25
20
15
10
5
0
2023-10-25 00:00:00
2023-10-26 00:00:00
2023-10-27 00:00:00
2023-10-28 00:00:00
2023-10-29 00:00:00
2023-10-30 00:00:00
Date

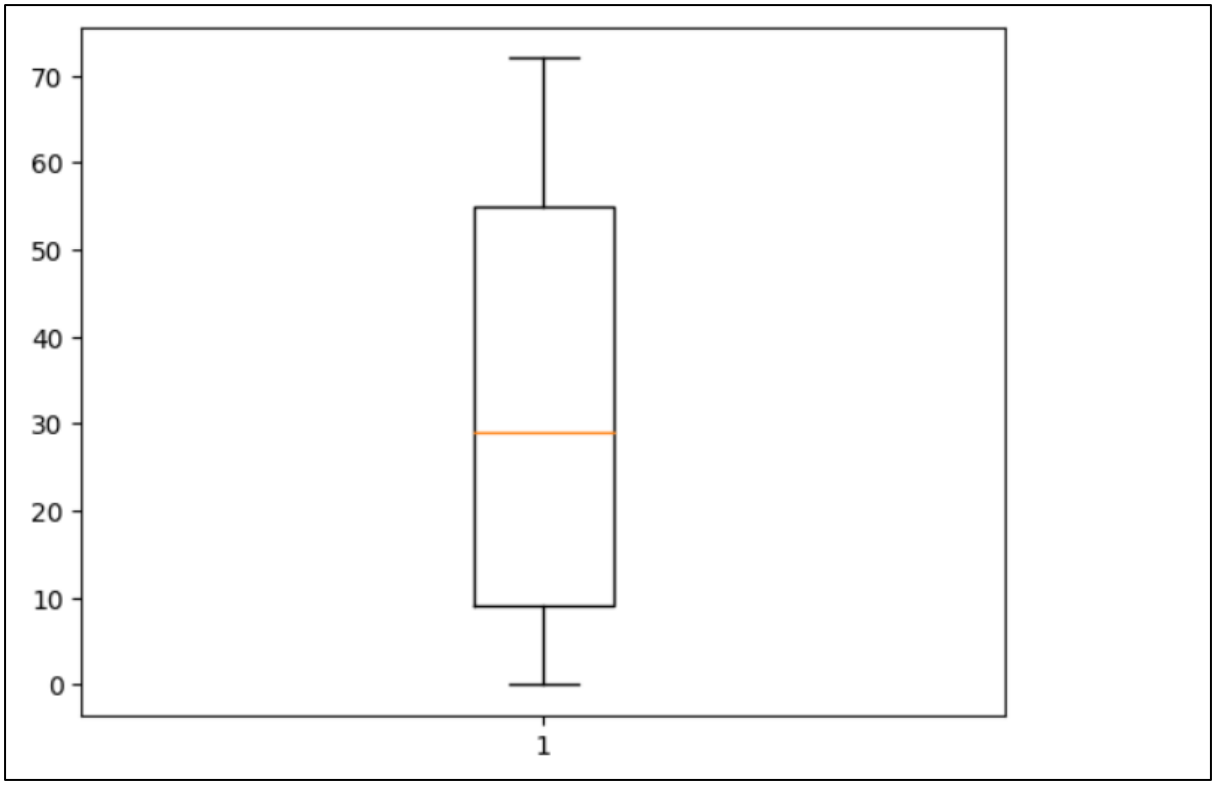

Temperature vs. Wind Speed
Wind Speed (m/s)
Temperature (°C)
1
2
3
4
5
20
22
24
26
28
30
32

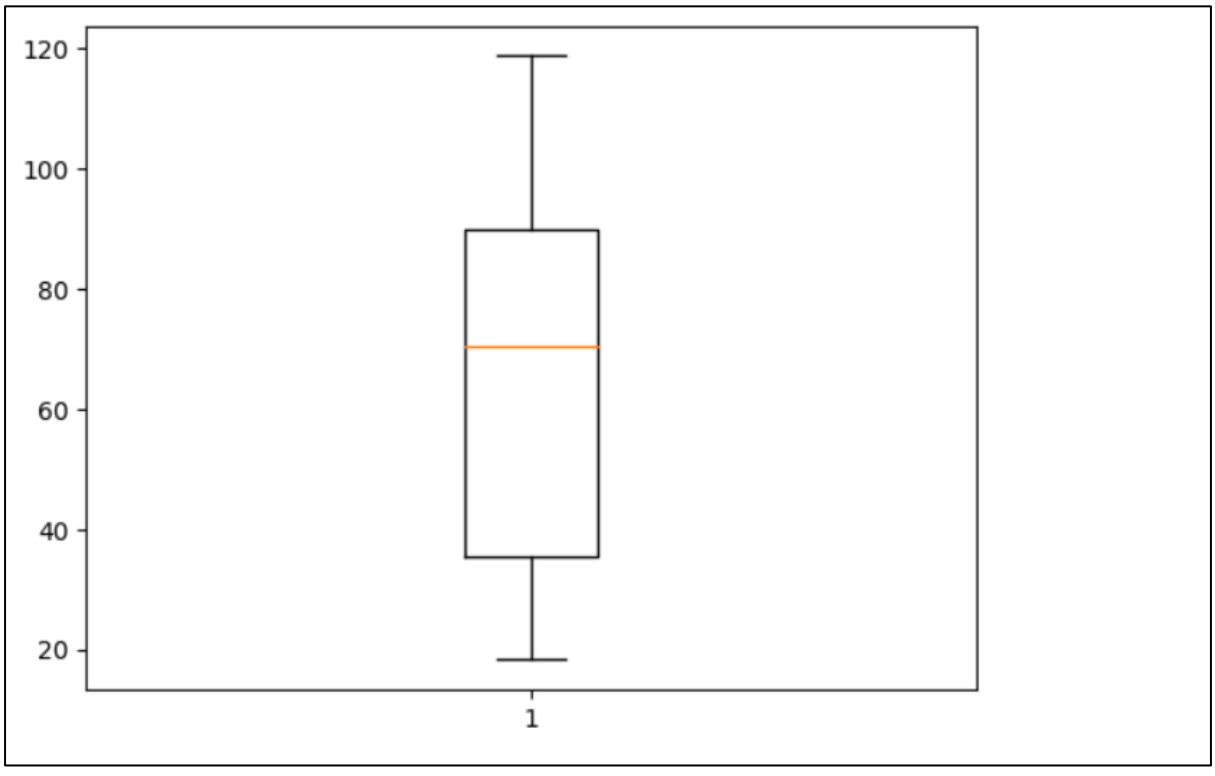

Temperature vs Humidity Heatmap
Timestamp
2023-10-25T06:00:00.000000000
2023-10-25T12:00:00.000000000
2023-10-25T18:00:00.000000000
2023-10-26T00:00:00.000000000
2023-10-26T06:00:00.000000000
2023-10-26T12:00:00.000000000
2023-10-26T18:00:00.000000000
2023-10-27T00:00:00.000000000
2023-10-27T06:00:00.000000000
2023-10-27T12:00:00.000000000
2023-10-27T18:00:00.000000000
2023-10-28T00:00:00.000000000
2023-10-28T06:00:00.000000000
2023-10-28T12:00:00.000000000
2023-10-28T18:00:00.000000000
2023-10-29T00:00:00.000000000
2023-10-29T06:00:00.000000000
2023-10-29T12:00:00.000000000
2023-10-29T18:00:00.000000000
2023-10-30T00:00:00.000000000
Temperature
humidity
55
50
45
40
35
30
25
20


Temperature vs Humidity Scatter Plot
Humidity (%)
Temperature (°C)
55
50
45
40
35
30
25
20
20
22
24
26
28
30
32

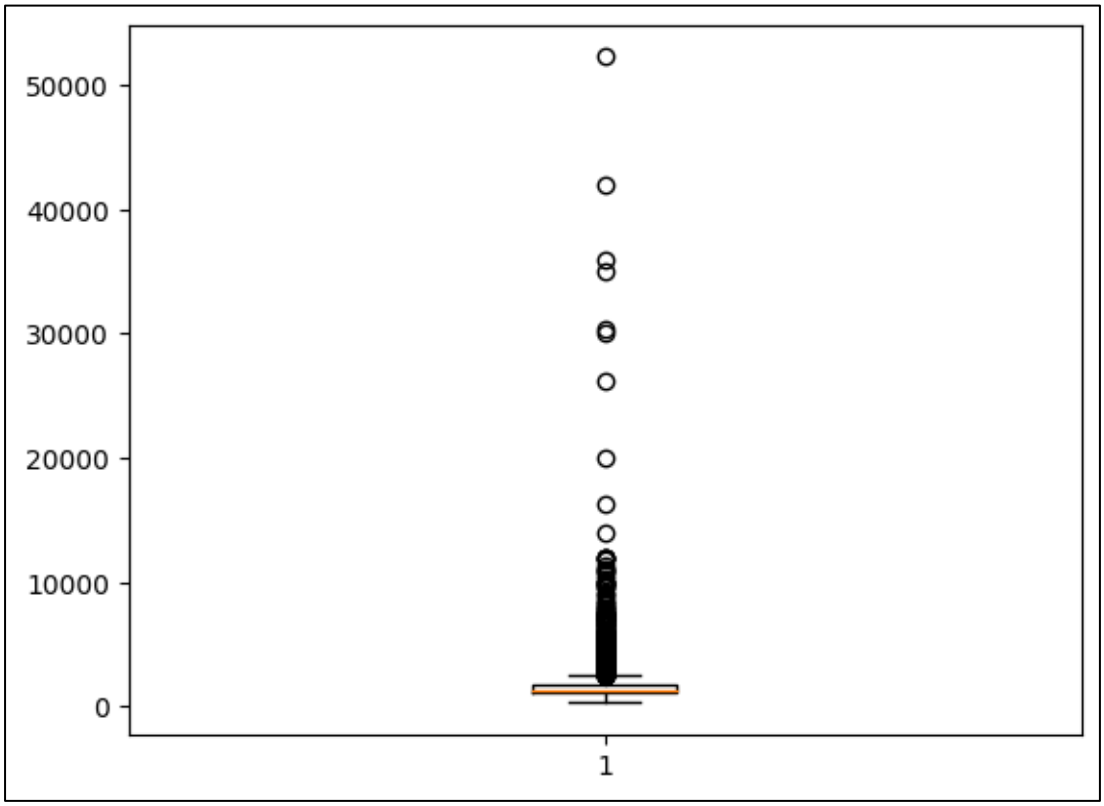
No.3

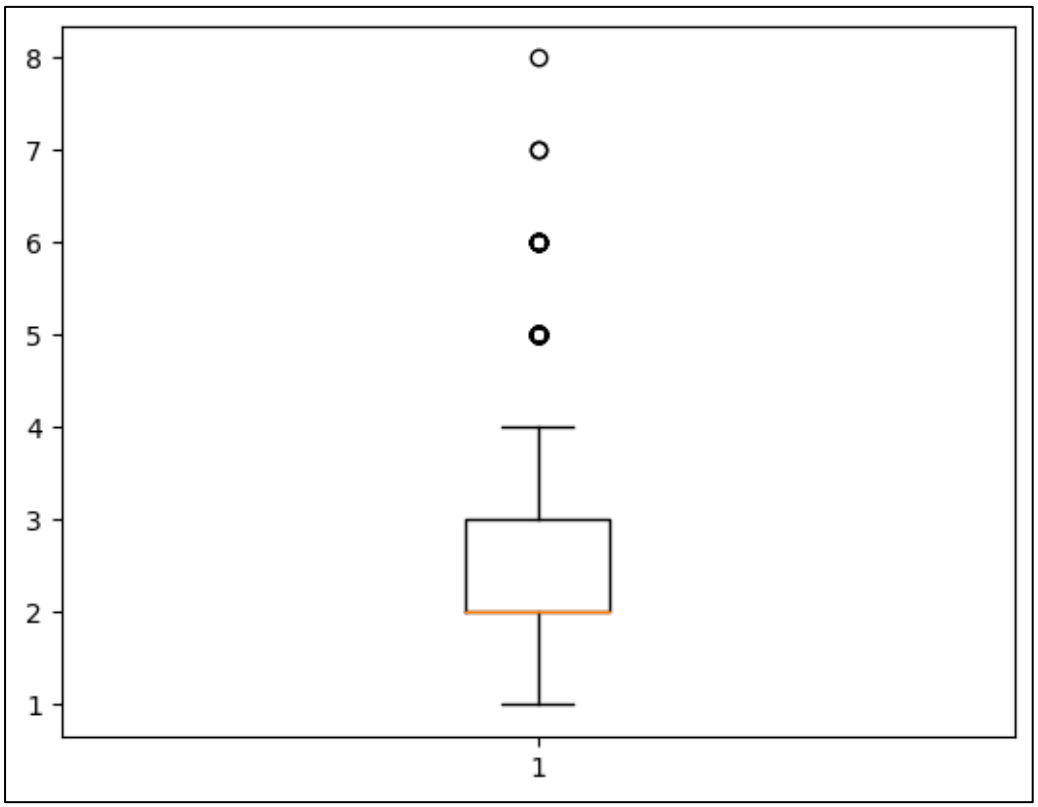

SeniorCitizen
tenure
MonthlyCharges
1.0
0.8
0.6
0.4
0.2
0.0
60
40
20
0
120
100
80
60
40
20
0.00
0.25
0.50
0.75
1.00
0
20
40
60
25
50
75
100

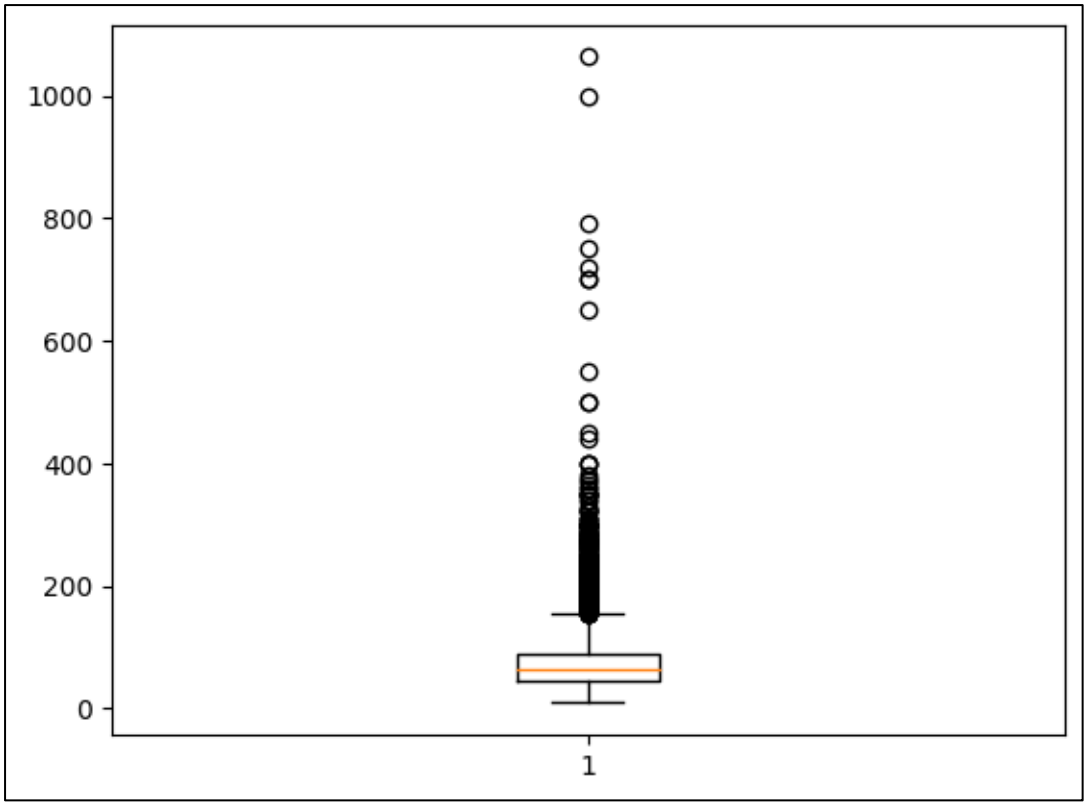

70
60
50
40
30
20
10
0
1

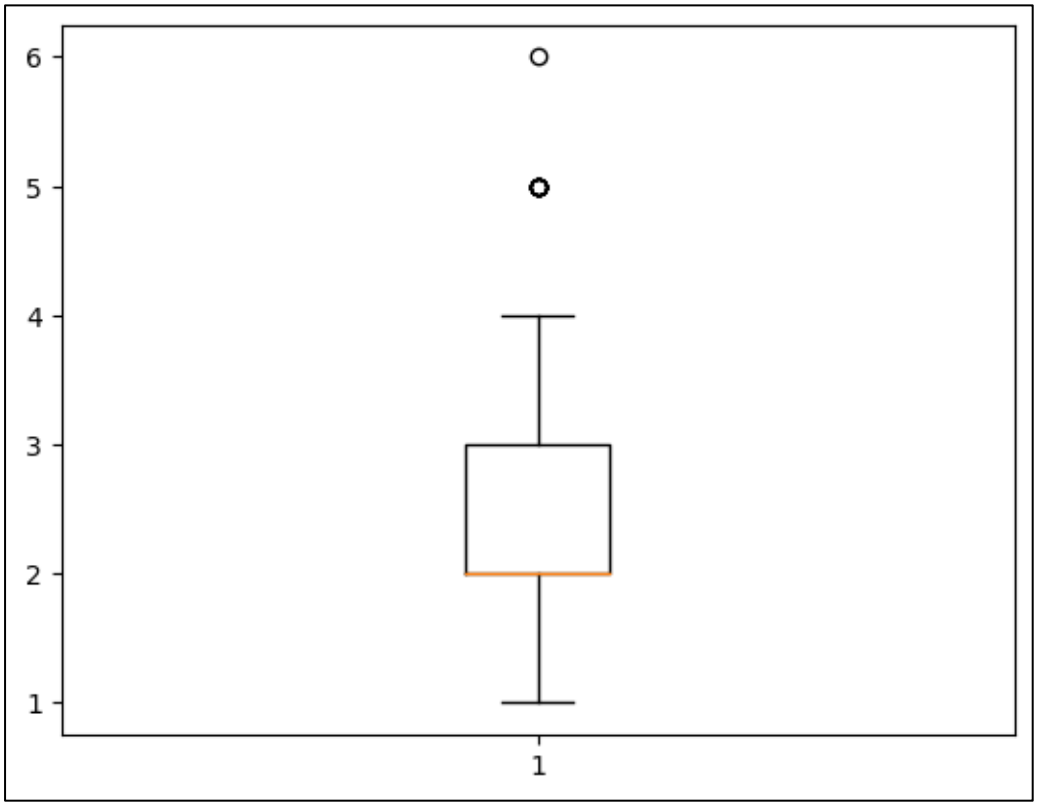

120
100
80
60
40
20
1

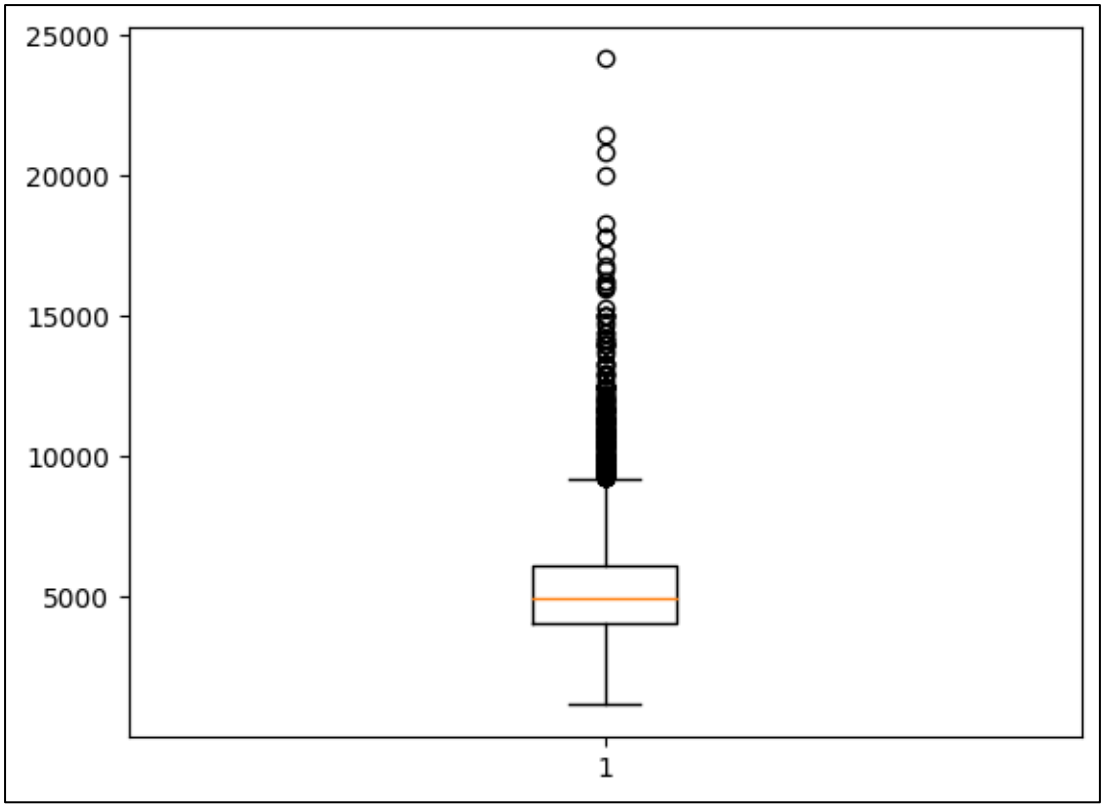
No.4

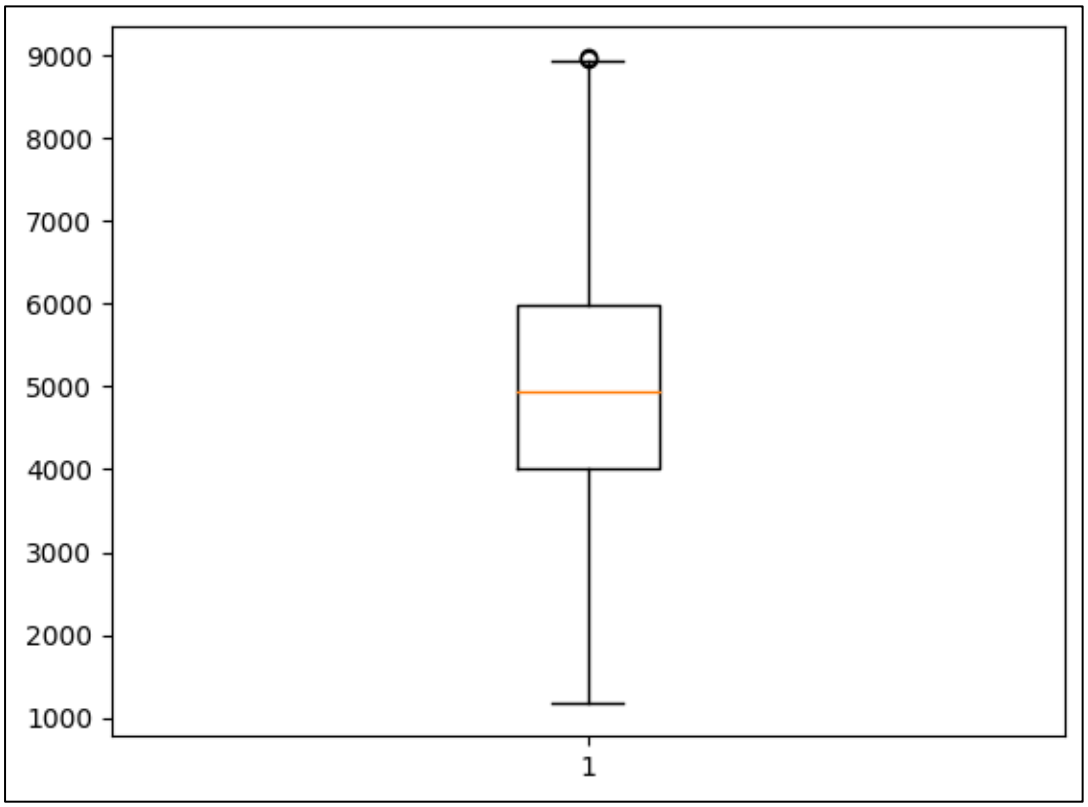




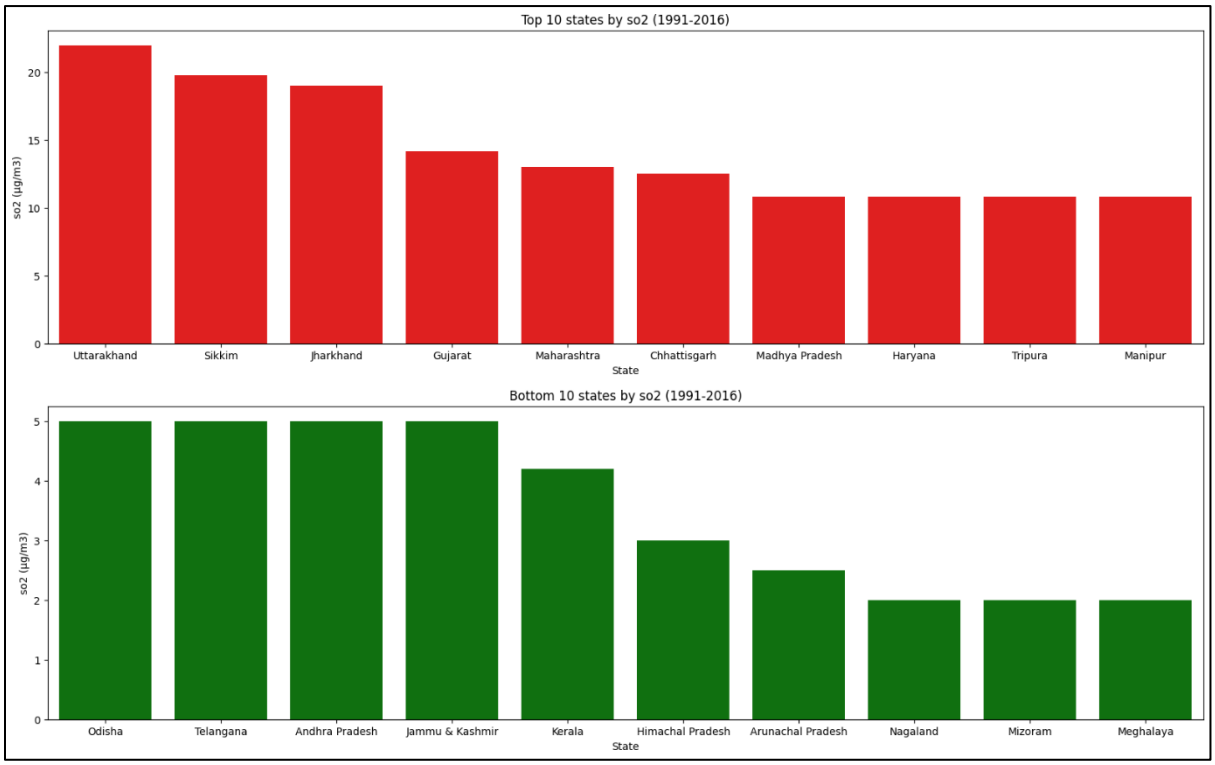

1000
800
600
400
200
0
1

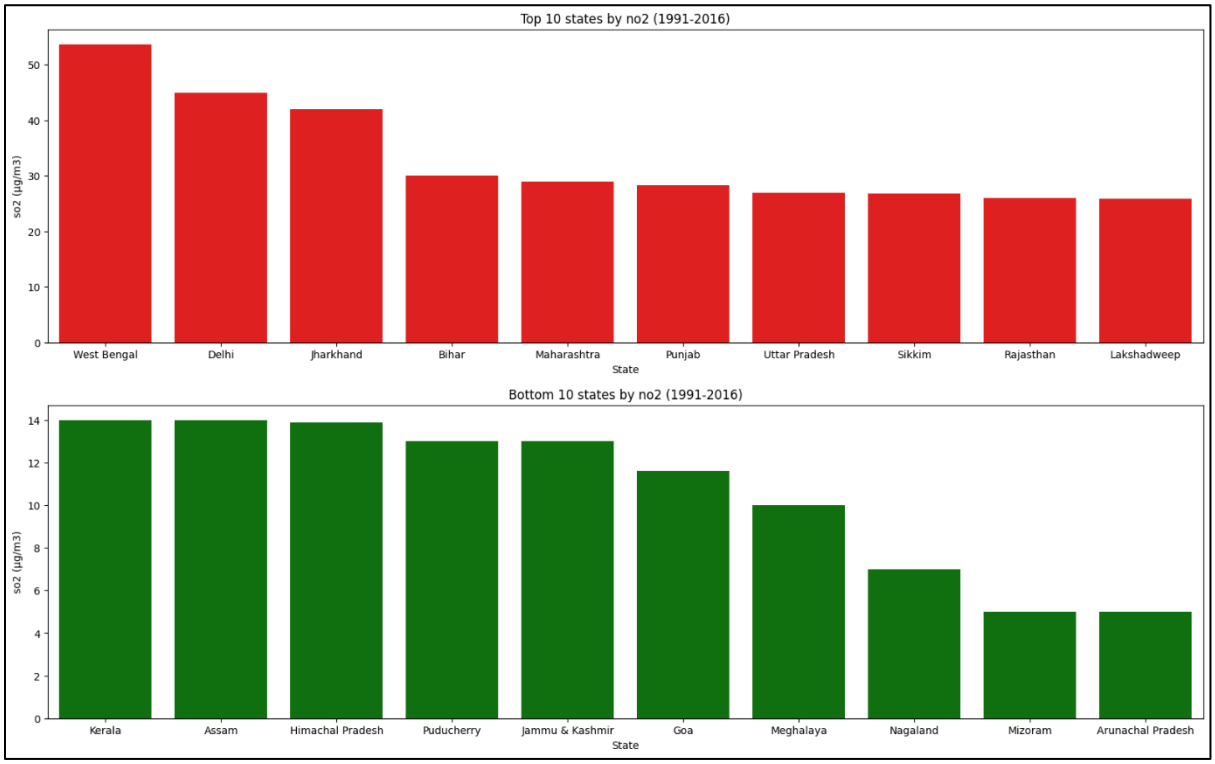

6
5
4
3
2
1
1

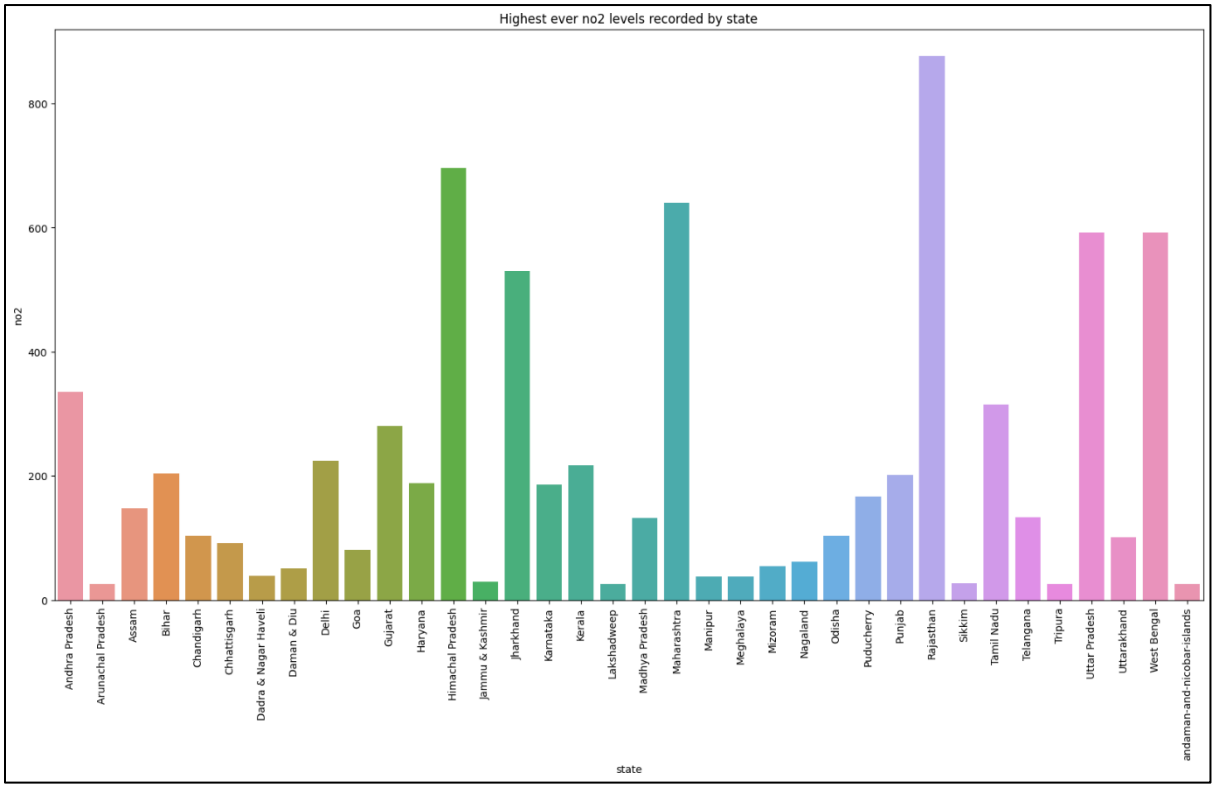

25000
20000
15000
10000
5000
1

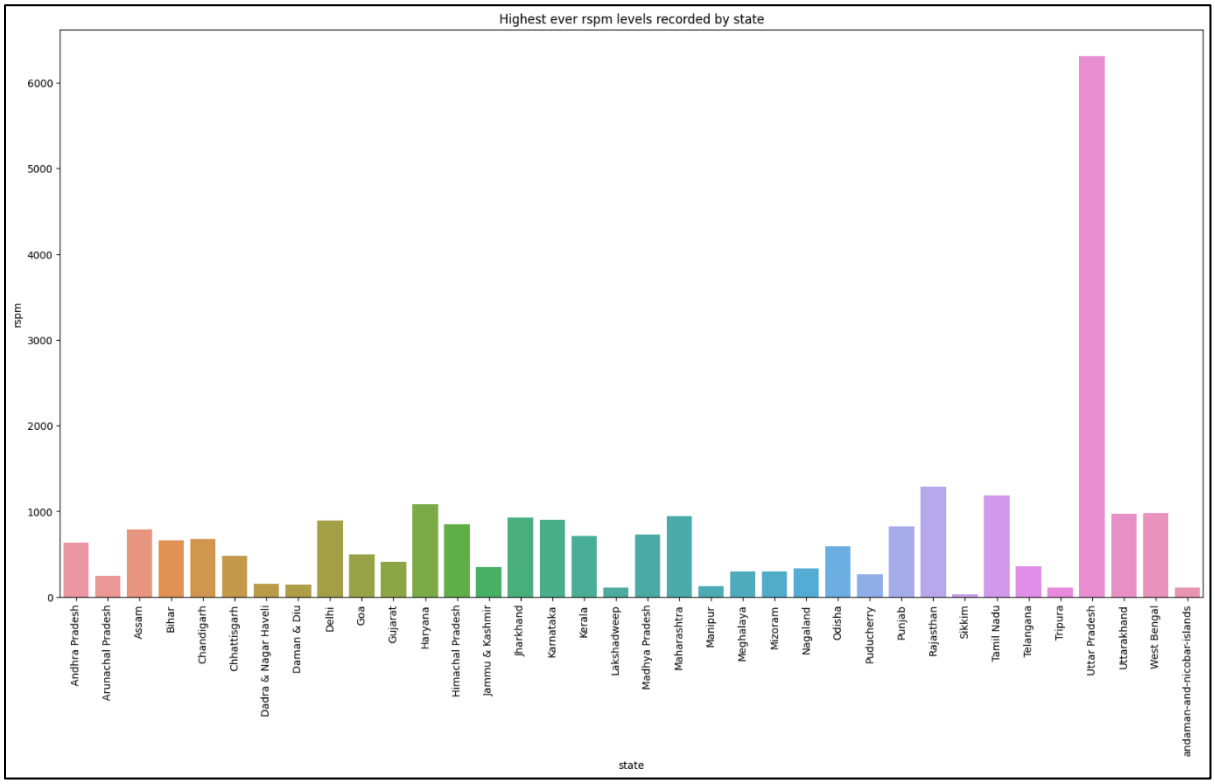

9000
8000
7000
6000
5000
4000
3000
2000
1000
1

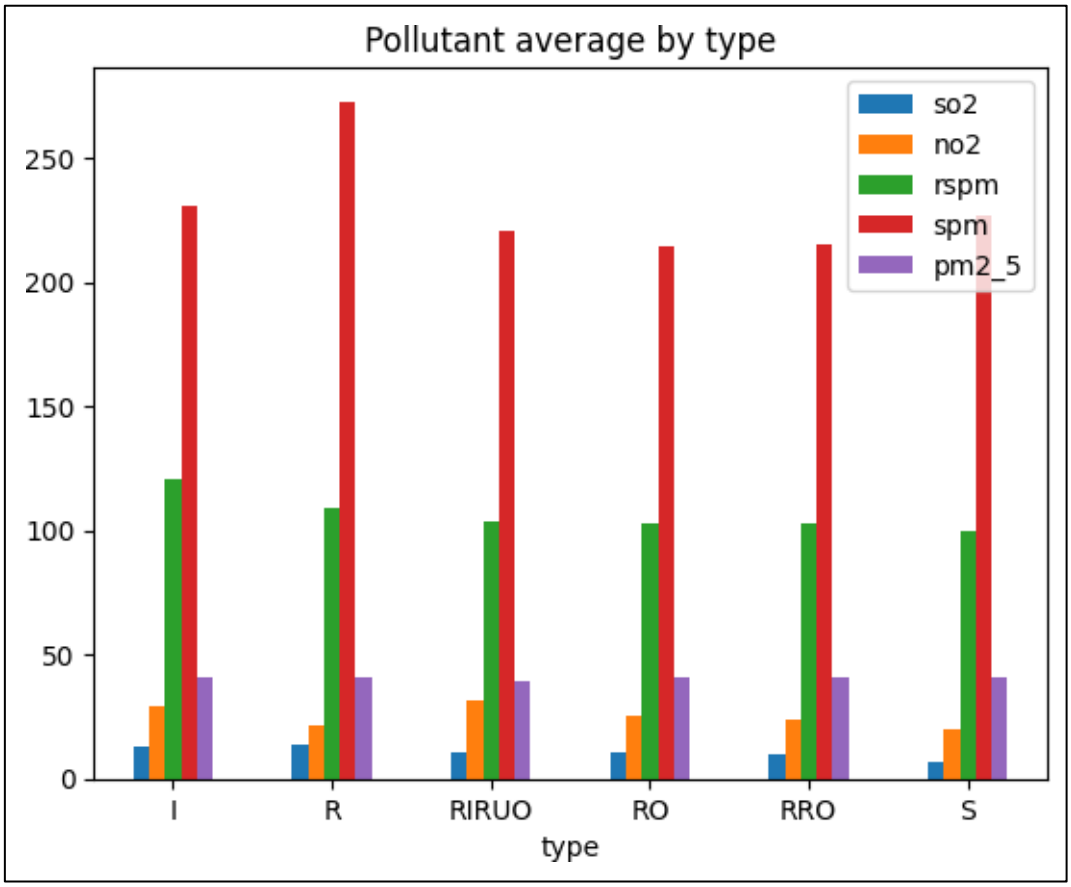
No.5

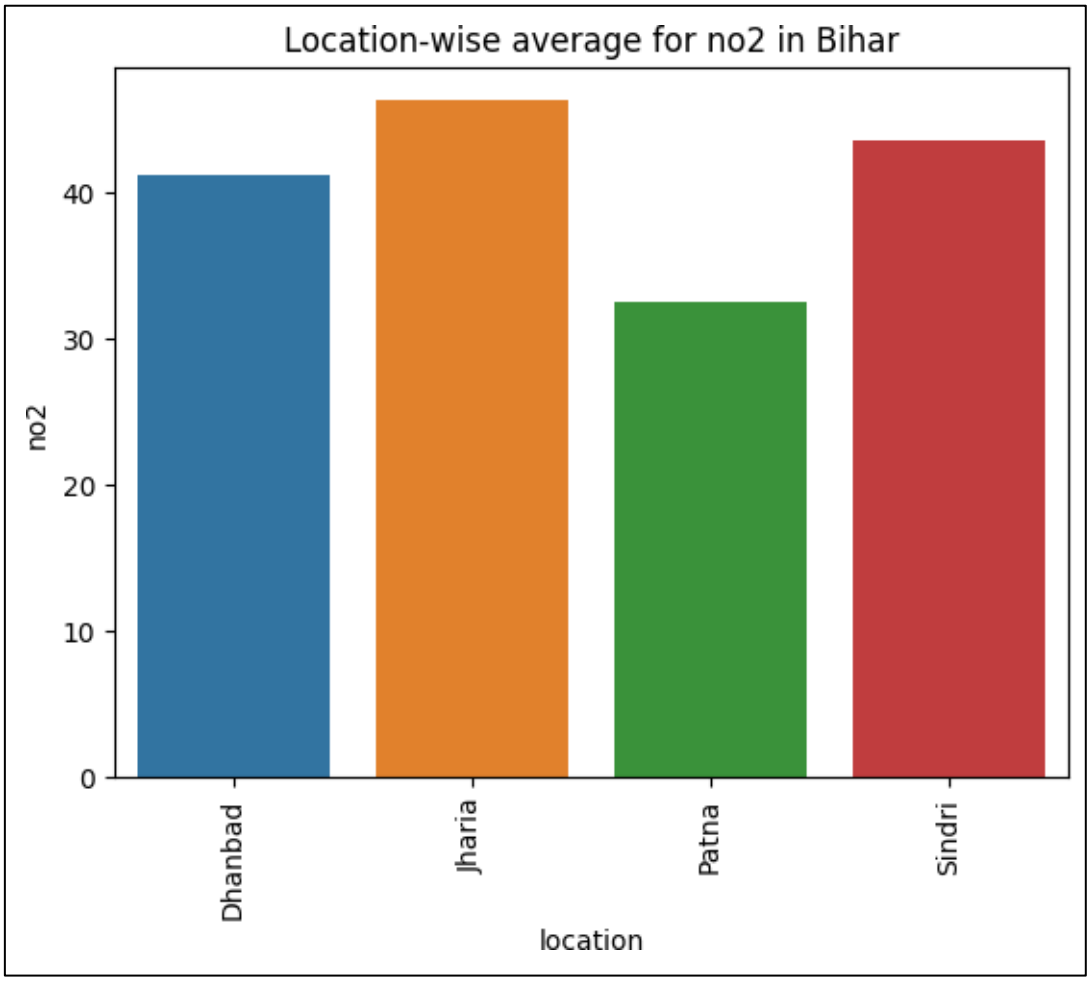

Top 10 states by so2 (1991-2016)
so2 (µg/m3)
0
5
10
15
20
Uttarakhand
Sikkim
Jharkhand
Gujarat
Maharashtra
Chhattisgarh
Madhya Pradesh
Haryana
Tripura
Manipur
State
Bottom 10 states by so2 (1991-2016)
so2 (µg/m3)
0
1
2
3
4
5
Odisha
Telangana
Andhra Pradesh
Jammu & Kashmir
Kerala
Himachal Pradesh
Arunachal Pradesh
Nagaland
Mizoram
Meghalaya
State


Top 10 states by no2 (1991-2016)
so2 (µg/m3)
0
10
20
30
40
50
West Bengal
Delhi
Jharkhand
Bihar
Maharashtra
Punjab
Uttar Pradesh
Sikkim
Rajasthan
Lakshadweep
State
Bottom 10 states by no2 (1991-2016)
so2 (µg/m3)
0
2
4
6
8
10
12
14
Kerala
Assam
Himachal Pradesh
Puducherry
Jammu & Kashmir
Goa
Meghalaya
Nagaland
Mizoram
Arunachal Pradesh
State

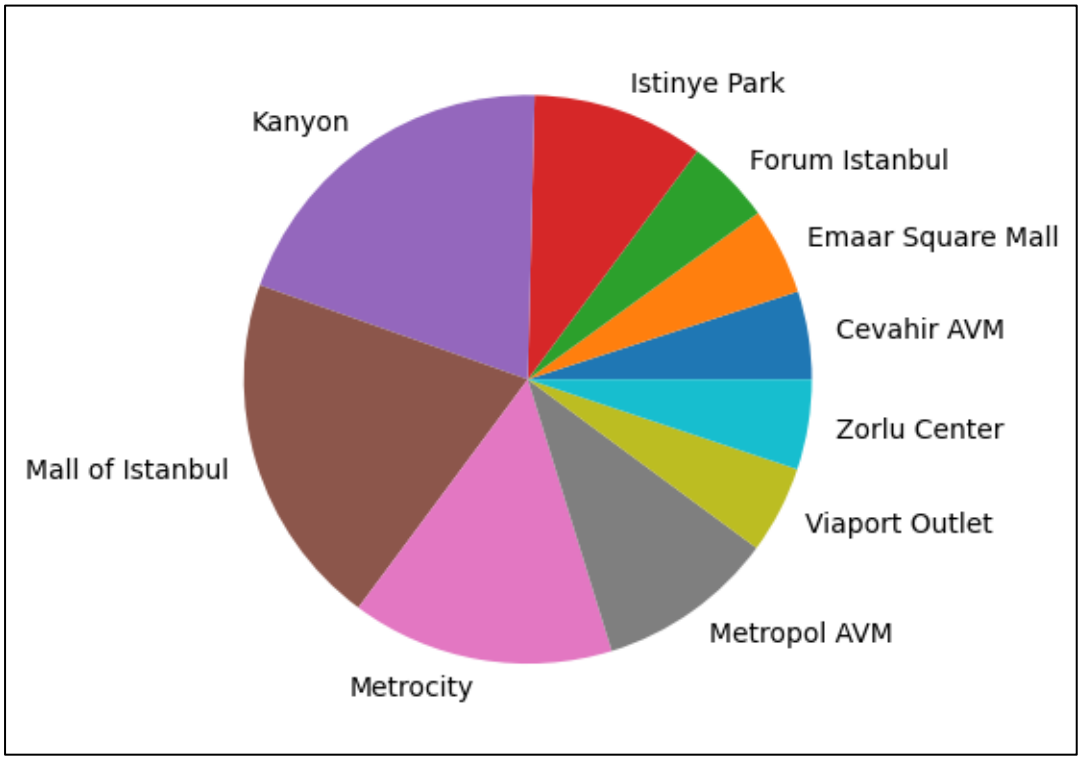

Highest ever no2 levels recorded by state
no2
0
200
400
600
800
Andhra Pradesh
Arunachal Pradesh
Assam
Bihar
Chandigarh
Chhattisgarh
Dadra & Nagar Haveli
Daman & Diu
Delhi
Goa
Gujarat
Haryana
Himachal Pradesh
Jammu & Kashmir
Jharkhand
Karnataka
Kerala
Lakshadweep
Madhya Pradesh
Maharashtra
Manipur
Meghalaya
Mizoram
Nagaland
Odisha
Puducherry
Punjab
Rajasthan
Sikkim
Tamil Nadu
Telangana
Tripura
Uttar Pradesh
Uttarakhand
West Bengal
andaman-and-nicobar-islands
state

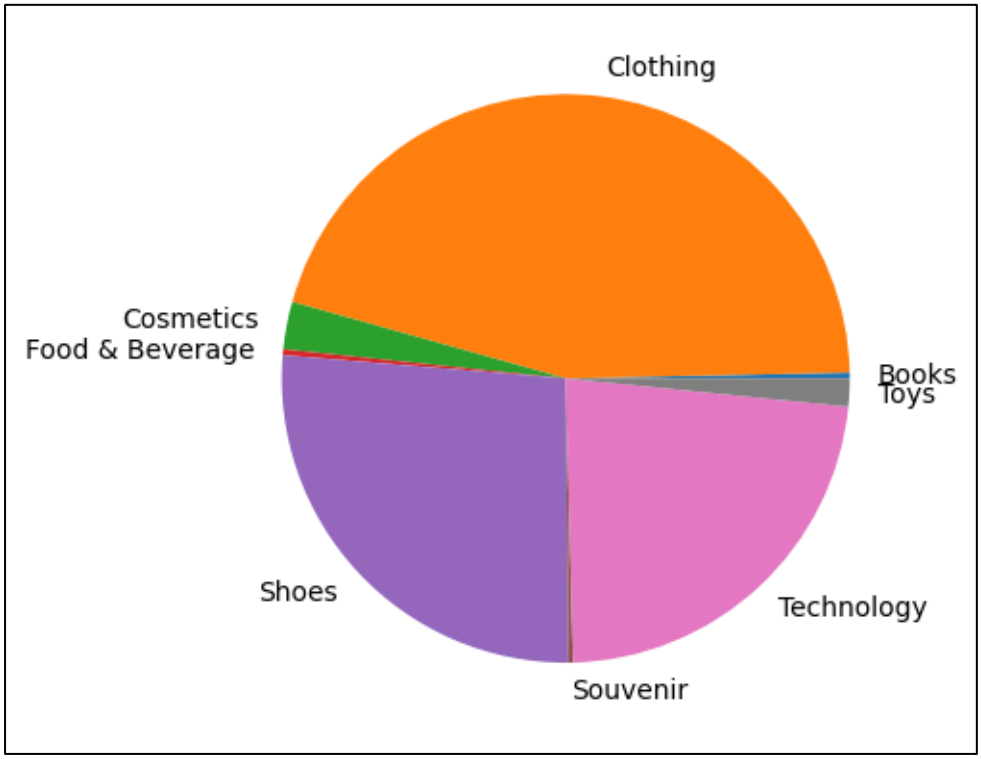

Highest ever rspm levels recorded by state
rspm
0
1000
2000
3000
4000
5000
6000
Andhra Pradesh
Arunachal Pradesh
Assam
Bihar
Chandigarh
Chhattisgarh
Dadra & Nagar Haveli
Daman & Diu
Delhi
Goa
Gujarat
Haryana
Himachal Pradesh
Jammu & Kashmir
Jharkhand
Karnataka
Kerala
Lakshadweep
Madhya Pradesh
Maharashtra
Manipur
Meghalaya
Mizoram
Nagaland
Odisha
Puducherry
Punjab
Rajasthan
Sikkim
Tamil Nadu
Telangana
Tripura
Uttar Pradesh
Uttarakhand
West Bengal
andaman-and-nicobar-islands
state

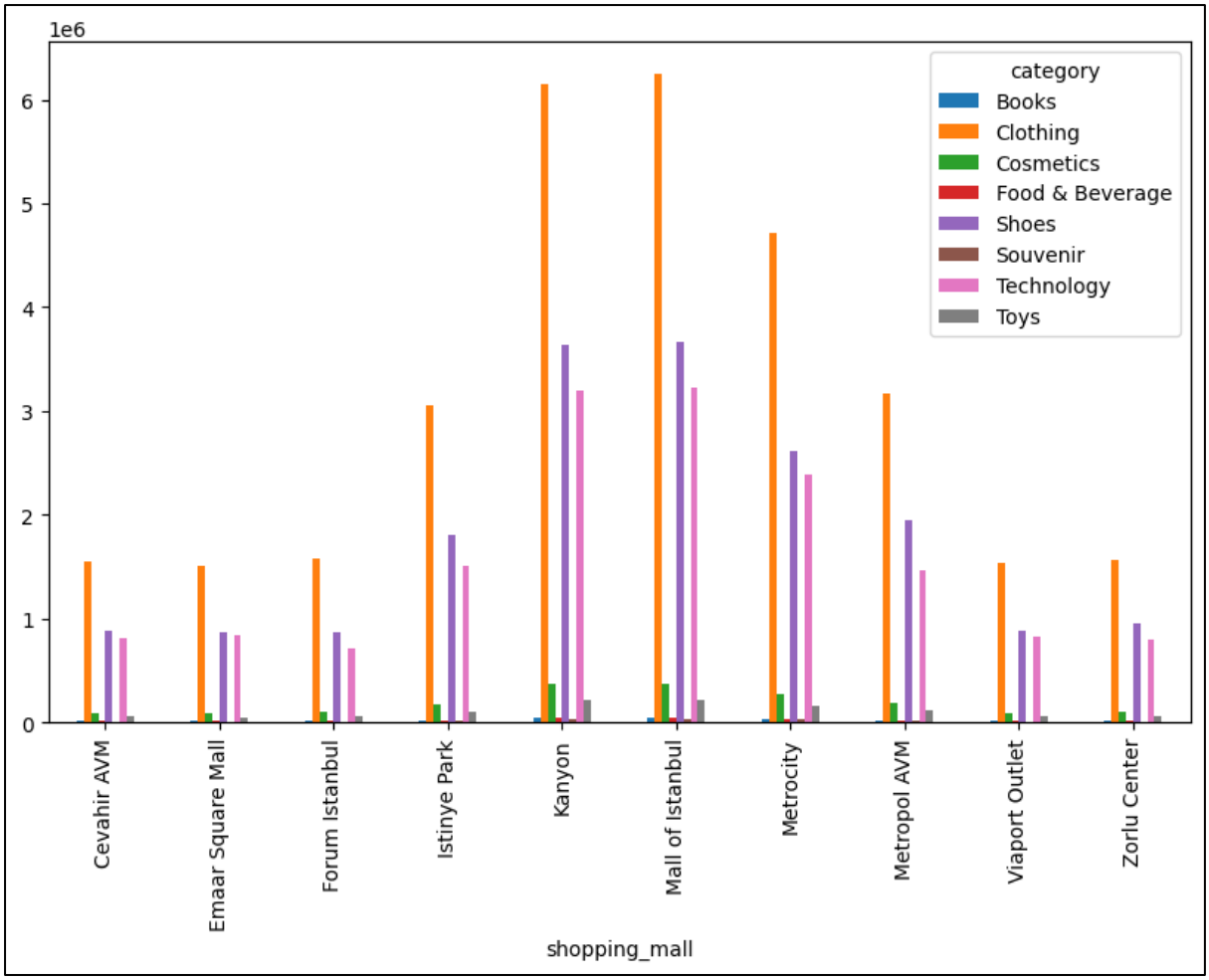

Pollutant average by type
so2
no2
rspm
spm
pm2_5
250
200
150
100
50
0
I
R
RIRUO
RO
RRO
S
type


Location-wise average for no2 in Bihar
no2
0
10
20
30
40
Dhanbad
Jharia
Patna
Sindri
location

No. 6


Istinye Park
Forum Istanbul
Emaar Square Mall
Cevahir AVM
Zorlu Center
Viaport Outlet
Metropol AVM
Metrocity
Mall of Istanbul
Kanyon


Clothing
Books
Toys
Technology
Souvenir
Shoes
Food & Beverage
Cosmetics


1e6
6
5
4
3
2
1
0
category
Books
Clothing
Cosmetics
Food & Beverage
Shoes
Souvenir
Technology
Toys
Cevahir AVM
Emaar Square Mall
Forum Istanbul
Istinye Park
Kanyon
Mall of Istanbul
Metrocity
Metropol AVM
Viaport Outlet
Zorlu Center
shopping_mall