A REPORT

ON

Analysing the Air Quality Index (AQI) data from different regions of India and extracting data of Delhi from Central Pollution Control Board website and to model and analyse, predict the data.

BY

Names of Students	I.D. No.(s)
1. YASH SANGHI	2019B4A30758P

BIRLA INSTITUTE OF TECHNOLOGY & SCIENCE, PILANI

A REPORT

ON

Analysing the Air Quality Index (AQI) data from different regions of India and extracting data of Delhi from Central Pollution Control Board website and to model and analyse, predict the data.

BY

Name(s) of Students	I.D. No.(s)	Disciplines
1. YASH SANGHI	2019B4A30758P	Mathematics(B4) & Electrical and Electronics

BIRLA INSTITUTE OF TECHNOLOGY & SCIENCE, PILANI

ACKNOWLEDGEMENTS

In the present world of competition, there is always a race between people to succeed, in which those who have the will to come forward and accept challenges undoubtedly move ahead. The projects are like a bridge between theoretical and practical learning. With this willing and high spirit, I joined this project and had a valuable learning experience. First of all, I would like to thank the Almighty God, who guided me to work on the right path of life. Next to him are my Parents, without whose constant motivation, this project would not be a reality. I am feeling obliged in taking the opportunity to sincerely thank my Industry mentor, **Dr Gaurav Purohit.** A special thanks to my esteemed teacher, **Prof Rakhee (Professor at Mathematics Department, Birla Institute of Technology and Science, Hyderabad Campus),** who is a source of constant guidance and paramount support throughout the project. At last, I am thankful to all my teachers and friends for always helping and believing in me. I have no valuable words to express my gratitude to all the favors received from every person in making this project a success.

BIRLA INSTITUTE OF TECHNOLOGY & SCIENCE PILANI
(RAJASTHAN)

Date of Start: 15th January, 2022

Date of Submission: 2nd May, 2022

Title of the Project: Analysing the Air Quality Index (AQI) data from different regions of India and extracting data of Delhi from Central Pollution Control Board website and to model and analyse, predict the data.

Name and ID. :

Yash Sanghi - 2019B4A30758P

Name(s) and designation(s) of the expert(s): Dr Gaurav Purohit (Supervisor)

Co-Supervisor : Prof Rakhee

Project Areas : Machine Learning, Data analysis

Signature of Students
Date: 02/05/2022

Signature of Faculty
Date:

TABLE OF CONTENTS

INTRODUCTION

Think of the AQI as a yardstick that runs from 0 to 500. The higher the AQI value, the greater the level of air pollution and the greater the health concern. For example, an AQI value of 50 or below represents good air quality, while an AQI value over 300 represents hazardous air quality.

For each pollutant an AQI value of 100 generally corresponds to an ambient air concentration that equals the level of the short-term national ambient air quality standard for protection of public health. AQI values at or below 100 are generally thought of as satisfactory. When AQI values are above 100, air quality is unhealthy: at first for certain sensitive groups of people, then for everyone as AQI values get higher.

The AQI is divided into six categories. Each category corresponds to a different level of health concern. Each category also has a specific color. The color makes it easy for people to quickly determine whether air quality is reaching unhealthy levels in their communities.

Extracting the Air Quality Index (AQI) data from different regions of Delhi from Central Pollution Control Board website and to model and predict the air quality in regions where the data is not available.

Real-time air quality information, such as the concentration of NO2 , PM2.5, and PM10, is of great importance to support air pollution control and protect humans from damage by air pollution. We are mainly focused on PM2.5 as it is the major contributor to air pollution

A Aqi dataset of more than 80000 dataponts was provided by my supervisor of different cities of india which is analysed and modelled to predict the AQI level for places where data is not available.

Selenium library was used to extract data set for 6 months of 7 regions of Delhi (namely Anand vihar, Dilshad garden, Mandir marg, Nsit Dwarka, RK Puram, Shadipur) at a 15 minute frequency

with more than 80,000 data points.

Extracted 3-year data to have a large enough sample to analyse.

Delhi was chosen as main city to further analyse as it has most number of towers to analyse air quality as compared to other citie so that better quality of data was extracted.

SOFTWARE AND CONCEPTS

i. Exploratory Data Analysis (EDA):

Exploratory Data Analysis is the starting point for building any machine learning model. Right after we gather the data our primary job is to analyse and draw conclusions from it. EDA is a complement to the inferential statistics which tends to be rigid with rules and formulas. EDA involves looking at and describing the data set from different angles and then summarizing it.
The main parts of EDA that involve are handling missing values, removing duplicates, outlier treatment, normalizing and scaling (numerical variables), encoding categorical variables (dummy variables), bivariate analysis. Some of these we have picked and put in the EDA, and few have been put in the Feature Engineering section that comes later in the report.

ii. Feature Engineering

It's used to prepare the proper input dataset, that is compatible with our machine learning algorithms. Feature engineering also helps in improving the performance of our Machine learning models.

Box-Cox transformations is used to stabilize variance in heavily skewed data which we often get in real world data analysis. We make the data more like normal distribution.

Principal Component Analysis (PCA) is a method we used to reduce the dimensionality of our large dataset, by transforming different variables with large dataset into smaller one that still has plenty information from the large dataset.

Onehot Encoding :

Machine learning models require all input and output variables to be numeric. This means that if your data contains categorical data, you must encode it to numbers before you can fit andevaluate a model. For categorical variables where no ordinalrelationship exists, the integer encoding may not be enough, at best, ormisleading to the model at worst. Forcing an ordinal relationship via an ordinal encoding and allowing the model to assume a natural orderingbetween categories may result in poor performance or unexpected results (predictions halfway between categories). In this case, a one-hotencoding can be applied to the ordinal representation. This is where theinteger encoded variable is removed and one new binary variable is added for each unique integer value in the variable.

SMOTE:

One approach to addressing imbalanced datasets is to oversample the minority class. The simplest approach involves duplicating examples in the minority class, although these examples don't add any new information to the model. Instead, new examples can be synthesized from the existing examples. This is a type of data augmentation for the minority class and is referred to as the **Synthetic Minority Oversampling Technique**, or **SMOTE** for short. SMOTE first selects a minority class instance a at random and finds its k nearestminority class neighbors. The synthetic instance is then created by choosing one of the k nearest neighbors b at random and connecting a and b to form a line segment in the feature space. The synthetic instances are generated as a convex combination of the two chosen instances a and b.

My Implementation:

Then Outlier Detection was done, this is done to trim away the most extreme observations. A box plot was drawn to visualize the distribution of different columns. \. First, we have written two utility functions to identify the outlierssemi automatically through univariate perspective. They were filtered using standard deviation from mean and interquartile range. Box plotswere plotted to visualize the outliers that were detected by semi automatically detectable functions. Look figure5 in appendix for the box plot. Actual Outlier detection using multivariate perspective was done after the splitting the dataset.

The process of outlier detection was resumed after the split dataset. This time we have done a more accurate process using multivariate outlier detection with the help of Mahalanobis Distance. The reason for choosing Mahalanobis distance over Euclidean distance was its failure in detecting the outliers with multidimensional data. Since Mahalanobis distance calculation uses the Eigenvalues of the variables instead of the original axis, it can make something like a feature scaling. With these functions of Mahalanobis distance and function to detect multiple outliers we are able to find outliers and remove the outliers. After calculation of Mahalanobis distance we have put up a threshold as 2*standard deviation from the mean. All the points which are above mean+2*std and which are lower than mean-2*std are considered as outliers and eventually removed from the dataset.

I splitted data into a test data set(X-test) and a train data set(X-train), trained label dataset(Y-train) and test label data set (Y-test) with a ratio of 7:3 in favour of the train data set. This was done directly using the "trains_test_split" function from scikit-learn model selection.

Feature Scaling:

After the addition of business-oriented features now we are left with ample number of features but before proceeding to model selection we have to do the feature selection. Features Scaling is really important process to do or else the results are going to distort and we would be getting unexpected and uneven results. The reason for this is that different features are in different units so in order to bring all the features to same ground and compare we do the scaling. Here we have applied the **max-min scaler** from sklearn library to all the **numerical features.** The features with applied scaling with max-min scaling is not done in place rather the max-min scaled features are added to the data frame where each new feature looks where the '_t' is appended to each numerical feature and added to the data frame.

Feature Transformation:

Now after scaling we have to ensure that all the features are distributed normally. For transforming each distribution of features into nearly gaussian distribution and finding which distribution is best related to target variable ,we have created a function power_transf. Power_transf function checks each feature with different transformation such as logp transformation, exponential transformation, square roottransformation, square transformation , power 0.25 ,power 4 transformation and it applies the transformation which results in the best correlation with the target variable and returns the besttransformation for each feature. For checking the correlation we have used the chi-square test and had run it on the contingency table or crosstable of each feature and the target variable for determining the relationbetween that feature and the target variable. The transformation with best chi2_contingency value is applied and returned with this function.This transformation is applied on all the previously scaled features. This transformation is done inplace.

One Hot Encoding:

All the categorical variables city names were converted encoded using One Hot Encoding. The reason for opting the One Hot Encoding was that all the features were nominal variables rather than ordinal variables where the instances of features have certain weight and priority over other instances. This one hot encoding was done using get_dummies from pandas.

Balancing Data Set:

I chose the option of over sampling. For that I tried to use two methods, one is doing over sampling with the help of random over sampler from sk-learn where it randomly picks the rows and duplicates them and appends it to the dataset. The second one was using the technique called SMOTE which is "Synthetic Minority Oversampling Technique" where the SMOTE algorithm synthetically produces new data points with the help of nearest neighbors to a point. It works little similar to KNN in selecting the target value and dependent variable values. We have tried both the procedures and ended up taking the best one that is the procedure of SMOTE to oversample the dataset and make it a balanced one.

Model Selection:

Finally, we were ready for the model selection where we have to select a machine learning algorithm and train the model and test the results. This is the heart of any machine learning problem. The problem statement given is so obvious that it is a classification problem. Moreover, we were having a data set which contains the information of enrolled customers and their previous purchase data, so it belongs to supervised classification. Therefore, we tried to use all the best and known supervised classification models. These models range from simple KNN, Naïve bayes, logistic regression to boosting algorithms like XGBoost, AdaBoost along with the decision tree algorithms as Decision tree classifier , Random forest classifier and SVC. With the help of metrics and hyper parameter tuning we have tried with different combinations for training the model. We have selected the best model with good parameters atlast.

Metrics:

Metrics does the quantitative assessment of performance any model. This is the final and crucial step in determining and building the best model. As discussed above we have tried different classification algorithms, it is an impossible task to pick one model randomly among all the models because this model is going to have a very important job to do in the future that is going to decide the fate of businesses. If some one uses this model for their targeted marketing and if it is not to best extent, then that can lead to loss. The art of choosing the right metrics is also really important, let me explain this with an example . For example we were given data set of cancer patients and we were given a task of predicting if some one has cancer or not with the help of few lab reports. Generally the chance of having cancer is not so common therefore if we were given a set such that is having more records of people who doesn't have cancer. Let us assume that we have trained our model with an algorithm that it predicts that no one has cancer and if we go with accuracy for checking the performance of the model then we would be ended up with a dreadful prediction machine. This model is so dreadful because if some person was really having cancer but that

machine diagnosed and predicted that he is not having any cancer then he is going to not take any medication and eventually he would be suffering a lot. Therefore, we have given a good priority to the metrics. We have primarily considered the accuracy, ROC score and looked at confusion matrices for deciding which one is better. Confusion matrix consists of True positive values, False positive value, True negative value and false negative value. Our task was to minimize the values in false positives and false negatives. From our results of experimentation we were not getting good values of accuracy with the help of basic models like logistic regression and Naïve bayes. With those models we were getting an accuracy less than 70 percent and moreover there we were getting many false positive values and false negative values. All the remaining models such Random forest, Support vectorclassifier and boosting algorithms such as XGBoost , By considering even the ROC scores XGBoost and SVC these three algorithms topped the race. For the final selection we have looked up to the matrix table even though all these algorithms had good confusionmatrices by having less number of false positives and false negatives there were minute differences. The final battle was won by XGBoost which had an accuracy around 87 to 88 percent, ROC score as 0.78. This was the algorithm which we finalized and trained the model to predict AQI of a particular location based on the given data.

Conclusion:

1. Vehicular pollution contents are more related to air quality index.
2. Delhi is the most polluted city in terms of vehicular pollution contents.
3. Ahmadabad is the most polluted city in terms of industrial pollution content.
4. After COVID19 pandemic there is gradual dicrease in vehicular pollution

contents, industrial pollution content.

Extra Gradient Boost classifier 100% accurately classify the target variable. Our model could be an evidence that how machine learningcould help us to do targeted marketing and earn larger profits.
We got highest accuracy of 89.6% by using random forest algorithm.

Following are the accuracies from other models:

```
Following are the accuracies from other models:
Accuracy score of randomforest: 0.896084337349 3
Accuracy score SVC: 0.876506024096385 6 Accuracy
score of XGB:0.878012048192771 1
```

This accuracy can be increased for larger dataset and by applying neural networks. Similar models can also be implemented for different industries to increase their customer footprint. We can also modify this for implementing in recommendation systems.

Appendices:

Table 1 AQI values, descriptors, and color codes

AQI	Values Levels of Health Concern	Colors
0-50	Good (G)	Green
51-100	Moderate (M)	Yellow
101-150	Unhealthy for sensitive groups (U-S)	Orange
151-200	Unhealthy (U)	Red
201-300	Very unhealthy (VU)	Purple
301-500	Hazardous (H)	Maroon

Figure 1

PM2.5	PM10	NO	NO2	NOx	NH3	CO	SO2	O3	Benzene	Toluene	Xylene	AQI	Air_quality
NaN	NaN	0.92	18.22	17.15	NaN	0.92	27.64	133.36	0.00	0.02	0.00	NaN	NaN
NaN	NaN	0.97	15.69	16.46	NaN	0.97	24.55	34.06	3.68	5.50	3.77	NaN	NaN
NaN	NaN	17.40	19.30	29.70	NaN	17.40	29.07	30.70	6.80	16.40	2.25	NaN	NaN
NaN	NaN	1.70	18.48	17.97	NaN	1.70	18.59	36.08	4.43	10.14	1.00	NaN	NaN
NaN	NaN	22.10	21.42	37.76	NaN	22.10	39.33	39.31	7.01	18.89	2.78	NaN	NaN

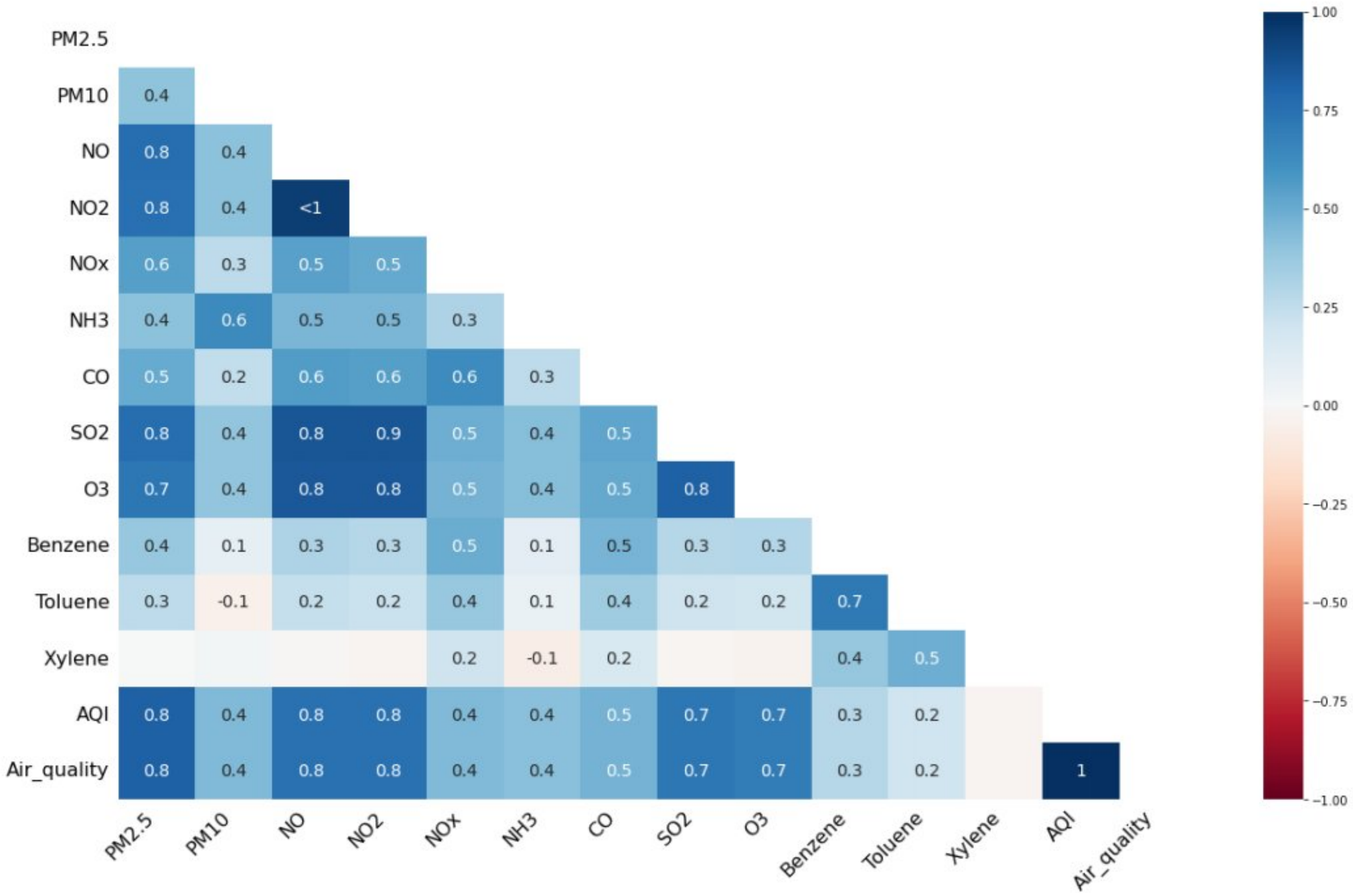
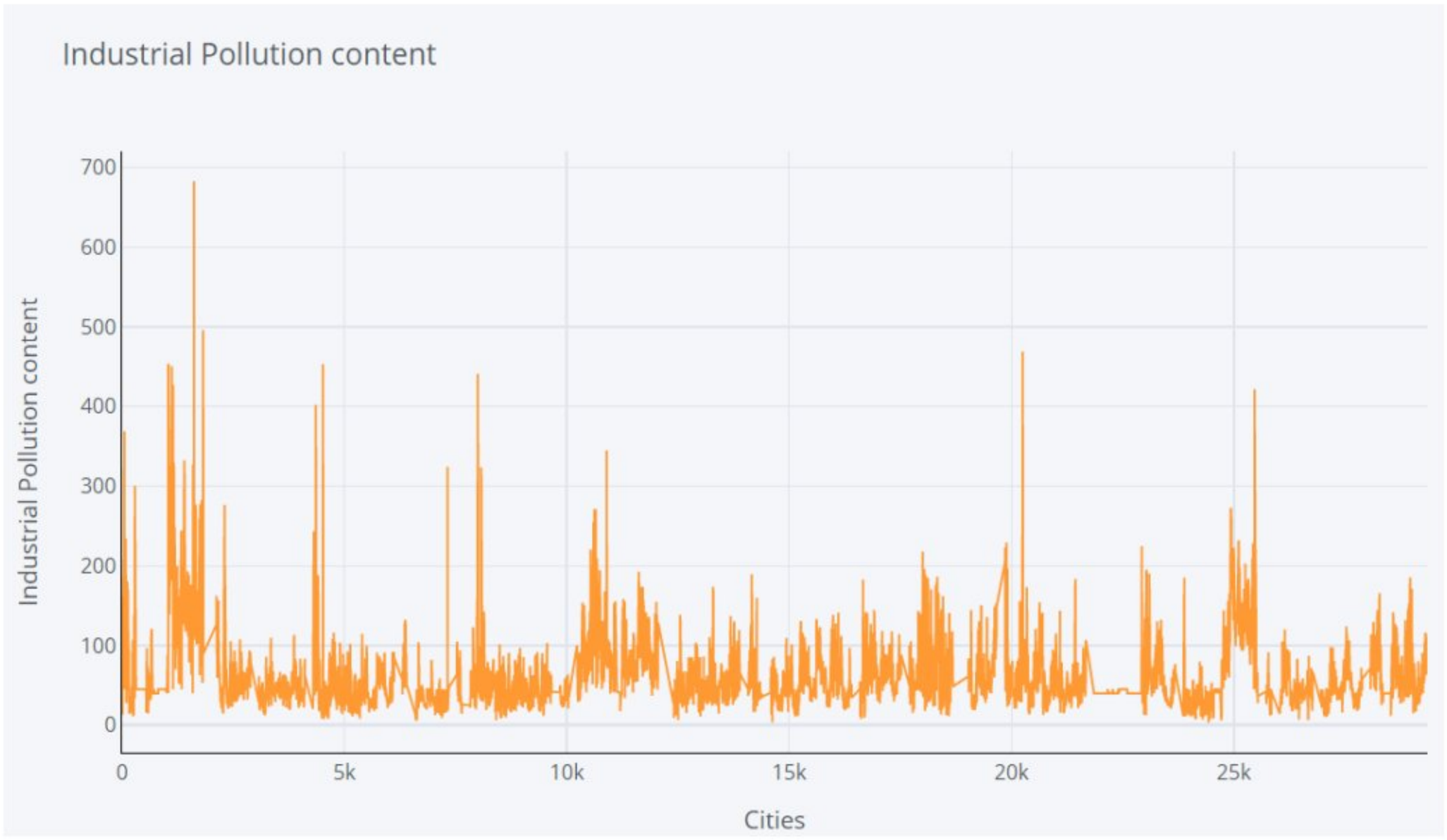


Figure 2



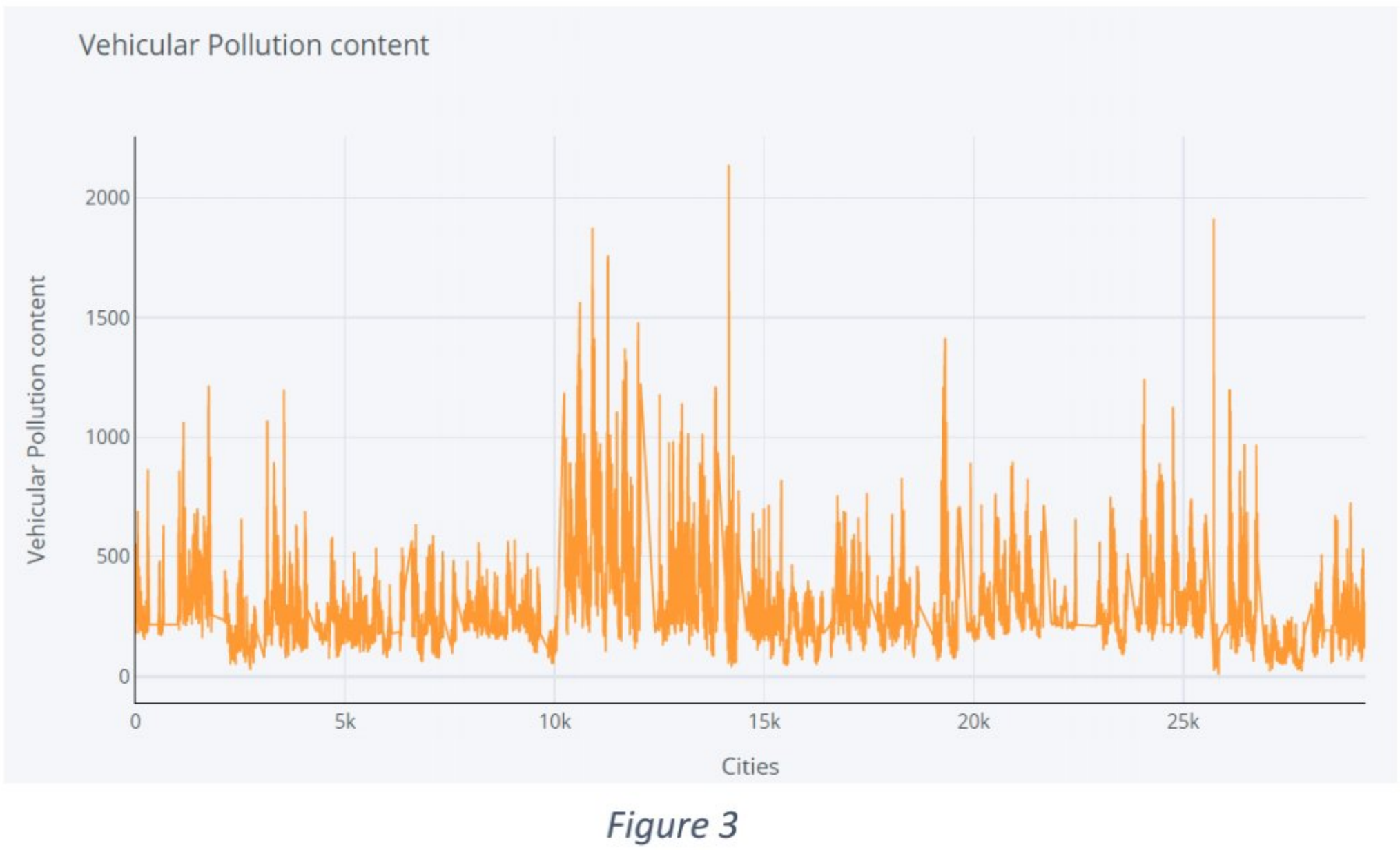
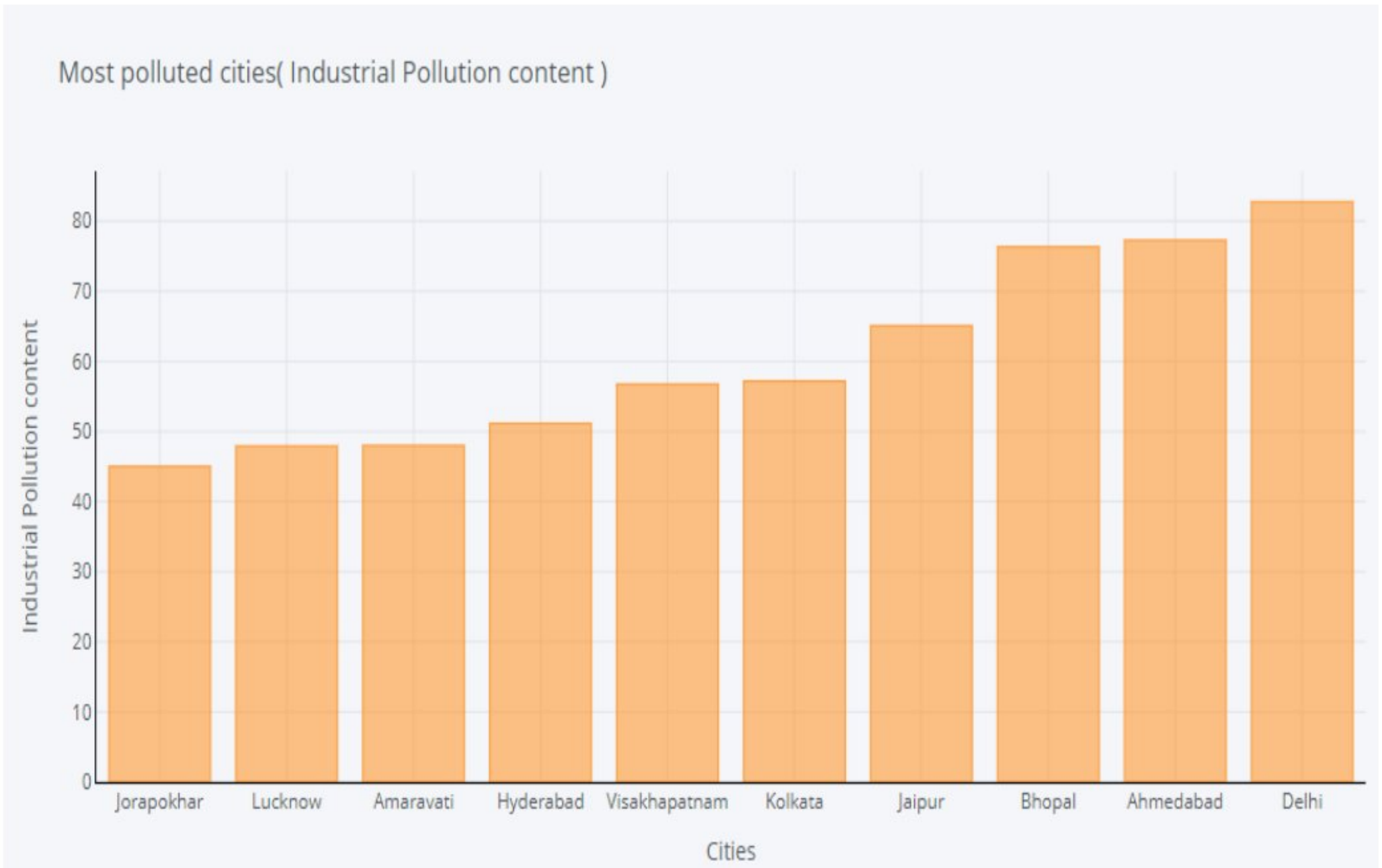


Figure 3



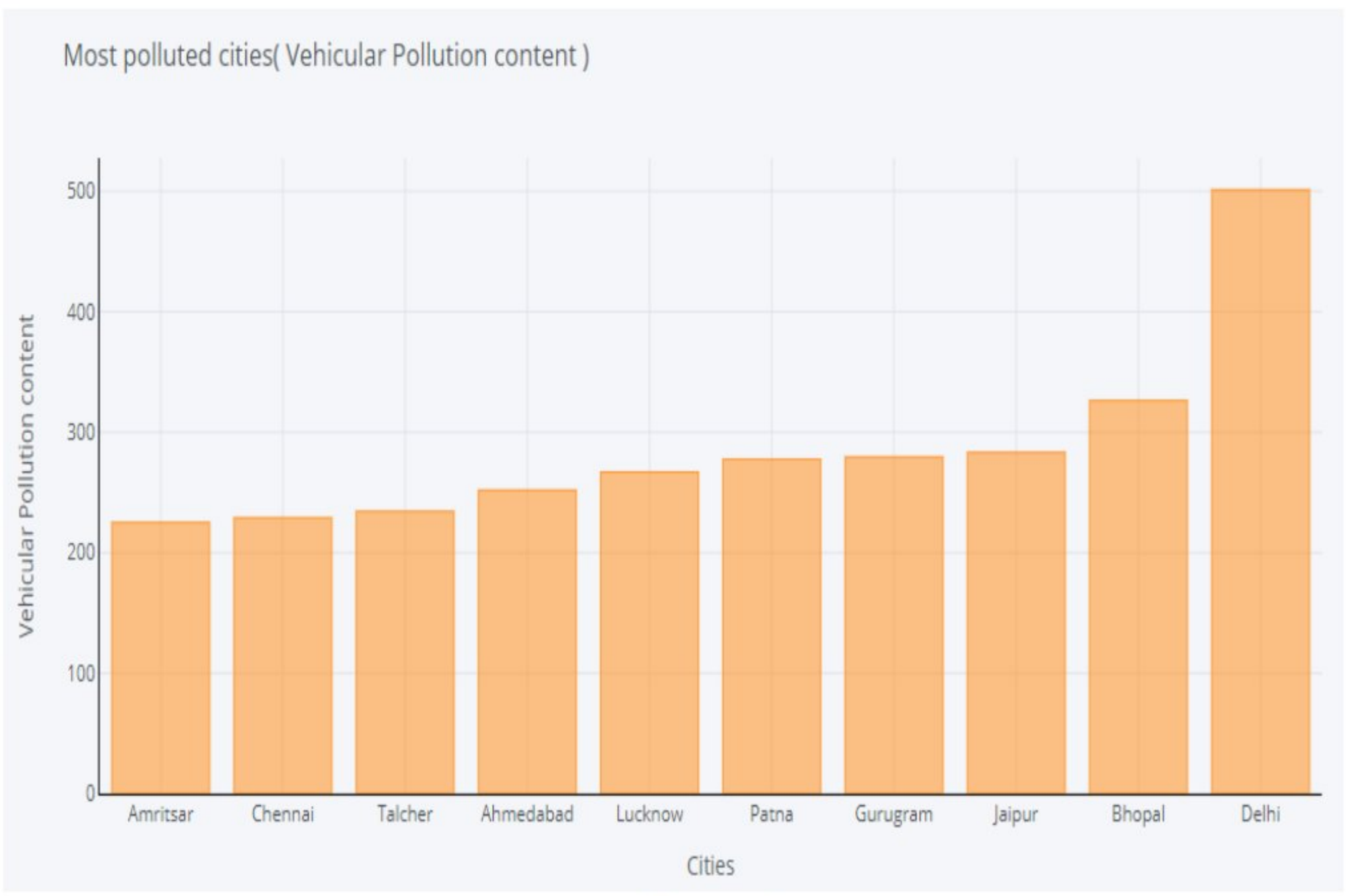
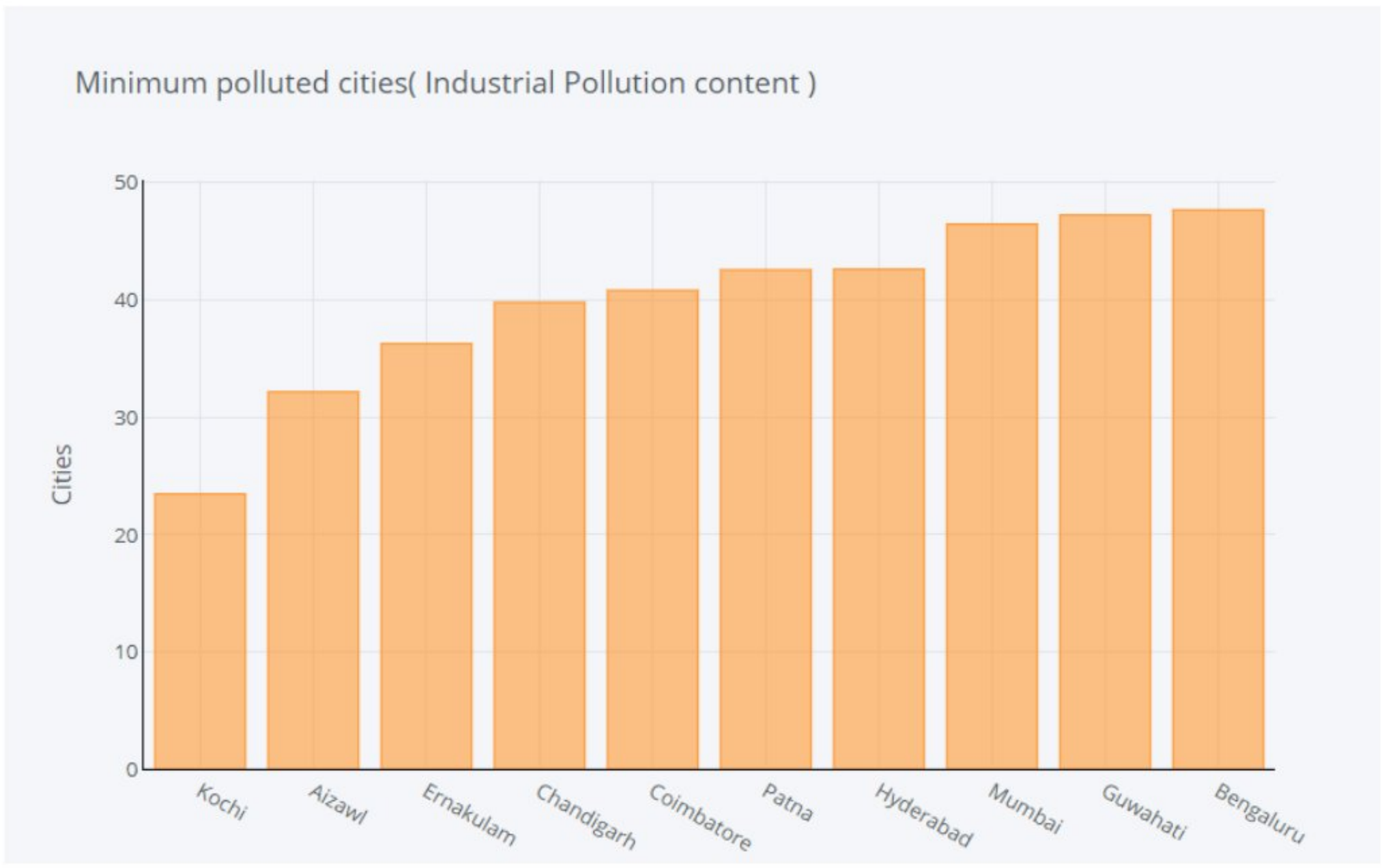
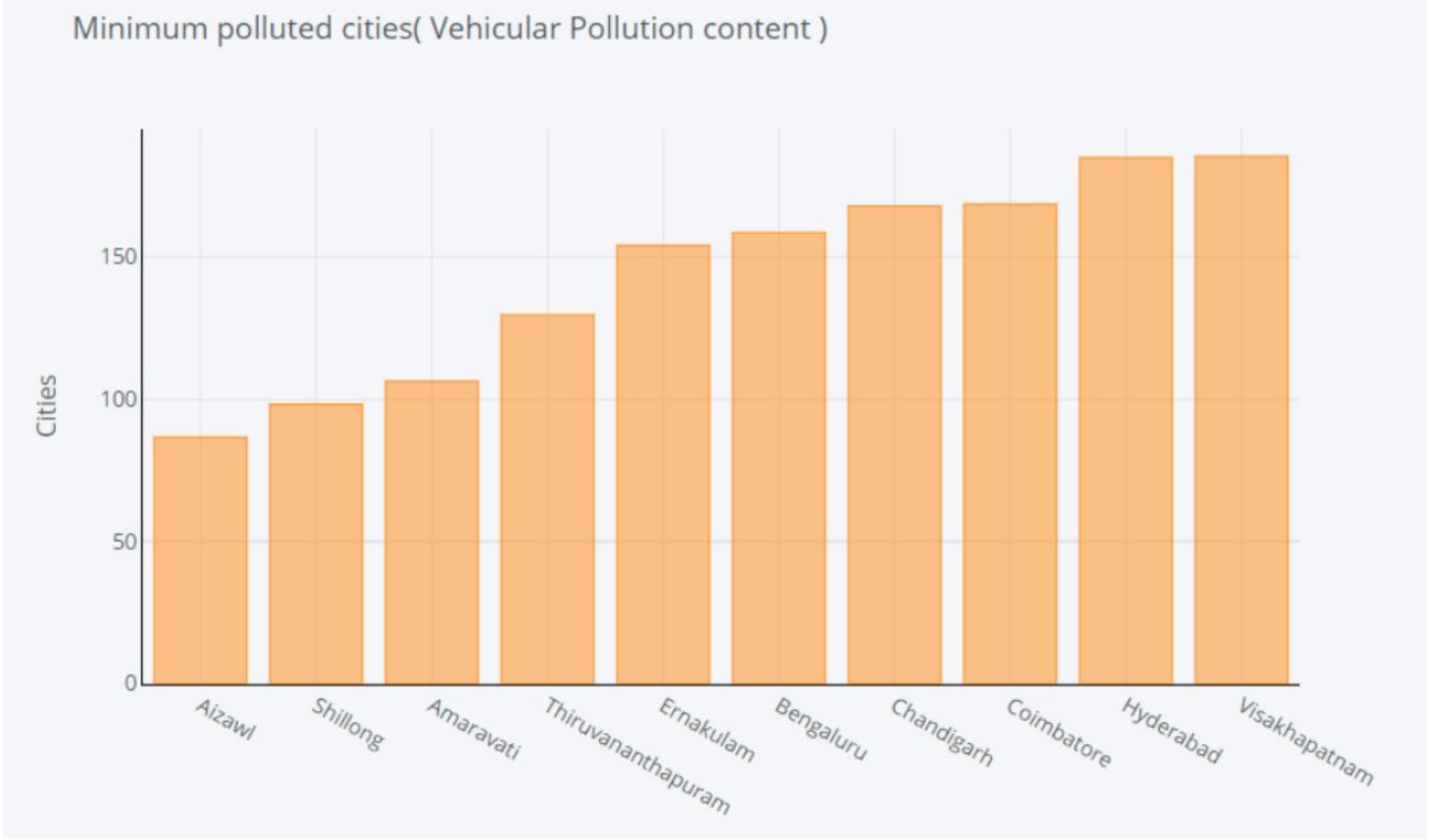
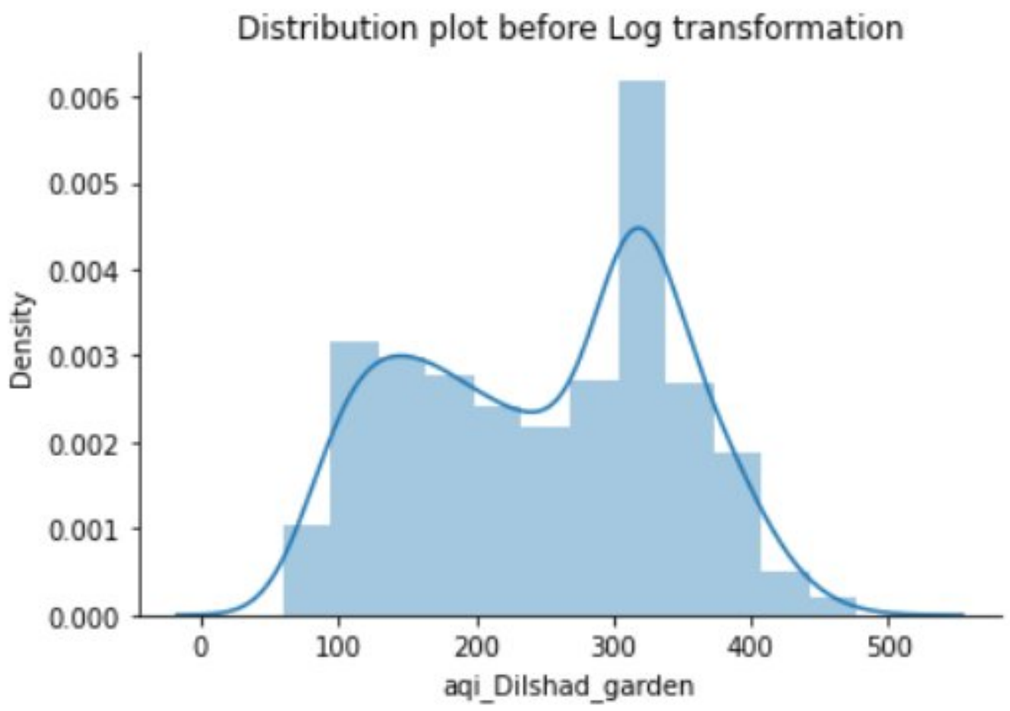


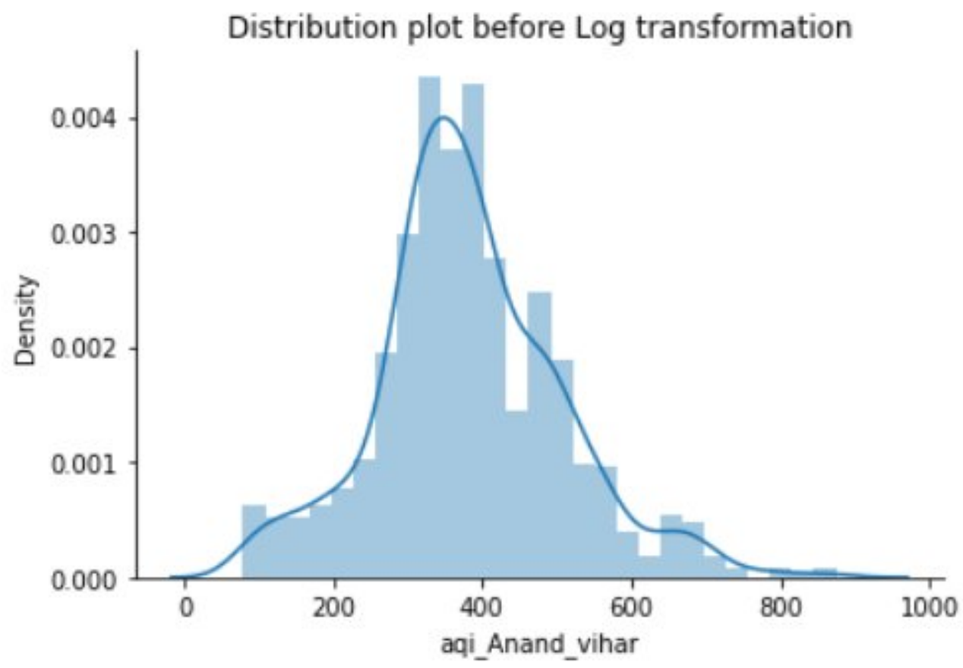
Figure 4

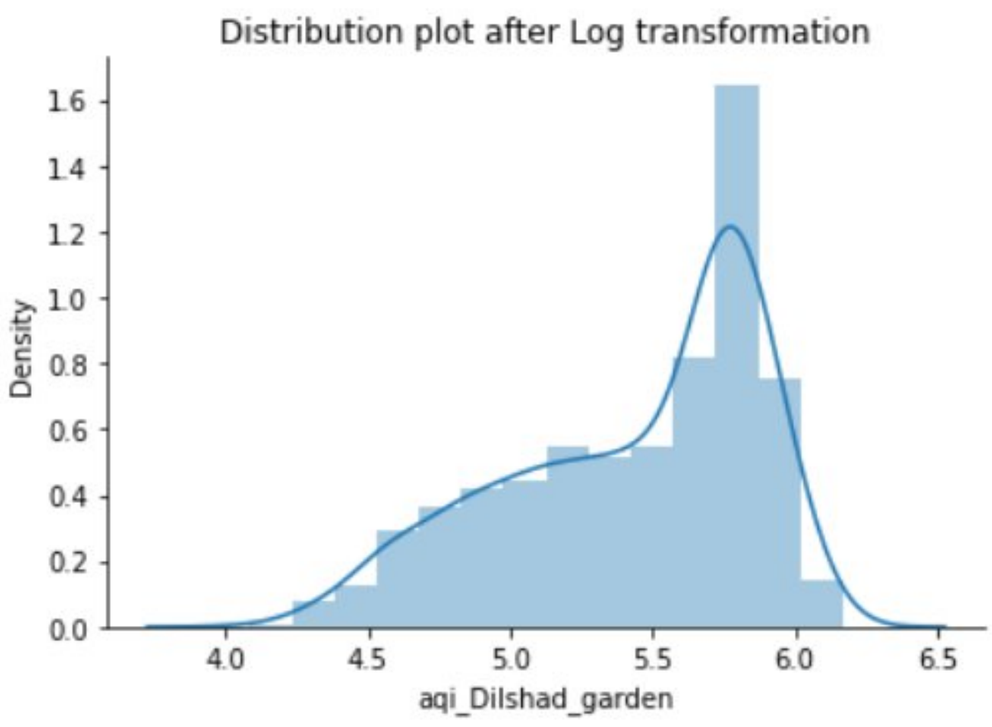

Minimum polluted cities(Industrial Pollution content)
Cities
50
40
30
20
10
0
Kochi
Aizawl
Ernakulam
Chandigarh
Coimbatore
Patna
Hyderabad
Mumbai
Guwahati
Bengaluru

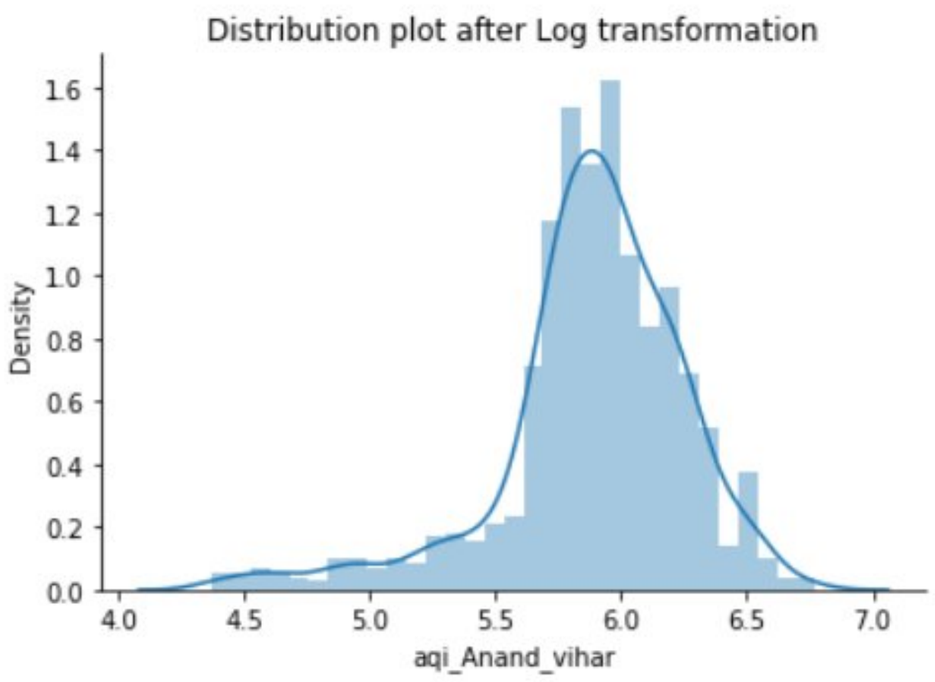

Minimum polluted cities(Vehicular Pollution content)
Cities
150
100
50
0
Aizawl
Shillong
Amaravati
Thiruvananthapuram
Ernakulam
Bengaluru
Chandigarh
Coimbatore
Hyderabad
Visakhapatnam

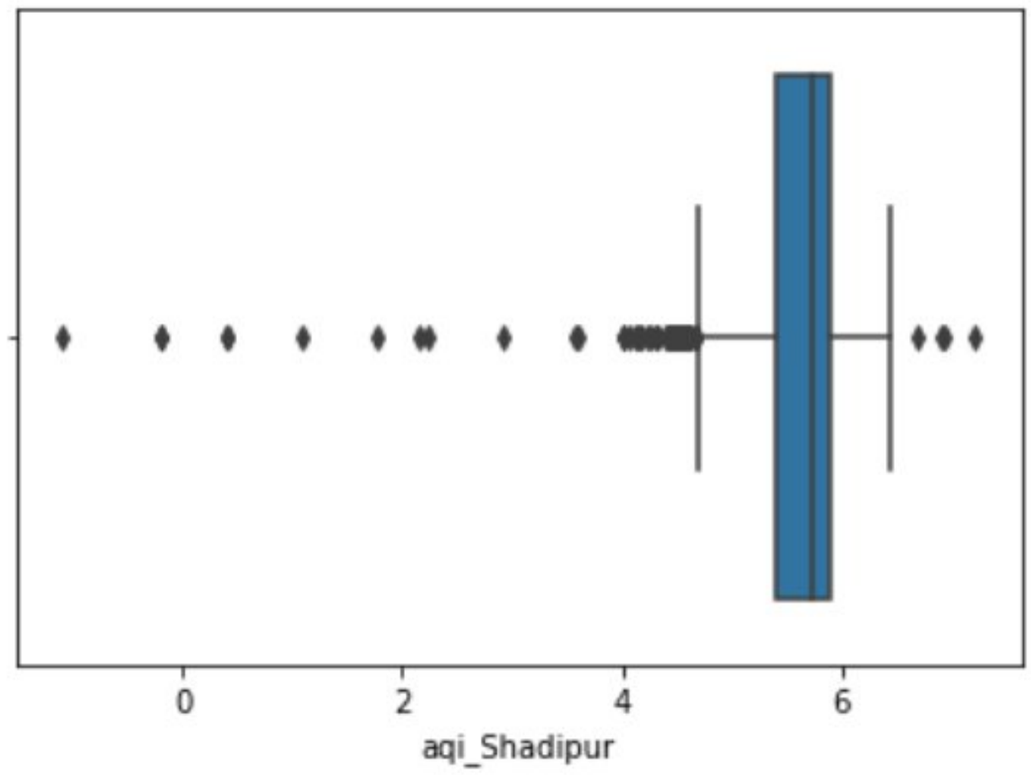
	City	PM2.5	PM10	NO	NO2	NOx	NH3	CO	SO2	O3	Benzene	Toluene	Xylene	AQI	Air_quality
City	1.000000	0.062446	-0.021395	0.047182	-0.078589	0.024948	-0.182565	-0.255886	-0.101327	0.001530	0.002070	-0.062730	-0.047806	-0.121778	-0.053770
PM2.5	0.062446	1.000000	0.520767	0.435245	0.358287	0.375938	0.164615	0.093695	0.120451	0.166064	0.030539	0.132609	0.087813	0.633654	0.408614
PM10	-0.021395	0.520767	1.000000	0.433864	0.352396	0.416523	0.200548	0.023932	0.165468	0.208398	0.027522	0.126400	0.030257	0.447686	0.208080
NO	0.047182	0.435245	0.433864	1.000000	0.465848	0.747409	0.155804	0.221445	0.174881	0.020788	0.045202	0.151930	0.084393	0.438039	0.259407
NO2	-0.078589	0.358287	0.352396	0.465848	1.000000	0.581938	0.151731	0.361750	0.388413	0.289435	0.042179	0.297913	0.188965	0.531952	0.259199
NOx	0.024948	0.375938	0.416523	0.747409	0.581938	1.000000	0.129433	0.231878	0.205912	0.089095	0.049004	0.184764	0.085447	0.436282	0.212592
NH3	-0.182565	0.164615	0.200548	0.155804	0.151731	0.129433	1.000000	-0.014252	-0.048679	0.076295	0.004093	0.020040	-0.030328	0.097916	0.077250
CO	-0.255886	0.093695	0.023932	0.221445	0.361750	0.231878	-0.014252	1.000000	0.476784	0.043311	0.066022	0.285559	0.181215	0.650475	0.215848
SO2	-0.101327	0.120451	0.165468	0.174881	0.388413	0.205912	-0.048679	0.476784	1.000000	0.160489	0.038955	0.273314	0.249808	0.454182	0.170843
O3	0.001530	0.166064	0.208398	0.020788	0.289435	0.089095	0.076295	0.043311	0.160489	1.000000	0.023982	0.135498	0.084323	0.199100	0.030637
Benzene	0.002070	0.030539	0.027522	0.045202	0.042179	0.049004	0.004093	0.066022	0.038955	0.023982	1.000000	0.693310	0.094190	0.052027	0.037358
Toluene	-0.062730	0.132609	0.126400	0.151930	0.297913	0.184764	0.020040	0.285559	0.273314	0.135498	0.693310	1.000000	0.289731	0.288749	0.160761
Xylene	-0.047806	0.087813	0.030257	0.084393	0.188965	0.085447	-0.030328	0.181215	0.249808	0.084323	0.094190	0.289731	1.000000	0.186991	0.092216
AQI	-0.121778	0.633654	0.447686	0.438039	0.531952	0.436282	0.097916	0.650475	0.454182	0.199100	0.052027	0.288749	0.186991	1.000000	0.467065
Air_quality	-0.053770	0.408614	0.208080	0.259407	0.259199	0.212592	0.077250	0.215848	0.170843	0.030637	0.037358	0.160761	0.092216	0.467065	1.000000

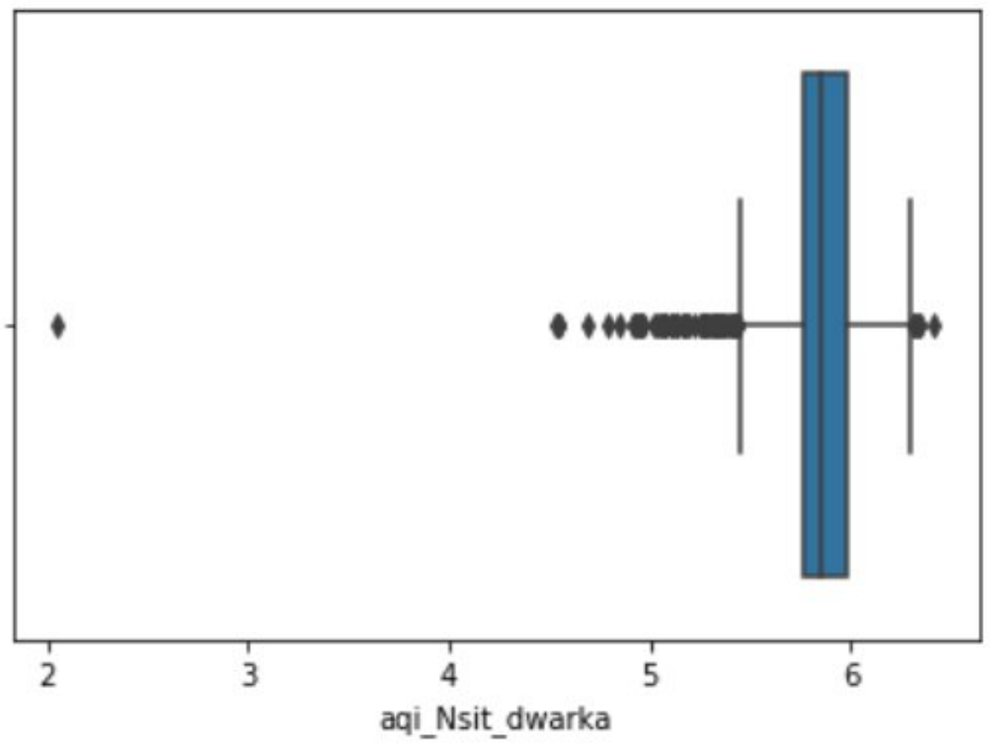


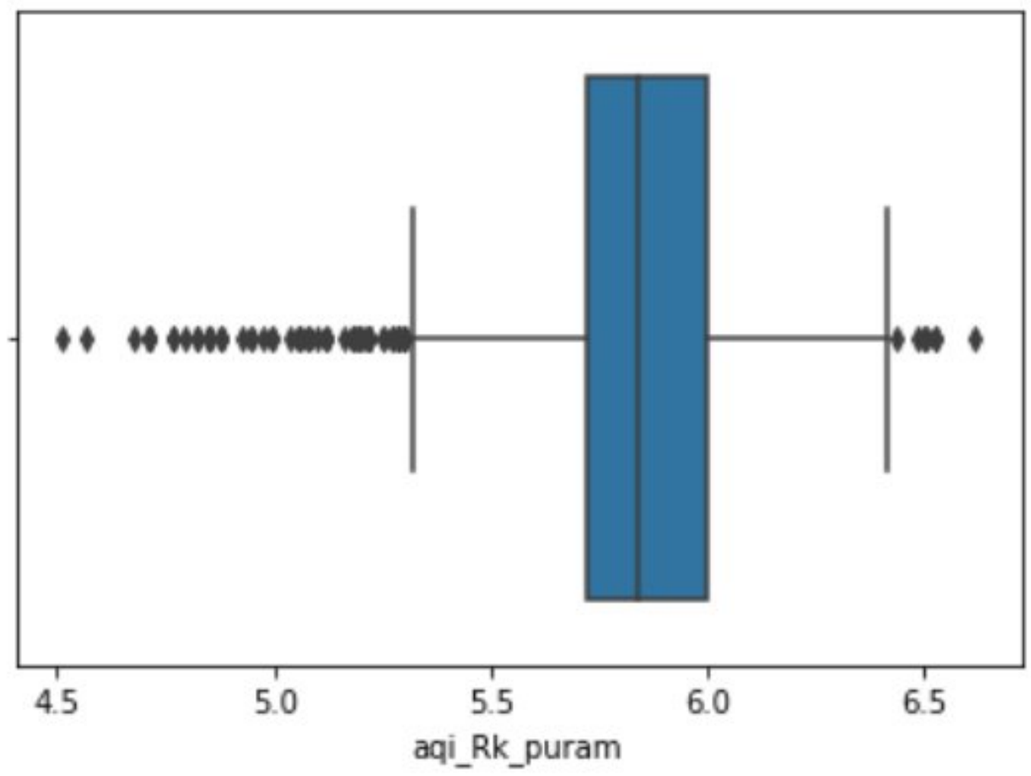


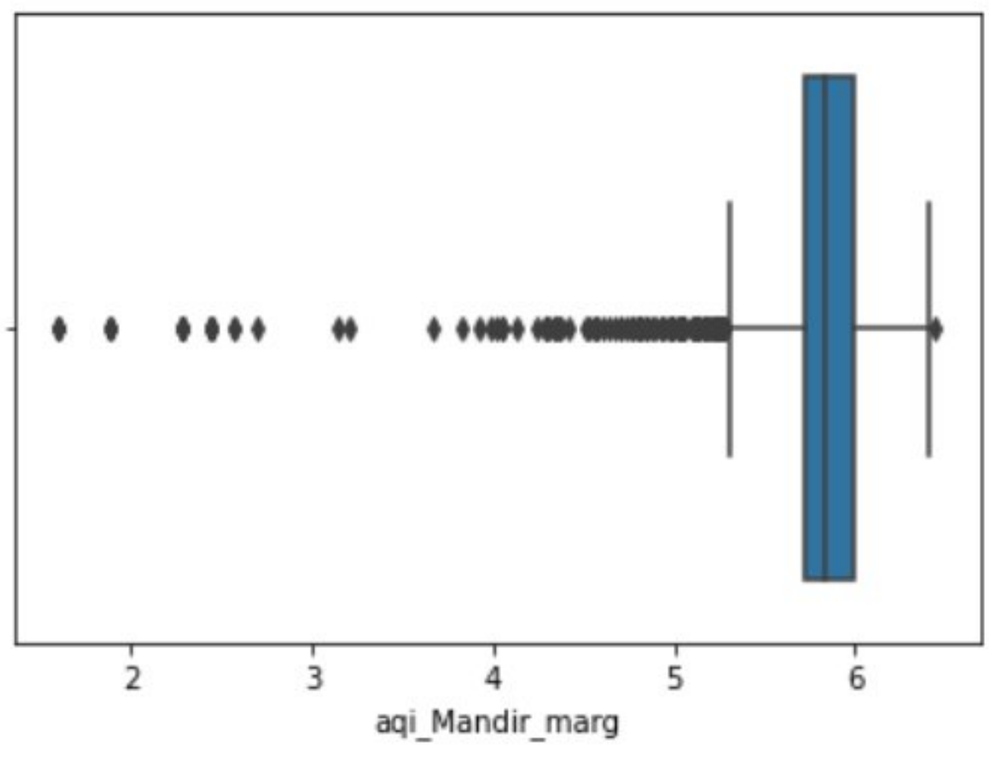


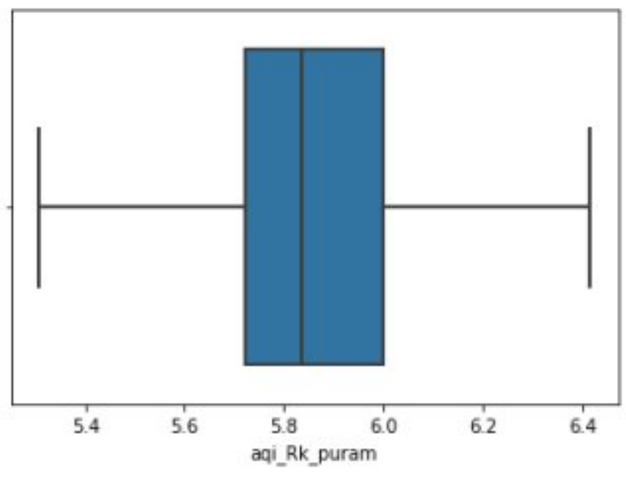


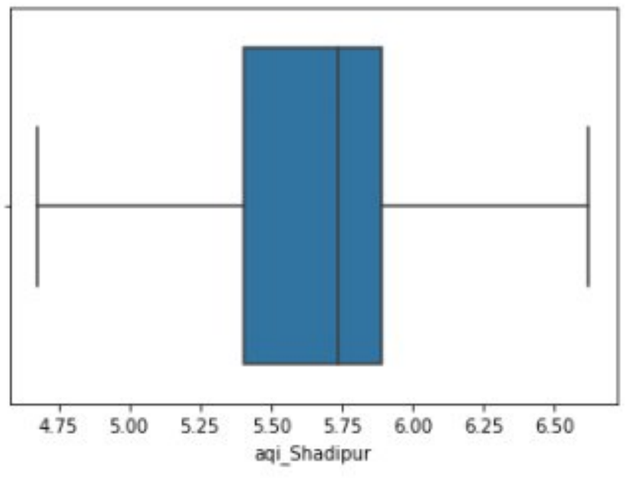

0
2
4
6
aqi_Shadipur

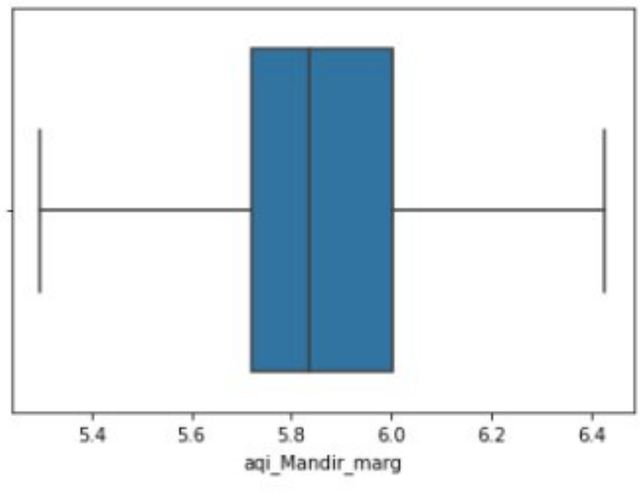

2
3
4
5
6
aqi_Nsit_dwarka

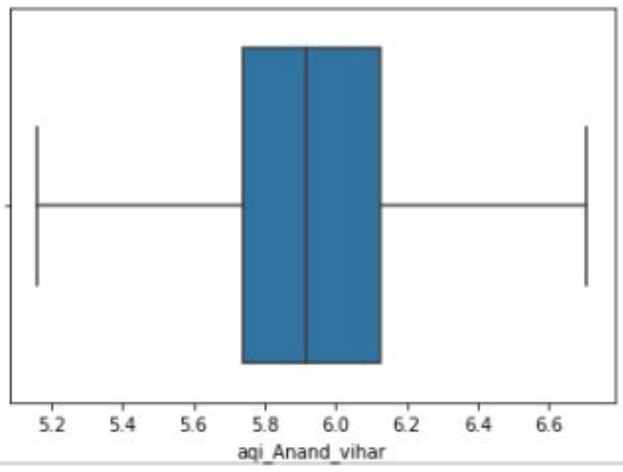

4.5
5.0
5.5
6.0
6.5
aqi_Rk_puram

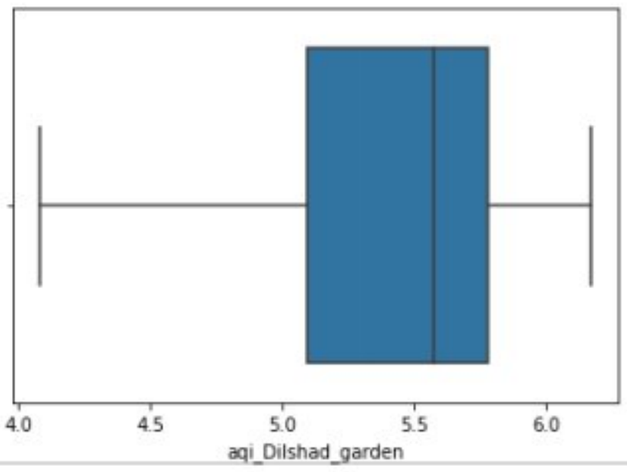

2
3
4
5
6
aqi_Mandir_marg


5.4
5.6
5.8
6.0
6.2
6.4
aqi_Rk_puram


4.75
5.00
5.25
5.50
5.75
6.00
6.25
6.50
aqi_Shadipur


5.4
5.6
5.8
6.0
6.2
6.4
aqi_Mandir_marg


5.2
5.4
5.6
5.8
6.0
6.2
6.4
6.6
aqi_Anand_vihar


4.0
4.5
5.0
5.5
6.0
aqi_Dilshad_garden

References:

1. Urban Air - Microsoft Research
2. U-Air: When Urban Air Quality Inference Meets Big Data Yu Zheng, Furui Liu, Hsun-Ping Hsieh Microsoft Research Asia, Beijing China
3. Spatio-Temporal Prediction modeling node - IBM Documentation
4. CPCB | Central Pollution Control Board
5. CCR (cpcbccr.com)
6. Fundamental Techniques of Feature Engineering for Machine Learning | by Emre Rençberoğlu | Towards Data Science
7. Box-Cox Transformation. When you are dealing with realworld… | by Eugine Kang | Medium iii. A Step-by-Step Explanation of Principal Component Analysis (PCA) | Built In
8. Documentation scikit-learn: machine learning in Python — scikit-learn 0.21.3 documentation
9. KNN Classification using Scikit-learn - DataCamp vi. Documentation of libraries like pandas, numpy, matplotlib, scikit learn.

Glossary:

Outliers: An outlier is an observation that lies an abnormal distance from other values in a random sample from a population. In a sense, this definition leaves it up to the analyst (or a consensus process) to decide what will be considered abnormal These points are often referred to as outliers.

Bivariate Analysis: Bivariate analysis is one of the simplest forms of quantitative (statistical) analysis. It involves the analysis of two variables (often denoted as X, Y), for the purpose of determining the empirical relationship between them.....Like univariate analysis, bivariate analysis can be descriptive or inferential.

Inferential Statistics: Inferential statistics allows you to make predictions ("inferences") from that data.

Discrimination Ability: The ability of that particular attribute in classification performance when all the other attributes are absent from the neural network training.

Categorical Attributes: Categorical features can only take on a limited, and usually fixed, number of possible values. For example, if a dataset is about information related to users, then you will typically find features like country, gender, age group, etc.