Course Project Milestone 4- Preliminary Analysis- Air Quality Prediction

Guruprasad Velikadu Krishnamoorthy and Hemalatha Subbiah

College of Science and Technology, Bellevue University

DSC630-T302: Predictive Analysis

Professor. Fadi Alsaleem

February 11, 2024

Air Quality Prediction

Introduction

Air pollution is a significant concern in the modern world. With the rise of the Industrial Revolution in the 19th century and the invention of new machinery and technologies, economies worldwide saw explosive growth in the following decades, accompanied by the migration of farmers from the countryside to fast-growing, crowded cities in search of factory jobs. (Kiger, 2021) While the industrialists were amazed by the profits being made, the governments were astonished by the progress made in the economy; little did they all know the side effects that changed the landscape of the earth forever, for the worse. The emission of gases from factories and the pollutants from automobiles had an everlasting impact on the face of the earth. In this project, the effects of air pollution and the ways to combat it by being able to predict it using machine learning algorithms will be discussed in detail.

Problem statement:

On average, it is estimated that a person inhales about 2000 gallons of air daily. Hence, it is essential that the air we breathe is of good quality. The polluted air, when inhaled, gets straight into our lungs, then enters the bloodstream, and can cause more damage to internal organs such as the brain, heart, etc., and young children are the most affected ones.

The aim of the project is to develop a machine-learning model capable of predicting the Air Quality Index (AQI) based on various environmental parameters and pollutant concentrations. The model should be capable of providing accurate AQI predictions for future time points, allowing for early detection of potential air quality issues.

Why the Problem is important:

According to a recent study, air pollution-related ailments are the fourth largest contributing factor to premature deaths, and about 4.5 million deaths around the world are reported to be related to air pollution. Some air pollutants such as mercury, Lead, and benzene can cause several health issues and, in some cases, even death. (10 Things You Never Knew Could Cause Lung Cancer, n.d.) In New Delhi, India, where the air quality is ranked among the worst, people reported that it felt like breathing poison during extreme smog conditions. The level of PM2.5 pollutant, which is small enough to enter the bloodstream, was reported to be 25 times more than the limits recommended by the World Health Organization in the city. (“Like Breathing Poison”: Children in India’s Delhi Hit Hard by Smog, n.d.)

Though some measures were taken in the last decade to bring awareness to people about the impacts of air pollution, it was too little, and it will be too late if all the countries around the globe don't work together to solve this looming Problem. If the current trend continues, in the future, people may be forced to pay for clean, breathable air, making it an absolute priority to take measures to clean up the air and provide a sustainable and healthy environment for our future generations.

Who would be interested in solving the Problem?

Pretty much everyone is affected by the rising air pollution-related problems around the world. Children, older adults, and people living with asthma and other breathing-related disorders are the most affected. People who have lower incomes and those who live near the sources of pollution, such as factory workers, may also face greater harm. Though it is believed that people in bigger cities are the most affected due to vehicle and industrial emissions, small cities and towns are not spared either. Agricultural activities, Fertilizer manufacturing, and livestock production that release Methane can all contribute to air pollution in rural areas. Hence, this is a universal problem that affects most of the population around the globe.

The availability of comprehensive air quality datasets can help develop predictive models that can forecast the Air Quality Index (AQI) with high accuracy. Being able to predict the air quality based on the pollutant concentration can not only alert the public to take precautions but can also help take measures in advance to be prepared or to avoid the situation altogether.

Source of the Data:

The data is extracted from the Environmental Protection Agency website. (Download Files | AirData | US EPA, 2015). A total of nine datasets will be used to predict the air quality in this project, such as Ozone concentration datasets, SO2, NO2, and Carbon Monoxide concentration datasets; datasets of pollutants such as PM2.5 and PM10; datasets containing meteorological data such as Temperature, Pressure, Humidity, etc. All these datasets are for the year 2022.

Some of the commonly used fields in the dataset are listed below:

- Columns to identify the site location where the readings were measured:
  - State code, County Code, Site Number, Latitude, Longitude, Local Site name, Address, State Name, County Name, City Name.
- Columns to identify the Pollutant/ Gases/ the Metrological quantity:
  - Parameter Name: Represents the Parameter of the Pollutant /Particulate/Toxin/Meteorological measure.
  - Parameter Code: Unique code assigned to the parameter describing the Pollutant /Particulate / Toxin /Meteorological measure.
  - Units of Measure: The unit in which the parameter is measured.
  - Observation Count: Number of observations captured on the specified date in the site location.
  - Arithmetic Mean: The Mean value of the quantity of the parameter captured on the given date at the given site.
- Column to identify the Air Quality:
  - AQI: Represents Air Quality Index value measured on the specified date at the specified site location.
- Date fields:

    - Date Local: Date when the parameter was measured and recorded.

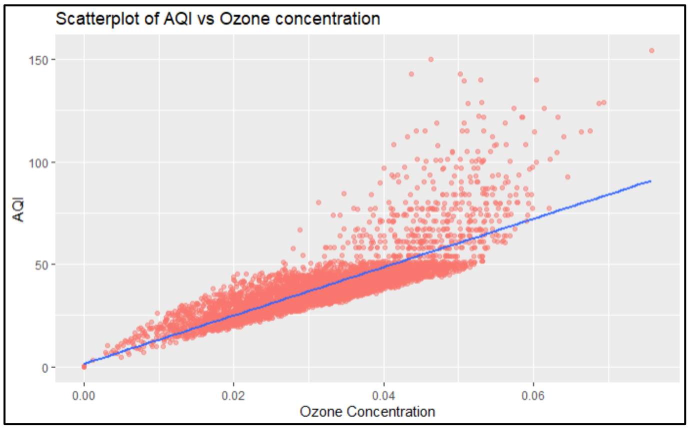
- Columns for Pollutants:
    - Ozone (O3): Concentration of ozone in the air
    - Carbon Monoxide (CO): The Concentration of carbon monoxide in the air.
    - Nitrogen Dioxide (NO2): Concentration of nitrogen dioxide in the air.
    - Sulfur Dioxide (SO2): Concentration of sulfur dioxide in the air.

Why is the data useful to the Problem:

The datasets contain information about the concentration of Gases such as Ozone, Sulfur Dioxide (SO2), and Nitrogen Dioxide (NO2) measured at different sites around the US for the year 2022. This, along with the concentration of particulates such as PM2.5 and PM10.0, and other parameters such as Temperature, pressure, and humidity, can provide valuable information to predict the Air Quality index. As all these metrics are available in individual datasets, they must be combined to extract useful insights from them. There are about 250,000 rows on average in these datasets, thus providing sufficient data required for the research.

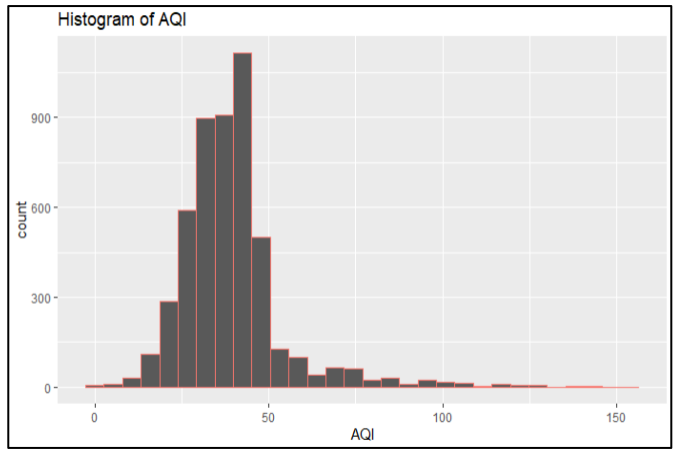
Exploratory Data Analysis and the Visualizations:

As most of the features used in the project are numerical, scatter plots to show the relationship between the pollutants or harmful gases versus the AQI (Air Quality Index) can be a useful visualization to understand the relationship between the two.



(Figure 1)

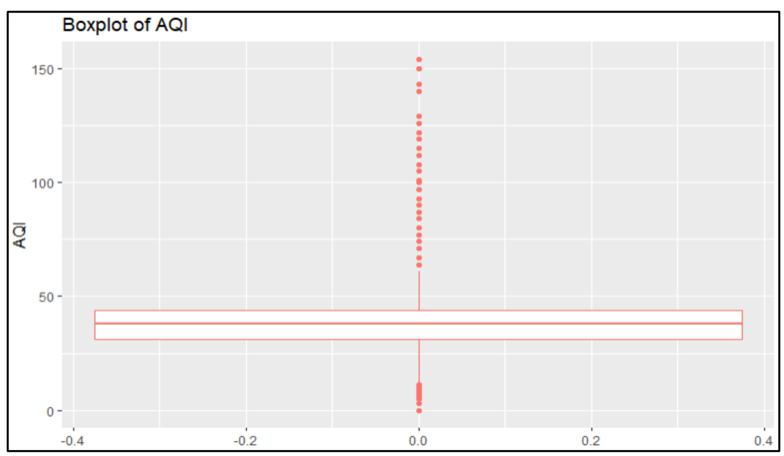
The scatter plot (Figure1) of AQI vs Ozone indicates that there is a positive correlation between and most of the AQI is between 20 and 50.



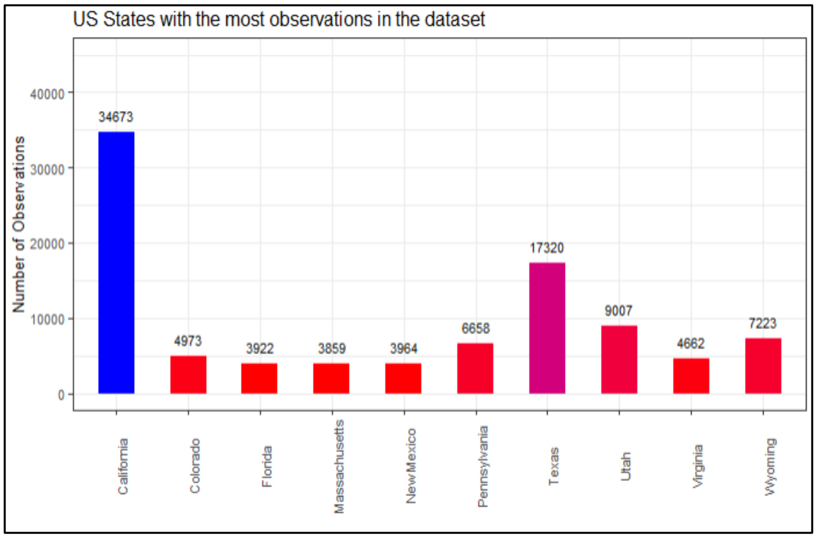
(Figure 2)

Also, a histogram (Figure 2) is plotted on the AQI to identify the outliers, and based on the nature of the outlier, a decision can be made to either eliminate them or impute the data.

During the process of Exploratory Data Analysis, null values were handled, and outliers were identified using the boxplot and manual method using R (Figure 3)



(Figure 3)



(Figure 4)

A Bar plot showing the distribution of the data across the US states has been plotted (Figure 4). The plot indicates that most observations in the dataset are from California and Texas followed by Utah and Wyoming.

```
# Checking for Nulls in the Ozone Dataframe
colSums(is.na(ozone_df))
```

```
# Computing the first and third quartiles
q1 <- quantile(ozone_df$AQI, 0.25)
q3 <- quantile(ozone_df$AQI, 0.75)
iqr <- q3 - q1
# Calculate the lower and upper cutoffs for outliers
lower <- q1 - 1.5 * iqr
upper <- q3 + 1.5 * iqr
# Filter AQI to find outliers
AQI_outliers <- ozone_df %>%
  filter(AQI > upper | AQI < lower)
# Printing the top few rows of Outliers
kbl(head(AQI_outliers[1:6,c(1:8)]), caption = "Outliers in the AQI",booktabs = T) %>% kabl
```

Process of the Data Preparation:

The data for this project was collected by combining multiple CSV files each containing information about the pollutant concentration, meteorological data, etc. Each row represents information about the pollutant concentration and the Air Quality Index for a location in the US on a given day and has additional details such as state, county, pollutant description, etc.

Data format conversion

After creating data frames for each dataset, data conversion was performed by converting some character columns to Factors and Dates.

```
# Function to create a dataframe by reading the contents from csv file
create_dataframe <- function(df,file_name){
    df <- read.csv(file_name,stringsAsFactors=FALSE)
    return(df)
}
# Function to rename the columns into required format
rename_columns <- function(df){
    df <- df %>% select(state_code=State.Code,county_code=County.Code,site_num=Site.Num,
                                  date_local=Date.Local,
                                 new_col=Arithmetic.Mean,AQI
                                 )
    return (df)
}
# Function to format the columns by changing the datatype to factors and Dates
format_columns <- function(df){
    df$state_code <- as.factor(df$state_code)
    df$county_code <-  as.factor(df$county_code)
    df$site_num <- as.factor(df$site_num)
    df$date_local <- as.Date(df$date_local)
    df$AQI[is.na(df$AQI)] <- 0
    return (df)
}
```

Renaming the features and merging the datasets

The field names were renamed for easier computations by removing spaces in the column names and replacing them with underscores. To prepare the final data, the Dataset containing NO2 data was joined with all other datasets based on common columns such as State ID, County ID, Site ID, and the Date of Observation.

```
# Calling the Join function to perform the joins
merge1_df=merge_dataframes(NO2_df,ozone_df,col_name="ozone")
merge2_df=merge_dataframes(merge1_df,SO2_df,col_name="SO2")
merge3_df=merge_dataframes(merge2_df,CO_df,col_name="CO")
merge4_df=merge_dataframes(merge3_df,df_81102,col_name="PM")
merge5_df=merge_dataframes(merge4_df,wind_df,col_name="wind")
merge6_df=merge_dataframes(merge5_df,temp_df,col_name="temp")
merge7_df=merge_dataframes(merge6_df,press_df,col_name="press")
merge8_df=merge_dataframes(merge7_df,rh_dp_df,col_name="RH")
```

Null handling and Duplicate checks

The nulls were then handled by replacing them with Median values for the column. The dataset was then checked for duplicates in the key columns and only the distinct values were retained.

```
}
# Function to join dataframes using the "left join"
merge_dataframes <- function(df1,df2,col_name){
    # Performs left join  and selects required columns
    merge_df <- left_join(df1,df2,by=c("state_code","county_code","site_num","date_local")
    # Replaces the nulls in numeric columns after left join with median values
    merge_df[[col_name]][is.na(merge_df[[col_name]])]<-median(merge_df[[col_name]],na.rm=T
    return(merge_df)
}
```

Model Development

Linear Regression Using R:

As the project is about predicting the Air quality index, which is a continuous Numeric variable, the Regression algorithm is used for building the models. Multiple models were built using both Python and R in this project. The final dataset derived by combining multiple datasets was split into Training and Test sets in approximately 75:25 ratio. Also, feature extraction or selection was deployed to reduce the number of features in the dataset. The Linear regression models were then trained using the training dataset and the performance was tested on the test set.

```
### Model Building
```{r}
# Standardizing the values to build the model using the scale function
merge8_df$NO2 <- scale(merge8_df$NO2)
merge8_df$ozone <- scale(merge8_df$ozone)
merge8_df$SO2 <- scale(merge8_df$SO2)
merge8_df$CO <- scale(merge8_df$CO)
merge8_df$wind <- scale(merge8_df$wind)
merge8_df$temp <- scale(merge8_df$temp)
merge8_df$press <- scale(merge8_df$press)
merge8_df$RH <- scale(merge8_df$RH)
```

```{r}
# Creating training and test datasets by splitting them in 75:25 ratio
gp <- runif(nrow(merge8_df))
train_df <- merge8_df[gp < 0.75, ]
test_df <- merge8_df[gp >= 0.75, ]
# Printing the rows in the train and test sets
nrow(train_df)
nrow(test_df)
```

 [1] 116618
 [1] 38850

```{r}
# Creating a linear regression model on the train dataset
model1 <- lm(AQI ~ NO2+ozone+SO2+PM+CO+wind+temp+press+RH+date_local
               , data=train_df )
summary(model1)
```
```

In R, the backward fit method from the olsrr package was used to identify the less impactful features before building the model. Then ols_step_best_subset from the same package was used to identify the model with better prediction results(R2) from each combination of the features used to build the model. The figure <> indicates that the model yields the best results with all the features included.

```{r}
library(olsrr)
backwardfit.p<-ols_step_backward_p(model1,prem=.05)
backwardfit.p
```

```
[1] "No variables have been removed from the model."
```

```{r}
modcompare<-ols_step_best_subset(model1)
modcompare
plot(modcompare)
```

The results of the linear regression model in R is shown in the figure below.

One of the things that stands out in the model results is that the model yields a high R-squared value of 0.83 which indicates the proportion of the variance in the AQI that is explained by the features of the model. The R-squared and adjusted R-squared values being very close indicates that the additional factors in the model are not being penalized and the model is a good representation of the data.

```
```{r}
model2 <- lm(AQI ~ NO2+ozone+SO2+PM+CO+wind+temp+press+RH+date_local
              , data=train_df )
summary(model2)
```
```

```
Call:
lm(formula = AQI ~ NO2 + ozone + SO2 + PM + CO + wind + temp +
    press + RH + date_local, data = train_df)

Residuals:
    Min      1Q  Median      3Q     Max
-27.715  -2.957  -0.996   2.157  78.297

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept) 91.6099824  2.6141941  35.043  < 2e-16 ***
NO2         10.7558082  0.0164190 655.082  < 2e-16 ***
ozone        0.1120981  0.0154664   7.248 4.26e-13 ***
SO2         -0.0844251  0.0139330  -6.059 1.37e-09 ***
PM           0.0263163  0.0012579  20.920  < 2e-16 ***
CO          -0.7082200  0.0167892 -42.183  < 2e-16 ***
wind        -0.1740548  0.0143214 -12.153  < 2e-16 ***
temp         0.0325254  0.0148194   2.195   0.0282 *
press       -0.0895958  0.0138854  -6.453 1.10e-10 ***
RH          -0.6858194  0.0147154 -46.605  < 2e-16 ***
date_local  -0.0040032  0.0001363 -29.360  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 4.772 on 116559 degrees of freedom
Multiple R-squared:  0.8302,	Adjusted R-squared:  0.8301
F-statistic: 5.697e+04 on 10 and 116559 DF,  p-value: < 2.2e-16
```

The Root mean square Error (RMSE) is one of the performance indicators for the model and is consistent for both the training and test datasets. The RMSE, which measures the average difference between the AQI values predicted by the model versus the actual AQI, indicates that we can expect an error of up to 4.77 while using the model predictions.

```{r}
### r_squared
r_squared <- function(predcol, ycol) {
  tss = sum( (ycol - mean(ycol))^2 )
  rss = sum( (predcol - ycol)^2 )
  1 - rss/tss
}
### rmse
rmse <- function(predcol, ycol) {
  res = predcol-ycol
  sqrt(mean(res^2))
}

rmse_train <- rmse(train_df$pred_AQI,train_df$AQI)
sprintf("The RMSE value of Training Dataset is %s", round(rmse_train,2) )
rmse_test <- rmse(test_df$pred_AQI,test_df$AQI)
sprintf("The RMSE value of Training Dataset is %s", round(rmse_test,2) )

# Evaluate the r-squared on both training and test data.and print them
rsq_train <- r_squared(train_df$pred_AQI,train_df$AQI)
sprintf("The R-squared value of Training Dataset is %s", round(rsq_train,2) )
rsq_test <- r_squared(test_df$pred_AQI,test_df$AQI)
sprintf("The R-squared value of Test Dataset is %s", round(rsq_test,2) )
```

```
[1] "The RMSE value of Training Dataset is 4.77"
[1] "The RMSE value of Training Dataset is 4.75"
[1] "The R-squared value of Training Dataset is 0.83"
[1] "The R-squared value of Test Dataset is 0.83"
```

Using Python:

The below techniques were applied to the final dataset before the model development in Python.

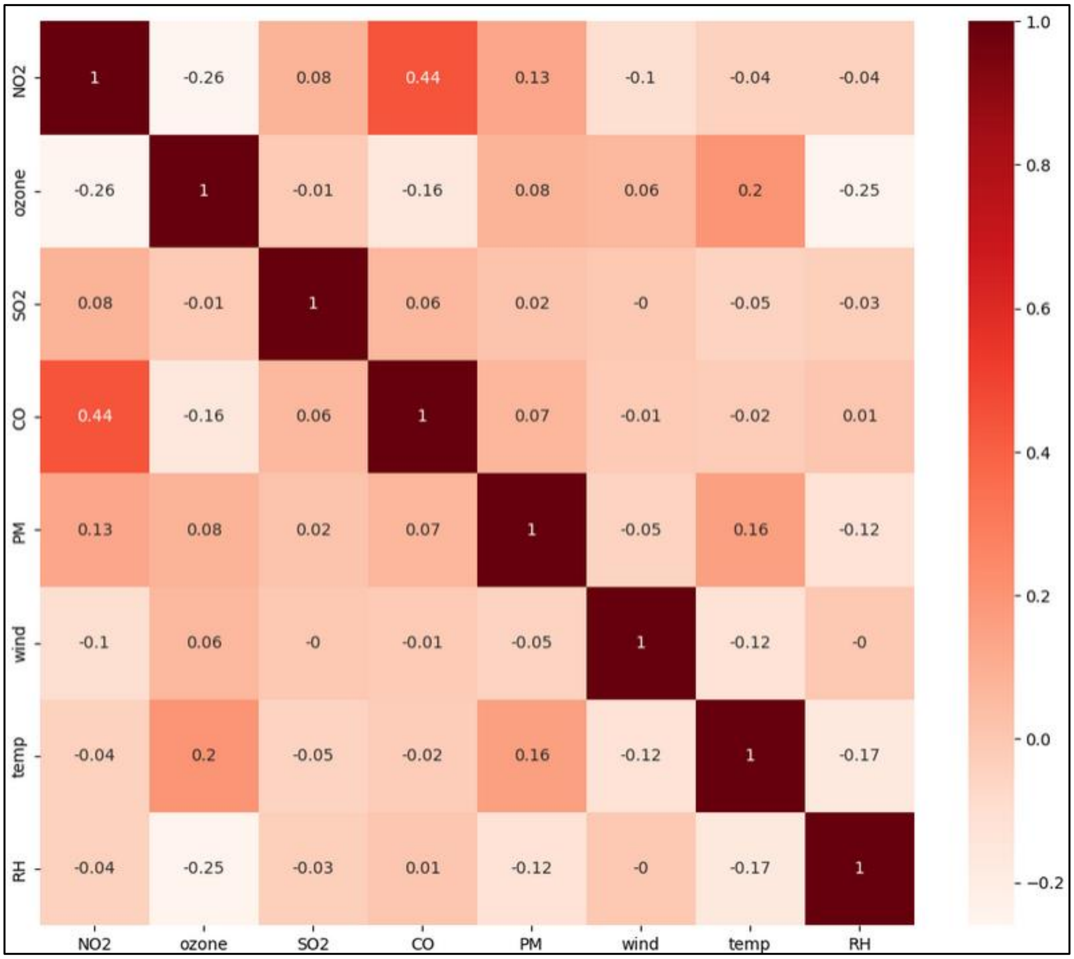
- Feature engineering techniques
- Experimenting with different algorithms
- Hyperparameter tuning to optimize the model performance.

As the project is about predicting the Air quality index, which is a continuous Numeric variable, the Regression algorithm is the appropriate machine learning algorithm(s) based on the nature of the problem and the type of data.

Feature Reduction based on the Correlation matrix:

The correlation matrix was leveraged for analyzing and identifying pairs of features with high correlation coefficients (close to 1 or -1) and the features with high correlation were removed to reduce redundancy (Figure 5). Only the features that had a high correlation with the target variable were retained for model building.

Correlation Matrix



(Figure 5)

Feature Reduction using the PCA method

The PCA- the dimensionality reduction technique was implemented to preserve most of the variability in the data and reduce the number of features in the dataset.

```python
# Standardize the feature matrix
features_test = scaler_pca.transform(X_test)
# Conduct PCA
features_pca_test = pca.transform(features_test)
# Show results
print("Original number of features:", features_test.shape[1])
print("Reduced number of features:", features_pca_test.shape[1])
X_test.shape[1]
```

```
Original number of features: 8
Reduced number of features: 6
```

Model building in Python:

The below regression algorithms were explored as a part of model development:

- Linear Regression
- Decision Tree Regression
- Random Forest Regression
- Gradient Boosting
- K Neighbors Regressor

Before starting the experimentation, the dataset was split into training and testing sets. The training set was used to train the models, while the testing set was used to evaluate the model performance.

For each regression algorithm chosen, the below steps were followed

- Train the model on the training data.
- Use the trained model to make predictions on the testing data

- Evaluate the performance of the model using appropriate regression evaluation metrics such as Mean Squared Error (MSE), Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), R-squared (R2), etc. d. Repeat steps a-c for each algorithm.

The below code incorporated all the steps discussed above.

```
def model_assess(X_train, X_test, y_train, y_test, model, title ):
    """
    This function will be used to build the model. It takes train and test attribues as the
    input and returns model parameters as output
    """
    model.fit(X_train, y_train)
    y_train_pred = model.predict(X_train)
    y_test_pred  = model.predict(X_test)

    train_mse = mean_squared_error(y_train, y_train_pred)
    train_r2 = r2_score(y_train, y_train_pred)
    test_mse = mean_squared_error(y_test, y_test_pred)
    test_r2 = r2_score(y_test, y_test_pred)

    r_squared = r2_score(y_test_pred,y_test)
    accuracy = round(r_squared*100,2)

    result = [str(title),test_mse, test_r2,r_squared,accuracy]

    return result
```

Interpreting the Model results

Model Results with No Feature Reduction:

	Algorithm	Test MSE	Test R2	R Squared	Accuracy
0	LinearRegression()	23.157948	0.826961	0.793633	79.36
1	KNeighborsRegressor()	21.482055	0.839484	0.815963	81.60
2	RandomForestRegressor()	17.442603	0.869667	0.854467	85.45
3	DecisionTreeRegressor(max_depth=3, max_feature...	22.332338	0.833130	0.802207	80.22
4	GradientBoostingRegressor(random_state=42)	17.199274	0.871485	0.851299	85.13

The Gradient Boosting Regressor and Random Forest Regressor outperformed the other algorithms based on the given metrics. They had a lower MSE, higher R2, and better accuracy compared to the other algorithms such as Linear Regression, K Nearest Neighbors Regressor, and Decision Tree Regressor. Among the two, the Gradient Boosting Regressor had a slightly lower MSE and higher R2, indicating slightly better performance overall.

Model Results with Pearson's Method Feature reduction:

	Algorithm	Test MSE	Test R2	R Squared	Accuracy
0	LinearRegression()	23.928997	0.821200	0.785549	78.55
1	KNeighborsRegressor()	22.712246	0.830292	0.811406	81.14
2	RandomForestRegressor()	22.510027	0.831803	0.814045	81.40
3	DecisionTreeRegressor(max_depth=3, max_feature...	22.332338	0.833130	0.802207	80.22
4	GradientBoostingRegressor(random_state=42)	18.987270	0.858125	0.835796	83.58

From the above observation, the Random Forest Regressor and Gradient Boosting Regressor demonstrate superior performance across all metrics compared to the other algorithms, with the Random Forest Regressor having a slight edge in terms of accuracy. These algorithms are well-suited for regression tasks where accurate prediction is crucial.

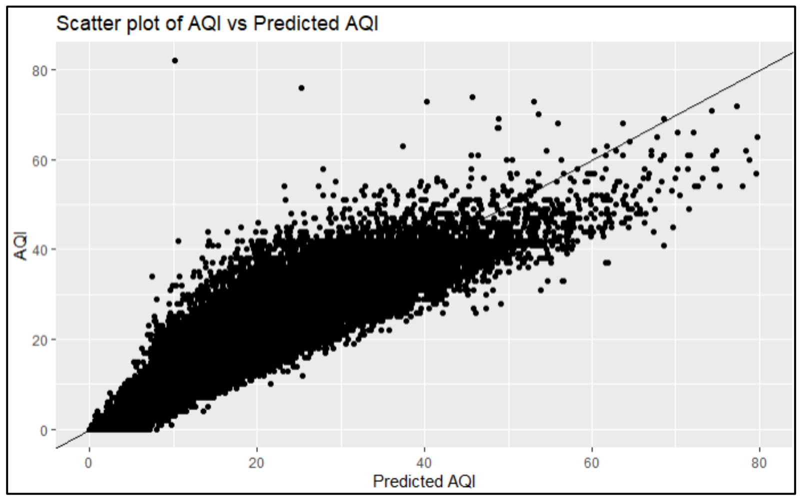
Model Results with PCA Feature Reduction:

	Algorithm	Test MSE	Test R2	R Squared	Accuracy
0	LinearRegression()	67.588826	0.494969	0.245893	24.59
1	KNeighborsRegressor()	27.964259	0.791048	0.750876	75.09
2	RandomForestRegressor()	26.463282	0.802263	0.753679	75.37
3	DecisionTreeRegressor(max_depth=3, max_feature...	61.658604	0.539280	0.147110	14.71
4	GradientBoostingRegressor(random_state=42)	38.324697	0.713634	0.568430	56.84

The K Nearest Neighbors Regressor performed better compared to the other algorithms, having the lowest MSE, highest R2, and higher accuracy. Linear Regression had the lowest accuracy and relatively poor performance. Random Forest Regressor and Decision Tree Regressor had similar performances but were outperformed by the K Nearest Neighbors Regressor and Gradient Boosting Regressor. Experimenting with various feature reduction techniques, it was discovered that the model performance with all the features was like the one with Pearson's feature reduction method. Hence the final model was built including all the features in the dataset thus not losing the impact of the pollutants on the AQI.

Other Observations:

The results of the linear regression model are plotted in the scatter plot (Figure 6) that compares the AQI versus the predicted AQI values. It indicates the predicted values are not too different from the actual values of AQI in the dataset.



(Figure 6)

The combined data set contains the concentration of pollutants and metrological parameters such as temperature, pressure, wind speed, etc. for a location on a given day. Many graphs are plotted on the final dataset as discussed in this section.

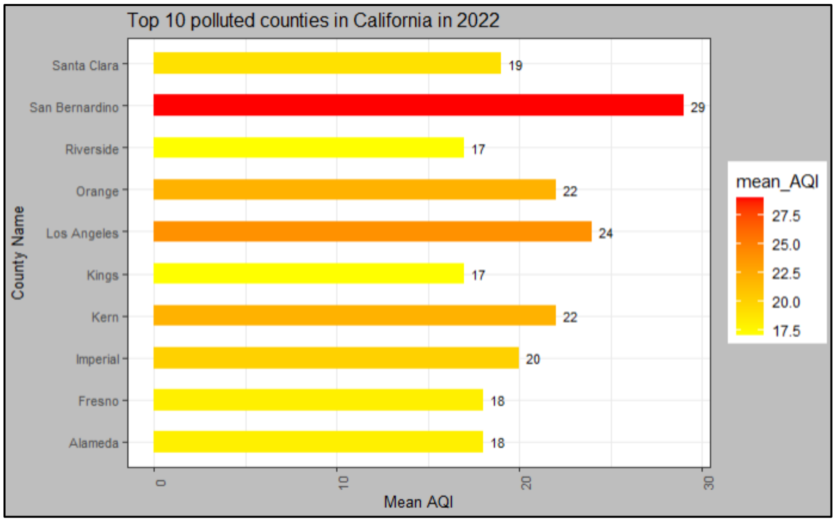
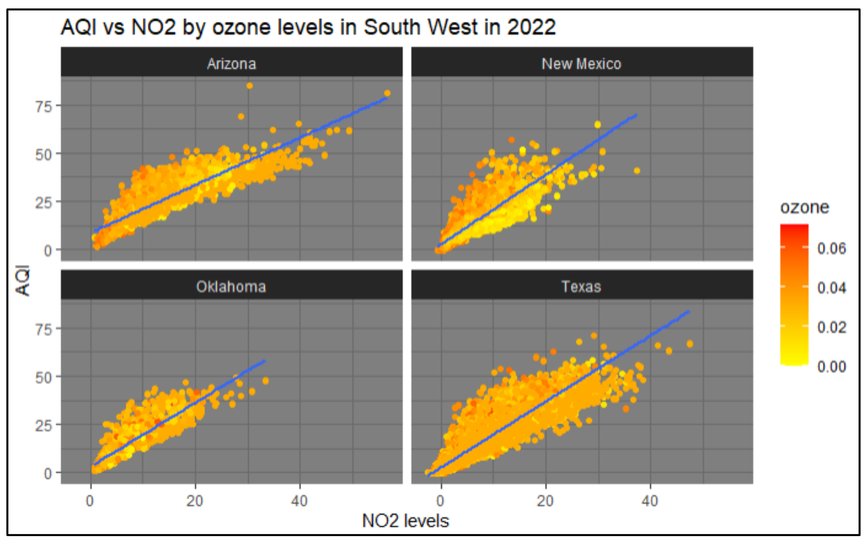


Figure 7 represents a bar plot that contains the most polluted counties in California in 2022 and their average AQI values. San Bernardino was the most polluted followed by Los Angeles counties based on the NO2 concentration levels.

(Figure 7)

Figure 8 represents multiple scatter plots of NO2 concentration versus AQI colored based on ozone values. Each subplot represents a state in the Southwest region. All 4 states represent a similar trend with ozone concentration uniform, though Oklahoma and Texas had higher Ozone levels between 20-40 AQI.



(Figure 8)

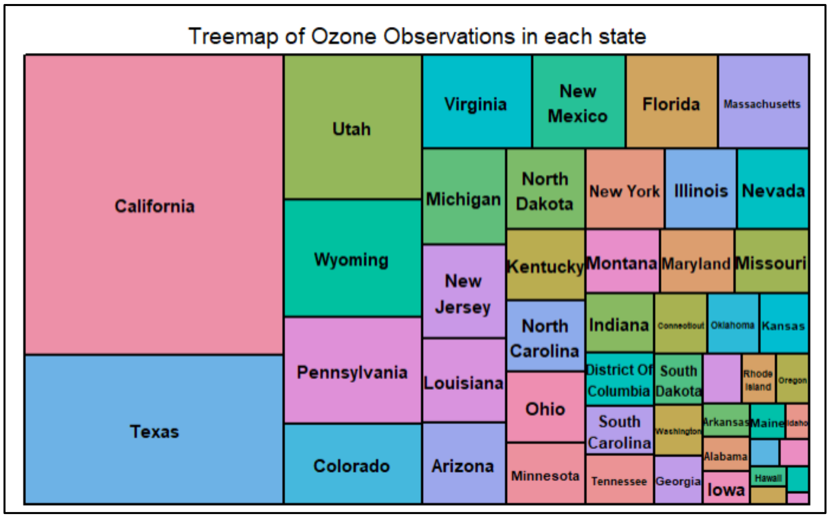
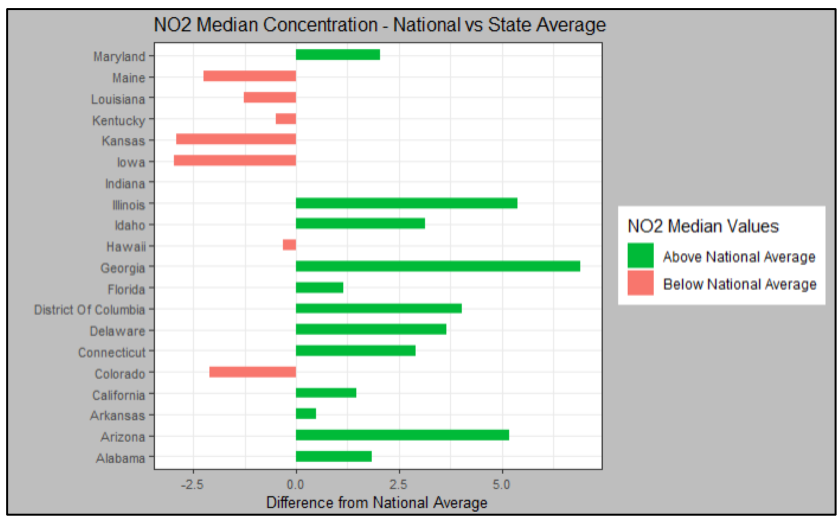


Figure 9 represents a Tree Map of the number of ozone observations in each state. As expected, California has the highest level followed by Texas and Utah.

(Figure 9)

Figure 10 represents the comparison of Median values of the national average of NO2 versus the top 20 states in the US. Surprisingly, the states of Georgia, Arizona, and Illinois are in the top states above the national average, while Kansas, Iowa, Colorado, and Maine are the lowest.



(Figure 10)

Conclusion and Recommendations:

The model results suggest that NO2 and CO are better predictors for AQI compared to other pollutants. Though ozone was not a strong predictor for AQI as per the model, the effects of ozone cannot be undermined as it can have detrimental effects on Air quality, so it was included during the model building.

Among all the states in the US, California had the greatest number of observations in the dataset, of which San Bernardino County was found the most polluted. A similar analysis can be extended to other states and regions to find the most polluted counties and measures can be taken to reduce the impacts on the Air quality in those regions. Furthermore, clustering can be done on the dataset to identify the clusters with the higher AQI.

Ethical Implications:

Though we don't have control over choosing the air that we breathe, several ethical impacts must be considered while analyzing the impacts of air pollution. In many cases, it is hard to identify the source of the origin of air pollution, reasonable measures can be taken to control it while it cannot be avoided or stopped completely. Some of the ethical implications are listed below:

- Though gas-powered vehicle emissions and industrial smoke have played a significant role in air pollution, they cannot be entirely replaced by sustainable solutions, as they can lead to many job losses affecting many families employed by the manufacturing industries. Care must be taken while publishing the results, keeping in mind the impact it can have on families.
- While determining the acceptable levels of greenhouse gases and pollutants for humans, careful assessment should be made while determining the values, as the acceptable levels for humans may cause significant damage to other ecosystems and species. (Brown, 2001)
- The acceptable levels should also be carefully assessed with international considerations in mind, as the gas emissions and pollutants from the developed countries are no longer a local issue. These impacts are already seen on the other side of the world, with extreme floods and drought conditions that were not seen in the past.

References:

Kiger, P. (2021, November 9). Seven negative effects of the Industrial Revolution. History. https://www.history.com/news/industrial-revolution-negative-effects

Air quality. (n.d.). Transport & Environment. https://www.transportenvironment.org/challenges/air-quality/

10 Things You Never Knew Could Cause Lung Cancer. (n.d.). WebMD. https://www.webmd.com/cancer/ss/slideshow-lung-cancer-surprising-causes

“Like breathing poison”: Children in India’s Delhi hit hard by smog. (n.d.). Al Jazeera. Retrieved December 8, 2023, from https://www.aljazeera.com/gallery/2023/11/10/photos-like-breathing-poison-delhi-children-hardest-hit-by-smog#:~:text=Many%20have%20asthma%20or%20pneumonia

Download Files | AirData | US EPA. (2015). Epa.gov. https://aqs.epa.gov/aqsweb/airdata/download_files.html

Air Quality Dataset. (n.d.). Www.kaggle.com. https://www.kaggle.com/datasets/fedesoriano/air-quality-data-set

UCI Machine Learning Repository: Air Quality Data Set. (n.d.). Archive.ics.uci.edu. https://archive.ics.uci.edu/ml/datasets/Air+Quality

Brown, D. (2001). The Ethical Dimensions of Global Environmental Issues. American Academy of Arts & Sciences. https://www.amacad.org/publication/ethical-dimensions-global-environmental-issues

US EPA, O. (2016, August 30). Air Data Basic Information. US EPA. https://www.epa.gov/outdoor-air-quality-data/air-data-basic-information

USA States to region. (n.d.). Www.kaggle.com. Retrieved January 13, 2024, from https://www.kaggle.com/datasets/omer2040/usa-states-to-region