Predicting Air Quality in Sacramento, California

Ksenia Lepikhina, Nihar Nandan Hemantha Kumara, Kristen Okorn

May 6, 2019

1 Introduction

Given the current state of the Earth, with climate change and human activities that are increasingly disruptive, air quality is more important now than ever. The World Health Organization estimates that 4.2 million people [1] die each year from causes directly related to air pollution. Currently, there are a number of cities with large populations in developing countries that have declining air quality. In the near future, poor air quality will begin to effect developed countries as well.

While air quality is a global issue, this paper plans to study the air quality in Sacramento, California. The overall goal of this project is to develop models capable of predicting air quality based off of various environmental and economic factors. Knowing which metrics influence air quality could help communities be aware of what causes unsafe pollution levels.

2 Problem Space

This paper concerns the link between air quality conditions, weather statistics, and economic factors in the Sacramento, California area over the course of five years (2014 - 2018).

Different weather conditions affect how pollutants are spread, and in some cases, which ones are emitted. For instance, sunshine can cause pollutants in the troposphere to undergo photolysis, thus changing their concentrations. Rainy or snowy conditions can wash out particulate matter and certain pollutants and high winds can transport them further away. Thus, by observing weather data for Sacramento, general trends in air quality patterns can be predicted.

While previous work [2] has looked into environmental factors effects on pollutants, we plan on incorporating economic data as well. Some of these factors include the city's population, their percentage of renewable energy, and median income. People generate pollutants by driving and as the population grows, the pollutant output increases and the air quality worsens. The distribution of how energy in California is generated could counteract the air quality degradation because of the actions the state is taking towards 100% clean energy [3]. Income could also be a significant factor because with more expendable income, the more money can be spent on energy to power a house, for example, which may lead to increased pollution.

3 Approach

The pollutants we choose to measure the air quality are ozone and carbon monoxide (CO). Ozone is chosen as it is one of the pollutants that undergoes photolysis after interaction with sunlight and should be correlated with seasonality. Carbon monoxide is selected because it is produced from incomplete combustion activities which might correlate with heating in cold weather or with increased frequency of people driving in inclement weather.

We use regression and classification to study whether economic and environmental data can predict air quality. The goal of the regression is to find an equation that best fits our data. From this equation, we should be able to predict new data given this equation. While regression is typically used to explain the data and not predict, it can be used for prediction as well. Regression works with continuous data (ozone and CO) and with new data we should be able to predict estimated values for ozone and CO.

Given that our data is temporal, we can study how the ozone and carbon monoxide values change over time. While time series analysis is a different class, we can briefly study whether or not our time series exhibits statistical properties that are consistent over time.

Returning to our discussion of machine learning models, when we have categorical target values instead of continuous target values, we can use classification. In order to be able to use this approach, we needed to encode our target values and make them binary. To elaborate, instead of using regression to predict a specific ozone or CO value, we would instead say, for example, that above a certain value, the air quality is poor as measured by ozone and that below that value, the air quality is healthy. The value we choose to split on for CO is 0.5 where if the CO value is greater than 0.5 then we assign a value of 1 otherwise we assign a value of 0. Similarly, for ozone we use 0.035 where values greater than that value are 1 and less than are 0.

The three classification algorithms we choose to use are Support Vector Machines, K-Nearest Neighbors and Random Forests. The regression algorithm we choose is a Gradient Boosting algorithm.

Previous classification work has been done with Support Vector Machines (SVMs) [2]. In the paper by Weizhen, they explore the applicability of SVMs to predict air pollutant concentration. Unlike the research done in the Weizhen paper, we use different weather data and expand the features to include economic factors as well.

Besides SVMs, we investigate how well the K-Nearest Neighbor (KNN) algorithm does to segregate our data into bad air quality and good air quality. While the end goal of each of these classification approaches is the same, the process is quite different. KNN will look at what the value of k neighbors are to determine the class of the data point or entry. Because all of our data is numeric, we can calculate how close the values are more easily and precisely than if we had categorical variables (ordinal or nominal).

While KNN has its benefits (non-linear, detects linear/non-linear data, works well with many data points), it also needs to be carefully tuned. KNN can have performance issues for many data points (not an issue for our data set) and is sensitive to bad features. Support Vector Machines compensate for some of the issues that KNN has. SVM's can be used in linear and non-linear ways depending on the kernel chosen and it works very well with a small number of points and should find the linear separation that exists. Another benefit to using an SVM is that it handles outliers well. It only uses the most relevant points to find the best linear separation because it is uses support vectors.

Another reasonable approach is a Random Forest algorithm. Random forests use multiple Decision Trees and average the results. This is called an ensemble approach. With the quote unquote nice data sets we received in class, we were able to observe that taking random splits of the features and averaging the output from multiple trees produced an excellent accuracy. Random forests are extremely popular for their ability to produce good results with little tuning.

Finally, to use Gradient Boosting regression we return to using the original values for ozone and carbon monoxide. With most regression algorithms, we can not use accuracy as our performance metric. Instead, we can study other values such as R^2 and the mean squared error to assess performance.

4 Data

The data sets that we have selected include data from the California Air Resources Board [4] and the National Weather Service [5] database. Additional data was collected from the California Energy Commission [6] and the City of Sacramento Open Data [7].

For the weather data, selected daily averages for five years (2014-2018) were utilized; only one of which was a leap year. In total, the data has 1,826 entries per feature. The features we included are maximum temperature, minimum temperature, average temperature, departure (difference between the average temperature and the 30 year normal temperature for the date [8]), HDD/CDD ("gauge of the amount of heating or cooling needed for a building using 65 degrees as a baseline" [8]), precipitation, new snow, and snow depth which are daily values.

The additional features we found were electricity/gas usage for residential and non residential, median income, population, and the percentage of each energy source which are yearly.

The final shape of our feature data is 1,826 rows and 26 columns.

The two target values for air quality that we chose to study are ozone and carbon monoxide. While data on a wide variety of different pollutants were available, those thought to be correlated most strongly to weather data were chosen. Our reasons for selecting ozone and carbon monoxide can be seen in the Approach section.

An example of one row in our data can be seen below:

Date	MaxTemp	MinTemp	AvgTemp	Departure	HDD	CDD	precipitation	NewSnow
2014-01-01	65	35	50.0	3.7	15	0	0.0	0.0

SnowDepth	Ozone	CO	Electricity(GWh)/Non-Residential	Electricity(GWh)/Residential
na	0.026	1.0	6258.537182	4736.128873

Gas(Therms)/Non-Residential	Gas(Therms)/Residential	MedianIncome(dollars)	Population
100.810185	172.911259	62203.0	485199.0

Wind	Solar	Small hydro	Geothermal	Biomass	Large hydro	Coal	Nuclear	Natural Gas	Unspecified
0.081	0.042	0.009	0.044	0.025	0.055	0.064	0.085	0.445	0.15

5 Results

As described in the Approach section, we implemented four models (Random Forest, K-Nearest Neighbor, Support Vector Machine, and Gradient Boosting Regression) for both ozone and carbon monoxide in order to predict air quality in Sacramento. The classification algorithms help us determine whether or not the air was healthy to breath or not. While our data was not binary originally, we split the data at a constant value of both CO and of ozone and created binary data based on those values. We then also implemented a regression algorithm to find a model that fit the data best as measured by the R^2 value.

While both classification and regression can be used for prediction, classification allows us to categorize the health of the air based on various environmental and economical factors. Regression on the other hand is best for understanding the data at hand and finding a model that best explains it.

Prior to beginning classification, some housekeeping was performed on the data for each of the four methods used. This included merging two columns in the data set, 'departure' and 'temperature departure'. These represent the same feature; the name was simply changed over the four year span of data that was utilized.

In addition, some weather features would occasionally have a letter (M or T) assigned to them based on meteorological conditions, which were replaced with zeros in order to make the variables continuous. This same strategy was used for NaN values on days where data was missing. Another correction method was attempted in which all NaN values in a column are replaced with the average value for that feature using a Numpy implementation. However, this method did not prove to be any more effective than replacement with zero in terms of accuracy, so the previous method is used. Regardless of the value the NaNs are replaced with the model performs more poorly than if we had all of the data. For example, for the temperature and other environmental-related factors, replacing a NaN value that occurred in the summer months with an average value (more suited to spring or fall) or a zero tended to worsen the results. By having some features on a certain date express weather extremes while others presented milder conditions or none at all, prediction accuracy is not as high as if 100 percent complete data was available. However, due to neither method attempted proving to be significantly better suited than the other, the simpler method is ultimately chosen.

Features are dropped one by one to see if any did not correlate with the ozone and carbon monoxide levels being predicted. Note, confounding is an issue that will be dealt with in future iterations of this work (see Discussion). The date column is the only one thrown away; as air quality should not be effected by the date. In future work, we plan on creating seasons as features.

Each method also utilizes SciKit Learn's train test split algorithm in order to divide the data into training and testing data so that the accuracy of both can be calculated and used for tuning. Similarly, SKLearn's accuracy score function is also used for its simplicity of calculation.

The first classification model we work on was the K-Nearest Neighbor algorithm. Scaling and normalization are attempted on this data set; however, these prove to worsen the accuracy on both testing and training. Thus, the parameters of the SciKit Learn KNeighborsClassifier are tuned until the closest agreement between training and testing was achieved. While

100 percent accuracy could be achieved for the training data, this model fit the testing data very poorly, suggesting that overfitting is occurring. Thus, a less accurate model that was consistent for both data sets is selected. The tuning parameters for ozone and carbon monoxide vary slightly, although having 6 nearest neighbors for each produce the best results.

Next, we looked at the Support Vector Machine algorithm. SVM performed poorly compared to the other classifications used. As mentioned before, we use binary data for the classification problems, including SVM. The SVM was the most computationally expensive among the models used. The accuracies for both ozone and CO was between 70-80%.

The final classification algorithm we use is a Random Forest. Overall, we expect this algorithm to perform best out of all of our classification algorithms. The random forest performs well for both ozone and carbon monoxide. The test accuracy was about 83% for ozone and about 88% for carbon monoxide. The parameters chosen for the SKlearn package were tuned using grid search cross validation. We choose to use 100 trees for each random forest. More trees do not improve our accuracy.

The overall results of the three classification methods can be seen in Figures **??** and **??**.

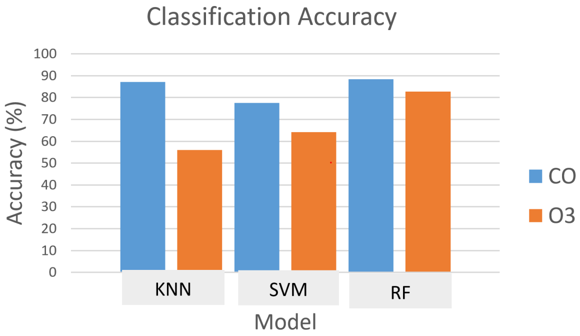


Figure 1: Classification Accuracy

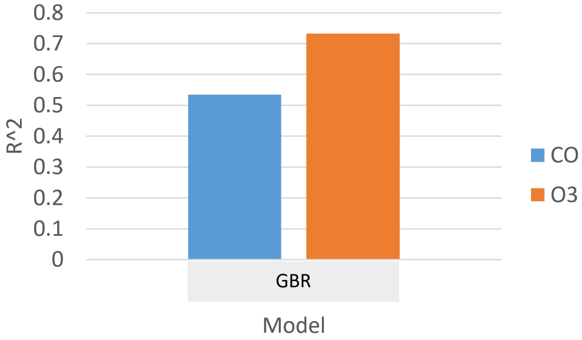
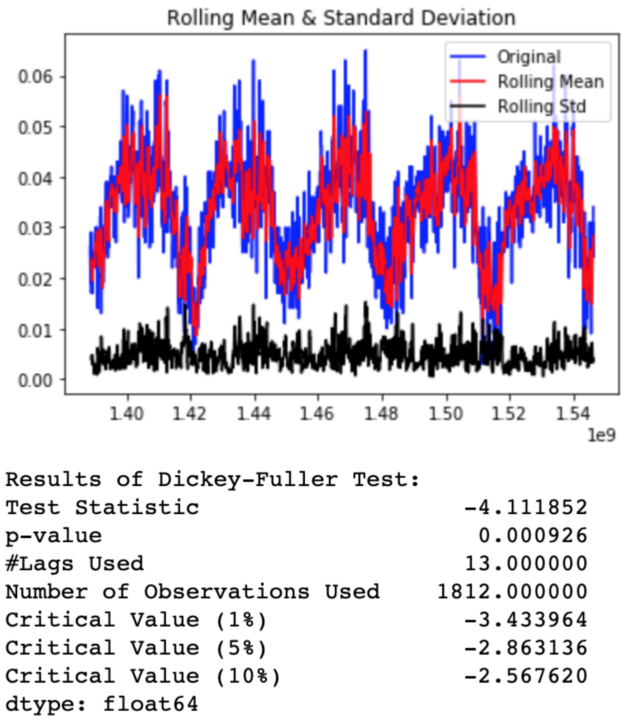


Figure 2: Regression coefficient of determination

As for regression, we look at the Gradient Boosting algorithm. Initially, we try using a basic multiple linear regression and then a logistic regression, but our accuracies are abysmal. We ended up choosing the gradient boosting algorithm because it is another ensemble method unlike the basic regression and logistic regression. This algorithm builds a model in a forward stagewise fashion. The loss faction that is chosen is the least squares loss function. We use a learning rate of 0.2 for ozone and 0.9 for CO. Lower learning rates produce worse results for CO. The learning rate parameter decreases the contribution of each individual regressor by the learning rate value. The ozone model uses approximately 275 estimators and CO uses 200. Increasing the number of estimators beyond those values did not improve the R^2 or MSE value.

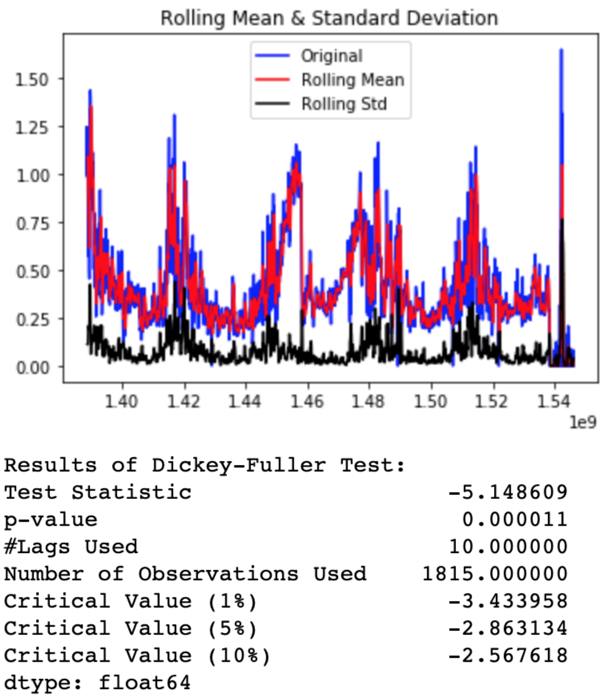
The Gradient Boosting algorithm is able to create a decent model for ozone but not for CO. The R^2 value for ozone is 0.73 for the test set and 0.53 for CO for the test set.

In addition to the various machine learning approaches we use for this problem, we also look at the time series of our target values as seen in Figure 3 and 4.



```
Results of Dickey-Fuller Test:
Test Statistic                  -4.111852
p-value                          0.000926
#Lags Used                      13.000000
Number of Observations Used   1812.000000
Critical Value (1%)             -3.433964
Critical Value (5%)             -2.863136
Critical Value (10%)            -2.567620
dtype: float64
```

Figure 3: Ozone TS



```
Results of Dickey-Fuller Test:
Test Statistic                  -5.148609
p-value                          0.000011
#Lags Used                      10.000000
Number of Observations Used   1815.000000
Critical Value (1%)             -3.433958
Critical Value (5%)             -2.863134
Critical Value (10%)            -2.567618
dtype: float64
```

Figure 4: CO TS

We use the Dickey-Fuller test to determine whether or not the time series are stationary. In the context of time series, stationarity implies that all statistical properties, such as mean, variance, autocorrelation, etc. are all constant over time. While seasonal data, such as ours, is not technically stationary, we mean that it is stationary in the same months year to year. This means we can expect the same behavior from our carbon monoxide and oxygen in the same months with respect to their statistical properties. This is called a cyclostationary model [9].

6 Discussion

The overarching goal of this project was to research how well certain predictors would be able to predict air quality. Ozone and carbon monoxide were chosen as they are well known and represent the measure of air quality well. We found that for the random forest machine learning algorithm was the best approach for the classification problem (is the air good or bad) and that gradient boosting was the best approach for the regression problem with results seen in the previous section.

If air quality is to be predicted further in advance than current weather forecasting provides, at-risk groups such as children, the elderly, pregnant women, and anyone with pre-existing respiratory conditions could avoid exposure entirely rather than waiting for current air quality readings to turn bad. This might entail staying indoors, temporarily relocating to a cleaner region, or wearing protective masks to prevent them from becoming ill. Overall, better predictions allow for prior warning and thus more time to act before air quality poses a risk.

Given the ranges for ozone and carbon monoxide, classification performs better for carbon monoxide and regression with ozone. Because ozone has a smaller range, it is challenging for the classification algorithms to segregate the values. However, regression is able to work with finer values. Carbon monoxide has a larger range and thus, we are able to separate the data with a single value for the binary classification better

As for the three classification methods, the random forest had the highest training and testing accuracies, as predicted. A random forest is composed of some sort of aggregation of decision trees. Like K-Nearest Neighbor, decision trees are capable of nonlinear separation. Random forests will typically out perform basic classification because they are considered an ensemble methods. The issue with ensemble methods is that they take longer, are more difficult to interpret, and do not necessarily perform better.

The KNN fits for ozone are consistent for both training and testing, with accuracies in the mid 80's. However, the fits for carbon monoxide are much more inconsistent - while the training data has a similar accuracy in the mid 80's, the best possible fit found for the test data is just below 60 percent, despite attempts to tune the train-test split and KNN parameters to yield the most agreement between training and testing. It appears that overfitting occured, although typical solutions to overfitting did not prove to be particularly useful in this case.

For the SVM, correlation typically negatively impacts our results if we use a linear kernel. However, with a kernel that is able to work with higher dimensions, the correlation should not affect the results. SVMs tend to fail when the data is messy and have multiple overlapping points. Because the data is cyclical (seasonally), then we do end up with numerous overlapping points which leads to less accurate predictions.

As for the gradient boosting regression, the MSE for ozone is approximately 0 which, while sounds good, likely indicates that we have high biased data as an MSE that low does not happen unless we are over fitting our data. However, this is the MSE for the test set so it likely indicates high bias. The MSE for CO is more reasonable at 0.02.

In future works, categorical features could be added in to distinguish what other human or environmental factors air quality relates to. For instance, rather than throwing away the date column, it could be split up into months or seasons to demonstrate how trends vary throughout the course of a year. This would likely correlate to both weather and anthropogenic features. For instance, carbon monoxide levels would likely be different in the winter months as opposed to the summer, not only because of the temperature and snowfall, but also due to the number of heating systems and vehicles being utilized.

In that same vein, the correlation between features could be addressed more fully in order to improve prediction accuracy and reduce bias. For instance, all of the temperature terms as well as heating degree days (HDD) and cooling degree days (CDD) were found to be correlated with time, as shown below in Figure 5. This issue was partially resolved by dropping the date column. However, in order to be more thorough in the future, it is important to test for correlation between the other features. This was a limitation in our dataset - although we had many features, having five of those be so similar did not provide us with as much information as we had originally hoped. A more varied dataset would be a huge improvement in future works.

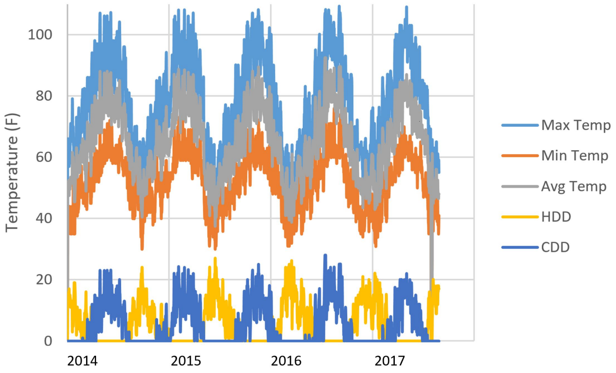


Figure 5: Timeseries for temperature-related features

While it would be simple to plot the features against each other with a pairs plot, we would likely see correlation but that is simply because time is a well known confounder [10]. This means that while the features may seem correlated, their is actually a factor that correlates with both features and causes a false correlation to appear. Intuitively, we can say that all of our features that relate to temperature likely exhibit either a positive or negative correlation. For example, given a certain average temperature, we can intuit that the minimum and maximum temperatures are within a certain range. Additionally, heating degree day (HDD) and cooling degree day (CDD) may be correlated. We can not make definitive statements on the correlation between the features until we take time into account and properly do the analysis. Given these statements, we must be careful about seasonality being a confounder. In future iterations of this work, we plan on studying timeseries more and accounting for time as a confounder in our results.

7 Code Link

Github: `https://github.com/niharnandan/ml-project`

References

[1] *Ambient air pollution: Health impacts.* URL: `https://www.who.int/airpollution/ambient/health-impacts/en/`.

[2] Weizhen et. al. Lu. “Air pollutant parameter forecasting using support vector machines”. In: *Proceedings of the 2002 International Joint Conference on Neural Networks* (2002).

[3] *What to Know About California’s Commitment to 100 Percent Clean Energy by 2045.* URL: `https://www.smithsonianmag.com/smart-news/california-commits-100-percent-clean-energy-2045-180970262/`.

[4] *California Air Resources Board.* URL: `https://ww2.arb.ca.gov/homepage`.

[5] *NOWData - NOAA Online Weather Data.* URL: `https://w2.weather.gov/climate/xmacis.php?wfo=sto`.

[6] *Total System Power*. URL: https://www.energy.ca.gov/almanac/electricity_data/total_system_power.html.

[7] *Sacramento Open Data*. URL: http://data.cityofsacramento.org/.

[8] *Understanding the Preliminary Monthly Climate Data (WS Form F-6)*. URL: https://w2.weather.gov/climate/f6.php.

[9] *Cyclostationary models*. URL: https://en.wikipedia.org/wiki/Cyclostationary_process#Cyclostationary_models.

[10] *Confounders in Time-Series Regression*. URL: https://www.mailman.columbia.edu/research/population-health-methods/confounders-time-series-regression.