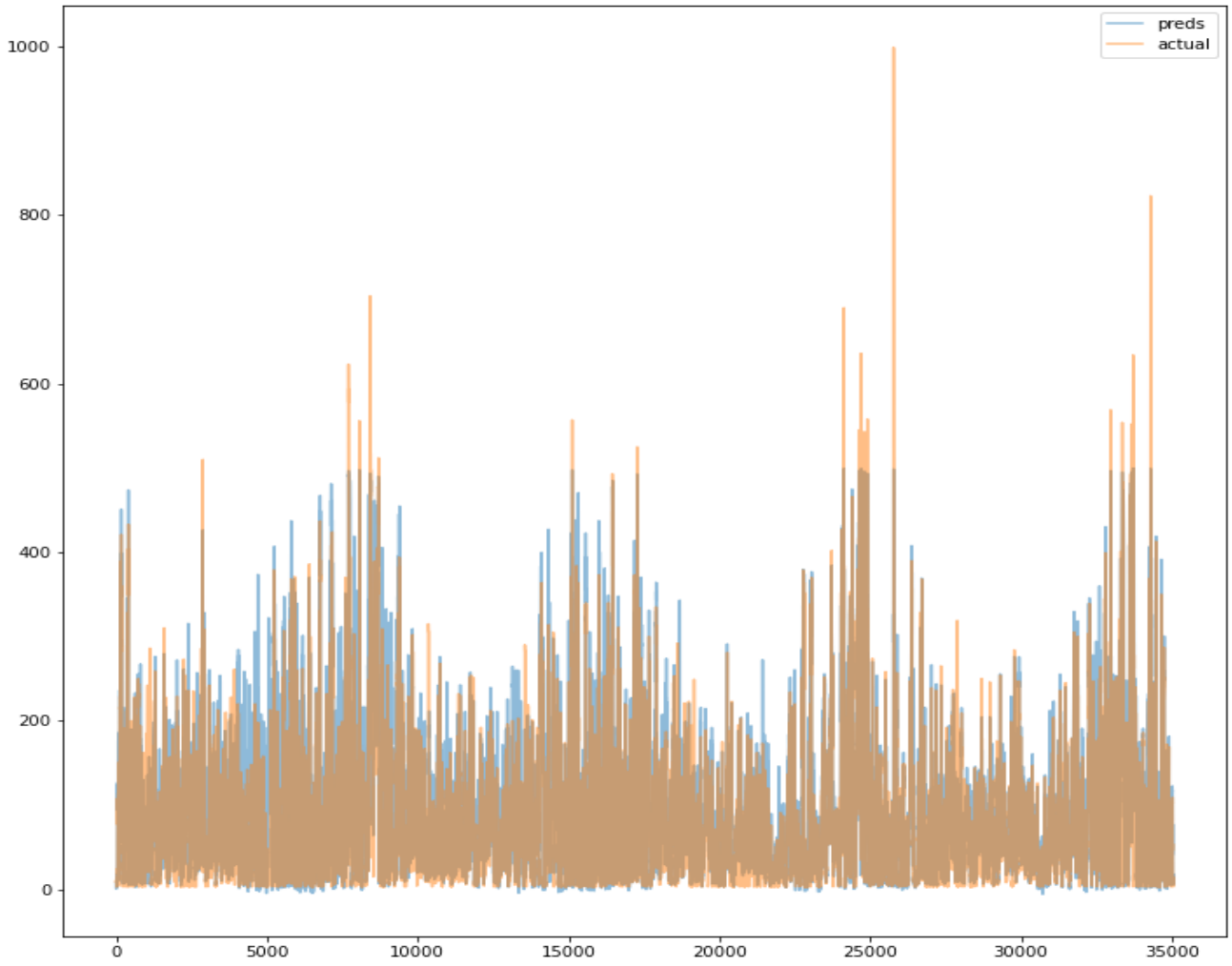
Alec's Power Corp. Power Plant Investment Analysis

Forecasting Air Pollution Levels Over Time To Predict Environmental Costs

Code: https://github.com/Alec-Schneider/BejingPollutionAnalysis



A Neural Network Forecast Vs Actual Data

Table of Contents

Introduction

Background

Alec's Power Corp. will be deciding upon a new location to build a new gas-burning power plant in China in order to meet power demands in the country. This new power plant will add to their excellent portfolio of power plants across China. However, the government is passing new legislation that legally requires new industrial sites to adhere to new air quality standards that come with severe punishments if violated. Before deciding on a final site, management must be able to confidently forecast whether emissions at their new plant will exceed the particle matter 2.5 (PM2.5) thresholds, as fines will cut into the company's profitability and investment goals. Management has projected that the new power plant will generate $45M in yearly revenues, which will be used as an assumption in the cost projection.

The legislation states that firms will be punished based on the concentration of particle matter of size 2.5 micrometers (PM2.5) thresholds for different time periods, each with different penalties. PM2.5 are very fine and inhalable particles that have been found to be dangerous to human beings, especially those who are more vulnerable such as kids and the elderly. According to the USA's EPA, sources of the particle matter can be "emitted directly from a source, such as construction sites, unpaved roads, fields, smokestacks or fires" and "form in the atmosphere as a result of complex reactions of chemicals such as sulfur dioxide and nitrogen oxides, which are pollutants emitted from power plants, industries and automobiles." The proposed thresholds and penalties:

Time Period	PM2.5 Threshold (ug/m^3)	Penalty
1 Hour	1000	$100K fine for each violation
3 Hours	750	$500K fine for each violation
24 Hours	450	$1M file for each violation
365 Days	200	45% of time period's revenues

We will use the emissions measurement data from all other power plants that the company owns to build a model that can forecast the PM2.5 in the air based on hourly emissions and weather data from March 2013 to February 2017. This model will then be used to predict the PM2.5 of three power plants that have similar characteristics as Alec's Power Corp's, and are located in the cities where the company has been approved to build a power plant, Tiantan, Wanliu, and Wanshouxigong.

Business Questions

1. Which new location would have exceeded the new law's emissions standards, based on the predictions of the model?
    a. Which emission thresholds were broken and how many times?
2. How accurate are the model's predictions?
3. Which site should be decided upon based on the analysis?

Data Preprocessing

Data Cleaning

Before a model can be built to forecast hourly PM2.5 levels we need to ensure that our data has no missing values or text data. This is because these types of data cause errors in the algorithms that are used to model relationships in the dataset. A timestamp will also be created out of the Year, Month, Day, and Hour variables so our data files will understand how yearly, quarterly, etc. patterns interact with the other measurements. Data cleaning methods applied:

1. For all numeric variables in our data, we will apply a Kalman Filtering method to impute any missing values in our dataset.
    a. At a high level, Kalman Filters is state-space model that uses the estimate of the current state and the current observation of the data to estimate what will happen at the next step in time. The drawback of this method is it does not perform well when there are a large number of consecutive missing values in the dataset. However, since the Kalman Filter is one of the few imputation methods that understands states in time series, it will be the preferred method for replacing missing numeric data.
2. For all categorical variables in our data, we will one hot encode them to create numeric columns that can capture the categorical data.
    a. One hot encoding is a method that converts a single categorical variable into n variables that take the value of either 0 or 1 to represent whether the categorical value is present in the row, where n is the number of unique values in the categorical variable.

Feature Engineering

After cleaning up our data file, we are now able to extract additional information from the existing variables to assist our models in learning the relationships in the data.

1. First, the numeric variables are standardized to center all values around 0 with outliers representing values less than -3 and greater than 3. Standardizing the input and output variables will assist our neural network to converge to a solution faster, as large numerical values may introduce performance issues within the algorithm. The StandardScaler method used to implement this scaling can best be summarized by the creator, scikit-learn:

> *Standardize features by removing the mean and scaling to unit variance. The standard score of a sample x is calculated as:*
>
> *z = (x - u) / s*
>
> *where u is the mean of the training samples or zero if with_mean=False, and s is the standard deviation of the training samples or one if with_std=False.*

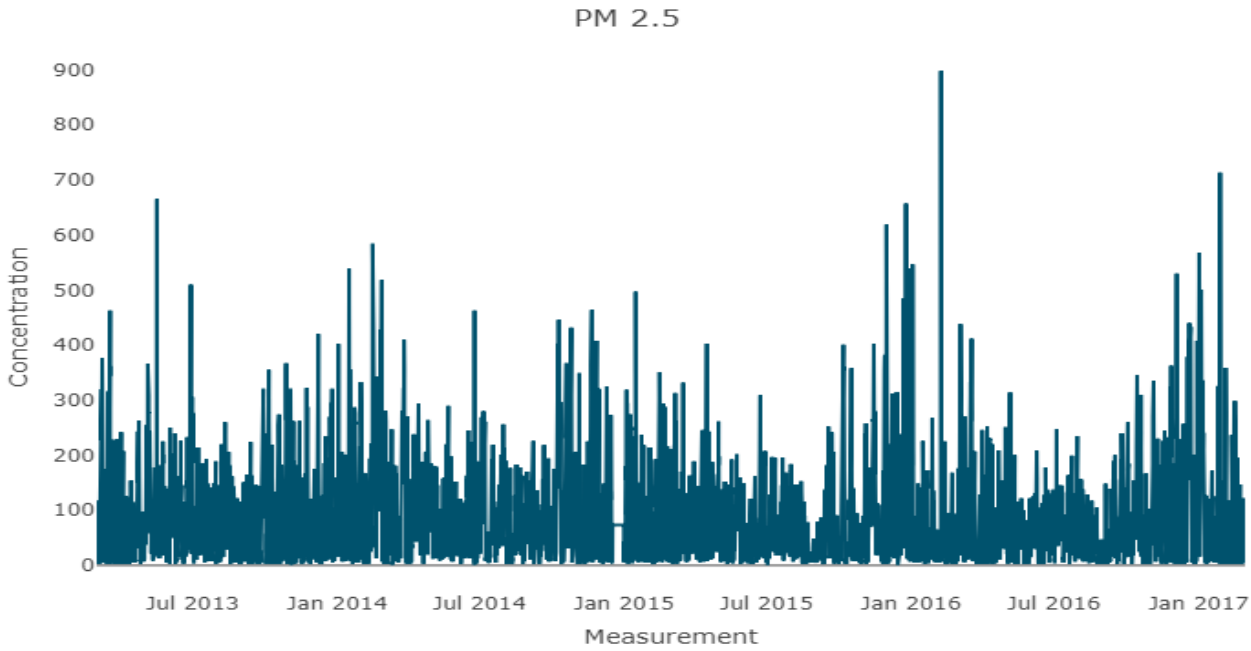
2. Second, we will create two columns that will give us a sense of wind direction and wind velocity. This will allow the model to pick up signals in the wind data. The steps to achieving this are:
    a. First, convert the wind direction column to angle degrees
    b. Second, convert the wind's degrees to radians
    c. Third, multiply the wind speed by the cosine of wind direction in radians
    d. Fourth, multiply the wind speed by the sine of wind direction in radians
3. Third, we will use the timestamp to create signals that represent the hour, day, and year
    a. First, convert the timestamp to the number of minutes that it represents
    b. Second, take the cosine of the minutes times 2 (pi / frac of year)
    c. Third, take the sine of the minutes times 2 (pi / frac of year)

Now the data is ready to be analyzed and used to train models to forecast PM2.5 pollution levels on power plant sites.

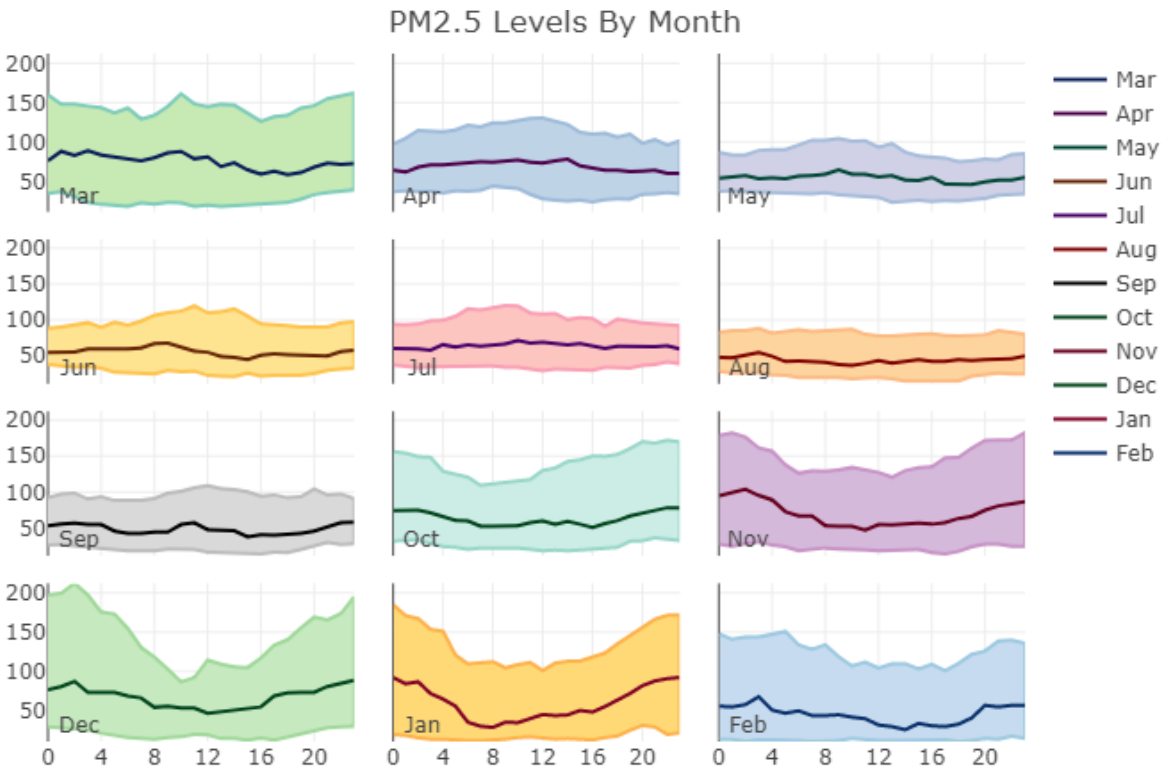
Exploratory Data Analysis

Now there may be variables in the data that are not important to the feature that we would like to predict (PM2.5), and may need to be excluded from the dataset. There may also be features that are very strongly correlated to each other, and variables like this confuse model. We will explore if any of these exist and need to be dropped from the data, but we would also like to explore our data so we can understand our baseline of the power plants we own. This is an extremely important step before building a model because this stage will allow us to optimize the model, interpret the results, and understand whether the model will be sufficient for decision making.

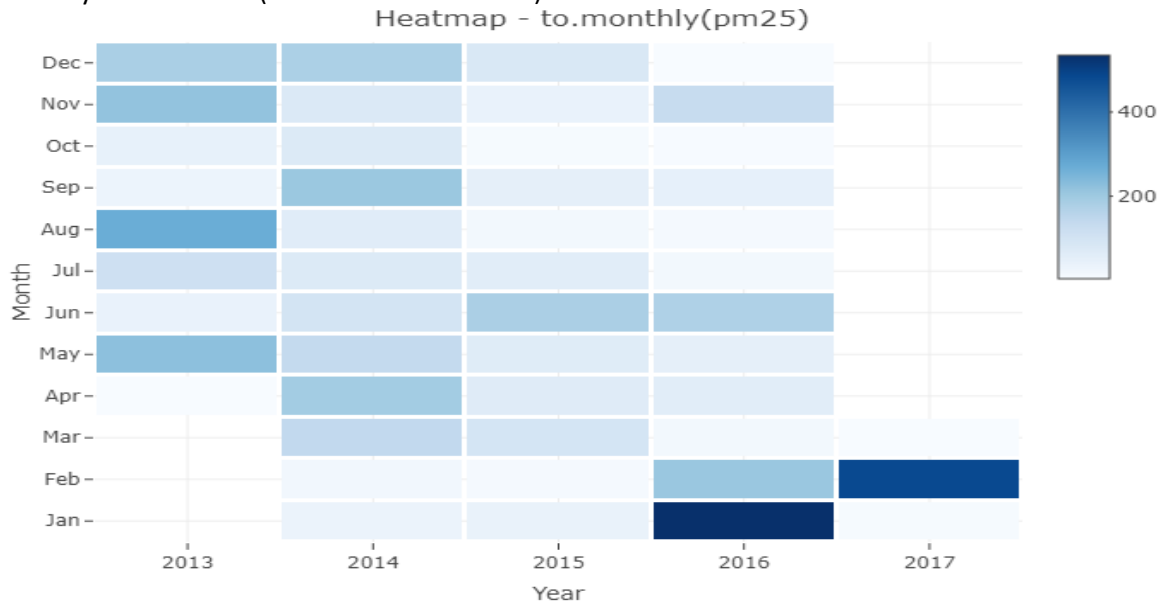
Exploratory analysis is best done with visualizations as patterns become easier to recognize. Let's look at PM2.5 levels over time:



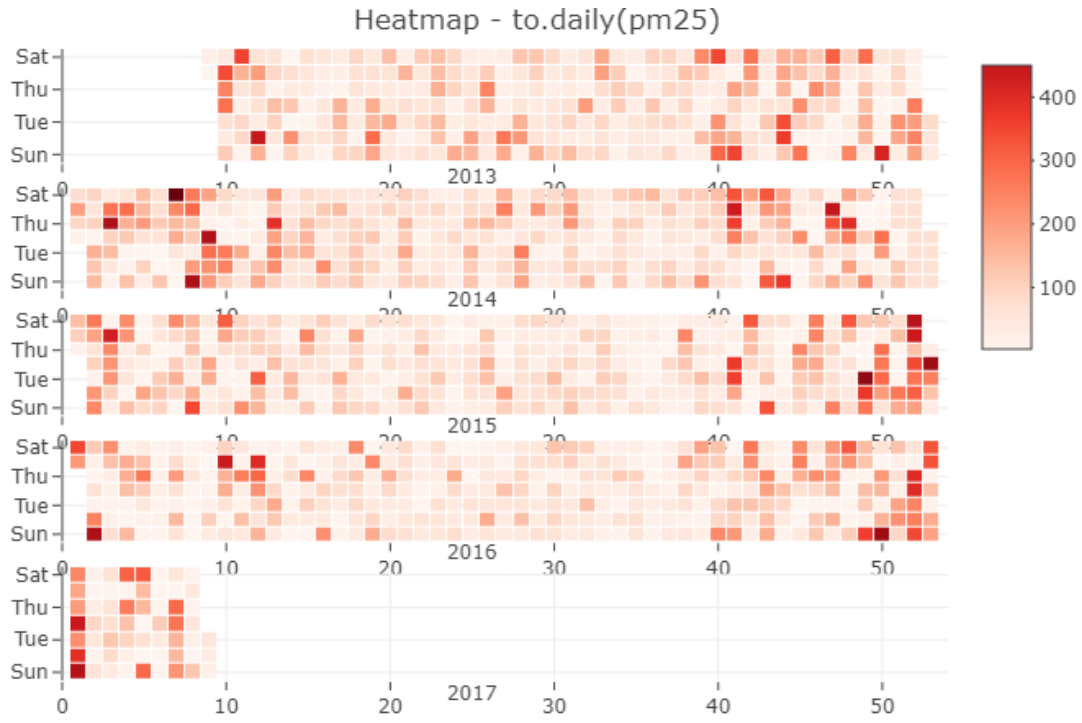
It is observed that PM2.5 levels oscillate around 100 (ug/m^3) with some large spikes and seasonality. Let us see if there are months with higher observed levels. months April through September have much lower measure PM2.5 when compared to October through March. The variation is also much higher in the October through March period, as seen by the colored areas surrounding the line for each month in the first chart below. This confirms that there is seasonality embedded in this data. This seasonality should be picked up by our multi-step forecasting model since we derived time signals during the Feature Engineering section of the analysis.



Monthly PM2.5 levels (Mar. 2013 – Feb. 2017)

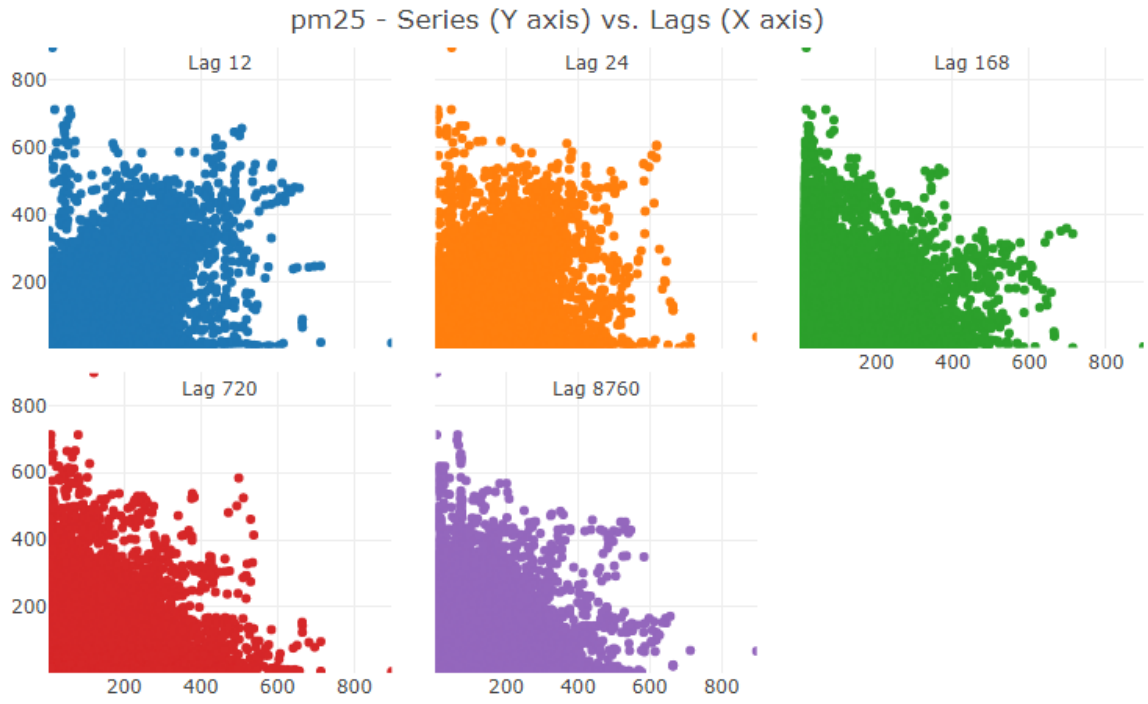


Daily PM2.5 levels (Mar. 2013 – Feb. 2017). Darker reds towards the beginning and end of years.

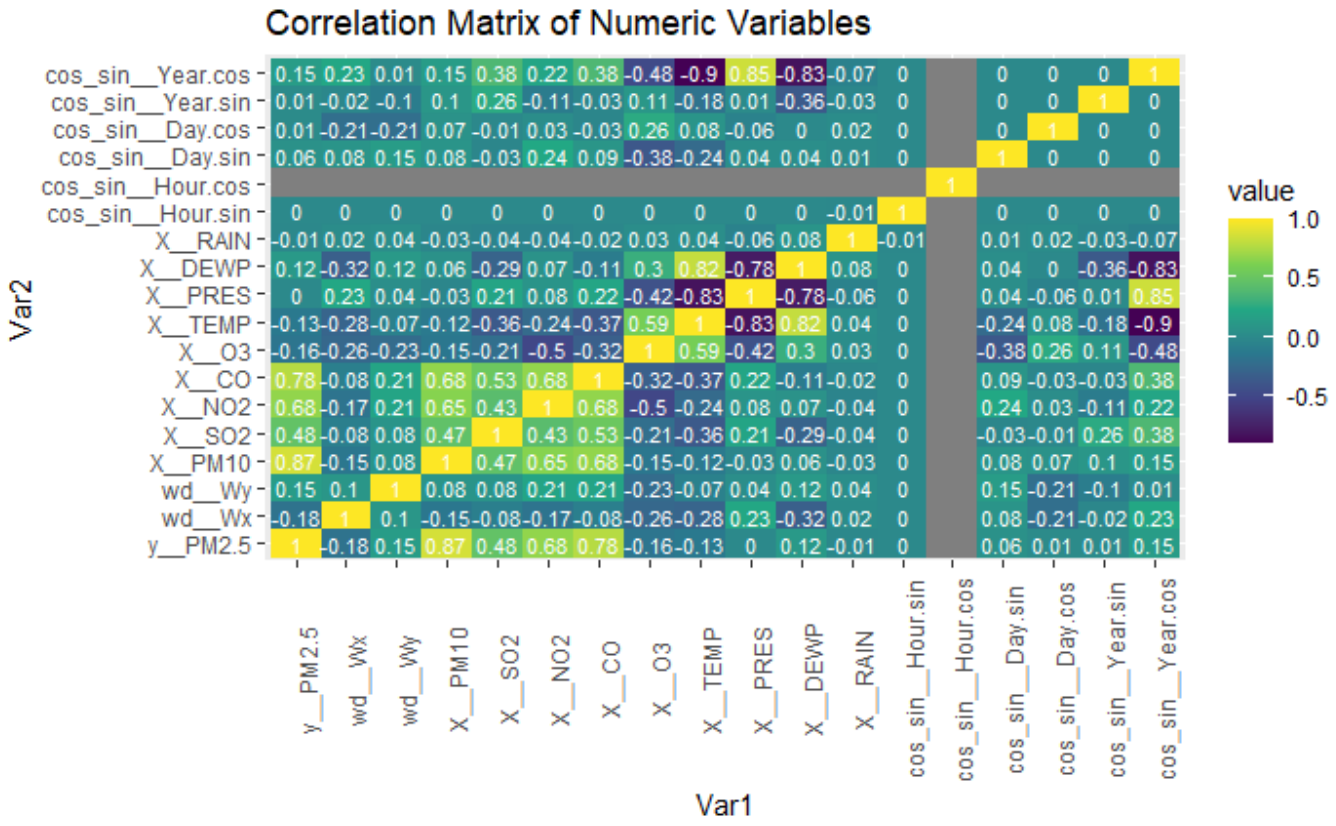


Correlation is a major part of predictive modeling, so we should understand how correlated PM2.5 is related to itself, and different variables in our data (wind direction, CO2, etc.,). The first plot below displays PM2.5 measurements at point x and x - n where n is the number of observations prior to x. This chart will show a straight diagonal line if there is perfect correlation, and more dispersion if they are the points are highly correlated. This plot is showing that PM2.5 values are correlated most with the values from 12 hours before the measure was observed, and become less correlated when we look back further in history.

Current PM2.5 values plotted against the value n lags ago:



The second view to measure correlation is to measure the correlation of a variable against all other variables. These correlation measures range from -1 to 1, where values closer to -1 or 1 indicate a strong linear relationship and a value near 0 suggest weak correlation. When using our target variable in the correlation calculation, we get a sense of what variables may have the largest impact on the target. If there are any measurement variables that have correlation values near 1 one with each other, than one of the variables should be dropped to improve model performance. Based on the correlation matrix below, X_PM10 has a high correlation with PM2.5, as PM10 is the same particles, but bigger. It appears that the weather information (RAIN, DEWP, TEMP, PRES) have some high correlations with each other, but not high enough for us to remove the data.



Now that we've understood our data, we are able to begin building a forecasting model.

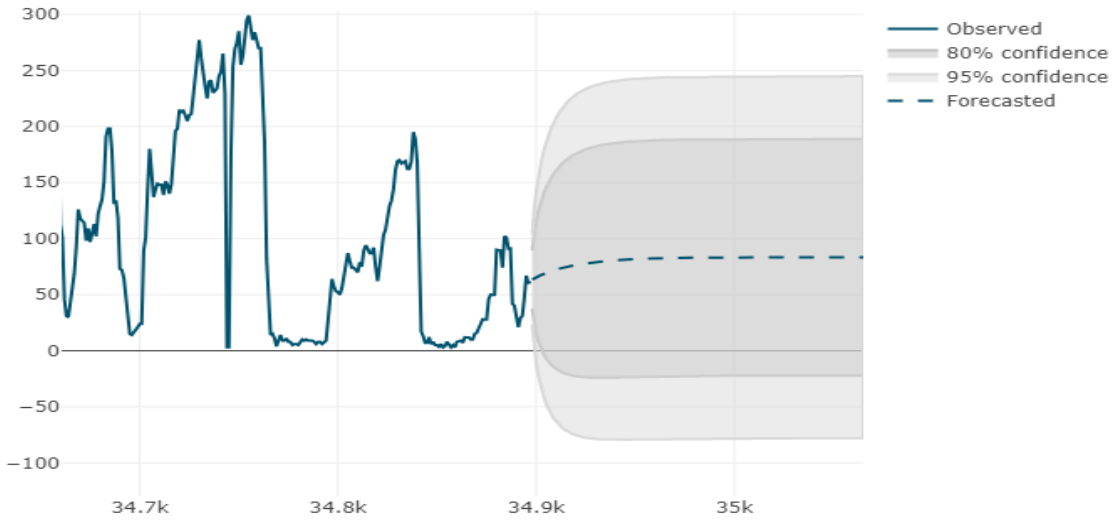
Modeling

Single-Step Forecasting – Auto Regressive Integrated Moving Average & Exponential Smoothing

Auto Regressive Integrated Moving Averages (ARIMAs) are linear regression models that use its own lag values as a variable to predict in the next period. They work great with time series data, as the prior period's value affects the next observed value in the data set. In the correlation discussion there was stronger correlation when using a small number of periods to look back in time. The ARIMA models built will attempt to capture this correlation when we set the time lag to 1.

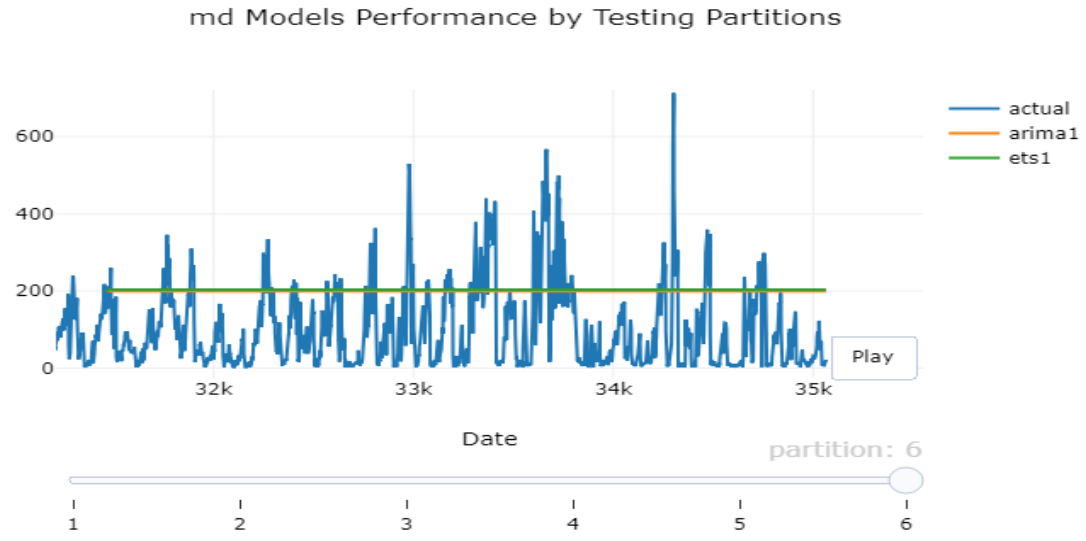
Exponential Smoothing (ETS) models are special cases of ARIMA models because they are non-linear exponential. ETS models however also uses a weighted sum of past observations, where the weights are higher for more recent observations. This is where ETS models get the "smoothing" name from, as the weighted average smooths out the predictions. Both models we discussed will be tested on the data

For our first mode we will build a simple baseline ARIMA model with no tweaks. This graph shows our baseline results, which are very poor and have wide variation, despite only forecasting a mean value. We will have to split the data into training and test set for the models to learn the patterns in PM2.5 pollution over time.



After splitting our pollution data into training and testing data for our models, we implemented multiple iterations of ARIMAs, and ETSs, and the best results were used to forecast on the test data. With ARIMA and ETS, we again get very poor predictions that only seem to capture the mean of the hourly PM2.5 levels. Due to extremely root mean squared error, which is larger than the average PM2.5 observation, we cannot reliably use these models to forecast pollution values for investment decisions. Next, we will use a neural network to try and learn the patterns of PM2.5, but this time using the other features we discussed in our dataset and predict multiple periods into the future.

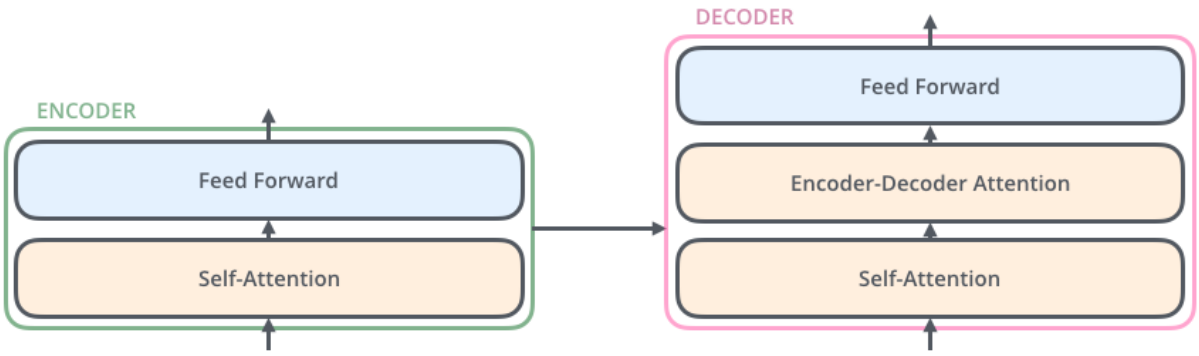
Model	Average Root Mean Squared Error
ARIMA	125.22
ETS	126.33



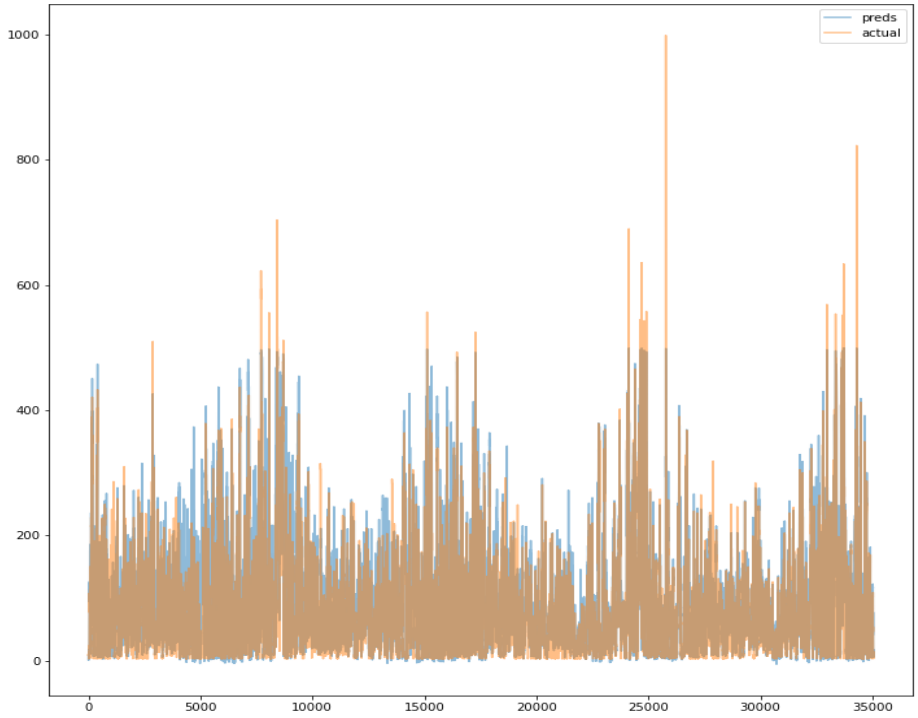
Multi-Step Forecasting – Transformer Neural Network

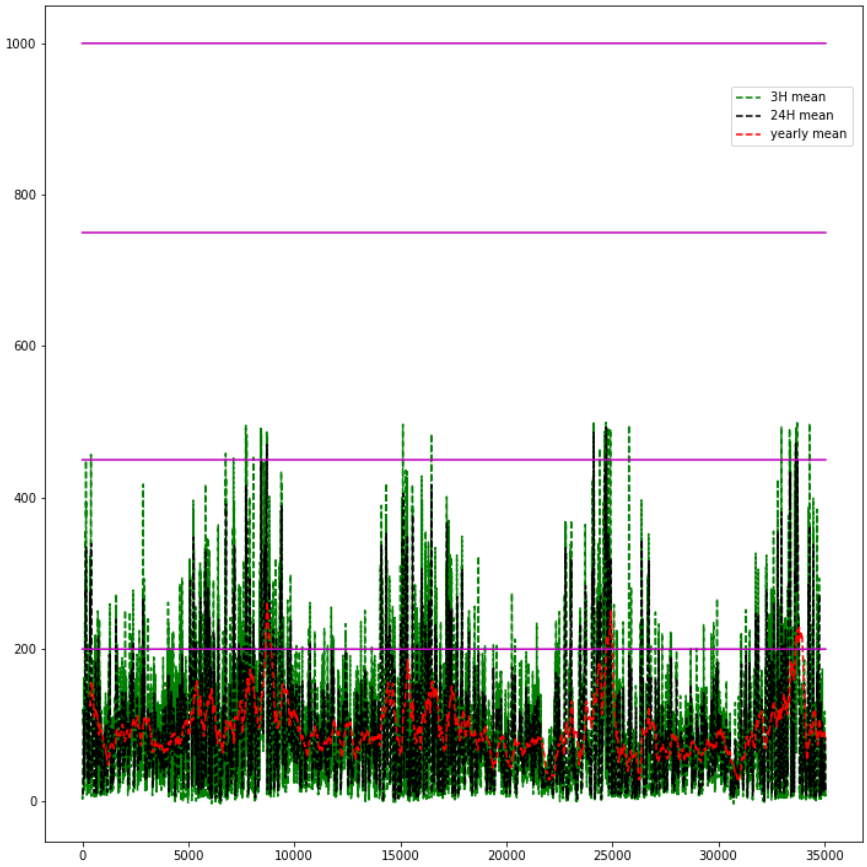
Since the ARIMA and ETS models had very poor performance when just using our target variable, PM2.5, to forecast the future, we will attempt to use neural network to find deep hidden patterns in the enitre dataset. Neural Networks attempt to act similarly to how a human brain works, by creating a large number of nodes that signal to other nodes that there is important information, and use the information to pass information to other nodes, in order to predict an outcome such as a identifying if an object is a dog or a human. Transformer Neural Networks are a new type of model that were just introduced in the 2017 research paper *Attention is All You Need* but have proven to be a game changer for sequence forecasting for time series forecasting. This is because Transformer models encode the current values positions and prior positions for each position in the series, and then feed the encodings to a neural network which feeds into a decoder. The decoder of the Transformer model then extracts important parts of the input sequence of data, to make predictions for different periods.

Encoder and Decoder Blocks of Transformer - Jay Alammar:



After fitting the model on the same training splits used with the ARIMA and ETS models, we can now predict the entire time series. Across all three locations, the Transformer model had a root mean square error of 0.3412 ug/m3, which compare extremely favorably to the ARIMA and ETS models that both had root mean square measure over 100. Based on this metric across multiple test set, we conclude that the Transformer architecture model can be used to model PM2.5 levels for investment decision making. The first chart below shows the predicted values in blue, true values in orange, and overlapping predictions as brown. It appears that the model will under predict outliers, which could become costly if not considered. The second chart displays the rolling 3-hour, 24-hour, and yearly means against the new PM2.5 regulation thresholds. It can be observed that the 24-hour rolling mean PM2.5 levels break the appropriate threshold multiple times. This threshold break will be valued in the final section of this analysis.




3H mean
24H mean
yearly mean
1000
800
600
400
200
0
0
5000
10000
15000
20000
25000
30000
35000

Conclusion

With the reliable Transformer model, we can now come up with cost projection for our power pants for each of the three considered locations, Tiantan, Wanliu, and Wanshouxigong.

Let's answer our Business Questions from the Introduction section of the analysis:

1. ***Which new location would have exceeded the new law's emissions standards, based on the predictions of the model?***
    a. ***Which emission thresholds were broken and how many times?***

Below is a forecasted cost report for all three sites under consideration. All new locations are projected to break the 24hour threshold, and only the 24hour threshold. However, each location would come at a significant cost of revenue, ranging from 22% - 25% of projected yearly revenue.

****

PRSA_Data_Tiantan_20130301-20170228.csv Forecasted Pollution Cost:

Threshold for 1 periods costs $0.00 on 0 violations

Threshold for 3 periods costs $0.00 on 0 violations

Threshold for 24 periods costs $63,000,000.00 on 63 violations

Threshold for 8760 periods costs $0.00 on 0 violations

Total Forecasted Pollution Fine (2013-2017): $63,000,000.00

Cost Per Year $15,739,219.71/year

34.98% Of Yearly Projected Revenues

****

PRSA_Data_Wanliu_20130301-20170228.csv Forecasted Pollution Cost:

Threshold for 1 periods costs $0.00 on 0 violations

Threshold for 3 periods costs $0.00 on 0 violations

Threshold for 24 periods costs $39,000,000.00 on 39 violations

Threshold for 8760 periods costs $0.00 on 0 violations

Total Forecasted Pollution Fine (2013-2017): $39,000,000.00

Cost Per Year $9,743,326.49/year

21.65% Of Yearly Projected Revenues

****

PRSA_Data_Wanshouxigong_20130301-20170228.csv Forecasted Pollution Cost:

Threshold for 1 periods costs $0.00 on 0 violations

Threshold for 3 periods costs $0.00 on 0 violations

Threshold for 24 periods costs $58,000,000.00 on 58 violations

Threshold for 8760 periods costs $0.00 on 0 violations

Total Forecasted Pollution Fine (2013-2017): $58,000,000.00

Cost Per Year $14,490,075.29/year

32.20% Of Yearly Projected Revenues

2. ***How accurate are the model's predictions?***

Across all three locations under consideration, the Transfomer model had a root mean square error of 0.3412 ug/m3, which was an over 100x improvement over baseline, and utilizes state-of-the-art neural network architecture.

3. ***Which site should be decided upon based on the analysis?***

Based on the forecast and cost projection, **none of these power plant locations should be considered for investment**. This is due to the fact that we are projecting a minimum pollution fine of 22% of revenues with a reliable model, that does will not account unforeseen one-hour spikes that may cause additional fines after the power plant's equipment depreciates over time. However, if management must decide to invest in a new power plant, the power plant should be built in Wanliu.

Citations

1. What are the air quality standards for PM? | air quality Planning unit | Ground-level Ozone | New England | US EPA. (2019, October 10). Retrieved March 28, 2021, from https://www3.epa.gov/region1/airquality/pm-aq-standards.html
2. Particulate matter (pm) basics. (2020, October 01). Retrieved March 28, 2021, from https://www.epa.gov/pm-pollution/particulate-matter-pm-basics
3. UCI machine Learning Repository: Beijing Multi-Site AIR-QUALITY data data set. (n.d.). Retrieved March 28, 2021, from https://archive.ics.uci.edu/ml/datasets/Beijing+Multi-Site+Air-Quality+Data#
4. Kalman filter. (2021, March 27). Retrieved March 28, 2021, from https://en.wikipedia.org/wiki/Kalman_filter#:~:text=In%20statistics%20and%20control%20theory,than%20those%20based%20on%20a
5. Sklearn.preprocessing.StandardScaler¶. (n.d.). Retrieved March 28, 2021, from https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.StandardScaler.html
6. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A., . . . Polosukhin, I. (2017, December 06). Attention is all you need. Retrieved March 30, 2021, from https://arxiv.org/abs/1706.03762
7. Alammar, J. (n.d.). The illustrated transformer. Retrieved March 30, 2021, from http://jalammar.github.io/illustrated-transformer/