Forecasting Fine Grained Pollutant Levels

Nilesh Patil and Jiang Shang

December 17, 2016

1 Introduction

Air pollution has attracted more and more attention in recent years, especially in fast growing urban cities in developing countries where air contaminant becomes part of people's daily experience. According to WHO, air pollution is a major environmental risk to health. By reducing air pollution levels, countries can reduce the burden of disease from stroke, heart disease, lung cancer, and both chronic and acute respiratory diseases, including asthma.The lower the levels of air pollution, the better the cardiovascular and respiratory health of the population will be, both long- and short-term[2]. Therefore, it is imperative that we have the ability to forecast pollutant levels so that local authorities can issue health alert more effectively and design area specific pollution control measures accordingly.

In every pollutant monitoring study, fine grained particulate measurements are mentioned alongside the gaseous pollutants. In general particulate matter in atmosphere is driven by natural activity and the most common source of particulate matter is still oceanic salt sprays. Other natural sources are volcanic activity, storms, forest and grassland fires etc. In recent times, human activities like increasing usage of fossil fuels has led to significant jump in anthropogenic aerosols (those made by human activity) and it now accounts for about 10% of total atmospheric aerosols[9]

Figure 1 depicts a heat map of yearly relative average PM2.5 value. We can see that it is highly correlated with human activity centers and exposed sea-surface.

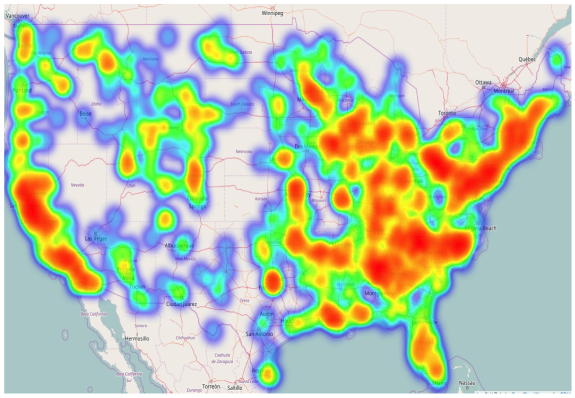

Figure 1: PM2.5 concentration across USA

The term fine grained pollutants refers to the particles in atmosphere having size smaller than 2.5µm. In our analysis and forecasting, we are focused on PM2.5 (Particulate Matter with less than 2.5µm diameter) since they are especially dangerous with a 36% increase in lung cancer per 10 µg/m3 beyond current safety standards[13].Since they are so small and light, they stay longer in air and this increases the chances of inhaling by humans and animals. PM2.5 are able to bypass the nose and throat and penetrate deep into the lungs and some may even enter the circulatory system. Long-term exposure to PM2.5 may lead to plaque deposits in arteries, causing vascular inflammation and a hardening of the arteries which can eventually lead to heart attack and stroke.

Scientists in the study estimated that for every $10\mu g/m^3$ increase in fine particulate air pollution, there is an associated 4%, 6% and 8% increased risk of all-cause, cardiopulmonary and lung cancer mortality, respectively[12]. The goal of our project is to construct a predictive model for PM2.5 across USA using input parameters such as temperature, wind, pressure,etc. Our ideal output is the PM2.5 level of the next 24 hours given historical conditions.

2 Literature Review

Our project is inspired from Dr. Yu Zheng's study in Beijing and Shanghai. Dr. Zheng's study uses two separate classifiers, one being spatial classifier based on an artificial neural network (ANN), the other one is temporal classifier based on a linear-chain conditional random field (CRF). Instead of inferring directly from the data, they first build these two classifiers and then use them to infer air quality. They also used co-training method, which is a semi-supervised learning technique that requires two views of the data. It assumes that each example is described by two different feature sets that provide different and complementary information about an instance[14]. Our project is different from Dr. Zheng's in that he zooms in a single city each time and predicts its AQI (Air Quality Index)value according to inferred PM2.5 level. His study also includes parameters such as human mobility, POIs(point of interest),and road network data with an aim exceeds the scope of our project.

Missing value is always the most excruciating part in the path of data analysis. In order to find the best way to treat missing values, we first need to know why they are missing at the first place. Generally speaking, there are four reasons why data are missing. Data may be missing completely at random, meaning the probability of missingness is the same for all units. In this case, throwing out missing values will not bias any result obtained for the objection of the study. Also, data maybe missing not completely at random. For example, it is quite possible that the response rate for question "what date is today?" turn out to be much higher than the question "how much salary do you earn?". It is not random because people are less likely to give up their personal information than to answer an objective question. Hence, ignoring such dis-randomness may lead to biased findings.

Also, data maybe missing because of unobserved predictors, meaning that not only it is not missing at random but it also depends on information that has not been recorded and this information also predicts the missing values. Finally, the most problematic reason for a value being missing is simply due to itself. For example, given that high income people are less likely to reveal their true income, asking this question may just censoring all of them out.This type of missing value is the most complicated in that the nature of the missing-data mechanism may force these predictive models to extrapolate beyond the range of the observed data[8]. Missing values is the biggest challenge for us in this project. And as we cannot either identify the reasons or to prove their validity for our missing values, we attempted multiple matrix imputation methods, and compare their advantages and disadvantages each respectively,which we will discuss more in depth later in the matrix imputation section.

3 Methodology Overview

3.1 Data

The Primary data source for our analysis and forecast is US-EPA[1]. EPA provides historical datasets flat files compressed in CSV(Comma Separated Variable) format and we are using the following datasets:

- Particulate readings across 250+ locations in USA from 2009 onward
- Atmospheric conditions across USA from 2009 onward
- Gaseous pollutant readings from 2009 onward
- Each of the above datasets are provided at a daily level

Here is a full list of the 26 columns in our raw data: State Code,County Code,Site Num,Parameter Code,Parameter Occurrence Code,Latitude,Longitude,Datum,Parameter Name,Sample Duration,Pollutant Standard,Date Local,Units of Measure,Event Type,Observation Count,Observation Percent,Arithmetic Mean,1st Max Value,1st Max Hour,Air Quality Index,Method Code,Method Name,Local Site Name,Address,State Name,County Name,City Name,CBSA Name,Date of Last Change.We then get rid of columns which we deemed irrelevant for our analysis, and boil down to 9 columns that provide us essential information for model building:

Field Position	Field Name	Description
1	Data Local	The calendar date for the summary
2	City Name	The name of the city where the monitoring site is located
3	County Name	The name of the county where the monitoring site is located
4	State Name	The name of the state where the monitoring site is located
5	Latitude	The monitoring site's angular distance north of the equator measured in decimal degrees
6	Longitude	The monitoring site's angular distance east of the prime meridian measured in decimal degrees
7	Arithmetic Mean	The average (arithmetic mean) value for the day

We use these 12 variables as our parameter inputs:temperature,pressure,relative humidity, wind speed, dew point, ozone, SO2, CO, NO2,PM2.5,month number and day of the week. Besides month number and day of the week,each of them come in a separate file contains the above-mentioned 26 columns, and we reduced the number of columns by the same method for each of them. We combine all of these parameters in one file with each row of the new file represents one county, one day, and corresponding readings of the day. We also computed weekly and monthly averages for each county in order to compare with daily readings for optimizing our model prediction accuracy. Instead of taking the mean of each variables, we use matrix imputation to replace missing values.

3.2 Matrix Imputation

As in most other data mining tasks involving large quantity of data, missing value is a commonplace. There are many ways for handling missing values, such as simply ignoring them, throwing out observation containing missing data, or replacing them with the attribute mean. In this project, we tried out several matrix imputation methods to infer missing data:

- KNN (K-Nearest Neighbor) Given N training vectors, KNN algorithm identifies the k nearest neighbors of 'm', the missing value. The value of these k neighbors is then weighted by their respective distance.First the smallest k distances are extracted into the variable smallest.distances Then, the corresponding values are extracted to knn.values. Finally, knn.weights normalizes the distances by the max distance, and are subtracted by 1. The result is the weighted mean of the values of the nearest neighbors and their weight based on their distance[5] In our case, due to the dispersion of our data, we set k=100, meaning to find 100 nearest rows which have a feature to fill in each row's missing value. To achieve this goal, KNN has to compute the pairwise distances between 1221755 samples.A memory error is thus raised due to insufficient space on personal laptop. In general, despite the fact the KNN algorithm is robust to noisy data, and it's particularly effective when the data is large, a grave drawback is that it has high computational cost because it needs to compute distance of each query instance to all training samples. This is also the reason we give up on using KNN as our final methods for matrix imputation.

- MICE (Multiple Imputation by Chained Equations) MICE works in the following steps: 1) It creates an simple imputation, such as mean, for every missing value as "place holders". 2) The "place holders" imputed for one variable ('var') is set back to missing. 3) The observed values from the variable "var" in Step 2 are regressed on the other variables in the imputation model, which may or may not consist of all of the variables in the dataset. 4) The missing values for "var" are then replaced with predictions (imputations) from the regression model. It then keeps repeating the last 3 steps until all the missing values are imputed for each variable[6].Although MICE theoretically reduce the biased of statistical uncertainty due to its multiple imputation nature, its assumes missing data are missing at random. We cannot prove that for our dataset, and thus implementing MICE when missing data is not missing at random will bias the imputed result.

- softImpute attempts to complete a matrix by iterative soft thresholding of SVD (Singular Value Decomposition).This allows us to efficiently compute an entire regularization path of solutions on a grid of values of the regularization parameter[10]. After comparing with

3.3 Model

The core idea of our approach is to use historical readings for physical parameters for predicting PM2.5 level of the next 24 hours.To achieve our goal, we use Linear Regression, Random Forest, and Gradient Boosting Machine respectively and divide our dataset into 70:30 training-test division as well as using a 10-fold cross validation.

- Linear Regression is the most simple therefore most best known model. It's representation is a linear equation. Making predictions is as simple as solving the equation for a specific set of inputs. Due to strict assumptions for a least squares based linear regression model, the model might not lead to a decent predictive function in a real world data-set unless we avoid a large number of pitfall carefully. As in our example shown in Figure 2, actual values and predicted ones scatter all over the plot, indicating that the linear regression model is not learning at all.

- RandomForest is one of the most widely used ensemble method. Random forests are a combination of tree predictors such that each tree in the ensemble is built from a sample drawn with replacement from the training set[3]. As the number of trees grow, the generalization error for forests becomes less and less and eventually converges.As our result shown in Figure 3. In addition,a randomly selected subsets of all features are used to split each node during the construction of the tree. Although this randomness may result in a black-box like approach, RandomForest model in general yields good result and is robust with respect to noise[7]. In fact, in this project, RandomForest is the most predictive model which outperforms both Linear Regression as well as Gradient Boosting Machine, with an error rate cluster around 0 within 20% of the real value excluding outliers.

- Gradient Boosting Machine produces a prediction model in the form of an ensemble of weak prediction models. The learning procedure consecutively fits new models to provide a more accurate estimate of the response variable.It allows for the optimization of arbitrary differentiable loss functions[4].The new base-learners are constructed to be maximally correlated with the negative gradient of the loss function, associated with the whole ensemble[11].

  In our project, we set the loss function to be least squares regression, and we choose 100 as the number of boosting stages to perform since GBM is fairly robust to over-fitting so a large number usually results in better performance.We also shrinks the contribution of each tree by 0.1, and we limits the number of nodes in each tree by setting max_depth equal to 3. As the result shown in figure 6, although GBM performs better than linear regression model, it still renders a higher error rate than RandomForest

4 Results

4.1 Linear Regression : model results

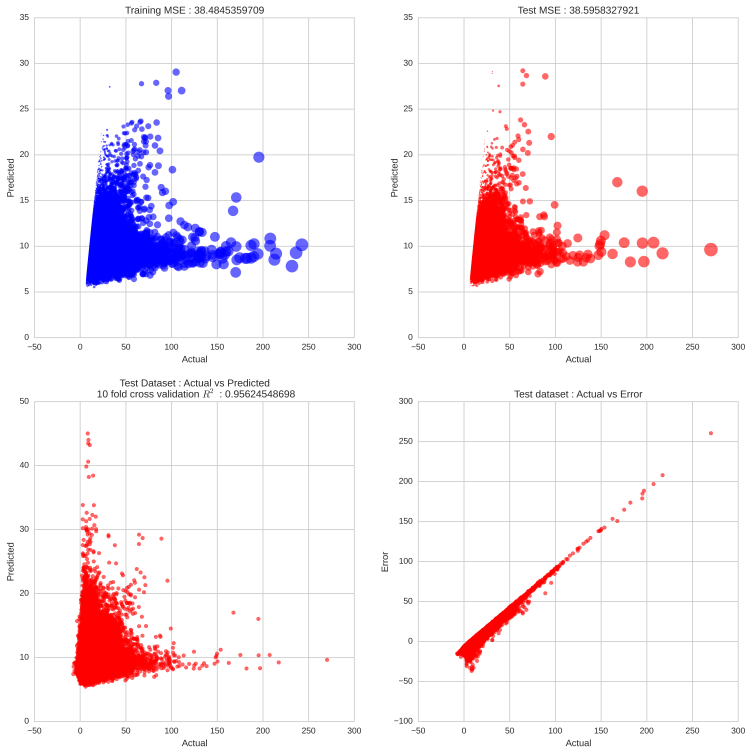

Figure 2: Result Obtained from Linear Regression Model

- The top-left plot shows the actual PM2.5 value against predicted value in the training set using linear regression with size of the datapoints set to the degree of error. A MSE (Mean Standard Error) of approximately 38.957 is calculated, which is not very supportive of the predictivity of our model
- The top-right plot shows the same thing except for testing dataset. Its MSE is approximately 39.091
- The lower-left plot shows there is no correlation in the test set between the predicted PM2.5 value and the real value. It reports a very low R-square value of approximately 0.04, meaning only 4% of variation in the prediction is due to the actual PM2.5 value
- The lower-right plot depicts what the real PM2.5 value is against prediction error. We observe a linear relationship, meaning that as the real PM2.5 value increase, prediction given by linear regression model becomes less and less accurate. In fact, even at lower PM values, the model isn't predicting at all.
- A conclusion we draw from analyzing these graph is that the linear regression model has no predictive power for our data

4.2 Random Forests

4.2.1 Determining optimal parameters

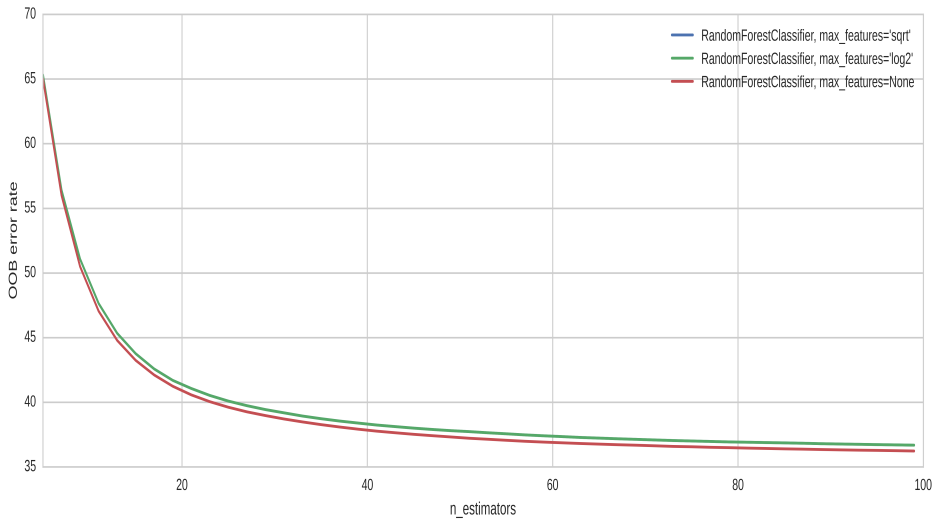

Figure 3: Performance Evaluation for Different Numbers of Splitting Features

Figure 3 shows the measurement of out-of-bag(OOB) error at the addition of each new tree during training using Random Forest.The resulting plot allows us to approximate a suitable value of n estimators at which the error stabilizes. Each colored curve represents a number of features to consider when looking for the best split. From the graph we can see that regardless of the number of features being consider when splitting, the resulting error rate stabilize approximately when n is greater than 20. Due to compute power constraint, we use n=50 for our model construction.

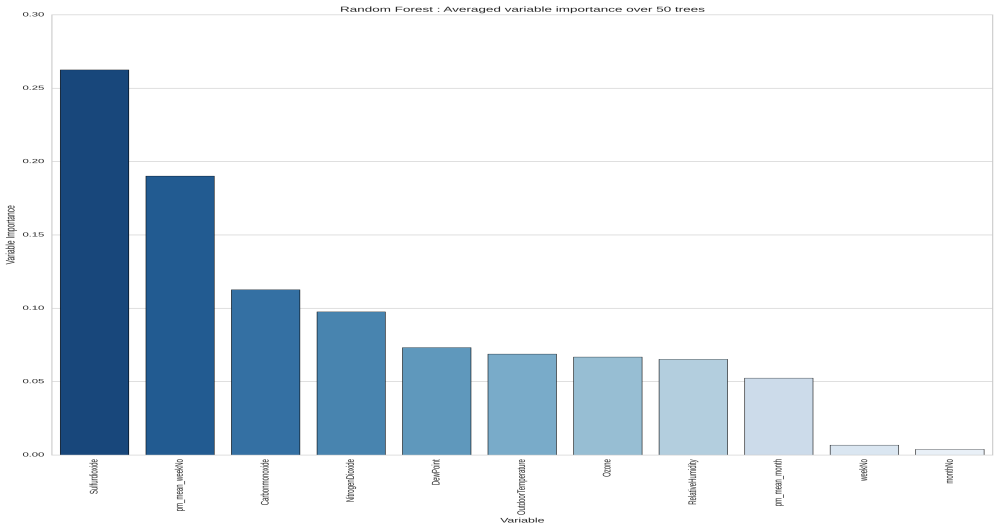

Figure 4: Variable Importance ranked by RandomForest Model

Figure 4 is quiet straight-forward in that it summarize the importance of each parameter over 50 trees. Far from what we surmised originally, week number and month number turn out to be the least influential factors in our prediction while SO2 plays the most important role in predicting the following day's PM2.5 level accounting for more than 25% of our model's accuracy with its weight exceeds even the weekly average PM2.5 of the previous week. All the other parameters are roughly equally important.

4.2.2 Random Forest - model results

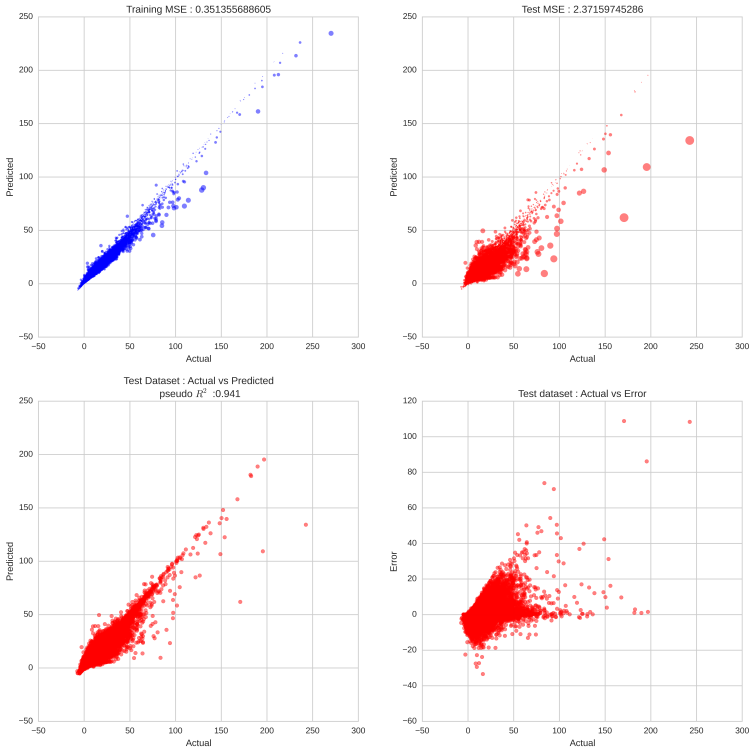

Figure 5: Results Obtained from Random Forest Model

Figure 5 is an evaluation of the performance of Random Forest model using grid search over a limited space with 50 trees and n=log2(subset_vars) variables for each split.

- The top-left plot shows the actual PM2.5 value against predicted value in the training set using random forest with size of the data-points set to the degree of error. A MSE (Mean Standard Error) of approximately 2.944 is calculated, which falls between the second and third standard deviation, indicating strong support for the predictive nature of our model
- The top-right plot shows the same viz for testing data-set. Its MSE is approximately 2.689
- The lower-left plot lays out predicted vs actual only and as we can see, our random forest model is doing a much better job than linear regression model, yet there are a few values being predicted incorrectly by huge margins as shown in the second plot
- The lower-right plot depicts what the real PM2.5 value is against prediction error. As we can see, most of the PM2.5 value is being predicted correctly with the error rate clustered around 0 within 20%.
- We can safely conclude that Random Forest has yielded far better result than what we can obtained from simple linear regression. The prediction accuracy has increase tremendously but sill have room to improve.

4.3 GBM - model results

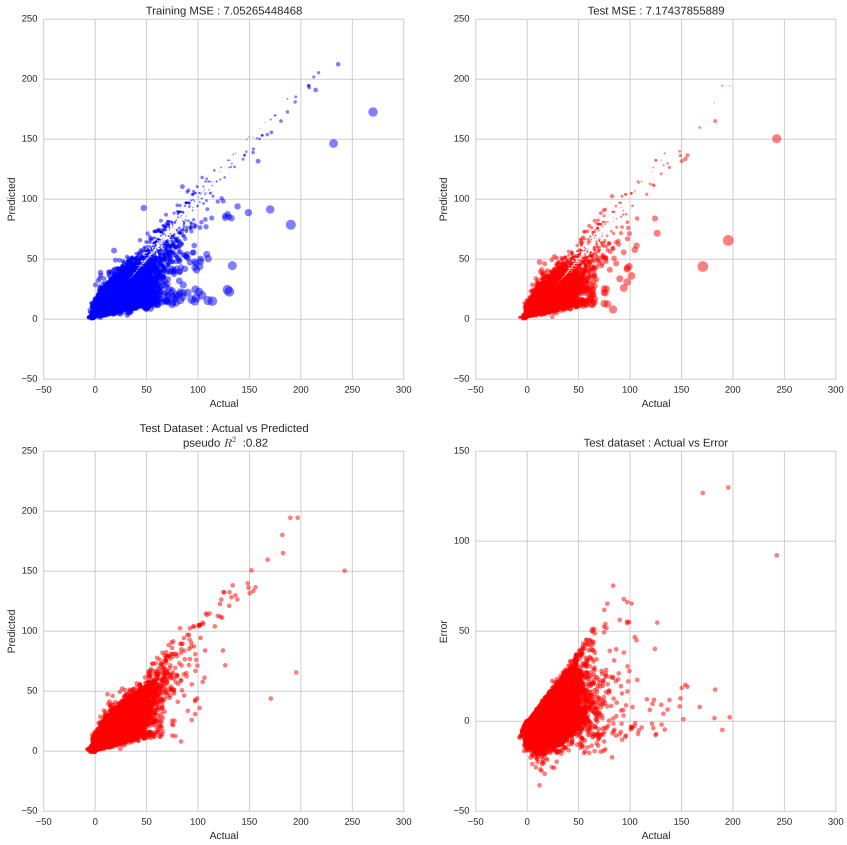

Figure 6: Results Obtained from Gradient Boosting Machine Model

The first plot shows the actual PM2.5 value against predicted value in the training set using GBM with size of the data-points set to the degree of error. A MSE of approximately 7.052 on training set and 7.17 on test set implies a better fit than linear regression. As the latter plots show, the fit is just slightly poorer than Random Forest, but still much more predictive in nature compared to linear model. One of the most important considerations was run-time for building the model. GBM has a much higher run-time than Random Forests.

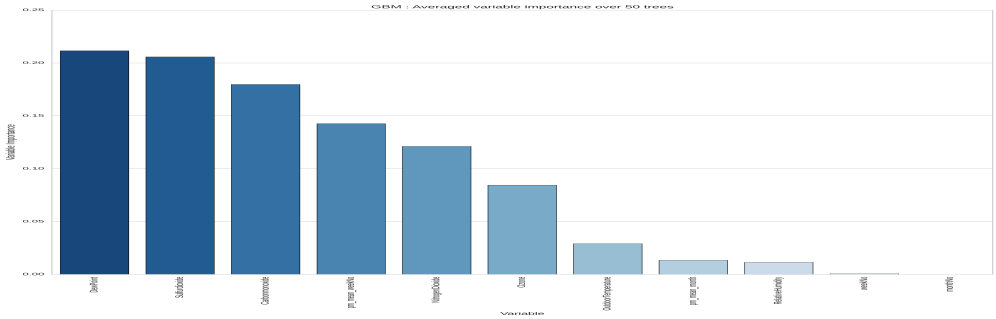

Figure 7: Variable Importance ranked by Gradient Boosting Machine Model

5 Problems encountered and Future Work

Our main issue stems from the fact that the data-set contains 70%+ missing values in each variable and large scale matrix imputation turned out to be essentially useless due to memory requirements. We used FancyImpute in python for imputing missing values but there results are almost always close to zero for each method that we were able to run.

- We would like to work on large scale matrix imputation using Spark and a distributed version of KNN.
- Our assumption is, that given the extremely biased nature of input parameters in the training data-set, if we are able to get better estimates of missing values in future iterations, our final model should see a noticeable increase in the performance metrics.
- We were not able to incorporate additional data (population, industry etc) in our models which might have led to better performance and our next steps are to work on augmenting our current data-set

References

[1] Airdata website file download page. http://www.who.int/mediacentre/factsheets/fs313/en/, 2016. [Online; accessed 15-December-2016].

[2] Ambient (outdoor) air quality and health. http://www.who.int/mediacentre/factsheets/fs313/en/, 2016. [Online; accessed 15-December-2016].

[3] Ensemble methods. http://scikit-learn.org/stable/modules/ensemble.html#random-forests, 2016. [Online; accessed 15-December-2016].

[4] Gradientboostingclassifer. http://scikit-learn.org/stable/modules/generated/sklearn.ensemble.GradientBoostingClassifier.html, 2016. [Online; accessed 15-December-2016].

[5] knn impute. http://finzi.psych.upenn.edu/library/imputation/html/kNNImpute.html, 2016. [Online; accessed 15-December-2016].

[6] Melissa J Azur, Elizabeth A Stuart, Constantine Frangakis, and Philip J Leaf. Multiple imputation by chained equations: what is it and how does it work? *International journal of methods in psychiatric research*, 20(1):40–49, 2011.

[7] Leo Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

[8] Andrew Gelman and Jennifer Hill. *Data analysis using regression and multilevel/hierarchical models*. Cambridge University Press, 2006.

[9] Mary Hardin and Ralph Kahn. Aerosols and climate change. http://earthobservatory.nasa.gov/Features/Aerosols/.

[10] Rahul Mazumder, Trevor Hastie, and Robert Tibshirani. Spectral regularization algorithms for learning large incomplete matrices. *Journal of machine learning research*, 11(Aug):2287–2322, 2010.

[11] Alexey Natekin and Alois Knoll. Gradient boosting machines, a tutorial. *Frontiers in neurorobotics*, 7, 2013.

[12] C Arden Pope III, Richard T Burnett, Michael J Thun, Eugenia E Calle, Daniel Krewski, Kazuhiko Ito, and George D Thurston. Lung cancer, cardiopulmonary mortality, and long-term exposure to fine particulate air pollution. *Jama*, 287(9):1132–1141, 2002.

[13] Ole Raaschou-Nielsen, Zorana J Andersen, Rob Beelen, Evangelia Samoli, Massimo Stafoggia, Gudrun Weinmayr, Barbara Hoffmann, Paul Fischer, Mark J Nieuwenhuijsen, Bert Brunekreef, et al. Air pollution and lung cancer incidence in 17 european cohorts: prospective analyses from the european study of cohorts for air pollution effects (escape). *The lancet oncology*, 14(9):813–822, 2013.

[14] Yu Zheng, Xiuwen Yi, Ming Li, Ruiyuan Li, Zhangqing Shan, Eric Chang, and Tianrui Li. Forecasting fine-grained air quality based on big data. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 2267–2276. ACM, 2015.