Final Project - Air Quality

subtitle

**Author:** Benhuang Qi   
Ying tung Lau  
Zoey Yang

Fan Peng

# Introduction

## Team Composition

When looking at our team, we would like to call ourselves the "PERFECT LADIES", a perfect combination of business and analytical minds. Among us we have:

Zoey, the all-round leader of our team who also happened to be a remarkable business talent. She always shows flashes of business instincts like no others.

Ying Tung, the core of our analytical mind. As a finance background student, she has the data analytics skills that outmatch those who worked their way through learning Math, Statistics or even Computer Science. She is 'the one' we turn to when we encounter any obstacles regarding data analyzing.

Fan, the member who has the biggest edge coming into this project. She has ample background knowledge of our topic of research as she was born and raised in Beijing—the city we researched on. Boasting ample analytics and quick learning skills, she tackles problems in a lightning fashion.

Chris, the only member of the team who has a Mechanic Engineering background. He brings to the table some different 'spices' when we cook up the ideas and solutions. He is also one of the "blue chip" coders and content producers in our team.

## Problem Context and Importance

Nowadays, climate change and air pollution are important issues facing us that are closely interlinked. Air pollutants include not only greenhouse gases but also methane, nitrous oxide and others. Emissions of them into the air can result in changes to the climate. For instance, when the pollution lands on ice and snow, it darkens them slightly, leading to less sunlight being reflected back into space and contributing to global warming.

Many environmental problems related to air pollution can be addressed by monitoring air quality and air pollutants. Specifically, quantification of particulate matter (PM) is crucial. According to the World Health Organization (WHO), each year an estimated 4.2 million people die from diseases such as lung cancer, chronic respiratory diseases, and heart disease caused by ambient air pollution. The particles in air with a diameter less than 2.5 micrometers, i.e. PM2.5, are mainly to blame for the serious health hazard, as the type of particles measured are so small that they can reach the bloodstream and can hurt the lungs and heart.

Currently, about 90% of the world’s population lives in regions where air quality levels exceed WHO limits. While there is such a large portion of the population suffering from the severe air quality problem, the air quality measurements are concentrated in the developed countries. The high-end techniques and associated expense for PM2.5 monitoring put up barriers for many countries, i.e. low- and middle-income countries where the population is more prone to diseases. Therefore, it is important to figure out effective solutions for these less affluent countries to monitor concentration of small particulate matter and address the environmental problems to assure public health.

# Research Questions and Hypotheses

As we identify and recognize the importance of our problem, we want to focus on the question of whether there is a practicable and cost-effective method to quantify the level of PM2.5. Specifically, we want to test the effectiveness of machine learning approaches in prediction and classification of PM2.5 values, using some more affordable and more available measurements.

To build models that fulfil the job, we have researched on important features related to PM2.5.

Firstly, we considered air pollutants variables. A key measurement of daily air quality is the Air Quality Index (AQI), which is calculated by PM2.5, PM10, SO2, NO2, O3, and CO. As the level of PM2.5 is directly related to the emission of gaseous pollutants, we have identified measurements of the four pollutants included in the AQI Index (i.e. SO2, NO2, O3, and CO) as our candidate predictor variables. We expected a strong correlation between the level of PM2.5 and the levels of these gaseous pollutants, and thus these variables would be the most important to have fairly accurate predictions.

Next, we considered the meteorological factors. A number of studies have studied and found consistent effects of meteorological conditions for different countries with different PM2.5 pollution levels and meteorological conditions. [[1]](#footnote-1) Therefore, we assumed a correlation between air quality and meteorological conditions and included common meteorological factors (i.e. temperature, precipitation, wind, relative humidity, and pressure) as our candidate predictor variables.

In addition to these emissions and meteorological factors, we also thought of time variables as influencing factors. We anticipated different levels of pollutants emissions at different times of the day due to levels of traffic and industrial activities. For instance, it is likely that maximum concentrations of PM2.5 are reached during daytime hours. Also, date of measurements and location of the monitoring sites should be controlled as we expected variations in PM2.5 concentration pattern due to these factors.

If our hypotheses were valid, we would be able to propose the most affordable methods for governments to measure air quality, i.e. predicting levels of PM2.5 with machine learning models built with various variables (pollutants, meteorology, time and location).

# Dataset: Beijing Multi-Site Air-Quality

Our data for analysis is a Beijing Multi-Site Air-Quality Dataset. Beijing, the capital city in China, has long been one of the world’s most polluted cities according to annual world air quality reports. Meanwhile, the city’s multiple air-quality monitoring sites and meteorological stations provide rich data on air-quality-related factors. Therefore, we have selected the Beijing city for a case study to illustrate our prediction models of PM2.5 concentration.

## Data Source and Data Description

The data is obtained from the UCI machine learning repository. The repository is a trusted source for data and is widely used by students and researchers for machine learning projects.

It provides important information related to data such as Data Set Characteristics, Data Source, Data Set Information, and Relevant Papers.

According to documentation, our dataset includes hourly air pollutants data from 12 nationally-controlled air-quality monitoring sites. The air-quality data are from the Beijing Municipal Environmental Monitoring Center. The meteorological data in each air-quality site are matched with the nearest weather station from the China Meteorological Administration. The time period is from March 1st, 2013 to February 28th, 2017.

A description of all columns in the dataset is shown in **Table 1**. There are basic information regarding location of the monitoring sites and measurement date and time, as well as measurements of most of the important variables we have identified, i.e. five air pollutants variables (PM2.5, SO2, NO2, CO, O3) and six meteorological variables (TEMP, PRES, DEWP, RAIN, wd, WSPM). Besides, as the dataset is originally collected and formed for an empirical research project, we have got some extra information regarding the data and the related study through reading the relevant paper listed on the repository.

**Table 1** Dataset Description

| **COLUMN NAME** | **DESCRIPTION** |
| --- | --- |
| **YEAR** | year of data in this row |
| **MONTH** | month of data in this row |
| **DAY** | day of data in this row |
| **HOUR** | hour of data in this row |
| **PM2.5** | PM2.5 concentration (ug/m^3) |
| **PM10** | PM10 concentration (ug/m^3) |
| **SO2** | SO2 concentration (ug/m^3) |
| **NO2** | NO2 concentration (ug/m^3) |
| **CO** | CO concentration (ug/m^3) |
| **O3** | O3 concentration (ug/m^3) |
| **TEMP** | temperature (degree Celsius) |
| **PRES** | pressure (hPa) |
| **DEWP** | dew point temperature (degree Celsius) |
| **RAIN** | precipitation (mm) |
| **WD** | wind direction |
| **WSPM** | wind speed (m/s) |
| **STATION** | name of the air-quality monitoring site |

## Data Preparation and Data Quality

Before starting to build models with our data, we have done a few data preprocessing steps.

Firstly, our data include some missing values denoted as NA; we have handled these values using both listwise and pairwise deletion methods. As air-quality variables are the ones that we pay attention to, we removed all the rows where observations of a majority of them are missing. By comparison, we kept the rows where observations of meteorological variables are missing as the relationship among the rest of the variables can still be analyzed. We then imputed the remaining missing entries of air-quality variables using the nearest non-NA observation after the missing entry. We imputed this way since commonly the air quality measurements in consecutive hours are close.

Next, we have dealt with the inconsistency in our data format of numerical values. As there are only a few entries of air-quality variables stored as floating-point values, we converted these values to integers.

Finally, we have created some new variables that might be helpful for modeling. We added a “SEASON” column containing Winter, Spring, Summer and Autumn based on the “MONTH” column, a “WD\_Class” (i.e. wind direction class) column containing NorthWind, SouthWind, and West/EastWind based on the “WD” column, and a “ZONE” column containing North and Central based on the “STATION” column.

Overall, our data has a good quality to start with in terms of completeness and consistency, after imputing or deleting missing data and uniforming number format. Also, our data is quite relevant to the business problem we have identified. The new variables we created further improve the usefulness of data for modeling and analysis, and the number of observations is large enough to generate meaningful insights related to the problem.

Some other data processing tasks that are required for specific modeling methods, such as normalization and one-hot coding, are also performed. These steps will be explained as we go to more details of our data and the modeling processes.

# Target Varaible and Feature understanding

With our wrangled dataset, we look into our target variable and candidate predictor variables with descriptive statistics and data visualization. It is helpful for understanding variable characteristics and determining which variables to include in the analysis and which might be redundant.

## Target Variable

For reasons we have stated in previous sections, PM2.5 is the target we are trying to understand and predict. It is a criteria pollutant with clear definition and well-established standards. In our dataset, we have hourly PM2.5 concentration measurements from Beijing Municipal Environmental Monitoring Center from March 1st, 2013 to February 28th, 2017.

In the context of regression modeling, we would like to predict the numerical value of PM2.5 concentration. The target variable is a continuous variable as in the original dataset. In the context of classification modeling, we would like to predict the level of PM2.5 concentration. The target variable is a binary variable that indicates whether the concentration of PM2.5 is above or below 150 μg/m3 for a particular day. According to the U.S. Environmental Protection Agency, it is unhealthy for the general population when PM2.5 concentration is above 150 μg/m3. A new column named “PM2.5\_Category” is created for the purpose.

## Predictor Variables

Our candidate features presented in the dataset include four air pollutant variables (SO2, NO2, CO, O3), five meteorology variables (TEMP, PRES, DEMP, RAIN, WD\_Class), three time variables (YEAR, HOUR, SEASON), and one location variable (ZONE). Among them, YEAR, WD\_Class, SEASON, and ZONE are categorical variables. We have converted these variables to dummies to better illustrate and model the relationships between them and the target variable. A full list of our candidate predictor variables is as follows:

**Table 2** Candidate Predictor Variables

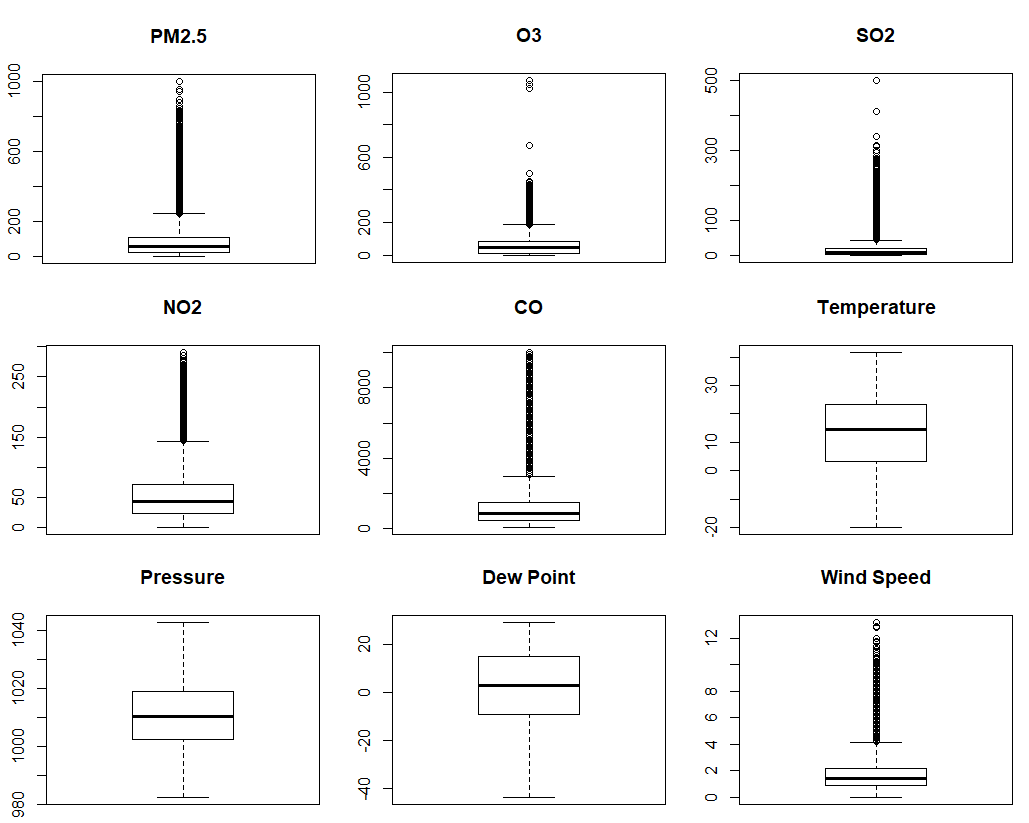
| **Pollutants** | **Meteorology** | **Time & Location** |
| --- | --- | --- |
| SO2 | TEMP | YEAR\_2016 |
| NO2 | PRES | YEAR\_2017 |
| CO | DEWP | HOUR |
| O3 | RAIN | SEASON\_Summer |
|  | WD\_Class\_Northwind | SEASON\_Winter |
| WD\_Class\_Southwind | ZONE\_North |
| WSPM |  |

## Descriptive statistics and Data Visualization

Table 3 shows the descriptive statistics of our numerical variables. We can see that our target variable and the air pollutants variables have a wide range of values. The variations in PM2.5, SO2, NO, and O3 concentration are relatively small. However, the standard deviation of the CO variable is exceptionally large, indicating there are many observations of extreme CO concentration levels. For meteorological variables, the observations are in much smaller ranges, suggesting no occurrence of extreme weather conditions. Also, there are considerable variations in variables TEMP and DEWP with respect to their means, which could be due to the city’s distinct seasons with special climate conditions. Besides, rain seems to be a rare event in the city as both of its mean and median are close to zero. The box plots shown in Figure 1 give more information about the distribution of values of our numerical variables.

**Table 3** Descriptive Statistics

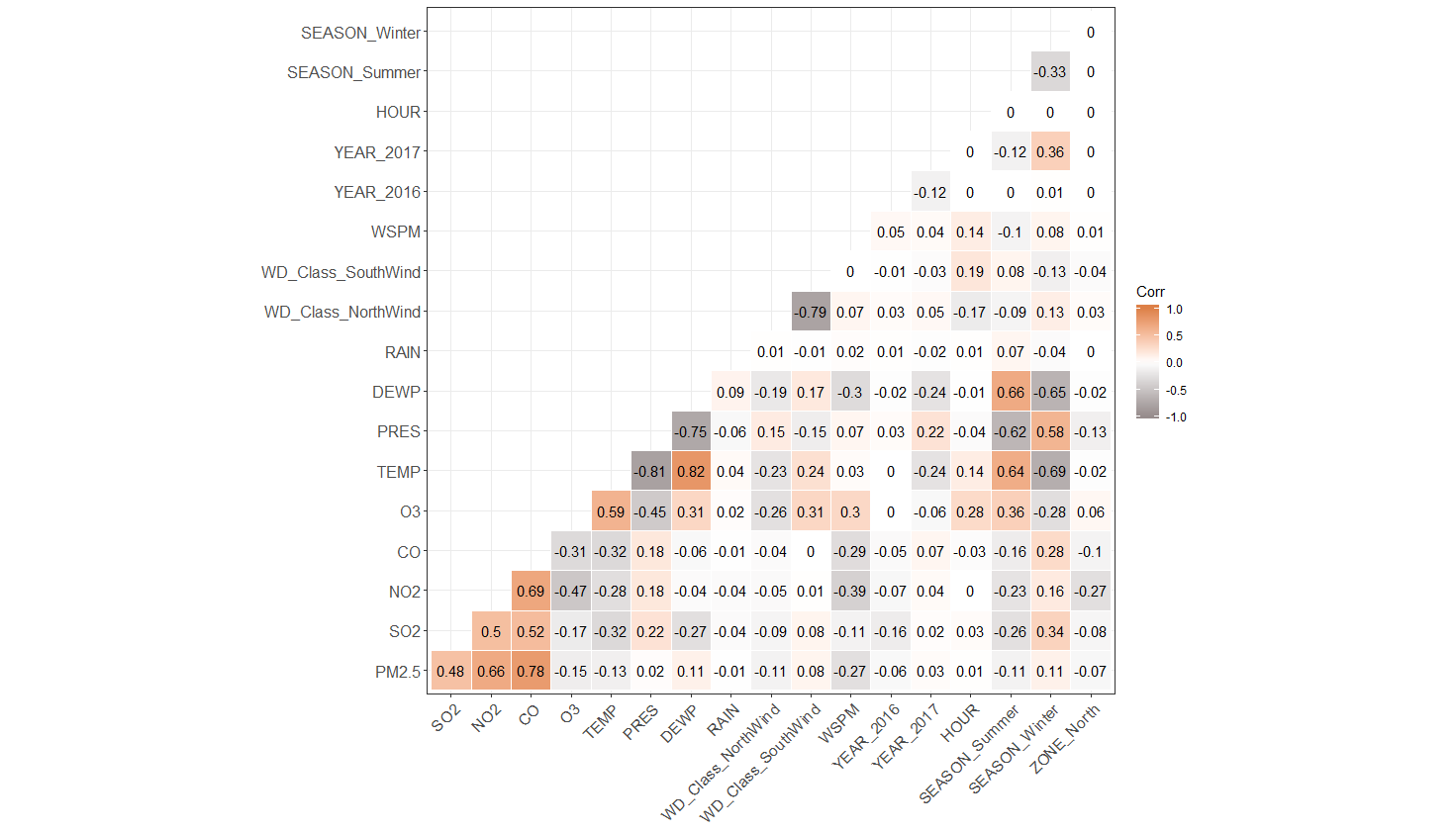
| **Variable** | **Min** | **Max** | **Mean** | **Median** | **Mode** | **SD** |
| --- | --- | --- | --- | --- | --- | --- |
| **PM2.5** | 2 | 999 | 80 | 55 | 3 | 81 |
| **SO2** | 0 | 500 | 16 | 7 | 2 | 22 |
| **NO2** | 1 | 290 | 51 | 43 | 21 | 35 |
| **CO** | 100 | 10000 | 1230 | 900 | 300 | 1159 |
| **O3** | 0 | 1071 | 57 | 44 | 2 | 57 |
| **TEMP** | -20 | 42 | 14 | 15 | 3 | 11 |
| **PRES** | 982 | 1043 | 1011 | 1010 | 1019 | 10 |
| **DEWP** | -33 | 27 | 2 | 2 | 15 | 14 |
| **RAIN** | 0 | 25 | 0 | 0 | 0 | 1 |
| **WSPM** | 0 | 13 | 2 | 1 | 1.1 | 1 |



**Figure 1** Boxplots of Numerical Variables

We have then examined thee relationships of our target and predictor variables with correlation matrix and scatterplots. Figure 2 is a colored correlation matrix that shows correlations among our target variable and candidate predictor variables. The correlation coefficients between each pair of variables are shown, and the corresponding colors indicate different degrees of correlation. Positive correlations are represented by orange color and negative correlations are represented by gray color, with varying color intensities in accord with the values.

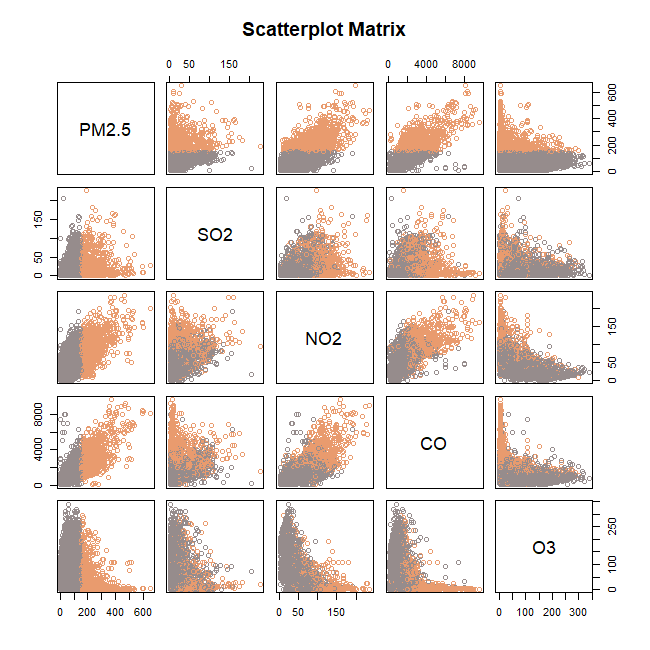
From the matrix we see that our target variable (i.e. PM2.5 concentration) is highly positively correlated with the concentration of pollutants variables SO2, NO2, and CO, meaning that with higher level of SO2, NO2, and CO in the air, the concentration of PM2.5 in the area is likely to be higher. The level of O3, in comparison, only has a weak correlation with our target variable. Looking at correlation coefficients with meteorological variables, we see that our target variable is moderately negatively correlated with temperature(TEMP), WD\_Class\_NorthWind dummy, and wind speed (WSPM), and is slightly positively correlated with dew point (DEWP), and WD\_Class\_SouthWind dummy. Besides, the matrix displays some correlation between our target variable and the time and location variables. Specifically, there are moderate negative correlations between PM2.5 concentration and SEASON\_Summer, ZONE\_North dummies. In like manner, we see a moderate positive correlation between PM2.5 concentration and SEASON\_Winter dummy.



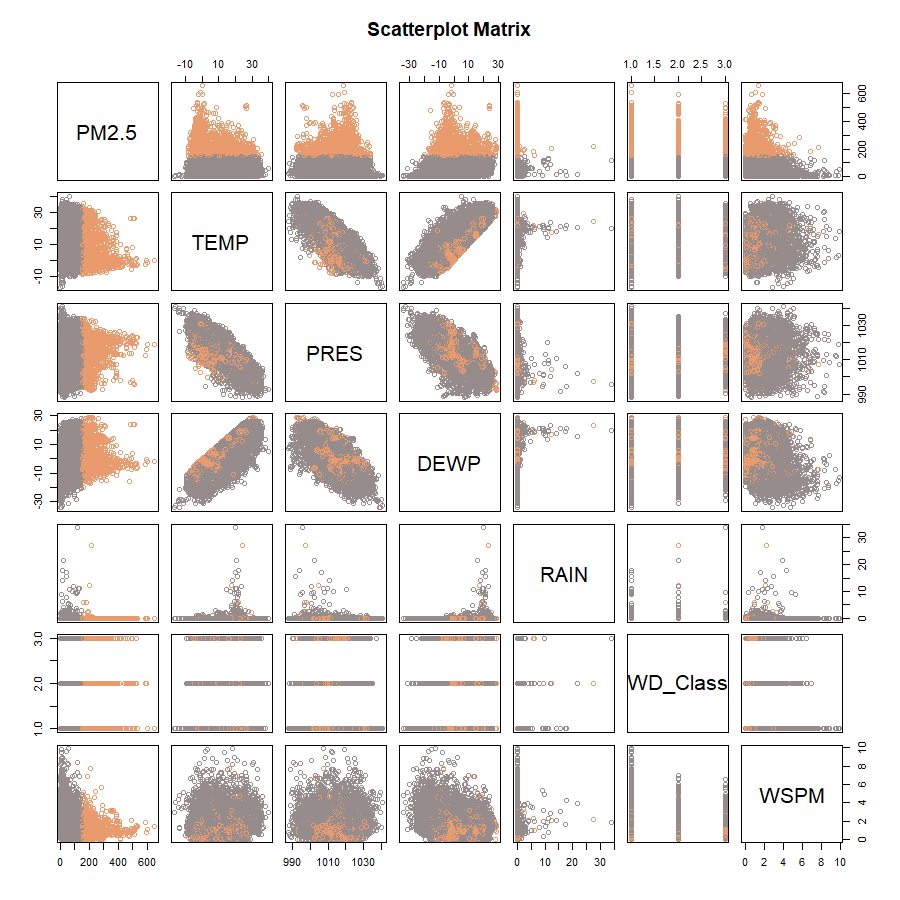
**Figure 2** Correlation Matrix

The relationships among the variables indicated here are consistent with our hypotheses. They are further explored with scatterplot matrices (shown in **Figure 3** and **Figure 4**). The plots are also colored, with the orange and gray color denoting a high and low level of PM2.5 concentration respectively (i.e. above or below 150 μg/m3), which is consistent with our target variable for classification problem.

**Figure 3** is a scatterplot matrix of our target variable and air pollutants variables. From the plot we see extremely high PM2.5 concentration occurs when the concentrations of NO2 and CO are high and when the concentration of O3 is low. From **Figure 4** with meteorological variables we see more observations of extremely high PM2.5 concentration while the temperature (TEMP) is low and the dew point (DEWP) is low. Also, the concentration of PM2.5 is likely to be low on windy days, as shown by the scatterplot between PM2.5 and wind speed (WSPM). The relationships indicate much worse air quality in the winter season.



**Figure 3** Scatterplot Matrix of PM2.5 and Pollutants Variables



**Figure 4** Scatterplot Matrix of PM2.5 and Meteorological Variables

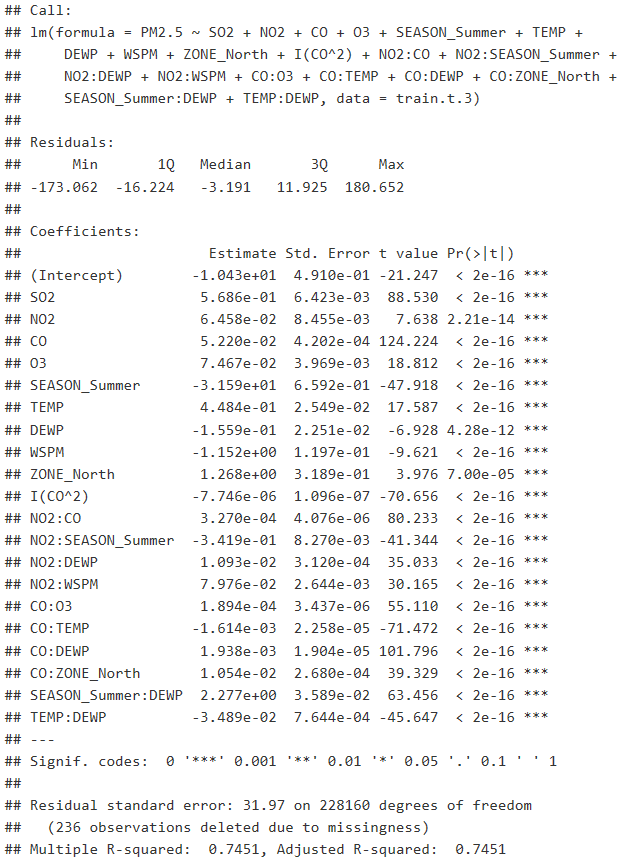
# Regression Models

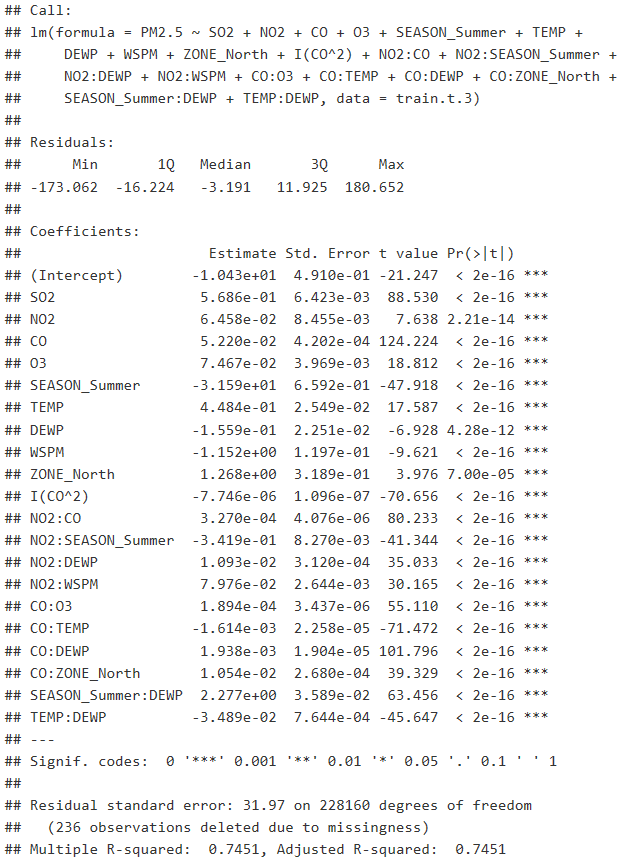
Regression refers to the prediction of the value of a continuous variable. As stated before, we want to test the effectiveness of machine learning approaches in prediction of PM2.5 values. We have built several regression models using the programming language R that predict the numerical value of PM2.5 concentration with our candidate predictor variables. In this section, we show the best regression model we have built and the best one built by DataRobot, an automated machine learning platform. We then compare and contrast these two models.

## Handcrafted Model: Multiple Linear Regression

The method we have used is Multiple Linear Regression (MLR). MLR is a popular statistical technique used for modeling relationship between a response variable and one or multiple explanatory variables. It estimates the coefficients of the regression formula from the data using the ordinary least squares (OLS) method.

Our best MLR model is as follows:





**Figure 5** Best Handcrafted Regression Model

In search of our best subset of predictors, we have used the exhaustive search method and tried to include interaction and polynomial terms in our model. As shown in **Figure 5**, our best model is built with 20 predictors: SO2, NO2, CO, O3, SEASON\_Summer, TEMP, DEWP, WSPM, ZONE\_North, CO^2, NP2:CO, NO2:SEASON\_Summer, NO2:DEWP, NO2:WSPM, CO:O3, CO:TEMP, CO:DEWP, CO:ZONE\_North, SEASON\_Summer:DEWP, TEMP:DEWP.

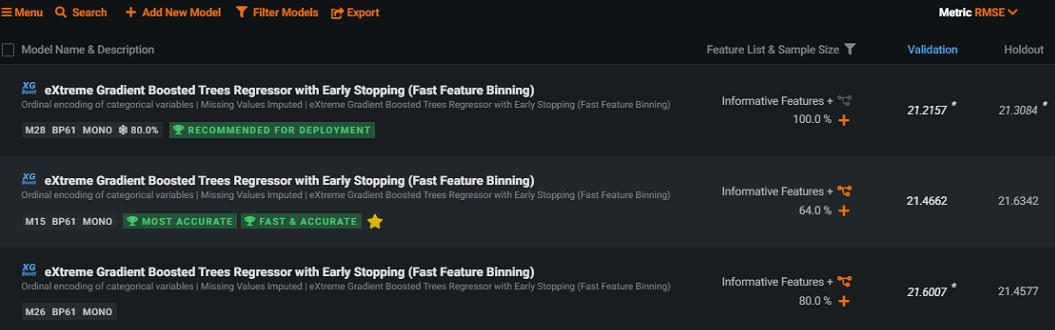
The prediction accuracy measures we have used to evaluate our model is the adjusted R-squared and RMSE. The adjusted R-squared and RMSE of our best model on training data are 0.7451 and 31.97. The validation adjusted R-squared and RMSE are 0.7261 and 42.27.

When building our model, we examined outliers using residual plots and further used Mahalanobis Distance, a measure of the distance between observations and their distribution to identify outliers to be removed. The residual plots of our model look reasonable after we removed the identified outliers. Meanwhile, the fitted regression model performs better than before as measured by adjusted R-squared and RMSE. However, although having not changed our model significantly, removal of outliers results in a potential overfitting problem as the predicting power on our validation set does not have an increase of the same magnitude, with the predicting power even becoming worse after excluding outliers.

Our model shows that the three classes of variables (air pollutants, meteorology, time) could be considered to predict the value of PM2.5 concentration. PM2.5 concentration is firstly correlated with air pollutant variables (especially SO2 and CO) and secondly with weather conditions. For instance, the coefficient on the SO2 variable indicates that keeping the values of other variables constant, for every unit increase in SO2 concentration, there is an average increase in PM2.5 concentration of 0.5. Also, interaction terms included in our model like SEASON\_Summer:DEWP take into consideration the interactive effects among variables, enabling us to figure out the exact effect on our target variable of a variable by controlling other variables correlated with it.

## DataRobot Model: Gradient Boosted Trees

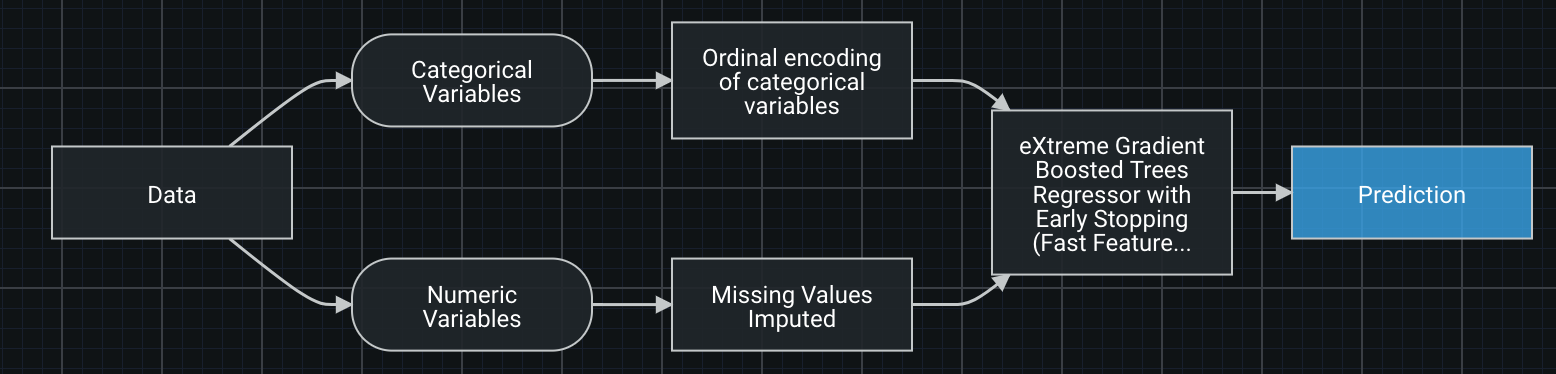
From DataRobot, we have got a total of 19 regresson models. The best-performing model is selected using the metric RMSE. As RMSE represents the error of estimate in regression, we have chosen the model with the lowest RMSE on the Holdout dataset. As shown in **Figure 6**, the best model is the eXtreme Gradient Boosted Trees Regressor with Early Stopping (Fast Feature Binning) model.



**Figure 6** Leaderboard of DataRobot Regression Models

As explained in the documentation of the DataRobot platform, Gradient Boosting Machines are a cutting-edge algorithm for fitting extremely accurate predictive models. As a gradient boosting method, GBM fits each successive tree to the residual errors from all the previous trees combined. The advantages of the method include requiring very little preprocessing, handling missing data elegantly, achieving a good balance between bias and variance, and being able to find complicated interaction term. Furthermore, GBMs can find the exact point in the training data where overfitting begins and halt one iteration prior to that, enabling modeling with the highest possible accuracy without overfitting.

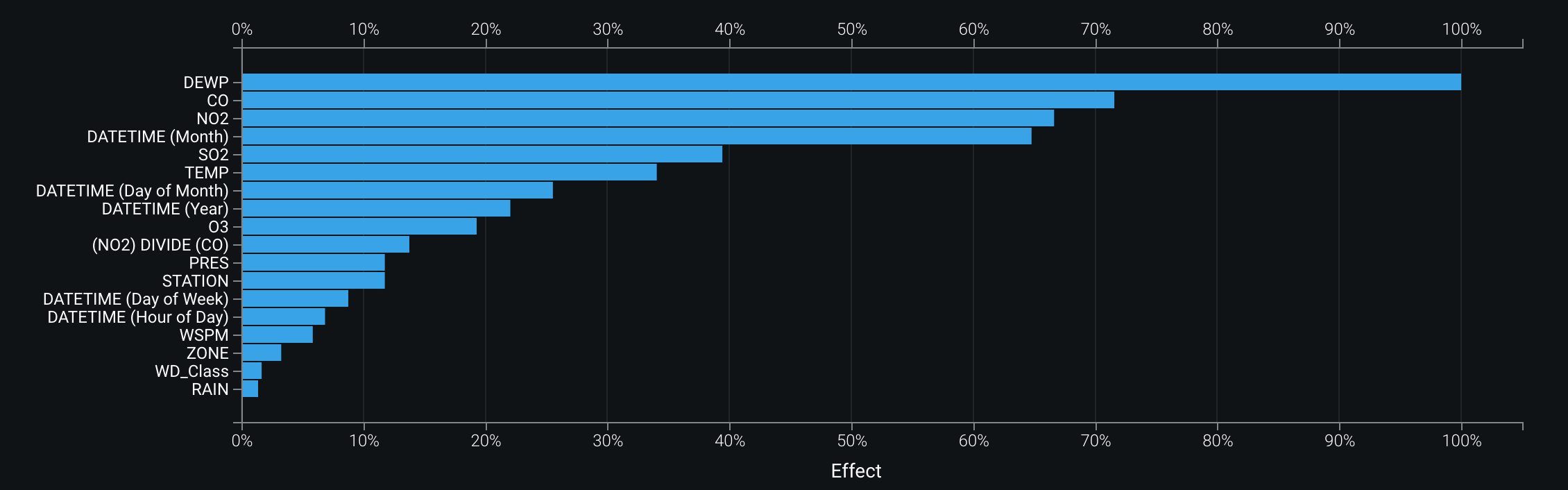
The automated modeling process is shown by the platform in a blueprint as follows:



**Figure 7** Blueprint of Best DataRobot Regression Model

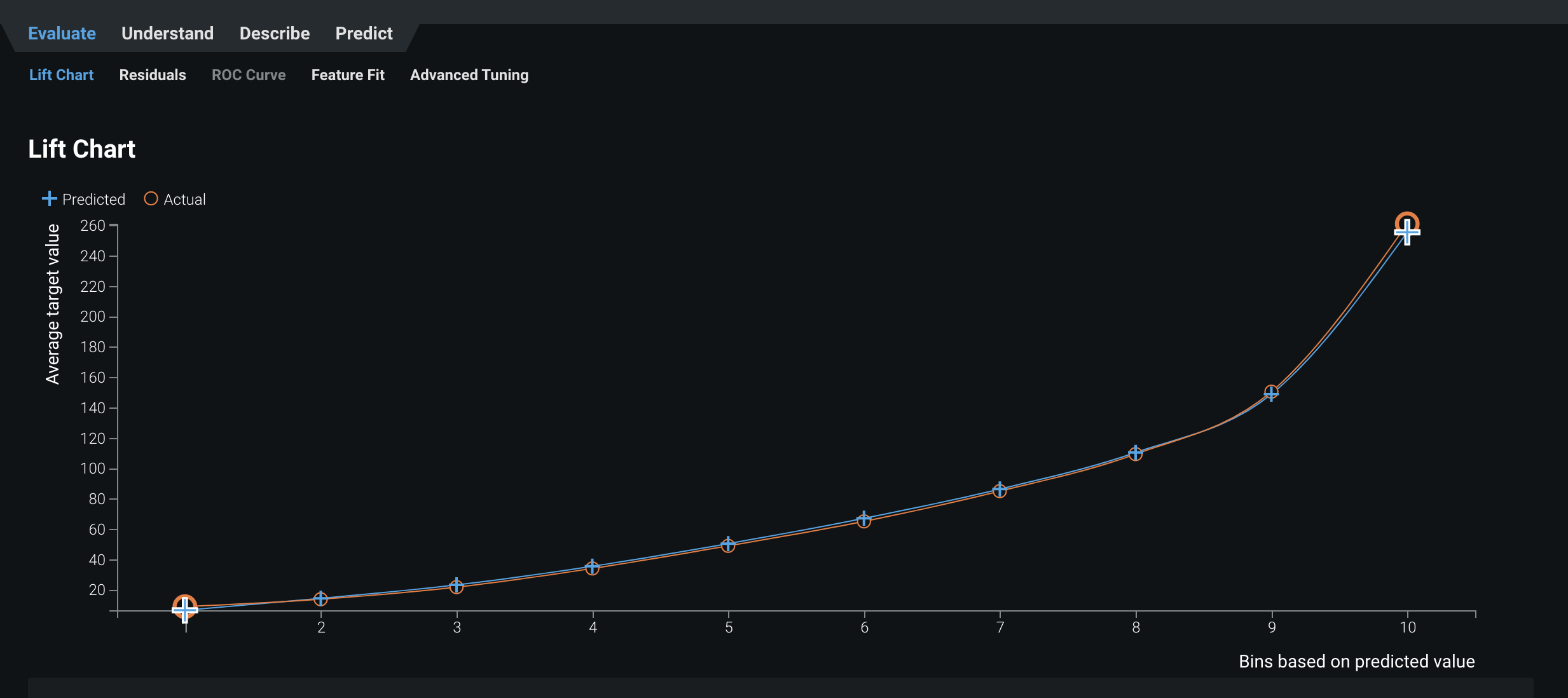
The blueprint is basically a map of how themodel is created by DataRobot. We see that since our dataset includes both categorical and numeric variables, in the first steps the algorithm separated these two types of variables and treated them differently. Categorical variables were converted to an ordinal scale, meaning recoding categories as integers based on either the lexicographic ordering of the categorical values, the frequency of the categorical values, the response, or randomly. For numeric variables, missing values were imputed with an arbitrary value (default -9999) or the median value. After these data processing steps, the Early Stopping Extreme Gradient Boosting Regressor was generated, which led to prediction results of the targeted variable.

We examine the “Feature Impact” subtab under the tab “Understand”. Feature Impact measures how much worse a regression model would perform if DataRobot made predictions after randomly shuffling that column (while leaving other columns unchanged). In DataRobot, it normalizes the scores so that the value of the most important column is 1. This technique is sometimes called Permutation Importance. As shown in **Figure 8**, our model includes a total of 18 features, which are DEWP, CO, NO2, DATETIME (Month), SO2, TEMP, DATETIME (Day of Month), DATETIME (Year), O3, (NO2) DIVIDE (CO), PRES, STATION, DATETIME (Day of Week), DATETIME (Hour of Day), WSPM, ZONE, WD\_Class, and RAIN, ranked by importance. The top five important variables among them are DEWP (dew point), CO, NO2, Month and SO2.



**Figure 8** Feature Impact of Best Regression Model

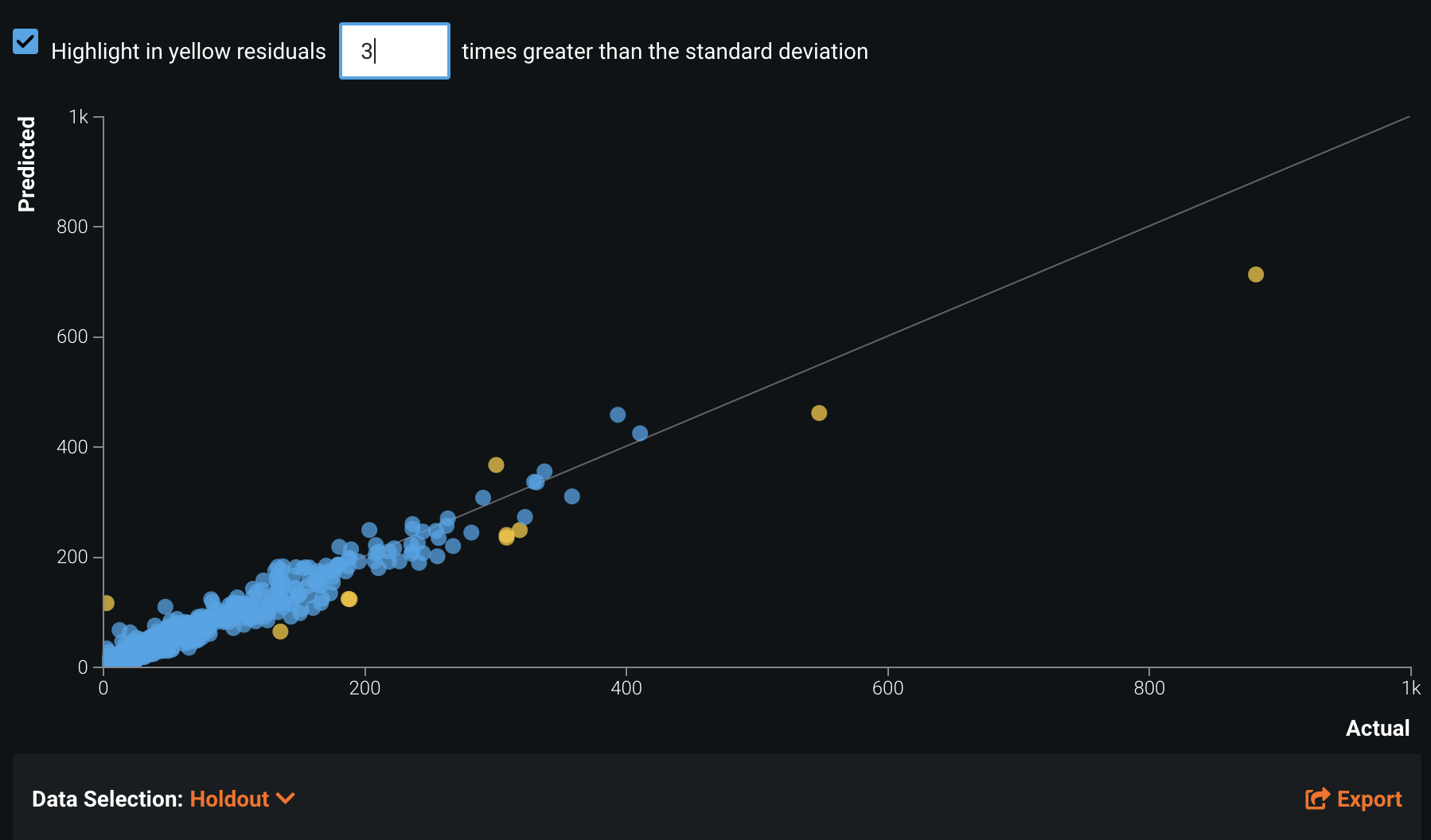
For model evaluation, we look at the same prediction accuracy measures, i.e. R-squared and RMSE. The DataRobot regression model has a comparatively high R-squared of 0.9291 and a low RMSE of 21.6342. Besides, DataRobot has generated a few charts that are useful for evaluating the model. **Figure 9** shows the Lift Chart. According to documentation, in general the steeper the line and the more closely the predicted line matches the actual line, the better the model. Also, a consistently increasing line is a good indicator. As the figure shows, our model on holdout data is well performed.



**Figure 9** Lift Chart of Best Regression Model

There seems to be no overfitting problem in this model, as the validation score of RMSE is 21.4662, which is almost the same as the holdout score of RMSE (21.6342). Furthermore, its R-square validation score is 0.9280 and its holdout score of R-square is very close (0.9291).

Besides, we have examined the residuals, i.e. the differences between predicted value and actual value, for Holdout data with a plot on DataRobot. As shown in **Figure 10**, the residual mean is -1.539, R-square is 0.9354 and Standard Deviation is 21.3134. We can see that actual data are well fitted to the prediction line. When we set 3 times the standard deviation as a boundary, residuals that are greater than 3 times standard deviation are regarded as outliers and marked yellow. We can clearly see most of the points are within or on the boundary that we made. Therefore, it seems that DataRobot has avoided the overfitting problem and does a very good job on the prediction.



**Figure 10** Residual Plot on Holdout Data

For model interpretation, we first viewed Prediction Explanations with calculated feature impact provided by DataRobot. Each explanation is a feature from the dataset and its corresponding value, accompanied by a qualitative indicator of the explanation’s strength—strong (+++), medium (++), or weak (+) positive or negative (-) influence. As **Figure 11** shown, we can only see 6 at most predictions and respective explanations. Although in the part of Explanation, it shows that the how impactful the features are for that specific prediction number, DataRobot does not show the exact coefficient to give us clearer sense about how these features perform to impact the predictions. Also, it does not give the information about overall importance of features to the targeted variable.



**Figure 11** Prediction Explanations of Best DataRobot Regression Model

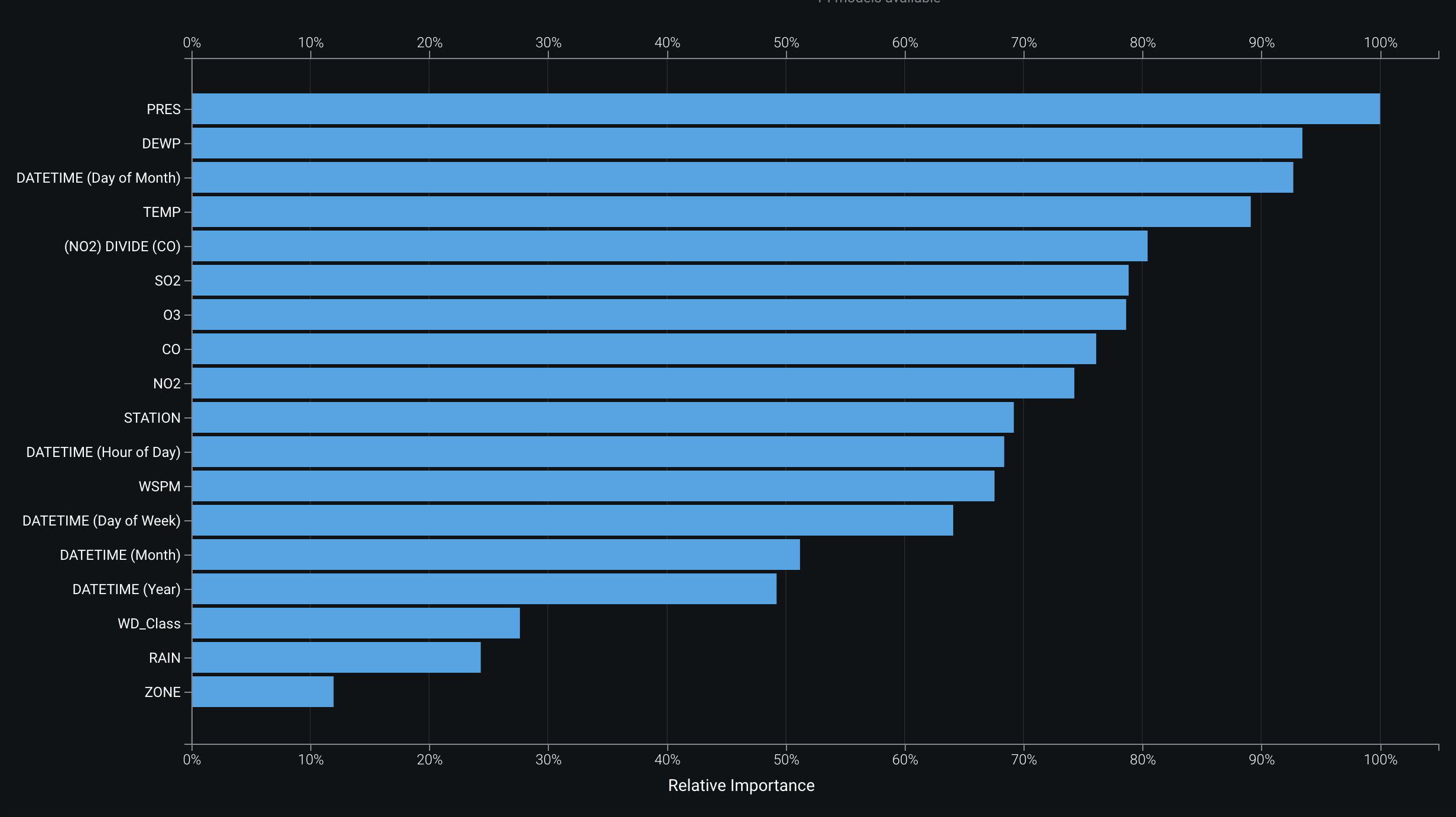
Therefore, we would rather use Feature Impact shown **Figure 8** to understand the relationship between our targeted variable and the features. The results suggest focusing on the value of dew point, concentration of CO, NO2 and SO2, month and temperature, day of month, year, O3, and NO2/CO when we want to predict the value of PM2.5 concentration.

## Model Comparison and Contrast

Our best-performing handcrafted regression model has an adjusted R-squared of 0.7451 and an RMSE of 31.97 on training data and an adjusted R-squared of 0.7261 and an RMSE of 42.27 on validation data. In comparison, the DataRobot best-performing model has an R-square of 0.9280 and an RMSE of 21.4662 on validation data, and almost same R-square and RMSE scores on holdout data of 0.9291 and 21.6342. Therefore, the prediction accuracy of the DataRobot model is way better than our handcrafted model.

One of the most important reasons of such a great difference in prediction accuracy is that we have not tried to utilize gradient boosted trees regression models on our own, not to mention the eXtreme version that improves the performance of greedy algorithm by reducing overfitting. The high prediction accuracy of the DataRobot model has shown the power of ensemble modeling mothods.

Besides the difference in overall performance, we have found very different results from the two models regarding the importance of variable. A little surprisingly, the DateRobot Model has ranked a few meteorological variables before the air pollutants variables. As show in **Figure** **12**, the top 5 important features by rank are PRES(pressure), DEWP (dew point), Day of Month, TEMP (temperature), (NO2) DIVIDE (CO). The difference might be due to normalization of feature values. In DataRobot, it normalizes the scores so that the value of the most important column is one. However, the multiple linear regression method we have used does not require data normalization. The differences in scales of variables could have affected their relative importance in prediction of the target variable.



**Figure 12** Relative Importance of DataRobot Regression Model Features

An advantage of our handcrafted model is that it generates coefficients for the predictor variables. The DataRobot model, in comparison, has used a method that cannot give coefficients of the variables. Coefficients give us insights about the relationships between the change in value of our target variable and the changes in predictor variable values. The DataRobot model does show much better performance overall. However, the feature importance is hard for us to interpret without knowing the coefficients.

As we can see, the relatively “Black Box” method of DataRobot has created some difficulties for interpretation of models. We have also found a puzzling difference of results shown in Feature Impact and Insights Figure regarding feature importance. As explained before, Feature Impact measures how much worse a regression model would perform if DataRobot made predictions after randomly shuffling that column (while leaving other columns unchanged). It seems that it is just another measure of the importance of features as Insights Figure. However, they have shown different results, probably due to the distinct calculation methods and definitions of importance. Therefore, we need to be careful with evaluation of variable importance using the results given by the platform, since it would affect our choice of predictor variables and thus the prediction result.

Despite the disadvantages, we think a fabulous point of DataRobot is that it has automatically offered the warning that alerts the redundancy in the variable. In our original dataset, PM2.5 and PM10 existed at the same time. However, these two variables are actually very similar, and both of them are the index of air quality. Therefore, it doesn't have much meaning to utilize PM10 to predict PM2.5, although it has very high accuracy. When we make our handcrafted model, we have deleted PM10 using our own domain knowledge. However, DataRobot has utilized its sensitive sensors to suggest deleting PM10 data as PM2.5 is the targeted variable because of issue of target leakage.

Furthermore, Datarobo also offers an option to include interaction terms in the model with advanced setting. As shown in our multiple linear regression modeling, interaction terms are usually useful variables that add accuracy to the prediction results. In the DataRobot model, an interaction term of NO2/CO is created. For our handcrafted model, we chose the important interaction terms through the exhaustive research approach. mer..

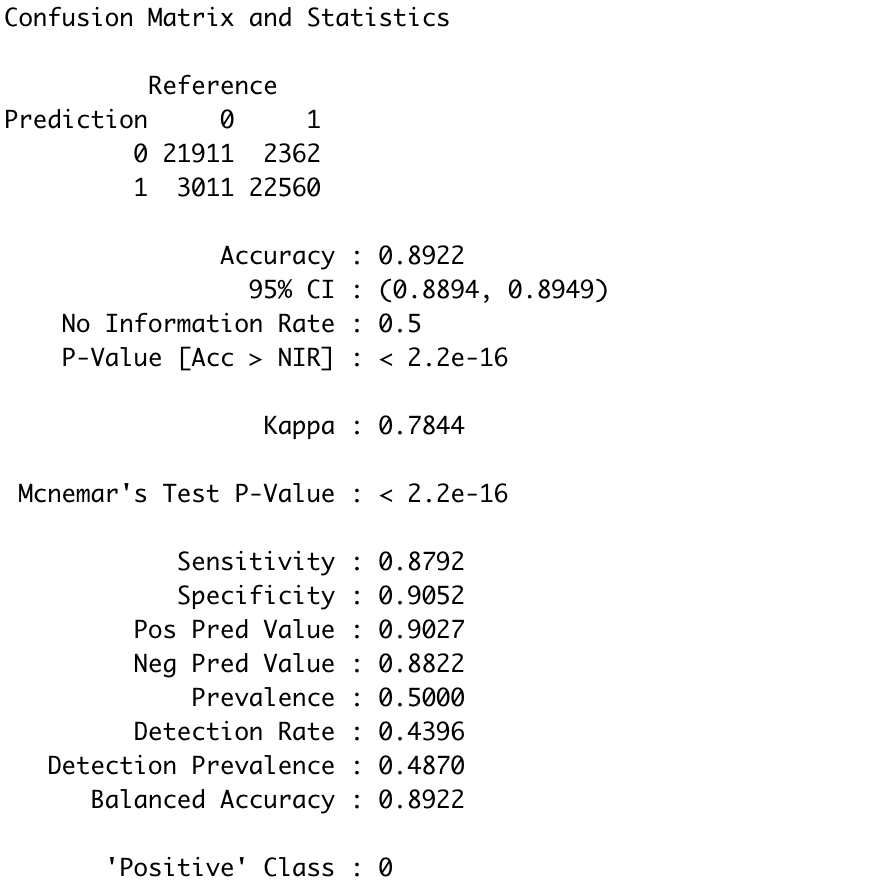
# Classification Models

Classification refers to identifying the class to which a new observation belongs based on a training dataset of observations whose classes are known. Besides prediction of PM2.5 concentration values, we would also like to test the effectiveness of machine learning approaches in classifying the level of PM2.5 concentration. We have built several classification models using R that predict whether the concentration of PM2.5 is above or below 150 μg/m3 for a particular day. In this section, we show the best classification model we have built and the best one built by DataRobot. We then compare and contrast these two models like before.

## Handcrafted Model: K-nearest Neighbors

We have built several classifications models with different machine learning methods, including logistic regression, K-nearest neighbors (KNN), and classification trees models. Among them, the best model is the one that uses the KNN algorithm.

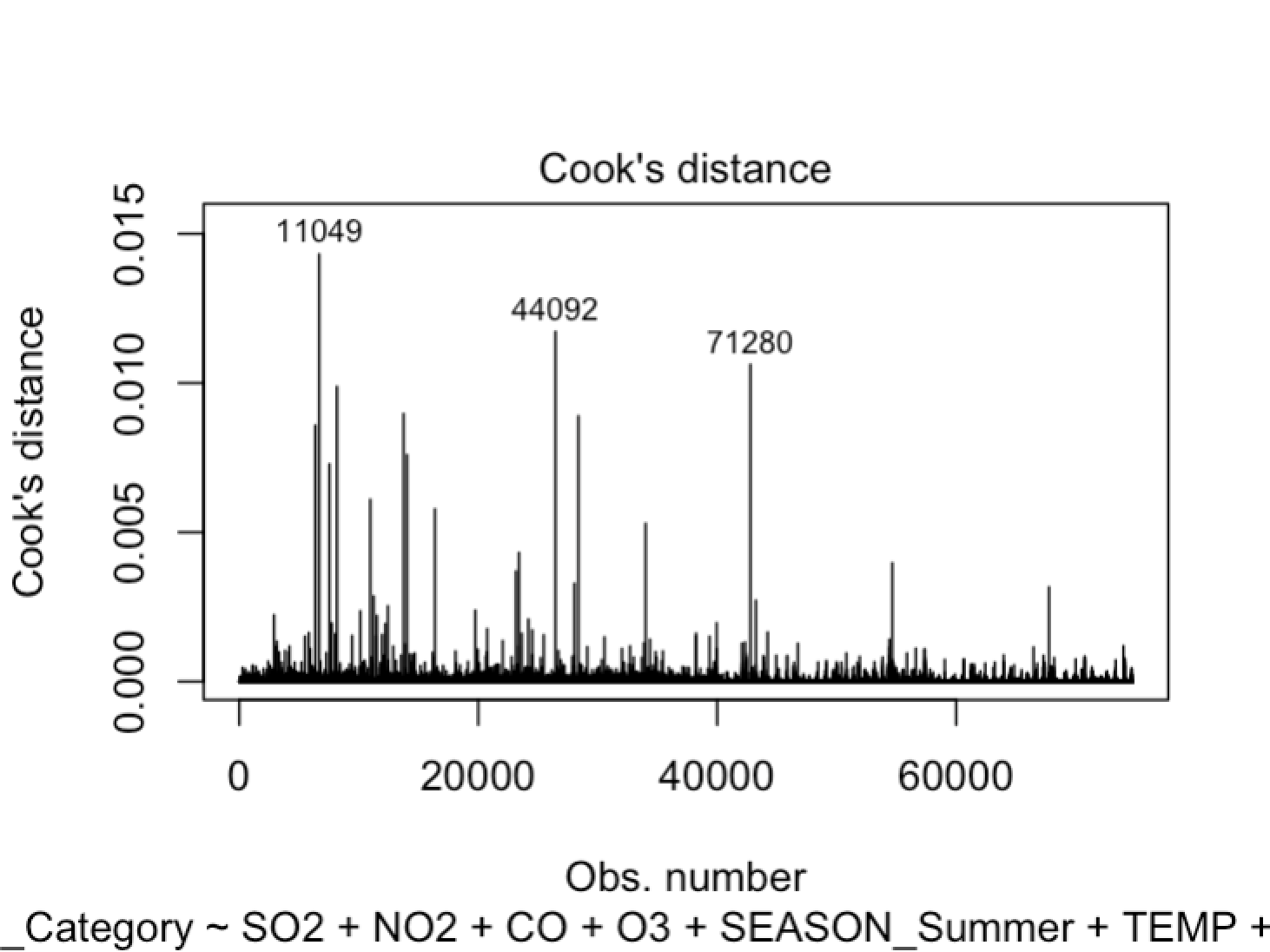
Our best k was chosen with an iteration method that compared the accuracy of models with various choices of k. The chosen k is 11, which provides the highest accuracy. The performance of our model is evaluated by Confusion Matrix and a few metrics generated from it. As shown in **Figure 13**, our best model has an accuracy of 0.8922, which was very impressive. Besides, the sensitivity, specificity, and F1 score is 0.879, 0.905, and 0.891 respectively.



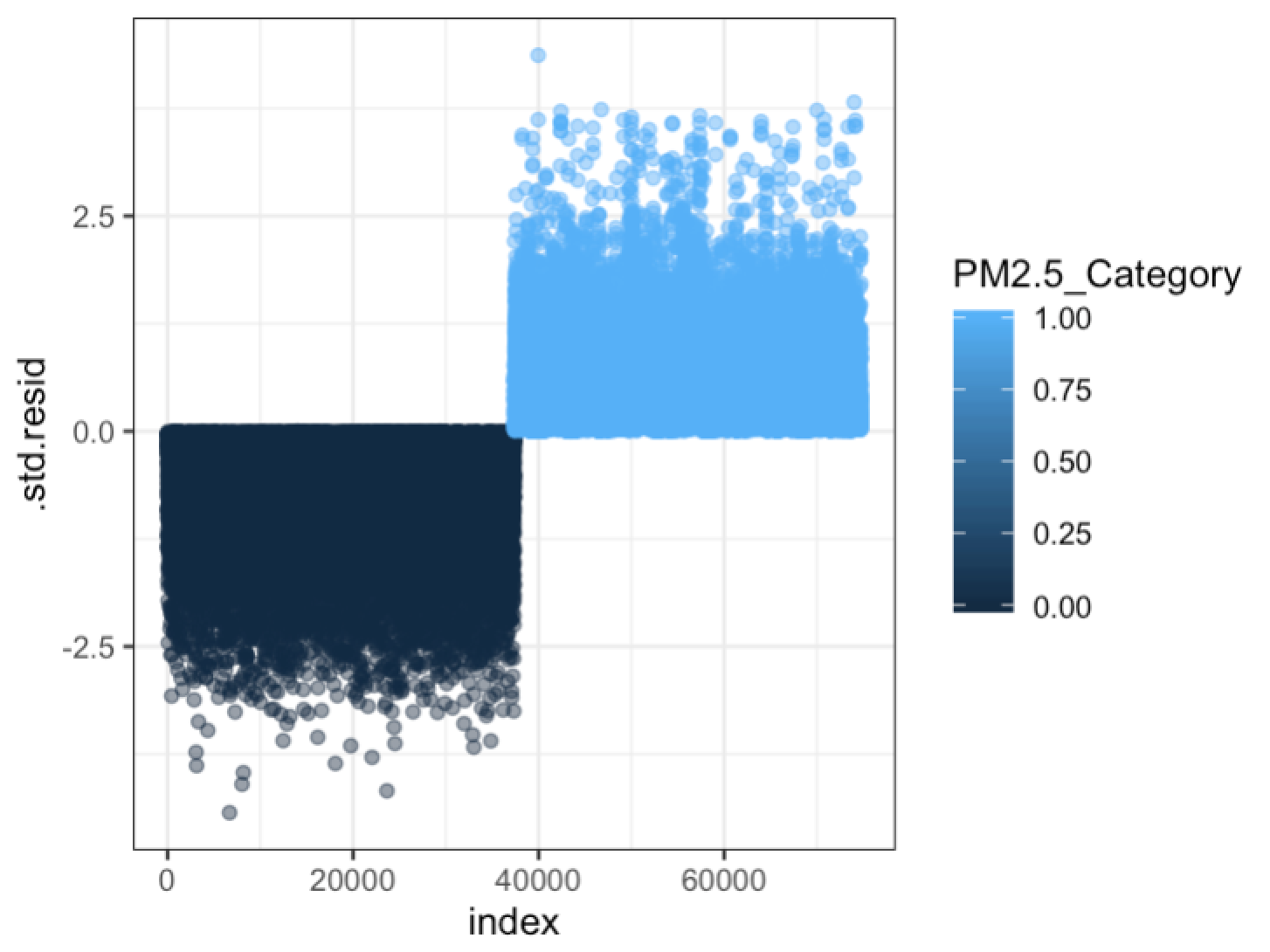
**Figure 13** Confusion Matrix and Statistics of Best Handcrafted Model

Before building the KNN model, we have normalized our variables using a simple equation: x - min(x) / max(x) - min(x). As the distances between pairs of samples are measured , which are influenced by the measurement units, normalization is an important step for building an accurate model.

Besides, as the performance of the KNN model can be affected by outliers. When we were building our classification models, we examined extreme individual data points with a visualization of the Cook’s distance and the standardized residual error (shown in **Figure 14** and **Figure 15**). After examination, data points with an absolute standardized residual above 3 have been filtered out as they represent possible outliers.



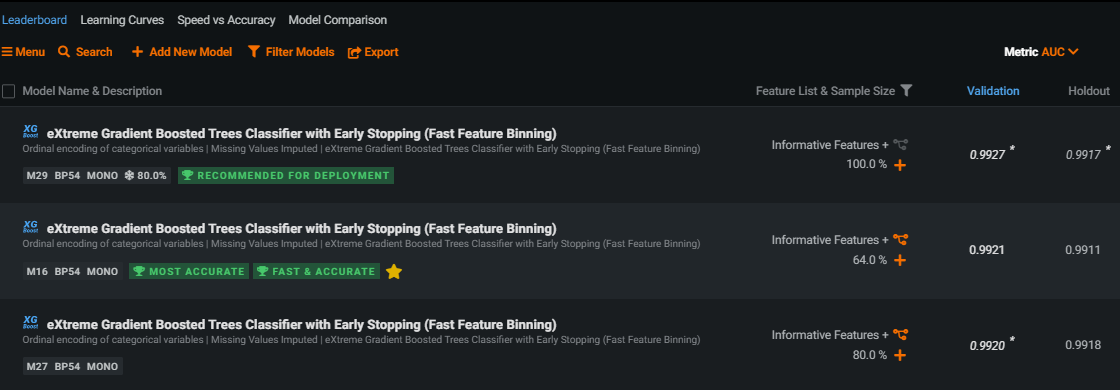
**Figure 14** Cook’s Distance



**Figure 15** Standardized Residual Error

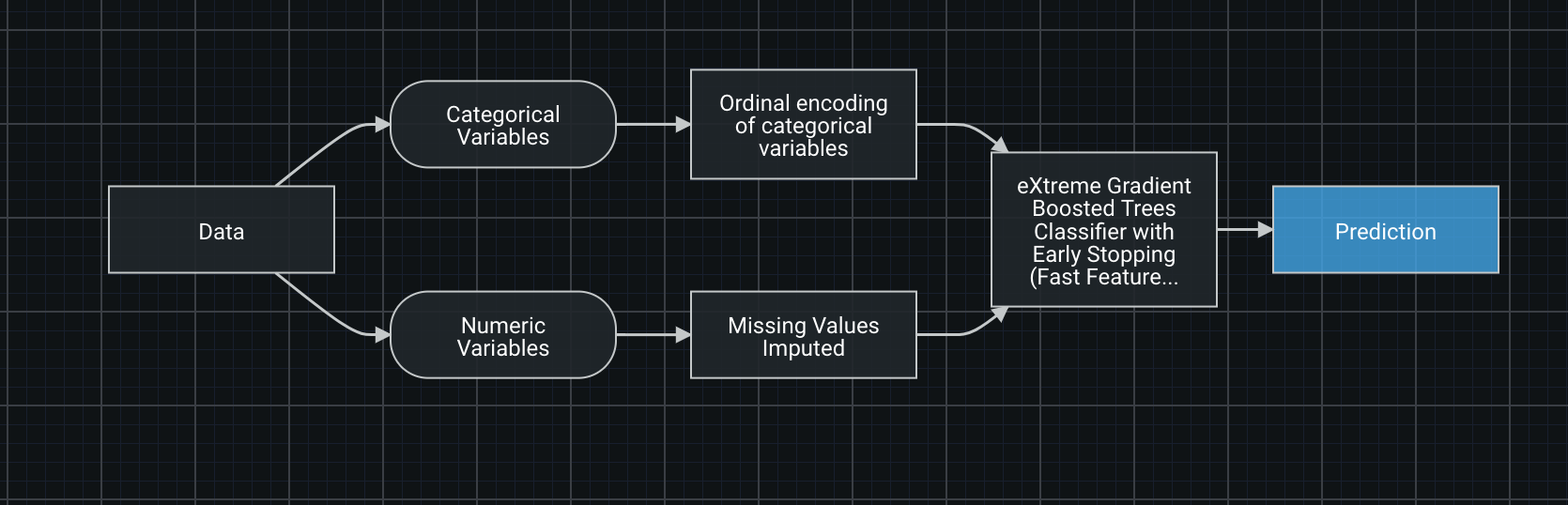
## DataRobot Model

From DataRobot, we have got a total of 20 classification models. The best-performing model is selected using the metric AUC (area under the curve). As AUC represents degree of separability, we have chosen the model with the largest AUC on the Holdout dataset. As shown in **Figure 16**, the best model is the eXtreme Gradient Boosted Trees Classifier with Early Stopping (Fast Feature Binning) model.



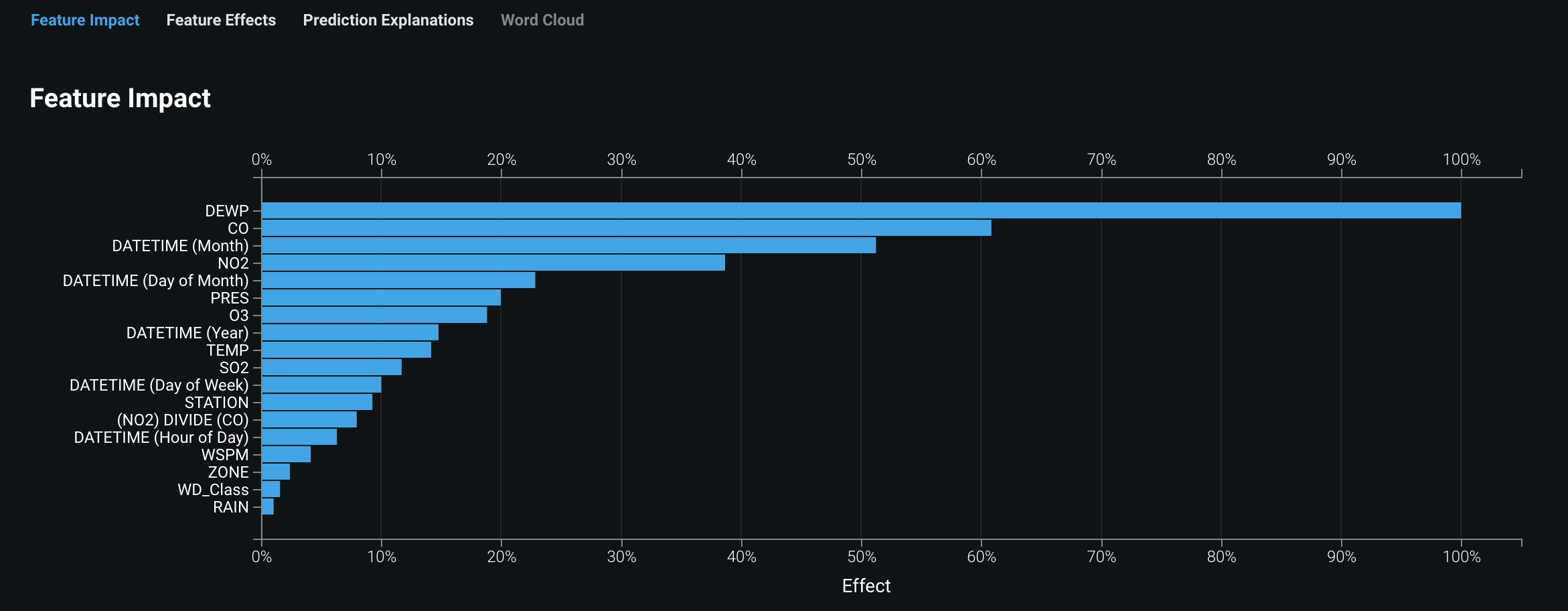
**Figure 16** Leaderboard of DataRobot Classification Models

The automated modeling process is shown by the platform in a blueprint as follows:



**Figure 17** Blueprint of Best DataRobot Regression Model

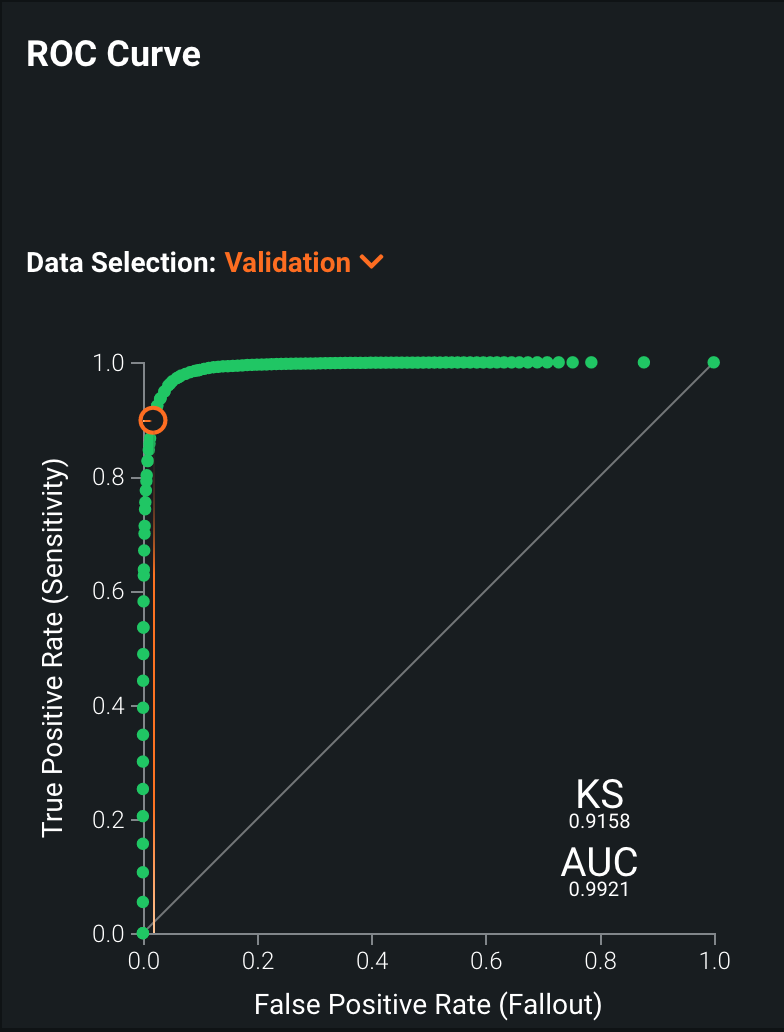
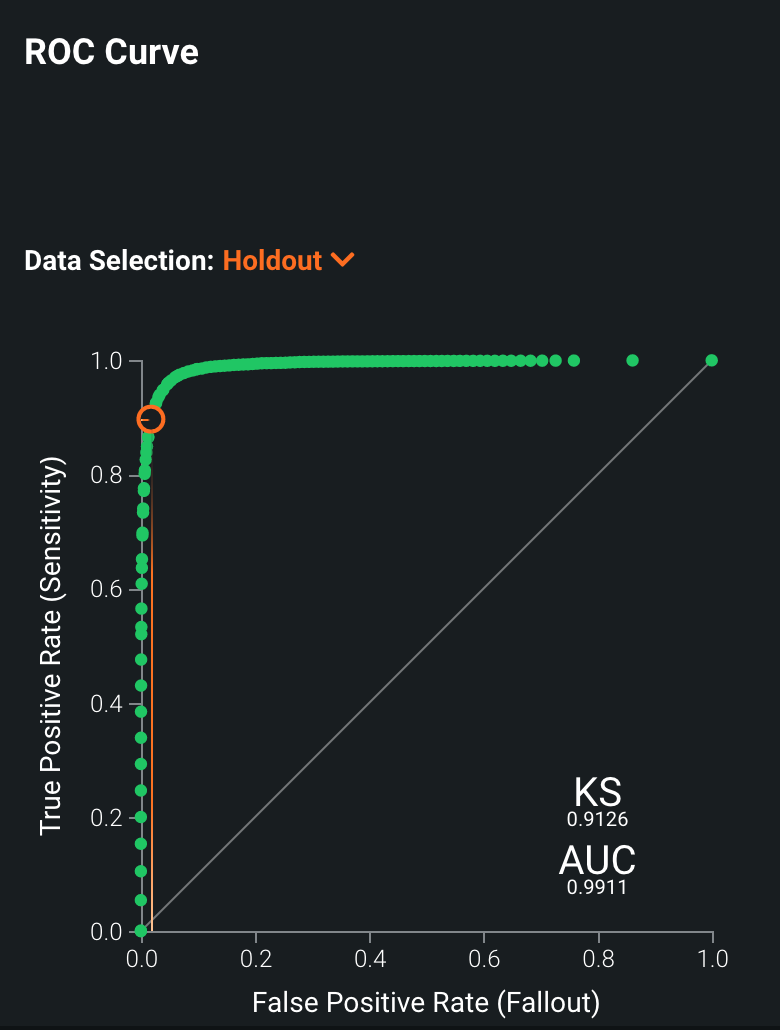
As before, we looked at the importance of these variables at the tab “Understand” (shown in **Figure 18**). The model includes a total of 18 features, which are DEWP, CO, DATETIME(Month), NO2, DATETIME(Day of Month), PRES, O3, DATETIME(Year), TEMP, SO2, DATETIME(Day of week), STATION, (NO2)DIVIDE(CO), DATETIME(Hour of Day), WSPM, ZONE, WD\_Class, and RAIN, ranked by feature impact.



**Figure 18** Relative Importance of DataRobot Classification Model Features

We also viewed details of features on the Data page by clicking the feature names. DataRobot run a calculation identifying the rows containing outliers after we clicked the Calculate outliers link. However, the outliers here are calculated by variables, which means we cannot find out easily how many outliers were omitted. Not to mention the methodology behind the identification of outliers.

The model does not seem to have an overfitting problem as the accuracy difference between validation set and holdout set is minimal (0.0002), and the gap between AUCs is only 0.001 (shown in **Figure 19**).

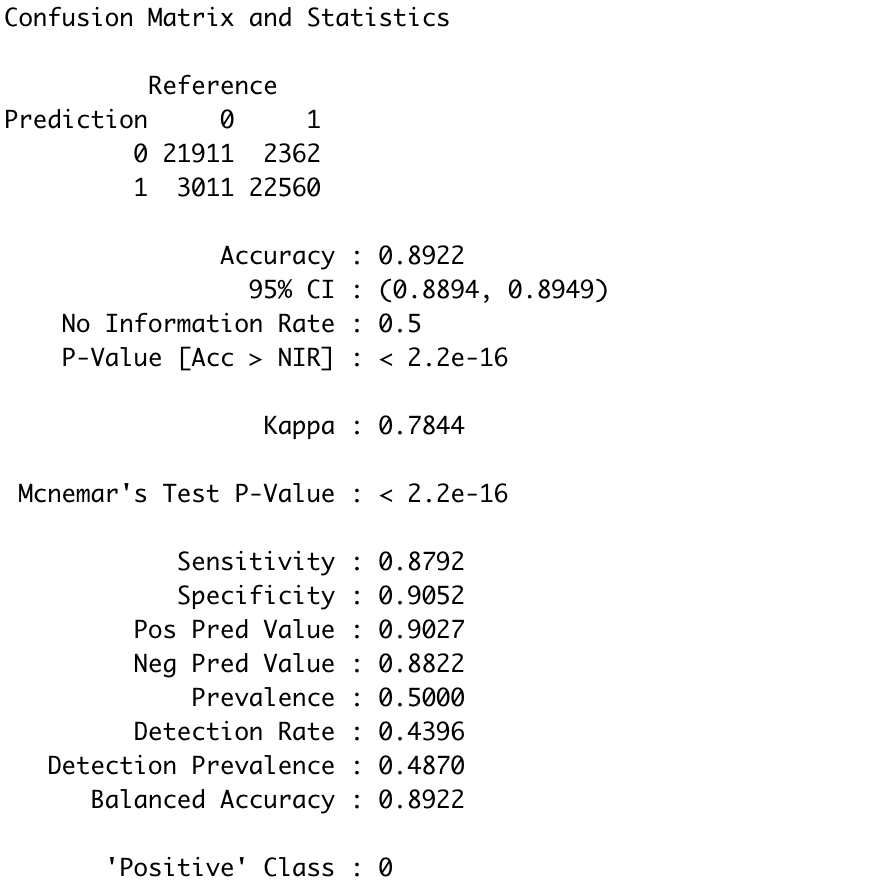
 

**Figure 19** ROC Curves and AUC

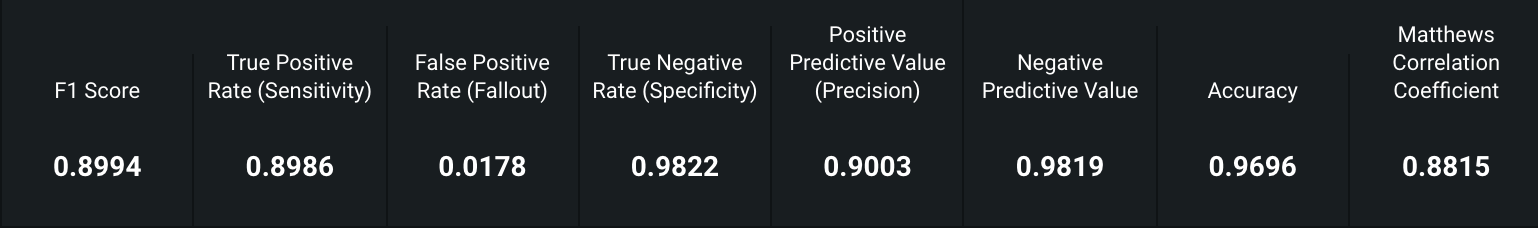
We further look at the result of the model at the “prediction explanation” subtab under the tab “Understand”. There are predictions as well as the criteria that the model used to make the prediction. As we can only see at most 6 predictions and its explanations, the explanation did not give us a very general view of the results.

## Model Comparison and Contrast

**Figure 20** shows evaluation metrics of our best model and DataRobot best model. Looking at the figure, we see that the accuracy of the DataRobot model is much better than ours.



Handcrafted Model



DataRobot Model

**Figure 20** Evaluation Metrics of Classification Models

One question, however, has been the elephant in the room once we take a closer look at the data robot model. That is the imbalance of the data. As we all know, accuracy alone is not a dependable factor, since Accuracy is derived from Sensitivity and Specificity. For a balanced data set with same amounts of positive and negative observations (the number of excellent candidates and poor performers are equal), accuracy is biased towards the higher value. For an unbalanced data set, the effect of either Sensitivity or Specificity on Accuracy will be erased accordingly. A good binary classification test always results with high values for all the three factors, Sensitivity, Specificity and Accuracy. However, we do not know the effect this imbalanced dataset had on the DataRobot model. It may somehow contribute to this model being able to outrun our handcrafted ones.

Regarding insights generated from two models, DataRobot model provides us some insights regarding variable importance that can be found easily under the “insight” tab. The histogram on the insight page could be a general depiction on the importance of each variable, measuring how much each variable is related to the target variable we are predicting. Though it's not showing the actual details of the metrics, it gives us an easy-to-interpret version of criteria. What surprised us is the variable "(NO2)DIVIDE(CO)" which is relatively important (92/100) compared to the most important variable. From the experience of regression it is hard to come up with different combinations of variables and it is a pain doing exhaustive research on these variables. The data robot seemed to have tackled this problem with ease, coming up with higher order variables that combined different original variables. The best handcrafted model we have built ( i.e. the KNN model), in comparison, generates almost zero insights.

# Conclusions and Reflections

## Reflections about the journey (process, roadblocks, pivoting, iterating)

Looking back at the projects we’ve completed with our Beijing Multi-Site Air-Quality dataset, this whole project seems to us like a process of searching for the best way to make predictions using machine learning methods following the guidance of Professor Arnold.

#### Data Wrangling

The raw data we are going to use is usually messy, inconsistent or even matched in real life. The problem slows down the whole data analysis process because a lot of time is needed on data cleaning and other preprocessing work rather than data analyzing itself. With our Beijing Multi-Site Air-Quality Dataset, we used Trifacta wrangler which can be leveraged across a variety of data platforms and makes the data cleaning, transforming and any pre-analyzing process much easier to pre-process our data. And this stage of data wrangle prepared us for further analyzing the whole dataset.

#### Regression

With our Beijing Multi-Site Air-Quality Dataset, we have constructed several multiple linear regression models on PM2.5. In our predictive modeling process, we have first examined the distribution and relationships of different variables with plots and summary statistics. This descriptive analysis on our dataset enables us to select our first set of regressions. Afterwards, in order to improve our model, we have used an iterative method to select a ‘best’ subset of regressors and also have dealt with the outliers in our dataset. With an exhaustive search approach, we have identified the interactive terms and polynomial terms that contribute to the prediction of our target variable. Throughout the process, our model performance on predicting values in the training set is continually improved as measured by adjusted R-squared and RMSE. Meanwhile, it is noteworthy that the model has a potential problem of overfitting as the predicting power on our validation set does not have an increase of the same magnitude, with the predicting power even becoming worse after excluding outliers. Therefore, it is important to partition our data in order to have a more reliable and valid assessment of our model.

To reflect on the regression process: from the first approach on exhaustive search and multiple regression, we got a model with the r-squared of 0.7. We tried to take intersection of variables into considerations, later we thought about adding higher-order variables, quadratic variables. Like its name, the process was exhausting, we knew the linear model cannot fit the data well, taking turns to try to find the ways to build multiple regression models was highly inefficient and the result is not so promising as well. More importantly, it's like trial and error: having the machine proving our hypothesis rather than letting the machine do the "learning" job.

Is there another way to classify the data better?

#### Classification

From the three classification models we built with three different methods—logistic regression (LR), KNN, and classification tree, we can see that they have similar performance measured with accuracy, sensitivity, specificity, and F1 scores. Nonetheless, during the modeling process, we find some advantages and disadvantages of each method compared with the others.

One problem we encountered is that it is difficult to compare two models with different Precision and Sensitivity, so to make them comparable, we use F-Score and reach the conclusion that the model based on the KNN method is the Best Model. Since our dataset is balanced, the specificity and sensitivity of each model returns the similar results, which means the accuracy is neither biased towards specificity or sensitivity. All of these three parameters are close to 1 which says the models are good. Therefore, we chose to focus on comparing the accuracy of three models and we have identified that KNN has performed the best based on accuracy.

#### Clustering

We then focused on clustering the data to optimise our regression prediction models.

In our data analysis, we first cluster our Air Quality data using hierarchical and k-means methods, and then perform multiple regression on our dataset, using PM2.5 as the target variable. By comparing different regression models that use different predictor variables (with and without cluster variables), we arrive at our best multiple regression.

For the clustering, we find that calculating the distance is a very important step. First, to avoid biased or inaccurate outcome and interpretation, we normalize our data. Also, we noticed that there are significant differences in clustering results when we use different methods to calculate the distance between clusters, we choose “ward.D”  to generate reasonable clusters as the results using other methods are either too hard to set a cutoff line or yield too many numbers of clusters.

For the regression, we do not see advantages of adding the cluster variables into the regression model. An alternative of utilizing the clustering results is to use the cluster variables as a substitution of a set of predictors and thus improve the parsimony of the model for classifying or predicting our target variable. However, it entails some amount of information loss, so that even with a smaller number of predictors the cluster variables are not chosen by the exhaustive search as a predictor in the best subset.

Regarding the potential problem of overfitting, we manage it using the data validation technique, which includes in-sample training and subsequently validation on a different set of data.  We partition our dataset into a training set and a validation set and pay attention to the error on the validation dataset, as an increase in the error is a sign of overfitting to the training dataset. In our case, the error of the validation dataset seems normal, and the technique enables us to have a more reliable and valid assessment of our model.

## overall conclutions, insights and surprises

After we have reviewed all of the models we have generated within this semester utilizing all kinds of fabulous tools from regression to clustering to analyze our Beijing Multi-Site Air-Quality dataset, DataRobot AI Engine is introduced to us this week. The DataRobot enterprise AI democratizes data science and automates the end-to-end process for building, deploying and maintaining AI at scale. It is powered by the latest open source algorithms. Therefore, we start a competition to compare the performance of our man-made Machine Learning and DataRobot-made Machine Learning tools.

Horizontally, we have compared the analysis result of our best performed handcrafted model to the best performed models generated by DataRobot, regarding regression and classification models respectively. The result is clear that the performance of both models built by DataRobot is much better than corresponding best models that were built by us.

Firstly, speaking of the overall performance, DataRobot-generated models have higher accuracy, higher R-square and lower RMSE than ours, which is understandable as we only have limited knowledge of the models. Therefore, the option of models that we are able to utilize is limited when we analyze the dataset. In contrast, DataRobot is powered by all of the latest open-source algorithms available online, which are mostly ensemble methods of machine learning. Furthermore, it is a system capable of applying our data into various models and ranking the performances of those models accordingly. Users can easily choose the best performed model automatically. Since they have a much more advanced algorithm and filtering system, DataRobot avoids a lot of potential problems that we have met in creating our own models. For example, in the regression model, our hand-crafted multiple linear regression model has utilized an exhaustive search method which would highly possibly cause overfitting. However, DataRobot best performed model -- eXtreme Gradient Boosted Trees Regressor with Early Stopping (Fast Feature Binning) highly avoids overfitting because of its advanced algorithm.

Regarding the predictors chosen which are based on importance of variables, the DataRobot-generated model has made different decisions from our model. One thing that surprises us is that,for our own model, we used the exhaustive search method to find the ‘best’ combination of predictors, including some interactive terms like Season\_Summer:DEWP, in order to achieve higher accuracy efficiently. However, the DataRobot-generated models achieved higher accuracy than ours without using our ‘interactive terms’. Among DataRobot-generated  models, it automatically creates one interaction predictor like NO2/CO and finally comes out with better performance. Therefore, it initiates our curiosity about how these algorithms generate and utilize interaction terms. Although DataRobot is able to offer all kinds of advanced analysis results based on its algorithm, it can’t show the exact coefficients like the linear regression model does. This would be hard for us to interpret how each predictor impacts the targeted variable at which exact level.

Overall, based on its higher accuracy and higher R-square, DataRobot gave us more convinced data and analysis results which we can learn from. Based on DataRobot analysis result, we would confidently suggest governments which are from less affluent countries to mainly utilize the data of PRES(pressure), DEWP (dew point), Day of Month, TEMP (temperature), (NO2) DIVIDE (CO), concentration of SO2, O3 and CO to predict PM2.5 and air pollution issue. We hope governments from all over the world will pay attention the problem of the air quality and use our analysis result to successfully predict PM2.5 by focusing on the important predictors that we have offered, and actively utilized the predicted result to work on the management of air quality in advance when PM2.5 is predicted to be high.

1. Reference: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6843796/> [↑](#footnote-ref-1)