Data Mining I - Practical Assigment Report

Prediction of Air Population

Beatriz Pinto , Filipe Justiça, Tiago Coelho
Faculty of Science University of Porto

12/20/2019

Abstract

This project provides a study of the data about Air Pollution in Beijing, China. We first analyse and describe the characteristics and relationships between the variables in the Beijing Multi-Site Air-Quality Data Data Set. Then we process the data and build different models to get predictions and review their performances. The model that had the best results was an ensemble of 7 binary classification models. The predictions for this model were 84.4% accuracy and 83.1% sucess in classifying bad air quality days.

1 Introduction

1.1 Context and Motivation

Air pollution is becoming a major concern for people living in city centers. The air pollution levels are constantly increasing due to transportation, industry, etc and turned into a big threat to public health. China is one of the countries that suffers more with this problem, specially in the bigger cities with higher population density. Just like you check for the UV light intensity before leaving the house, or going to the beach on hot summer days, it is crucial in these cases for people to know if they are save to go outside, whether they should wear masks or not, etc. With this in mind, a few years ago in China, there has been established the Air Quality Index (AQI). This is based on the level of five atmospheric pollutants (SO2, NO2, PM10, C0, 03), which can be measured in quite different ways. Given this variability and the difficulty in takingmeasurements of this kind, it would be better to have a model that could map other type of common observations like temperature and wind speed, and make predictions on the air quality expected in that day. Machine Learning has become very popular in the last few years for providing algorithms that can capture the knowledge from observed cases and make inferences in unseen cases, learning from experience. Which is exactly what a task like this requires. The purpose of this work is to try to learn from the data acquired in several weatherer forecast stations from the data set Beijing Multi-Site Air-Quality Data Data Set and try to build one or more Machine Learning model that is able to make predictions on future values based on the given information.

1.2 Strategy and Structure

As in any other Machine Learning problem that involves dealing with large amounts of data, we follow the traditional Data Mining pipeline. We start by exploring the context of the problem, gaining some domain knowledge and setting a plan and goals for the task ahead in **Problem Definition**. We then can begin preparing the data set to the analysis in **Data Pre-processing**. This means getting to know the type of variables we have, dealing with data quality issues, doing feature extraction and engineering,among others. In **Exploratory Data Analysis** we get a better insight of the structures and relationships present by using some data visualization tools. And finally, in **Predictive Models**, we describe the models we have built and performed best, as well as their results and the criteria used to classify them.

2 Problem Definition

Our data set is a multivariate time-series with measurements for 18 different meteorological and air-quality attributes, collected in 12 different nationally-controlled air-quality monitoring stations. This data belongs to the Beijing Municipal Environmental Monitoring Center and ranges from March 1st, 2013 to February 28th, 2017.

The attributes include quantities like temperature, wind speed, rain, etc, as well as the concentrations of 6 air pollutants taken hourly. These pollutants ($S0_2$, NO_2, CO, O_3, $PM10$, $PM2.5$) are the ones used by the center to directly calculate the **Air Quality Index (AQI)**. To determine the index, we first assign an individual score to each pollutant level for a given day **(IAQI)**, and the formula for this score can vary depending on the populatant, as mentioned in documentation[1]. The final daily AQI is the maximum value of those individual scores.

Given the big threat that air pollution presents to public health, the air quality index is very important not also to track trends and changes in air pollution, but also to inform and warn the population. However, the current way of calculating this index is a function of hourly measures of pollutants concentrations, and does not allow forecasting any future values.

Therefore, our task in this project is to get more understanding of the mechanisms that influence air quality. We aim to learn the relationships between air pollutants levels and the meteorological variables in the data set. The end goal is to map those variables into a model that gives good predictions of the air quality index for a given day.

Tipically the values for the index are then translated into categories according to a set of ranges that describe the air quality: {"Excellent", "Good", "Lightly Polluted", "Moderately Polluted", "Heavily Polluted", "Severely Polluted"}. Given this, we thought it would be more convinient for the generic user to have a prediction in the form of these kind of labels, instead of a number. So we turned the regression problem into a classification one.

3 Data Cleaning and Pre-Processing

The first step is to import the data set into our R environment in the appropriate format. We imported the necessary libraries, including *readr*, *dplyr*, *tidyimpute*, etc and the *csv* file with the data using the *read.table()* function. This way we can set arguments for things like keeping the headers of the table, the separation between values (","), the decimal separator character (".") and the string for *NA* values, and it automatically creates the desired data frame.

```
##   No year month day hour PM2.5 PM10 SO2 NO2  CO O3 TEMP   PRES  DEWP RAIN  wd WSPM station
## 1  1 2013     3   1    0     7    7   3   2 100 91 -2.3 1020.3 -20.7    0 WNW  3.1 Huairou
## 2  2 2013     3   1    1     4    4   3  NA 100 92 -2.7 1020.8 -20.5    0 NNW  1.5 Huairou
## 3  3 2013     3   1    2     4    4  NA  NA 100 91 -3.2 1020.6 -21.4    0  NW  1.8 Huairou
## 4  4 2013     3   1    3     3    3   3   2  NA NA -3.3 1021.3 -23.7    0 NNW  2.4 Huairou
## 5  5 2013     3   1    4     3    3   7  NA 300 86 -4.1 1022.1 -22.7    0 NNW  2.2 Huairou
## 6  6 2013     3   1    5     4    4   3   3 200 85 -4.2 1022.3 -24.5    0   N  4.3 Huairou
```

As we can see from the print of the table's top 5 row's we have 18 attributes, and only 2 of them are categorical, and some might have missing values. There are in total more than 35 thousand observations.

3.1 Data Quality Issues

After having the data frame with the data set we have to deal with any data quality issues. The first thing we do is identify and correct any missing values. Since we wish to make predictions on daily values, and our

observations are in hours, we do this with respect to groups of 24 observations. Running a cycle through every 24 entries for each day, we substitute any missing value by the mean of its two neighbors. If for instance, there is a variable with more than 15 missing values for one day, we replace each one by the moving average of the 6 closest non *NA* values (3 left and 3 right).

The function used to do this (*na_ma*) does not work with categorical variables, unless all of the observations have the same values (since we are computing means, and there is no mean for a group of different strings). In the data set we have, as mentioned before, two categorical variables: wind direction (wd) and station. The second has only one value (since we are only looking for observations of a single station - Huairou), and so did not raise any problem. But wind speed had many nominal values and we could not use this function for those. So, since we had already figure that this attribute had nearly zero correlation and was irrelevant for the air quality index, we decided to remove its column from the data frame. If necessary we can always add it again to the new cleaned data frame.

In addiction, we also checked if there were any duplicated data by looking for repeated rows and confirmed that there were none. And now our data is ready to be manipulated in data pre-processing.

3.2 Feature Extraction

We now have the filtered raw **hourly** data with the values for the pollutants and the meteorological attributes (as well as some other information). But what we want to predict are **daily** values for the Air Quality Index. Therefore we can use the expression[1] that gives its value for a given day. First we calculate the Individual Air Quality Index, I_{pi}, for each one of the pollutants, as said before, which is given by:

$$I_{pi} = \frac{I_{Hi} - I_{Lo}}{BP_{Hi} - BP_{Lo}} * (C_{pi} - BP_{Lo}) + I_{Lo}$$

Where C_{pi} is the rounded concentration of the pollutant pi, and can be the average concentration value of the day, or, in the case of CO and O_3, the maximum value registered. From there we have $\{BP_{Hi},\ BP_{Lo}\}$ as the breakpoints that are greater and smaller than the concentration C_{pi}, respectively. And $\{I_{Hi},\ I_{Lo}\}$ are the AQI value corresponding to those same breakpoints. The breakpoints for each of the pollutants are different and to make things easier we have created a csv table with them, as well as the correspondent AQI values. This table is available in the first section of the Appendix with the name *Aqi*. Finnaly, the total AQI for that day will be the maximum value of the individual AQI's of the pollutants:

$$AQI = max(I_{p1}, I_{p2}, I_{p3}, I_{p4}, I_{p5}, I_{p6})$$

With this we built a new data frame *vaqi* with the daily AQI values and a variable that indicates the maximal pollutant, which gave origin to the AQI value.

Those AQI values are our targets for the predictive models. We can make predictions with the meteorological attributes as input, giving the average values for each day as predictor variables. The target can either be the values of the AQI (regression model), or a class/label for the air quality in that day (classification model). The latter was considered to be more suitable for this task.

3.3 Feature Engineering

Now that we know the target variables and have some understanding of the problem, we can infer some features that migh be relevant in the prediction of the AQI. Eventhough we did not have much expert or domain knowledge, there were a few things we can get just from common sense.

We had already discard the wind's direction, since we considered it as an irrelevant feature for the task. If the direction was to influence the amount of pollution so would the speed and we already had that attribute. This is beacause speed and direction are usually correlated, for example in the Portugal the wind coming from the North is usally the strongest and fastest one.

The first new feature we tought of was the season. We wanted to see if there were any pollutants that woulds express themselves differently throughout the year. Seasons are a good way of capturing this as they compile the profiles of the meteorological variables together and let us see bigger dependencies. So we created this new attribute paying attention to the months and days of each observation and setting a label for each of the ranges with the names of the seasons.

Then, for easier interpretation of the values in data visualization, we also create a categorical variable for the AQI. This one simply reads the breakpoints and gives a label from "Excellent" to "Severely Polluted".

At last, we considered that the AQI from the previous days could have some correlation with the current day's AQI. So we also created 3 new variables $\{AQ_{1d},\ AQ_{2d},\ AQ_{3d}\}$ that give the values from the previous 3 days (this is done later in a more refined data frame, before training the models).

3.4 Data Transformation

Since we wish to predict the values for the air quality index, for a given day, using a set of daily meteorological variables, we need to do the same thing to the rest of the attributes and get their values for each day. To do this we determine the mean of each attribute for each day and create a new data frame only with daily values.

In our data set we deal with variables with very different domains and scales. With that in mind we thought necessary to normalize our data before training the models. This way we make sure that an attribute does not overshadow others just because its scale has much higher values.

The chosen methods to do this normalization were *center*, meaning that we subtract to all values the mean of the attribute, and *scale*, that divides the values by their standard deviation. Again, this is only done before training and compiling our models.

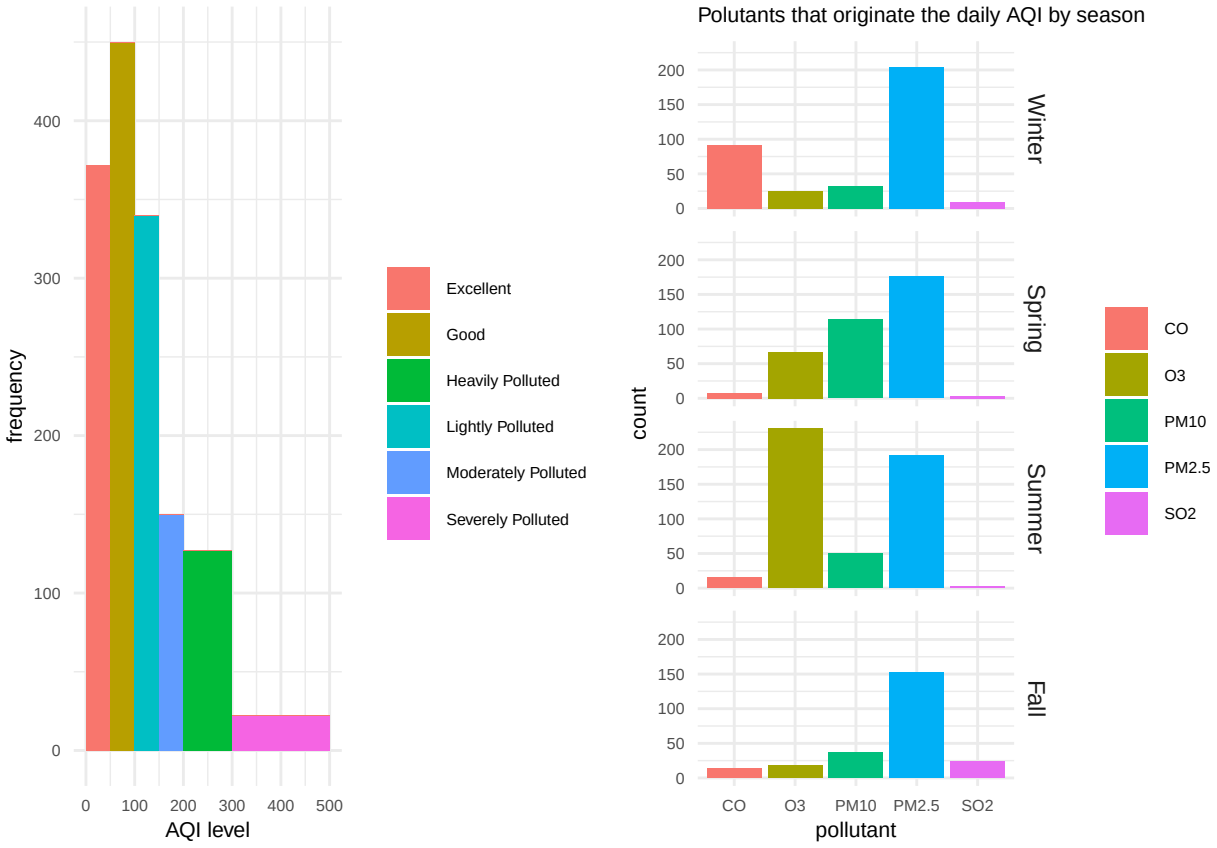
4 Exploratory Data Analysis

In this section we aim to get a better understanding of the variables we are dealing with. Data visualization help us see the patterns and relationships in data so we know what to expect from a model.

4.1 Data Visualization

We start by plotting a histogram with the frequencies of the AQI categories {"Excellent", "Good", "Lightly Polluted", "Moderately Polluted", "Heavily Polluted", "Severely Polluted" }.

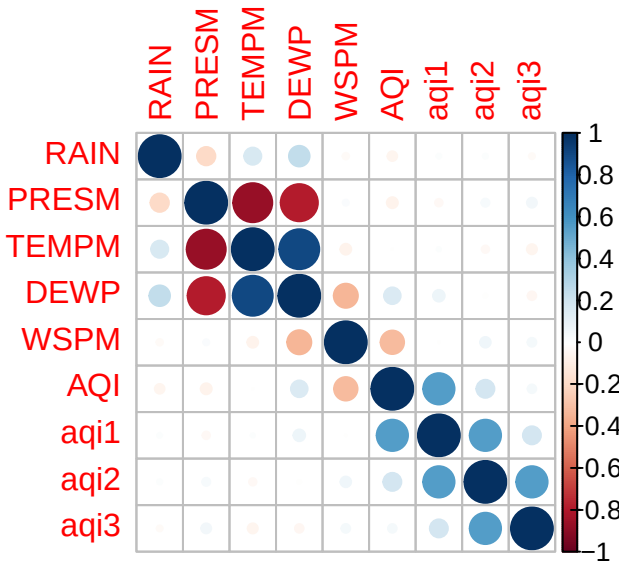
From here we can see that we get many more cases for the 3 less polluted labels, than for the 3 worst one. There are very few observations with *Heavily Polluted* and *Severely Polluted* AQI. This can be a problem when training the model and predicting because it will have much more cases of good air quality than bad and the models will perform better at classifying lower air quality index (better air quality). So we know from where that we will have to approach these when modeling by generating synthetic samples.



The plot on the right shows the number of times each of the pollutents was responsable for the AQI of the day grouped by seasons. The first thing we have noticed is that *PM2.5* is almost always the major pollutent and therefore the one that determines the AQI. That makes sense since we have seen in bibliography that *PM2.5* the main cause for visible air pollution (suspended particles in the air), and there have been many reports of *fog* in Beijing.

Another thing we can easily see is that the O_3 component is clearly higher in hotter seasons. This pollutent was the maximal one in the Summer and Spring, when the temperaure and pressure are higher.

Following this we decided to plot the correlations between the variables. Here we have already included the new attributes of the previous days AQI's that we though of using in the predictive models.




```
## NULL
```

We have done some other kinds of experiences with data visualization including for example principle component analysis. But we intentionally let those of the report since thay did not help us get to any objective conclusion. Nevertheless, some of these plots can be found in the section *Data Visualization* of the Appendix.

5 Predictive Models

As said before, we figure that a classification model would be more suitable for this problem. With this in mind the first step was to try some models of this type to predict for a given day, one of the 6 classes of the AQI (from "Excellent" to "Severely Polluted").

We focused mainly on models used in class such us *Naive Bayes*, *Classification Trees* and *Random Forest*. This envolves of course setting a seed, partitioning the data in training and test sets, if necessary doing one hot enconding, training the model and evaluating its performance. When looking at the models' performances in respect to their accuracies, we realized they were very low. The highest would be around 40%.

This was expected given the complexity of the problem, but we wanted to have a model that was good at telling people how the air quality will be in the following day or couple of days. So we decided to simplify the task and turn this into a binary classification problem.

Before moving on to the Binary Models, it is important to mention that we did not discarded the regression models right way. We still tryed them to see how they would perform. Since they did not do very well and we thought it would be more useful to have classes for the index anyway, we did not elaborate more on them and moved on to more complex classification binary models.

5.1 Binary Classification Models

We started with turning the 6 classes into 2, by joining the first 3 as a new "Good Air Quality" class, and the last 3 as "Bad Air Quality" class. We also did one hot enconding for the only categorical variable remaining, season.

We then partitioned the observations in the train and test sets with proportions (80%/20%), respectively. Following this, given the imbalance of the target variable (AQI) in the available observations, it was necessary to level the training set. This means making sure we have a 1:1 relation between the two air quality classes, so that the model learns equally well how to classify both. To do this we used the function *smote* from the package *DMwR*.

Before start training the models we run some configurantions and make sure to perform 10-fold cross validation. Then we trained the chosen models simultaneously using the packages *caret* and *caretEnsemble*.

The models used were *rpart2*, *svmLinear*, *svmPoly*, *rf*, *naive_bayes*, *knn* and *nnet*. Those were trained in 2 different ways: a first list of the models was trained to have the highest *accuracy*, and a second one had as end goal to give a better value for the metric *recall*. The *recall* represents how good the model is in classifying bad air quality days correctly. We did this because we identified in previous models and trainings the presence of false positives, meaning the classification of bad air quality days as good air quality days. And for this particular task that would present risks and decrease the usability of the model.

After training all these models, we trained two ensembles, each one being the linear combination of the 2 previous lists (one trained for the accuracy metric and the other for recall).

We also tried to create ensemble models using a smaller number of models by discarding the ones with worst performance and less correlation. But we have found that this actually decreased the performance and quality of the results. Therefore we left them out of this report and analysis.

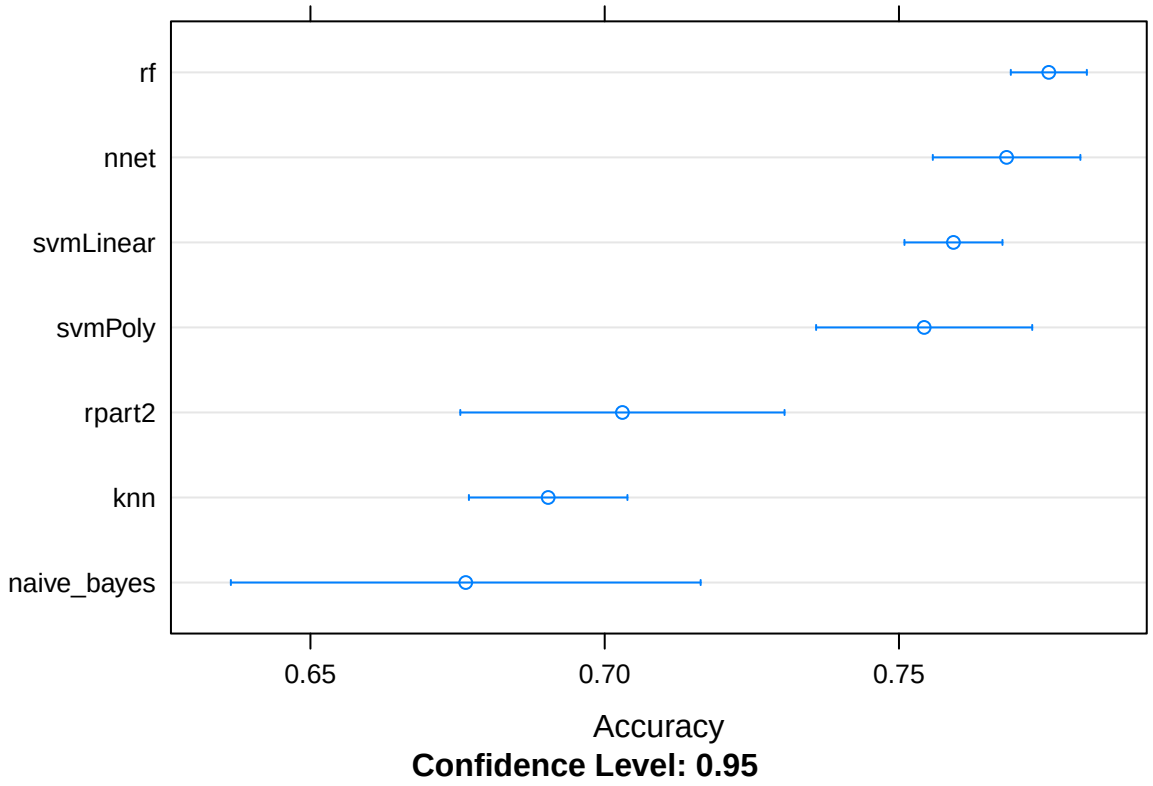
5.2 Results

Lastly, we made predictions for each model with the test set and analysed the results for each one in terms of accuracy and recall.

5.2.1 Models trained for Accuracy

The following plot shows the accuracy of the predictions for the 7 binary classification models trained to maximize that same metric. We can see that the *naive_bayes* performed the worst and also has the larger variance.

The best performing models look to be the *random forest (rt)*, the neural net (*nnet*) and the suport vector machine (*svmLinear*), with accuracies above 75% and smaller variances.

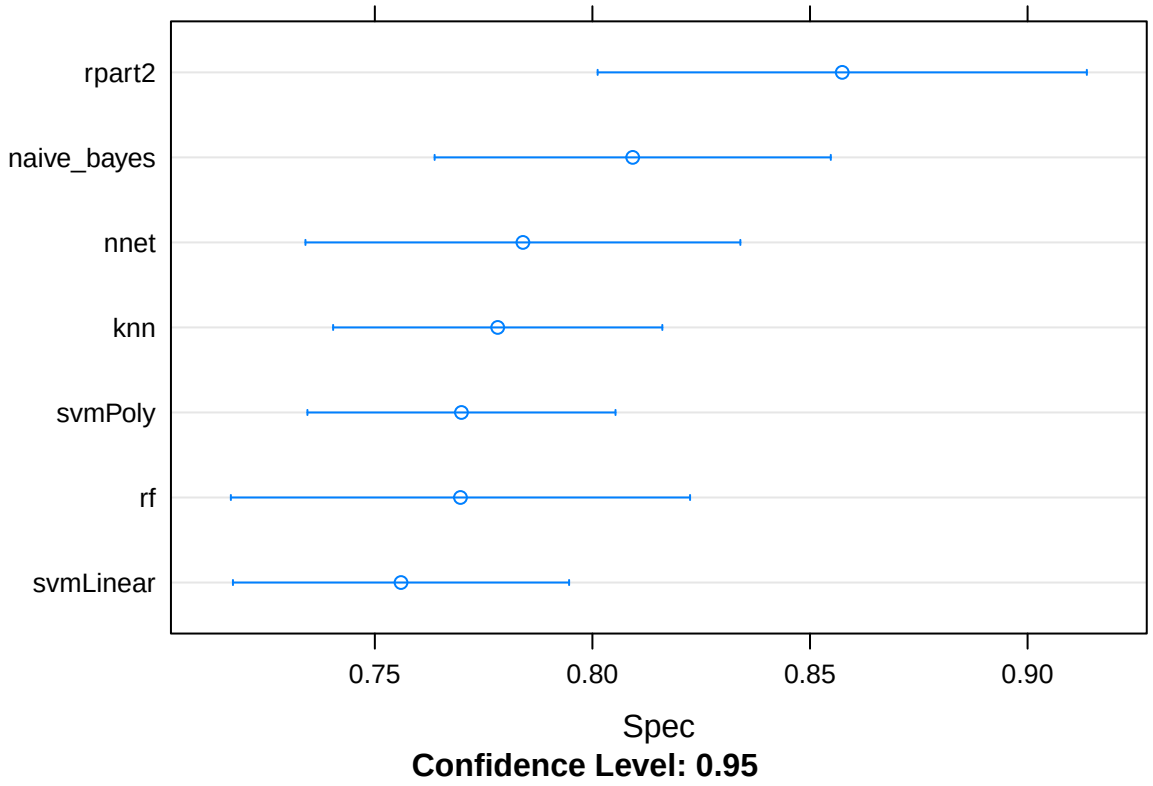


5.2.2 Models trained for Recall

In the following image we have the recall results for the models of the second list, trained to increase the performance of the model in classifying bad air quality days.

We can see that these predictions have larger confidence intervals, meaning larger variance. With the *rpart2* performing the best by far.

From the list of best performing models and comparing it from the previous one (considering the accuracy), we get the intuition that there is always a trade-off between accuracy and specificity (recall).



5.2.3 Model Comparisons

In the next figure we compiled the results of the predictions for both lists of models and ensembles considering accuracy and recall, so we can easily compare them.

As expected, globally the models trained for accuracy gives more accurate predictions than the models trained for recall, but the reverse is not true.

Both ensembles are good in terms of giving the best metric possible (there is no model with better accuracy/recall than the ensemble all together).

```
##   ensemble_1       KNN SVM_Linear SVM_Poly      NNET NAIVE_BAYES        RF     RPART
## 1  0.8448276 0.7586207  0.8034483      0.8 0.7896552   0.5896552 0.7965517 0.7482759

##   ensemble_1       KNN SVM_Linear  SVM_Poly      NNET NAIVE_BAYES       RF     RPART
## 1  0.8305085 0.7966102  0.7627119 0.7966102 0.7966102   0.8644068 0.779661 0.7457627

##   ensemble_2       KNN SVM_Linear  SVM_Poly      NNET NAIVE_BAYES        RF     RPART
## 1  0.8137931 0.7586207  0.8034483 0.8103448 0.7862069   0.6517241 0.7931034 0.6448276

##   ensemble_2       KNN SVM_Linear SVM_Poly     NNET NAIVE_BAYES        RF     RPART
## 1   0.779661 0.7966102  0.7627119 0.779661 0.779661   0.8474576 0.7627119 0.8644068
```

We conclude that the best model to get more accurate predictions is *ensemble_1*. And the best model to get fewer false positives is also *emsemble_1*. So globally this is the best model with 84.4% accuracy and recall 83.1%.

Conclusions

Our goal was finding a Machine Learning model that was able to make predictions on future values of pollution level based on the given information, that would tell what people need to know about pollution daily (if it is going to be polluted or not) and at the same time that wouldn't predict days without pollution when the day was actually polluted.

In the end we reached that goal by making a linear combination of some models we tested, because some of them were better in accuracy and others in not predicting false non polluted days. In fact, we trained some of the models to have better accuracy and others to prevent predictions of false not polluted days. We found out that the best model was the linear combination of the models that were trained to get the best accuracy, since it is the one that predicts the pollution of the days better and the second best at giving false not polluted days. This models gave predictions with 84.4% accuracy and 83.1% sucess in classifying bad air quality days.

This analysis was done for only one of the 12 stations in Beijing, so it would be interesting in the future to complete our study and predictions with spatial data of them all together.

We could also enhance our model and try to predict with 3 classes of air quality ("Good", "Ok", "Bad"), instead of only 2, but that would probably give worse performances.

References

[1] Fanyu Gao, 2013, Evaluation of the Chinese New Air Quality Index (GB3095-2012): Based on Comparison with the US AQI System and the WHO AQGs