CREATING A MODEL WHICH ACCURATELY PREDICTS AND MINIMIZE(THE COST OF) FAILURES IN APS SCANIA TRUCKS

Machine Learning Hackathon Project
18th November 2018
Submitted by

Shyam Kumar V N
shyam.vn@iiitb.org
MS2018017

Bosco Sebastian
bosco.sebastian@iiitb.org
PH2018007

S, Arunkumar
Arunkumar.sivasubramaniam@iitb.org
PH2018005

1. Contents

1. Problem statement

The dataset consists of data collected from heavy Scania trucks in everyday usage. The system in focus is the Air Pressure system (APS) which generates pressurized air that are utilized in various functions in a truck, such as braking and gear changes. The dataset's positive class consists of component failures for a specific component of the APS system. The negative class consists of trucks with failures for components not related to the APS. The data consists of a subset of all available data, selected by experts.

The goal of the project is to minimize maintenance cost of the Air Pressure System of the Scania Trucks. This is done by predicting the failure of APS from the data provided.

From the data set repo's description, our goal is to minimize the costs associated with:

- Unnecessary checks done by a mechanic. ($10)
- Missing a faulty truck, which may cause a breakdown in the future. ($500)

There for the total cost is calculated by:

Total cost = Cost_10 * No. of false predictions + 500 * No. of failed predictions

2. Dataset

This dataset was released by Scania CV AB on the UCI Machine Learning Repository as part of the Industrial Challenge 2016 at The 15th International Symposium on Intelligent Data Analysis (IDA) in 2016. The challenge was about predicting failure of Scania's APS (Air Pressure System) in trucks to enable preventive maintenance and therefore reduce costs. The dataset is anonymized and contains binned/encoded values due to proprietary reasons.

There are 171 attributes in the data set. The attributes are anonymized by proprietary reasons other than the class feature. The dataset's positive class consists of component failures for a specific component of the APS system. The negative class consists of trucks with failures for components not related to the APS.

3. Data Understanding / Analysis of dataset

- Number of Instances: The training set contains 60000 examples in total in which 59000 belong to the negative class and 1000 positive class. The test set contains 16000 examples.
- Number of Attributes: 171
- Attribute Information: The attribute names of the data have been anonymized for proprietary reasons. It consists of both single numerical counters and histograms consisting of bins with different conditions. Typically, the histograms have open-ended conditions at each end.

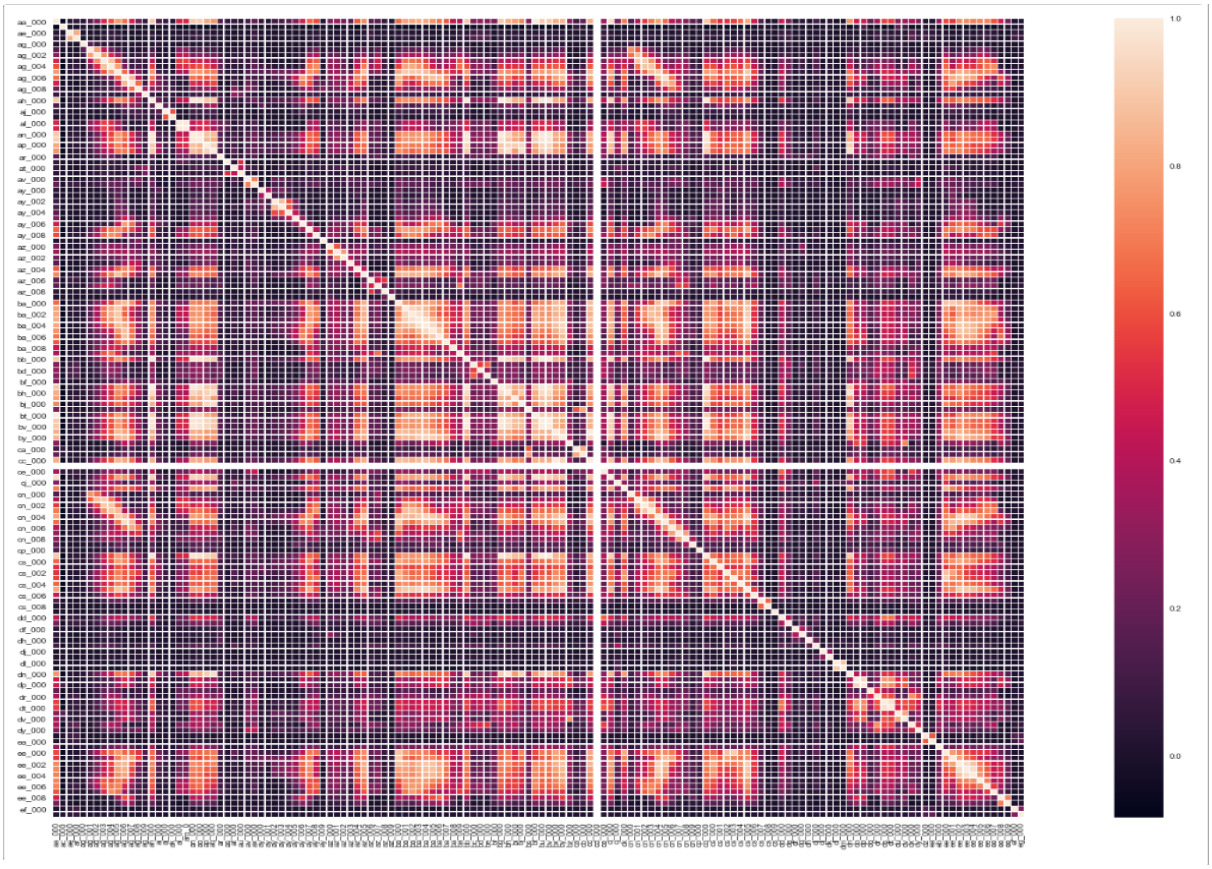
From the below figure, we observed that every column in this dataset has missing or NULL values.

```
In [70]: #checking for null values in the training data.
         train_data.isnull().sum().values

Out[70]: array([    0,     0, 46329,  3335, 14861,  2500,  2500,   671,   671,
                  671,   671,   671,   671,   671,   671,   671,   671,   645,
                  629,   629,  4400,   642,   629,   642,   589,   642,   589,
                 2723,   629,   629,   629,  2500,  2501,   671,   671,   671,
                  671,   671,   671,   671,   671,   671,   671,   671,   671,
                  671,   671,   671,   671,   671,   671,   671,   671,   688,
                  688,   688,   688,   688,   688,   688,   688,   688,   688,
                  645,  2725,  2727,  2503,  2500,   642,   642,   589,   589,
                23034, 27277, 39549, 44009, 46333, 47740, 48722, 49264,   726,
                  167,   691,   691,  3257,   473,  2723,  4356,   726,  3255,
                  676,  2502, 14861, 14861, 14861,   338,   338,   338,  9553,
                 9877,   687,   687,   687,   687,   687,   687,   687,   687,
                  687,   687, 14861,  2724,   691, 46329,   669,   669,   669,
                  669,   669,   669,   669,   669,   669, 13808, 13808,
                13808, 13808, 13808, 13808, 13808, 13808,  2503,  2724,
                 4008,  4008,  4008,  4006,  4007,  4007,  4008,  4009,   691,
                 2724,  2726,  2726,  2726,  2727,  2727,  2726,  2726,  2723,
                 2724,  2723,  2723,  4007, 10239,  9553,   671,   671,   671,
                  671,   671,   671,   671,   671,  2724,  2723],
               dtype=int64)
```

The dataset contains up to 82% missing values per attribute.

The correlation map for the attributes is given below: We can see some blocks of highly correlated features, we will try using cross-validation grid search to see if removing highly correlated features helps.



Furthermore, many of the attributes contain outliers.

4. Feature Engineering

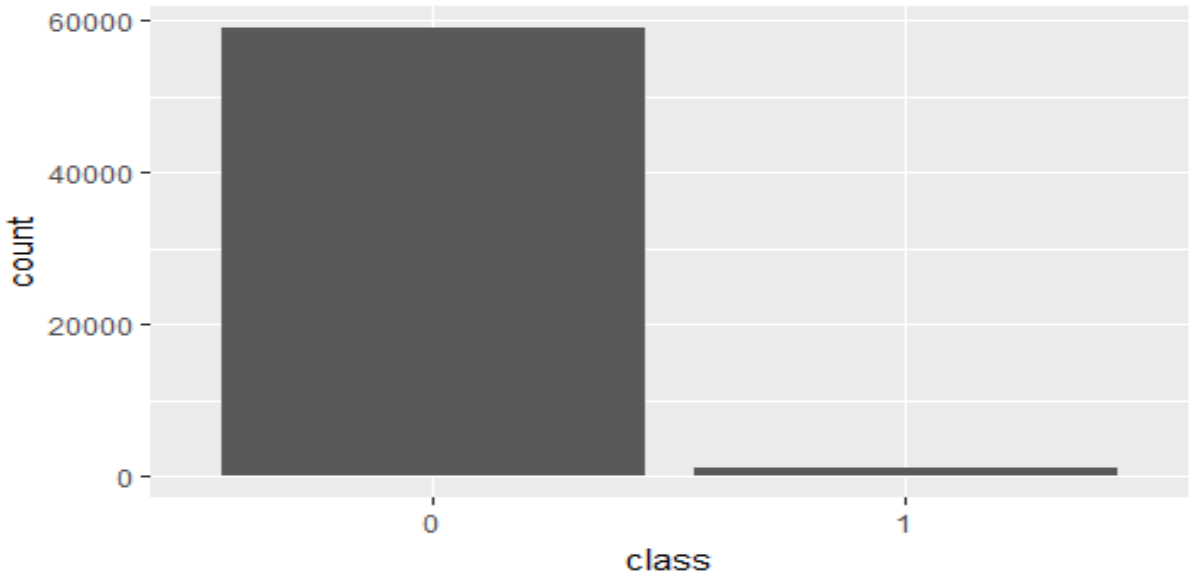
The first step in data-based modeling is to review the data for missing values and errors. This particular data had some attributes missing up to 82% of values. This could be due to lack of time to measure all attributes in between operations. This also meant that attributes with such high amounts of missing data would be the ones ignored by employees as not critical. As a result, eight attributes that have more than 50% missing values have been removed and We are dropping columns with more number of null values. If the ratio of null values to row number is less than 10 percent, the column is dropped from the final feature list.

Then we are checking for the correlation of each column with the target. If there is no correlation between result and the target, we are removing the column from the final list of columns.

The data contains more than 90 percent of negative class, which prevent model from learning necessary information for the minority class and making an accurate prediction. Hence, we used SMOTE function from imblearn library. It identifies the minority class in the sample data and randomly generates data for that class such that in the end both the classes exist in equal proportion.

The set is very unbalanced with one label (0) being more frequent than the other (1). The algorithm needs to adjust for that. It is done using 'class_weight' hyperparameter which is the ratio of number of 0s to 1s in the label.

The histogram below reveals a significant unbalanced pattern between the positive and negative classes.



Also we had quite a few features over 70% null ratio. So we integrated by dropping of high null ratio features into cross-validation grid.

5. Visualization

The data has a lot of features, because of that, is very difficult to visualize hierarchical graphs.

6. Modeling

For other missing values, median imputation was used. After evaluating several classifiers including Gaussian Naive Bayes Classification, Decision Tree Classification, Random Forest Classification, and AdaBoost Classification, XGBoost Classification, we observed that Random Forest was giving better result. Since we were using Random Forest as there is no need for normalization and is extremely good at finding which feature is correlated to the next.

Also given the high dimensional dataset, we chose a combination of feature engineering and feature reduction whilst constantly evaluating the results using a Random Forest and we used throughout the challenge
We integrated the following functions into cross-validation grid search to find the best model:

1. Pre-processing (dropping of high null ratio features, imputation, etc)
2. Feature engineering (creating new features by marking null values, etc)

The reason for putting everything in the cross-validation grid search is that we could collectively optimize all the parameters, e.g, what feature null ratio before dropping feature, what imputation strategy, etc.

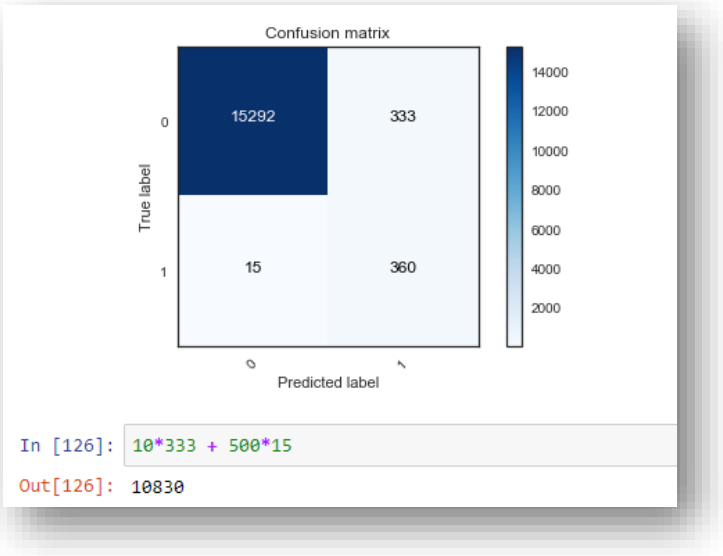
7. Training

Test data.

- n_test 16,000
  - n_neg_test 15,625 97.7%
  - n_pos_test 375 2.3%

8. Conclusion

An early detection of a failure in an Air Pressure System in trucks can save the company a lot money. We showed how the forecasts can be adapted to a cost function using a threshold on the confidence of a Random Forest.



With the custom threshold of 0.19, we brought the cost to 10830. We could significantly lower the maintenance cost.

9. Relevant Papers

- Costa C.F., Nascimento M.A. (2016) IDA 2016 Industrial Challenge: Using Machine Learning for Predicting Failures. In: BostrÃ¶m H., Knobbe A., Soares C., Papapetrou P. (eds) Advances in Intelligent Data Analysis XV. IDA 2016. Lecture Notes in Computer Science, vol 9897. Springer, Cham
- Gondek C., Hafner D., Sampson O.R. (2016) Prediction of Failures in the Air Pressure System of Scania Trucks Using a Random Forest and Feature Engineering. In: BostrÃ¶m H., Knobbe A., Soares C., Papapetrou P. (eds) Advances in Intelligent Data Analysis XV. IDA 2016. Lecture Notes in Computer Science, vol 9897. Springer, Cham
- Cerqueira V., Pinto F., SÃ¡ C., Soares C. (2016) Combining Boosted Trees with Metafeature Engineering for Predictive Maintenance. In: BostrÃ¶m H., Knobbe A., Soares C., Papapetrou P. (eds) Advances in Intelligent Data Analysis XV. IDA 2016. Lecture Notes in Computer Science, vol 9897. Springer, Cham
- Ozan E.C., Riabchenko E., Kiranyaz S., Gabbouj M. (2016) An Optimized k-NN Approach for Classification on Imbalanced Datasets with Missing Data. In: BostrÃ¶m H., Knobbe A., Soares C., Papapetrou P. (eds) Advances in Intelligent Data Analysis XV. IDA 2016. Lecture Notes in Computer Science, vol 9897. Springer, Cham