## Problem statement

In the trucking industry, maintenance, breakdown, and safety are a few of the major concerns. There are preventive, routine maintenance to confirm the current state of equipment in the truck and avoid any breakdowns. And there are expenses after the breakdown (due to a missed routine maintenance) to get the truck back on the road. Unnecessary preventive checks cost money and add burden to the company, and on the other hand, a missed maintenance may cause a breakdown and cost the company even more. There is a need to predict the maintenance timeline to minimize both types of expenses.

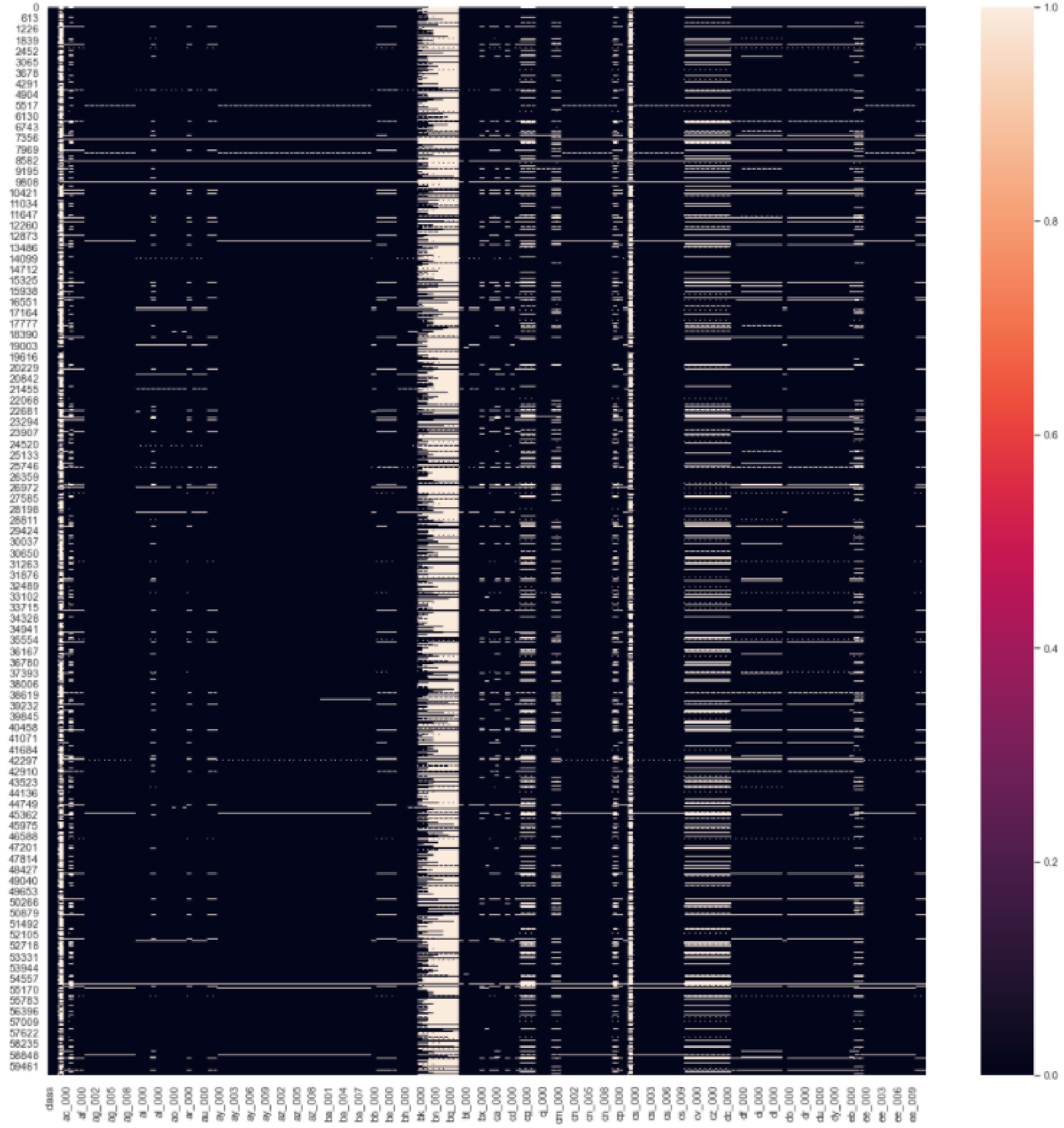
## Proposal

There are many different parts in a truck that require maintenance. In this case study, we will focus on Air pressure system (APS). The APS generates pressurized air to be used in multiple functions of the truck operations while on the move. APS should be inspected routinely to confirm its quality and needed parts replacement. A missed checkup may cause even more to fix. The dataset for APS failure at Scania trucks (https://archive.ics.uci.edu/ml/datasets/APS+Failure+at+Scania+Trucks) has 60000 observations of 171 attributes including “class”. The class can have one of the two possible values negative and positive. Negative represents the failure due to a non-APS component and Positive represents the component failure due to APS component failure. This is a supervised learning problem and I will map the negative class to 0 and positive to 1. All other columns are anonymized by the publisher due to copyrights and confidentiality. This should not be an issue for this project as we only predict and minimize the cost of failures with the combinations of these attributes. I will use the logistics regression model and the random forest classifiers.

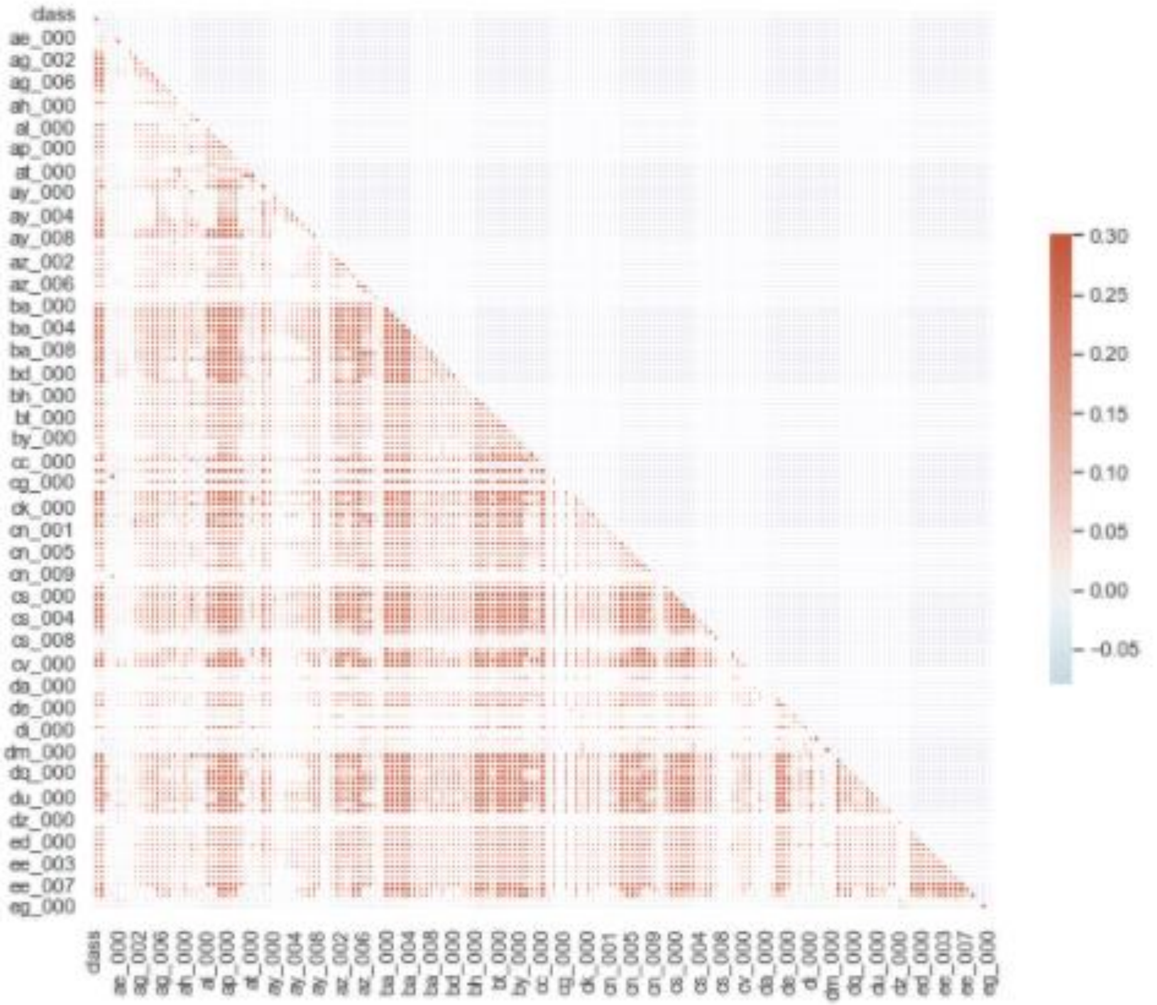
Initial exploratory data analysis has revealed the class imbalance in the data set. There are more records for negative class and way too few for positive class. At this point in the analysis, I plan to use the area under the precision-recall curve. The scikit implementation of this is average precision. I chose this matric because it is stable under class imbalance.

## Exploratory Data Analysis

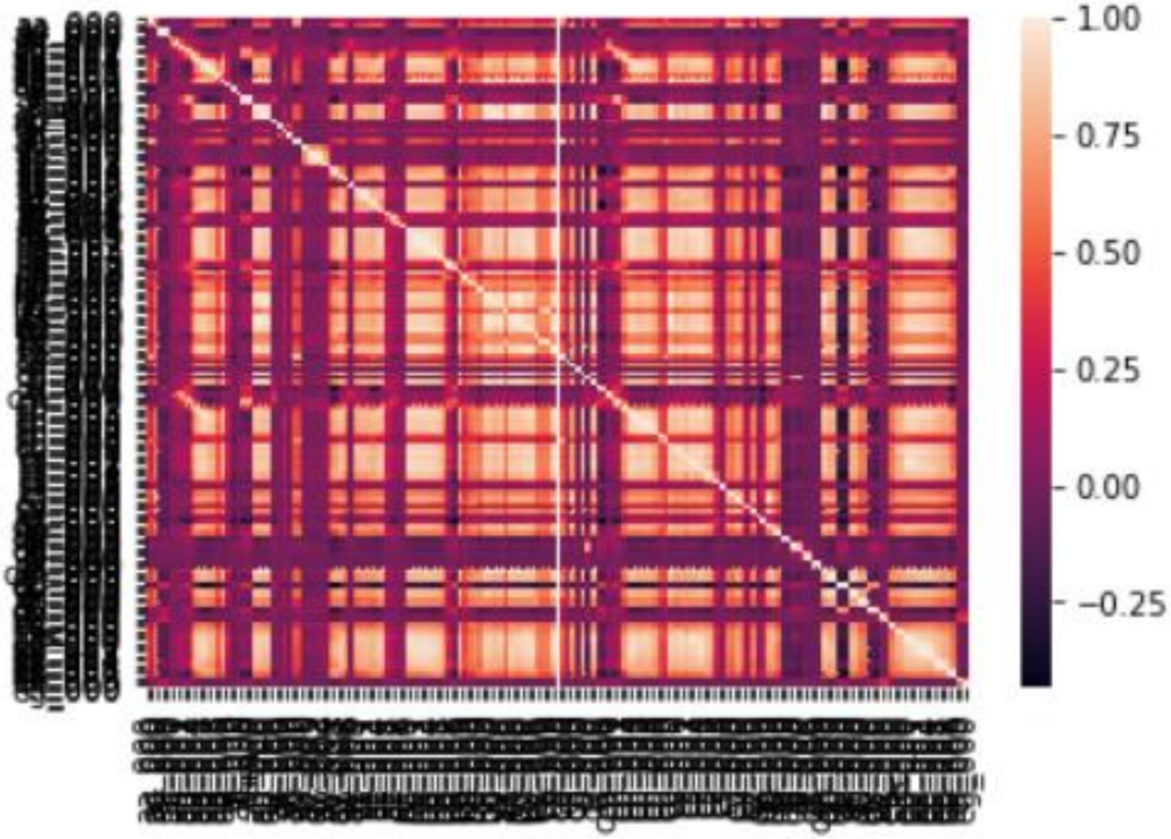
Many of the features in the dataset has missing values. It is important to impute missing values with the optimal strategy. Below is the heat map of missing values. Given that data set has too many missing values, I thought of removing features with excessive missing values. I chose the threshold to be 70%, which means drop the features which has more than 70% of the missing values. There were 10 such features and after reduction, we are left with 161 features including the target variable.



The next step was to figure out the correlation between the features and target variable. I used corr() to generate a correlation matrix and then plot it on a hit map masking the upper half. This analysis reveals that there are no strong correlations between the features and target variable. The strongest correlation was 0.30 which is a weak correlation. Below is the diagram showing the correlation matrix.

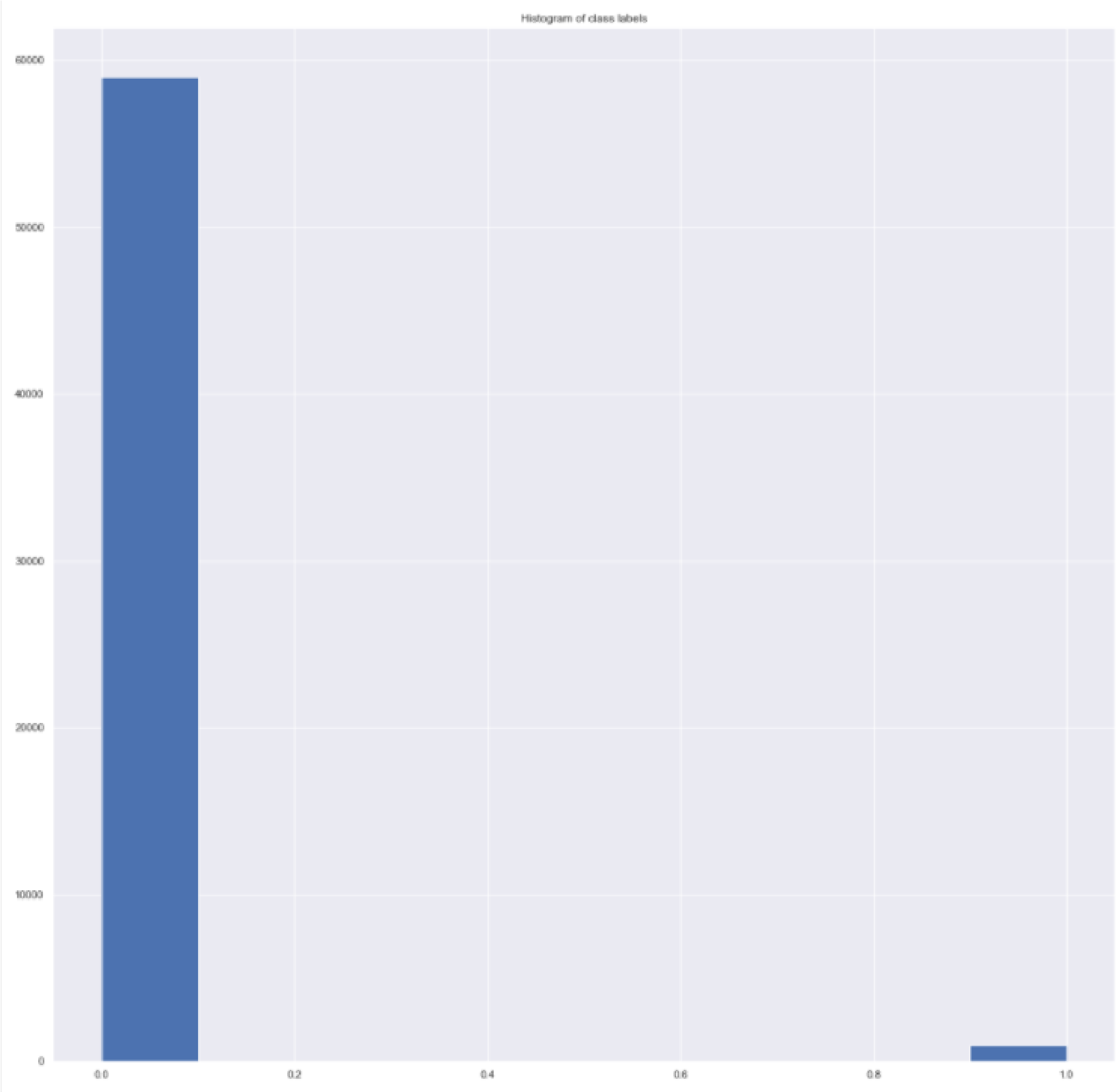


It looks like some of the features are correlated to each other. I plotted another correlation matrix of spearman correlation to check out.



Excuse the feature names not being visible. All feature names are anyway anonymized and hold no value for the purpose of this graph. Looking above it looks features are correlated to each other’s, instead of target variables. We need those features to be correlated to the target variables and not have any correlation amongst the features. We will use the PCA for feature engineering t help solve this problem, discussed in next section.

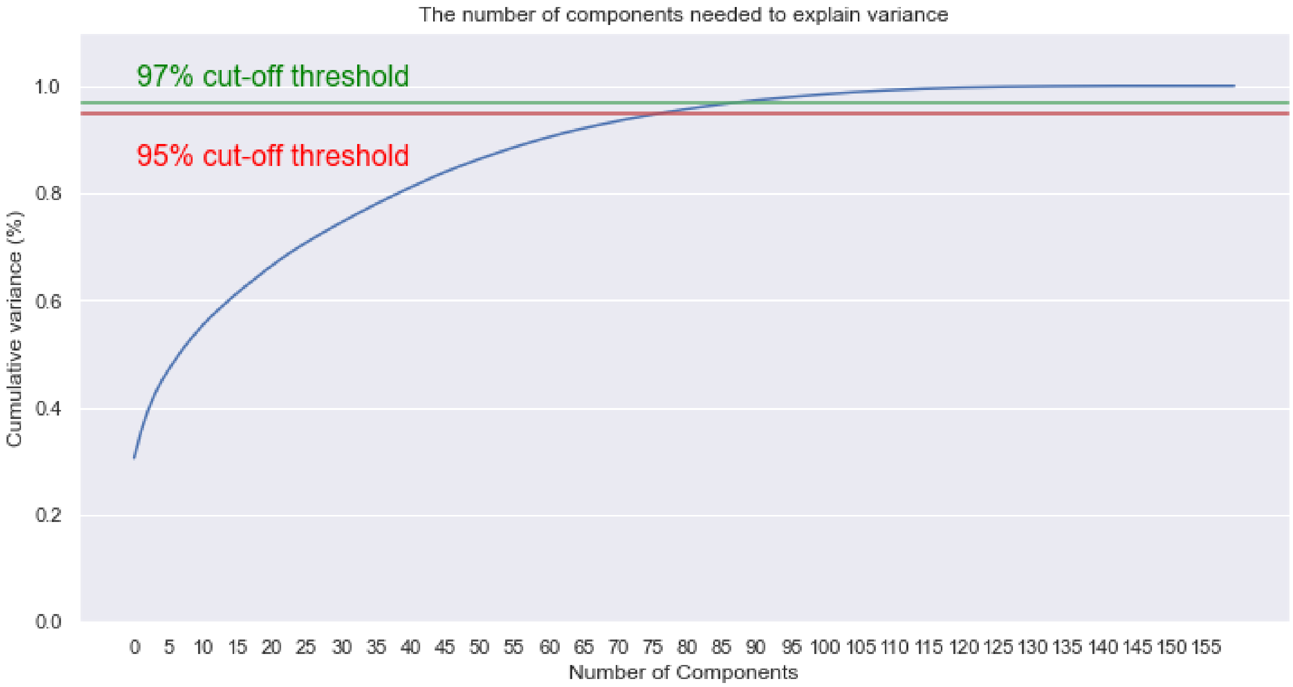
In this data set we have the target variable of the class. Class has only two values negative and positive. Negative means a non-APS failure, and it means that there was no APS failure but there was a mechanic checkup. The positive class means the failure was due to APS failure, it is classified as missed maintenance. Dataset has very few positive class observation (only 1000 out of 60000). This makes the data set highly imbalanced. Below diagram shows the distribution of class values in the data set.



## Feature Engineering

In my data set I see a lot of missing values for many attributes. I used sklearn’s SimpleImputer to impute the missing values. I chose mean strategy to impute missing values with the mean of the attribute. It was necessary to impute before doing any feature reduction or feature selection. I next performed dimensionality reduction using PCA. I checked the correlation between the features using spearman correlation coefficient. In an ideal world all the features would be uncorrelated to each other and will be correlated to the target variable. But this experiment revealed that features are also correlated to each other. In this case PCA was a good choice for feature reduction.

I plotted the cumulative variance with the number of components in a line chart. I also plotted the 95% and 97% threshold to see the correlation between components and the target variable. Plot showed that 95% of the variance could be explained by 75 components and 97% by 85 components. I think PCA is a great fit for feature reduction in this data set.



## Model evaluation and model selection

I trained logistic regression model 20 different class weights and both penalty of l1 and l2. Using five-fold cross-validation, I ended up with a total of 80 fits with the best at C = 100 and a penalty of l2. I used the F1 score to validate the performance of the models.

Then I trained a random forest classifier with five different number of estimators and max depth. I again used the five-fold cross-validation and F1 score to validate the performance. I ended up with 75 model fittings with a max depth of 10 and the number of estimators 200 to produce the best performing model. Comparing the two models Random forest classifier did much better than the logistics regressor. I also tried different imputation techniques to compare and see if accuracy improves with different techniques. I tried mean, median, and most frequent and, found that median worked best with random forest classifier.

## Cost reduction

https://archive.ics.uci.edu/ml/datasets/APS+Failure+at+Scania+Trucks has a cost scheme to help calculate the cost and evaluate the model. A stop for maintenance check is less expensive (suggested 10) versus a break down due to missed maintenance check (suggested 500).
Cost-metric of miss-classification:

Cost_1 = 10 -- False Positive - Maintenance without failure
Cost_2 = 500 -- False negative - Missed maintenance causing failure in APS
So total cost of misclassification would be (10 X FP) + (500 X FN)

I trained a model for both RandomForest and LogisticsRegression with the best hyperparameters from above and compared the cost for both models on the test set.

I also updated the scoring method to be my custom scorer, as the intent is to reduce the cost, not just be accurate in prediction. The cost break-up above shows that a true negative (a preventive maintenance) costs only $10 and a missed maintenance costs $500, which is 50 times more expensive. So, an optimal model need not to be the most accurate, rather it needs to reduce the cost.

Below are the costs using the best hyperparameters for both the classifiers.

**Logistics regressor:**

```
Confusion Matrix:
 [[15565 60]
 [ 150  225]]
---------------------------------------------
Cost 1 (FP) = 150
Cost 2 (FN) = 60
```

```
Total cost = 31500
```

**Random forest classifier:**

```
Confusion Matrix:
 [[15611 14]
 [ 126 249]]
------------------------------------------------
Cost 1 (FP) = 126
Cost 2 (FN) = 14
Total cost = 8260
```

## Result

Random forest classifier seems to reduce the maintenance cost better than logistics regressor. I have tried it with all the imputation techniques including median, mean and most frequent. I found that random forest classifier worked better with all three imputations over the logistics regressor. And within those three imputation techniques I tried, median worked the best.

The next steps should be to try out other imputation techniques like KNN imputation, hot deck and cold deck to see if other techniques achieve better costing. Also, we should try XG boost classifier.