Predicting Machine failure for Pitney Bowes

Balakumaran Ramaswamy Kannan, Eun Hee Noh, Noyonika Roy
Baruch College Zicklin School of Business
balakumaran.kannan@baruchmail.cuny.edu, eunhee.noh@baruchmail.cuny.edu, noyonika.roy@baruchmail.cuny.edu

Abstract

Predicting machine failure is an important step in pre-emptive customer service that will allow Pitney Bowes to have a higher customer approval rating and hence higher client retention as the downtime for the clients reduces significantly. The machines are always connected to the cloud making the process of data collection streamlined and collected in real-time. This resource can be utilized to predict the failure of machines in the next seven days through prediction modelling. Seven days gives the company enough time to initiate the replacement process.

A snapshot of the data collected in the cloud, where the machines are connected, is provided with information for **40 thousand machines** giving that many rows of data. This project employs various machine learning libraries such as LightGBM and Xgboost and feature engineering methods such as SHAP to select relevant features, optimize our solution and improve the accuracy of our final model.

Introduction

Pitney Bowes Data Challenge is set by the one of the leaders in the mailing meters space. Participants are required to predict which machines are most likely to fail in the next 7 days. A snapshot of the Cloud-connected meter health has been provided for 40k meters as a training sample set. The following files were provided:

train updated 0413202.csv - contains the training data set
- 40500 Rows ~ 16.1MB

test_for_submission.csv - contains the test data set
- 4501 Rows ~1.8 Mb

Data

Dataset used in this study was given as part of Pitney Bowes ' Baruch Data Challenge. 54 Variables contained in the dataset are shown in the following table:

Train and Test files :	
deviceid	A unique is representing a machine
avg_time_charging_lag1	Average time taken to charge 1..14 day(s) before snapshot
charging_rate	Rate of charging 1..14 day(s) before snapshot
avg_time_discharging_lag1	Average time taken to discharge on 1..14 day(s) before snapshot
charge_cycle_time_below_12	Total charges cyles time less than 12 units
discharging_rate_lag4	Rate of charging on 1..14 day(s) before snapshot
chargecycles	total cycles of charge
dischargecycles	total cycles of discharge
total_off_time	total time device was off
number_times_restart	restart number of times
avg_volt_change_charging	avg voltage change while charging
avg_volt_change_discharging	avg voltage change while discharging
avg_time_charging	avg time while charging on 1..14 day(s) before snapshot
avg_time_discharging	avg time while discharging on 1..14 day(s) before snapshot
max_voltage_day	max voltage reached
piececount	total labels printed
cycle_time	total cycles time
LastRecord	1st april date of snapshot
Date Deployed	Date when meters were deployed
Fail_7	TARGET VARIABLE

Feature Engineering

To extract the most important variables we used the following methods:

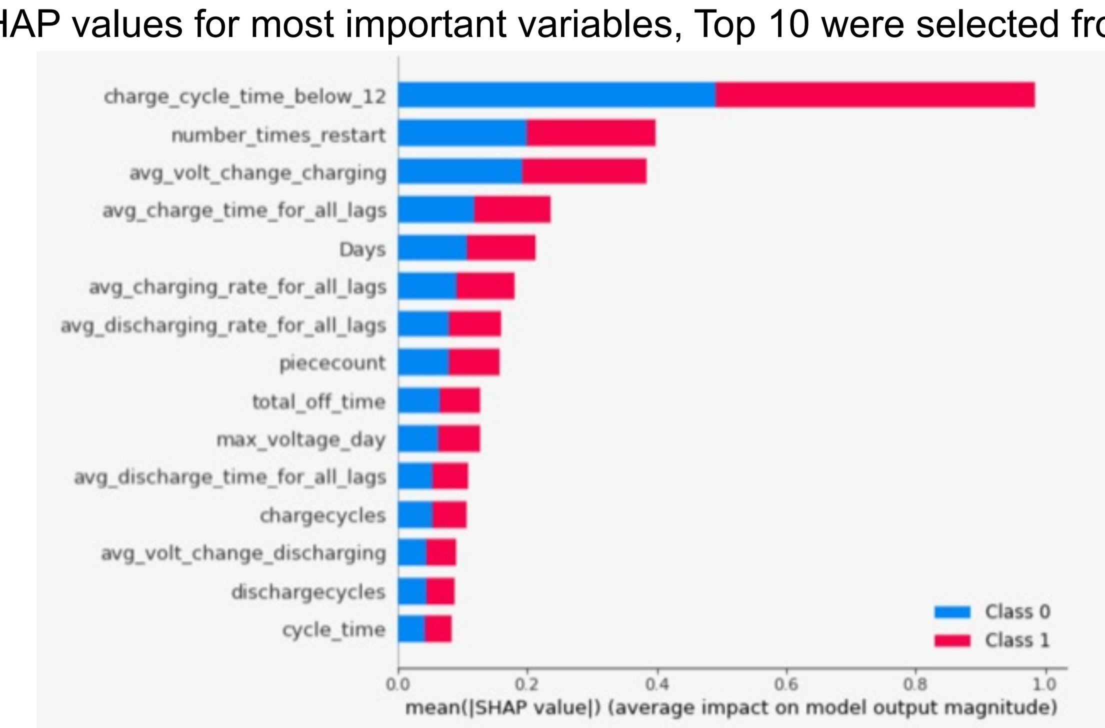
1. Aggregated and derived variables:

SECONDARY FEATURES	
avg_charging_rate_for_all_lags	Average of average charging rate for all days (14)
avg_charge_time_for_all_lags	Average of average charging time for all days (14)
avg_discharge_time_for_all_lags	Average of average discharging time for all days (14)
avg_discharging_rate_for_all_lags	Average of average discharging rate for all days (14)
Days	LastRecord- Date deployed

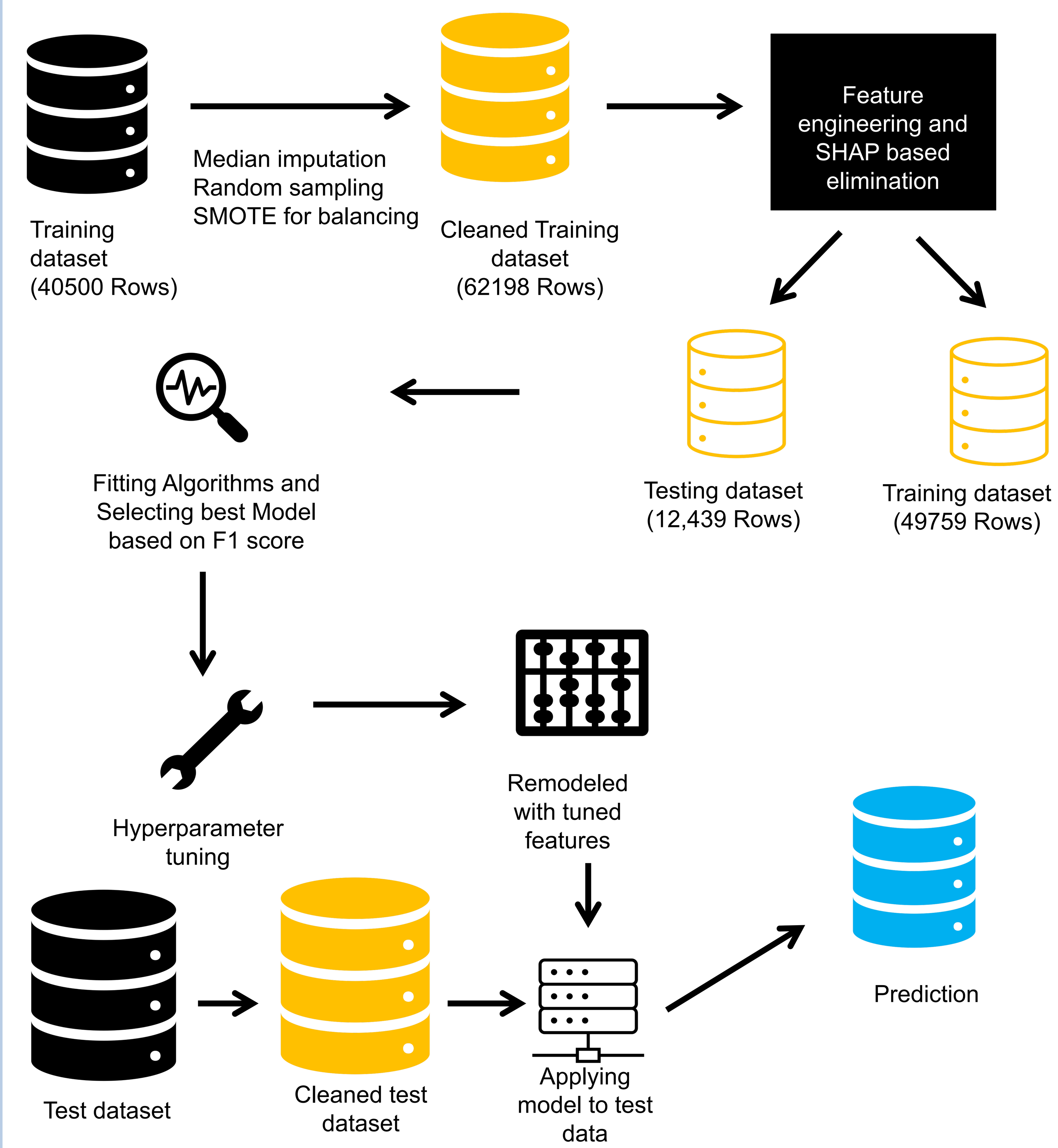
Device id was removes as a variable temporarily, Last record and Date Deployed were also removed and replaced with Days. This changed the total number of features to 16 from 54. The total summary of the current features is;

N	Variable
1	charge_cycle_time_below_12
2	chargecycles
3	dischargecycles
4	total_off_time
5	number_times_restart
6	avg_volt_change_charging
7	avg_volt_change_discharging
8	max_voltage_day
9	piececount
10	cycle_time
11	avg_charging_rate_for_all_lags
12	avg_charge_time_for_all_lags
13	avg_discharge_time_for_all_lags
14	avg_discharging_rate_for_all_lags
15	Days
16	Fail_7

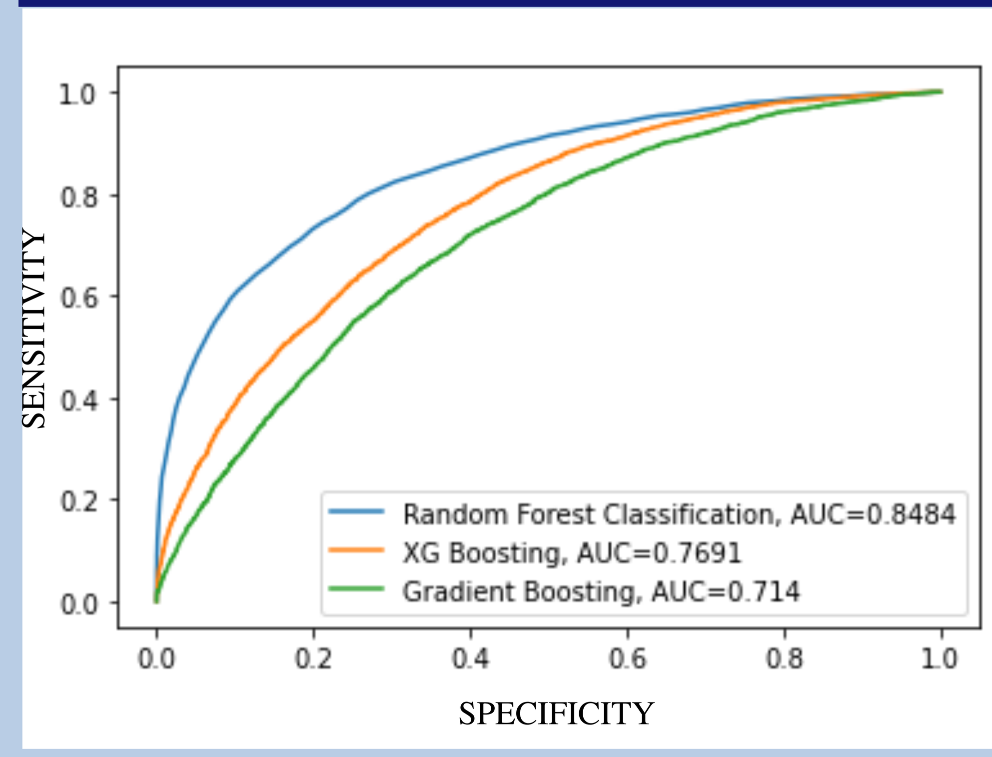
2. SHAP values for most important variables, Top 10 were selected from this:



METHODOLOGY



Models & Results

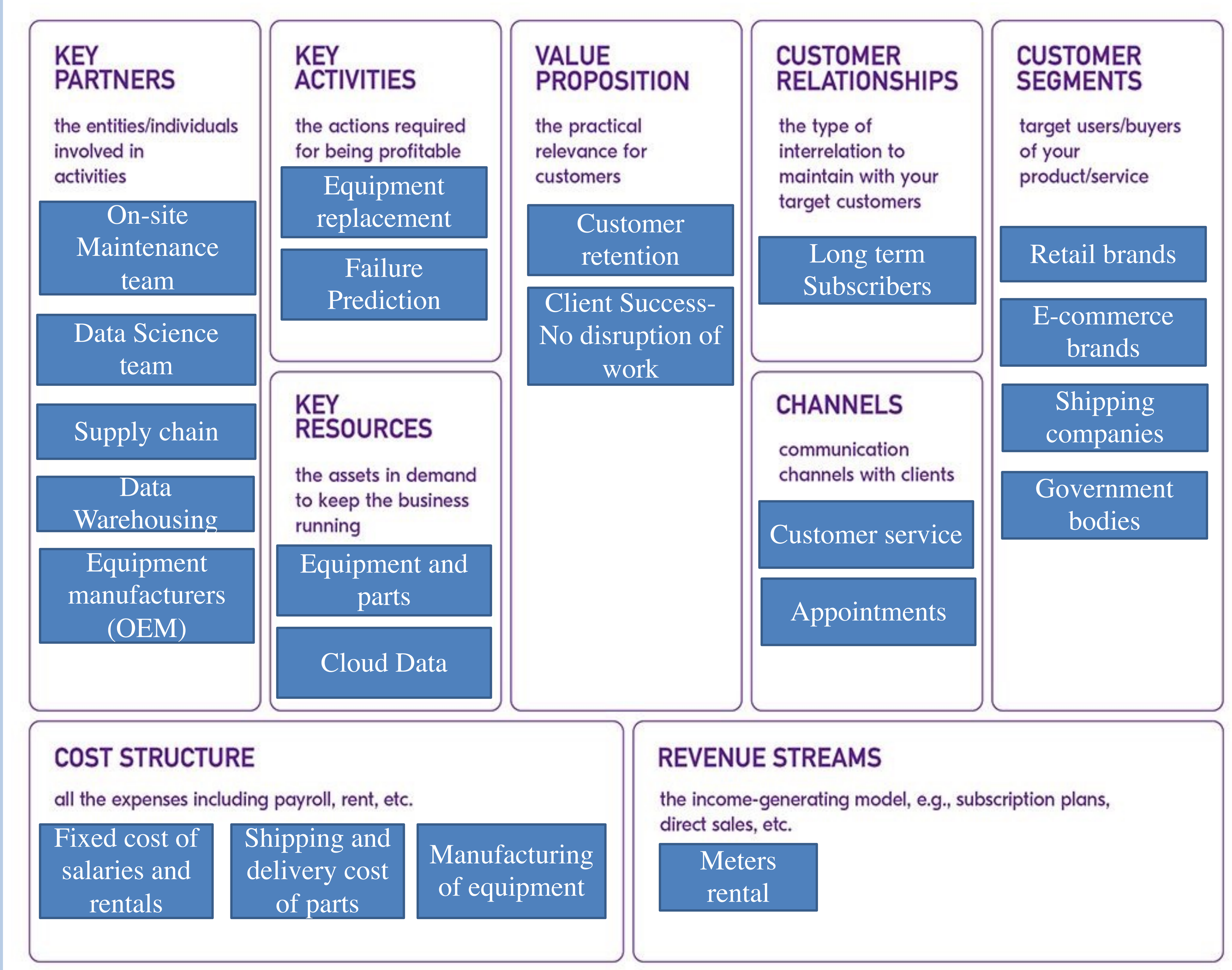


A few algorithms were fitted on the given dataset and the top 3 algorithms were selected.

From this Random Forest was selected to be tuned as it had the highest AUC Score

Models	AUC	AP-SCORE	Recall Score	Precision Score	F1-Score
Random Forest	0.864	0.875	0.803	0.768	0.780
Gradient Boosting	0.711	0.688	0.689	0.645	0.662
Xgboost	0.770	0.757	0.733	0.688	0.710
AdaBoost	0.691	0.660	0.670	0.637	0.653

Lean Canvas



Conclusions

- Predictive maintenance is a useful tool in avoiding down-time for the clients of Pitney Bowes
- Feature selection played a big role is increasing the accuracy of the model, after multiple runs, 10 was selected as the optimum number of features
- The model is very economical as it takes very less computation time
- Random Forest proved to be the model with the highest accuracy
- Hyper-tuning of parameters helped reduce overfitting
- Further improvement can be made on the model to improve the accuracy by training the model on random samples of the dataset and by changing the arguments of hyper tuning

Acknowledgement

The poster was made by Team 2 of the Baruch Data Challenge held by Pitney Bowes. We would like to thank Shivayogi Biradar, David Messineo and Christian Bernards from Pitney Bowes for providing support and setting up the challenge. We would also like to thank Justyn Macarwcyz for coordinating with the teams.