12/2/2021

Predictive Maintenance of Aircraft Engine

Raja Muthu

IE7374 : SENSOR ANALYTICS IN ENGINEERNING SYSTEMS

Contents

Introduction

Predictive Maintenance

"Use of data-driven, proactive maintenance methods that are designed to analyze the condition of equipment and help to predict when maintenance should be performed"[1]

Predictive maintenance evaluates the condition of equipment by performing periodic (offline) or continuous (online) equipment condition monitoring. By making use of data science and predictive analytics to estimate when a given device might fail so that corrective maintenance can be scheduled before the point of failure. The goal is to schedule maintenance at the most convenient and most cost-efficient moment, allowing equipment's lifespan to be optimized to its fullest, but before the equipment fails.

Predictive vs Preventive Maintenance

The difference between preventive maintenance and predictive maintenance lies in the data being analyzed. While a predictive maintenance technician relies on monitoring and analyzing data from the actual, current condition of the equipment in operation, preventive maintenance relies on historical data, averages, and life expectancy statistics to predict when maintenance activities will be required.

Importance of Predictive Maintenance

Predictive maintenance insights are an extremely valuable asset in improving the overall maintenance and reliability of an operation. Benefits include:

- minimize the number of unexpected breakdowns
- maximize asset uptime and improve asset reliability
- reduce operational costs by performing maintenance only when necessary
- maximize production hours
- improve safety
- streamline maintenance costs through reduced equipment, inventory costs, and labor.

Need for predictive maintenance in Aircraft Engine

In current world transportation aircrafts plays a significant role. It helps consumer in reducing time spent on travel when compared to other traditional services. With current advancement in technology a lot of improvement has been made in the aircraft to make it simpler and safer to use. The Aircrafts now has a very high load carrying capacity and can stay afloat for longer duration. But with the advancement there are some crucial issues in the service as well in terms of maintenance time and cost.

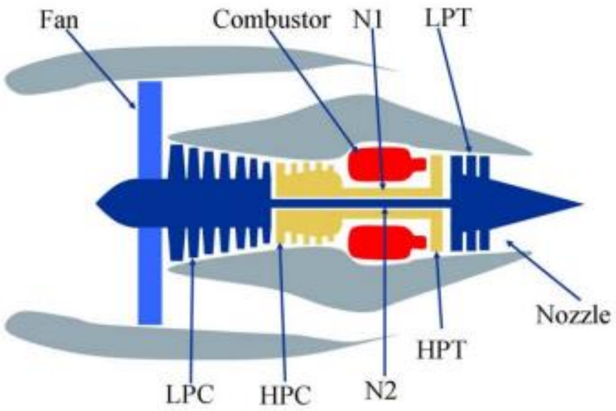
By estimating when maintenance of an aircraft is required we can,

- Improve operational safety
- Increase equipment usage
- Avoid unscheduled maintenance there by increasing productivity
- Change operational characteristics (such as load) which in turn may prolong the life of the component.

Data Set

The data is generated for the Prognostics and Health Management (PHM) competition at PHM'08. The aim of the task was to estimate remaining life of an unspecified system using different conditional data of the machine, irrespective of the underlying physical process.

The Data is generated by simulating a realistic large commercial turbofan engine using C-MAPSS tool.



C-MAPSS (Commercial Modular Aero Propulsion System Simulation)

The software is coded in the MATLAB® and Simulink® environment, and includes a number of editable input parameters that allow the user to enter specific values of his/her own choice regarding operational profile, closed-loop controllers, environmental conditions, etc.

C-MAPSS simulates an engine model of the 90,000 lb. thrust class and the package includes an atmospheric model capable of simulating operations at

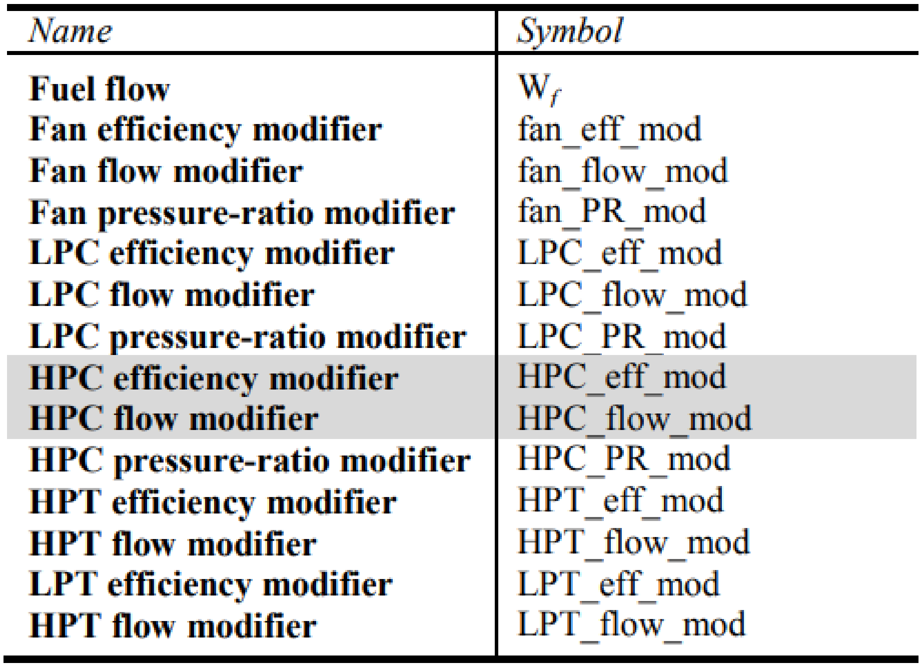
(i) Altitudes ranging from sea level to 40,000 ft,
(ii) Mach numbers from 0 to 0.90, and
(iii) Sea-level temperatures from –60 to 103 °F.

The package also includes a power management system that allows the engine to be operate over a wide range of thrust levels throughout the full range of flight conditions.

System model

The C-MAPSS system consists of 14 inputs (Table 1) and can produce multiple outputs. The inputs include fuel flow and a set of 13 health-parameter inputs that allow the user to simulate the effects of faults and deterioration in any of the engine's five rotating components (Fan, LPC, HPC, HPT, and LPT). The HPC stands for High Pressure compressor and HPT for High Pressure Turbine similarly LPC stands for Low Pressure compressor and LPT for Low Pressure Turbine.

Out of 58 outputs measured by the system 21 are used for generating our sample data.



Name	*Symbol*
Fuel flow	W_f
Fan efficiency modifier	fan_eff_mod
Fan flow modifier	fan_flow_mod
Fan pressure-ratio modifier	fan_PR_mod
LPC efficiency modifier	LPC_eff_mod
LPC flow modifier	LPC_flow_mod
LPC pressure-ratio modifier	LPC_PR_mod
HPC efficiency modifier	HPC_eff_mod
HPC flow modifier	HPC_flow_mod
HPC pressure-ratio modifier	HPC_PR_mod
HPT efficiency modifier	HPT_eff_mod
HPT flow modifier	HPT_flow_mod
LPT efficiency modifier	LPT_eff_mod
HPT flow modifier	LPT_flow_mod

Table 1

Four different sets of data were simulated under different combinations of operational conditions and fault modes. Each set consists of Train, Test and RUL (for Test Data)

- Sample 1 : consists of 100 engine trajectory simulations
- Sample 2 : consists of 260 engine trajectory simulations
- Sample 3 : consists of 100 engine trajectory simulations
- Sample 4 : consists of 248 engine trajectory simulations

I have used the first set from the data sample, which consists of 20631 rows of training data points generated from 100 different engines. The test data is simulated for the 100 engines to various stages of their lifetime and it contains 16596 data points.

While the engines in training data are simulated till the point of failure i.e., till the remaining life is zero.

Test Data engines are simulated to different unknown stage, and the corresponding value in RUL file gives the remaining useful life of that engine.

Remaining Useful and percent RUL Calculation

- Training data

  The RUL at each cycle of engine is calculated as the difference between Maximum cycles an engine can run and current cycle i.e.,

RUL = Max_Cycle – Current_Cycle

- Test Data
  The Maximum cycle value of each engine is calculated and added with the RUL to determine its Max_Cycle and corresponding RUL at each cycle is calculated by taking difference between Max_Cycle and Current_Cycle

Max_Cycle = Cycle value till engine is simulated + RUL value from RUL file

RUL = Max_Cycle – Current_Cycle

The Percentage of RUL is calculated as Percent, **P = RUL / Max_Cycle**

Assumption and Labeling

The assumption is that if an engine is left with 20% of it's percent RUL, then it requires maintenance.

From the above stated assumption the maintenance label is set to 1 if the RUL percent is less than 20 and 0 otherwise.

Maintenance Label 1 if RUL percent < 20

Maintenance Label 0 if RUL percent > 20

Analysis

Feature Selection

In machine learning and statistics, feature selection, also known as variable selection, attribute selection or variable subset selection, is the process of selecting a subset of relevant features for use in model construction. It is desirable to reduce the number of input variables to both reduce the computational cost of modeling and, in some cases, to improve the performance of the model.

The below boxplot shows the descriptive statistic of the given training data, to look at the values of different sensor readings.

As we can see from the plot 1(Appendix) condition 3, Temperature sensor T2, Pressure sensor P2, P15, Engine pressure ratio epr, fuel ratio farb and fan speed readings Nf_dmd and PCNfR_dmd doesn't contribute any relevant information to the data set and therefore we can drop them.

Principal Component Analysis

I have performed the PCA as latent space only for observational significance. To visually see the effect of all the 14 features with the RUL. It clearly shows engine cycles with higher RUL are clustering together.

And the data points of engine which are towards the end of there life cycle are clustering together but sparsely.

Correlation Plot

The Correlation Plot gives us the idea of how individual feature relates to RUL value. Key take away from this plot are, as the engine reaches towards it's failure point the temperature and it's related values are increasing while Pressure and it's related values are decreasing.

Model Development

Using the correlation plot the features are further eliminated based on there impact or importance in RUL. From the starting 24 features we are left with 14 features.

A classification model is developed using these 14 features to predict and determine if an engine needs maintenance.

Logistic Regression Model

I started with the logistic regression model as it's a simpler model and wanted to see it's performance to the data before moving on with complex data models.

Grid search cross validation was performed with different parameters to the Logistic regression model, which also included 5-fold cross validation. The models were compared based on accuracy score of training data and the best combination of solver and regularization value is selected.

Liblinear, lbfgs, sag, saga solvers with regularization value(C) of 1, 5, 10 and 15 were compared and the accuracy is seen for Liblinear with C = 10.

Accuracy of Training Data : 95.3%

Area under curve Training Data: 0.920

Confusion Matrix of Training Data

True Value	Predicted Value		
		0	1
	0	16057	411
	1	556	3607

Accuracy of Testing Data : 71.7%

Area under curve Training Data: 0.852

Confusion Matrix of Testing Data

True Value	Predicted Value		
		0	1
	0	8843	3707
	1	0	546

From the confusion matrix it's very clear that the model is able to predict true positive correctly but it's false positive prediction rate is very high.

Support Vector Machine

Similar to logistic regression model support vector machine model was tested using grid search with different combination of kernels ,regularization values and cross validation of 5-fold.

Linear, rbf, polynomial kernel with regularization value of 1,5 and 10 were compared and the best accuracy for training data set is obtained for Poly kernel with C= 1.

Accuracy of Training Data : 95.9%

Area under curve Training Data: 0.920

Accuracy of Testing Data : 78.9%

Area under curve Training Data: 0.888

Confusion Matrix of Training Data

True Value	Predicted Value		
		0	1
	0	16218	250
	1	595	3568

Confusion Matrix of Testing Data

True Value	Predicted Value		
		0	1
	0	9795	2755
	1	2	544

From the confusion matrix it's very clear that the model is able to predict true positive correctly but it's false positive prediction rate is high but there's approximately drop in 1000 false predictions when compared to logistic regression model.

Random Forest Classification

The random forest model was also selected based on the accuracy score calculated for different number of estimators used during classifications. For the grid search method model performance of different random forest models with estimator values equal to 100,250,500 and 750 were compared with 5-Fold cross validation. Best performance is achieved for model with 500 estimators.

In terms of performance metric i.e., time taken to train the model ana make the prediction, Random forest method takes the longest duration while logistic regression the shortest.

Accuracy of Training Data : 100%

Accuracy of Testing Data : 85.7%

Area under curve Training Data: 1.0

Area under curve Training Data: 0.917

Confusion Matrix of Training Data

True Value	Predicted Value		
		0	1
	0	16468	0
	1	0	4163

Confusion Matrix of Testing Data

True Value	Predicted Value		
		0	1
	0	10600	1950
	1	8	538

The accuracy for training data is 100 % and the model is able to correctly classify positive and negative classes. Even for Testing data there's a high accuracy and significant drop in number of false positives.

K- Nearest Neighbors

Based on the number of neighbors included for classification model development the performance changes. I have compared the models for neighbors ranging from 1 to 17 using the grid search method with 5-fold cross validation. The KNN model takes comparatively less time to train and predict the data than Random Forest models. I was able to achieve best performance for models with 17 neighbors and the results can be seen below.

Accuracy of Training Data : 95.8%

Accuracy of Testing Data : 86.8%

Area under curve Training Data: 0.919

Area under curve Training Data: 0.929

Confusion Matrix of Training Data

True Value		Predicted Value	
		0	1
	0	16190	278
	1	596	3567

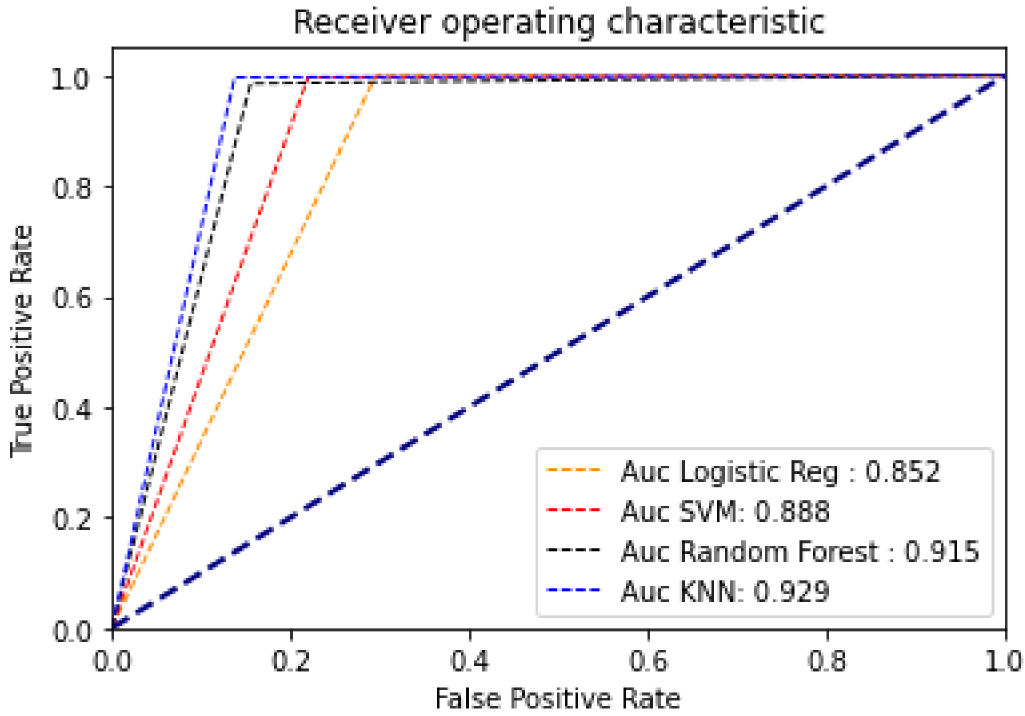
Confusion Matrix of Testing Data

True Value		Predicted Value	
		0	1
	0	10824	1726
	1	2	544

Even though the accuracy on training data is less compared with what we obtained in random forest, KNN model performance is much better on testing data. And both false positive and false negative values are minimized.

Conclusions

The plot shown below is the ROC of testing data for various models, both from accuracy results and auc values we can say the KNN model performance is the best.



From the Scatter Plot (Plot 5 in Appendix) we have the PCA results plotted with the labels from Test data and the KNN Predicted labels titled Predictions. We can clearly observe a definite decision boundary between Orange and Blue.

Further work

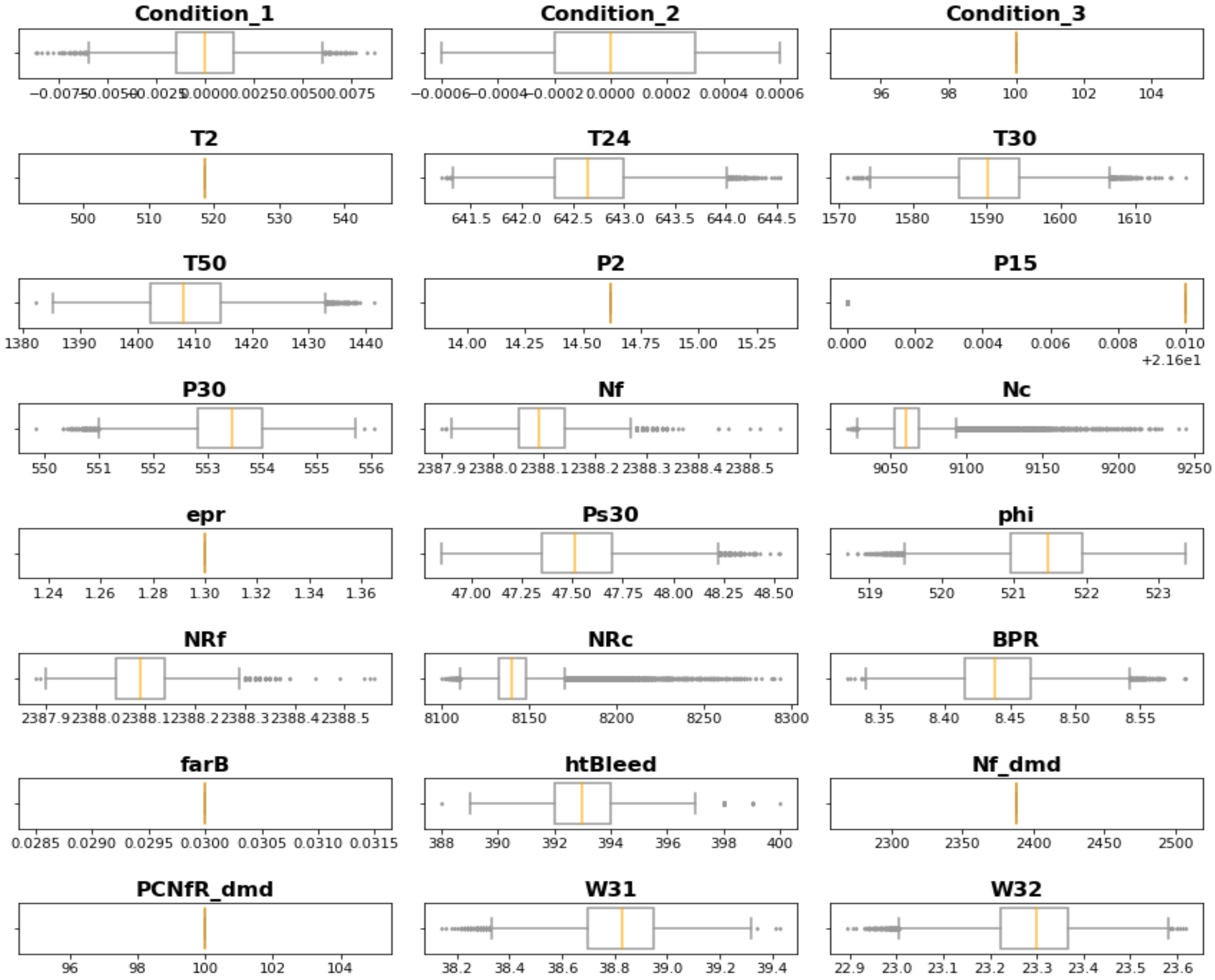
The data can be further converted into multiclass problem by setting a second percent cut off value at 9-13 %, which can notify the engine can not be used without performing maintenance. And to show the engine is currently working at critical level on a verge of failure.

References

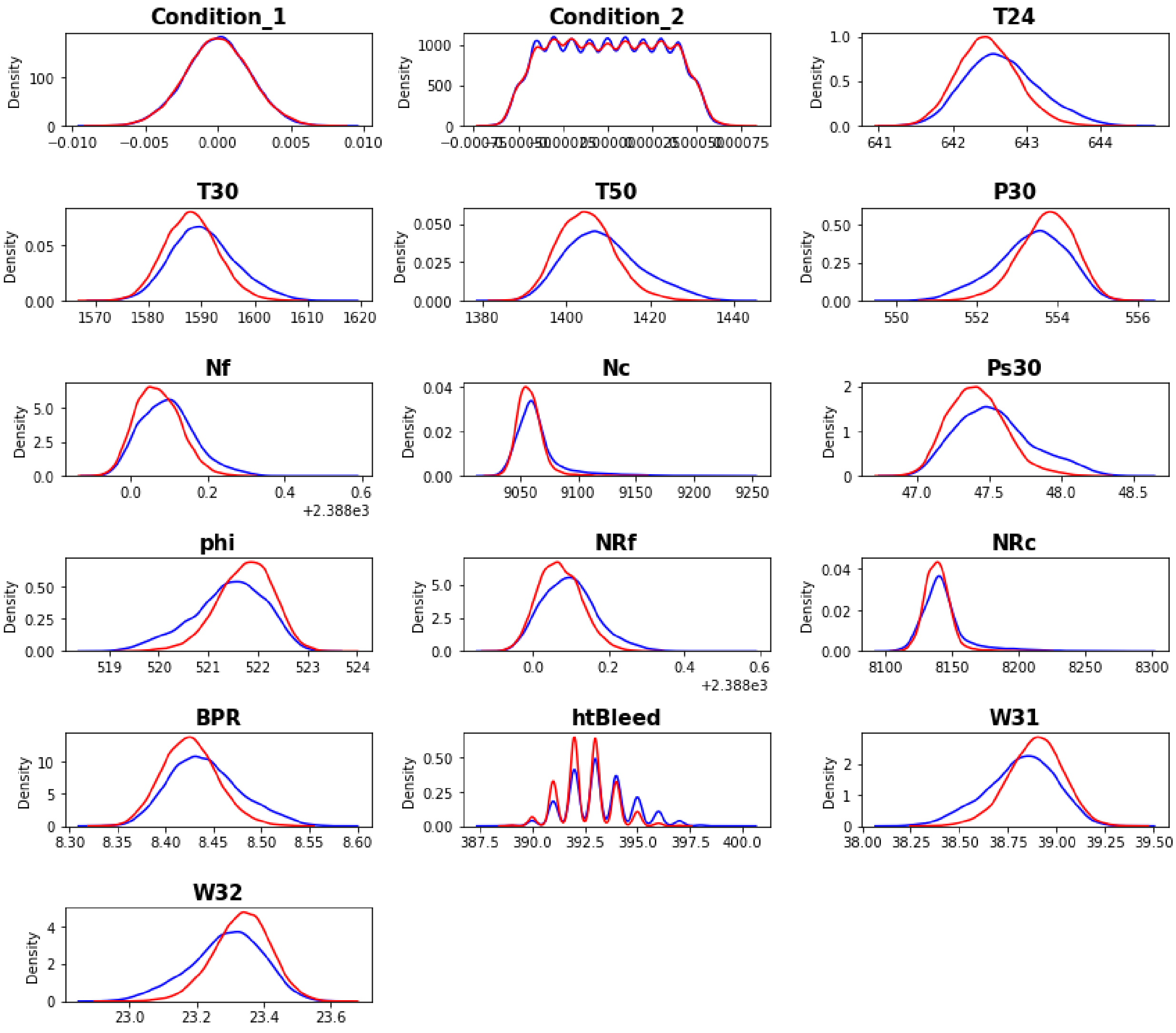
1. A. Saxena and K. Goebel (2008). "Turbofan Engine Degradation Simulation Data Set", NASA Ames Prognostics Data Repository (http://ti.arc.nasa.gov/project/prognostic-data-repository), NASA Ames Research Center, Moffett Field, CA
2. Nasa Prognostic Data Repository
3. Grid Search CV
4. Logistic Regression
5. Support Vector Classifier
6. Random Forest Classifier
7. K-nearest neighbor

Appendix

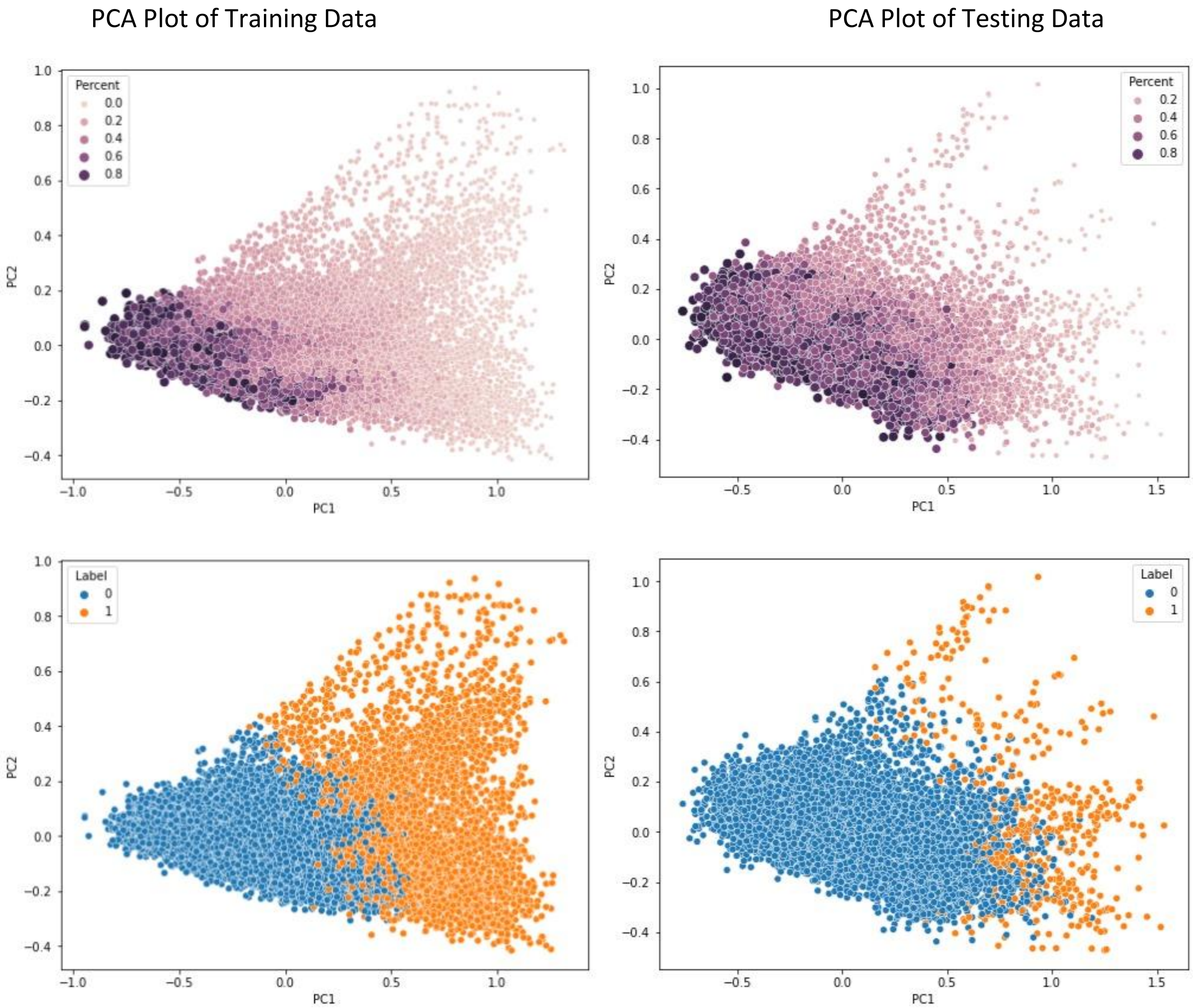
Plot 1 : Distribution Plot of Training Data



Plot 2 : Density plot of Training(Blue) and Test Data



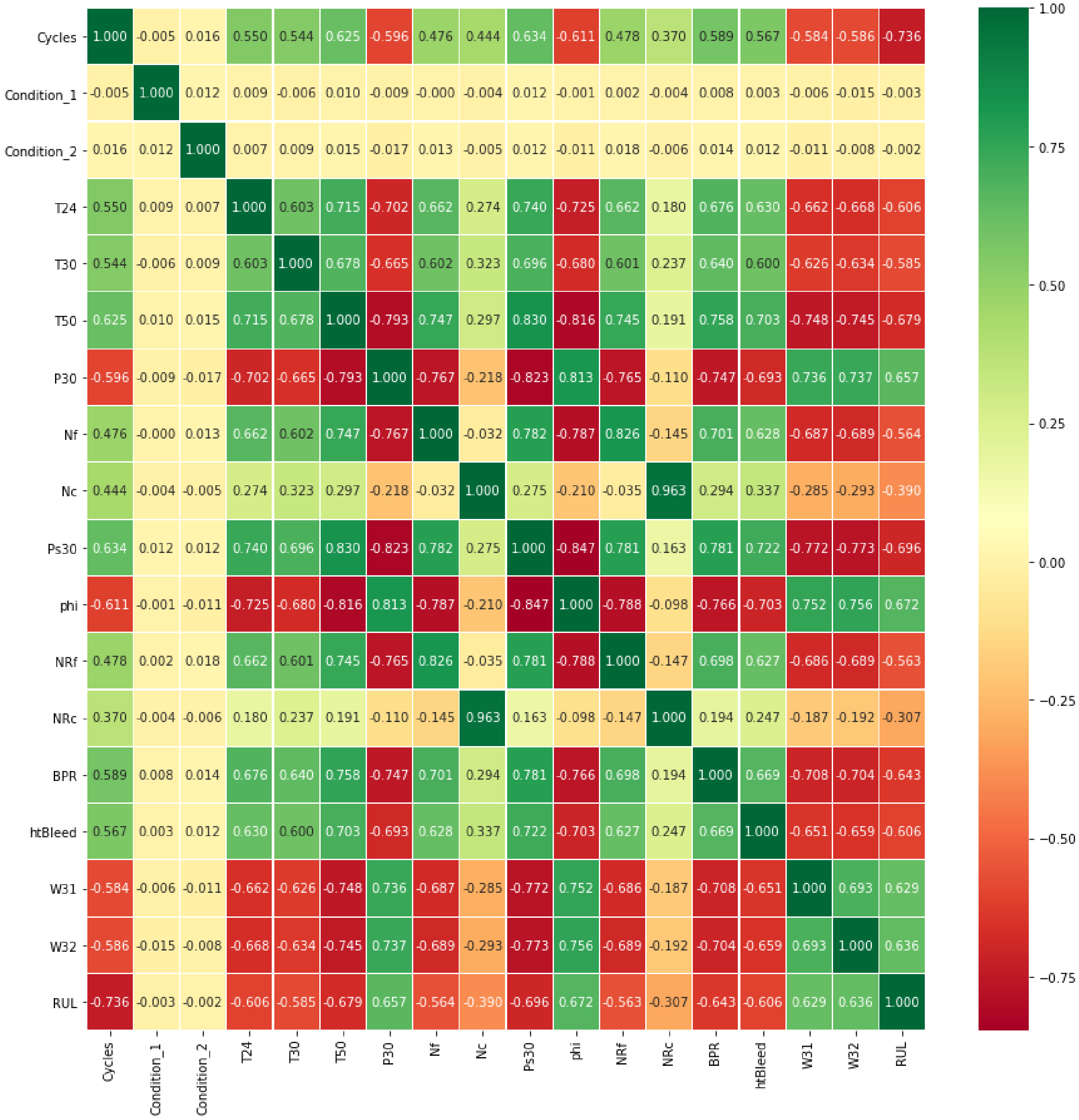
Plot 3 :



PCA Plot of Training Data

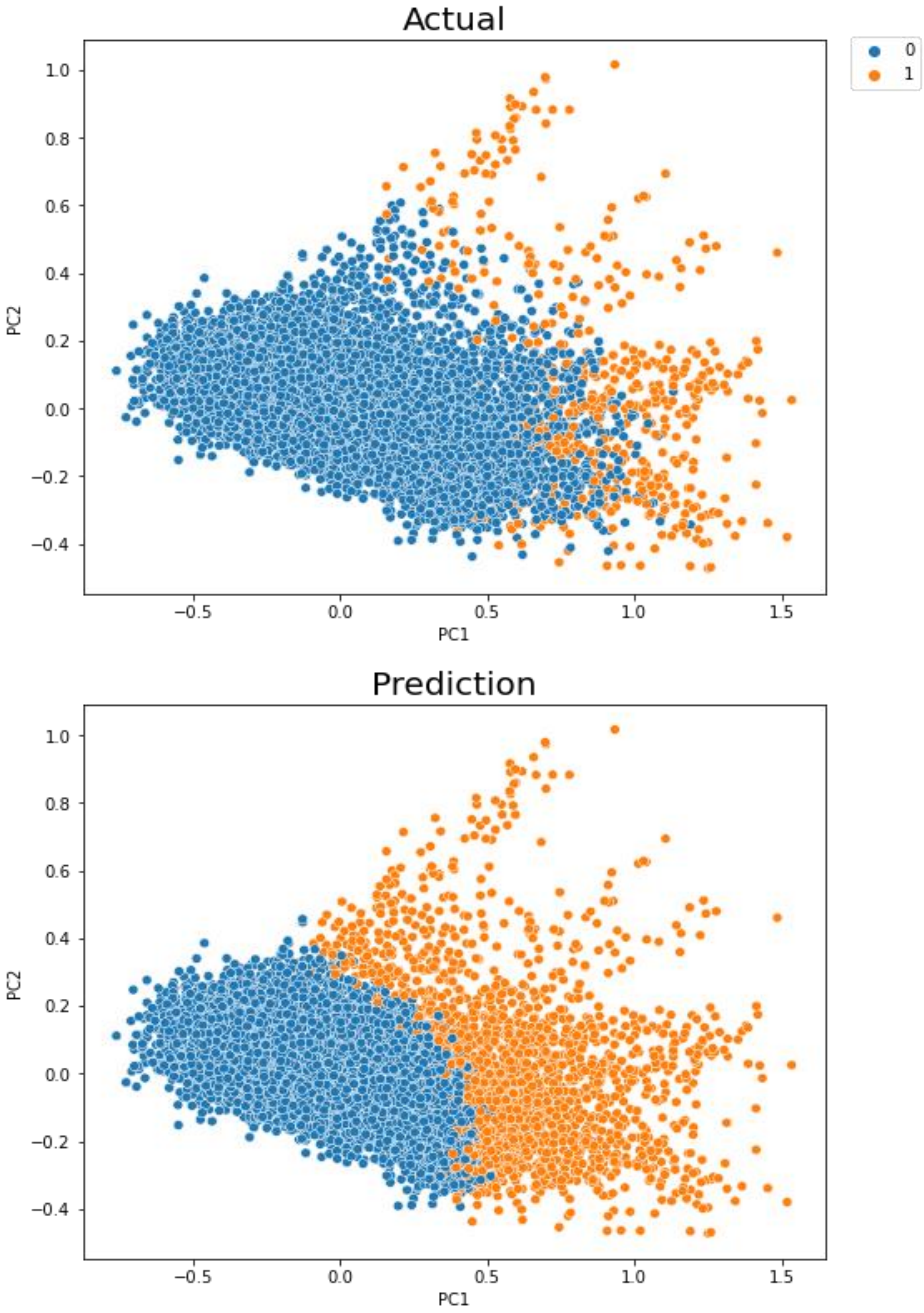
PCA Plot of Testing Data

Plot 4 : Correlation Plot

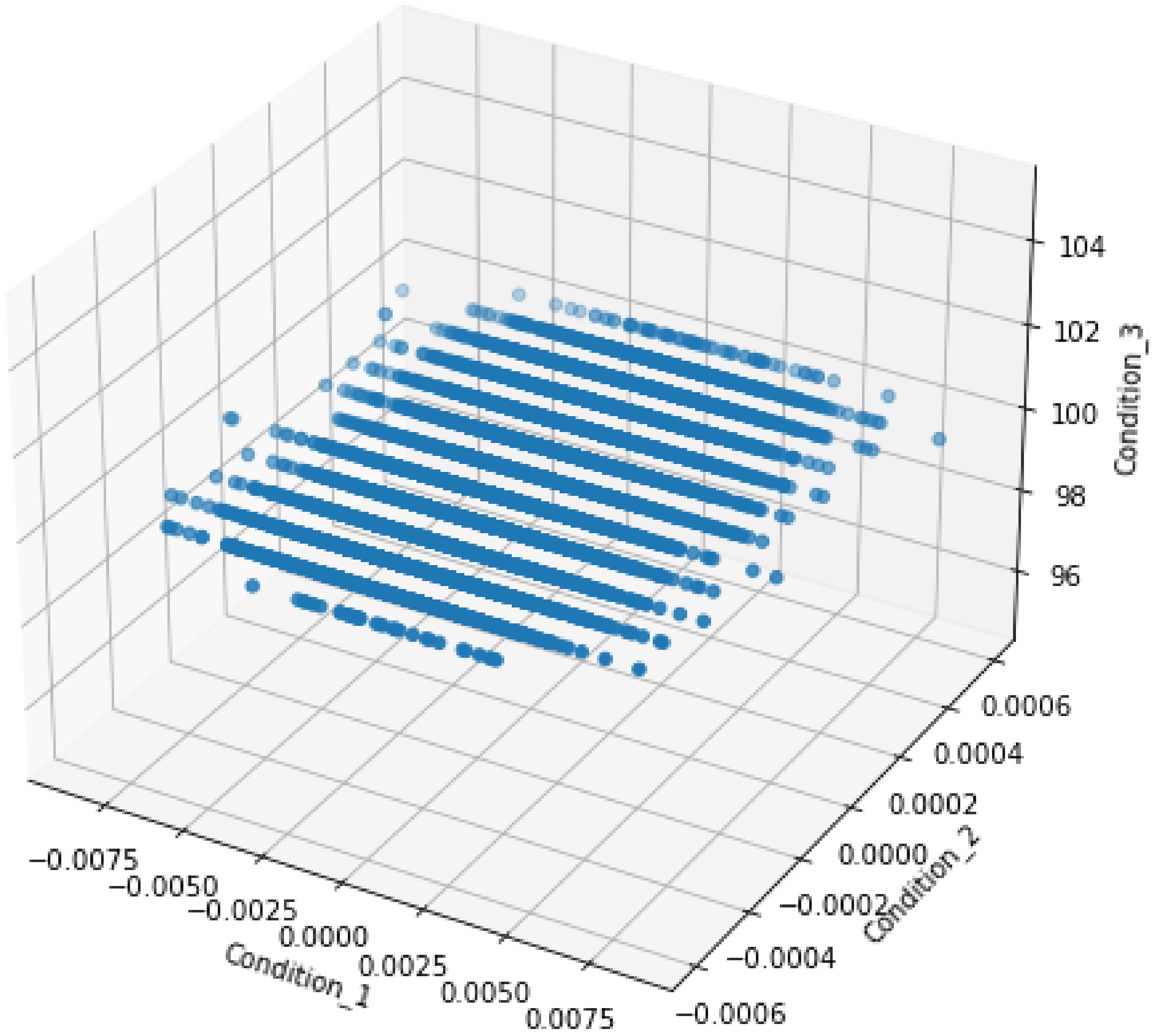


	Cycles	Condition_1	Condition_2	T24	T30	T50	P30	Nf	Nc	Ps30	phi	NRf	NRc	BPR	htBleed	W31	W32	RUL
Cycles	1.000	-0.005	0.016	0.550	0.544	0.625	-0.596	0.476	0.444	0.634	-0.611	0.478	0.370	0.589	0.567	-0.584	-0.586	-0.736
Condition_1	-0.005	1.000	0.012	0.009	-0.006	0.010	-0.009	-0.000	-0.004	0.012	-0.001	0.002	-0.004	0.008	0.003	-0.006	-0.015	-0.003
Condition_2	0.016	0.012	1.000	0.007	0.009	0.015	-0.017	0.013	-0.005	0.012	-0.011	0.018	-0.006	0.014	0.012	-0.011	-0.008	-0.002
T24	0.550	0.009	0.007	1.000	0.603	0.715	-0.702	0.662	0.274	0.740	-0.725	0.662	0.180	0.676	0.630	-0.662	-0.668	-0.606
T30	0.544	-0.006	0.009	0.603	1.000	0.678	-0.665	0.602	0.323	0.696	-0.680	0.601	0.237	0.640	0.600	-0.626	-0.634	-0.585
T50	0.625	0.010	0.015	0.715	0.678	1.000	-0.793	0.747	0.297	0.830	-0.816	0.745	0.191	0.758	0.703	-0.748	-0.745	-0.679
P30	-0.596	-0.009	-0.017	-0.702	-0.665	-0.793	1.000	-0.767	-0.218	-0.823	0.813	-0.765	-0.110	-0.747	-0.693	0.736	0.737	0.657
Nf	0.476	-0.000	0.013	0.662	0.602	0.747	-0.767	1.000	-0.032	0.782	-0.787	0.826	-0.145	0.701	0.628	-0.687	-0.689	-0.564
Nc	0.444	-0.004	-0.005	0.274	0.323	0.297	-0.218	-0.032	1.000	0.275	-0.210	-0.035	0.963	0.294	0.337	-0.285	-0.293	-0.390
Ps30	0.634	0.012	0.012	0.740	0.696	0.830	-0.823	0.782	0.275	1.000	-0.847	0.781	0.163	0.781	0.722	-0.772	-0.773	-0.696
phi	-0.611	-0.001	-0.011	-0.725	-0.680	-0.816	0.813	-0.787	-0.210	-0.847	1.000	-0.788	-0.098	-0.766	-0.703	0.752	0.756	0.672
NRf	0.478	0.002	0.018	0.662	0.601	0.745	-0.765	0.826	-0.035	0.781	-0.788	1.000	-0.147	0.698	0.627	-0.686	-0.689	-0.563
NRc	0.370	-0.004	-0.006	0.180	0.237	0.191	-0.110	-0.145	0.963	0.163	-0.098	-0.147	1.000	0.194	0.247	-0.187	-0.192	-0.307
BPR	0.589	0.008	0.014	0.676	0.640	0.758	-0.747	0.701	0.294	0.781	-0.766	0.698	0.194	1.000	0.669	-0.708	-0.704	-0.643
htBleed	0.567	0.003	0.012	0.630	0.600	0.703	-0.693	0.628	0.337	0.722	-0.703	0.627	0.247	0.669	1.000	-0.651	-0.659	-0.606
W31	-0.584	-0.006	-0.011	-0.662	-0.626	-0.748	0.736	-0.687	-0.285	-0.772	0.752	-0.686	-0.187	-0.708	-0.651	1.000	0.693	0.629
W32	-0.586	-0.015	-0.008	-0.668	-0.634	-0.745	0.737	-0.689	-0.293	-0.773	0.756	-0.689	-0.192	-0.704	-0.659	0.693	1.000	0.636
RUL	-0.736	-0.003	-0.002	-0.606	-0.585	-0.679	0.657	-0.564	-0.390	-0.696	0.672	-0.563	-0.307	-0.643	-0.606	0.629	0.636	1.000

Plot 5 : KNN results plotted with PCA result



Plot 6 : Discussion topic effect of Conditions.


Condition_3
Condition_2
Condition_1