#### A Project Report On

**Predictive Analytics for Equipment Maintenance**

**Prepared by**

**JAY VIJAYBHAI JOSHI**

# INDEX

|  |  |  |
| --- | --- | --- |
| **Chapter no.** | **Chapter Name.** | **Page No.** |
| **Chapter-1** | Problem Summary | 4  5  6  7  8 |
| **Chapter-2** | Introduction |
| **Chapter-3** | Aims & Objectives |
| **Chapter-4** | Literature Review & Prior Art Search |
| **Chapter-5** | Solution Strategy |
| **Chapter-6** | Study of current systems and Problems associated with them | 9 |
| **Chapter-7** | Dataset Description | 10 |
| **Chapter-8** | Data Preparation and Data Exploration | 12 |
| **Chapter-9** | DATA MODELING AND EVALUATION | 17 |
| **Chapter-10** | Advantages | 27 |
| **Chapter-11** | Conclusion | 28 |

## Abstract

"Predictive Analytics for Equipment Maintenance" project is developed to predict the Remaining Useful Life (RUL) of jet engines. Remaining Useful Life (RUL) is use to predict life-span of a components with the purpose of reduce the catastrophic failure events in both manufacturing and service sectors. Because the failure of engines is often cause of major accidents and casualties. So, the safety and the reliability of engines are vital to the performance of aircraft. However, it is very difficult to ensure their safeness and trustworthiness due to it’s complex structures, and engine failure has arisen inevitably due to effects of environment, aging, and variable loading as the working time increases. For this reason, it is necessary to detect degradation, predict how soon an engine will fail, implement maintenance process, and finally prevent catastrophic failure. The approach used here is data driven approach, That means the data collected from a jet engine by sensors is used to predict the Remaining Useful Life (RUL) of a jet engines. Data driven techniques can divided into two categories: statistical techniques and artificial intelligence techniques (neural networks, fuzzy systems, etc.).In this project we created many models , and we used run-to-failure data of many jet-engines to train the models in a way that it can predict the RUL of other jet engines by using jet engine’s data given by the sensors. Engine RUL prediction provides crucial data for maintenance choices. Additionally, in this project we mainly focused on predicting the Remaining Useful Life (RUL) of jet engines, and we trained the model for that thing. This way, we can restore assets to a state to continuously perform its design specifications. So in short this project helps to predict Remaining Useful Life (RUL) of jet engines, which helps to minimize the tragic or sudden failures and it’s helpful for making maintenance decisions.

#### 1.Problem Summary:

Jet Engines are the core component of jets, if a jet engine fails it may cause major accident and many casualties. So the safety and reliability of jet engines are very important.

But it is difficult to ensure their safety and reliability due to their complex structure, an engine failure may occur due to aging, effects of environment etc.to prevent the failure of a jet engine ,it is necessary to detect degradation , predict how soon the engine will fail , implement maintenance process.

But traditional maintenance of engines are either reactive that means fixing or replacing the jet engine component after failure ,or proactive that means speculating a certain level of performance degradation with no input from the jet engine itself and maintaining the jet engine on a routine schedule whether maintenance is actually needed or not .both are inefficient and quite wasteful.

#### 2.Introduction

As the main part and power source of jet, the reliable operation of an engine is critical for ensuring the reliability and safety of the jet, and to maintain its availability, and reduce its maintenance costs. "Predictive Analytics for Equipment Maintenance" project is developed to predict the Remaining Useful Life (RUL) of jet engines. As we mentioned above RUL indicates the component lifetime before it can no longer perform its function, which is also an important way to minimize the production loss, save maintenance costs and avoid serious machine breakdowns of the equipment before its failure.

Approaches to predict system life time can be broadly categorized into three types: physics based models, data driven approaches and hybrid approaches. However, it is too expensive to apply a physics-based model to a system like this which is quite complex. Besides, this approach has shown notable limitations due to the assumptions and simplifications of the adopted models. Data driven approach provides accurate Remaining Useful Life( RUL)predictions for a complex system, which can be applied quickly and cheaply compared to the physics-based model. Data driven techniques can divided into two categories: statistical techniques and artificial intelligence techniques (neural networks, fuzzy systems, etc.).

The approach used here in this project is data driven approach. The sensors collects the data from jet engines and those data are use to predict the Remaining Useful Life (RUL) of the jet engines. But for that In this project we created many models, and we used run-to- failure data of many jet-engines to train the models in a way that it can predict the Remaining Useful Life (RUL) of other jet engines by using jet engine’s data given by the sensors. Now this trained model can predict Remaining Useful Life (RUL) on unknown data of any other jets.

In addition to that in this project we mainly focused on predicting the Remaining Useful Life (RUL) of jet engines, and we trained the model for that thing by training the model on the run-to-failure data of the equipment which we wanted to predict. So in this project we can predict the Remaining Useful Life (RUL) of jet engines by using deep learning, artificial intelligence to ensure the engine integrity and safety, to reduce production loss, save maintenance costs and making wise maintenance decisions.

#### 3.Aim & Objective:

The main Aim of **Predictive Analytics for Equipment Maintenance** is to predict the Remaining Useful Life (RUL) of jet engines to predict how soon an engine will fail, which helps to minimize the tragic or sudden failures and it’s helpful for making maintenance decisions and it also ensure the engine integrity and safety.

#### Objective

* Predict the RUL of jet engines
* Avoid fatal machine breakdowns
* Save maintenance costs.
* Reduce production loss

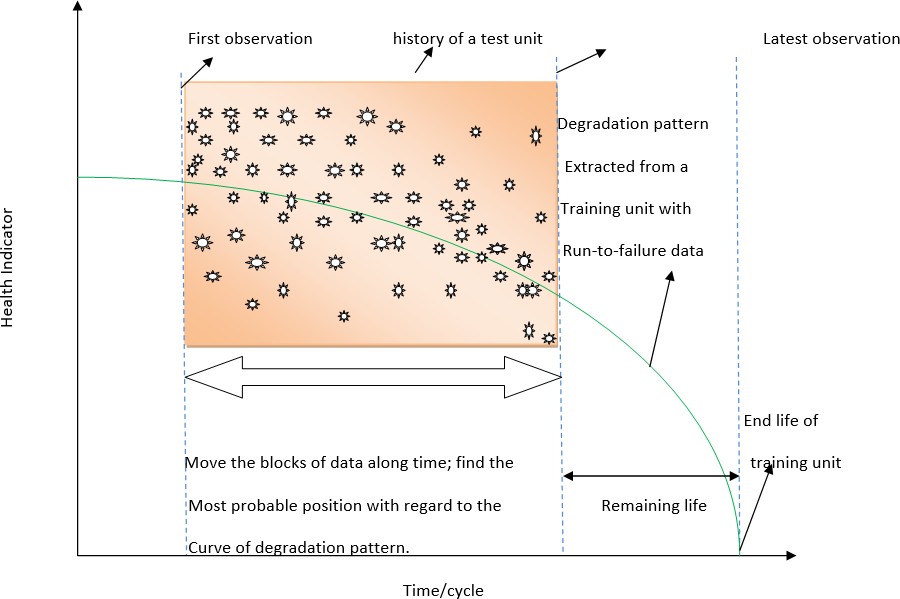
#### 4.Literature Review & Prior Art Search

We wanted to develop an accurate models to predict the remaining useful life of equipment , so we read many papers and articles from different many websites. We referred number of patents from google patents to understand the concepts and basic stuff about the predictive analysis and to understand it’s use and importance. We wanted to learn about keras (it is an open-source library that gives a Python interface for artificial neural networks) and matrix as this thing are necessary for our project so we read articles from google python class, edureka etc.

It’s is very complex and costly to collect data of so many jet engines which we can’t afford , so that’s why we used the Kaggle version of the very well known data set for asset degradation modeling from NASA. It includes Run-to-Failure data from turbo fan jet engines. In that engine degradation simulation was carried out using C-MAPSS. Records many sensor channels to characterize fault evolution.

We read few articles named “How Neural Networks process input data”, “Artificial Intelligence vs. Machine Learning vs. Deep Learning: What’s the Difference”, “Why Data Normalization is necessary for Machine Learning models” on medium.com site to learn how data normalizes and process in neural network. After searching for the similar system like ours, the systems who works on predictive analytics. We realized that there are not many systems that provides this types of services which we are trying to provide, most of them are heavily focused on stock markets and other financial activities, and that thing encourages us more to develop this project.

We also read the paper named “Predictive maintenance for industrial products using big data”. Which is about cloud-based predictive maintenance service which collects the data from multiple industrial client for storage and analysis on a cloud platform. We read another the article named “Jet Engine Remaining Useful Life (RUL) Prediction” on medium.com website in order to properly understand our project. The main aim of that article is to document the implementation of a model that can be used to perform predictive maintenance on commercial turbofan engine. the predictive maintenance approach used there is a also data-driven approach, meaning that data collected from the jet engine is used to perform predictive maintenance.to be specific, the project goal is to build a model to estimate the RUL of a jet engine based on run-to-failure data of a similar jet engines.



**5.Solution Strategy:**

the main strategy will be to use the dataset to train the regression model to predict The Remaining Useful Life (RUL). since the data is in the form of a time trajectory of many sensor data, then it requires to fused these sensors into a condition indicator that help in identifying the occurrence of a failure.

When the model is in the testing mode it will do the comparison that how similar the testing fused signal to the training fused signal. And based on that similarity, a prediction is made.

Since the training data consists of run-to-failure data, whereas the testing data contains data up to undefined health state, then the training process will include training the model on a part of the data before the failure appears to simulate the use of the model in prediction mode. for detailed strategy, please refer to a following figure.

**Figure-1: RUL prediction strategy**

### 6.Study of current systems and Problems associated with them:

We studied and analyze The current popular systems like ours, and we found out that there are very few systems for predicting the Remaining Useful Life (RUL)of the equipment, there are many predictive analytics system for market and other financial purposes but not many systems focuses on predicting RUL of equipment only, So we decided to create a system whose main focus is to predict RUL of equipment.

**7. Dataset Description:**

The commercial modular aero-population system simulation (C-MAPSS) developed at NASA datasets is one of the most commonly used benchmark datasets in the remaining useful life (RUL) prediction; they contain four sets of monitoring data (FD001-FD004) collected under various engine operating conditions [[1].](#one) There are a train, a test, and an RUL set in every batch of monitoring data. The training datasets, which consist of complete run-to-failure data, are used to train the algorithm to predict RUL. On the other hand, a small amount of data can only be used to construct test datasets until the method is unsuccessful. Multiple turbofan engines were simulated over time in the datasets, and the following data was included in each row: number of the engine unit, time (in cycles), three operational settings, and 21 sensor readings as seen in [Table 1](#table_1)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Dataset | Operating conditions | Fault modes | Train dataset size | Test dataset size |
| |  | | --- | | FD001 | | 1 | 1 | 100 | 100 |
| |  | | --- | | FD002 | | 6 | 1 | 260 | 259 |
| FD003 | 1 | 2 | 100 | 100 |
| FD004 | 6 | 2 | 248 | 249 |

Table 1: Overview of Turbofan Datasets [[2]](#two)

Each column represents a different variable and each row represents a snapshot of data received during a single cycle of operation. Four different operation scenarios are modeled using a set of trajectories. Only the first sub-dataset which is FD001 considered here. The unit number and operating time which is in cycles are given in the first two columns, while operational parameters are given in the third, fourth, and fifth columns. The remaining columns in [Table 2](#table_2) are sensor measurements, as can be seen. There are 21 different types of sensors in each engine, including pressure, temperature, and shift speed sensors, which gather information on various engine parameters [[4]](#four). The sensors’ capacity to record engine performance decline is the key thing for achieving RUL prediction.



Table 2: Overview of FD001 Sub-Dataset

**Sensor Selection:**

In an aircraft engine, different sensors give different responses to the performance degradation process. Due to noise or lack of sensitivity to degradation trends, some sensors exhibit ambiguous patterns or trends. The accuracy of RUL prediction may be reduced if insensitive data has been chosen. In order to increase the RUL prediction models’ accuracy, sensors that are more sensitive to performance degradation are chosen as inputs. For this, there is a method called slope analysis which is used for sensitivity measurement. Slope analysis has three main steps which are as follows [[3]](#three)

Step 1: In the initial step, curve fitting is performed on the degradation data for each parameter of a particular engine. After that, the sensitivity of the degradation data is examined using the slopes, which are the parameters of the best-fit curves.

Step 2: In the second step, the mean values of all the parameters of each engine in Step 1 that belong to the same sensor are calculated. Next, the different average parameter values for the different sensors demonstrate the distinct sense of the degradation data.

Step 3: In the last step, for determining the engine’s RUL, degradation data with higher slopes are chosen.

### 8.Data Preparation and Data Exploration:

In Our Project data exploration and data preparation are often intertwined and involved going back and forth between the two stages, thus we decided to make a combine report of both the stages as it would provide better understanding of the process.

Bad data ruins the model,The saying 'Garbage In, Garbage Out', has been in existence since the invention of computers. But since the invention of machine learning and AI (with their strong need for data quality), the saying is more relevant than ever. This project is highly data-sensitive. For this reason, the quality of data being used in process has a significant impact on its success. Solving complex problems demands a lot of rich, quality data, For predictive analytics that solves complex problems, Model is useless if the data is bad. Thomas C. Redman said that wrong data is the leading enemy to the profitable and widespread use of machine learning. Training data governs the performance of this model, So bad data return bad results and giving out incorrect information. some examples of bad data are: Incomplete or missing data, Incorrect data, Biased data. So we have to do some processes like data normalization, removing ambiguities manually etc. to solve this issue.

It is a time consuming and least enjoyable process as we need to do processes like data pre-processing and data visualization which is one of the most tedious and crucial processes. In simple words, It transforms the data and enriches it to improves the accuracy of the outcome.

Here, dataset required to train machine learning and deep learning algorithm is gathered from different sensors of machine at fixed interval time. Sequence. of data-point is collected from various sensors and our model is trained on that.The dataset we have used in this project is Turbofan Engine Degradation provided by NASA. This dataset consist of 3 settings and 21 sensors. Three operations settings are Altitude, Mach Number and Throttle Resolver.

**Also Different type of sensors used are as follow :**

\*Pressure sensors

\*Temperature sensors

\*Optical sensors

\*Chemical sensors

\*Water quality (pH, conductivity, turbidity)

\*Smoke sensors

\*Level sensors (liquid and solid materials)

\*Gas sensors

\*Accelerometer

\*Humidity sensors

\*Gyroscope sensors

Also it contains 3 settings on which machine was operating during that time. Train-data consist sensor and setting data of total 100 turbofan engine taken at fixed interval of time until failure occur & Testdata consist of values of each sensors taken at fixed interval of time without failure event recorded.

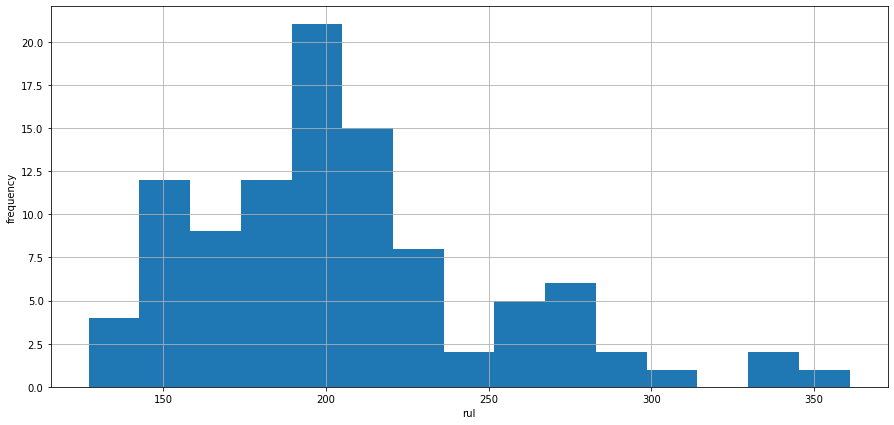
In the data exploration, we first read the CSV file into the data Frame which were train\_df,test\_df and y\_test respectively.After that we performed operations like checking for outliers between column 2 and 25 and visualizing the graphs so that we can identify the trends and zero variance columns in the dataset moreover,other insights regarding the changing pattern can also be observed by this.

Then we dropped all the columns of train\_df which were having NAN values as they were insignificant and later we renamed rest of the columns for better understanding.

Furthermore, we added two column which named as unit and cycles. Then found the max of cycles and later perform max minus cycles and added rul column.We performed deletion of insignificant columns and renaming operation on test\_df too,also,we called the function which use to make new rul column which is used to find remaning useful life cycles from test\_df data set.

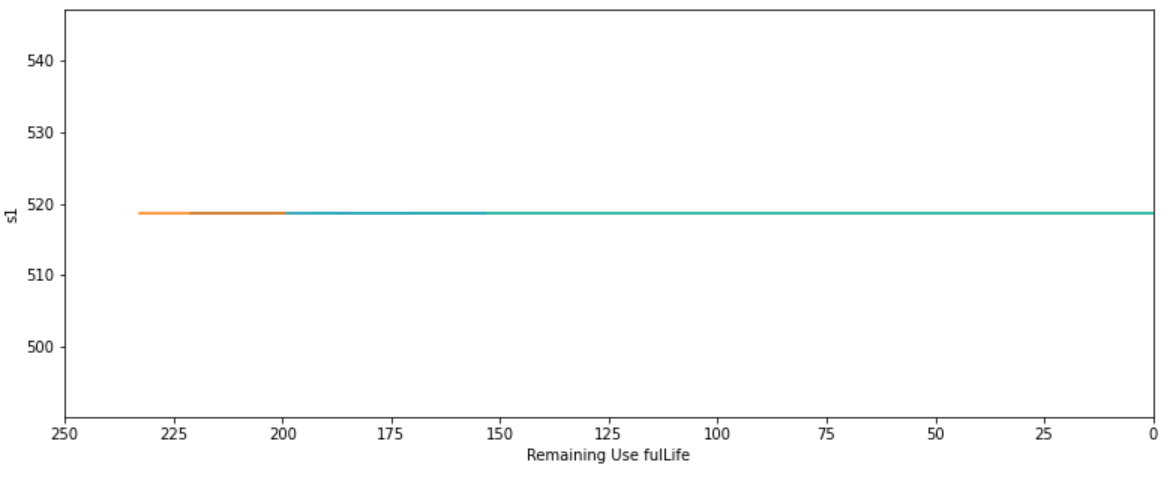
**Plotting:**

plotting allow us to understand the dataset better,that’s why we plotted histogram of max rul and signals of each sensors.

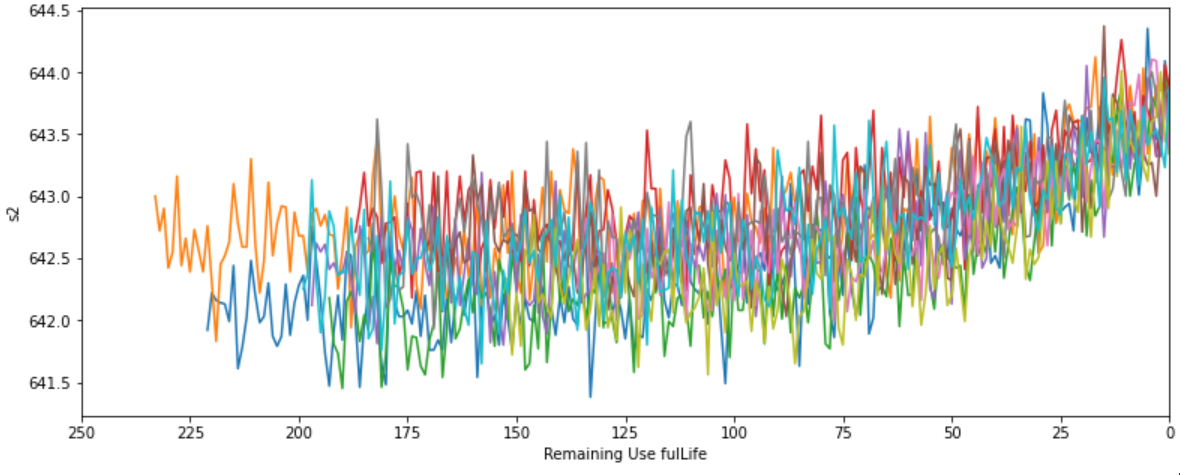


Histograms confirms that the most engines collapse around 200 cycles and for those engines last over 300 cycles the distribution is right skewed.

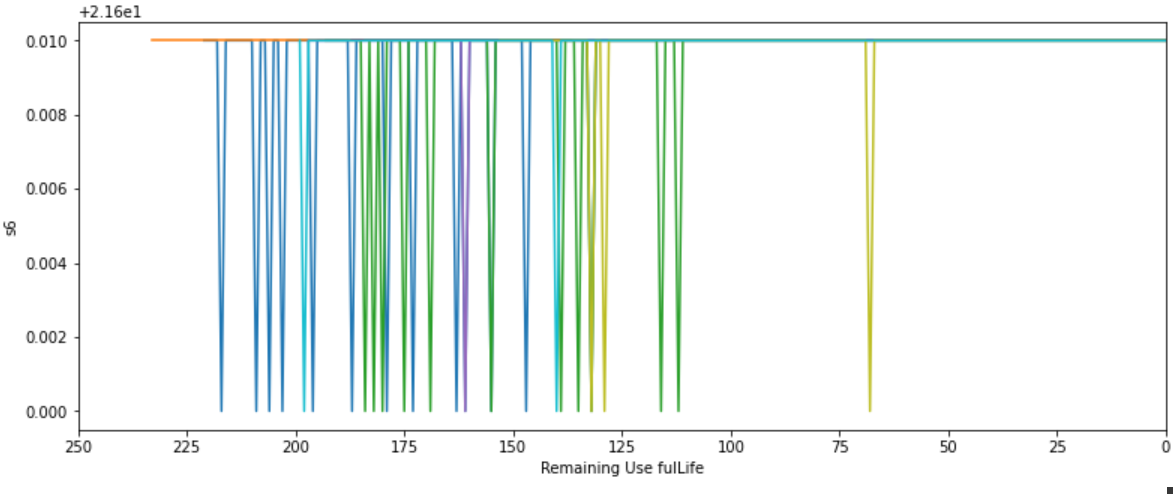
Because many number of engines, plotting every engine for every sensor is not possible. Thus, we decided to plot engines whose unit is divisible by 10 with a remainder of 0. We revert the X-axis so RUL declines along the axis, and when RUL is zero it indicates the engine failure.Here, because of so many sensors we’ll discuss about the few graphs which are representative for the entire set.



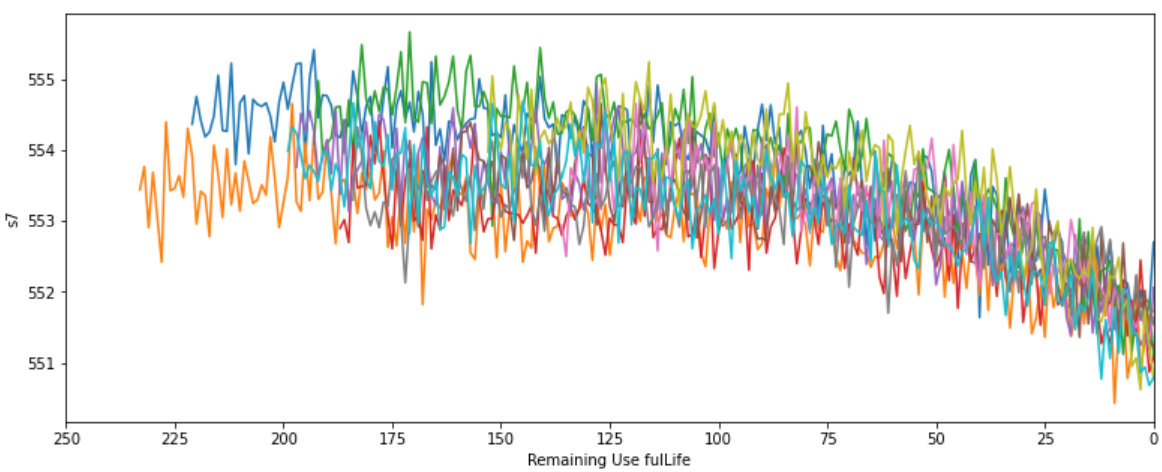
Here,graphs of sensors 1, 10, 18 and 19 looks quite similar to each other, and flat line in the graph indicates that these sensors do not contain any useful information.



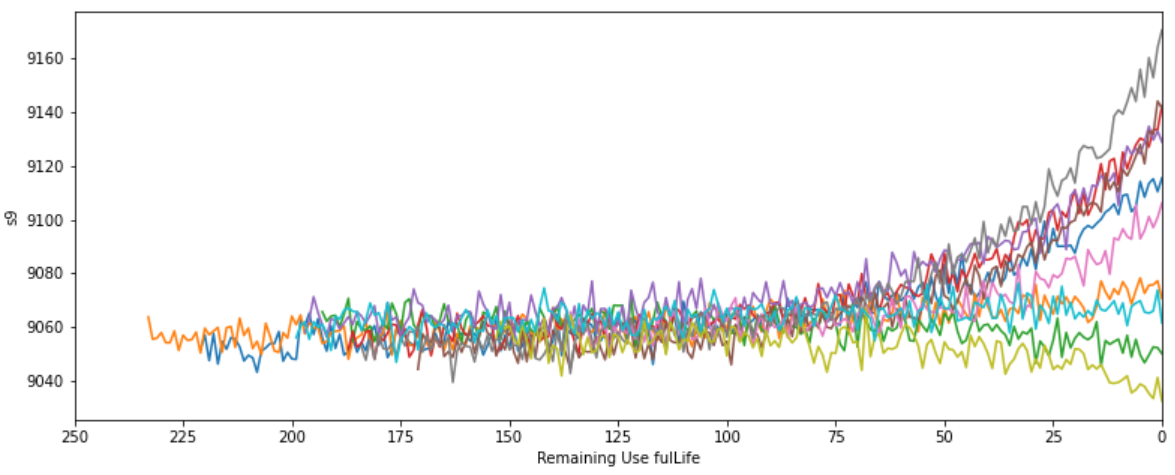
Furthermore,there is a rising trend in sensor 2 which can be seen in the above graph, this type of similar pattern can also be seen for sensors 3, 4, 8, 11, 13, 15 and 17.



sensor 6 shows that it peak downwards at times but there isn’t any clear relation to the decreasing RUL.



there is a declining trend in sensor 7 which can be seen in the above graph, this type of similar trend can also be notice for sensors 12,20 and 21.



And here, sensor 9 has same patterns as sensor 14.

Hence, Based on our Analysis we noticed that sensors many sensors contained no information about the RUL because the values of those sensors were remain the same throughout time.

So to deal with this situation we made a new function whose name is std\_drop , this function is use to drop all the column whose values of standard deviation are less than threshold 0.0001.we called the function to do same thing with test\_df, of dropping column whose values of standard deviation is less than threshold.Furthermore,we made new function called correlation, which is use to find the correlation between the all the column,moreover, we made a set of all columns, and use the 'for' loop to find the correlation between all columns, also we made threshold of 0.9,later we delete all those column whose correlation was greater than 0.9 .We used the same correlation function for the test\_df too.After that we Normalized the data using Min-max scalar given in the sklearn preprocessing from cycles to RUL, here,RUL not included use slicing operation for both test and train dataset.



Before Normalization

Graphical user interface, application

Description automatically generated

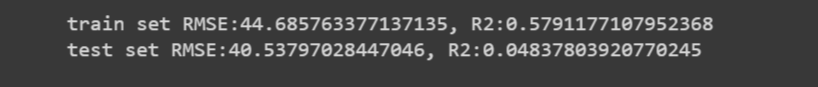
After Normalization

After Normalizing both the training set and testing set we appended new column TTF(time to failure),TTF calculates the time from when an item or a product was putted into the service until it fails. It can be expressed as cycles, hours,days and so on.Later,we vectorized the data and converted data into an array and checked how many life-to-death cycles we do have and what do they look like?

Chart, line chart

Description automatically generated

Then,we defined a function in order to evaluate our models and we decided to include Root Mean Squared Error (RMSE) because it provides an indication that how many time cycles predictions are off on average, and described Variance to show what fraction of our variables which are dependent can be explained by the independent variables that we use.We dropped the unit, cycle, op\_settings and sensors which didn’t contained any information for both training and test set. RUL column is stored in its own variable for our training set . Moreover, last time cycle of each engine is only thing important for us in the test set because we only have True or real RUL values for them.Placing a linear regression is a simple task to perform as we instantiate the model by just calling method of linear regression and assigning it to the variable. Later , we predicted on both sets(test and train) in order to understand the behaviour of our model with the data.



To recapitulate , we performed data exploration and created a baseline model with a test RMSE of 40.5379.we tried to transformed the data by removing missing and insignificant data in order to improves the accuracy of the outcome,also in this whole process we created many functions like std\_drop, correlation, fractionTTF,evaluate and so on in order to reach our goal which was data enrichment.Moreover, through out the whole processes we tried to practiced Pair programming which allowed us to improved code quality and helped us in making less mistakes as a consequences we encountered less bugs also, it allowed us to share information naturally during the entire process.

**9. DATA MODELING AND EVALUATION:**

**PREDICTION OF RUL USING ARTIFICIAL NEURAL NETWORK MODEL**

We decided to implement multiple models to get the best performance from our data. One of them is Artificial Neural Network.

As there are only 17 input parameters, we created a small ANN. It has 2 hidden layers with 34 and 17 neurons respectively. Input layer was separate with 17 parameters. For the output layer only 1 neuron is used as this is a regression problem. We used ReLU as our activation function on both the hidden layers to increase the non linearity in the model. We also used Adam optimiser. The main reason for this was the inputs. One input parameter (Cycle) was not normalised.

If another optimiser like SGD is used then it will lead to bias in the system as well as it may also face exploding gradient problem.

**TEST MODEL 1**

The output for this model was 100% accuracy and 94% validation accuracy, which may be considered overfitting.

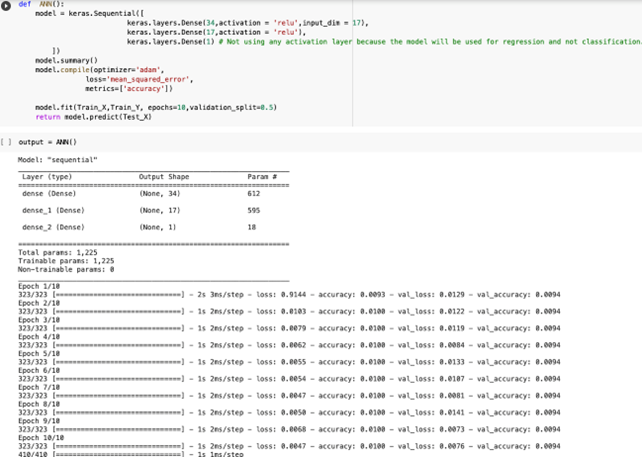


FIG 12 – ANN VERSION 1

So one of the parameter is changed for the next model. We changed the validation split from 0.5 to 0.3 so that it gets more data to train with.

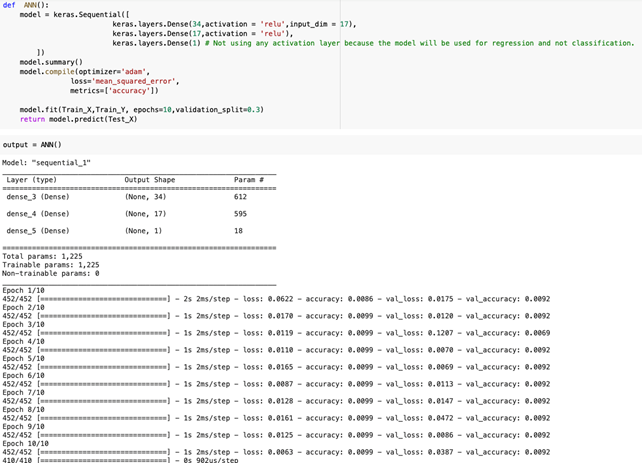


FIG 13 – ANN VERSION 2

**TEST MODEL 2**

The output for this model was 99% accuracy and 92% validation accuracy, which may be considered overfitting.

We also tried the SGD optimiser, and the result were as expected. We saw exploding gradient problem in that.

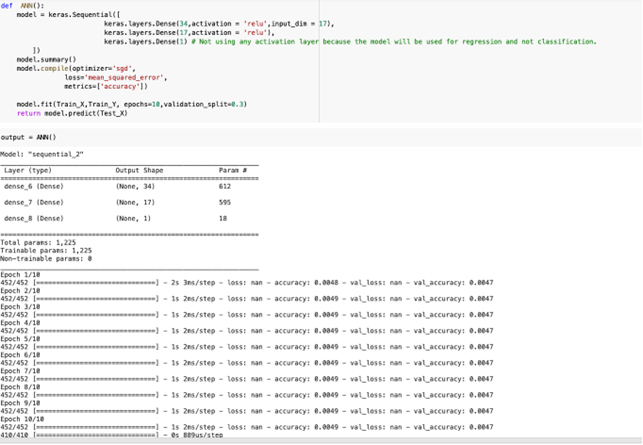


FIG 14 – ANN VERSION 3

**TEST MODEL 3**

Decided to stick with the **2nd model as final model** to test the accuracy with the testing data.

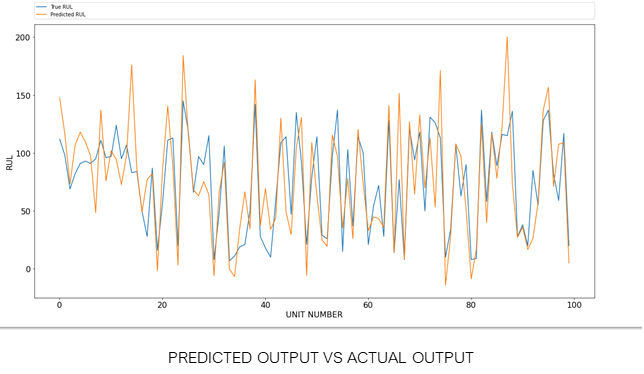


FIG 15 – ANN VERSION 3 PREDICTED VS ACTUAL GRAPH

Some necessary operations were done on the predicted output so that they could be compared with actual output.

Here we can see that our predicted output does follow the general trend of the actual output.

For the evaluation, we found value of RMSE as 30, and R2 as 0.47. R2 is used as an accuracy measure. 0.47% means that our accuracy on the test set is only 47% but nothing much can be done as the training data output value was not accurate and only a guess of the real value. But in the testing data actual output values are provided. So this is under expectations as well.



FIG 15- ANN ACCURACY

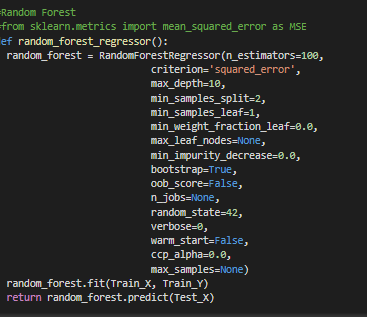
RMSE 30 means that our predicted output is within 30 cycles of the actual output which can be considered good.

# **DATA** **PIPELINE**

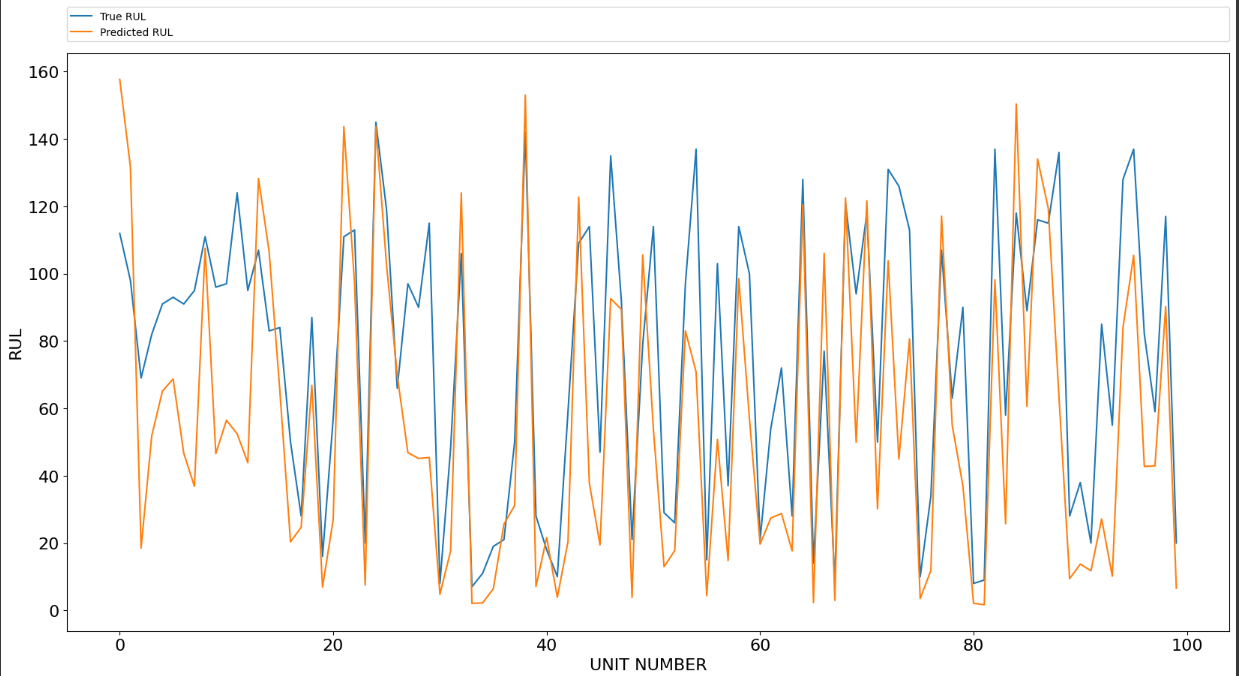
The original dataset is randomly separated into train data and test data, and the dataset is separated into training datasets (Train\_X, Train\_Y) and testing datasets (Test\_X, Test\_Y). The datasets with the prefix “\_X” are inputs to the model, whereas the datasets with the prefix “\_Y” are outputs to the model.

# **RANDOM FOREST**

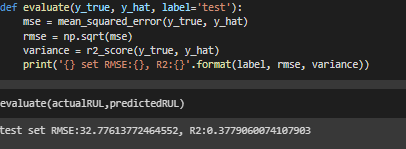
The machine learning model used to solve the regression problem is the Random Forest (RF). RF is a bagging method of ensemble model that develops multiple decision trees to make a final prediction. 100 subsets of train data are prepared using bootstrapping. These subsets of data are used to build 100 decision trees. The parameters of RF are 100 estimators, each node must have one leaf and each sample must split twice. The prediction is done based on the average 100 trees prediction. The maximum depth is restricted to 10 to reduce the complexity and overfitting. The error metric used to evaluate the RF model performance is a squared error. The figure below shows the Python code of RF executed using sci-kit in Python.

  
FIG 16 – RANDOM FOREST

Below shows the trends predicted (orange) and true RUL (blue) of the test which vary more with less and a high number of units. The Test data was not shown to the model while training helps the model to show the correctness of the predicted value.

  
FIG 17- RANDOM FOREST ACTUAL VS PREDICTED GRAPH

The RMSE and R2 (accuracy) of the RF are 32.77 and 37, respectively. The performance is not up to expectations but with respect to the RMSE the life cycle will be within 32.

  
FIG 18 – RANDOM FOREST ACCURACY

**XGBOOST**

Extreme Gradient Boosting (XGB) is a distributed gradient-boosted decision tree algorithm. It gained popularity with fast training and better performances compared to other machine learning models [1]. It is a boosting method of the ensemble model . It is based on the gradient boosting framework, which involves adding new models to the ensemble that corrects the errors made by previous models. A low gradient rate will create many trees to reduce the loss function, so with different trials(cross-validation), we choose to keep the learning rate at 0.07. To build the best tree we use n\_estimators as 450 although it built a higher number of trees it improves the learning. The error metric used to evaluate the XGBoost model performance is a mean squared error.  For generalized performance reg\_alpha is set to 0.75 this is identified by cross-validation. To build a less complex tree and reduce the overfitting the growth policy is set to loss guide (level wise). XGB with the above parameters is trained on Train\_X and Train\_Y. The trained XGB is evaluated by predicting on test split (Test\_X) comparing with true value (Test\_Y) for error.

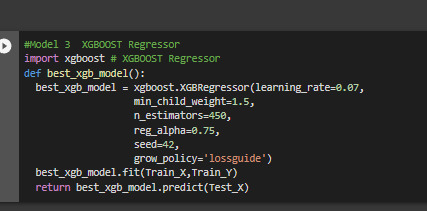
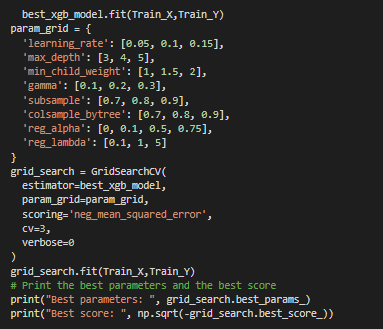
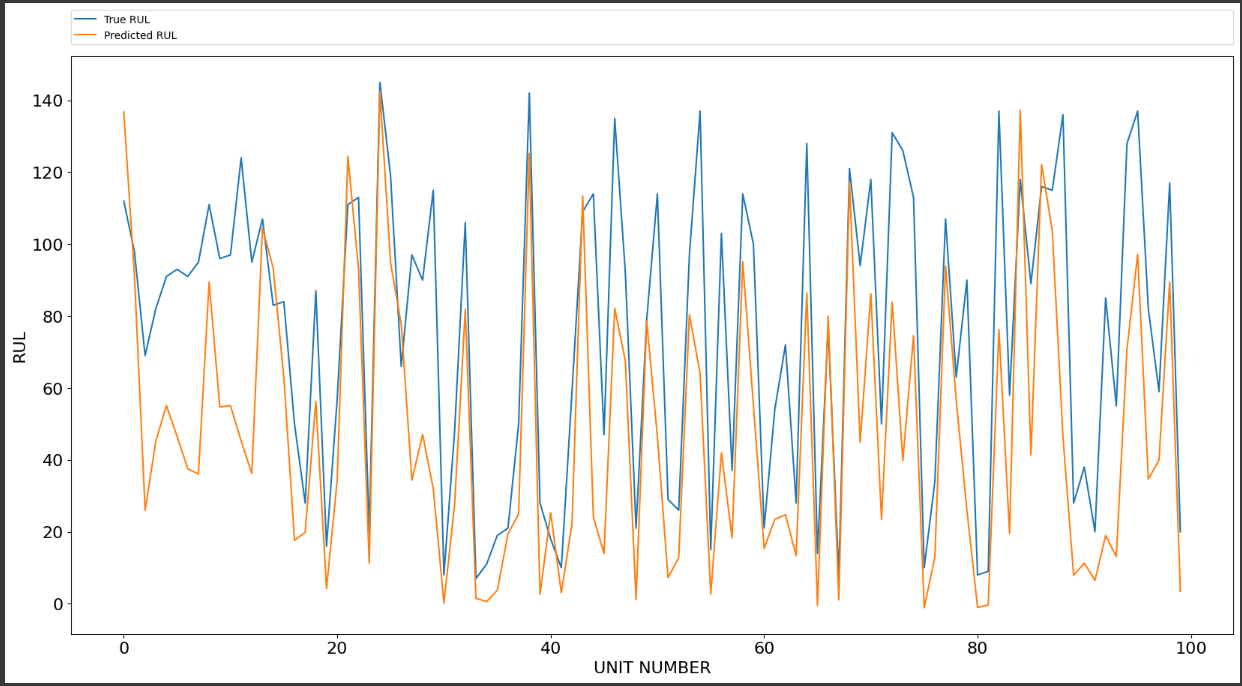


FIG 19 – XGBOOST

  
FIG 20 – XGBOOST

The plot below illustrates the trends of the predicted value and real-time RUL.

  
FIG 21 – XGBOOST ACTUAL VS PREDICTED GRAPH

The RMSE and R2 (accuracy) of the XGBOOST are 37.00 and 20.6, respectively. The performance is not up to expectations but with respect to the RMSE, the life cycle will be within 37.

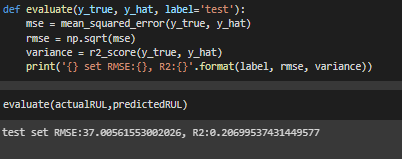


FIG 22 – XGBOOST ACCURACY

**MODEL DEPLOYMENT**

**WEB APPLICATION**

Finally we have create a web application to access our model and predict the output. The web app is created using streamlit library of python. We have provided the user to input the data and the output is displayed after user is done.

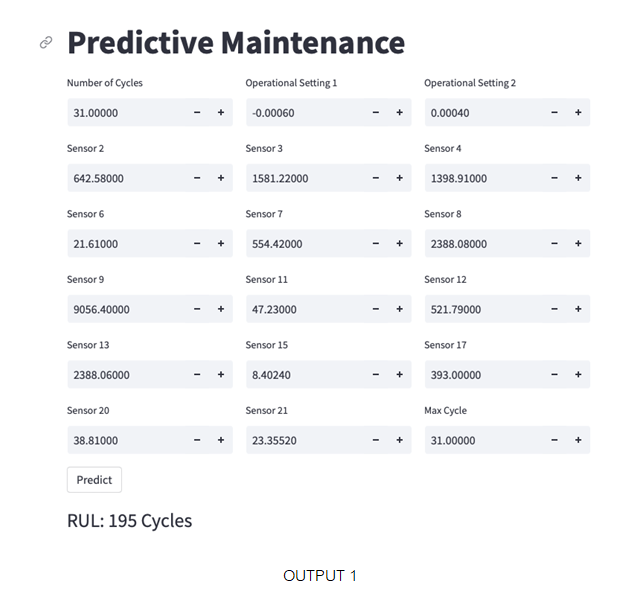


FIG 23 – WEBAPP

Also deployed the webapp on streamlit.

[Text Wrapping Break]Here is the link: <https://xgagandeep-cs711-team1-model-pvjvf9.streamlit.app/>

### 10.Advantages:

The main advantage of “Predictive Analytics for Equipment Maintenance” are as follows:

**Predict Remaining Useful Life (RUL)** The main advantage of “predictive analytics for equipment maintenance” project is that it predicts that how soon an engine will fail, So that we can implement maintenance process, and ultimately prevent catastrophic failure.

**Provides important information for maintenance decisions** traditionally maintenance of engines are either purely reactive that means fixing or replacing the jet engine component after failure, or blindly proactive that means guessing a certain level of performance degradation with no input from the jet engine itself and maintaining the jet engine on a routine schedule whether maintenance is actually needed or not. both are inefficient and quite wasteful. But by using predictive analytics for

equipment maintenance we can predict the RUL and know when the engine might stop functioning which helps a lot to make a maintenance decisions because if we know RUL of engine, we can tell that does it required maintenance or not.

**Reliability and safety of the jet** By predicting RUL of engine we Can take right maintenance decisions and that good decisions will make the jet more reliable and safe.

**Save maintenance costs and Reduce production loss** By predicting RUL of engine we can take right maintenance decisions , we can avoid unnecessary maintenance process and we do maintenance process on the right time when it really requires, so by using this project we can do the calculated maintenance and by doing that we save unnecessary maintenance cost and can reduce the production loss.

### 11. CONCLUSION

We are created the project entitled “predictive analytics for equipment maintenance”. The main purpose of this project is to predict the Remaining Useful Life(RUL) of jet engines.We develop this project because as we mentioned earlier there are not many systems focuses on predicting RUL of equipment , So we decided to create a system whose main focus is to predict RUL of equipment. We done all of the above activity and for this project and tried to make it error free, robust.

### References:

##### We Referred the Following:-

1. For NASA Turbofan Jet Engine Data Set: <https://www.kaggle.com/behrad3d/nasa-cmaps>
2. For Understanding How Neural Networks process input data: <https://medium.com/ai-in-plain-english/my-notes-on-neural-networks-adf3e49657f8>
3. For learning the difference between Artificial Intelligence, Machine Learning, Deep Learning:

[https://medium.com/ai-in-plain-english/artificial-intelligence-vs-machine-learning-vs-deep-learning-](https://medium.com/ai-in-plain-english/artificial-intelligence-vs-machine-learning-vs-deep-learning-whats-the-difference-dccce18efe7f) [whats-the-difference-dccce18efe7f](https://medium.com/ai-in-plain-english/artificial-intelligence-vs-machine-learning-vs-deep-learning-whats-the-difference-dccce18efe7f)

1. For getting the understanding about the Jet Engine Remaining Useful Life (RUL) Prediction project: [https://medium.com/@hamalyas\_/jet-engine-remaining-useful-life-rul-prediction-a8989d52f194](https://medium.com/%40hamalyas_/jet-engine-remaining-useful-life-rul-prediction-a8989d52f194)
2. For understanding about Why Data Normalization is necessary for Machine Learning models:

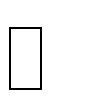
[https://medium.com/@urvashilluniya/why-data-normalization-is-necessary-for-machine-learning-](https://medium.com/%40urvashilluniya/why-data-normalization-is-necessary-for-machine-learning-models-681b65a05029) [models-681b65a05029](https://medium.com/%40urvashilluniya/why-data-normalization-is-necessary-for-machine-learning-models-681b65a05029)

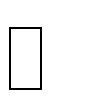
1. For reading about Predicting the Remaining Useful Life of an Aircraft Engine Using a Stacked Sparse Autoencoder with Multilayer Self-Learning:

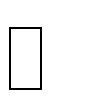
<https://www.hindawi.com/journals/complexity/2018/3813029/>

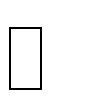
1. For learning about python: <https://www.edureka.co/> <https://www.learnpython.org/>
2. And we also referred: <https://www.youtube.com/> <https://www.google.com/>

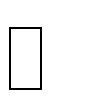
<https://colab.research.google.com/drive/1E-572Bww1PCN4t_Kw96IjMz_W_0j5dsl>

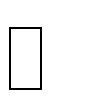
 A. Saxena, K. Goebel, D. Simon, and N. Eklund, Damage Propagation Modeling for Aircraft Engine Run-to-Failure Simulation, in the Proceedings of the 1st International Conference on Prognostics and Health Management (PHM08), Denver CO, Oct 2008.

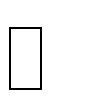
 T. Wang, J. Yu, D. Siegel, J. Lee, “A similarity-based prognostics approach for remaining useful life estimation of engineered systems”, Proc. Int. Conf. Prognostics Health Manage., pp. 1–6, Oct. 2008.

 Asenjo, Juan L., John Strohmenger, Stephen Thomas, Nawalaniec, Bradford Henry Hegrat, Joseph A. Harkulich, Jessica Lin Korpela, Jenifer Rydberg Wright, Rainer Hessmer, John Dyck, Edward Alan Hill, and Salvatore T. Conti. "US20140336791A1 - Predictive Maintenance for Industrial Products Using Big Data." Google Patents. Google, 13 Nov. 2014. Web.

 Sopko, Victor , III, and James McMeal Buckley. "US7249030B2 - Method and System for Providing Maintenance and Management Services for Long-term Capital Equipment or Fixtures by Providing a Performance Warranty." *Google Patents*. Google, 24 July 2007. Web.

 Garrow, Gary R., et al. “US7457763B1 - Predictive Maintenance System.” Google Patents, Google, 25 Nov. 2008, patents.google.com/patent/US7457763B1/en.

 Ridolfo, Charles F. “US6735549B2 - Predictive Maintenance Display System.” *Google Patents*, Google, 11 May 2004, patents.google.com/patent/US6735549B2/en.

 DELAYE, Guillaume, et al. US8868287B2 - Prediction of Maintenance Operations on an Aircraft Engine. Google Patents, Google, 21 Oct. 2014, patents.google.com/patent/US8868287?oq=predictive+analy+tics+for+equipment+maintenan ce+for+jet.