MACHINE LEARNING-BASED PREDICTIVE ANALYTICS FOR AIRCRAFT ENGINE

A PROJECT REPORT

Submitted by

NUTHALAPATI SEVITHA

JEELA SINDHU

BONDU VENNELA

PAVITRA L

BACHELOR OF ENGINEERING

In

COMPUTER SCIENCE AND ENGINEERING

GRT INSTITUTE OF ENGINEERING AND TECHNOLOGY

ANNA UNIVERSITY::CHENNAI 600025

CONTENTS

Machine Learning-Based Predictive Analytics for Aircraft Engine

Team ID	**PNT2022TMID36087**
Project Name	**Machine Learning-Based Predictive Analytics for Aircraft Engine**
Team Members	**Nuthalapati Sevitha (Team Leader)** **Jeela Sindhu (Team Member 1)** **Bondu Vennela (Team Member 2)** **Pavitra L (Team Member 3)**

1. INTRODUCTION

1.1 Project Overview

Machine learning is a branch of artificial intelligence that uses statistical technique and mathematical algorithms to enable a machine to learn from data, to analyze data patterns, and to make decisions with minimal human intervention. Data is now the most valuable asset for enterprises in every industry. Companies are using data-driven insights for competitive advantage. With that, the adoption of machine learning-based data analytics is rapidly taking hold across various industries, producing autonomous systems that support human decision-making. This work explored the application of machine learning to aircraft engine performance prediction.Supervised machine-learning algorithms for regression and classification were employed to study patterns in an existing, open-source database of production and research turbofan engines, and resulting in predictive analytics for use in predicting performance of new turbofan designs.

1.2 Purpose

Predictive analytics help us to understand possible future occurrences by analyzing the past.Predictive odelling solutions are a form of data-mining technology that works by analyzing historical and current data and generating a model to help predict future outcomes. Machine learning, on the other hand, is a subfield of computer science that, as per Arthur Samuel's definition from 1959, gives 'computers the ability to learn without being explicitly programmed'.

2. LITERATURE SURVEY

2.1 Existing Problem

The majority of the returns we receive from the field are found not to be issues with the turbocharger itself, but in most cases they are problems with the system’s installation, inadequate pre-lubrication, or other operational issues.Typically a mechanic must inspect and diagnose operational issues that may include an inability for the aircraft to reach altitude; pressurization issues; the system’s inability to reach the maximum-rated manifold pressure; a surging or dropping off of manifold pressure when climbing or descending; and/or oil leaks from the compressor or turbine side of the turbocharger.

2.2 References

1. http://www.jet-engine.net/civtfspec.html.
2. https://www.geaviation.com/commercial
3. https://www.pw.utc.com/productsandservices/products/commercial-engines
4. https://www.rolls-royce.com/products-and-services/civil-aerospace
5. https://www.cfmaeroengines.com/
6. http://cs229.stanford.edu/notes/cs229-notes3.pdf
7. https://www.coursera.org/learn/machine-learning

2.3 Problem Statement Definition

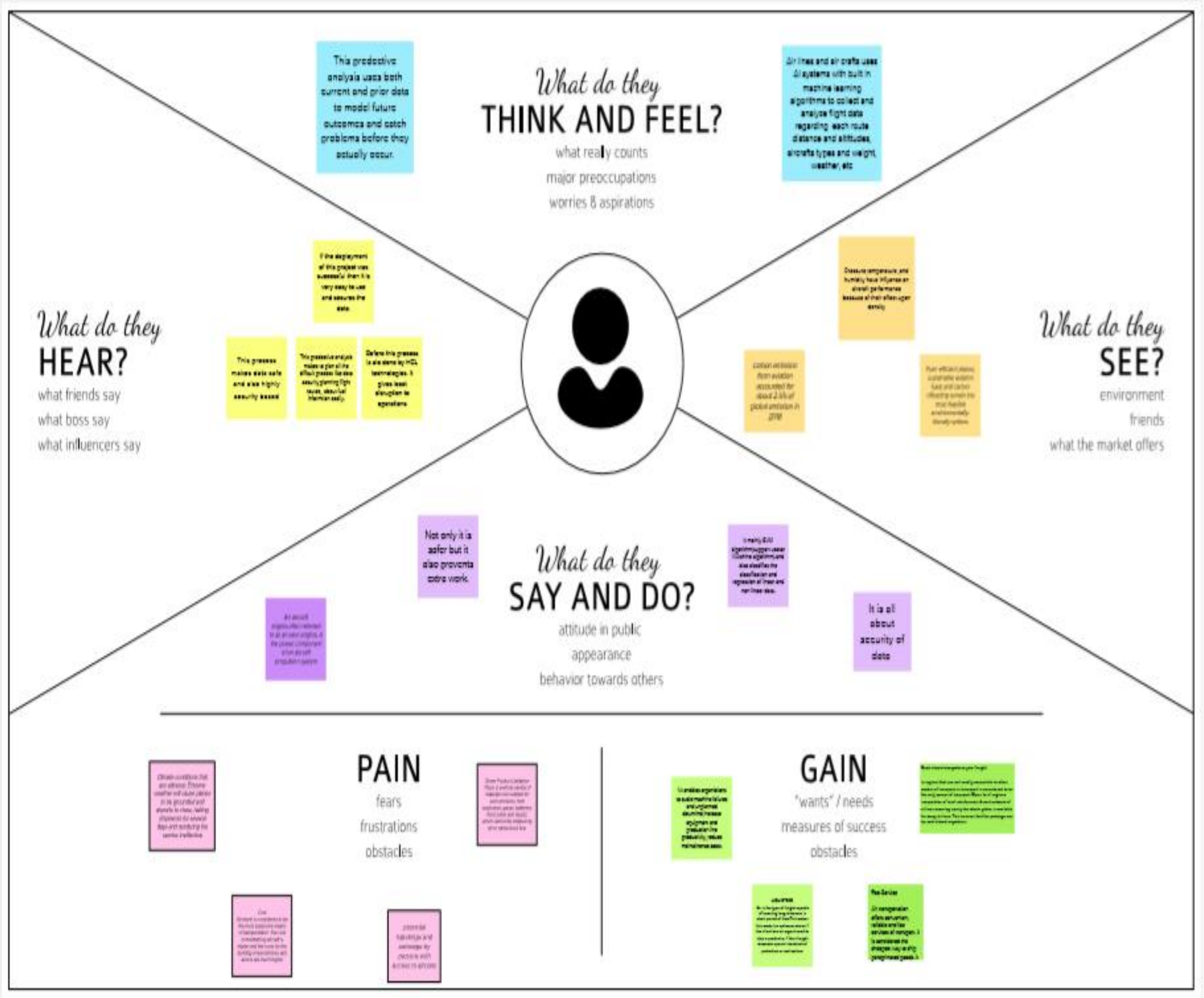
This article aims to prove that Machine Learning (ML) methods are effective for Predictive Maintenance (PdM) and to obtain other developing methods that suitable applied on PdM, especially for aircraft engine, and potential method that can apply on future research, and also compared between articles in International and Indonesia institution. Maintenance factors are important to prognostic the states of a machine. PdM is one of the factor strategies based on real time data to diagnosis a failure of the machine through forecasting remaining useful life (RUL), especially on aircraft machine where the safety is priority due to enormous cost and human life. ML is the technique that accurately prediction through the data. Applied ML on PdM is the huge contribution for saving cost and human life guarantee of safety.The capacity of machinery working cannot last forever, sometimes it will be broken-down because of out-date operation. Machinery system that included sensors are just monitoring state of the machine, but cannot make a report the machine in good or bad condition.

3. IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas

Empathy maps are "quick and dirty" personas. Generally, empathy maps are low–fidelity works in progress that capture and articulate the facets of a representative user as currently understood and viewed by a team. The facets are thinks, feels, says, and does.

As your team identifies what they know about the user and places this information on a chart, you gain a more holistic view of your user's world and the problem or opportunity space. By having a more holistic view, you gain insights that add layers of context about the relationships between the users and their experiences. A more holistic view can also reveal the ways in which your user most naturally engages with what your team designs and builds. In other words, your designs should reach out to the user. Empathy maps can help you do that.



What do they THINK AND FEEL?
what really counts
major preoccupations
worries 8 aspirations
This predictive analysis uses both current and prior data to model future outcomes and catch problems before they actually occur.
Air lines and air crafts uses AI systems with built in machine learning algorithms to collect and analyse flight data regarding each route distance and altitudes, aircrafts types and weight, weather, etc
What do they HEAR?
what friends say
what boss say
what influencers say
If the deployment of this project was successful then it is very easy to use and secures the data.
This process makes data safe and also highly security based
What do they SEE?
environment
friends
what the market offers
carbon emission from aviation accounted for about 2.5% of global emission in 2018
What do they SAY AND DO?
attitude in public
appearance
behavior towards others
Not only it is safer but it also prevents extra work.
It is all about security of data
PAIN
fears
frustrations
obstacles
GAIN
"wants" / needs
measures of success
obstacles

3.2 Ideation & Brainstorming

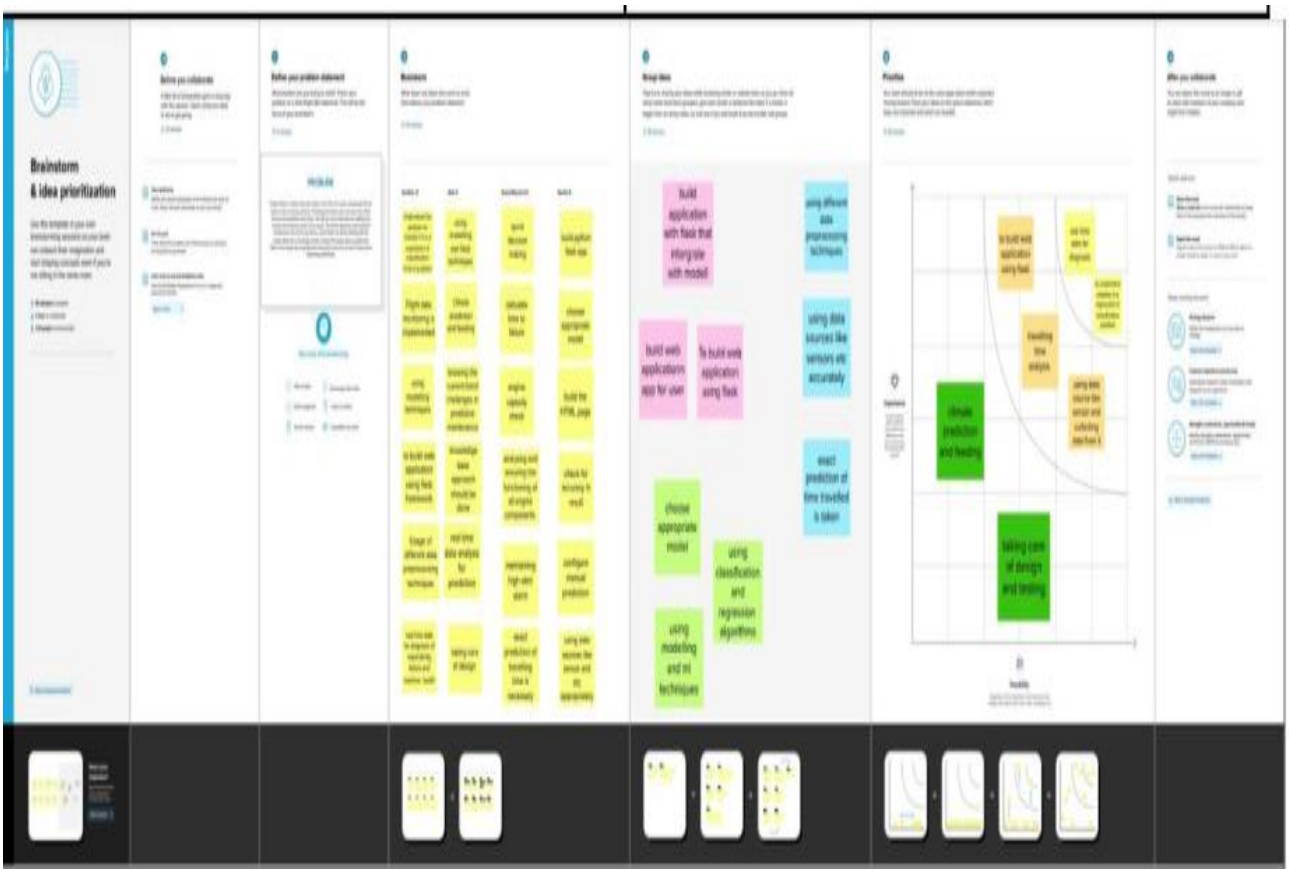
Ideation is the process of generating big ideas. Enterprise Design Thinking explains big ideas by contrasting them with features:

Big idea: Algorithms to predict the future from the past

Feature: Charts with lines that show prediction

Moving to big ideas takes your mind out of the problem space and into the realm of solutions. This realm is where you innovate and create revolutionary, rather than evolutionary, designs.

Brainstorming is a group creativity technique by which efforts are made to find a conclusion for a specific problem by gathering a list of ideas spontaneously contributed by its members.


Brainstorm
& idea prioritization
build application with flask that integrate with model
build web applications app for user
To build web application using flask
using different data preprocessing techniques
using data sources like sensors etc accurately
exact prediction of time travelled is taken
choose appropriate model
using classification and regression algorithms
using modelling and ml techniques
to build web application using flask
travelling time analysis
climate prediction and feeding
taking care of design and testing

3.3 Proposed Solution

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	To predict the failure of an engine by using Machine Learning to save loss of time and money thus improving productivity.
2.	Idea / Solution description	Machine learning is a type of artificial intelligence that allows software applications to become more accurate at predicting outcomes without being explicitly programmed to do so. ..Structural failure where a broken connecting rod, crank , valve, or camshaft is present account for seventeen percent of engine failure occurs. ..The failure can be detected by installing the sensors and keeping a track of the values.
3.	Novelty / Uniqueness	An air craft engine(or aero engine) is a propulsion system for an aircraft. ..Aircraft engines are the key module or the heart in aviation progress.
4.	Social Impact / Customer Satisfaction	The advent of human flight not only boosted our power of movement, but also enhanced our vision.
5.	Business Model (Revenue Model)	The reliability analysis is also important to predict their scheduled maintenance even and the Remaining Useful Life (RUL) of engine parts.
6.	Scalability of the Solution	This app can help customers to get updates of the flight of any part of the flight.

3.4 Problem Solution Fit

1. Customer Segment(S)

Who is your customer?

We will walk through our thought process and how we came to cluster the customers, what features were used and what methods were implemented to get a desired result.

This article is divided into six sections

- Business problem
- Data preparation
- Model implementation
- Result
- Future work

2. Jobs-To-Be-Done/ Problems

Which jobs to be done do you address for your customers?

There could be more than one explore different sides.Create a process that outline the workflow of what an agent should do when he or she receives a customer query with the focus of handling it promptly and efficiently.Ensure that your agents are aware of their roles and responsibilities along with who they are accountable to if and When there are lapses in service.

3. Triggers

what triggers customers to act?

To accurate predict the failure of an engine and track the flight. Preventable fuel problems as exhaustion, mismanagement, contamination or misfueling. Mechanic failure by under torqueing cylinder.

4. Emotions: Before/After

How do customers feel when they face a problem or a job and afterwards?

Frequently complaining customers who don't complain at all in fact they don't even bother responding to our emails.They simply stop engaging with you because they lose hope in your services.

5. Available Solutions

Which solution are available to the customers when they face the Problem Or need to get the job done?what have they tried in the past?

What Pros and cons do these solutions have ?

Pros

- Improved focus on core business activities
- Increased efficiency
- Controlled cost

Cons

- Difficulty with quality control

6. Customer Constraints

Customer constraints is the action a company takes in response to a service failure In an effort to convert previously dissatisfied customs into a loyal ones. Successful Companies have a process that are not only mitigates incoming customer Complaints, but also make the customer feel really good about the experience.

In the long term, service recovery has a positive impact on customer retention, word-of-mouth. And while most companies placed A greater focus on customer acquisition than on customer retention, we all know that acquiring new customer is anywhere from five to twenty five times more expensive then retaining an existing one.

7. Behaviour

What does your customer do to address the problem and get the job done?

Visiting the official page of airlines and service guarantee encourage customers to complain as they effect customs perceptions of reliability but are tenable only when the company is already focused on service quality. Empowering employees is a powerful tool for effective service recovery as the workspace will be able to think for themselves and make decisions on their own for their benefit of the firm customers.

8.CHANNELS of BEHAVIOUR

Online

What kind of actions do customers take online?

Extract online channels from #7 Findings suggest that causes, magnitude and consequences of service failures influence customers positive and negative emotions.

Offline

What kind of actions do customers take offline?

Extract offline channels from #7 and use them for customer development. Customer recovery is the action a company takes in response to service failure in an effort to convert previously dissatisfied customers into loyal ones. Successful companies have a process that not only mitigates incoming customers complaints but also make the customer feel really good about the experience.

9. Problem Root Cause

What is the real reason that this problem exists? What is the back story behind the need to do this job?

The overall structure of the situation will indicate several directions in which you analysis can proceed in more complex situations,however you will have to probe more deeply into both the things an 2/3 the processes that make up the structure. you will be trying to make clear the components of each change over time.

10. Your Solution

If you are working on existing business, write down your current solution first, fill in the canvas, and check how much it fits reality.

If you are working on a new business proposition, then keep it blank until you fill in the canvas and come up with a solution that fits within customer limitations, solves a problem and matches customer behaviour. Business ought to understand customer need as it is vital to match the competitive market place. Broadly customers needs about delivering a better experience by exceeding their expectations. Provide faster solutions. Improve your products and services. Reduce the tickets. No of support tickets.

4. REQUIREMENT ANALYSIS

4.1 Functional Requirement

FR No.	Functional Requirement(Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Registration through Form Registration through Gmail Registration through LinkedIN Registration through phone number
FR-2	User Confirmation	Confirmation via Email Confirmation via OTP
FR-3	User Login	Login using credentials
FR-4	Search	Get the aircraft engine details
FR-5	GPS	Track the mal-function in aircraft engine
FR-6	Analysis	Fetch Dataset
FR-7	Prediction	It will predict the aircraft engine failure and solve the problem

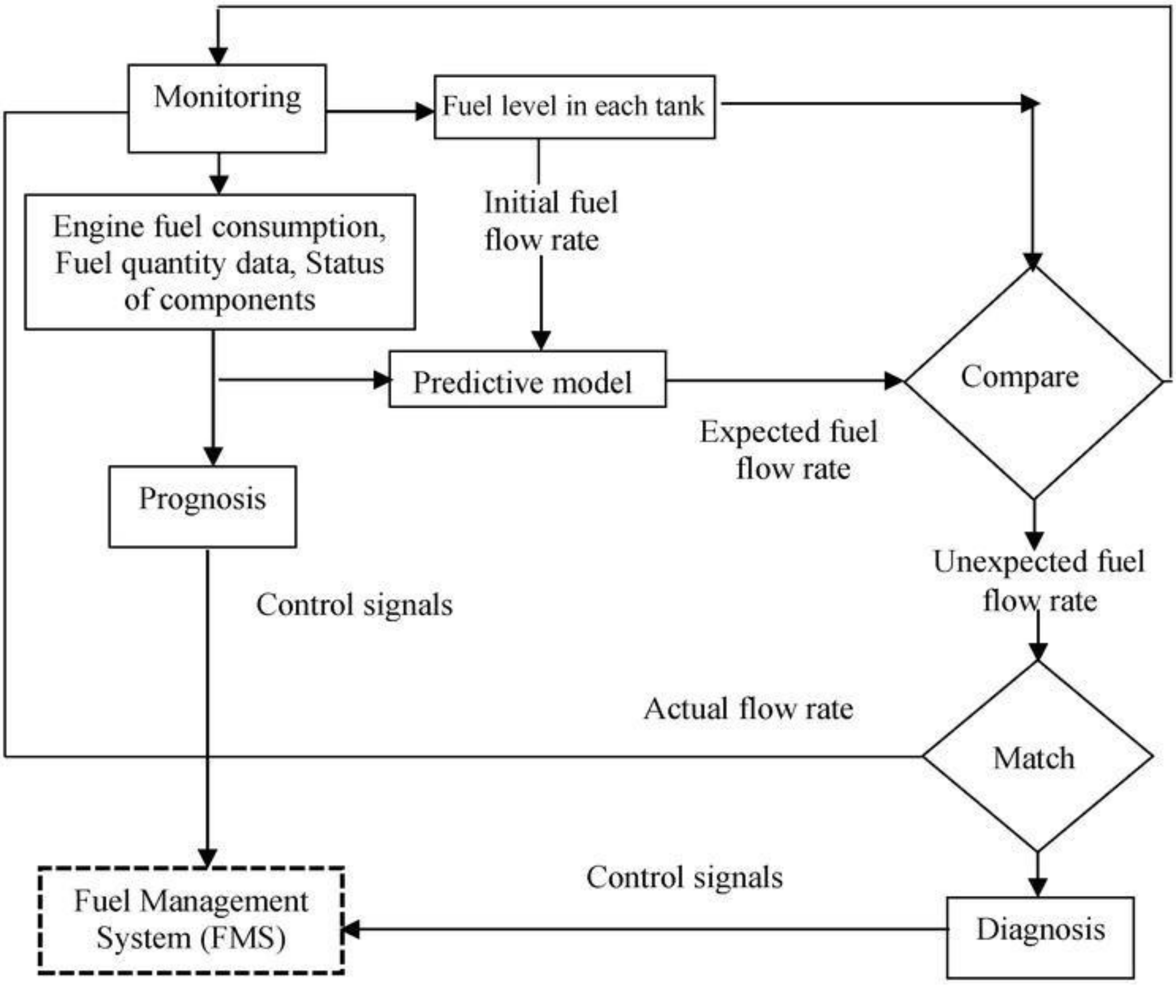
4.2 Non-Functional Requirement

FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	It indicates how efficiently and easy users can learn and use a system.
NFR-2	**Security**	Assures all data inside the system or its part will be protected against malware attacks or unauthorized access.
NFR-3	**Reliability**	Specifies the probability of the software performing without failure for a specific number of uses or amount of time.
NFR-4	**Performance**	It deals with the measure of the system response time under different load conditions.
NFR-5	**Availability**	The system accessible for a user at a given point of time.

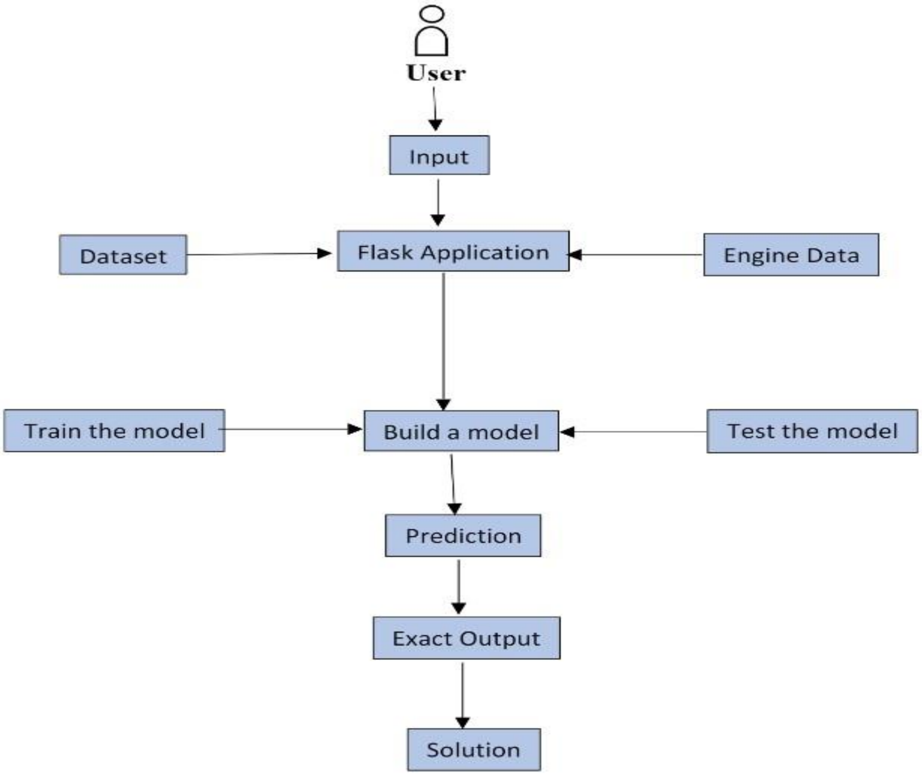
5. PROJECT DESIGN

5.1 Data Flow Diagram

A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It shows how data enters and leaves the system, what changes the information, and where data is stored.A data flow diagram (DFD) maps out the flow of information for any process or system. It uses defined symbols like rectangles, circles and arrows, plus short text labels, to show data inputs, outputs, storage points and the routes between each destination. Data flowcharts can range from simple, even hand-drawn process overviews, to in-depth, multi-level DFDs that dig progressively deeper into how the data is handled. They can be used to analyze an existing system or model a new one. Like all the best diagrams and charts, a DFD can often visually “say” things that would be hard to explain in words, and they work for both technical and nontechnical audiences, from developer to CEO. That’s why DFDs remain so popular after all these years. While they work well for data flow software and systems, they are less applicable nowadays to visualizing interactive, real-time or database-oriented software or systems.



5.2 Solution & Technical Architecture



5.3 User Stories

Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Registration		As a user, I can register for the application by entering my email, password, and confirming my password.	I can access my account / dashboard	High	Sprint-1
	USN-1	As a user, I will receive confirmation email once I have registered for the application	I can receive confirmation email & click confirm	High	Sprint-1
	USN-3	As a user, I can register for the application through Facebook	I can register & access the dashboard with Facebook Login	Low	Sprint-2

Login	USN-4	As a user, I can log into the application by entering email & password		High	Sprint-1
Dashboard	USN-5	As a user, I can search my requirements			
Registration	USN-2	As a user, I can register for the application by entering my email, password, and confirming my password.	I can access my account/ dashboard	High	Sprint-1
	USN-2	As a user, I will receive confirmation email once I have registered for the application	I can receive confirmation		
Checking user security and requirements	USN-1	When user facing any security issue	Administrator will solve the users requirements	High	Sprint-2
Monitoring the users verifications	USN-2	When users verify through email, captcha	Administrator will monitor the verification stage issues	High	

6. PROJECT PLANNING & SCHEDULING

6.1 Sprint Planning & Estimation

Product Backlog, Sprint Schedule, and Estimation

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	5	High	4
Sprint-1	Facebook Registration	USN-2	As a user, I can register for the application through Facebook	4	Medium	4
Sprint-1	Gmail registration	USN-3	As a user, I can register for the application through Gmail	3	Low	4
Sprint-2	Login	USN-4	As a user, I can log into the application by entering email & password	5	High	4
Sprint-2	Facebook	USN-5	As a user, I can log in into this application through Facebook	4	Medium	4
Sprint-2	Email	USN-6	As a user, I can log in into this application by entering my Google Account	3	Low	4
Sprint-3	Analyzing / Detecting Problems	USN-7	As a user, I can able analyze the defects in Aircraft Engine	5	High	4
Sprint-3	Analyzing / Detecting Problems	USN-8	As a user, I can able to view the repeated problems occurs in Aircraft Engine	4	Medium	4
Sprint-3	Analyzing / Detecting Problems	USN-9	As a user, I can able to find the defects occurs in Aircraft Engine	4	Low	4
Sprint-4	Solution	USN-10	As a user, I can view the solution for minor problems of the Aircraft Engine	3	Medium	4
Sprint-4	Solution	USN-11	As a user, I can view the solution for major problems of the Aircraft Engine	5	High	4
Sprint-4	Solution	USN-12	As a user, I can find the solution and suggestion to maintain for regular services	4	Low	4

6.2 Sprint Delivery Schedule

Project Tracker, Velocity & Burndown Chart:

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	20	6 Days	24 Oct 2022	29 Oct 2022	20	29 Oct 2022
Sprint-2	20	6 Days	31 Oct 2022	05 Nov 2022	20	05 Nov 2022
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022	20	12 Nov 2022
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022	20	19 Nov 2022

6.3 Reports From JIRA

Velocity:

SPRINT DURATION:6 Days

VELOCITY OF THE TEAM : 20 (Points per Sprint)

TOTAL AVERAGE VELOCITY

$$AV = sprint\ valuation\ / velocity$$

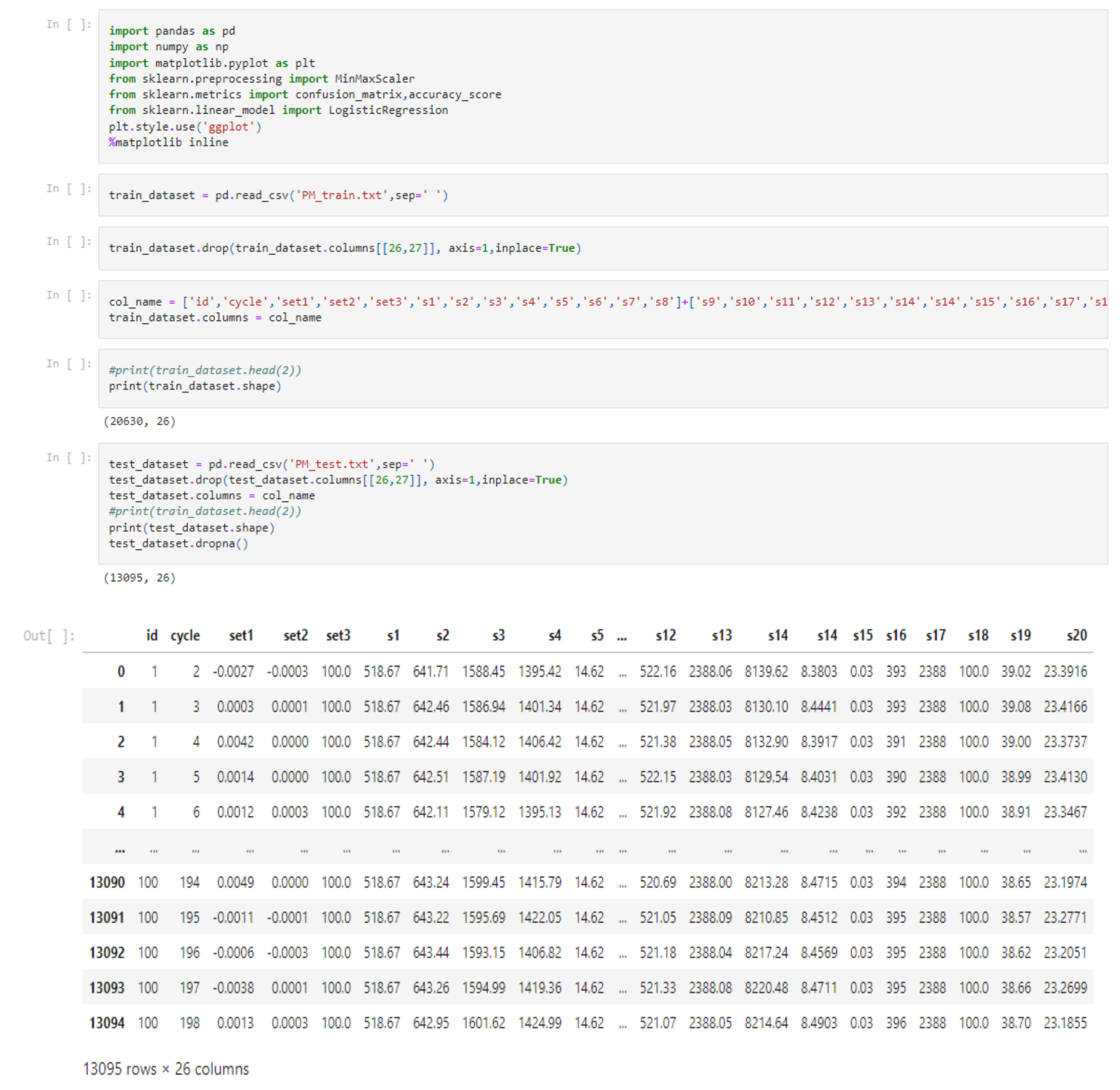
= 20 /6

= 3.33 Story points per day

7. CODING & SOLUTIONING

7.1 Features:

Feature 1: Data_preprocessing



```
In [ ]: import pandas as pd
        import numpy as np
        import matplotlib.pyplot as plt
        from sklearn.preprocessing import MinMaxScaler
        from sklearn.metrics import confusion_matrix,accuracy_score
        from sklearn.linear_model import LogisticRegression
        plt.style.use('ggplot')
        %matplotlib inline
```

```
In [ ]: train_dataset = pd.read_csv('PM_train.txt',sep=' ')
```

```
In [ ]: train_dataset.drop(train_dataset.columns[[26,27]], axis=1,inplace=True)
```

```
In [ ]: col_name = ['id','cycle','set1','set2','set3','s1','s2','s3','s4','s5','s6','s7','s8']+['s9','s10','s11','s12','s13','s14','s14','s15','s16','s17','s1
        train_dataset.columns = col_name
```

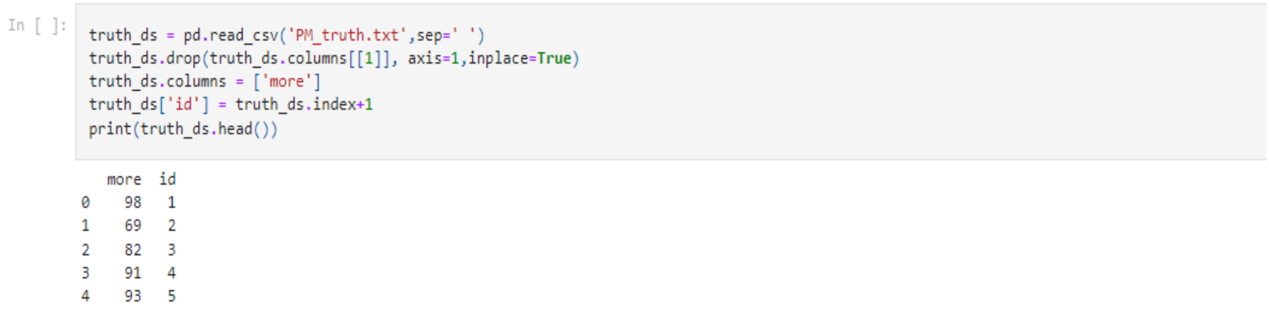
```
In [ ]: #print(train_dataset.head(2))
        print(train_dataset.shape)

        (20630, 26)
```

```
In [ ]: test_dataset = pd.read_csv('PM_test.txt',sep=' ')
        test_dataset.drop(test_dataset.columns[[26,27]], axis=1,inplace=True)
        test_dataset.columns = col_name
        #print(train_dataset.head(2))
        print(test_dataset.shape)
        test_dataset.dropna()

        (13095, 26)
```

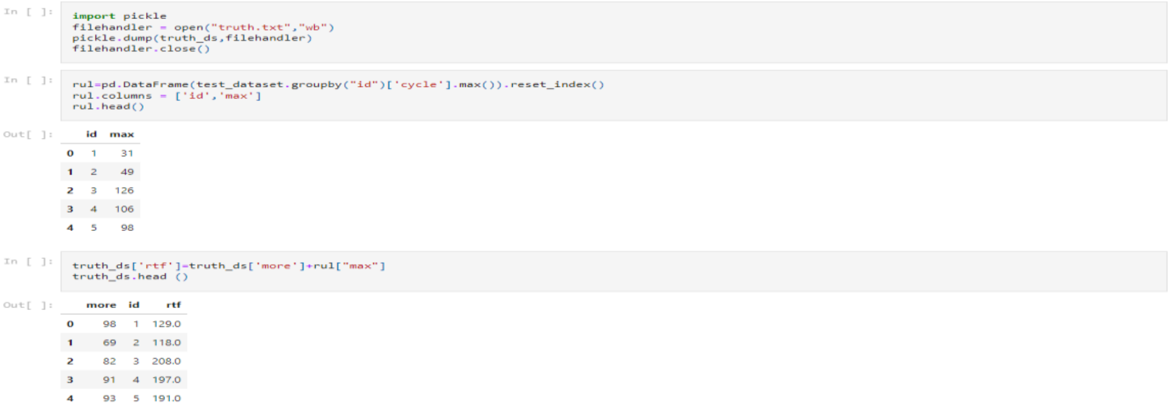
Out[ ]:

	id	cycle	set1	set2	set3	s1	s2	s3	s4	s5	...	s12	s13	s14	s14	s15	s16	s17	s18	s19	s20
0	1	2	-0.0027	-0.0003	100.0	518.67	641.71	1588.45	1395.42	14.62	...	522.16	2388.06	8139.62	8.3803	0.03	393	2388	100.0	39.02	23.3916
1	1	3	0.0003	0.0001	100.0	518.67	642.46	1586.94	1401.34	14.62	...	521.97	2388.03	8130.10	8.4441	0.03	393	2388	100.0	39.08	23.4166
2	1	4	0.0042	0.0000	100.0	518.67	642.44	1584.12	1406.42	14.62	...	521.38	2388.05	8132.90	8.3917	0.03	391	2388	100.0	39.00	23.3737
3	1	5	0.0014	0.0000	100.0	518.67	642.51	1587.19	1401.92	14.62	...	522.15	2388.03	8129.54	8.4031	0.03	390	2388	100.0	38.99	23.4130
4	1	6	0.0012	0.0003	100.0	518.67	642.11	1579.12	1395.13	14.62	...	521.92	2388.08	8127.46	8.4238	0.03	392	2388	100.0	38.91	23.3467
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
13090	100	194	0.0049	0.0000	100.0	518.67	643.24	1599.45	1415.79	14.62	...	520.69	2388.00	8213.28	8.4715	0.03	394	2388	100.0	38.65	23.1974
13091	100	195	-0.0011	-0.0001	100.0	518.67	643.22	1595.69	1422.05	14.62	...	521.05	2388.09	8210.85	8.4512	0.03	395	2388	100.0	38.57	23.2771
13092	100	196	-0.0006	-0.0003	100.0	518.67	643.44	1593.15	1406.82	14.62	...	521.18	2388.04	8217.24	8.4569	0.03	395	2388	100.0	38.62	23.2051
13093	100	197	-0.0038	0.0001	100.0	518.67	643.26	1594.99	1419.36	14.62	...	521.33	2388.08	8220.48	8.4711	0.03	395	2388	100.0	38.66	23.2699
13094	100	198	0.0013	0.0003	100.0	518.67	642.95	1601.62	1424.99	14.62	...	521.07	2388.05	8214.64	8.4903	0.03	396	2388	100.0	38.70	23.1855

13095 rows × 26 columns



```
In [ ]: truth_ds = pd.read_csv('PM_truth.txt',sep=' ')
        truth_ds.drop(truth_ds.columns[[1]], axis=1,inplace=True)
        truth_ds.columns = ['more']
        truth_ds['id'] = truth_ds.index+1
        print(truth_ds.head())

   more  id
0    98   1
1    69   2
2    82   3
3    91   4
4    93   5
```



In [ ]:

```
import pickle
filehandler = open("truth.txt","wb")
pickle.dump(truth_ds,filehandler)
filehandler.close()
```

In [ ]:

```
rul=pd.DataFrame(test_dataset.groupby("id")['cycle'].max()).reset_index()
rul.columns = ['id','max']
rul.head()
```

Out[ ]:

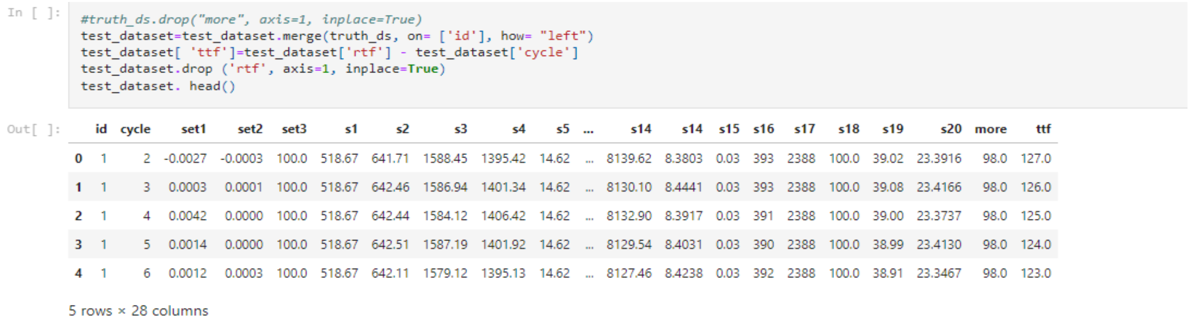
	id	max
0	1	31
1	2	49
2	3	126
3	4	106
4	5	98

In [ ]:

```
truth_ds['rtf']=truth_ds['more']+rul["max"]
truth_ds.head ()
```

Out[ ]:

	more	id	rtf
0	98	1	129.0
1	69	2	118.0
2	82	3	208.0
3	91	4	197.0
4	93	5	191.0



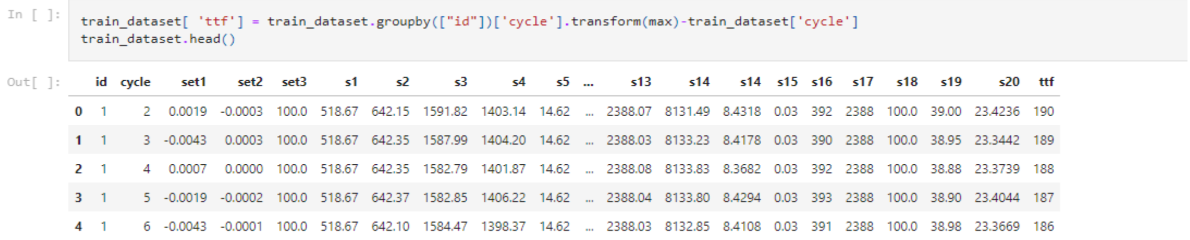
In [ ]:

```
#truth_ds.drop("more", axis=1, inplace=True)
test_dataset=test_dataset.merge(truth_ds, on= ['id'], how= "left")
test_dataset[ 'ttf']=test_dataset['rtf'] - test_dataset['cycle']
test_dataset.drop ('rtf', axis=1, inplace=True)
test_dataset. head()
```

Out[ ]:

	id	cycle	set1	set2	set3	s1	s2	s3	s4	s5	...	s14	s14	s15	s16	s17	s18	s19	s20	more	ttf
0	1	2	-0.0027	-0.0003	100.0	518.67	641.71	1588.45	1395.42	14.62	...	8139.62	8.3803	0.03	393	2388	100.0	39.02	23.3916	98.0	127.0
1	1	3	0.0003	0.0001	100.0	518.67	642.46	1586.94	1401.34	14.62	...	8130.10	8.4441	0.03	393	2388	100.0	39.08	23.4166	98.0	126.0
2	1	4	0.0042	0.0000	100.0	518.67	642.44	1584.12	1406.42	14.62	...	8132.90	8.3917	0.03	391	2388	100.0	39.00	23.3737	98.0	125.0
3	1	5	0.0014	0.0000	100.0	518.67	642.51	1587.19	1401.92	14.62	...	8129.54	8.4031	0.03	390	2388	100.0	38.99	23.4130	98.0	124.0
4	1	6	0.0012	0.0003	100.0	518.67	642.11	1579.12	1395.13	14.62	...	8127.46	8.4238	0.03	392	2388	100.0	38.91	23.3467	98.0	123.0

5 rows × 28 columns



In [ ]:

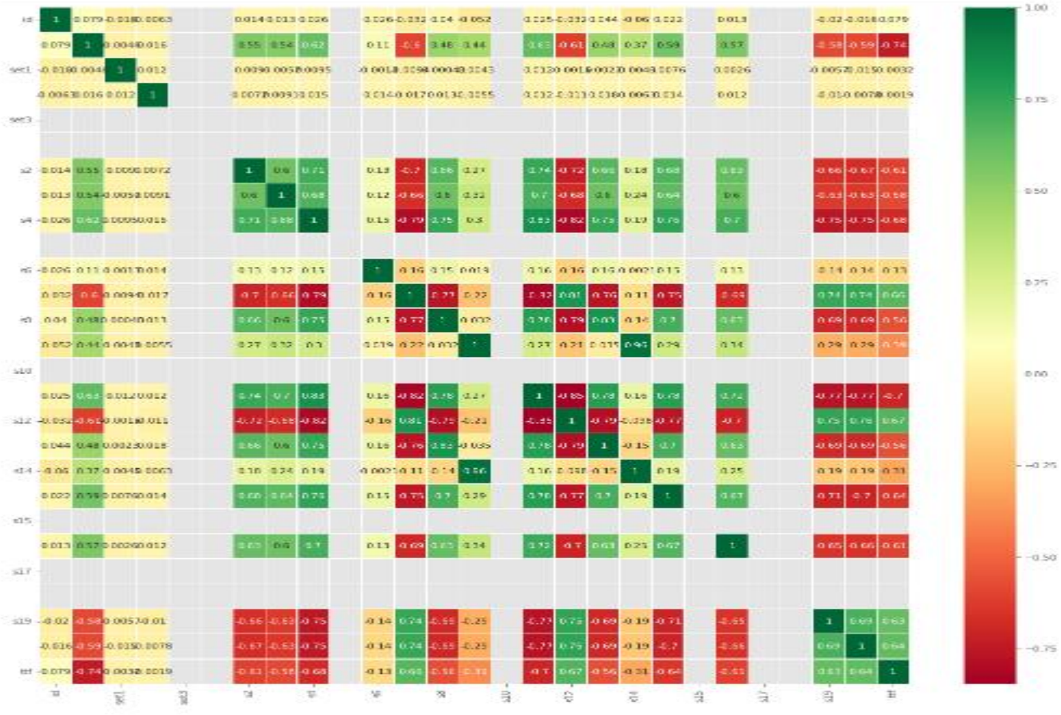
```
train_dataset[ 'ttf'] = train_dataset.groupby(["id"])['cycle'].transform(max)-train_dataset['cycle']
train_dataset.head()
```

Out[ ]:

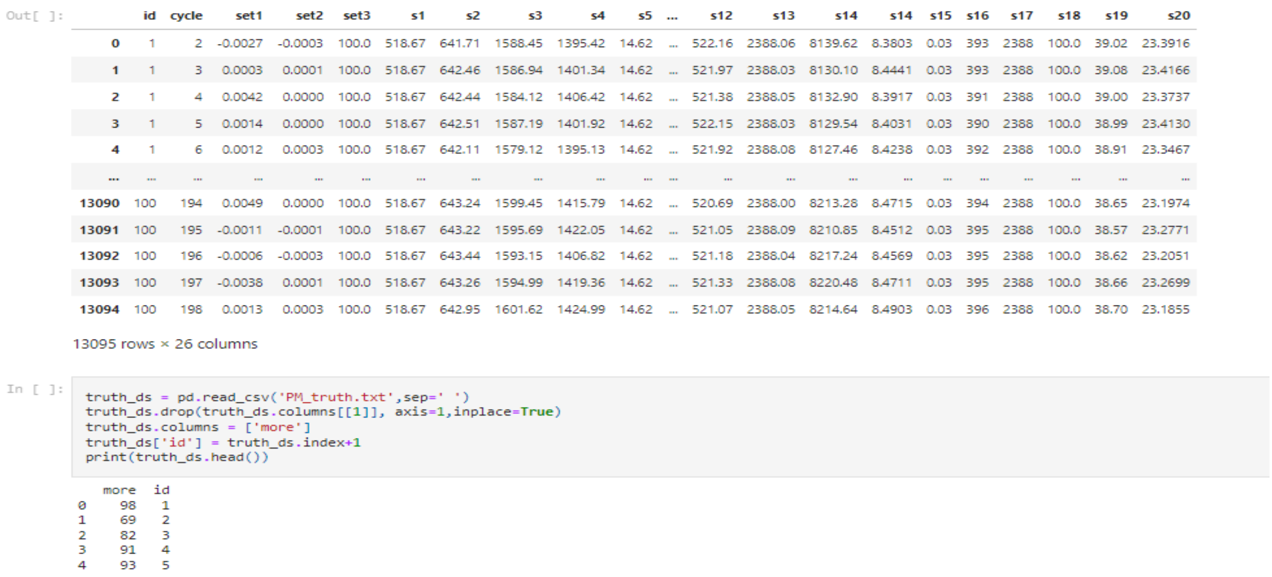
	id	cycle	set1	set2	set3	s1	s2	s3	s4	s5	...	s13	s14	s14	s15	s16	s17	s18	s19	s20	ttf
0	1	2	0.0019	-0.0003	100.0	518.67	642.15	1591.82	1403.14	14.62	...	2388.07	8131.49	8.4318	0.03	392	2388	100.0	39.00	23.4236	190
1	1	3	-0.0043	0.0003	100.0	518.67	642.35	1587.99	1404.20	14.62	...	2388.03	8133.23	8.4178	0.03	390	2388	100.0	38.95	23.3442	189
2	1	4	0.0007	0.0000	100.0	518.67	642.35	1582.79	1401.87	14.62	...	2388.08	8133.83	8.3682	0.03	392	2388	100.0	38.88	23.3739	188
3	1	5	-0.0019	-0.0002	100.0	518.67	642.37	1582.85	1406.22	14.62	...	2388.04	8133.80	8.4294	0.03	393	2388	100.0	38.90	23.4044	187
4	1	6	-0.0043	-0.0001	100.0	518.67	642.10	1584.47	1398.37	14.62	...	2388.03	8132.85	8.4108	0.03	391	2388	100.0	38.98	23.3669	186

In [ ]:

```
import seaborn as sb
sb.heatmap(train_dataset.corr(),annot=True,cmap='RdYlGn',linewidths=0.2)
fig=plt.gcf()
fig.set_size_inches(18,18)
plt.show()
```



Feature 2: train_mode



Out[ ]:

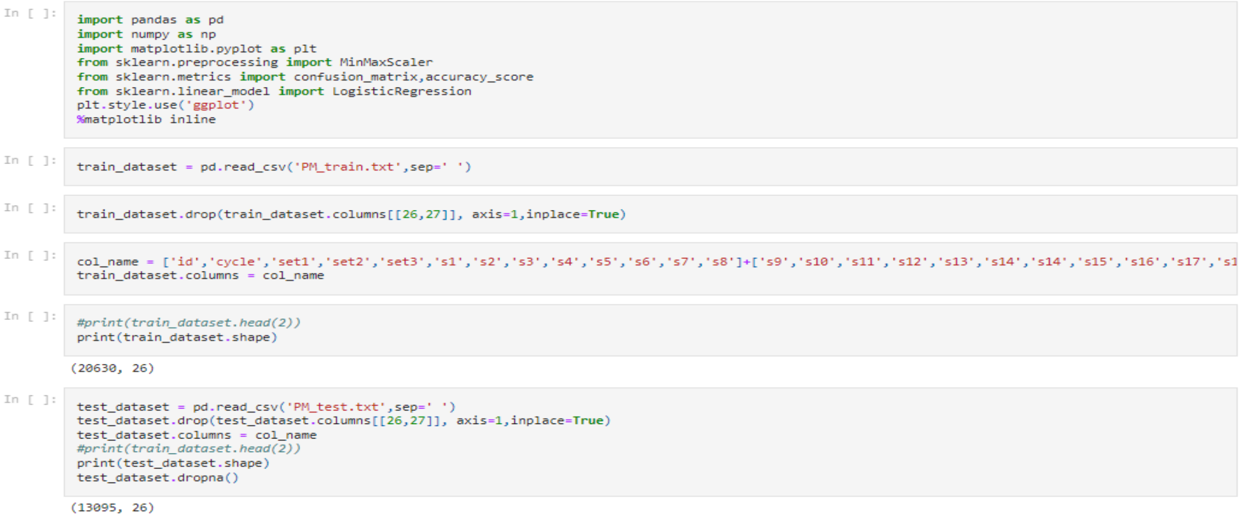
	id	cycle	set1	set2	set3	s1	s2	s3	s4	s5	...	s12	s13	s14	s14	s15	s16	s17	s18	s19	s20
0	1	2	-0.0027	-0.0003	100.0	518.67	641.71	1588.45	1395.42	14.62	...	522.16	2388.06	8139.62	8.3803	0.03	393	2388	100.0	39.02	23.3916
1	1	3	0.0003	0.0001	100.0	518.67	642.46	1586.94	1401.34	14.62	...	521.97	2388.03	8130.10	8.4441	0.03	393	2388	100.0	39.08	23.4166
2	1	4	0.0042	0.0000	100.0	518.67	642.44	1584.12	1406.42	14.62	...	521.38	2388.05	8132.90	8.3917	0.03	391	2388	100.0	39.00	23.3737
3	1	5	0.0014	0.0000	100.0	518.67	642.51	1587.19	1401.92	14.62	...	522.15	2388.03	8129.54	8.4031	0.03	390	2388	100.0	38.99	23.4130
4	1	6	0.0012	0.0003	100.0	518.67	642.11	1579.12	1395.13	14.62	...	521.92	2388.08	8127.46	8.4238	0.03	392	2388	100.0	38.91	23.3467
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
13090	100	194	0.0049	0.0000	100.0	518.67	643.24	1599.45	1415.79	14.62	...	520.69	2388.00	8213.28	8.4715	0.03	394	2388	100.0	38.65	23.1974
13091	100	195	-0.0011	-0.0001	100.0	518.67	643.22	1595.69	1422.05	14.62	...	521.05	2388.09	8210.85	8.4512	0.03	395	2388	100.0	38.57	23.2771
13092	100	196	-0.0006	-0.0003	100.0	518.67	643.44	1593.15	1406.82	14.62	...	521.18	2388.04	8217.24	8.4569	0.03	395	2388	100.0	38.62	23.2051
13093	100	197	-0.0038	0.0001	100.0	518.67	643.26	1594.99	1419.36	14.62	...	521.33	2388.08	8220.48	8.4711	0.03	395	2388	100.0	38.66	23.2699
13094	100	198	0.0013	0.0003	100.0	518.67	642.95	1601.62	1424.99	14.62	...	521.07	2388.05	8214.64	8.4903	0.03	396	2388	100.0	38.70	23.1855

13095 rows × 26 columns

In [ ]:

```
truth_ds = pd.read_csv('PM_truth.txt',sep=' ')
truth_ds.drop(truth_ds.columns[[1]], axis=1,inplace=True)
truth_ds.columns = ['more']
truth_ds['id'] = truth_ds.index+1
print(truth_ds.head())
```

```
   more  id
0    98   1
1    69   2
2    82   3
3    91   4
4    93   5
```



In [ ]:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.preprocessing import MinMaxScaler
from sklearn.metrics import confusion_matrix,accuracy_score
from sklearn.linear_model import LogisticRegression
plt.style.use('ggplot')
%matplotlib inline
```

In [ ]:

```
train_dataset = pd.read_csv('PM_train.txt',sep=' ')
```

In [ ]:

```
train_dataset.drop(train_dataset.columns[[26,27]], axis=1,inplace=True)
```

In [ ]:

```
col_name = ['id','cycle','set1','set2','set3','s1','s2','s3','s4','s5','s6','s7','s8']+['s9','s10','s11','s12','s13','s14','s14','s15','s16','s17','s1
train_dataset.columns = col_name
```

In [ ]:

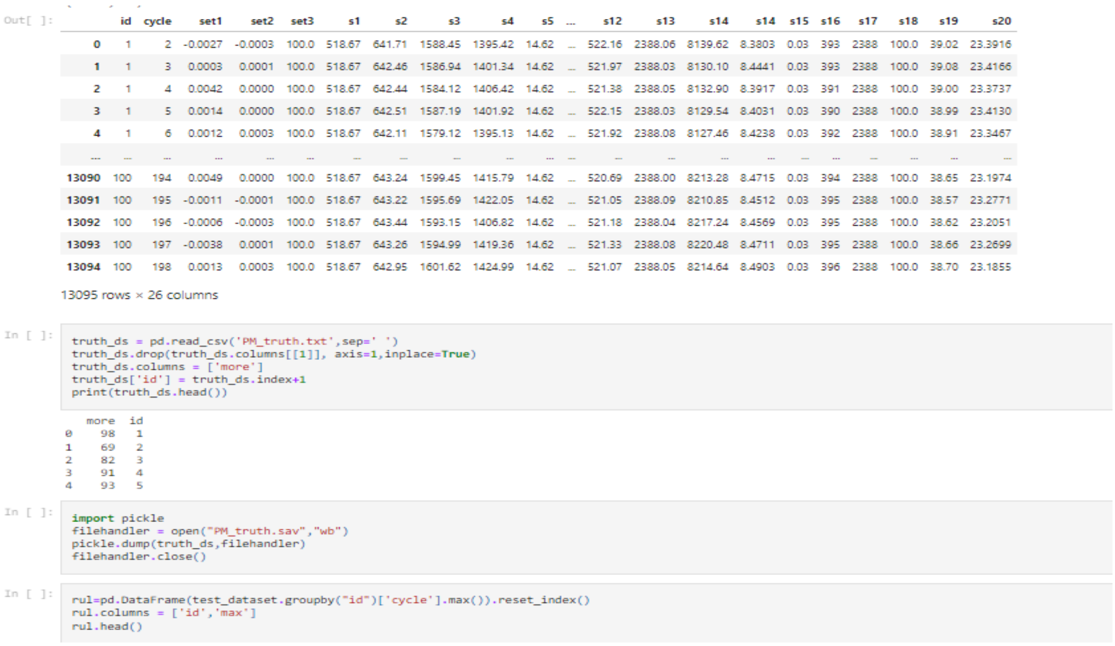
```
#print(train_dataset.head(2))
print(train_dataset.shape)
```

```
(20630, 26)
```

In [ ]:

```
test_dataset = pd.read_csv('PM_test.txt',sep=' ')
test_dataset.drop(test_dataset.columns[[26,27]], axis=1,inplace=True)
test_dataset.columns = col_name
#print(train_dataset.head(2))
print(test_dataset.shape)
test_dataset.dropna()
```

```
(13095, 26)
```



Out[ ]:

	id	cycle	set1	set2	set3	s1	s2	s3	s4	s5	...	s12	s13	s14	s14	s15	s16	s17	s18	s19	s20
0	1	2	-0.0027	-0.0003	100.0	518.67	641.71	1588.45	1395.42	14.62	...	522.16	2388.06	8139.62	8.3803	0.03	393	2388	100.0	39.02	23.3916
1	1	3	0.0003	0.0001	100.0	518.67	642.46	1586.94	1401.34	14.62	...	521.97	2388.03	8130.10	8.4441	0.03	393	2388	100.0	39.08	23.4166
2	1	4	0.0042	0.0000	100.0	518.67	642.44	1584.12	1406.42	14.62	...	521.38	2388.05	8132.90	8.3917	0.03	391	2388	100.0	39.00	23.3737
3	1	5	0.0014	0.0000	100.0	518.67	642.51	1587.19	1401.92	14.62	...	522.15	2388.03	8129.54	8.4031	0.03	390	2388	100.0	38.99	23.4130
4	1	6	0.0012	0.0003	100.0	518.67	642.11	1579.12	1395.13	14.62	...	521.92	2388.08	8127.46	8.4238	0.03	392	2388	100.0	38.91	23.3467
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
13090	100	194	0.0049	0.0000	100.0	518.67	643.24	1599.45	1415.79	14.62	...	520.69	2388.00	8213.28	8.4715	0.03	394	2388	100.0	38.65	23.1974
13091	100	195	-0.0011	-0.0001	100.0	518.67	643.22	1595.69	1422.05	14.62	...	521.05	2388.09	8210.85	8.4512	0.03	395	2388	100.0	38.57	23.2771
13092	100	196	-0.0006	-0.0003	100.0	518.67	643.44	1593.15	1406.82	14.62	...	521.18	2388.04	8217.24	8.4569	0.03	395	2388	100.0	38.62	23.2051
13093	100	197	-0.0038	0.0001	100.0	518.67	643.26	1594.99	1419.36	14.62	...	521.33	2388.08	8220.48	8.4711	0.03	395	2388	100.0	38.66	23.2699
13094	100	198	0.0013	0.0003	100.0	518.67	642.95	1601.62	1424.99	14.62	...	521.07	2388.05	8214.64	8.4903	0.03	396	2388	100.0	38.70	23.1855

13095 rows × 26 columns

In [ ]:

```
truth_ds = pd.read_csv('PM_truth.txt',sep=' ')
truth_ds.drop(truth_ds.columns[[1]], axis=1,inplace=True)
truth_ds.columns = ['more']
truth_ds['id'] = truth_ds.index+1
print(truth_ds.head())
```

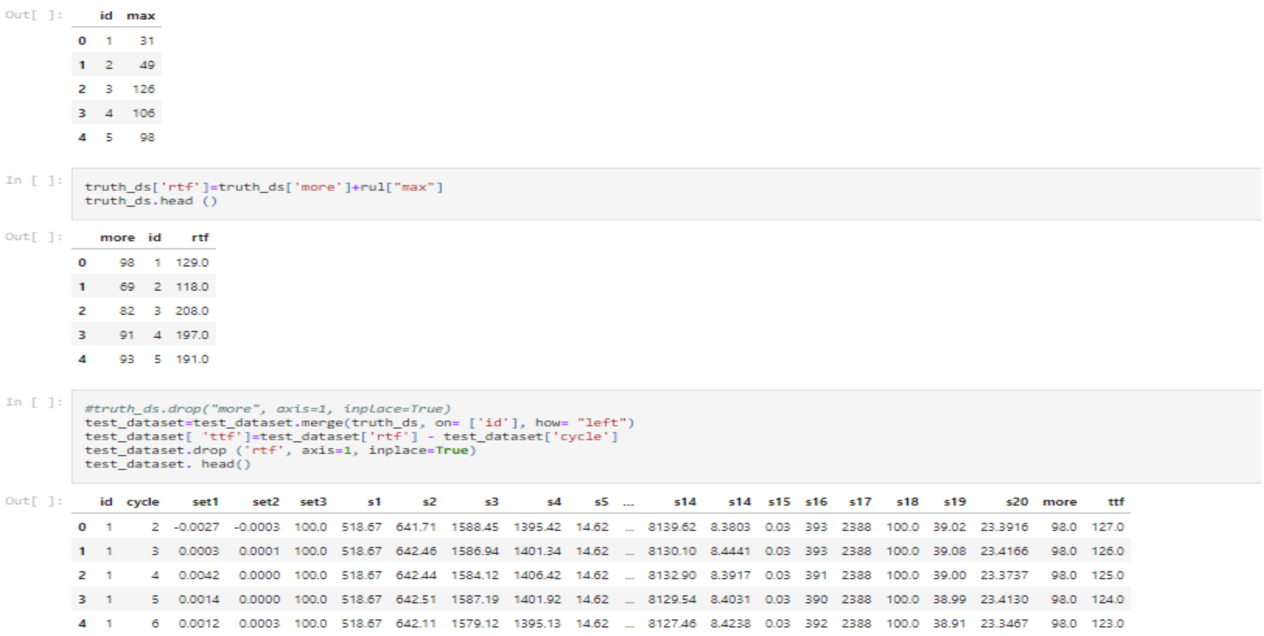
```
   more  id
0    98   1
1    69   2
2    82   3
3    91   4
4    93   5
```

In [ ]:

```
import pickle
filehandler = open("PM_truth.sav","wb")
pickle.dump(truth_ds,filehandler)
filehandler.close()
```

In [ ]:

```
rul=pd.DataFrame(test_dataset.groupby("id")['cycle'].max()).reset_index()
rul.columns = ['id','max']
rul.head()
```



Out[ ]:

	id	max
0	1	31
1	2	49
2	3	126
3	4	106
4	5	98

In [ ]:

```
truth_ds['rtf']=truth_ds['more']+rul["max"]
truth_ds.head ()
```

Out[ ]:

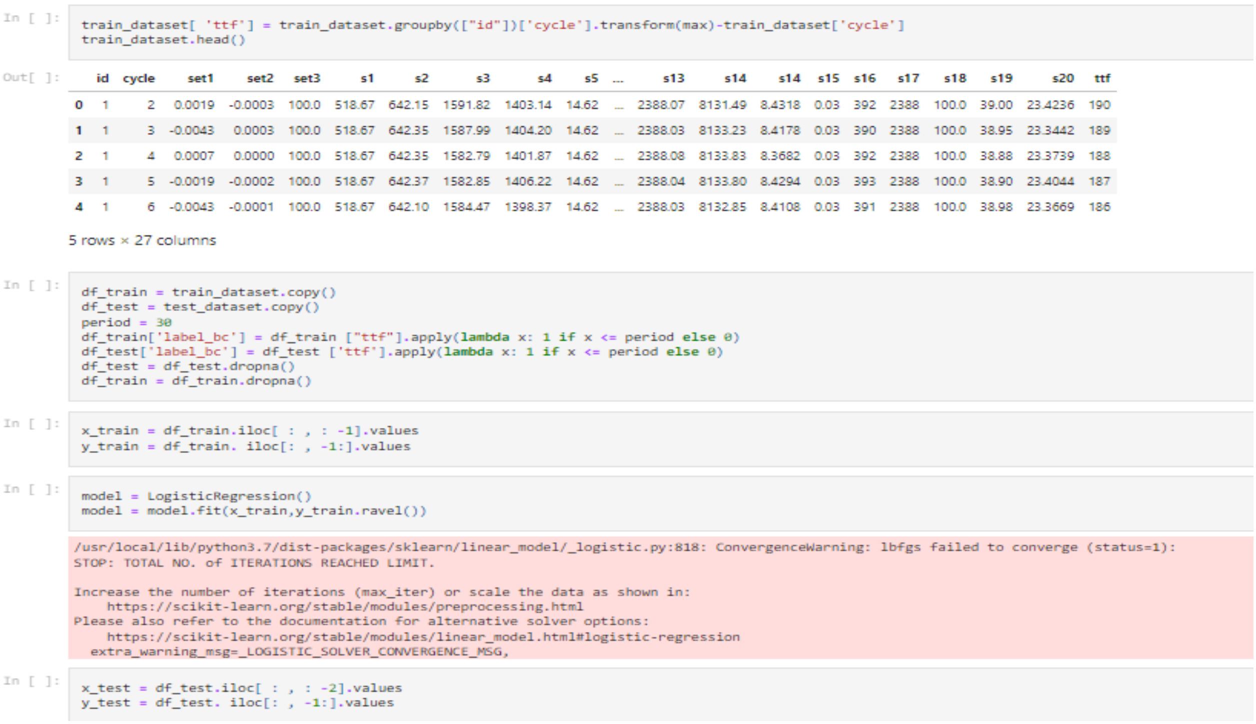
	more	id	rtf
0	98	1	129.0
1	69	2	118.0
2	82	3	208.0
3	91	4	197.0
4	93	5	191.0

In [ ]:

```
#truth_ds.drop("more", axis=1, inplace=True)
test_dataset=test_dataset.merge(truth_ds, on= ['id'], how= "left")
test_dataset[ 'ttf']=test_dataset['rtf'] - test_dataset['cycle']
test_dataset.drop ('rtf', axis=1, inplace=True)
test_dataset. head()
```

Out[ ]:

	id	cycle	set1	set2	set3	s1	s2	s3	s4	s5	...	s14	s14	s15	s16	s17	s18	s19	s20	more	ttf
0	1	2	-0.0027	-0.0003	100.0	518.67	641.71	1588.45	1395.42	14.62	...	8139.62	8.3803	0.03	393	2388	100.0	39.02	23.3916	98.0	127.0
1	1	3	0.0003	0.0001	100.0	518.67	642.46	1586.94	1401.34	14.62	...	8130.10	8.4441	0.03	393	2388	100.0	39.08	23.4166	98.0	126.0
2	1	4	0.0042	0.0000	100.0	518.67	642.44	1584.12	1406.42	14.62	...	8132.90	8.3917	0.03	391	2388	100.0	39.00	23.3737	98.0	125.0
3	1	5	0.0014	0.0000	100.0	518.67	642.51	1587.19	1401.92	14.62	...	8129.54	8.4031	0.03	390	2388	100.0	38.99	23.4130	98.0	124.0
4	1	6	0.0012	0.0003	100.0	518.67	642.11	1579.12	1395.13	14.62	...	8127.46	8.4238	0.03	392	2388	100.0	38.91	23.3467	98.0	123.0



In [ ]:

```
train_dataset[ 'ttf'] = train_dataset.groupby(["id"])['cycle'].transform(max)-train_dataset['cycle']
train_dataset.head()
```

Out[ ]:

	id	cycle	set1	set2	set3	s1	s2	s3	s4	s5	...	s13	s14	s14	s15	s16	s17	s18	s19	s20	ttf
0	1	2	0.0019	-0.0003	100.0	518.67	642.15	1591.82	1403.14	14.62	...	2388.07	8131.49	8.4318	0.03	392	2388	100.0	39.00	23.4236	190
1	1	3	-0.0043	0.0003	100.0	518.67	642.35	1587.99	1404.20	14.62	...	2388.03	8133.23	8.4178	0.03	390	2388	100.0	38.95	23.3442	189
2	1	4	0.0007	0.0000	100.0	518.67	642.35	1582.79	1401.87	14.62	...	2388.08	8133.83	8.3682	0.03	392	2388	100.0	38.88	23.3739	188
3	1	5	-0.0019	-0.0002	100.0	518.67	642.37	1582.85	1406.22	14.62	...	2388.04	8133.80	8.4294	0.03	393	2388	100.0	38.90	23.4044	187
4	1	6	-0.0043	-0.0001	100.0	518.67	642.10	1584.47	1398.37	14.62	...	2388.03	8132.85	8.4108	0.03	391	2388	100.0	38.98	23.3669	186

5 rows × 27 columns

In [ ]:

```
df_train = train_dataset.copy()
df_test = test_dataset.copy()
period = 30
df_train['label_bc'] = df_train ["ttf"].apply(lambda x: 1 if x <= period else 0)
df_test['label_bc'] = df_test ['ttf'].apply(lambda x: 1 if x <= period else 0)
df_test = df_test.dropna()
df_train = df_train.dropna()
```

In [ ]:

```
x_train = df_train.iloc[ : , : -1].values
y_train = df_train. iloc[: , -1:].values
```

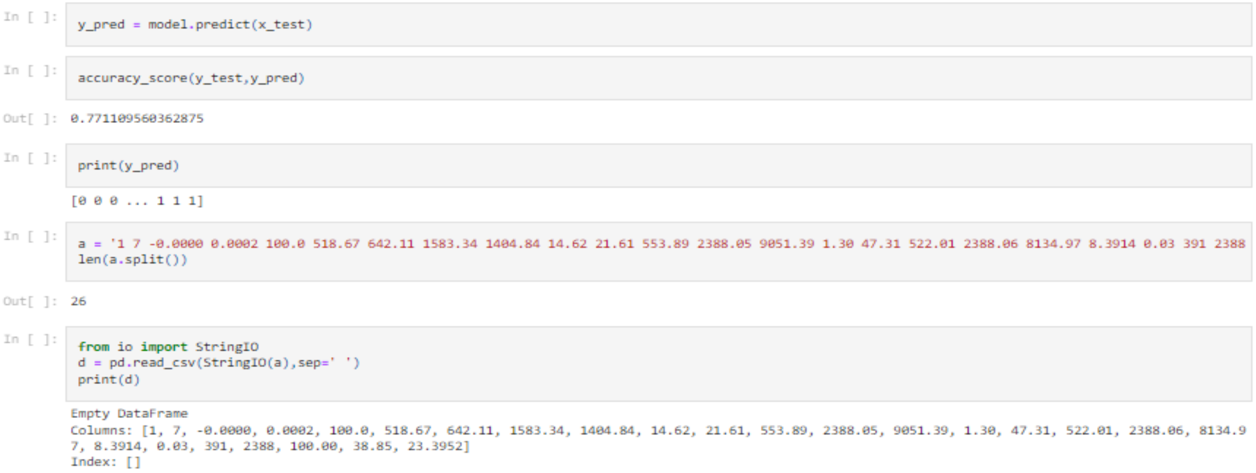
In [ ]:

```
model = LogisticRegression()
model = model.fit(x_train,y_train.ravel())
```

```
/usr/local/lib/python3.7/dist-packages/sklearn/linear_model/_logistic.py:818: ConvergenceWarning: lbfgs failed to converge (status=1):
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

Increase the number of iterations (max_iter) or scale the data as shown in:
    https://scikit-learn.org/stable/modules/preprocessing.html
Please also refer to the documentation for alternative solver options:
    https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression
  extra_warning_msg=_LOGISTIC_SOLVER_CONVERGENCE_MSG,
```

In [ ]:

```
x_test = df_test.iloc[ : , : -2].values
y_test = df_test. iloc[: , -1:].values
```



In [ ]:

```
y_pred = model.predict(x_test)
```

In [ ]:

```
accuracy_score(y_test,y_pred)
```

Out[ ]:

```
0.771109560362875
```

In [ ]:

```
print(y_pred)
```

```
[0 0 0 ... 1 1 1]
```

In [ ]:

```
a = '1 7 -0.0000 0.0002 100.0 518.67 642.11 1583.34 1404.84 14.62 21.61 553.89 2388.05 9051.39 1.30 47.31 522.01 2388.06 8134.97 8.3914 0.03 391 2388
len(a.split())
```

Out[ ]:

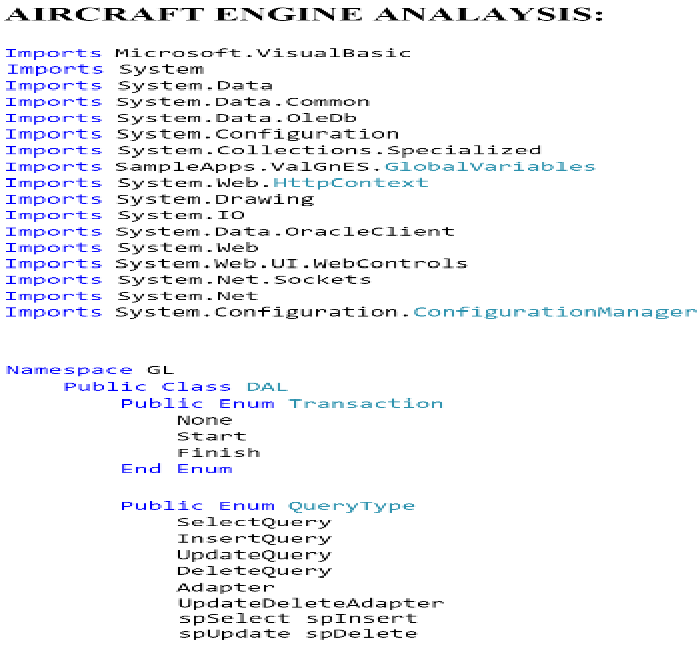
```
26
```

In [ ]:

```
from io import StringIO
d = pd.read_csv(StringIO(a),sep=' ')
print(d)
```

```
Empty DataFrame
Columns: [1, 7, -0.0000, 0.0002, 100.0, 518.67, 642.11, 1583.34, 1404.84, 14.62, 21.61, 553.89, 2388.05, 9051.39, 1.30, 47.31, 522.01, 2388.06, 8134.9
7, 8.3914, 0.03, 391, 2388, 100.00, 38.85, 23.3952]
Index: []
```

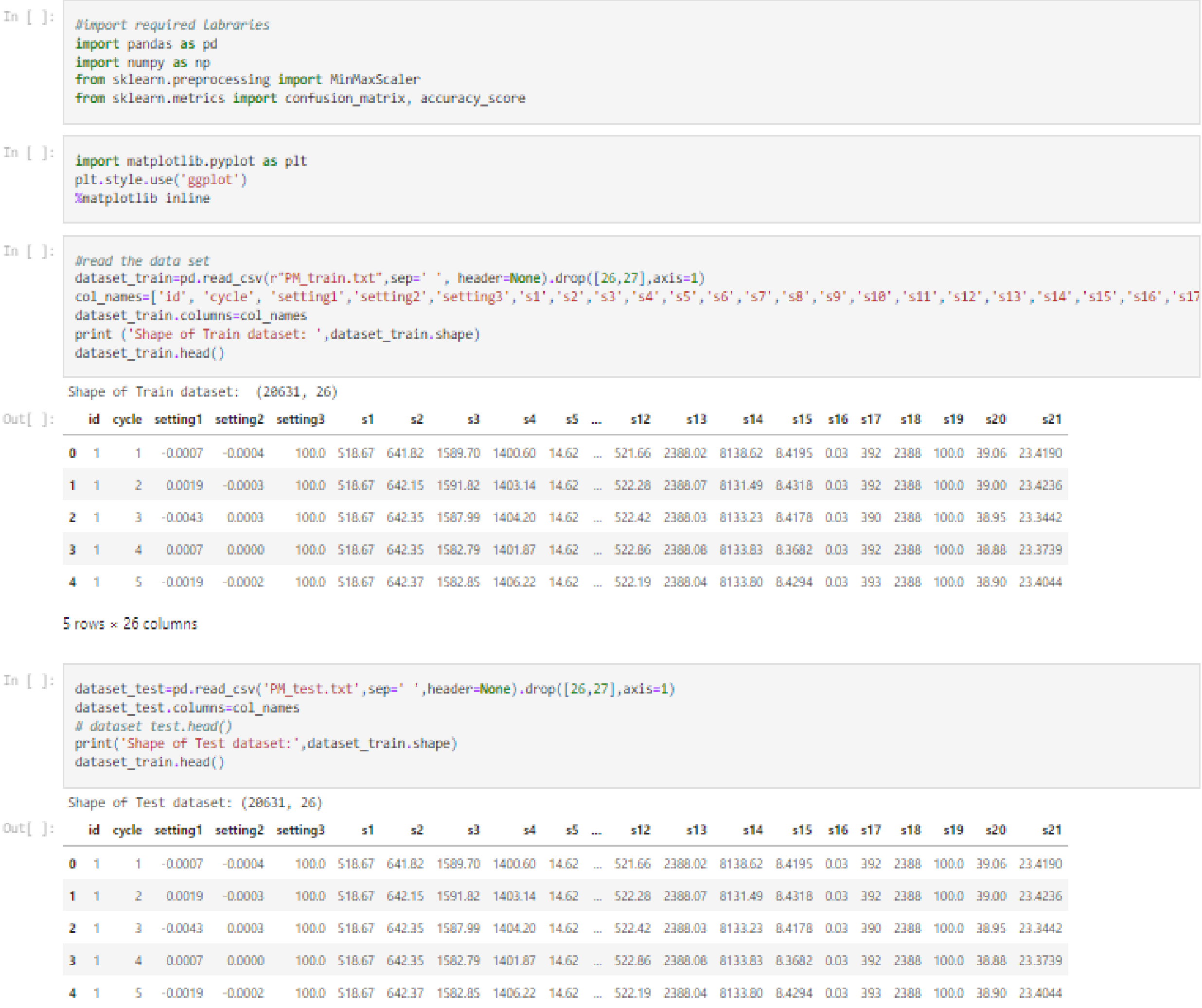
Feature 3:Aircraft Engine Analaysis



AIRCRAFT ENGINE ANALAYSIS:

```
Imports Microsoft.VisualBasic
Imports System
Imports System.Data
Imports System.Data.Common
Imports System.Data.OleDb
Imports System.Configuration
Imports System.Collections.Specialized
Imports SampleApps.ValGnES.GlobalVariables
Imports System.Web.HttpContext
Imports System.Drawing
Imports System.IO
Imports System.Data.OracleClient
Imports System.Web
Imports System.Web.UI.WebControls
Imports System.Net.Sockets
Imports System.Net
Imports System.Configuration.ConfigurationManager


Namespace GL
    Public Class DAL
        Public Enum Transaction
            None
            Start
            Finish
        End Enum

        Public Enum QueryType
            SelectQuery
            InsertQuery
            UpdateQuery
            DeleteQuery
            Adapter
            UpdateDeleteAdapter
            spSelect spInsert
            spUpdate spDelete
```

Feature 4:model_build



```
In [ ]:
#import required labraries
import pandas as pd
import numpy as np
from sklearn.preprocessing import MinMaxScaler
from sklearn.metrics import confusion_matrix, accuracy_score
```

```
In [ ]:
import matplotlib.pyplot as plt
plt.style.use('ggplot')
%matplotlib inline
```

```
In [ ]:
#read the data set
dataset_train=pd.read_csv(r"PM_train.txt",sep=' ', header=None).drop([26,27],axis=1)
col_names=['id', 'cycle', 'setting1','setting2','setting3','s1','s2','s3','s4','s5','s6','s7','s8','s9','s10','s11','s12','s13','s14','s15','s16','s17
dataset_train.columns=col_names
print ('Shape of Train dataset: ',dataset_train.shape)
dataset_train.head()
```

```
Shape of Train dataset:  (20631, 26)
```

Out[ ]:

	id	cycle	setting1	setting2	setting3	s1	s2	s3	s4	s5	...	s12	s13	s14	s15	s16	s17	s18	s19	s20	s21
0	1	1	-0.0007	-0.0004	100.0	518.67	641.82	1589.70	1400.60	14.62	...	521.66	2388.02	8138.62	8.4195	0.03	392	2388	100.0	39.06	23.4190
1	1	2	0.0019	-0.0003	100.0	518.67	642.15	1591.82	1403.14	14.62	...	522.28	2388.07	8131.49	8.4318	0.03	392	2388	100.0	39.00	23.4236
2	1	3	-0.0043	0.0003	100.0	518.67	642.35	1587.99	1404.20	14.62	...	522.42	2388.03	8133.23	8.4178	0.03	390	2388	100.0	38.95	23.3442
3	1	4	0.0007	0.0000	100.0	518.67	642.35	1582.79	1401.87	14.62	...	522.86	2388.08	8133.83	8.3682	0.03	392	2388	100.0	38.88	23.3739
4	1	5	-0.0019	-0.0002	100.0	518.67	642.37	1582.85	1406.22	14.62	...	522.19	2388.04	8133.80	8.4294	0.03	393	2388	100.0	38.90	23.4044

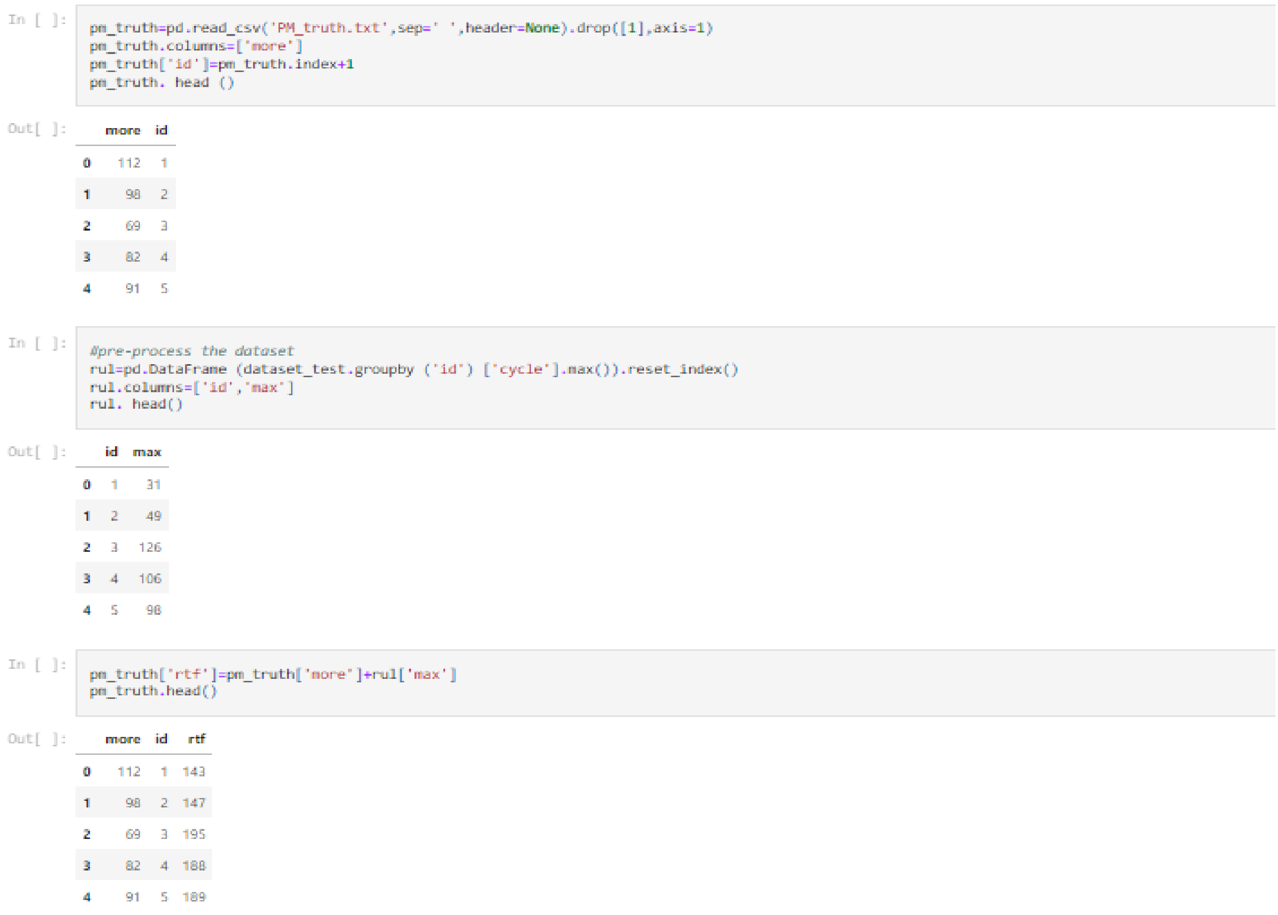
5 rows × 26 columns

```
In [ ]:
dataset_test=pd.read_csv('PM_test.txt',sep=' ',header=None).drop([26,27],axis=1)
dataset_test.columns=col_names
# dataset_test.head()
print('Shape of Test dataset:',dataset_train.shape)
dataset_train.head()
```

```
Shape of Test dataset: (20631, 26)
```

Out[ ]:

	id	cycle	setting1	setting2	setting3	s1	s2	s3	s4	s5	...	s12	s13	s14	s15	s16	s17	s18	s19	s20	s21
0	1	1	-0.0007	-0.0004	100.0	518.67	641.82	1589.70	1400.60	14.62	...	521.66	2388.02	8138.62	8.4195	0.03	392	2388	100.0	39.06	23.4190
1	1	2	0.0019	-0.0003	100.0	518.67	642.15	1591.82	1403.14	14.62	...	522.28	2388.07	8131.49	8.4318	0.03	392	2388	100.0	39.00	23.4236
2	1	3	-0.0043	0.0003	100.0	518.67	642.35	1587.99	1404.20	14.62	...	522.42	2388.03	8133.23	8.4178	0.03	390	2388	100.0	38.95	23.3442
3	1	4	0.0007	0.0000	100.0	518.67	642.35	1582.79	1401.87	14.62	...	522.86	2388.08	8133.83	8.3682	0.03	392	2388	100.0	38.88	23.3739
4	1	5	-0.0019	-0.0002	100.0	518.67	642.37	1582.85	1406.22	14.62	...	522.19	2388.04	8133.80	8.4294	0.03	393	2388	100.0	38.90	23.4044



In [ ]:

```
pm_truth=pd.read_csv('PM_truth.txt',sep=' ',header=None).drop([1],axis=1)
pm_truth.columns=['more']
pm_truth['id']=pm_truth.index+1
pm_truth. head ()
```

Out[ ]:

	more	id
0	112	1
1	98	2
2	69	3
3	82	4
4	91	5

In [ ]:

```
#pre-process the dataset
rul=pd.DataFrame (dataset_test.groupby ('id') ['cycle'].max()).reset_index()
rul.columns=['id','max']
rul. head()
```

Out[ ]:

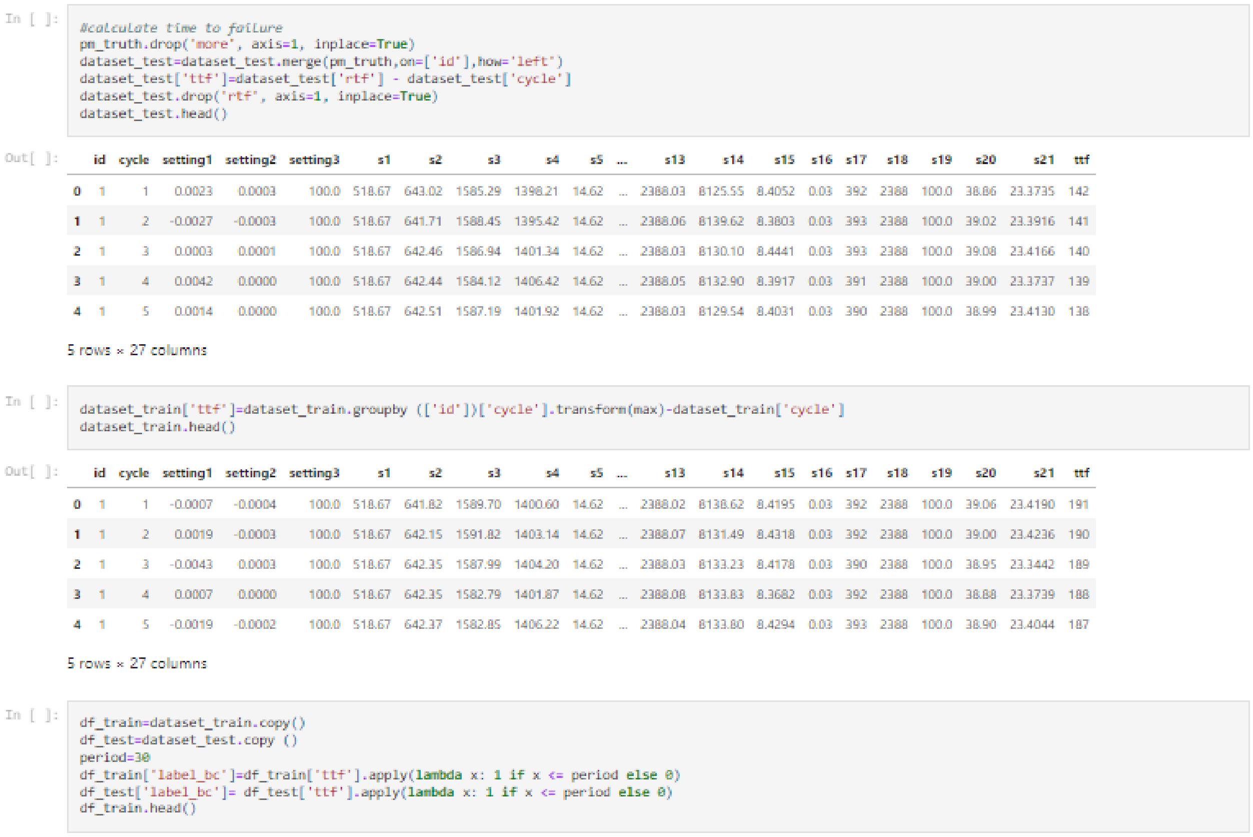
	id	max
0	1	31
1	2	49
2	3	126
3	4	106
4	5	98

In [ ]:

```
pm_truth['rtf']=pm_truth['more']+rul['max']
pm_truth.head()
```

Out[ ]:

	more	id	rtf
0	112	1	143
1	98	2	147
2	69	3	195
3	82	4	188
4	91	5	189



In [ ]:

```
#calculate time to failure
pm_truth.drop('more', axis=1, inplace=True)
dataset_test=dataset_test.merge(pm_truth,on=['id'],how='left')
dataset_test['ttf']=dataset_test['rtf'] - dataset_test['cycle']
dataset_test.drop('rtf', axis=1, inplace=True)
dataset_test.head()
```

Out[ ]:

	id	cycle	setting1	setting2	setting3	s1	s2	s3	s4	s5	...	s13	s14	s15	s16	s17	s18	s19	s20	s21	ttf
0	1	1	0.0023	0.0003	100.0	518.67	643.02	1585.29	1398.21	14.62	...	2388.03	8125.55	8.4052	0.03	392	2388	100.0	38.86	23.3735	142
1	1	2	-0.0027	-0.0003	100.0	518.67	641.71	1588.45	1395.42	14.62	...	2388.06	8139.62	8.3803	0.03	393	2388	100.0	39.02	23.3916	141
2	1	3	0.0003	0.0001	100.0	518.67	642.46	1586.94	1401.34	14.62	...	2388.03	8130.10	8.4441	0.03	393	2388	100.0	39.08	23.4166	140
3	1	4	0.0042	0.0000	100.0	518.67	642.44	1584.12	1406.42	14.62	...	2388.05	8132.90	8.3917	0.03	391	2388	100.0	39.00	23.3737	139
4	1	5	0.0014	0.0000	100.0	518.67	642.51	1587.19	1401.92	14.62	...	2388.03	8129.54	8.4031	0.03	390	2388	100.0	38.99	23.4130	138

5 rows × 27 columns

In [ ]:

```
dataset_train['ttf']=dataset_train.groupby (['id'])['cycle'].transform(max)-dataset_train['cycle']
dataset_train.head()
```

Out[ ]:

	id	cycle	setting1	setting2	setting3	s1	s2	s3	s4	s5	...	s13	s14	s15	s16	s17	s18	s19	s20	s21	ttf
0	1	1	-0.0007	-0.0004	100.0	518.67	641.82	1589.70	1400.60	14.62	...	2388.02	8138.62	8.4195	0.03	392	2388	100.0	39.06	23.4190	191
1	1	2	0.0019	-0.0003	100.0	518.67	642.15	1591.82	1403.14	14.62	...	2388.07	8131.49	8.4318	0.03	392	2388	100.0	39.00	23.4236	190
2	1	3	-0.0043	0.0003	100.0	518.67	642.35	1587.99	1404.20	14.62	...	2388.03	8133.23	8.4178	0.03	390	2388	100.0	38.95	23.3442	189
3	1	4	0.0007	0.0000	100.0	518.67	642.35	1582.79	1401.87	14.62	...	2388.08	8133.83	8.3682	0.03	392	2388	100.0	38.88	23.3739	188
4	1	5	-0.0019	-0.0002	100.0	518.67	642.37	1582.85	1406.22	14.62	...	2388.04	8133.80	8.4294	0.03	393	2388	100.0	38.90	23.4044	187

5 rows × 27 columns

In [ ]:

```
df_train=dataset_train.copy()
df_test=dataset_test.copy ()
period=30
df_train['label_bc']=df_train['ttf'].apply(lambda x: 1 if x <= period else 0)
df_test['label_bc']= df_test['ttf'].apply(lambda x: 1 if x <= period else 0)
df_train.head()
```

```
In [ ]:
x=df_train.iloc[:,:-1].values
y=df_train.iloc[:,-1].values
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.3,random_state=3)
y
```

```
Out[ ]: array([0, 0, 0, ..., 1, 1, 1])
```

```
In [ ]:
from sklearn.linear_model import LogisticRegression
model=LogisticRegression()
model.fit(x_train,y_train)
```

```
/usr/local/lib/python3.7/dist-packages/sklearn/linear_model/_logistic.py:818: ConvergenceWarning: lbfgs failed to converge (status=1):
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

Increase the number of iterations (max_iter) or scale the data as shown in:
    https://scikit-learn.org/stable/modules/preprocessing.html
Please also refer to the documentation for alternative solver options:
    https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression
  extra_warning_msg=_LOGISTIC_SOLVER_CONVERGENCE_MSG,
```

```
Out[ ]: LogisticRegression()
```

```
In [ ]:
#Check the metrics of the model
from sklearn.metrics import accuracy_score
y_predlog=model.predict(x_train)
accuracy_score(y_predlog,y_train)
```

```
Out[ ]: 1.0
```

```
In [ ]:
y_pred_test=model.predict(x_test)
accuracy_score(y_pred_test,y_test)
```

```
Out[ ]: 0.9998384491114701
```

```
In [ ]:
from sklearn.metrics import confusion_matrix
cm1=confusion_matrix(y_test,y_pred_test)
cm1
```

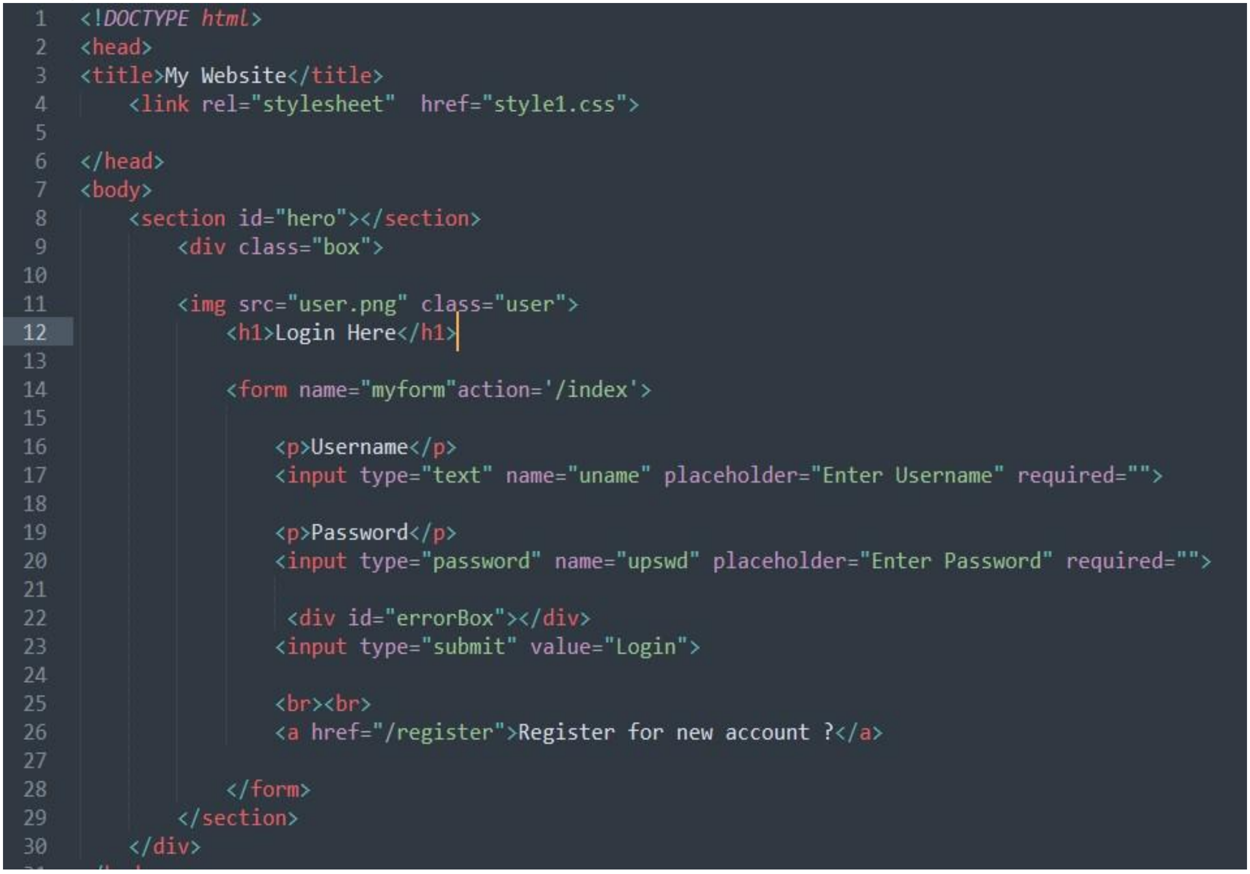
```
Out[ ]: array([[5297,    1],
               [   0,  892]])
```

```
In [ ]:
#saving the model
import joblib
joblib.dump(model, "engine_model.sav")
```

```
Out[ ]: ['engine_model.sav']
```

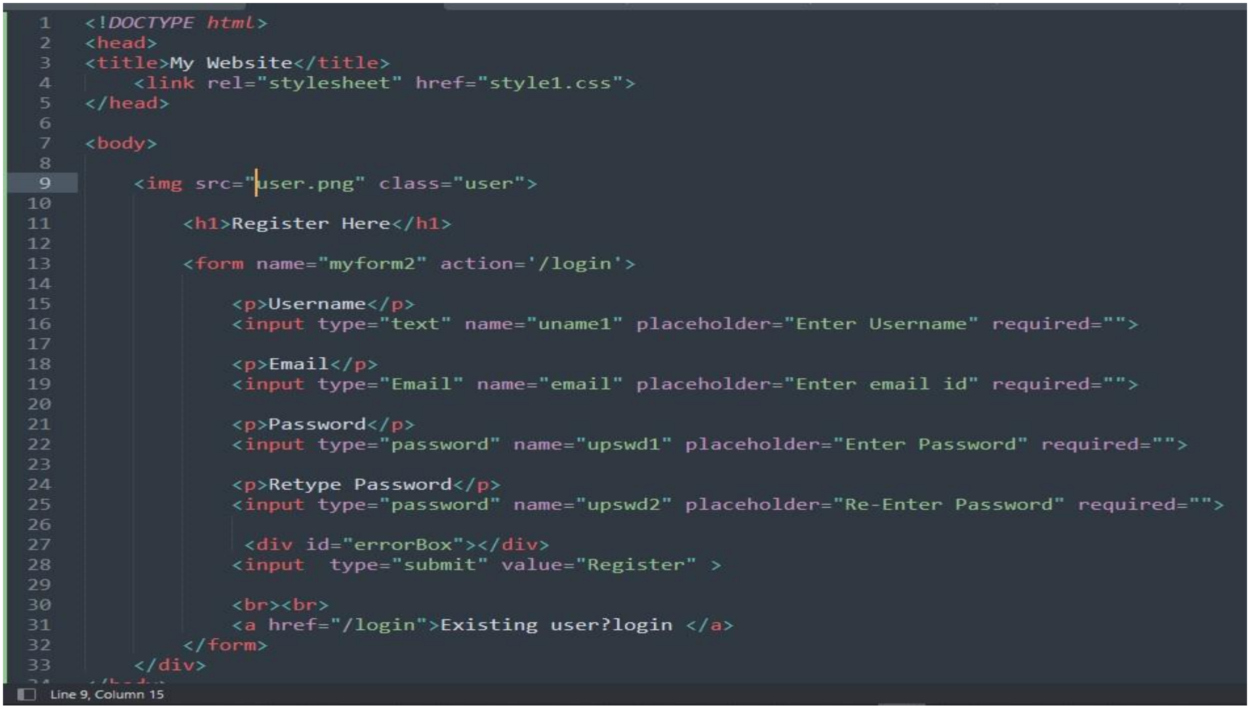
7.2 Other Features

Login.html:



```
<!DOCTYPE html>
<head>
<title>My Website</title>
    <link rel="stylesheet"  href="style1.css">

</head>
<body>
    <section id="hero"></section>
        <div class="box">

        <img src="user.png" class="user">
            <h1>Login Here</h1>

            <form name="myform"action='/index'>

                <p>Username</p>
                <input type="text" name="uname" placeholder="Enter Username" required="">

                <p>Password</p>
                <input type="password" name="upswd" placeholder="Enter Password" required="">

                 <div id="errorBox"></div>
                <input type="submit" value="Login">

                <br><br>
                <a href="/register">Register for new account ?</a>

            </form>
        </section>
    </div>
```

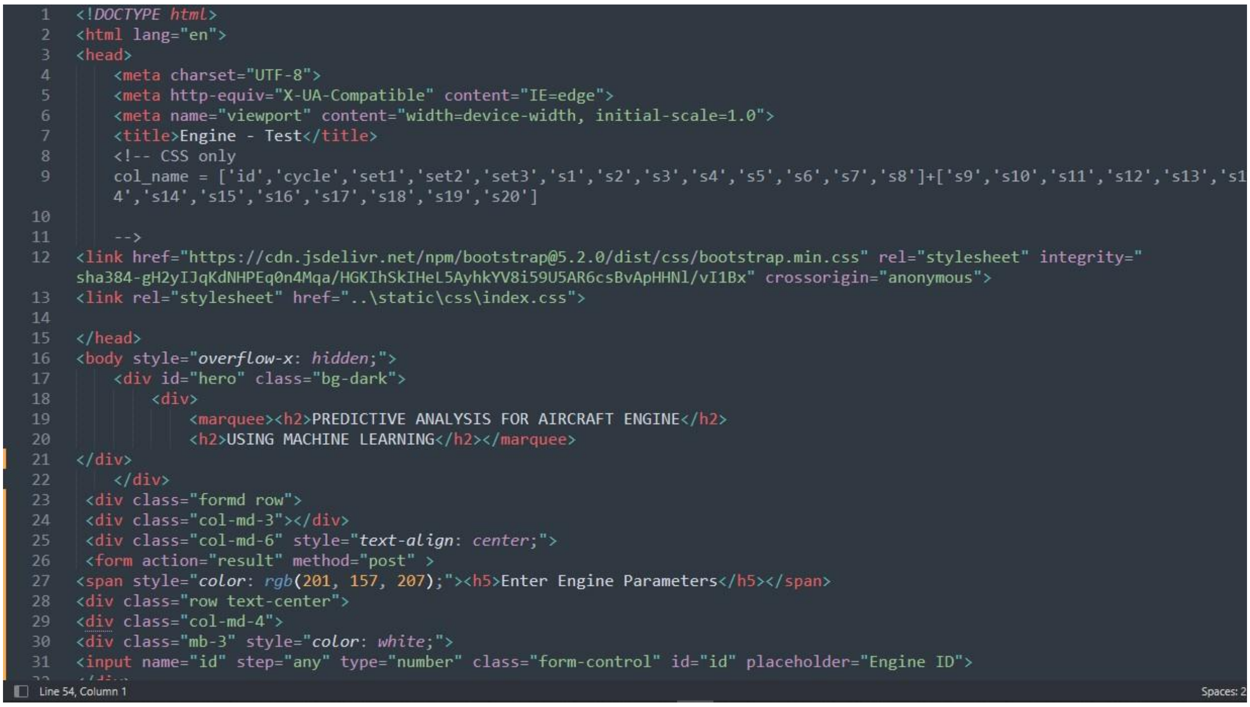
Register.html:



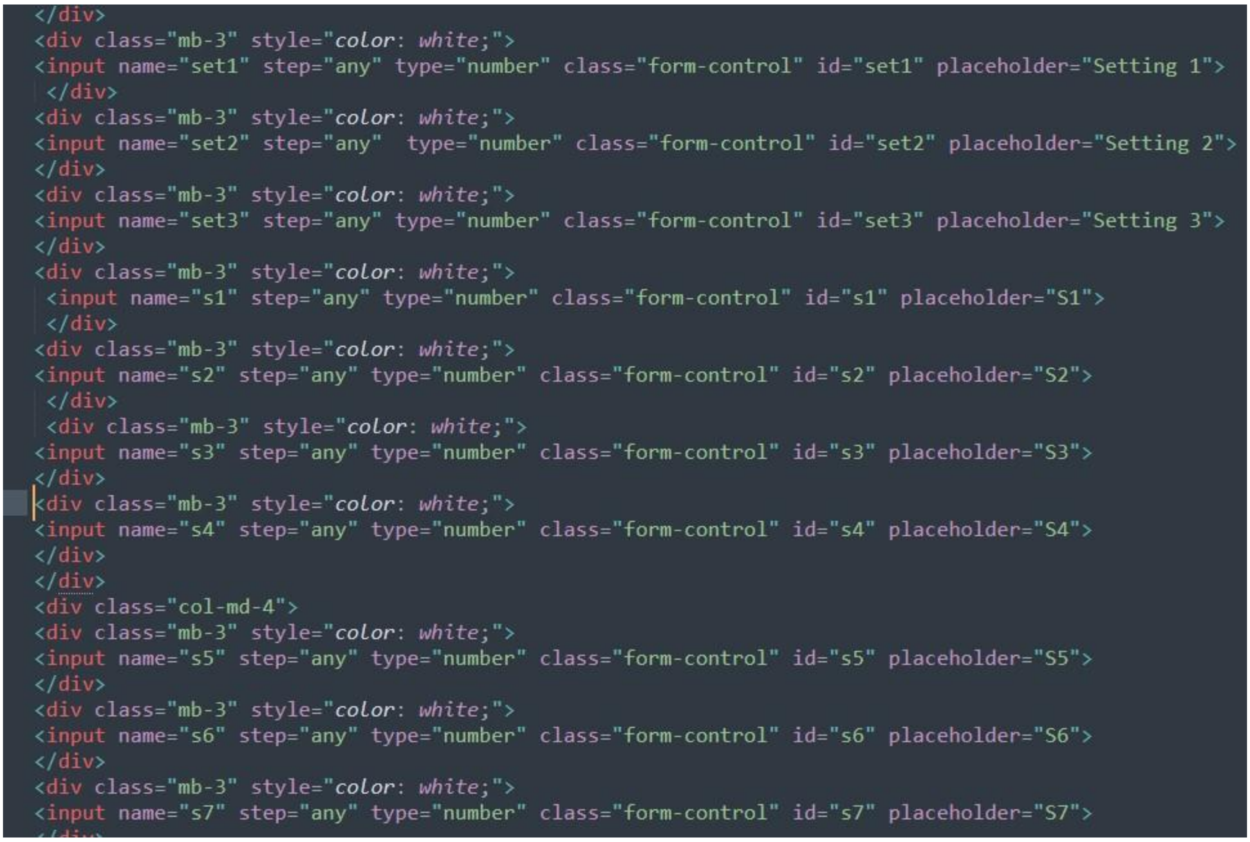
```
<!DOCTYPE html>
<head>
<title>My Website</title>
    <link rel="stylesheet" href="style1.css">
</head>

<body>

    <img src="user.png" class="user">

        <h1>Register Here</h1>

        <form name="myform2" action='/login'>

            <p>Username</p>
            <input type="text" name="uname1" placeholder="Enter Username" required="">

            <p>Email</p>
            <input type="Email" name="email" placeholder="Enter email id" required="">

            <p>Password</p>
            <input type="password" name="upswd1" placeholder="Enter Password" required="">

            <p>Retype Password</p>
            <input type="password" name="upswd2" placeholder="Re-Enter Password" required="">

             <div id="errorBox"></div>
            <input  type="submit" value="Register" >

            <br><br>
            <a href="/login">Existing user?login </a>
        </form>
    </div>
```

Line 9, Column 15

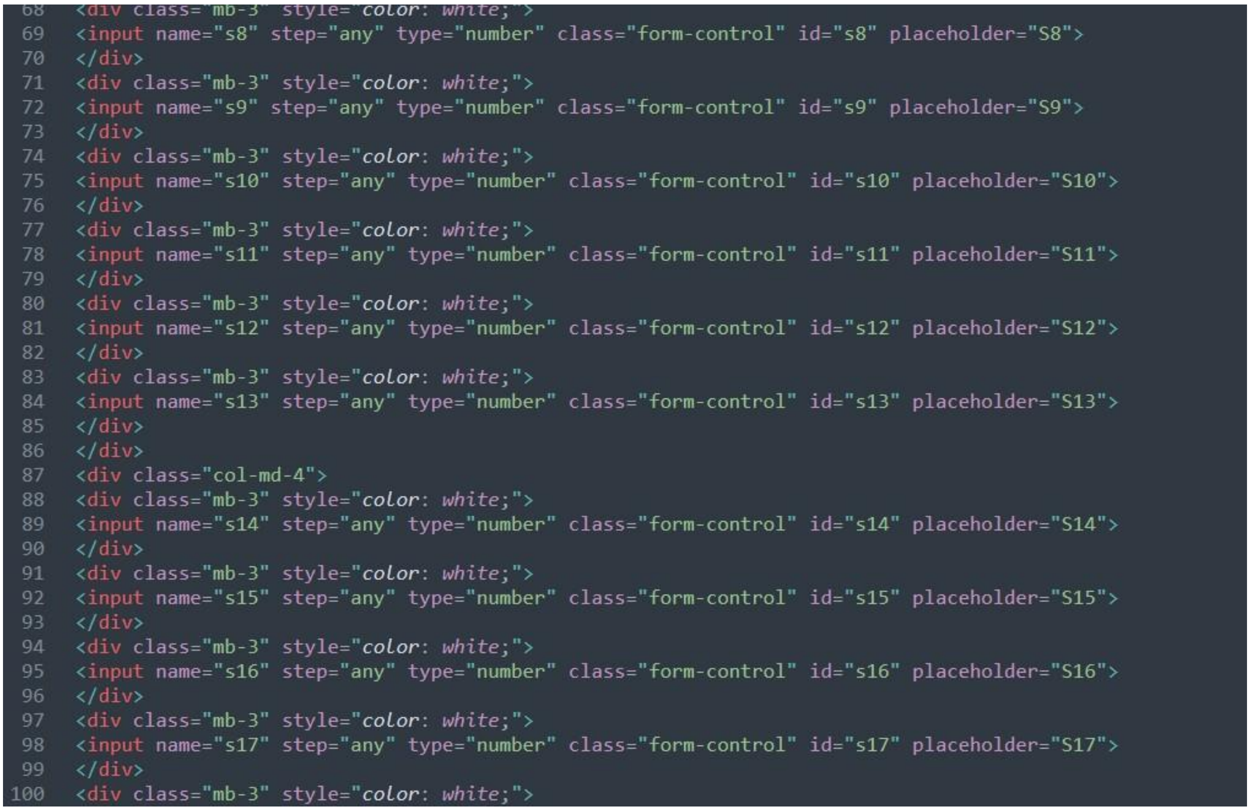
Index.html



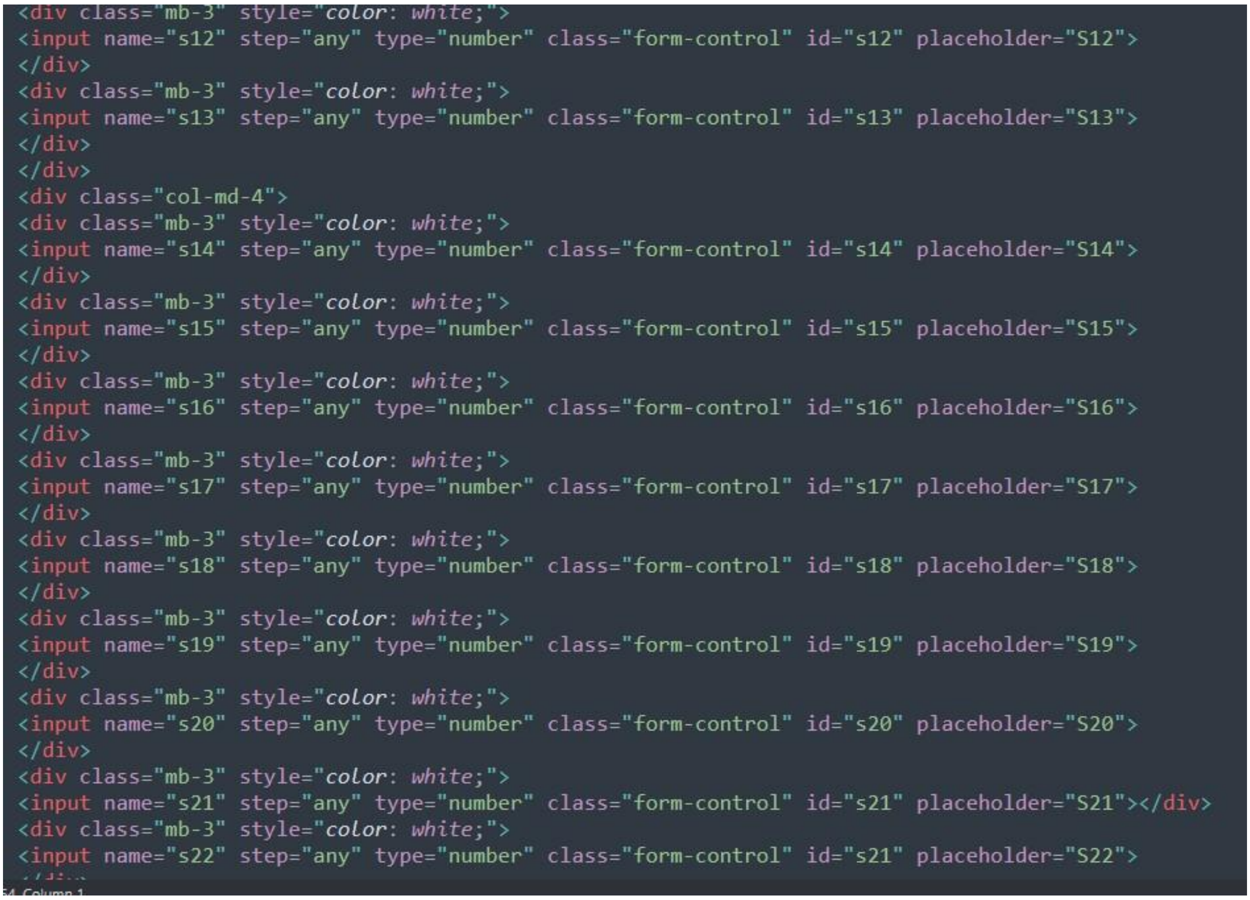
```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Engine - Test</title>
    <!-- CSS only
    col_name = ['id','cycle','set1','set2','set3','s1','s2','s3','s4','s5','s6','s7','s8']+['s9','s10','s11','s12','s13','s14','s14','s15','s16','s17','s18','s19','s20']

    -->
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.0/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-gH2yIJqKdNHPEq0n4Mqa/HGKIhSkIHeL5AyhkYV8i59U5AR6csBvApHHNl/vI1Bx" crossorigin="anonymous">
<link rel="stylesheet" href="..\static\css\index.css">

</head>
<body style="overflow-x: hidden;">
    <div id="hero" class="bg-dark">
        <div>
            <marquee><h2>PREDICTIVE ANALYSIS FOR AIRCRAFT ENGINE</h2>
            <h2>USING MACHINE LEARNING</h2></marquee>
</div>
    </div>
 <div class="formd row">
 <div class="col-md-3"></div>
 <div class="col-md-6" style="text-align: center;">
 <form action="result" method="post" >
<span style="color: rgb(201, 157, 207);"><h5>Enter Engine Parameters</h5></span>
<div class="row text-center">
<div class="col-md-4">
<div class="mb-3" style="color: white;">
<input name="id" step="any" type="number" class="form-control" id="id" placeholder="Engine ID">
```



```
</div>
<div class="mb-3" style="color: white;">
<input name="set1" step="any" type="number" class="form-control" id="set1" placeholder="Setting 1">
 </div>
<div class="mb-3" style="color: white;">
<input name="set2" step="any"  type="number" class="form-control" id="set2" placeholder="Setting 2">
</div>
<div class="mb-3" style="color: white;">
<input name="set3" step="any" type="number" class="form-control" id="set3" placeholder="Setting 3">
</div>
<div class="mb-3" style="color: white;">
 <input name="s1" step="any" type="number" class="form-control" id="s1" placeholder="S1">
 </div>
<div class="mb-3" style="color: white;">
<input name="s2" step="any" type="number" class="form-control" id="s2" placeholder="S2">
 </div>
 <div class="mb-3" style="color: white;">
<input name="s3" step="any" type="number" class="form-control" id="s3" placeholder="S3">
</div>
<div class="mb-3" style="color: white;">
<input name="s4" step="any" type="number" class="form-control" id="s4" placeholder="S4">
</div>
</div>
<div class="col-md-4">
<div class="mb-3" style="color: white;">
<input name="s5" step="any" type="number" class="form-control" id="s5" placeholder="S5">
</div>
<div class="mb-3" style="color: white;">
<input name="s6" step="any" type="number" class="form-control" id="s6" placeholder="S6">
</div>
<div class="mb-3" style="color: white;">
<input name="s7" step="any" type="number" class="form-control" id="s7" placeholder="S7">
```

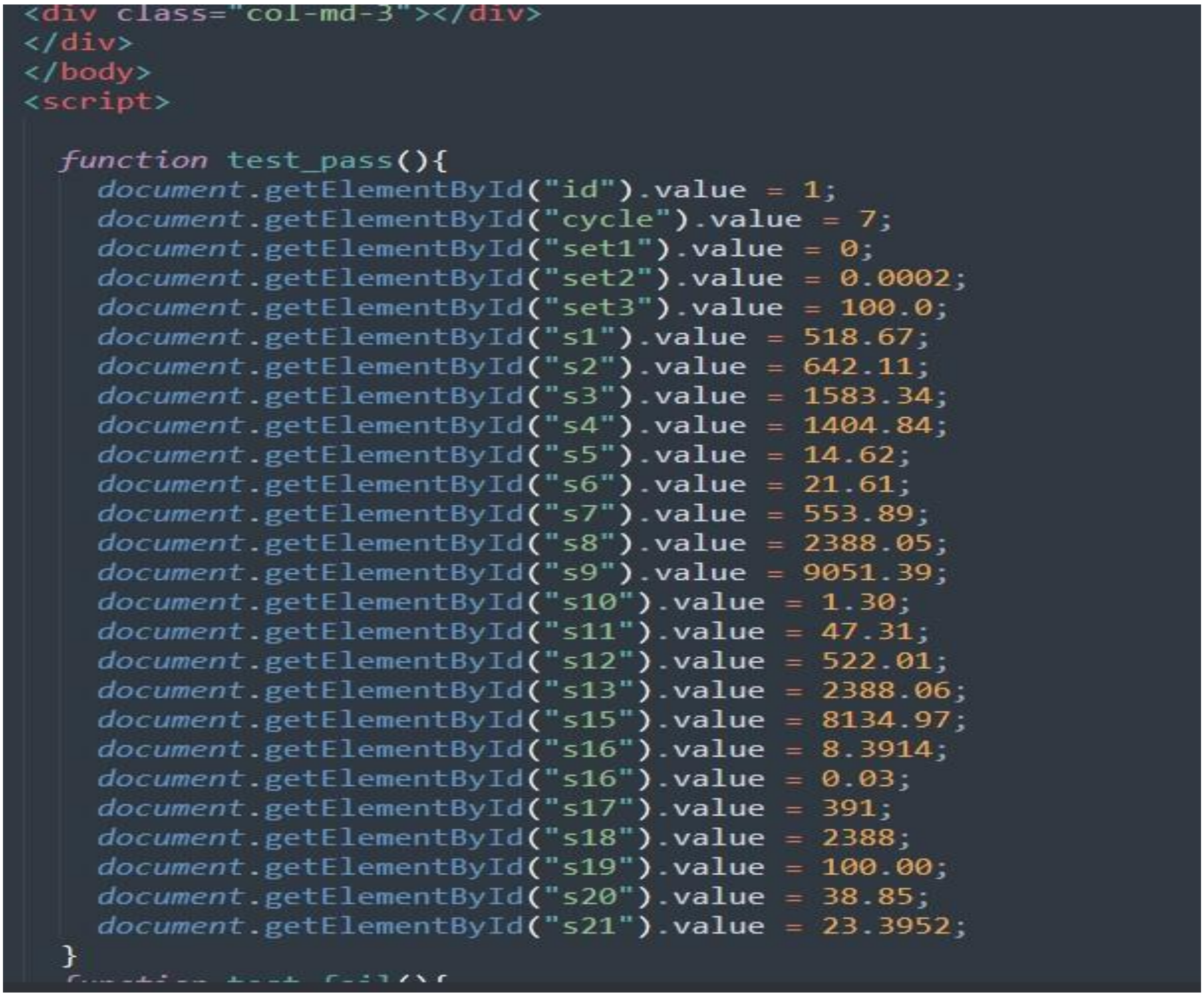


```
<div class="mb-3" style="color: white;">
<input name="s8" step="any" type="number" class="form-control" id="s8" placeholder="S8">
</div>
<div class="mb-3" style="color: white;">
<input name="s9" step="any" type="number" class="form-control" id="s9" placeholder="S9">
</div>
<div class="mb-3" style="color: white;">
<input name="s10" step="any" type="number" class="form-control" id="s10" placeholder="S10">
</div>
<div class="mb-3" style="color: white;">
<input name="s11" step="any" type="number" class="form-control" id="s11" placeholder="S11">
</div>
<div class="mb-3" style="color: white;">
<input name="s12" step="any" type="number" class="form-control" id="s12" placeholder="S12">
</div>
<div class="mb-3" style="color: white;">
<input name="s13" step="any" type="number" class="form-control" id="s13" placeholder="S13">
</div>
</div>
<div class="col-md-4">
<div class="mb-3" style="color: white;">
<input name="s14" step="any" type="number" class="form-control" id="s14" placeholder="S14">
</div>
<div class="mb-3" style="color: white;">
<input name="s15" step="any" type="number" class="form-control" id="s15" placeholder="S15">
</div>
<div class="mb-3" style="color: white;">
<input name="s16" step="any" type="number" class="form-control" id="s16" placeholder="S16">
</div>
<div class="mb-3" style="color: white;">
<input name="s17" step="any" type="number" class="form-control" id="s17" placeholder="S17">
</div>
<div class="mb-3" style="color: white;">
```

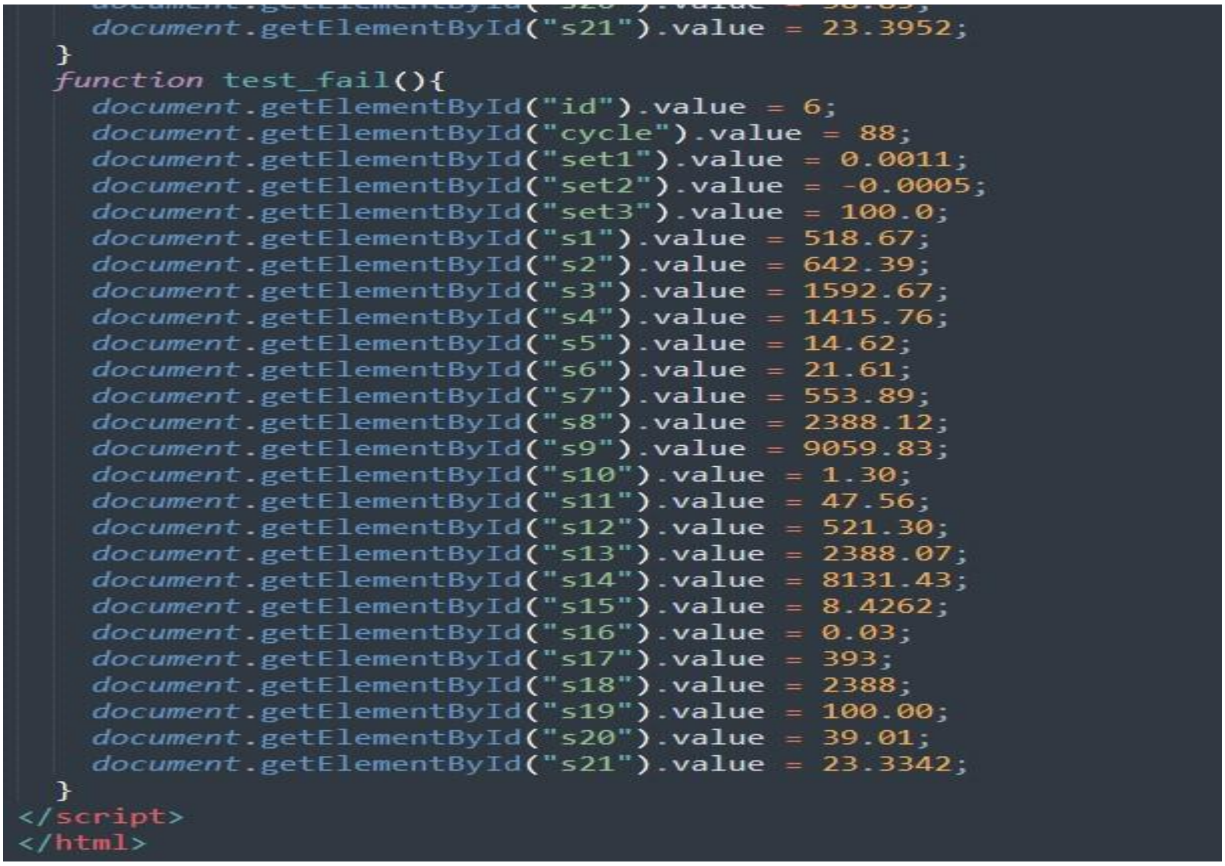


```
<div class="mb-3" style="color: white;">
<input name="s12" step="any" type="number" class="form-control" id="s12" placeholder="S12">
</div>
<div class="mb-3" style="color: white;">
<input name="s13" step="any" type="number" class="form-control" id="s13" placeholder="S13">
</div>
</div>
<div class="col-md-4">
<div class="mb-3" style="color: white;">
<input name="s14" step="any" type="number" class="form-control" id="s14" placeholder="S14">
</div>
<div class="mb-3" style="color: white;">
<input name="s15" step="any" type="number" class="form-control" id="s15" placeholder="S15">
</div>
<div class="mb-3" style="color: white;">
<input name="s16" step="any" type="number" class="form-control" id="s16" placeholder="S16">
</div>
<div class="mb-3" style="color: white;">
<input name="s17" step="any" type="number" class="form-control" id="s17" placeholder="S17">
</div>
<div class="mb-3" style="color: white;">
<input name="s18" step="any" type="number" class="form-control" id="s18" placeholder="S18">
</div>
<div class="mb-3" style="color: white;">
<input name="s19" step="any" type="number" class="form-control" id="s19" placeholder="S19">
</div>
<div class="mb-3" style="color: white;">
<input name="s20" step="any" type="number" class="form-control" id="s20" placeholder="S20">
</div>
<div class="mb-3" style="color: white;">
<input name="s21" step="any" type="number" class="form-control" id="s21" placeholder="S21"></div>
<div class="mb-3" style="color: white;">
<input name="s22" step="any" type="number" class="form-control" id="s21" placeholder="S22">
</div>
```

54 Column 1

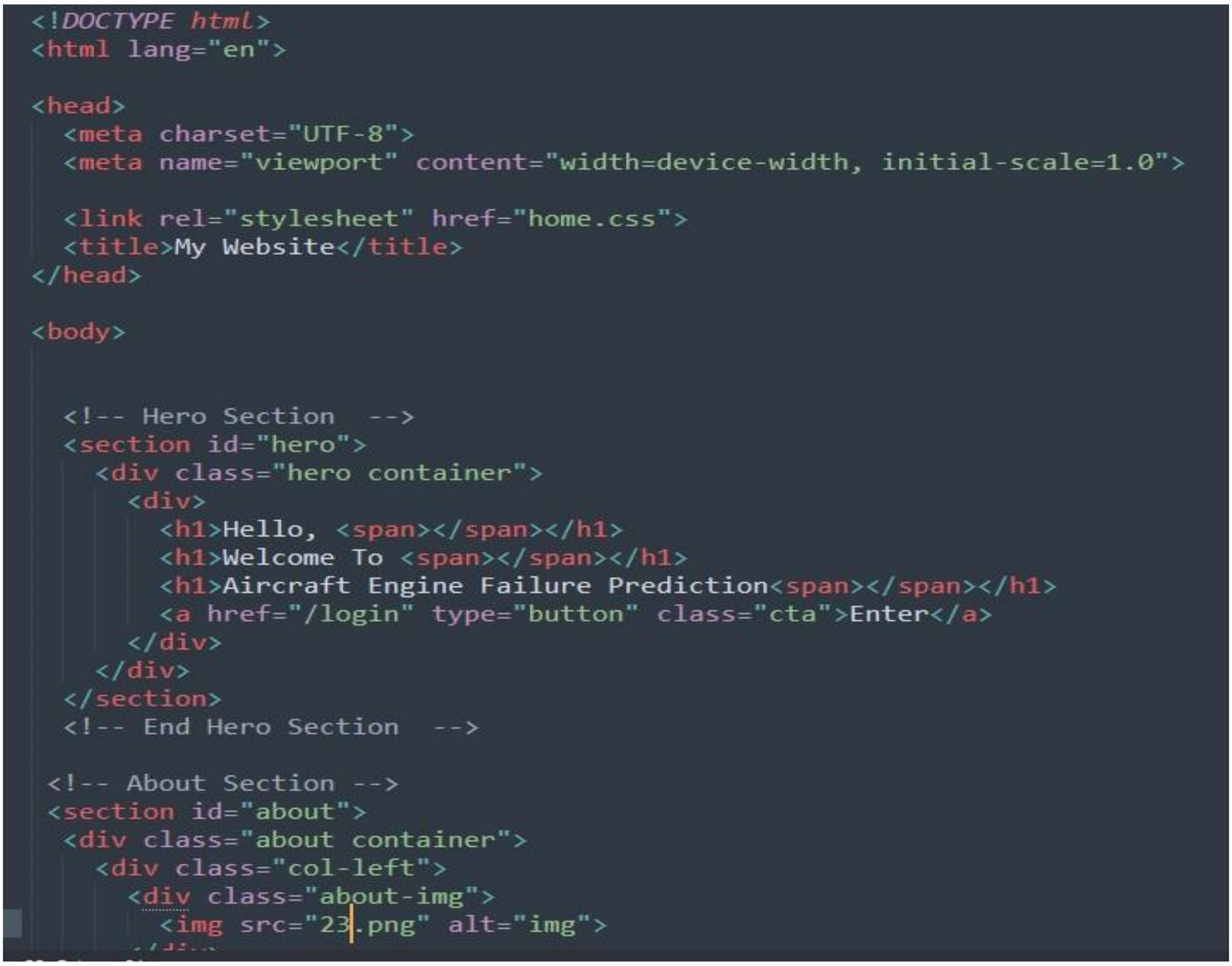


```
<div class="col-md-3"></div>
</div>
</body>
<script>

  function test_pass(){
    document.getElementById("id").value = 1;
    document.getElementById("cycle").value = 7;
    document.getElementById("set1").value = 0;
    document.getElementById("set2").value = 0.0002;
    document.getElementById("set3").value = 100.0;
    document.getElementById("s1").value = 518.67;
    document.getElementById("s2").value = 642.11;
    document.getElementById("s3").value = 1583.34;
    document.getElementById("s4").value = 1404.84;
    document.getElementById("s5").value = 14.62;
    document.getElementById("s6").value = 21.61;
    document.getElementById("s7").value = 553.89;
    document.getElementById("s8").value = 2388.05;
    document.getElementById("s9").value = 9051.39;
    document.getElementById("s10").value = 1.30;
    document.getElementById("s11").value = 47.31;
    document.getElementById("s12").value = 522.01;
    document.getElementById("s13").value = 2388.06;
    document.getElementById("s15").value = 8134.97;
    document.getElementById("s16").value = 8.3914;
    document.getElementById("s16").value = 0.03;
    document.getElementById("s17").value = 391;
    document.getElementById("s18").value = 2388;
    document.getElementById("s19").value = 100.00;
    document.getElementById("s20").value = 38.85;
    document.getElementById("s21").value = 23.3952;
  }
```

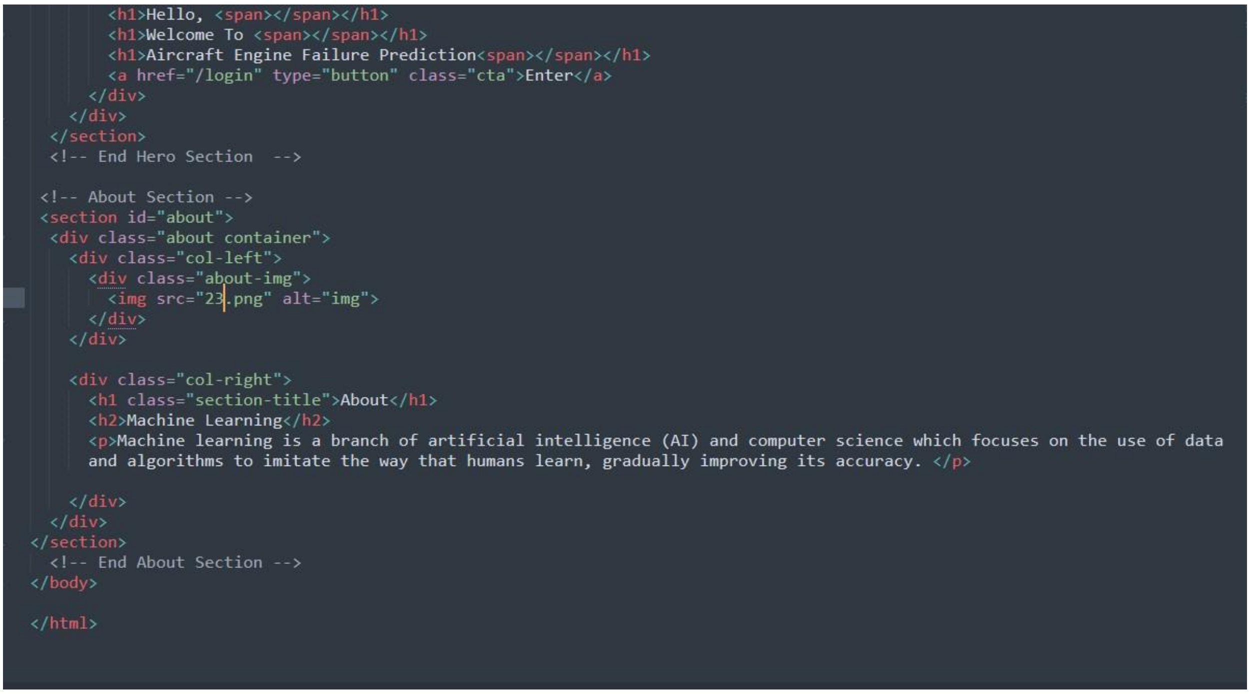


```
    document.getElementById("s21").value = 23.3952;
  }
  function test_fail(){
    document.getElementById("id").value = 6;
    document.getElementById("cycle").value = 88;
    document.getElementById("set1").value = 0.0011;
    document.getElementById("set2").value = -0.0005;
    document.getElementById("set3").value = 100.0;
    document.getElementById("s1").value = 518.67;
    document.getElementById("s2").value = 642.39;
    document.getElementById("s3").value = 1592.67;
    document.getElementById("s4").value = 1415.76;
    document.getElementById("s5").value = 14.62;
    document.getElementById("s6").value = 21.61;
    document.getElementById("s7").value = 553.89;
    document.getElementById("s8").value = 2388.12;
    document.getElementById("s9").value = 9059.83;
    document.getElementById("s10").value = 1.30;
    document.getElementById("s11").value = 47.56;
    document.getElementById("s12").value = 521.30;
    document.getElementById("s13").value = 2388.07;
    document.getElementById("s14").value = 8131.43;
    document.getElementById("s15").value = 8.4262;
    document.getElementById("s16").value = 0.03;
    document.getElementById("s17").value = 393;
    document.getElementById("s18").value = 2388;
    document.getElementById("s19").value = 100.00;
    document.getElementById("s20").value = 39.01;
    document.getElementById("s21").value = 23.3342;
  }
</script>
</html>
```

Home.html



```
<!DOCTYPE html>
<html lang="en">

<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">

  <link rel="stylesheet" href="home.css">
  <title>My Website</title>
</head>

<body>


  <!-- Hero Section  -->
  <section id="hero">
    <div class="hero container">
      <div>
        <h1>Hello, <span></span></h1>
        <h1>Welcome To <span></span></h1>
        <h1>Aircraft Engine Failure Prediction<span></span></h1>
        <a href="/login" type="button" class="cta">Enter</a>
      </div>
    </div>
  </section>
  <!-- End Hero Section  -->

 <!-- About Section -->
 <section id="about">
  <div class="about container">
    <div class="col-left">
      <div class="about-img">
        <img src="23.png" alt="img">
```



```
        <h1>Hello, <span></span></h1>
        <h1>Welcome To <span></span></h1>
        <h1>Aircraft Engine Failure Prediction<span></span></h1>
        <a href="/login" type="button" class="cta">Enter</a>
      </div>
    </div>
  </section>
  <!-- End Hero Section  -->

 <!-- About Section -->
 <section id="about">
  <div class="about container">
    <div class="col-left">
      <div class="about-img">
        <img src="23.png" alt="img">
```

```
      </div>
    </div>

    <div class="col-right">
      <h1 class="section-title">About</h1>
      <h2>Machine Learning</h2>
      <p>Machine learning is a branch of artificial intelligence (AI) and computer science which focuses on the use of data
      and algorithms to imitate the way that humans learn, gradually improving its accuracy. </p>

    </div>
  </div>
</section>
  <!-- End About Section -->
</body>

</html>
```

8. TESTING

8.1 Test Cases

- Login Page
- Prediction Page
- Result Page

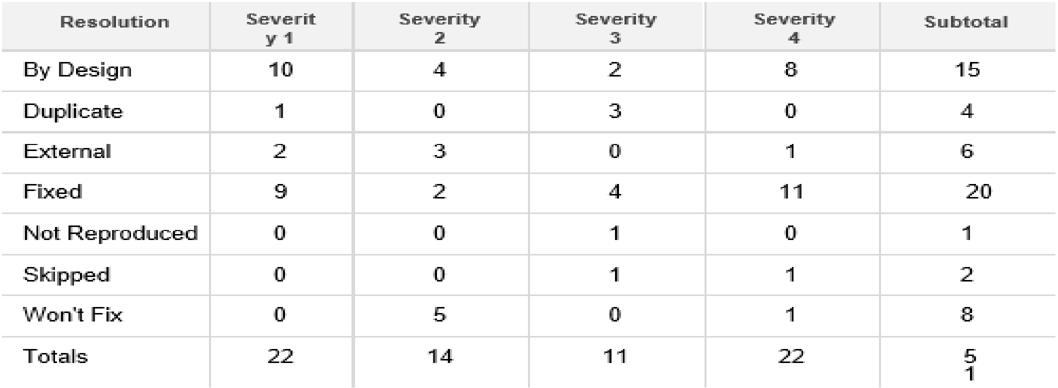
8.2 User Acceptance Testing

1. Purpose of Document

The purpose of this document is to briefly explain the test coverage and open issues of the [Product Name] project at the time of the release to User Acceptance Testing (UAT).

2. Defect Analysis

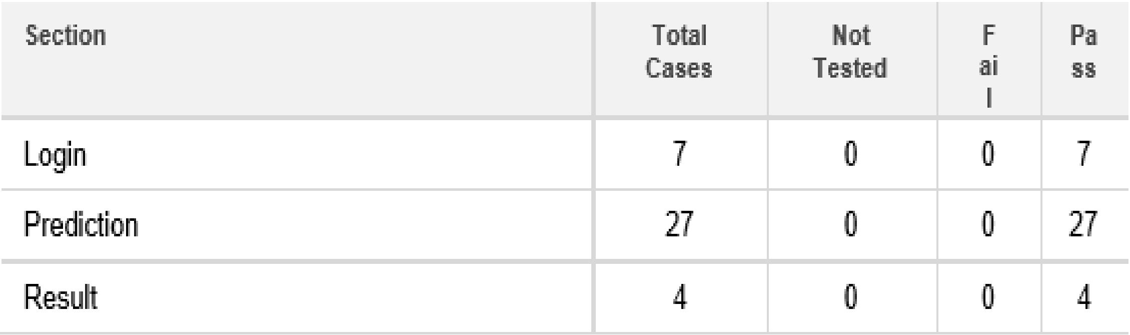
This report shows the number of resolved or closed bugs at each severity level, and how they were resolved



Resolution	Severity 1	Severity 2	Severity 3	Severity 4	Subtotal
By Design	10	4	2	8	15
Duplicate	1	0	3	0	4
External	2	3	0	1	6
Fixed	9	2	4	11	20
Not Reproduced	0	0	1	0	1
Skipped	0	0	1	1	2
Won't Fix	0	5	0	1	8
Totals	22	14	11	22	51

3. Test Case Analysis

This report shows the number of test cases that have passed, failed, and untested

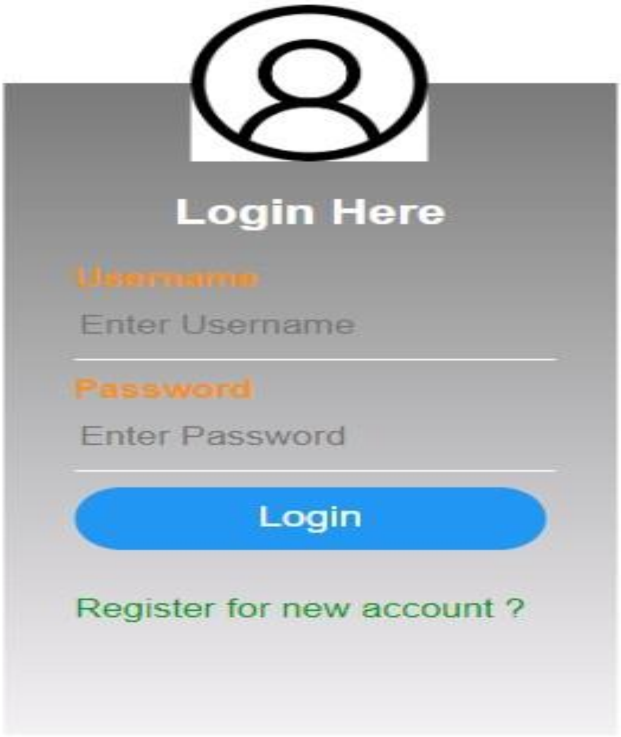


Section	Total Cases	Not Tested	Fail	Pass
Login	7	0	0	7
Prediction	27	0	0	27
Result	4	0	0	4

9. RESULTS

9.1 Performance Metrics

Login.html



Login Here

Username

Enter Username

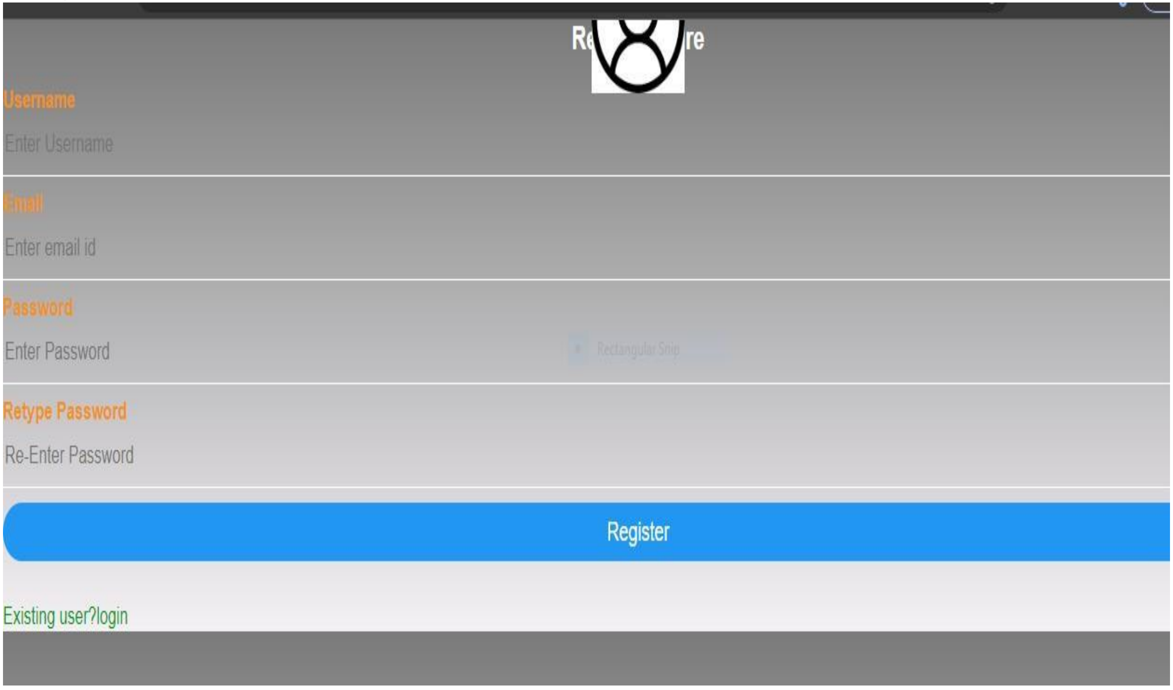
Password

Enter Password

Login

Register for new account ?

Register.html



Re re

Username

Enter Username

Email

Enter email id

Password

Enter Password

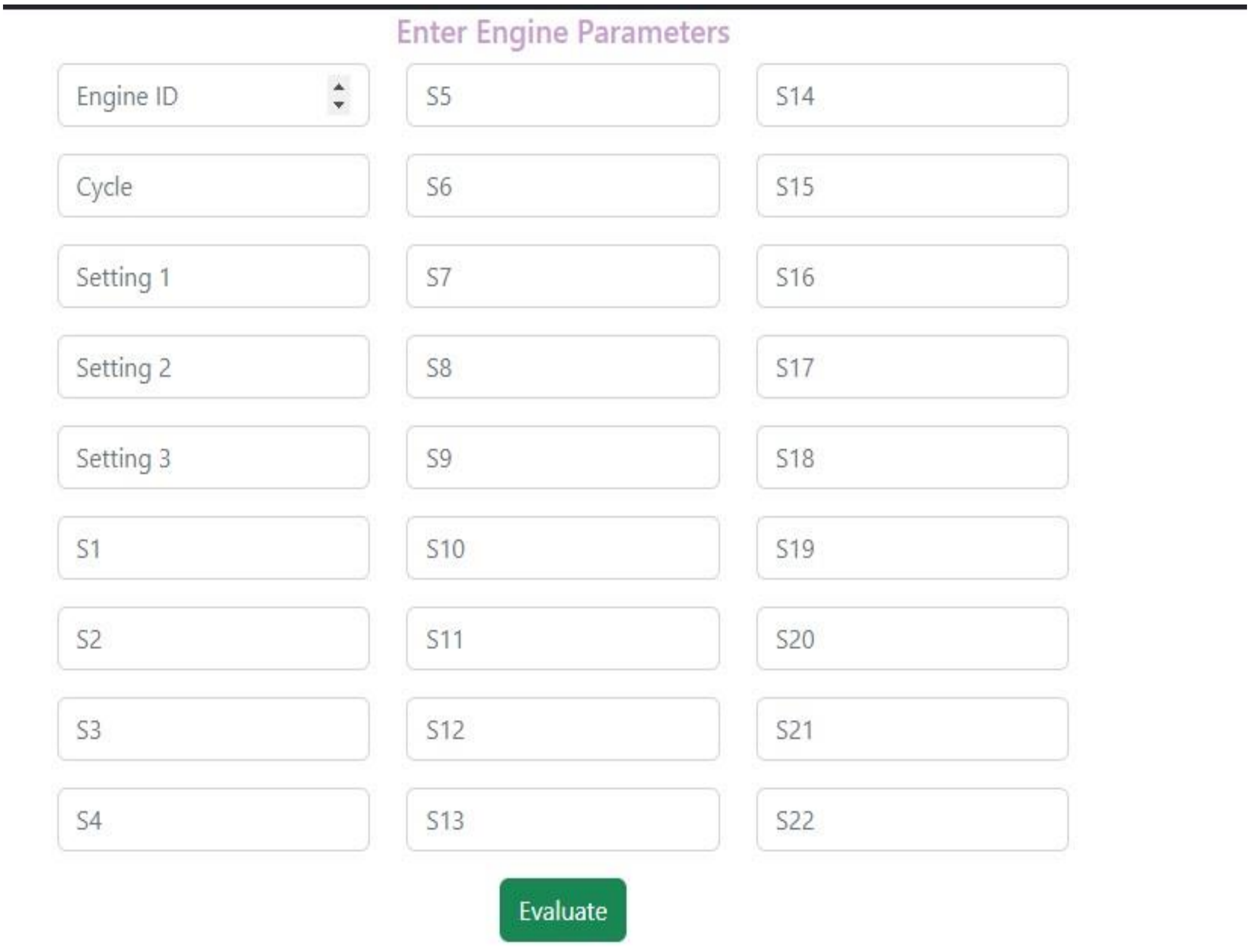
Retype Password

Re-Enter Password

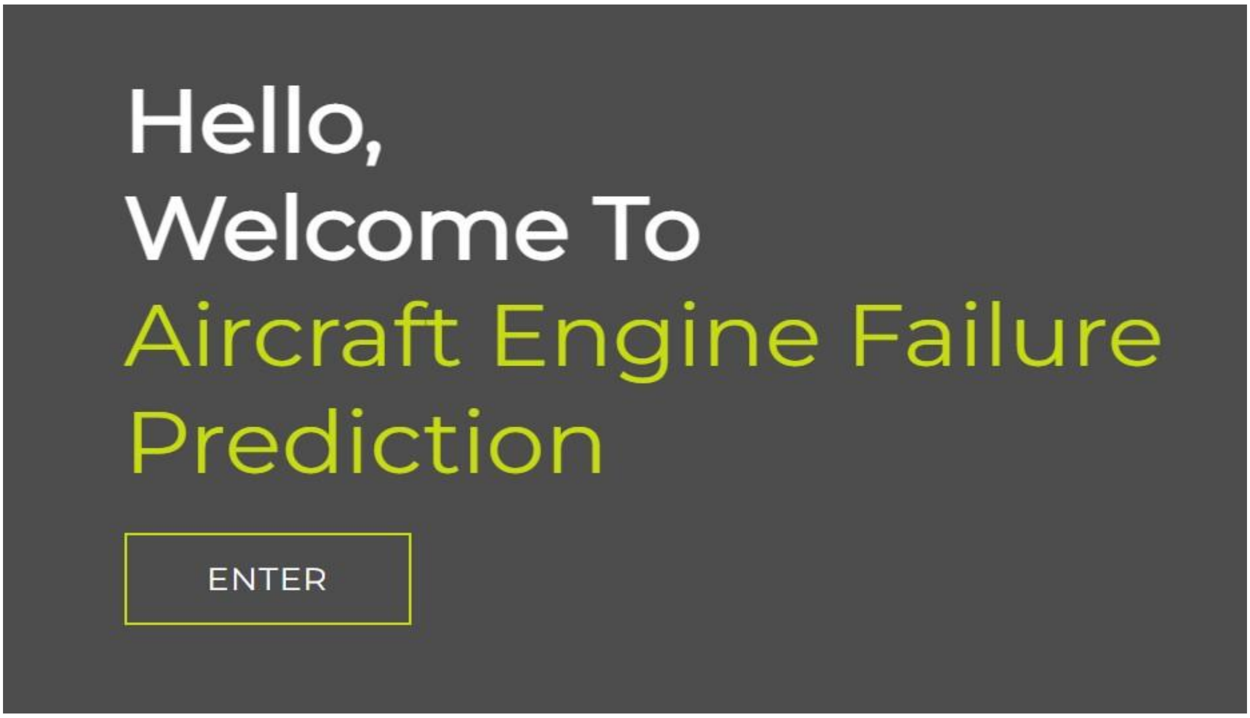
Register

Existing user?login

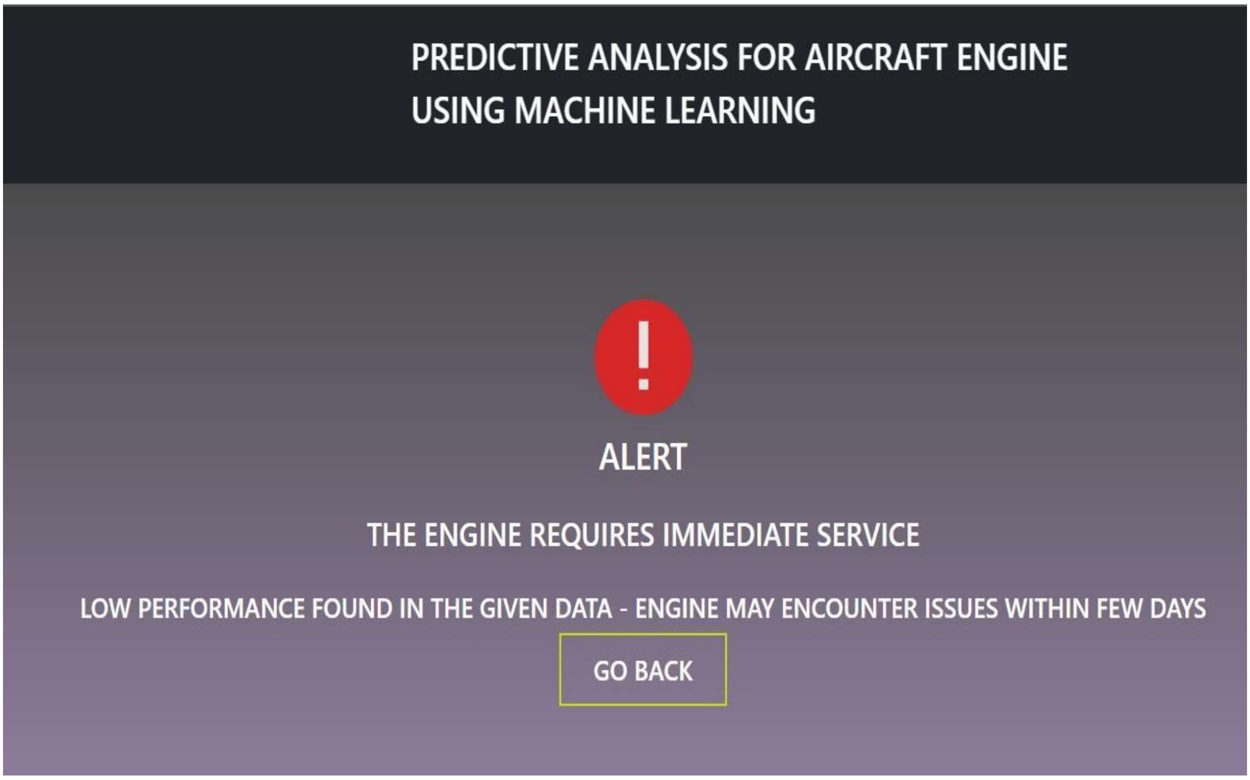
Index.html


Enter Engine Parameters
Engine ID
Cycle
Setting 1
Setting 2
Setting 3
S1
S2
S3
S4
S5
S6
S7
S8
S9
S10
S11
S12
S13
S14
S15
S16
S17
S18
S19
S20
S21
S22
Evaluate

Home.html



Hello,
Welcome To
Aircraft Engine Failure Prediction



PREDICTIVE ANALYSIS FOR AIRCRAFT ENGINE USING MACHINE LEARNING

ALERT

THE ENGINE REQUIRES IMMEDIATE SERVICE

LOW PERFORMANCE FOUND IN THE GIVEN DATA - ENGINE MAY ENCOUNTER ISSUES WITHIN FEW DAYS

10. ADVANTAGES & DISADVANTAGES

ADVANTAGES

Machine learning and data science can predict future events, trends, and customer behavior to a certain extent. These predictions can enable businesses to make better decisions about where to allocate resources and how to respond to changes in the market .

Machine learning algorithms use historical data as input to predict new output values. Recommendation engines are a common use case for machine learning. Other popular uses include fraud detection, spam filtering, malware threat detection, business process automation (BPA) and Predictive maintenance.

With the ever-growing volume of data generated every day, it is increasingly difficult for humans to process and make sense of all this information. Machine learning can help businesses handle large amounts of data more efficiently and effectively and even use decision trees to take action on the information.

As humans after gaining experience improve themselves in the same way machine learning improve themselves and become more accurate and efficient in work. This led to better decisions. For example, in the weather forecast, the more data. And experience the machine gets the more advanced forecast it will provide.

DISADVANTAGES

Although machine learning is considered to be more accurate it is highly vulnerable. For example, a set of programs provided to the machine may be biased or consist of errors. The same program is used to make another forecast or prediction then there will be a chain of errors that could be formed which may, although recognized but take some time to find out the source of the error.

The more data a machine gets the more accurate and efficient it becomes thus more data is required to input to the machine for better forecasting or decision making. But it may sometimes not be possible. Also, the data must be unbiased and of good quality. Data requirements are problematic sometimes.

As we have already seen that a little manipulation or biased data could lead to a long drawn error chain and therefore there are chances of the inaccuracy of interpretation also. Sometimes data without any error could also be interpreted inaccurately by the machine as the data provided previously may not fulfill all the basics of the machine

11. CONCLUSION

Overall, the results show that by bringing together sufficient (big) high quality data, robust machine learning algorithms, and data science, machine learning-based predictive analytics can be an effective tool for engine design-space exploration during the conceptual design phase. It would help to identify the best engine design expeditiously amongst several candidates. The promising results of the predictive analytics show that machine-learning techniques merit further exploration for application in aircraft engine conceptual design.To further improve the accuracy (and reduce the uncertainty) of TSFC prediction, the database needs to be expanded. However, the limitation of publicly available engine data is a challenge to overcome.

12. FUTURE SCOPE

Early predictions avoid the accident and other problems.

- The process maintenance become easier.
- Predicting future also saves the money and the resources.
- Controls the machine and its performance.
- Train model in various machines can useful for the performance and maintenance.
- Machine learning algorithms can used for the models and the models monitor the performances.
- The algorithms can be update in high performance like the solution it will find itself.

13. APPENDIX

SOURCE CODE

The source code has been uploaded in GitHub. To refer the final source code click

SOURCE CODE

GITHUB & PROJECT DEMO LINK:

The GitHub link: https://github.com/IBM-EPBL/IBM-Project-46277-1660744360

The project link: https://youtu.be/q4Z76WWM4l0