FSM Mid-Internship Review

INTP23-ML-05
Equipment Failure Prediction for Predictive maintenance

Nancy
Indira Gandhi Delhi Technical University for women

Under Mentorship of
Dhairya

IITD-AIA FOUNDATION FOR SMART MANUFACTURING

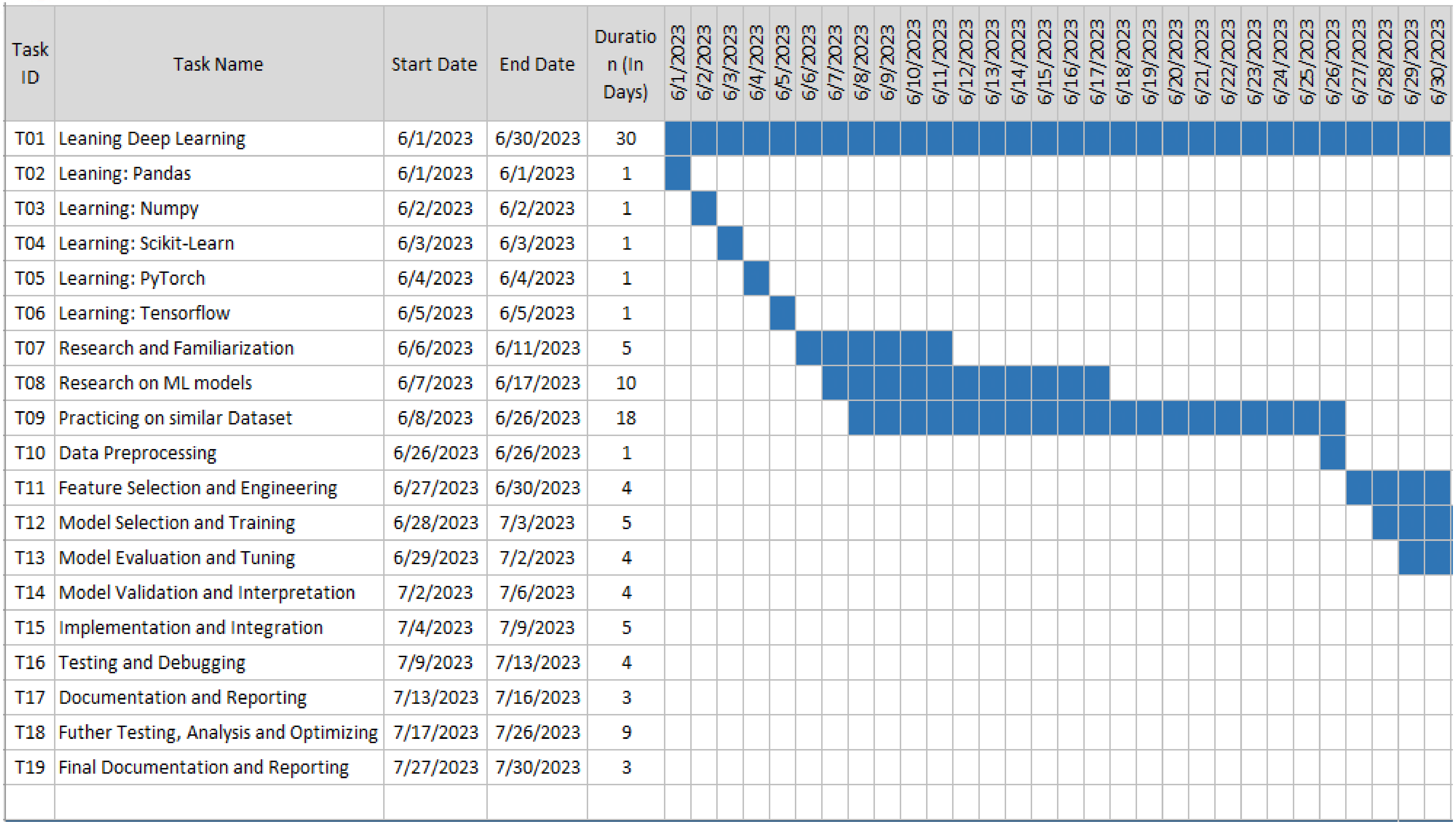
Objectives

This project aims to enhance maintenance strategies, reduce downtime, and optimize the reliability and performance of critical machinery. The objective of equipment failure for predictive maintenance would be to predict when the machine is likely to fail so that maintenance can be scheduled based on equipment conditions. This approach can help minimize equipment downtime while also reducing maintenance costs by eliminating unnecessary scheduled maintenance.

Objectives:

- Understanding about predictive maintenance and the smart factory (90%) and challenges faced in smart manufacturing
- Learning various Machine learning and Deep learning techniques to develop a model to predict equipment failure. (90%)
- Learn how to deal with a real dataset: as of now I am working on similar dataset NASA turbofan engine dataset (70%)
- Learn about working of sharing Machine (on this dataset we will be working to predict equipment failure) (90%)
- To create a model which predicts the equipment failure with the highest possible accuracy. (70% on the similar dataset)
- Implement multiple models and try to find the best model. (50% on the similar dataset)

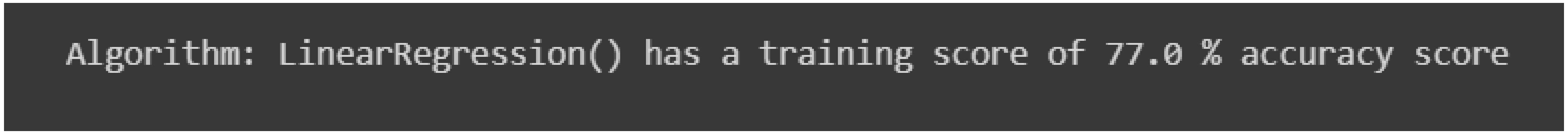
Timeline - Gantt chart



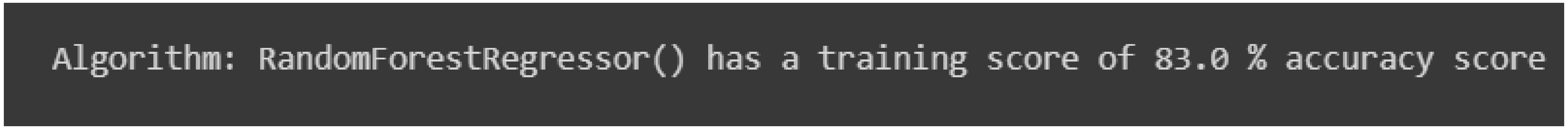
Task ID	Task Name	Start Date	End Date	Duration (In Days)	6/1/2023	6/2/2023	6/3/2023	6/4/2023	6/5/2023	6/6/2023	6/7/2023	6/8/2023	6/9/2023	6/10/2023	6/11/2023	6/12/2023	6/13/2023	6/14/2023	6/15/2023	6/16/2023	6/17/2023	6/18/2023	6/19/2023	6/20/2023	6/21/2023	6/22/2023	6/23/2023	6/24/2023	6/25/2023	6/26/2023	6/27/2023	6/28/2023	6/29/2023	6/30/2023
T01	Leaning Deep Learning	6/1/2023	6/30/2023	30	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦
T02	Leaning: Pandas	6/1/2023	6/1/2023	1	🟦																													
T03	Learning: Numpy	6/2/2023	6/2/2023	1		🟦																												
T04	Learning: Scikit-Learn	6/3/2023	6/3/2023	1			🟦																											
T05	Learning: PyTorch	6/4/2023	6/4/2023	1				🟦																										
T06	Learning: Tensorflow	6/5/2023	6/5/2023	1					🟦																									
T07	Research and Familiarization	6/6/2023	6/11/2023	5						🟦	🟦	🟦	🟦	🟦	🟦																			
T08	Research on ML models	6/7/2023	6/17/2023	10							🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦													
T09	Practicing on similar Dataset	6/8/2023	6/26/2023	18								🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦				
T10	Data Preprocessing	6/26/2023	6/26/2023	1																										🟦				
T11	Feature Selection and Engineering	6/27/2023	6/30/2023	4																											🟦	🟦	🟦	🟦
T12	Model Selection and Training	6/28/2023	7/3/2023	5																												🟦	🟦	🟦
T13	Model Evaluation and Tuning	6/29/2023	7/2/2023	4																													🟦	🟦
T14	Model Validation and Interpretation	7/2/2023	7/6/2023	4																														
T15	Implementation and Integration	7/4/2023	7/9/2023	5																														
T16	Testing and Debugging	7/9/2023	7/13/2023	4																														
T17	Documentation and Reporting	7/13/2023	7/16/2023	3																														
T18	Futher Testing, Analysis and Optimizing	7/17/2023	7/26/2023	9																														
T19	Final Documentation and Reporting	7/27/2023	7/30/2023	3																														

Screenshots of development

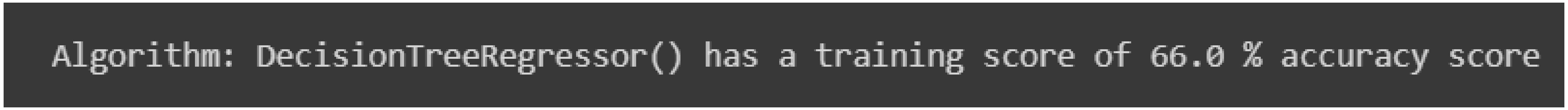
This month, I tried to learn various concepts of Machine learning and deep learning which will helps me to implement model more accurate way. I learnt about working of shearing machine and time series analysis. Leant about ensemble and boosting methods. Results of models I have implemented:



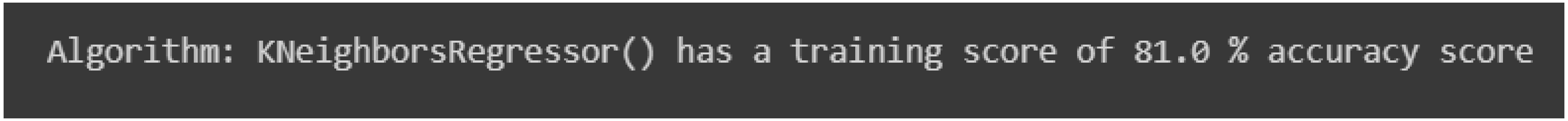
```
Algorithm: LinearRegression() has a training score of 77.0 % accuracy score
```



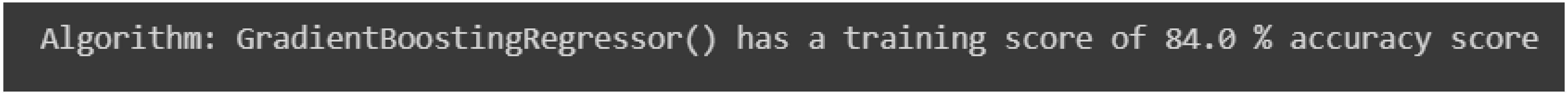
```
Algorithm: RandomForestRegressor() has a training score of 83.0 % accuracy score
```



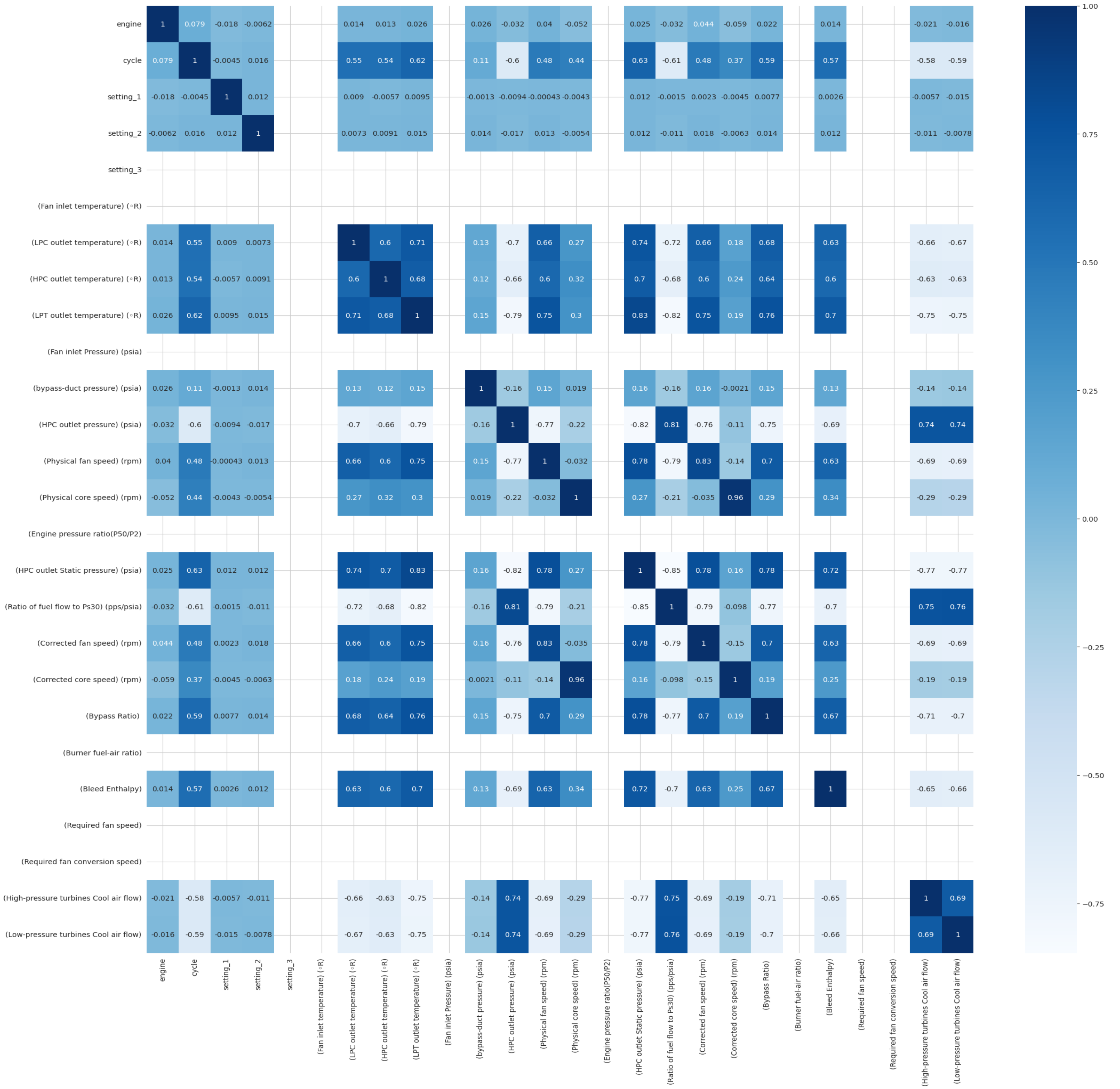
```
Algorithm: DecisionTreeRegressor() has a training score of 66.0 % accuracy score
```



```
Algorithm: KNeighborsRegressor() has a training score of 81.0 % accuracy score
```



```
Algorithm: GradientBoostingRegressor() has a training score of 84.0 % accuracy score
```



Correlation Metrics of features of NASA turbofan engine dataset

{Out of 27 columns or 27 Features only 10 Features are important to determine RUL(REMAING USEFUL LIFE)}

Thank You

nancy045btcse20@igdtuw.ac.in