A project report on

Predictive Maintenance

Submitted by

Y.V. Sai Yaswanth Reddy (12307855)

I. Maria Tejaswi (12309910)

P. Lavanya (12300764)

Manan Awasthi (12306419)

P. Balaji (12308401)

Course Code: PETV79L

Students of

B.TECH

COMPUTER SCIENCE AND ENGINEERING

LOVELY PROFESSIONAL UNIVERSITY

2023-2027

Submitted to

Prof. Mahipal Singh Papola

1.ABSTRACT

2.INTRODUCTION

3.DATA MINING TASK IDENTIFICATION

4.METHODS/TECHNIQUES APPLIED AND THEIR BRIEF DESCRIPTION

4.1. DECISION TREE
4.2. RANDOM FOREST
4.3. XG-BOOST

5.DATASET DESCRIPTION

5.1. FEATURE DESCRIPTION

5.2. EXPLORATORY DATA ANALYSIS

5.3. CORRELATION MATRIX

6. DATA PREPROCESSING

6.1. DATA MANIPULATION
6.2. FEATURE SCALING
6.3. LABEL ENCODING
6.4. FEATURE SELECTION

7. MODEL EVALUATION

7.1 TRAIN/TEST SPLIT
7.2 CONFUSION MATRIX
7.3 CLASSIFICATION REPORT
7.4 AREA UNDER CURVE

8. RESULTS AND DISCUSSION

8.1 Comparison of various models:

9.SCREENSHOTS OF RESULTS

10. CONCLUSION

REFERENCES

Lovely Professional University, Punjab

BONAFIDE CERTIFICATE

Certified that this project report **"Predictive System Maintenance"** is the bonafide work of "Y.V. Sai Yaswanth Reddy, I. Maria Tejaswi, P. Lavanya, Manan Awasthi, and P. Balaji" who carried out the project work under my supervision.

SIGNATURE

<<Name of the Supervisor>>

Y.V. Sai Yaswanth Reddy

I. Maria Tejaswi

P. Lavanya

Manan Awasthi

P. Balaji

SIGNATURE

<<Signature of the HOD>>

SIGNATURE

<<Name>>

HEAD OF THE DEPARTMENT

<<Signature of the Supervisor>>

1.ABSTRACT

This project focuses on predicting system failures in machines using machine learning. The dataset contains sensor data such as air temperature, process temperature, torque, toolwear, and rotational speed when it reaches their threshold they may occur the system failure, which in turn results in system failure. The following project aims to taking data from a Kaggle and used it to train classification models that will be able to predict future system failure. The project uses the concepts of Data Preprocessing, Classification and Ensemble Learning.

2.INTRODUCTION

To prevent unforeseen breakdowns and costly stoppages, this project endeavors to reliably forecast if a machine is likely to fail in the near future, leveraging real-time sensor and operating data. Equipment failure in manufacturing, aerospace, energy, and logistics industries not only results in economic losses but also jeopardizes safety and productivity. Thus, predictive maintenance is one of the most critical strategies used in industrial operations, where the companies identify early warning signs of failure and act to prevent a breakdown from happening.

As machines become more sophisticated and coupled with digital monitoring systems, the amount and level of operational data grow exponentially. Processing and analyzing this data manually is challenging in big organizations where hundreds of machines are running around the clock. Therefore, automated failure prediction with the help of machine learning has emerged as a promising solution to manage such problems efficiently.

Failure prediction is a part of predictive analytics, where one attempts to measure the probability of failure based on past and current data. In the traditional methods, statistical models were applied to detect failures, but they tend to fail with multivariate complex patterns imbalanced data, and scalability issues. To overcome these downsides, data mining and machine learning have gained popularity in handling more precise, scalable, and flexible solutions.

In order to identify patterns in machine behavior and predict failures, this project utilizes advanced machine learning models, including Decision Tree, Random Forest, and XG BOOST. Feature engineering is utilized to extract valuable insights from raw data. This system provides industries with a handy and efficient tool to enhance operating continuity, lower maintenance expenditure, and enhance safety.

3.DATA MINING TASK IDENTIFICATION

Finding hidden patterns, trends, and valuable information in large datasets is referred to as data mining. In order to extract relevant information that can aid in well-informed decision-making, a variety of analytical techniques must be applied. Data mining tasks can be divided into regression, classification, clustering, data visualization, and other categories based on the kinds of insights that can be gleaned.

About 10% of the failure detections in this project's dataset are present. In the dataset, "1" denotes failure detection and "0" denotes no failure detection. Therefore, predicting whether a system will detect failure is the project's task. Since there are only two classes for the output—failure or no failure—the issue is classified as a binary classification problem. The project uses a variety of machine learning algorithms, including XG BOOST, Random Forest, and Decision Tree, to classify the data.

4. METHODS/TECHNIQUES APPLIED AND THEIR BRIEF DESCRIPTION

We used following classification models:

1.Decision tree

2.Random Forest

3.XG-Boost

4.1. Decision tree:

A Decision Tree is a supervised learning algorithm primarily employed in classification problems. It splits the dataset into subsets based on feature values and constructs a tree-like model of decision. Each of the internal nodes of the tree is a decision made based on a feature, and each leaf node is an outcome or class label.

Advantages of Using Decision Tree:

- Easy to Interpret and Understand: It is easy to see how the decisions are made by visualizing the tree structure.
- No feature scaling required: Both unscaled and scaled data can be used by the model.
- Supports both Numerical and Categorical data, in which appropriate for read world sensor data.
- Automatic feature selection by the tree and highlighting the most significant features while training.

4.2. Random Forest

The Random Forest is a machine learning algorithm and belongs to a class of classifiers called ensemble methods, which build a number of decision trees and use their output to make a prediction with higher accuracy and stability. It is a supervised learning method and is very common amongst classification and regression tasks.

The model operates to generate many decision trees based on various random data points and properties. When predicting, each tree outputs a value, and final prediction is the majority vote of all the trees.

Advantages of Using Random Forest:

- High Accuracy - Various trees (which are generated using different samples, input variables and random subsets of input data in the model) are combined and it makes the predictions more accurate in compare with any individual trees.
- Prevents Overfitting: The model is less likely to overfit due to the randomness in data sampling and feature selection.
- Can Handle Missing and Noisy Data: It is robust enough to handle missing data and noisy data without compromising its performance.
- Both Classification and Regression Capable: It is very useful in the case of different problems.
- No Feature Scaling Required: Works reasonably well even if the input features are not scaled, though it is recommended to scale features to speed up convergence.

4.3. XG-BOOST (Extreme Gradient Boosting)

XGBoost is a strong and effective machine learning model belonging to the family of boosting ensemble methods. The boosting technique is a method used to construct many weak learners in series and combine them to create a strong learner. The concept is to place stronger emphasis on the areas where the earlier models did poorly, with increasing accuracy.

XGBoost handles the usual bias-variance trade-off for machine learning models. A high bias causes underfitting and high variance results in overfitting. However, XGBoost employs multiple methods of balancing between them to generalize to unseen data more effectively.

Advantages of Using XGBoost:

- **Regularization:**

Contrary to the regular Gradient Boosting Machines which do not have this feature, XGBoost provides L1 and L2 regularization for controlling model complexity and reducing the overfitting of the model.

This is the main reason XGBoost is sometimes called a "regularized boosting" algorithm.

- **Parallel Preprocessing:**

The reason XGBoost is fast is that it trains with the help of parallel computations supported, hence, the learning process gets a significant speedup.

- **High Flexibility:**

By allowing users to define a custom objective function and evaluation metrics, it brings a high level of flexibility for a wide range of problem domains.

- **Advanced Tree Pruning:**

Unlike GBM which decides to stop the growth of trees prematurely if there is no improvement, XGBoost creates trees until a maximum depth is reached, and then it prunes backward in order to achieve better efficiency.

5. DATASET DESCRIPTION

The dataset contains 10000 observations and 10 features. Each observation represents a system maintenance and all the features together contributes about the system maintenance.

5.1 FEATURES DESCRIPTION:

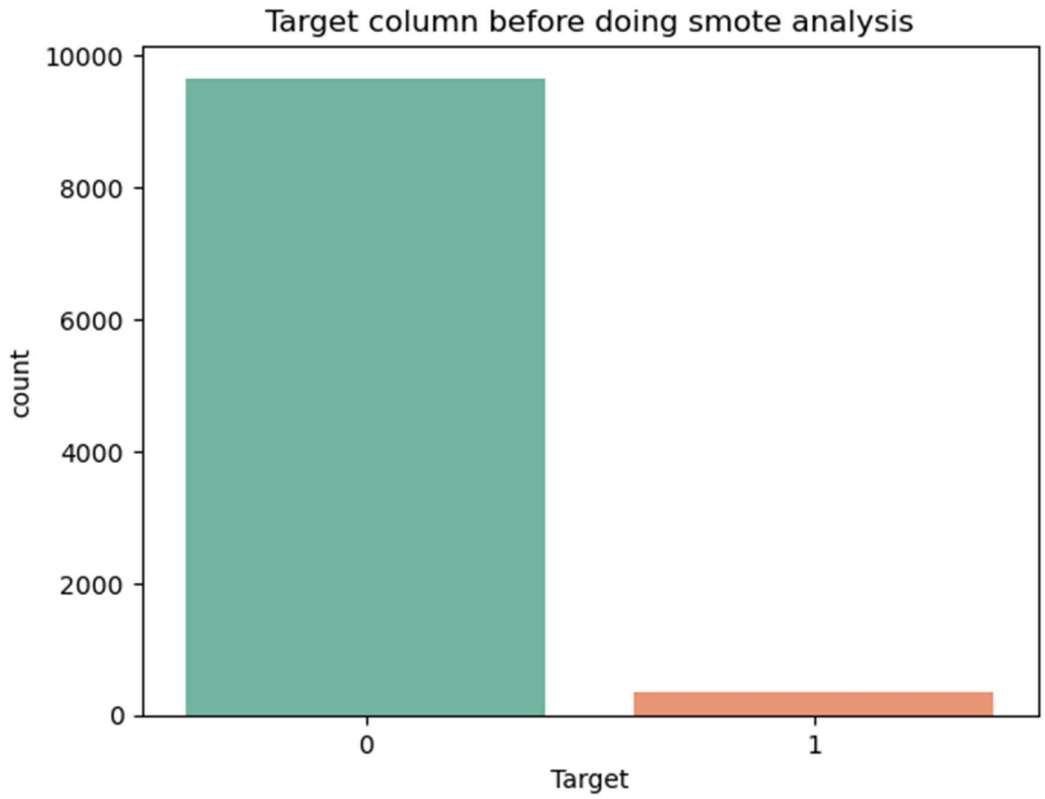
Type	It shows the type of machine whether it is low, medium or high
Air Temperature	It is sensor in which it shows how much air temperature is required
Process Temperature	It is sensor in which it shows the temperature for being processed
Rotational Speed	Speed of the machine
Torque	The Rotational force applied to the machine components
Tool wear	It tracks the duration of usage of the tool under operational stress
Tor*Toolwear	The cumulative mechanical stress or load the tool has experienced over time
Temp_diff	How much hotter the machine is compared to the surrounding air
power	How much work the machine is doing at any moment
Target	It shows whether the machine is failure or running good

So, in raw dataset there is one categorical useful column, which is a type column and using label encoding I changed into numerical.

5.2. EXPLORATORY DATA ANALYSIS

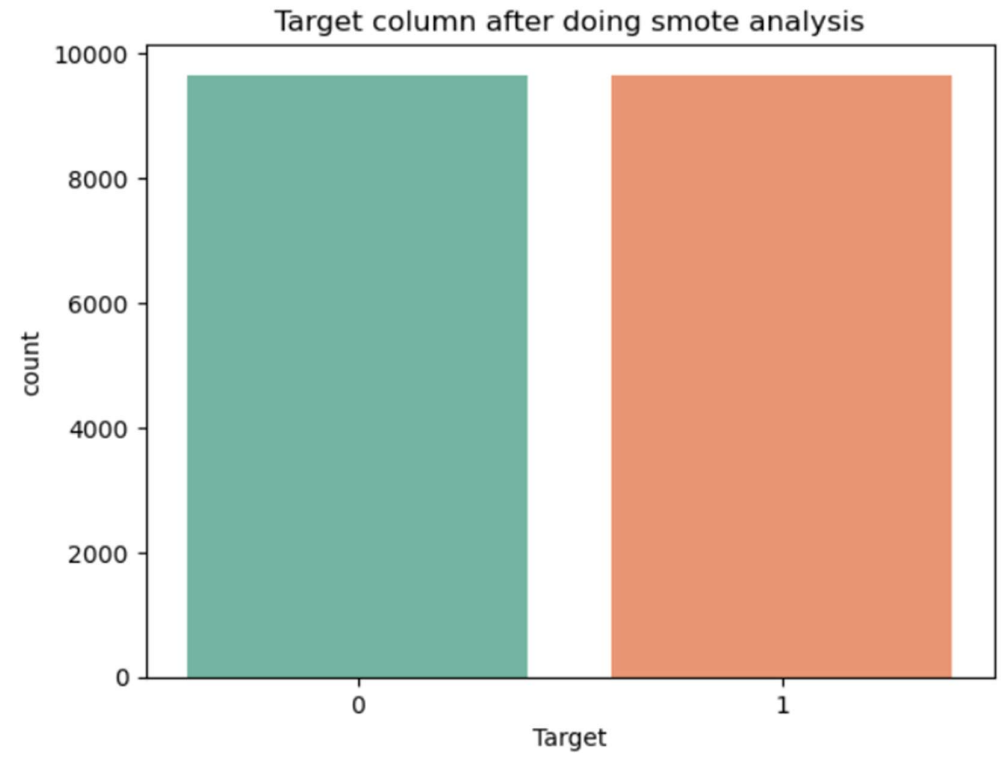
In statistics, exploratory data analysis (EDA) is an approach to analysing data set to summarize their main characteristics, often with visual methods.

- Frequency of target data

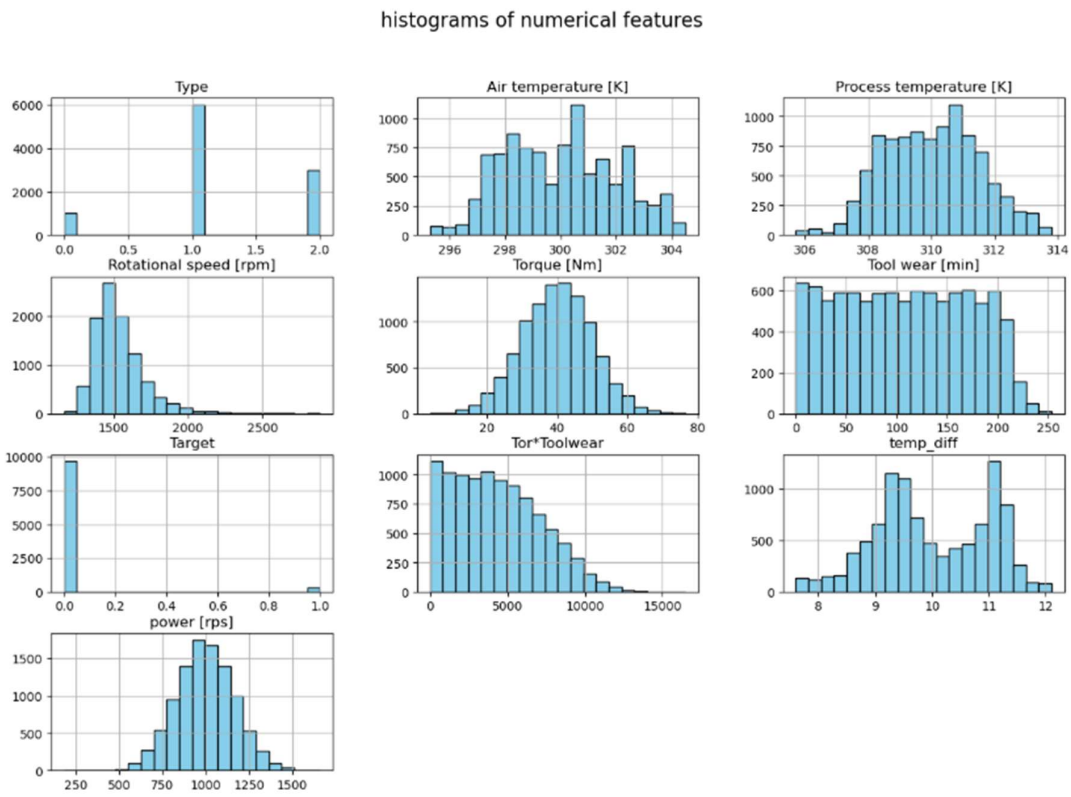


By the above countplot chart we say that the no failure is in 90% of the times and 10% of the times it is in failure and the target column is not in balances. So, to balance the target column, used SMOTE analysis.

- After doing smote analysis some synthetic data was added to ones as before the data is fully biased towards the zero.



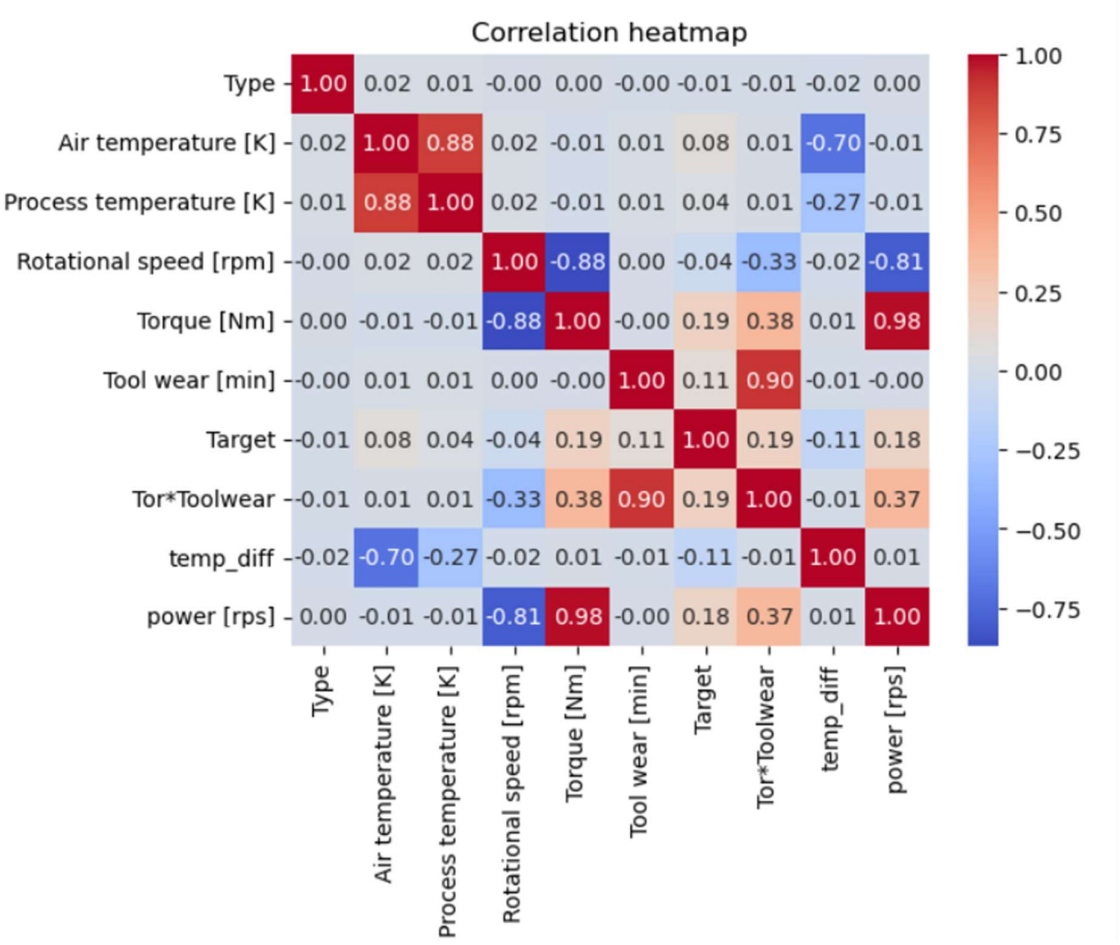
- Histogram for all numerical data after converting all categorical type data into numerical



Histograms are used to visualize the distribution of numerical data by dividing the values into intervals (bins). Each bar in the histogram represents the frequency of data points falling into that range. By analysing the histograms of the features in the dataset, we can gain insights into the nature of the data, detect skewness, and potentially identify outliers.

5.3. CORRELATION MATRIX

This correlation matrix measures the linear correlation between two variables. The resulting value is in between [-1,1] in which -1 means perfect negative correlation and +1 means perfect positive correlation and 0 represents no correlation in which the 0 correlation doesn't affect the output variable.



6. DATA PREPROCESSING

Data Preprocessing is a data mining technique that involves transforming raw data into an understandable format. These transformations are required because the real world data is generally incomplete like lacking attribute values, lacking certain attributes of interest, or containing only aggregate data. Noisy like containing outliers. Therefore, Data preprocessing is a proven method of resolving such issues.

6.1. DATA MANIPULATION

1. A variable named UDI, Product ID and Failure Type was simply dropped because all the features inside it were unique and had no significance on the output.

6.2. FEATURE SCALING

Feature scaling is the process to vacillate the quantities of all features through a common factor. It works on the continuous variable.

It can make the results different a lot for certain algorithms but have little or none impact for other algorithms. In the majority of cases, the input data has units and large range. Left on their own, these algorithms process only the magnitude and ignore the unit.

There are Four Common methods to perform feature scaling

1. Standardization
2. Min-Max Scaling
3. Mean Normalization
4. Robust Scaling

- In my project I used Mean Normalization in which the values are in between [-1,1] with μ=0.

$$x' = \frac{x - \text{mean}(x)}{\max(x) - \min(x)}$$

6.3. LABEL ENCODING

Label encoding is when you assign a value to each of the categories a variable holds so that the resulting model could get a handle on this variable a bit more to perform better in prediction. This is because there are some Machine Learning libraries which won't accept the categorical data as input. We therefore make them numerical. We use this label encoding if source column of our choice contains 1 or 2 or 3 unique variables if more than 1 hot encoding is required.

In the project I had applied it on the Type column in which it has 3 unique variables as 'L', 'M' and 'H' represents low, medium and high are converted into 'L=1', 'M=2' and 'H=3'.

6.4. FEATURE SELECTION

It is the process of finding and selecting the most useful features in a dataset and it is an important step of the machine learning. Unnecessary features decrease the training speed and decrease the model performance on the test set.

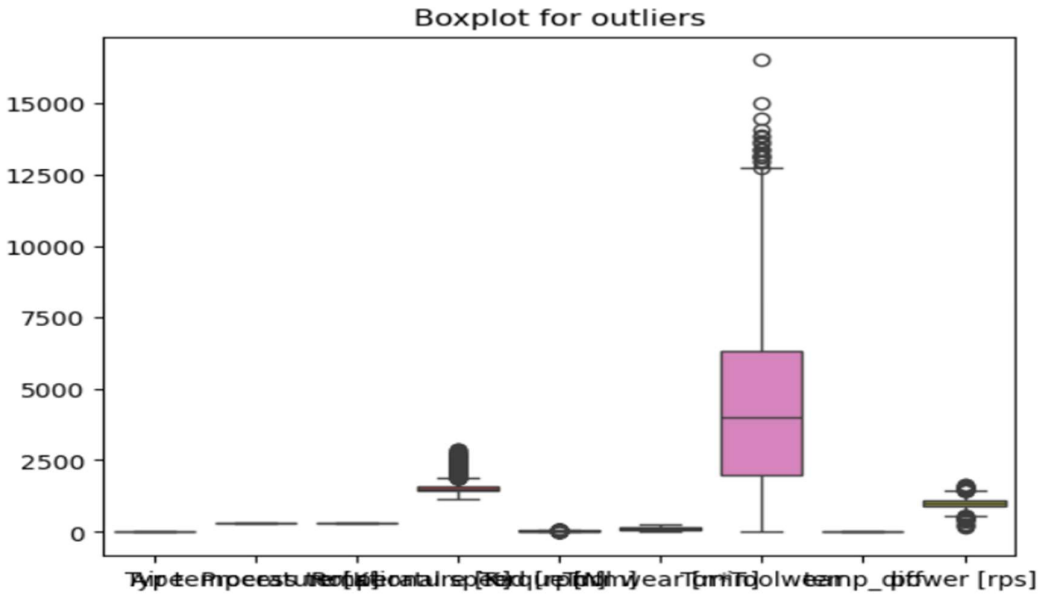
There are many benefits of performing feature selection before modelling, some of them are:

- Reduces Overfitting
- Improves Accuracy
- Reduces Training Time

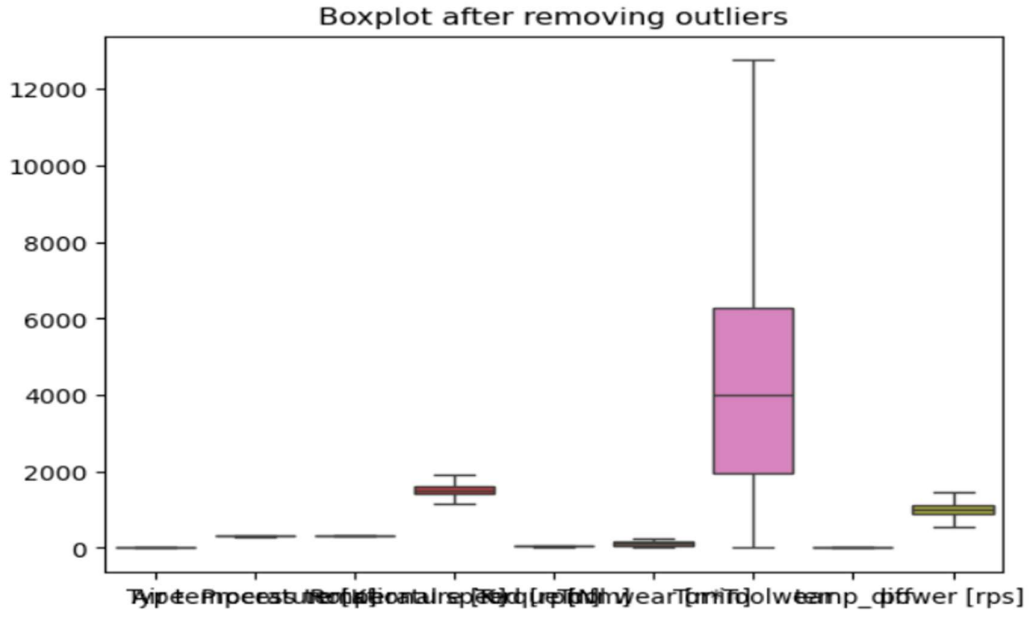
6.5. FEATURE ENGINEERING

- It is a process in which we add some additional data to the dataset so that additional data will shows more effect on test data and increase model performance.
- In project I added three additional columns those are power, tor*toolwear and temperature_difference
- And there are some outliers in the dataset in which may cause the problem to the test set
- So, we Identified those outliers using Box plot and removed those outliers by using Inter Quantile Range (IQR)

- Box plot used to show the outliers in the dataset



- Box plot after applying IQR



7. MODEL EVALUATION

Predictive Modelling works on constructive feedback principle. Therefore, after building a model, it takes feedback from metrics, make improvements and continue until it achieves a desirable accuracy. Evaluation metrics explain the performance of a model. There are various different kinds of evaluation techniques to measure the performance of a model depending upon the type of model and implementation plan of model. The metric used in out project and their description are as follows.

7.1. Train/Test split

Splitting the dataset into two parts, so that the model can be trained and tested on different data. Better estimate of out-of-sample performance, but still a "high variance" estimate. Useful due to its speed, and flexibility. Data can be split into 80% of training and 20% of testing.

7.2. CONFUSION MATRIX

A confusion matrix is a n*n matrix, where n is the number of classes being predicted. For the problem in hand, we have n=2 because it is a binary classification and hence, we get a 2*2 matrix

	POSITIVE	NEGATIVE
POSITIVE	TRUE POSTIVE(TP)	FALSE POSITIVE(FP)
NEGATIVE	FALSE NEGATIVE(FN)	TRUE NEGATIVE(TN)

Some important evaluation measures for a confusion matrix are as follows:

ACCURACY:
The proportion of the total number of predictions that were correct.

$$Accuracy= (TP + FP)/ (TP + FP + TN + FN)$$

Precision:

The proportion of positive cases that were correctly identified.

$$Precision=(TP)/(TP+FP)$$

Recall:

The proportion of actual positive cases which are correctly identified.

Sensitivity=(TP)/(TP+FN)

Specificity

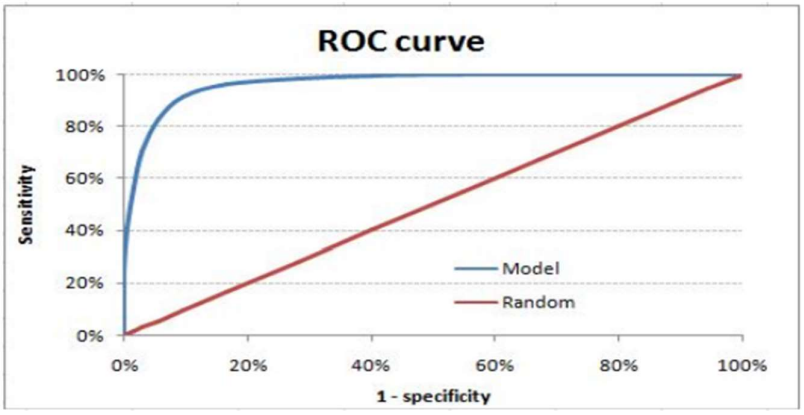
The proportion of actual negative cases which are correctly identified.

Specificity=(TN)/(TN+FP)

7.3. AREA UNDER THE ROC CURVE(AUC-ROC)

If we look at the confusion matrix above, we observe that for a probabilistic model, we get different value for each metric. Hence, for each sensitivity, we get a different specificity.

The ROC curve is the plot between sensitivity and (1- specificity). (1- specificity) is also known as false positive rate and sensitivity is also known as True Positive rate.



To bring this curve down to a single number, we find the area under this curve (AUC). Note that the area of entire square is always 1.

Following are the few thumb rules:

- 0.90-1 = excellent
- 0.80-0.90 = good
- 0.70-0.80 = fair
- 0.60-0.70 = poor
- 0.50-0.6 = fail

8. RESULTS AND DISCUSSION

8.1 Comparison of various models:

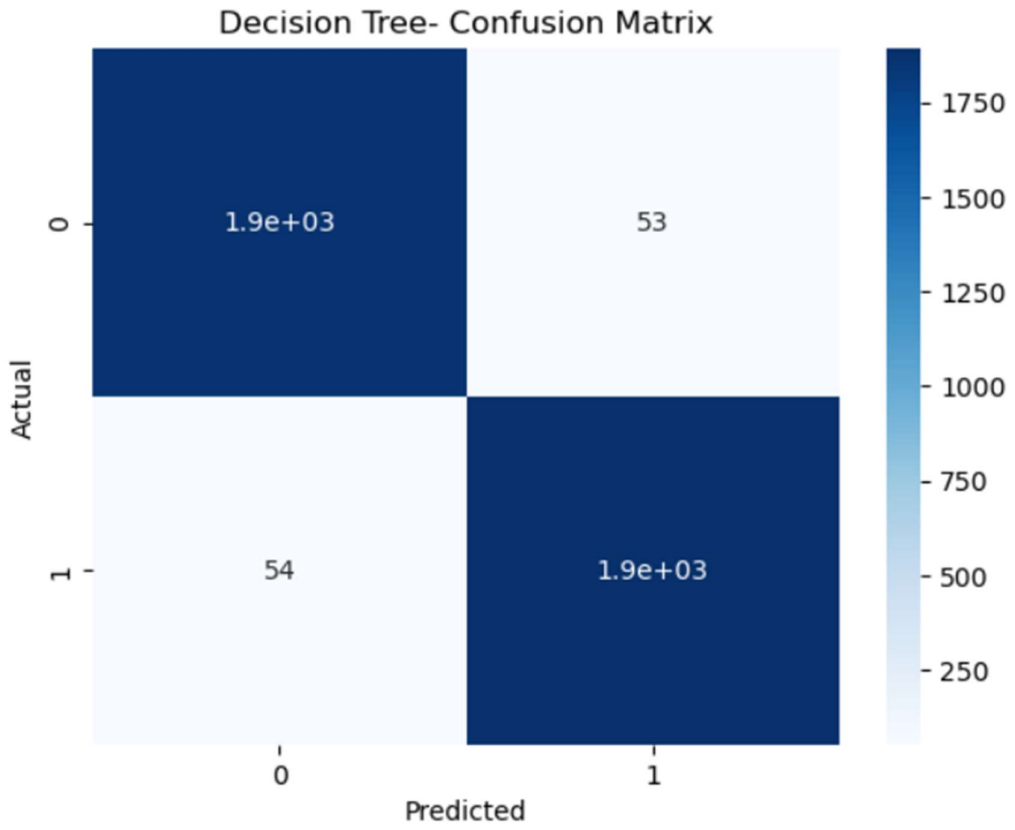
MODEL	Decision Tree	Random Forest	XGBoost
Precision for 0	0.97	0.99	0.99
Precision for 1	0.97	0.98	0.98
Recall for 0	0.97	0.98	0.98
Recall for 1	0.97	0.99	0.99
F1-score for 0	0.97	0.98	0.99
F1-score for 1	0.97	0.98	0.99
Overall Accuracy	0.97	0.98	0.99

So, from above table we can conclude that xgboost has a good accuracy rate with 0.99 so we test the model by saving that xgboost model.

9.SCREEN SHOT OF RESULTS

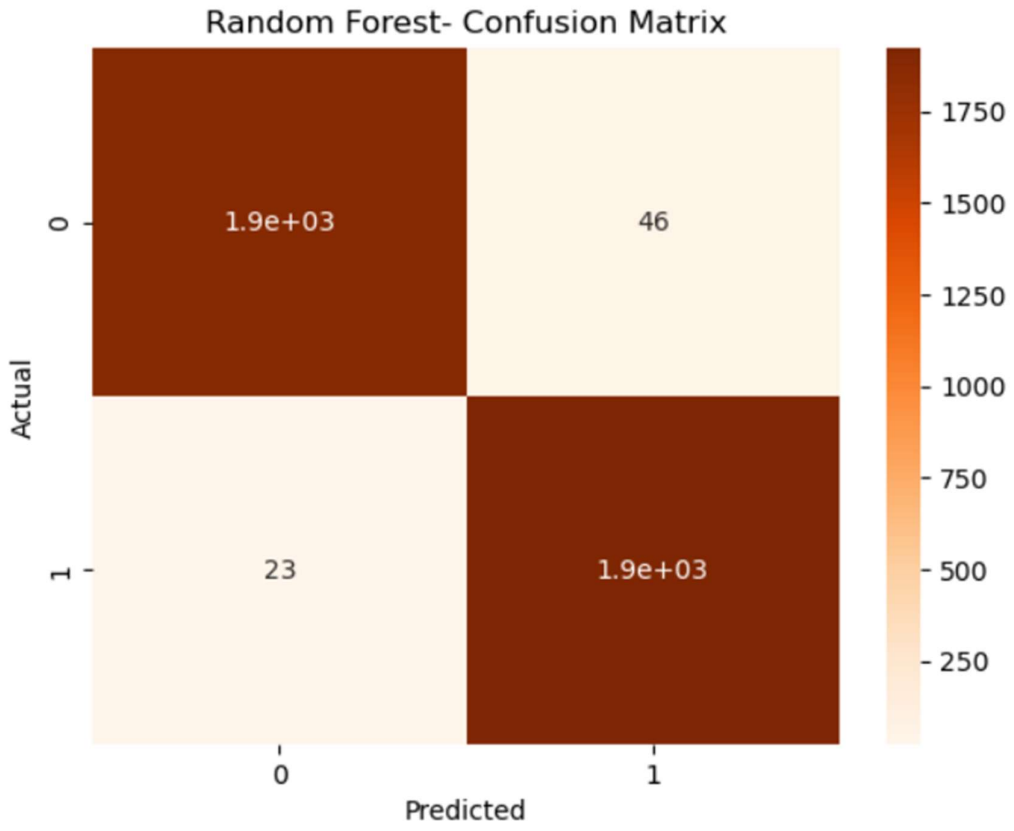
Decision Tree:

Confusion Matrix of a Decision Tree Model



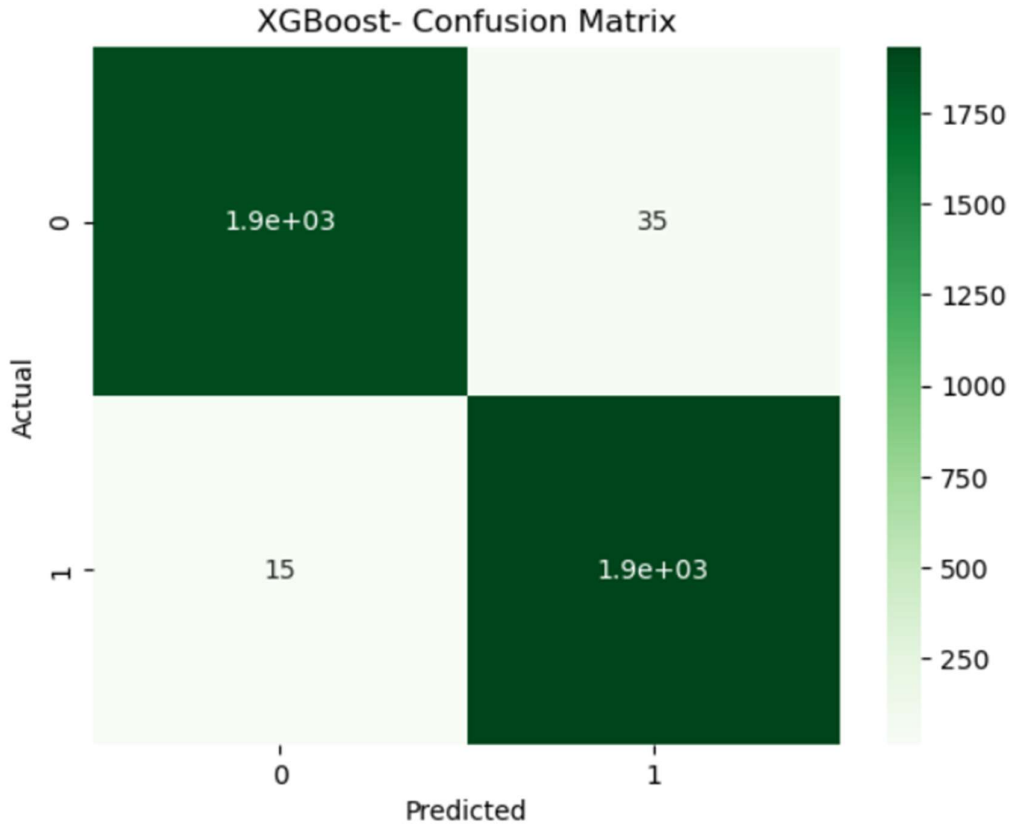
Random Forest:

The confusion matrix for a random forest model

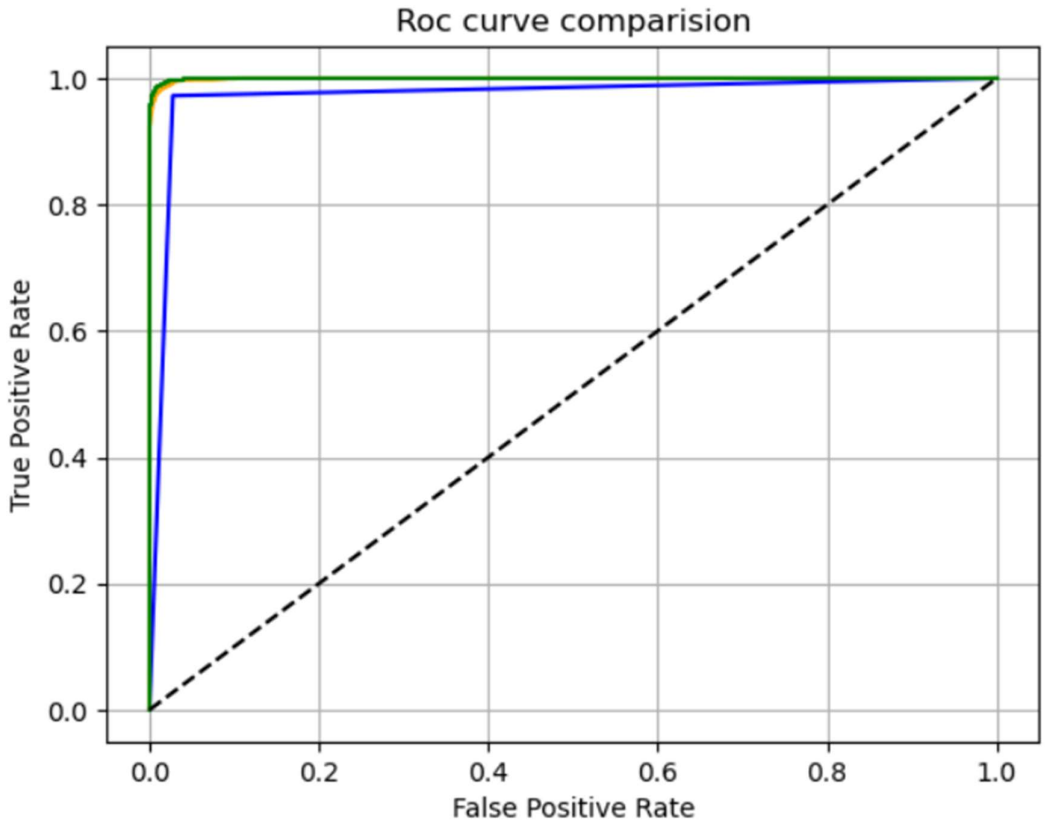


XG-Boost:

Confusion Matrix of a XG-Boost Model



AUC-ROC Curve Comparision screenshot for three models:



10. CONCLUSION

So, from above all comparsions we can conclude that the XGoost model is the best model for testing the project. And no special scaling made as the accuracy rates for above almost 1.

REFERENCES:

● https://www.analyticsvidhya.com/blog/2016/03/complete-guide-parameter-tuning-xgboost-with-codes-python/

● https://www.analyticsvidhya.com/blog/2015/11/beginners-guide-on-logistic-regression-in-r/

● https://www.danielsoper.com/statcalc/calculator.aspx?id=8

● https://sebastianraschka.com/Articles/2014_about_feature_scaling.html#about-standardization

● https://www.statisticshowto.datasciencecentral.com/probability-and-statistics/chi-square/

● http://www.insightsbot.com/blog/2AeuRL/chi-square-feature-selection-in-python

● https://towardsdatascience.com/the-dummys-guide-to-creating-dummy-variables-f21faddb1d4 0

● https://hub.packtpub.com/4-ways-implement-feature-selection-python-machine-learning/

● https://cmdlinetips.com/2018/02/how-to-get-frequency-counts-of-a-column-in-pandas-datafra me/

● https://www.youtube.com/watch?v=V0u6bxQOUJ8