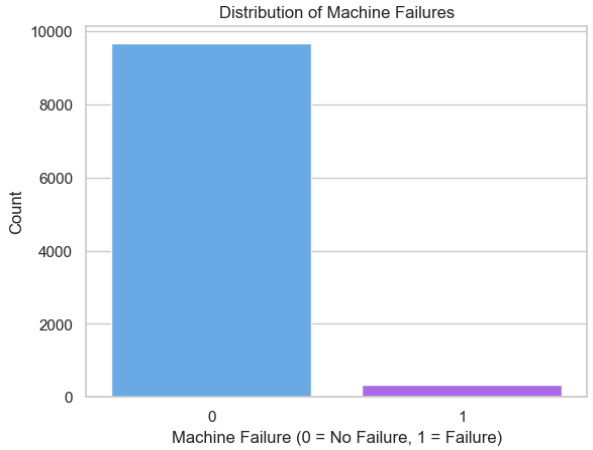
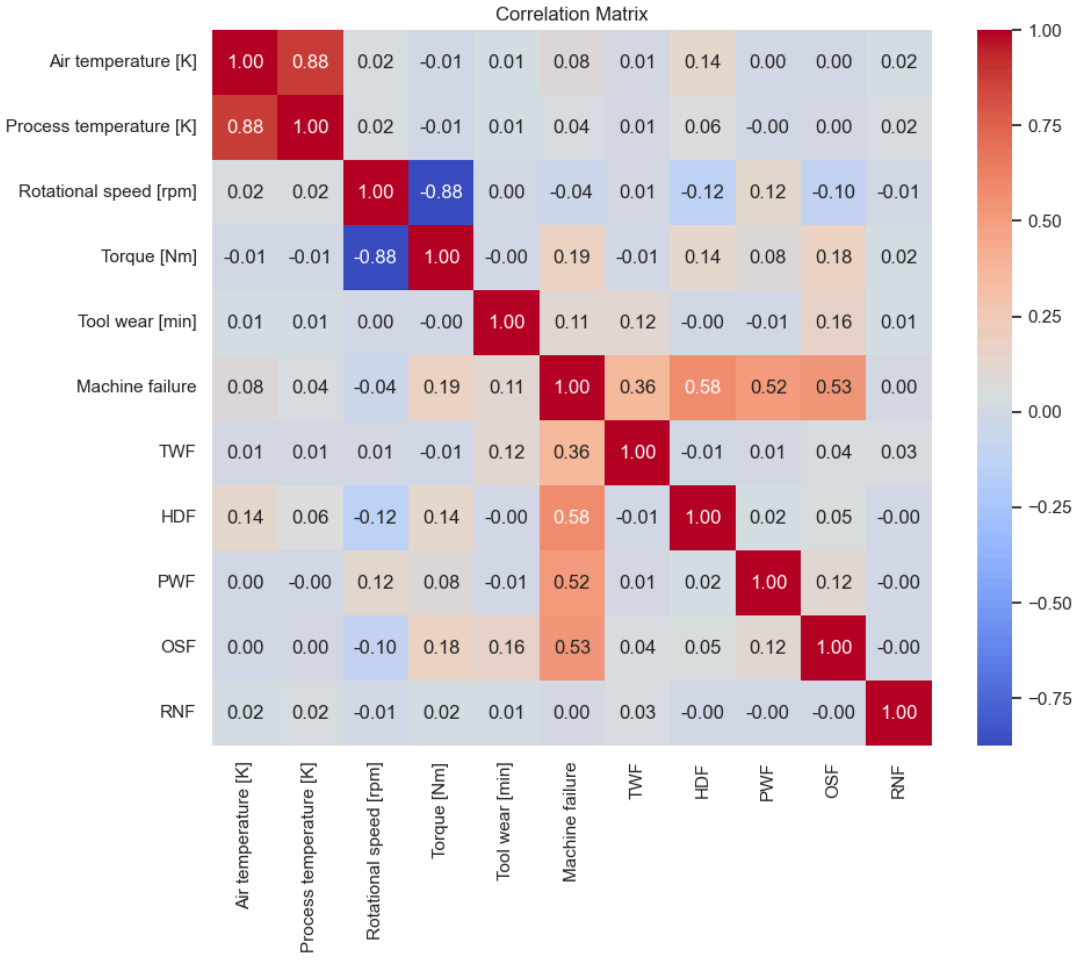
The dataset I have chosen is the “AI4I 2020 Predictive Maintenance Dataset” from UCI ML
https://archive.ics.uci.edu/dataset/601/ai4i+2020+predictive+maintenance+dataset
There are 6 features - Product ID, Air temperature, Process temperature, Rotational speed, Torque and Tool wear.
There are 5 target variables. Four of them are intermediate (like flags) out of which if any one is 1 then there is a machine failure (final target)

Exploratory Data Analysis

I am using seaborn and matplotlib for the EDA.
I will be looking at the distribution of the classes in the dataset, since there are only two (0 = no failure - 9661 examples and 1 = failure - 339 examples), we will see if one class is dominating which is - therefore our model may become biased.
I will use SMOTE to overcome class imbalance in the model training.



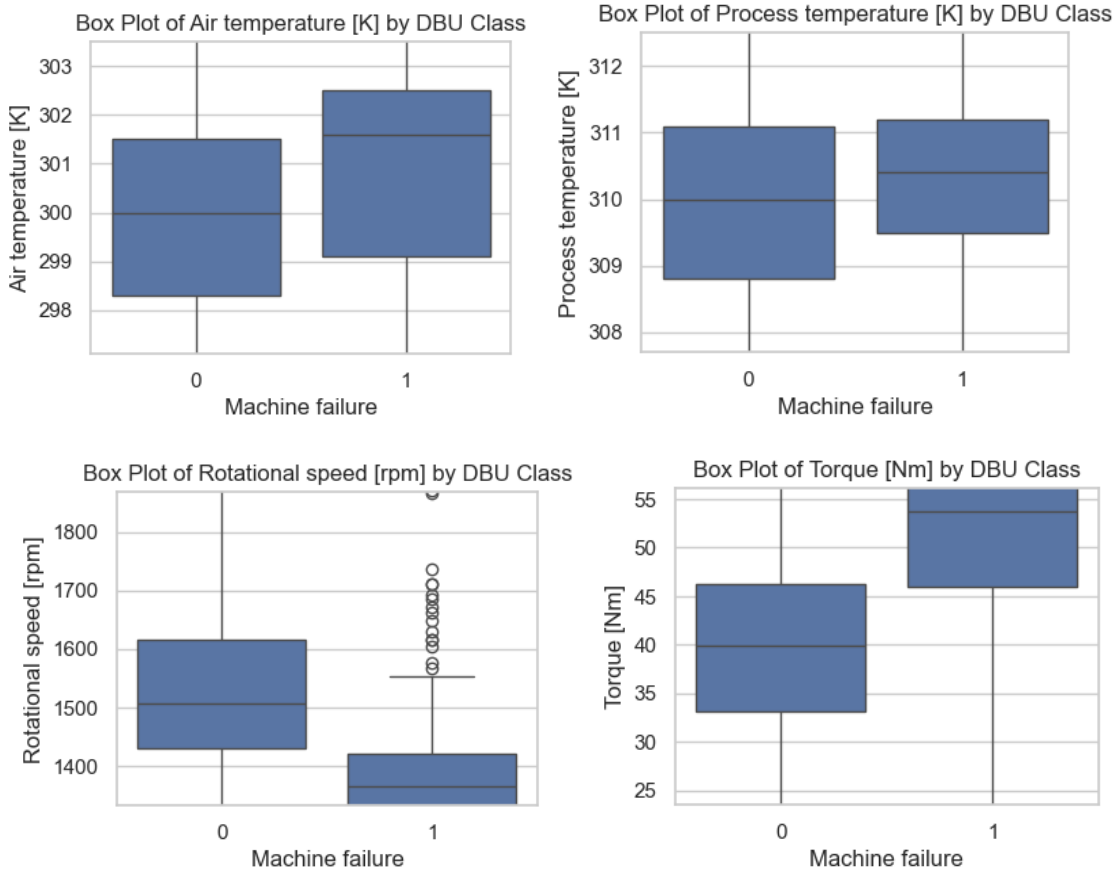


Observing this, we can form some additional hypotheses about the data.

1. Air temperature and Process temperature are very correlated since this was a synthetic dataset and the process temperature values were generated by using random walk by adding +10K to air temperature.
2. Rotational speed and torque have a strong negative correlation, meaning the slower the rotational speed, the higher will be the torque generated since power supplied is the same.

Now I will choose one out of each, whichever will give me the best split in the target class, which ensures efficiency of my model.

I will plot a box plot of these with respect to the features to see which has the best seperation.



Between the air and process temperature, I will choose to keep the air because the median of two classes is a little bit far apart, and between the rotational speed and torque, i will choose the torque, because the former has some outliers which might reduce the size of my dataset if I choose to remove.

The tool wear box plot is similar to the air-temperature plot therefore I will omit it. In the end I have kept 4 features

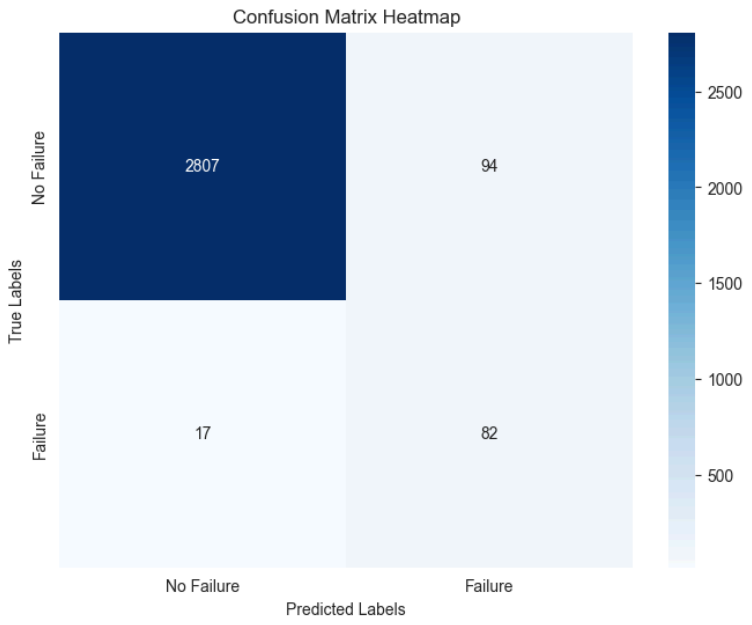
Model Selection

Selected Random forest as it is giving better results than other models.

- `max_depth=3`: Limits the depth of the tree to prevent overfitting.
- `class_weight='balanced'`: Assigns weights inversely proportional to class frequencies to handle class imbalance.

Model	Accuracy	F1 Score	Recall
Random Forest	0.96	0.59	0.82
Logistic Regression	0.83	0.24	0.79

Results



The model performed well on the dominant class and misclassified few of the failures, this was an improvement on other models.

Conclusion

The model gives good results and identifies key features, but its performance is limited due to class imbalance and lack of complexity. Further tuning and ensemble methods can improve outcomes.