INTRODUCTION

In part 2 of this project, classification algorithms are used to learn how to classify single faults. Eight machine learning algorithms are compared in terms of performance by evaluating metrics like precision, recall, accuracy, f1-scores and area under the (receiver operating) curve. The empirical results are shown in the tables below. Each model is fitted with and without hyperparameter tuning.

RESULTS

GRU

Classification Metric	Estimation Type	Evaluation Metric				
		Precision	Recall	Accuracy	F1-score	ROC-AUC
XG Boost	*Without Hyperparameter Tuning*	1.67%	9.09%	86.36%	2.82%	66.00%
	With Hyperparameter Tuning	3.45%	27.27 %	81.82%	6.12%	55.00%
Logistic regression	*Without Hyperparameter Tuning*	2.76%	63.64 %	50.40%	5.28%	58.00%
	With Hyperparameter Tuning	2.79%	63.64 %	50.99%	5.34%	57.00%
Adaboost	*Without Hyperparameter Tuning*	2.96%	36.36 %	72.73%	5.48%	63.00%
	With Hyperparameter Tuning	3.05%	45.45 %	67.39%	5.71%	57.00%
Gradient Boosting	*Without Hyperparameter Tuning*	0.00%	0.00%	85.57%	0.00%	62.00%
	With Hyperparameter Tuning	2.78%	9.09%	91.11%	4.26%	51.00%
Decision Trees	*Without Hyperparameter Tuning*	1.59%	9.09%	85.77%	2.70%	44.00%
	With Hyperparameter Tuning	2.29%	27.27 %	73.12%	4.23%	51.00%
SVM	*Without Hyperparameter Tuning*	2.33%	45.45 %	51.19%	3.89%	
	With Hyperparameter Tuning	1.37%	18.18 %	69.76%	2.55%	
Random Forests	*Without Hyperparameter Tuning*	0.00%	0.00%	91.50%	0.00%	54.00%
	With Hyperparameter Tuning	0.00%	0.00%	91.90%	0.00%	47.00%
KNN	*Without Hyperparameter Tuning*	2.13%	18.18 %	80.04%	3.81%	49.00%
	With Hyperparameter Tuning	16.95%	9.09%	86.56%	2.86%	49.00%

For anomalies classified using the best (after hyperparameter tuning) GRU model, the highest accuracy of 91.8% was achieved using random forest model with hyperparameter tuning. The precision, recall, and f1-scores are 0 since the data was very imbalanced (few anomalies overall). This model had an roc-auc score of 0.47 which is below the 0.5 cut off point. A better model is random forests without hyperparameter tuning which achieved an accuracy of 91.5% and an roc-auc score of 0.54.

LTSM

Classification Metric	Estimation Type	Evaluation Metric				
		Precision	Recall	Accuracy	F1-score	ROC-AUC
XG Boost	*Without Hyperparameter Tuning*	4.98%	50.00 %	60.28%	9.05%	61.00%
	With Hyperparameter Tuning	4.81%	25.00 %	77.47%	8.06%	52.00%
Logistic regression	*Without Hyperparameter Tuning*	2.53%	30.00 %	51.58%	4.67%	46.00%
	With Hyperparameter Tuning	2.42%	30.00 %	4.94%	4.48%	40.00%
Adaboost	*Without Hyperparameter Tuning*	3.67%	20.00 %	76.09%	6.20%	55.00%
	With Hyperparameter Tuning	5.03%	50.00 %	60.67%	9.13%	56.00%
Gradient Boosting	*Without Hyperparameter Tuning*	2.74%	10.00 %	82.41%	4.30%	52.00%
	With Hyperparameter Tuning	8.00%	10.00 %	91.87%	8.89%	53.00%
Decision Trees	*Without Hyperparameter Tuning*	5.36%	15.00 %	86.17%	7.89%	53.00%
	With Hyperparameter Tuning	5.50%	55.00 %	60.87%	10.00%	58.00%
SVM	*Without Hyperparameter Tuning*	3.80%	35.00 %	62.45%	6.86%	
	With Hyperparameter Tuning	3.73%	30.00 %	66.60%	6.63%	
Random Forests	*Without Hyperparameter Tuning*	8.00%	10.00 %	91.90%	8.89%	56.00%
	With Hyperparameter Tuning	7.41%	10.00 %	91.50%	8.51%	52.00%
KNN	*Without Hyperparameter Tuning*	3.92%	20.00 %	77.47%	6.56%	53.00%
	With Hyperparameter Tuning	4.00%	10.00 %	88.74%	6.56%	51.00%

For anomalies generated using the best LTSM score, the highest accuracy was again achieved using random forests, this time without hyperparameter tuning. This model had an accuracy of 91.9% and roc-auc of 56%. Other well-performing models are random forests with hyperparameter tuning (accuracy=91.5%, roc-auc = 52%) and Gradient Boosting without hyperparameter tuning (accuracy=91.87%, roc-auc = 53%). The precision, recall, and f1-scores are low since the data was very imbalanced (few anomalies overall).

RNN

Classification Metric	Estimation Type	Evaluation Metric				
		Precision	Recall	Accuracy	F1-score	ROC-AUC

XG Boost	*Without Hyperparameter Tuning*	1.53%	33.33 %	83.40%	16.00%	59.00%
	With Hyperparameter Tuning	5.00%	16.67 %	81.03%	7.69%	50.00%
Logistic regression	*Without Hyperparameter Tuning*	5.94%	70.83 %	45.45%	10.97%	54.00%
	With Hyperparameter Tuning	5.94%	70.83 %	45.45%	10.97%	58.00%
Adaboost	*Without Hyperparameter Tuning*	5.13%	25.00 %	74.51%	8.51%	56.00%
	With Hyperparameter Tuning	7.97%	45.83 %	7.23%	13.58%	60.00%
Gradient Boosting	*Without Hyperparameter Tuning*	4.17%	12.50 %	82.21%	6.25%	59.00%
	With Hyperparameter Tuning	6.90%	8.33%	90.32%	7.55%	51.00%
Decision Trees	*Without Hyperparameter Tuning*	6.67%	16.67 %	84.98%	9.52%	55.00%
	With Hyperparameter Tuning	5.85%	50.00 %	59.49%	10.48%	55.00%
SVM	*Without Hyperparameter Tuning*	4.97%	37.50 %	63.04%	8.78%	
	With Hyperparameter Tuning	4.00%	29.17 %	63.44%	7.04%	
Random Forests	*Without Hyperparameter Tuning*	0.00%	0.00%	88.74%	0.00%	61.00%
	With Hyperparameter Tuning	0.00%	0.00%	88.74%	0.00%	53.00%
KNN	*Without Hyperparameter Tuning*	7.37%	29.17 %	79.25%	11.76%	56.00%
	With Hyperparameter Tuning	8.89%	16.67 %	8.79%	11.59%	54.00%

For anomalies generated using RNN, the highest accuracy was achieved by the Gradient Boosting algorithm with hyperparameter tuning accuracy=90.32%, roc-auc = 51%). The precision, recall, and f1-scores are low since the data was very imbalanced (few anomalies overall).

ANN

Classification Metric	Estimation Type	Evaluation Metric				
		Precision	**Recall**	**Accuracy**	**F1-score**	**ROC-AUC**
XG Boost	*Without Hyperparameter Tuning*	4.76%	64.71 %	55.34%	8.87%	63.00%
	With Hyperparameter Tuning	2.47%	11.76 %	81.42%	4.08%	48.00%
Logistic regression	*Without Hyperparameter Tuning*	4.04%	64.71 %	47.23%	7.61%	57.00%
	With Hyperparameter Tuning	3.93%	64.71	45.65%	7.41%	55.00%

Adaboost	*Without Hyperparameter Tuning*	4.11%	35.29 %	70.16%	7.36%	63.00%
	With Hyperparameter Tuning	5.82%	64.71 %	63.64%	10.68%	64.00%
Gradient Boosting	*Without Hyperparameter Tuning*	3.85%	17.65 %	82.41%	6.32%	59.00%
	With Hyperparameter Tuning	2.38%	5.88%	88.74%	3.39%	49.00%
Decision Trees	*Without Hyperparameter Tuning*	1.69%	5.88%	85.38%	2.63%	46.00%
	With Hyperparameter Tuning	4.64%	64.71 %	54.15%	8.66%	59.00%
SVM	*Without Hyperparameter Tuning*	3.89%	64.71 %	45.06%	7.33%	
	With Hyperparameter Tuning	4.19%	41.18 %	66.40%	7.61%	
Random Forests	*Without Hyperparameter Tuning*	2.94%	5.88%	90.32%	3.92%	54.00%
	With Hyperparameter Tuning	2.78%	5.88%	89.92%	3.77%	49.00%
KNN	*Without Hyperparameter Tuning*	2.46%	17.65 %	73.72%	4.32%	43.00%
	With Hyperparameter Tuning	1.41%	5.88%	83.00%	2.27%	46.00%

For anomalies generated using ANN, the best model was random forests without hyperparameter tuning accuracy=90.32%, roc-auc = 54%). The precision, recall, and f1-scores are low since the data was very imbalanced (few anomalies overall).

The best anomaly classification model appears to be random forests followed by gradient boosting.