AER 850 Project 1

Nazem Eylji

500957556

Table of Contents

List of Figures

List of Tables

Introduction

This project aims to help the student form a deep understanding of machine learning pipelines and how to be able to build one successfully. The goal of this specific task is to predict the specific maintenance step required for disassembling an inverter in the FlightMax Fill Motion Simulator using part coordinates. The dataset includes X, Y, and Z coordinates as features, with the disassembly step as the target variable. The project involves data processing, feature visualization, correlation analysis, and model development, focusing on the implementation of classification algorithms like Random Forest, Support Vector Machines (SVM), and K-Nearest Neighbors (KNN). Hyperparameter optimization is performed using Grid Search and Randomized Search, and model performance is evaluated through accuracy, precision, and F1 score. Additionally, model stacking is applied to assess its impact on performance. The goal is to determine the best-performing model, which will then be tested on unseen coordinates to validate its predictive capability in AR-guided aerospace maintenance applications.

Data Visualization

As shown in figures 1-3, Pandas was utilized to visualize the distribution of x, y, and z coordinates using histograms. To enhance the visualization better, a kernel density estimate curve was applied to each histogram giving a better representation of the data distribution.

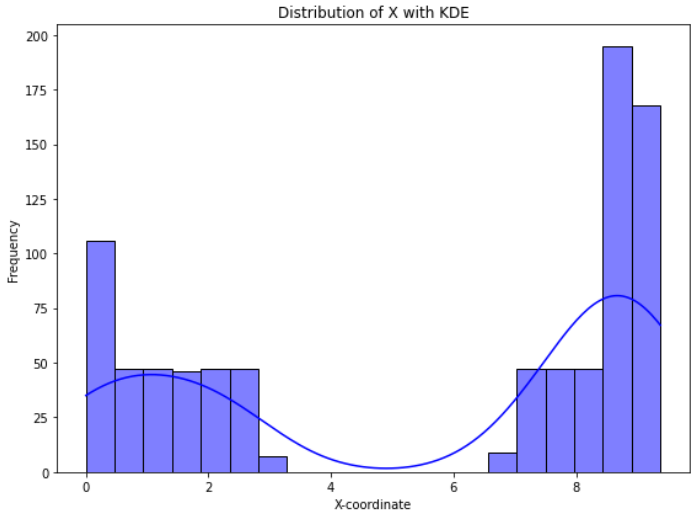

Figure 1 - Distribution of x

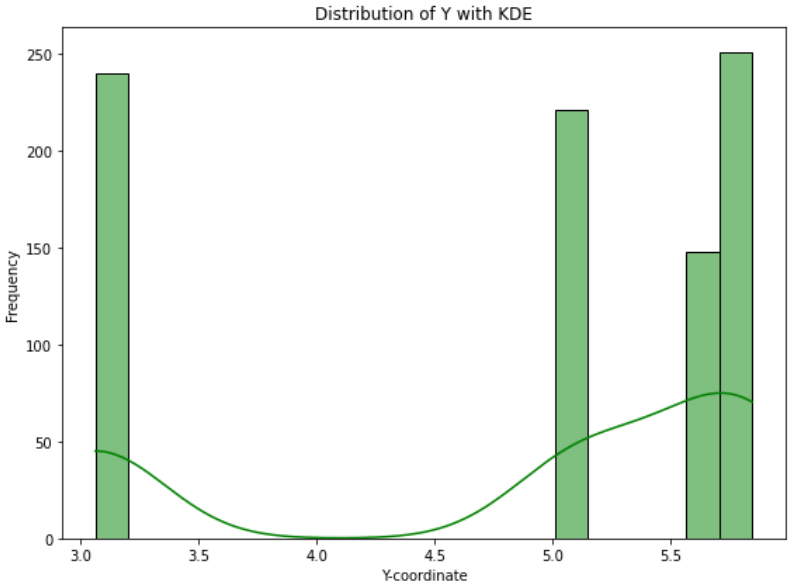

Figure 2 - Distribution of y

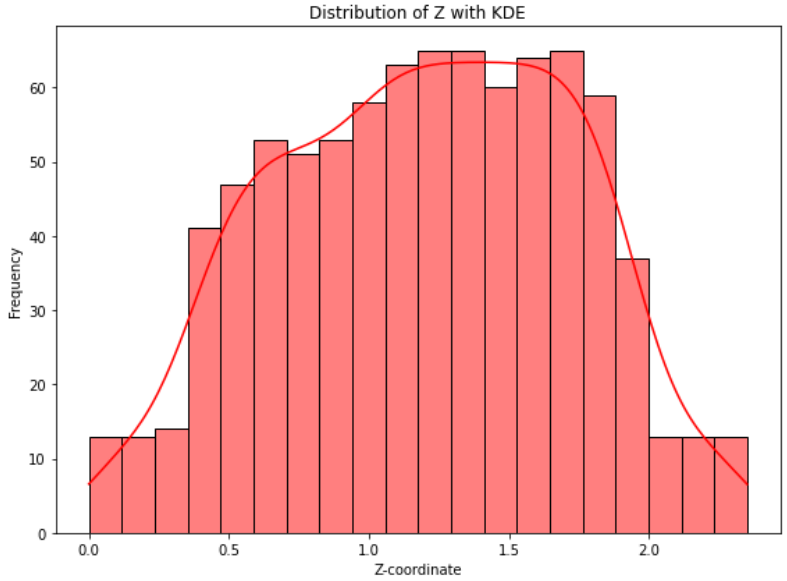

Figure 3 - Distribution of Z

Correlation Analysis

Upon examining the correlation matrix as shown in Figure 4, there were no features that showed signs of a strong linear relationship which required no change in our data.

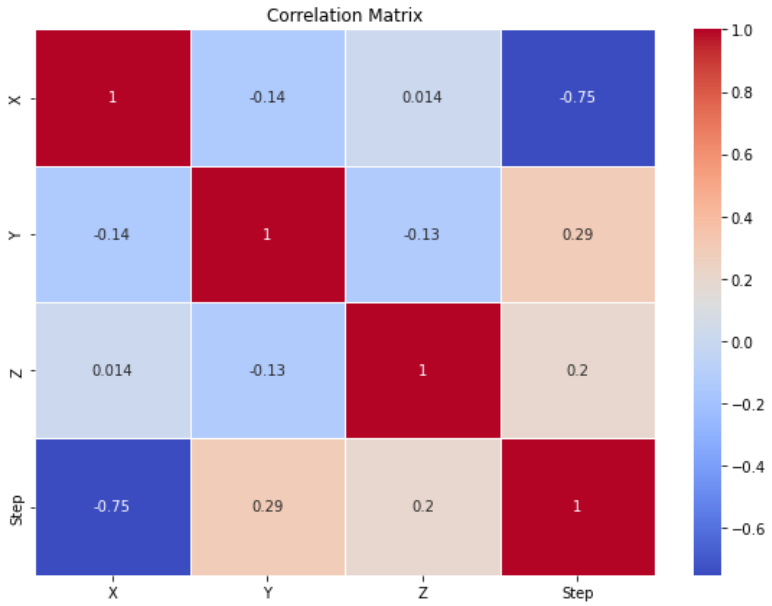

Figure 4 - Correlation Matrix

Classification Model Development/Engineering

Random Forest Classifier

Random Forest uses an ensemble of decision trees to improve classification accuracy. It handles complex, non-linear relationships between features and target variables, making it suitable for predicting maintenance steps based on 3D coordinates.

Support Vector Machine (SVM)

SVM finds the optimal hyperplane to separate classes with a maximum margin. Although not specifically designed for multiclass problems, it was included for comparison with other models.

K-Nearest Neighbors (KNN)

KNN classifies points based on their nearest neighbors in feature space. It makes no assumptions about data distribution and serves as a simple baseline model in this project.

RandomizedSearch Random Forest

RandomizedSearchCV optimizes Random Forest by sampling hyperparameters from defined ranges. This approach speeds up the tuning process while maintaining good model performance.

Model Performance Analysis

Model Performance Analysis, we evaluated the performance of each classification model using key metrics: Accuracy, Precision, and F1 Score. For each model (Random Forest, SVM, KNN, and RandomizedSearch Random Forest), we computed these metrics on the test data to assess their effectiveness in predicting the correct maintenance step. Additionally, we visualized the results using confusion matrices to observe how well each model performed across all classes. The table below highlights the performance of each model.

Table 1 - Results of each classification model

Model	Accuracy	Precision	F1 Score
RandomForest	0.97	0.97	0.97
SVM	0.98	0.98	0.98
KNN	0.98	0.98	0.98
RandomizedSearch_RF	0.97	0.97	0.97

From the results, SVM and KNN are tied for the best performance across all key metrics (accuracy, precision, and F1 score). Both models performed consistently well, with SVM offering a slight edge regarding versatility and applicability to high-dimensional data.

Thus, if we were to choose the best model, SVM would likely be the top choice due to its overall strong performance across all metrics and its robustness in separating classes using hyperplanes. However, KNN is also a strong contender, particularly when data relationships are simpler and computational efficiency is a concern.

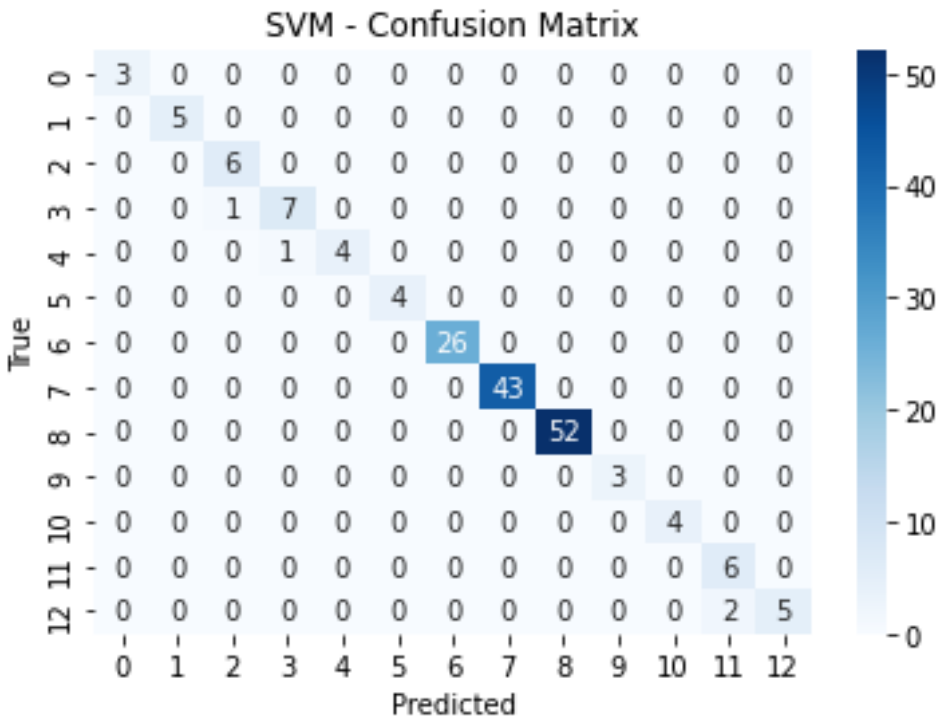

Figure 5 - SVM Confusion Matrix

Stacked Model Construction

Table 2 - Stacked model results in comparison with previous models

Model	Accuracy	Precision	F1 Score
RandomForest	0.97	0.97	0.97
SVM	0.98	0.98	0.98
KNN	0.98	0.98	0.98
RandomizedSearch_RF	0.97	0.97	0.97
Stacked_Model	0.99	0.99	0.99

As shown in in Table 2, he Stacked Model outperformed all other models, with an Accuracy of 0.99, a Precision of 0.99, and an F1 Score of 0.99. This suggests that combining multiple models improved overall performance, making the stacked model the best-performing one in this comparison

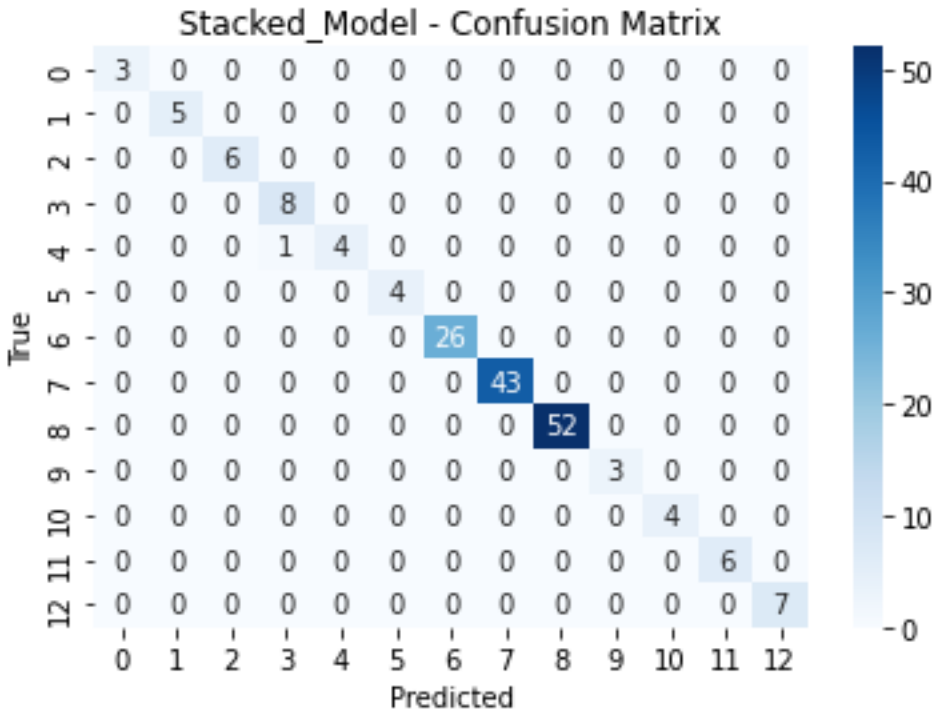

Figure 6 - Stacked Model Confusion Matrix

Model Evaluation

The results for each of the given testing coordinates were as follows:

- Coordinates [9.375 3.0625 1.51] -> Predicted Maintenance Step: 5
- Coordinates [6.995 5.125 0.3875] -> Predicted Maintenance Step: 8
- Coordinates [0. 3.0625 1.93] -> Predicted Maintenance Step: 13
- Coordinates [9.4 3. 1.8] -> Predicted Maintenance Step: 6
- Coordinates [9.4 3. 1.3] -> Predicted Maintenance Step: 4

Conclusion

In this project, various machine learning classification models were developed and evaluated to predict maintenance steps based on 3D coordinates. We implemented and fine-tuned several models, including RandomForest, SVM, KNN, and a Stacked Model. Each model's performance was assessed using key metrics such as Accuracy, Precision, and F1 Score.

The SVM and KNN models performed similarly, achieving high accuracy and balanced precision and F1 scores, making them strong individual classifiers. However, the Stacked Model, which combined the outputs of multiple models, outperformed all individual models, achieving the highest accuracy of 0.99 and providing the most reliable predictions overall. This demonstrates

the effectiveness of model stacking in improving prediction accuracy by leveraging the strengths of multiple models.

Ultimately, the Stacked Model proved to be the best-performing classifier in this project, highlighting the benefits of combining multiple algorithms for more accurate and robust predictions. This result suggests that a hybrid approach to machine learning, where different models are combined, can yield better outcomes in complex classification tasks.

Link to Github Repository

GitHub - NazemEylji/AER850-Project-1

References

[1] R. Faieghi, "L4 - Common Machine Learning Models - AER850 - Intro to Machine Learning - F2024," *Torontomu.ca*, 2024. https://courses.torontomu.ca/d2l/le/content/948065/viewContent/6032115/View

[2] R. Faieghi, "L5 - Classification - AER850 - Intro to Machine Learning - F2024," *Torontomu.ca*, 2024. https://courses.torontomu.ca/d2l/le/content/948065/viewContent/6037093/View