CSCE 5214-004 SOFTWARE DEVELOPMENT FOR AI

Assignment 2 - Develop Unsupervised Machine Learning Models

Eeshwar Vannemreddy

11596826

Task 1: Data Preparation

This data preparation step includes CAN Bus log data loading, extraction, and cleaning. Making a well-organized dataset appropriate for model building afterwards. The data is now prepared for further analysis, which can involve anomaly identification using machine learning methods like the Isolation Forest or clustering using K-Means, by choosing relevant parameters and limiting possible attacks.

Data Extraction and Transformation

There are several important steps in the data preparation process. Initially, three separate CAN Bus log files are used to load the data: these log files each record vehicle data under various operating conditions. The labels "df0," "df1, and "df2" refer to these files.

Data extraction is a crucial step after data loading. This means analyzing the log entries to extract relevant data, including the "PID" (Parameter ID) and its message. The columns "Time_Stamp" and "Bus_id" are left out of the analysis since they don't add anything to the findings.

Choosing the Appropriate PIDs

Finding and choosing relevant PIDs is an important component of this data preparation. The dataset is streamlined in this data processing stage so that it may focus on these two factors.

One-hot encoding

One-hot encoding is used on the "PID" column to make data representation and analysis easier. The categorical PID values are transformed using this method into binary columns for "RPM" and "Speed." The encoded columns that are produced simplify the dataset and prepare it for additional data processing.

Column Renaming

Renaming the columns to be clearer and more concise is the last step in the data transformation process. The columns are now called "RPM" and "Speed," appropriately designating the two main vehicle parameters that are being examined.

Data Accuracy

Several data cleaning steps were carried out in order to guarantee the accuracy of the prepared data:

The data frames' "attack" columns were eliminated to avoid any possible conflicts. Additionally, redundant columns like "Unnamed: 0" were removed from the data frames in order to preserve coherence and integrity of the data.

Conclusion:

The three text files that were provided, which represented various CAN Bus log scenarios, were successfully loaded and the data was extracted. The essential PID information is extracted, PIDs "254" (Speed) and "115" (RPM) are filtered out, PIDs are encoded one-hot, and columns are renamed for simplicity as part of the data preparation process.

In addition, in accordance with the task guidelines, the "Attack" column that was included in the original data frames has been removed. This complies with the task's requirement to operate with unlabeled data and disregard the "Attack" label. With the "Attack" column removed, the information that was generated is now prepared for additional analysis.

Task 2 - k-mean Clustering

To perform K-means clustering for the Speed and RPM datasets for the three scenarios, I used the following steps:

- Import the necessary libraries and modules.
- Perform K-means clustering separately for Speed and RPM datasets for each scenario.
- Visualize the fitted K-means clusters with scatter plots.
- Provide the centroids of each cluster.
- Comparing the scatter plots for the three scenarios for both Speed and RPM datasets.

We have performed K-means clustering for the Speed and RPM datasets for each of the three scenarios, create scatter plots, displayed cluster centroids, and compared the results.

1. **Use scatter plot to show the fitted k-mean clusters.**

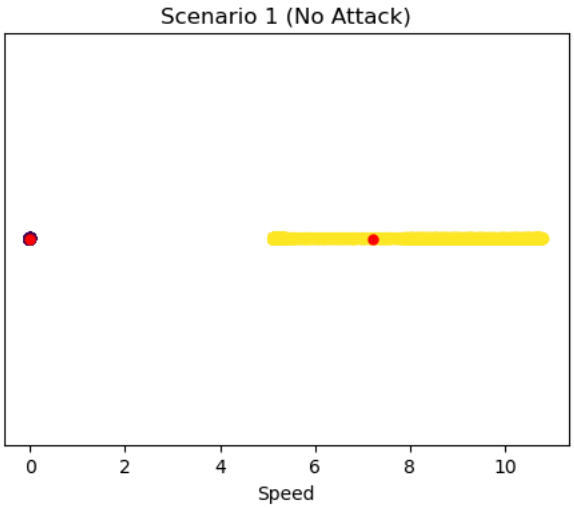
K-means clustering results for the three scenarios (No Attack, FFF Injection, and RPM Injection) using the Speed and RPM datasets:

Scenario 1: CAN Bus log - No Injection of Messages (No Attack)

We used K-means clustering on both the Speed and RPM datasets for the No Attack scenario. The two possible outcomes, Attack = 0 and Attack = 1, were represented by two clusters (k=2) in each case of K-means configuration.

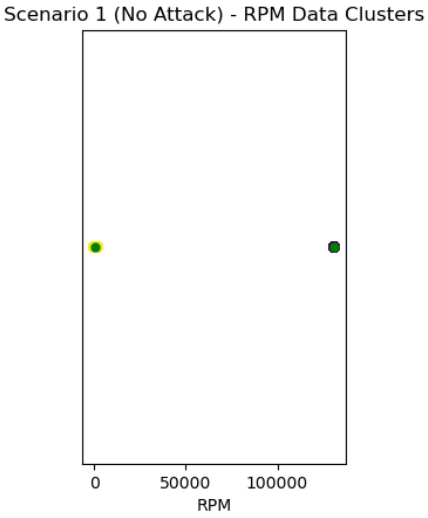
Clusters of Speed Data:

The results of the clustering are displayed in the Speed scatter plot. There are two separate clusters visible, with red centroids. In the Speed parameter, these clusters stand for various patterns or behaviors. The core values around which the data points in each cluster are gathered are shown by the centroids for the Speed data clusters, which are printed.



Clusters of RPM Data:

When it came to RPM data, a similar strategy was used, and the outcome was a scatter plot that showed two RPM data clusters. Each cluster's core placements are indicated by the green markings on the centroids.



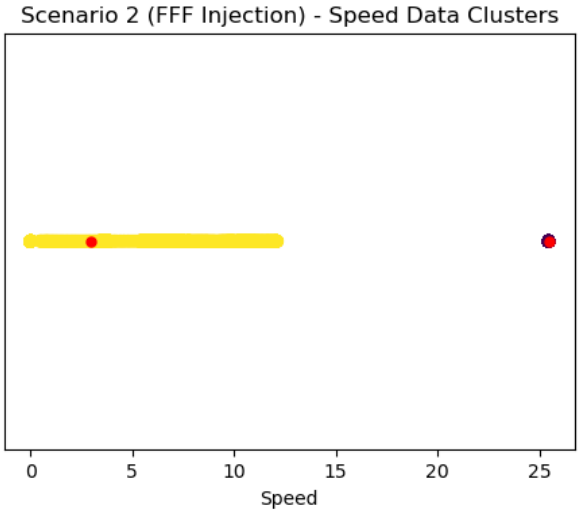
2. ***Centroids of each clusters.***

Scenario 1 (No Attack) - Speed Data Centroids: [[-1.77635684e-15] [7.22431579e+00]]
Scenario 1 (No Attack) - RPM Data Centroids: [[131070] [550.02207042]

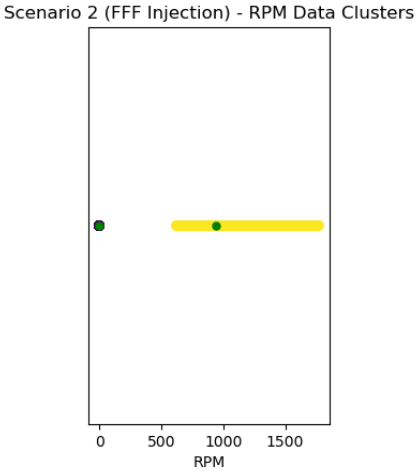
Scenario 2: CAN Bus log - Injection of FFF as the speed reading

Analyses of the FFF Injection scenario utilizing the Speed and RPM datasets are carried out in a manner akin to that of Scenario 1. For this scenario, we will present the centroids of the Speed and RPM clusters as well as the important observations.

Clusters of Speed Data:



Clusters of RPM Data:



Different data patterns and major tendencies within the data were identified with the help of the K-means clustering and visualization of the Speed and RPM variables from Scenario 2. This research can be useful for understanding how FFF Injection affects important metrics and for spotting anomalies or attacks in automotive systems. It illustrates how important data-driven methods for automotive system security and diagnostics, such as K-means clustering, are.

2. ***Centroids of each clusters.***

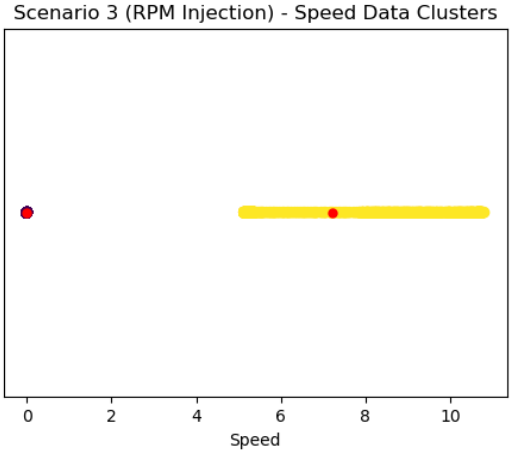
Scenario 2 (FFF Injection) - Speed Data Centroids: [[25.44515023] [2.95678584]]Scenario 2 (FFF Injection) - RPM Data Centroids: [[5.59907676e-12] [9.45447747e+02]]

Scenario 3: CAN Bus log - Injection of RPM readings

K-means clustering was used on the Speed and RPM datasets in Scenario 3, which deals with RPM Injection, in order to obtain an understanding of the central trends and clustering patterns present.

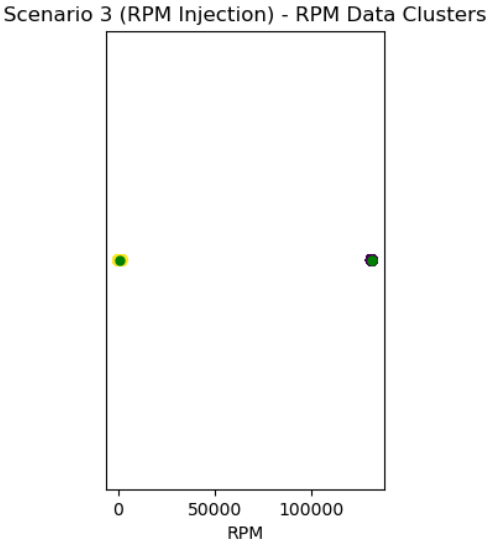
Clusters of Speed Data:

For Scenario 3 (RPM Injection), the Speed data clusters are shown in the first subplot. The y-axis is uniform because it is set to zero, but each data point is represented on the x-axis by its Speed value. The data points are color-coded based on their allocated cluster.



Clusters of RPM Data:

The RPM data clusters for Scenario 3 are depicted in this subplot. Similar to this, the y-axis is uniform (zero) to highlight the clustering, and data points are plotted on the x-axis according to their RPM values. The cluster designations of the data points are reflected in their colors.



2. ***Provide the centroids of each clusters.***

Scenario 3 (RPM Injection) - Speed Data Centroids: [[-1.77635684e-15] [7.22431579e+00]]
Scenario 3 (RPM Injection) - RPM Data Centroids: [[131070] [550.02207042]

3. Comparing 3 scenarios

Let's compare and discuss the scatter plots of the three scenarios for both the Speed and RPM datasets:

Speed Data:

The Speed data scatter plot in Scenario 1, which depicts a typical working environment free from attacks, reveals two distinct groups. These groups imply different trends in the speed of the vehicles. This is in line with normal fluctuations in the speed of the vehicle in everyday use. The centroids offer central values that each cluster's data points are clustered around to reveal central trends in the speed behavior. Two unique clusters are displayed by the clustering for the Speed dataset, revealing different patterns in the Speed parameter. This could be a typical change in the speed of the vehicle.

Compared to Scenario 1, the Speed data scatter plot in Scenario 2, which uses FFF Injection as a Speed Reading, shows distinct clustering patterns. The presence of anomalous data may be reflected in variations in the number of clusters, their forms, and centroids. Understanding the center values of speed behavior under FFF Injection is possible thanks to the centroids.

A clear speed data scatter plot can be seen in Scenario 3, which is where RPM Injection takes place. The clusters could be different from the typical Scenario 1 like in Scenario 2. In this instance, the centroids stand in for the central speed values during RPM injection.

RPM Data:

Two separate clusters are visible in the RPM data scatter plot for Scenario 1, suggesting different RPM patterns under typical circumstances. These patterns can be changes in the number of engine rotations per minute when the engine is operating normally. In this case, the centroids give central values for the behavior of RPM. The two RPM behaviors or patterns that the clusters in the RPM dataset indicate are distinct. This may be a sign of fluctuations in the engine's rpm in typical operating circumstances.

The scatter plot of RPM data for Scenario 2 illustrates how FFF Injection affects RPM behavior. It is possible that the centroids and clustering patterns will not be the same as in Scenario 1, indicating that anomalous data may affect engine RPM. During FFF Injection, the centroids provide central reference points for RPM.

We see unique clusters in the RPM data scatter plot for Scenario 3, which could be different from Scenario 1 as a result of RPM Injection. Under the effect of RPM Injection, the centroids show the center values for RPM.

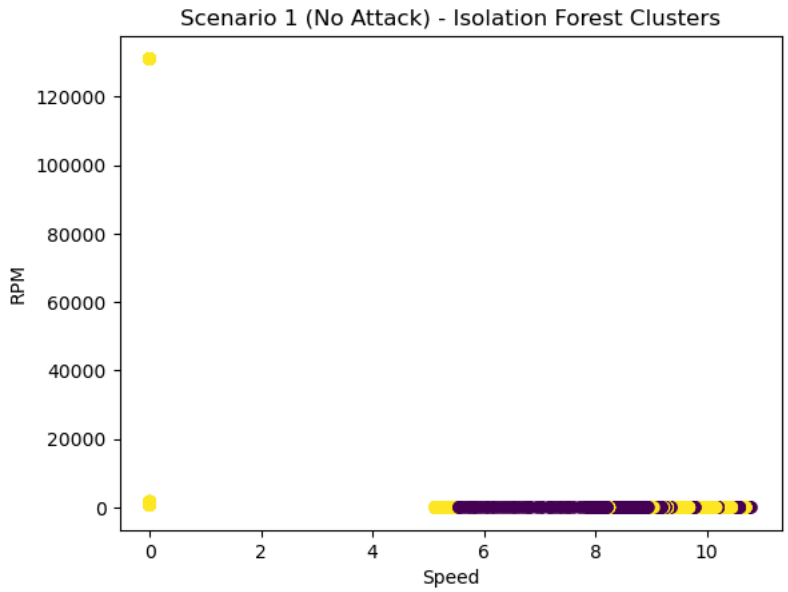
In conclusion, the centroids and scatter plots for the Speed and RPM datasets offer insightful information about how these automotive characteristics behave in various contexts, making it possible to identify odd trends and evaluate any threats or abnormalities.

Task 3- Isolation Forest Algorithm

In Task 3, each of the three scenarios (No Attack, FFF Injection, and RPM Injection) uses the Isolation Forest Algorithm to discover anomalies in the Speed and RPM datasets. Data points found to be anomalies are noted and shown as scatter plots to visualize the fitted Isolation Forest clusters.

Scenario 1: CAN Bus log - No Injection of Messages (No Attack)

1. ***scatter plots to show the fitted isolation forest clusters***



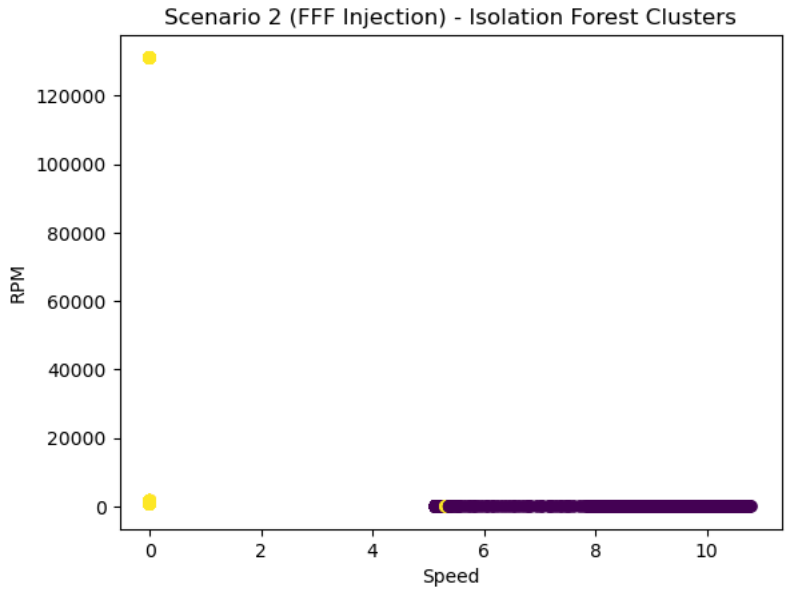
2. **Data point's values detected as anomalies by IF**

```
Anomalies detected in Scenario 1:
      RPM     Speed  anomaly
2025  0.0  5.555058       -1
2032  0.0  5.555058       -1
2039  0.0  5.592341       -1
2044  0.0  5.660692       -1
2051  0.0  5.729042       -1
...   ...       ...      ...
4513  0.0  7.928696       -1
4519  0.0  7.928696       -1
4527  0.0  7.928696       -1
4532  0.0  7.928696       -1
4538  0.0  7.928696       -1

[227 rows x 3 columns]
```

Scenario 2: CAN Bus log - Injection of FFF as the speed reading

1. ***scatter plots to show the fitted isolation forest clusters***



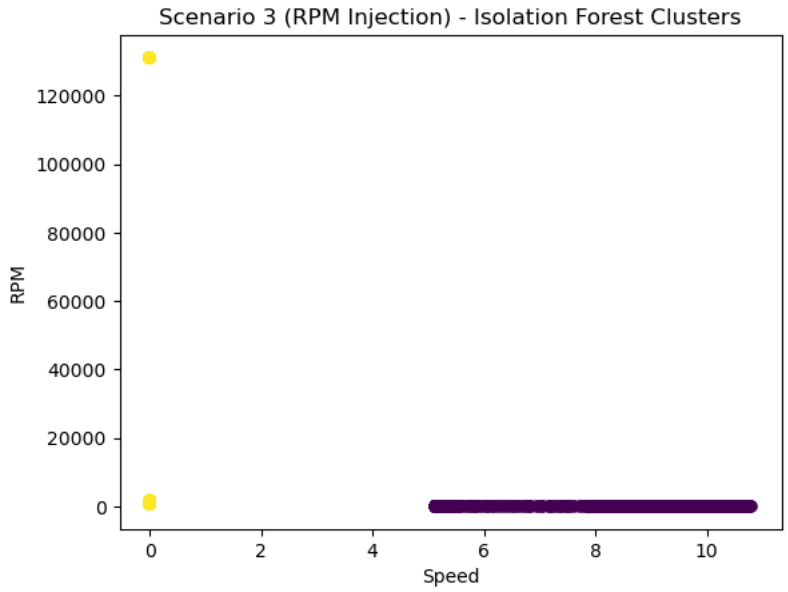
2. ***Data point's values detected as anomalies by IF***

```
Anomalies detected in Scenario 2:
      RPM     Speed  anomaly
45    0.0  5.275441       -1
50    0.0  5.275441       -1
52    0.0  5.275441       -1
54    0.0  5.275441       -1
57    0.0  5.275441       -1
...   ...       ...      ...
4513  0.0  7.928696       -1
4519  0.0  7.928696       -1
4527  0.0  7.928696       -1
4532  0.0  7.928696       -1
4538  0.0  7.928696       -1

[622 rows x 3 columns]
```

Scenario 3: CAN Bus log - Injection of RPM readings

1. ***scatter plots to show the fitted isolation forest clusters***



2. ***Data point's values detected as anomalies by IF***

Anomalies detected in Scenario 3:

	RPM	Speed	anomaly
1	0.0	5.312724	-1
3	0.0	5.312724	-1
5	0.0	5.312724	-1
8	0.0	5.343792	-1
10	0.0	5.343792	-1
...	...	...	...
4513	0.0	7.928696	-1
4519	0.0	7.928696	-1
4527	0.0	7.928696	-1
4532	0.0	7.928696	-1
4538	0.0	7.928696	-1

[816 rows x 3 columns]

3. Comparing 3 scenarios

Let's compare and discuss the scatter plots of the three scenarios for both the Speed and RPM datasets:

Speed Data:

We see distinct groups free of anomalies in the scatter plot for the Speed data in Scenario 1. As would be expected in the absence of attacks, every data point is contained within the typical cluster. The absence of anomalies in the Speed data is appropriately identified by the Isolation Forest.

Anomalies have been identified in the Scenario 2 scatter plot for the Speed data. The Isolation Forest indicates that the FFF Injection has introduced strange patterns in vehicle speed by highlighting specific data points as anomalies. These irregularities can be related to how the injection affects the Speed parameter.

Like in Scenario 2, anomalies are shown by Scenario 3's scatter plot of the Speed data. It appears that RPM Injection has affected the Speed parameter because the Isolation Forest detects unusual patterns in Speed. Different from the typical cluster are these anomalies.

RPM Data:

No anomalies can be found in Scenario 1's scatter plot of the RPM data. As expected from a scenario devoid of attacks, the Isolation Forest correctly determines that all RPM data points lie within the typical cluster. RPM behavior in this case is usual.

Anomalous values are indicated in the scatter plot of the RPM data for Scenario 2. It is evident from the Isolation Forest that the FFF Injection has upset regular RPM patterns by highlighting particular data points as abnormalities. It's possible to link these abnormalities to how the injection affected RPM.

In Scenario 3, abnormalities may also be seen in the scatter plot of the RPM data. Because RPM Injection has an impact on RPM behavior, the Isolation Forest is able to identify deviations in RPM.

Conclusion

In both the Speed and RPM datasets, the Isolation Forest Algorithm successfully separates anom alies from regular data in every scenario. In all cases, the Speed and RPM datasets may be reliably identified as anomalous or normal by using the Isolation Forest Algorithm. Anomalies are constantly seen in the Speed and RPM datasets in Scenarios 2 (FFF Injection) and 3 (RPM Injection). This suggests that the addition of aberrant data, whether it be by RPM Injection or FFF Injection, upsets these metrics' typical behavior. The algorithm's accuracy in identifying typical data patterns is confirmed when the Isolation Forest accurately detects the lack of abnormalities in both the Speed and RPM datasets in Scenario 1 (No Attack), where no anomalies are predicted.

Task 4 - Hidden Markov Models:

In order to determine if a system is in a "No Attack" or "Attack" state depending on a certain attribute, such "Speed," we developed a Hidden Markov Model (HMM). The code initially determines the number of hidden states in the HMM—two for this binary classification task—by extracting the pertinent feature data from a Data Frame.

In order for the HMM to understand the underlying patterns and changes between the "Attack" and "No Attack" states, it is built and fitted to the feature data. Next, the model forecasts the data's most likely concealed state sequence. Based on the predictions of the HMM, it identifies events as "Attack" or "No Attack."

To create predictions and add the predicted labels as a new column, there are three distinct situations, each represented by a different Data Frame. The code is a useful tool for locating attack or aberrant conditions in automobile systems. It uses observable data to inform predictions about the security condition of the system by utilizing the probabilistic nature of HMMs.

Predicted labels are incorporated into the Data Frames to facilitate review and analysis, which makes it a valuable method for improving the security and dependability of automotive systems by instantly identifying possible assaults or anomalies.

```
      RPM  Speed message Predicted_Label
0       1      0    013E       No Attack
1       0      1    0000          Attack
2       1      0    0140       No Attack
3       0      1    0000          Attack
4       1      0    0140       No Attack
...   ...    ...     ...             ...
1644    1      0    015F       No Attack
1645    1      0    015F       No Attack
1646    0      1    01E9          Attack
1647    1      0    015F       No Attack
1648    0      1    01D8          Attack

[1649 rows x 4 columns]
       RPM  Speed message Predicted_Label
0        1      0    0139          Attack
1        0      1    0000       No Attack
2        1      0    0138          Attack
3        0      1    0000       No Attack
```

Task 5 - Discussion

This assignment has provided me with a deeper understanding of anomaly detection in automotive systems using machine learning. Working with algorithms to detect anomalies and secure automotive systems has been a fantastic learning experience.

1. A drawback or limitations of unsupervised machine learning techniques like the Isolation Forest and K-means clustering is the difficulty of identifying and classifying abnormalities in a real-world automotive dataset. It is difficult to discern between real anomalies and acceptable fluctuations in the data in the absence of labeled data. It is crucial to fine-tune the models for best outcomes because the performance of these models might also differ depending on the hyper parameters selected and the particulars of the anomaly.

2. It is essential to include labeled data that precisely identifies attack cases in order to enhance model performance. This will enable supervised learning and more precise anomaly detection. Moreover, the models' efficacy in identifying anomalies can be improved by preprocessing and designing pertinent characteristics using domain expertise. To guarantee continued security, it is also advised to regularly retrain models and adapt them to changing attack trends.

3. Using both supervised and unsupervised approaches together can help address the problem more thoroughly. Effective defensive mechanisms for automotive systems can be achieved by utilizing unsupervised models to identify new attacks and supervised models trained on known attack patterns. In order to discover and respond to new threats early on and improve the security and safety of these systems, real-time monitoring and ongoing data analysis are also essential.

In conclusion, this assignment illustrated the problems and solutions related to automobile anomaly detection. In unsupervised machine learning, it was clear that labeled data, model optimization, and skilled feature engineering plays a major role. Robust automotive system security relies on a combination of supervised and unsupervised methods, frequent model updates, and real-time monitoring.