Classification of Accelerometer data

Raw data Interpretation:

The datasets consist of accelerometer measurements from helicopters, recorded over a 1-minute period at a frequency of 1024 Hz. This results in time series data captured at a total of 60 * 1024 = 61,440 equally spaced time points. The dataset includes anomalies, offering valuable insights for detecting irregularities in helicopter behaviour.

Feature extraction is essential to simplify high-dimensional time series data (accelerometer measurements), by identifying the most relevant information that represents the underlying patterns. This reduces computational complexity and enhances model interpretability, allowing for more accurate classification or anomaly detection in helicopter behaviour. By transforming raw data into meaningful features, we can improve the effectiveness of machine learning algorithms in capturing critical insights.

Feature Extraction:

Four types of features are extracted from the data. They are

1. Time domain features
2. Frequency domain features
3. PCA components
4. Auto Correlation features

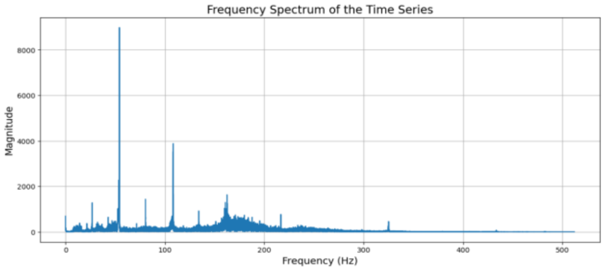
Time domain features:

- **Mean**: The average value of the data series, providing a baseline level. In anomaly detection, deviations from the mean can indicate outliers or unusual behaviour.
- **Standard Deviation**: Measures the dispersion of the data points from the mean. A high standard deviation indicates high variability, which can signal anomalies if the variability exceeds expected thresholds.
- **Variance**: The square of the standard deviation, representing the spread of the data. Like standard deviation, variance helps identify the stability of the data over time, where unexpected changes can highlight potential anomalies.
- **Minimum and Maximum**: These provide the range of the data. Monitoring extreme values can help identify outliers, as data points that fall outside this range can signify abnormal events.
- **Skewness**: Indicates the asymmetry of the distribution. A significant skew can suggest anomalies, especially if data is expected to follow a normal distribution.

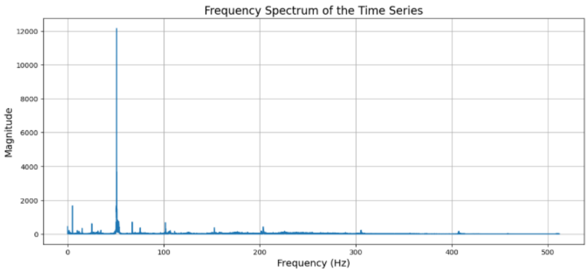
- **Kurtosis**: Measures the "tailed ness" of the distribution. High kurtosis can indicate heavy tails, suggesting the presence of outliers, while low kurtosis can suggest a lack of extreme values.
- **Root Mean Square (RMS)**: This quantifies the magnitude of the series and can highlight abnormal fluctuations. In anomaly detection, unusually high RMS values can indicate significant deviations in signal strength.
- **Zero Crossing Rate**: The rate at which the signal crosses the zero line. This feature can indicate changes in the underlying process and is useful for detecting shifts or anomalies in the signal's behaviour.

Frequency domain Features:

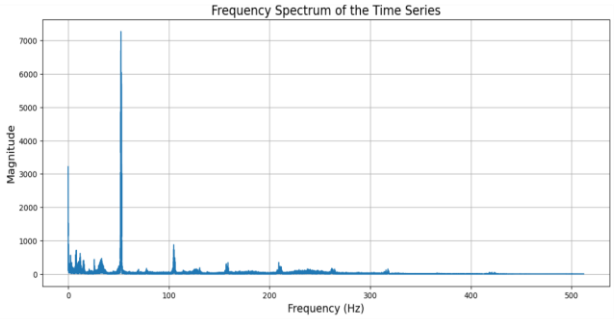
- **Dominant Frequency:** This feature represents the frequency with the highest amplitude in the signal. Identifying the dominant frequency helps detect periodic behaviours in data. Deviations from expected dominant frequencies can indicate anomalies, such as changes in system behaviour or failures. The below figures 1-4 represent the Fourier spectrum of the time series data points. It can be noticed that for the Dominant Frequency changes among different data points.



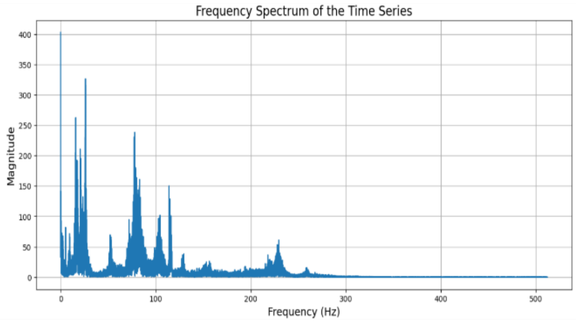
1.Normal datapoint



2.Normal datapoint



3.Normal datapoint



4.Abormal datapoint

- **Total Power:** This quantifies the overall energy present in the frequency domain. It is calculated as the sum of the squares of the FFT magnitudes. A sudden change in total power can signal significant shifts or abnormalities in the underlying process, such as mechanical wear or electrical issues.

- **Spectral Entropy:** This measures the unpredictability or complexity of the frequency distribution. High spectral entropy indicates a more complex signal, while low entropy suggests periodicity.
- **Frequency Variance:** This feature quantifies the spread of the frequency components in the FFT. High variance can indicate diverse frequency content, which may arise from irregularities or anomalies in the signal.

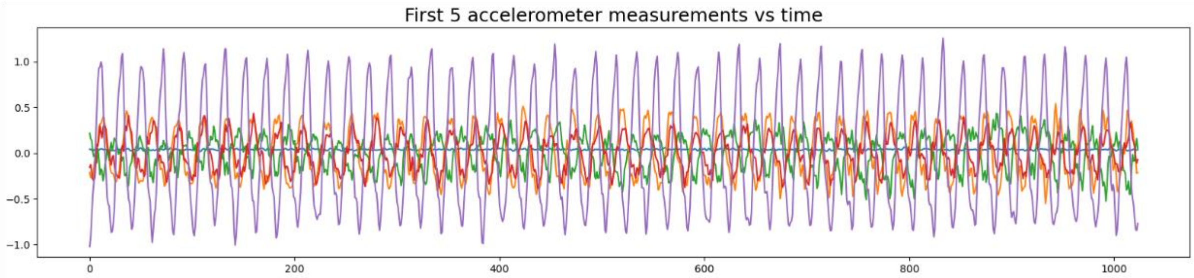
PCA components:

Principal Components 1 to 5: These components represent the directions of maximum variance in the data, effectively reducing its dimensionality while retaining the most important features. By focusing on the primary axes of variance, PCA highlights the underlying structure of the data.

- **Importance in Anomaly Detection:** Anomalies often manifest as deviations in the principal component space. Monitoring the values of these components can help identify outliers that do not conform to the expected patterns, thus improving the robustness of anomaly detection algorithms.
- **Dimensionality Reduction**: PCA simplifies complex datasets by transforming them into a lower-dimensional space. This not only enhances computational efficiency but also makes it easier to visualize and analyse the data, allowing for more effective anomaly detection

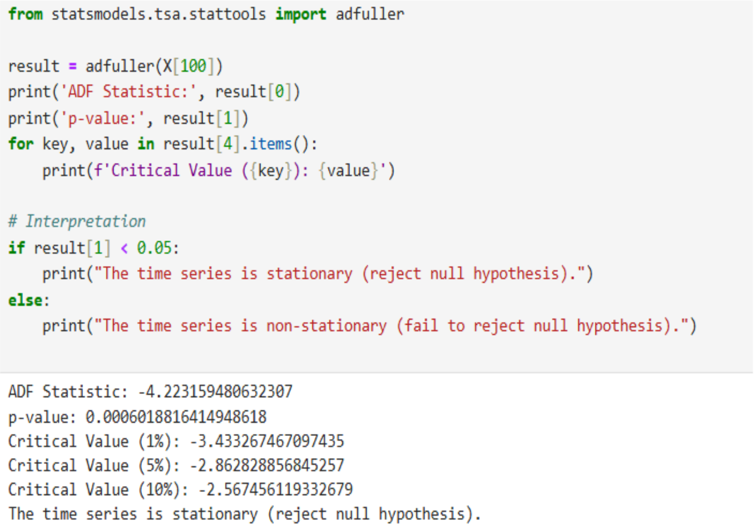
Auto Correlation features:

The figure 5 shows the plot of first five accelerometer measurements in the data. It can be observed that the there is some sort of trend and the current value at time (t) might depend on the previous values at time (t-1, t-2 etc)



5. Accelerometer measurements vs time plot

The correlation (or relationship) between a time series and its lagged versions over different time intervals can be known using ACF or PACF. Specifically, it calculates how well the time series at time t is related to the time series at previous time steps (lags). **Stationarity** is a critical condition for applying ACF and PACF (and for using time series models). Without stationarity, the correlations detected by ACF and PACF may be misleading. The **Dickey-Fuller** test, also known as the Augmented Dickey-Fuller (ADF) test, is a statistical test used to check if a time series is stationary or if it contains a unit root, meaning it is non-stationary. The stationarity can be checked as shown in the figure 6. If p-value is less than 0.05, then it means that time series is stationary.

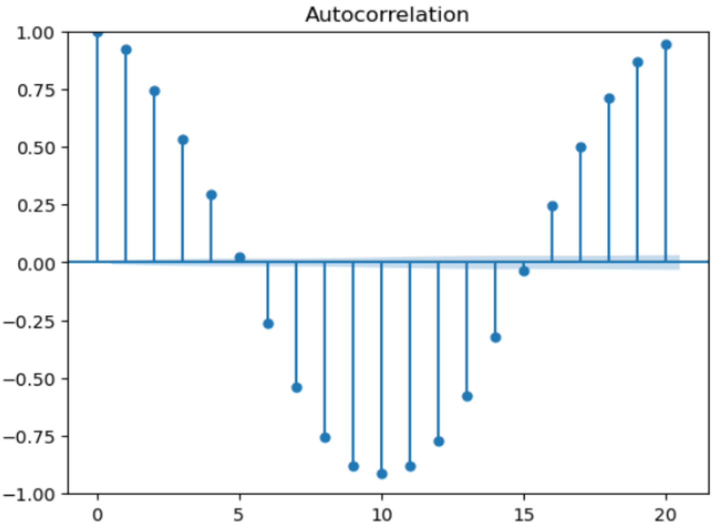


```
from statsmodels.tsa.stattools import adfuller

result = adfuller(X[100])
print('ADF Statistic:', result[0])
print('p-value:', result[1])
for key, value in result[4].items():
    print(f'Critical Value ({key}): {value}')

# Interpretation
if result[1] < 0.05:
    print("The time series is stationary (reject null hypothesis).")
else:
    print("The time series is non-stationary (fail to reject null hypothesis).")
```

```
ADF Statistic: -4.223159480632307
p-value: 0.0006018816414948618
Critical Value (1%): -3.433267467097435
Critical Value (5%): -2.862828856845257
Critical Value (10%): -2.567456119332679
The time series is stationary (reject null hypothesis).
```

6.Dickey-Fuller Test

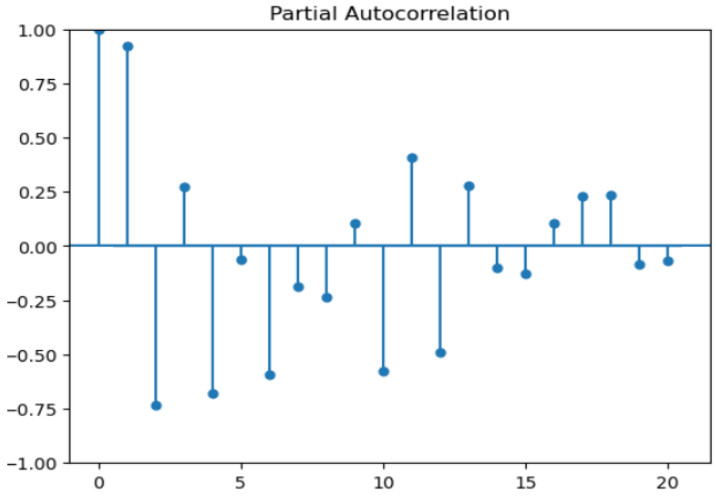
The ACF (Autocorrelation Function) plot shows the correlation between the time series and its lagged values. It is useful for identifying the number of **MA (Moving Average)** terms. Look for the point at which the ACF values significantly decrease or cut off. The below plot 7 shows ACF plot for most of the time series data points in the dataset that we are working on.



7.ACF plot

The **PACF (Partial Autocorrelation Function)** plot 8 shows the partial correlation of the time series with its lagged values, controlling for the values of shorter lags. It helps determine the number of **AR**

(Autoregressive) terms. Look for the lag where the PACF shows a sharp drop after a significant spike. The below plot shows PACF plot for most of the time series data points in the dataset that we are working on.



8.PACF Plot

As ACF plot shows a gradual decay (e.g., exponential, or sinusoidal), it suggests the presence of an **AR process**. As observed in ACF plot, the sinusoidal pattern repeats after a lag of 5. Hence the acf features till lag 5 are extracted from the data.

Final Dataset Description:

After the feature extraction, the data frame finally consists of **23 features** which are the following:

'mean', 'std_dev', 'variance', 'min', 'max', 'skewness', 'kurtosis', 'rms', 'zero_crossing_rate', 'acf_lag_1', 'acf_lag_2', 'acf_lag_3', 'acf_lag_4', 'acf_lag_5', 'dominant_freq', 'total_power', 'spectral_entropy', 'freq_variance', 'principal_component_1', 'principal _component_2', 'principal _component_3', 'principal_component_4', 'principal _component_5'

The Labels are 0.0 and 1.0:

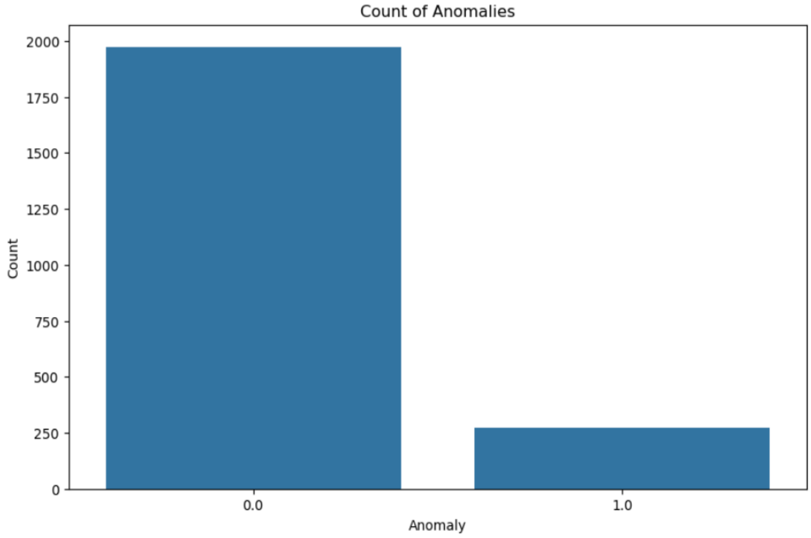
1.0 indicates the **Normal accelerometer data**

2.0 indicates the **Abnormal accelerometer data**

EDA:

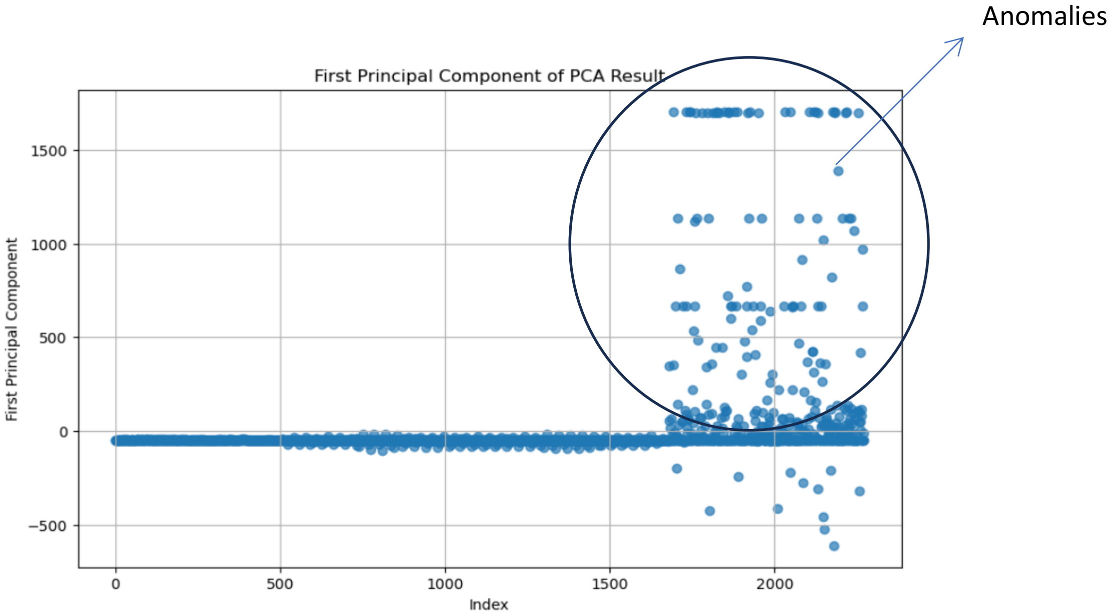
The below figure 9 shows the count plot of the datapoints with normal and abnormal behaviour. The Label 0.0 indicates the normal behaviour of the accelerometers and the Label

1.0 represents the abnormal behaviour of the accelerometers. It indicates that the dataset is unbalanced where the normal points are more than the abnormal points.



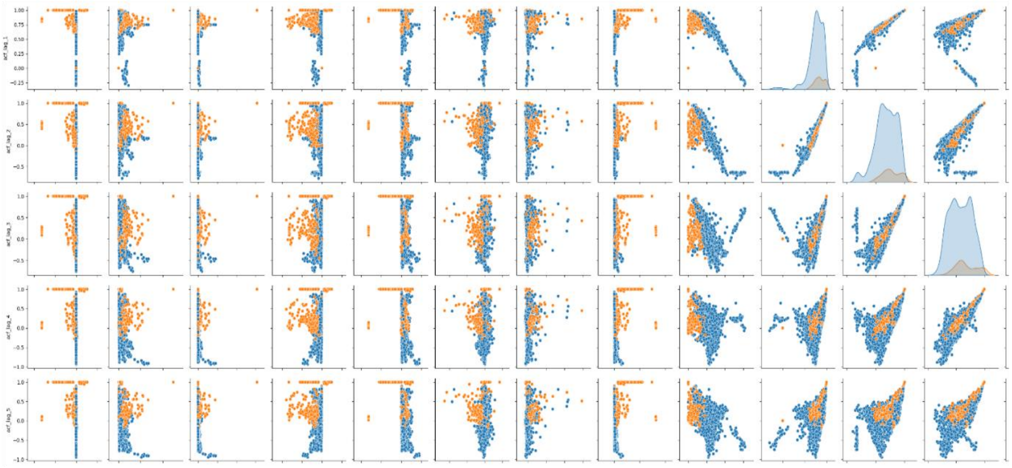
9. Count plot of Labels

The datapoints from the feature pricipal_component_1 are plotted in a scatter plot 10 as shown below. It is observed that the accelerometer data points with normal behaviour almost lie on a single line. The abnormal data points are scattered everywhere in the plot indicating the anomalies.



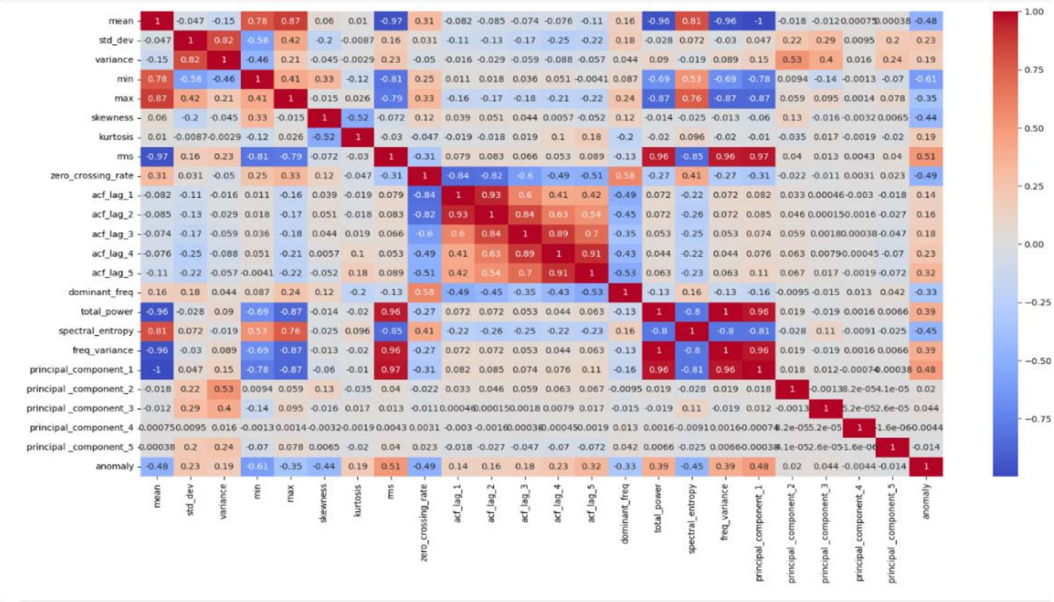
10. Scatter plot – First Principal Component

The figure 11 shows the Pair plots between different features present in the data frame. A pair plot generates a matrix of scatter plots for each pair of features in a dataset, along with histograms (or kernel density plots) along the diagonal, giving insight into the data's structure, patterns, and distributions. Pair plots are used for checking pairs that show clear separations between classes. For example, if different classes cluster distinctly in a scatter plot, the feature pair is likely useful for classification. The features like total_power, freq_variance, principal_component_1 showed clear separation between the output class.



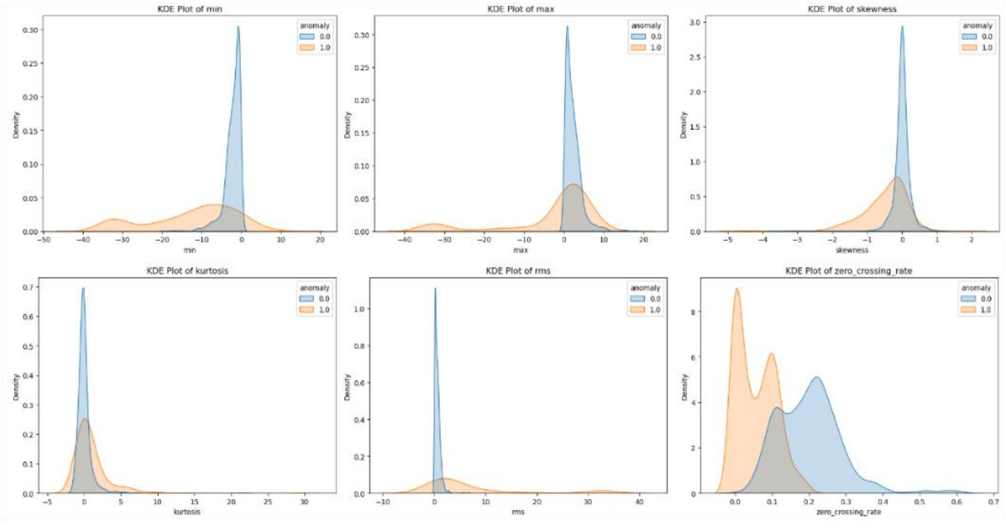
11.Pair Plot between features

The below figure 12 represents the correlation matrix. A correlation matrix represents the correlation coefficients between pairs of variables in a dataset. Each cell in the matrix shows the strength and direction of the linear relationship between two variables, with values ranging from -1 to +1. It can be observed from the below matrix that the zero_crossing_rate, acf_lag_4, total_power, freq_variance, principal_component_1 are highly correlated with the output (Anomaly).



12. Correlation matrix

KDE can estimate the distribution of features within each class. By building a KDE for each class based on training data, you can model the likelihood of a data point belonging to each class. The KDEs for different features are plotted in figure 13 to see which features are more contributing to the classification. It is observed that mean, std_dev, min, max, rms, zero_crossing_rate separates the normal and abnormal datapoints in a good way.



13. Kernel Density Function

Feature Selection, Model Fitting and Evaluation:

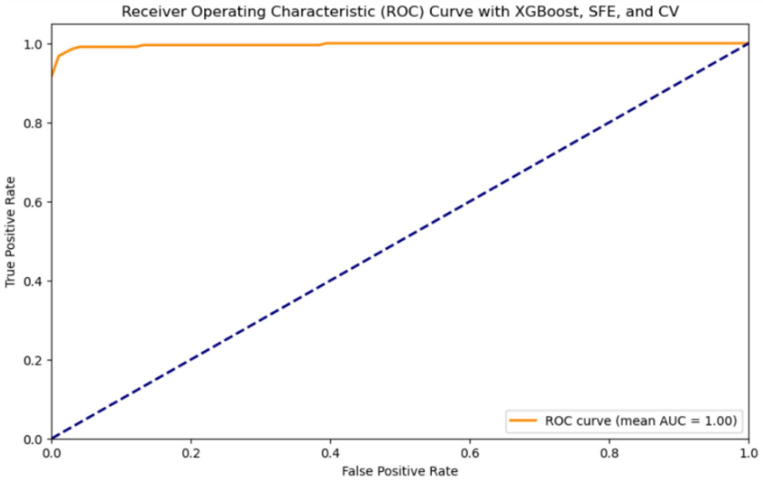
Recursive Feature Elimination (RFE) is used here for feature selection. RFE is a feature selection technique commonly used to reduce the number of features by selecting the most important ones for a model. RFE works iteratively, ranking features by their importance and eliminating the least important features step-by-step until it reaches the desired number of features. It focuses on selecting the most relevant features, which enhances the model's ability to generalize to new data. When combined with Cross-Validation (CV), RFE ensures that feature selection is robust, evaluated across different data splits, and not biased by a specific training set, leading to more reliable and consistent feature selection.

XGBoost is used here for anomaly detection because it is a powerful, efficient, and scalable machine learning algorithm that excels at handling large, high-dimensional datasets like the Airbus accelerometer data. It is robust to noise and outliers, which is important for detecting anomalies in real-world sensor data. XGBoost also provides feature importance, helping to identify which features contribute most to anomaly detection. Additionally, its ability to perform well with both small and large datasets, and its flexibility through hyperparameter tuning, makes it well-suited for complex tasks like anomaly detection. 80 percent of the data is used for training and 20 percent for testing.

Results Summary:

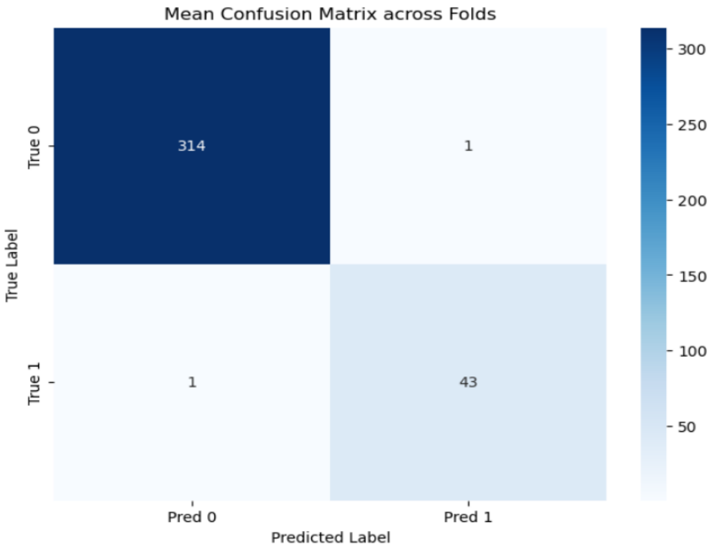
The features that are finally selected from RFE are **'mean', 'std_dev', 'min', 'max', 'skewness', 'zero_crossing_rate', 'acf_lag_4', 'acf_lag_5', 'dominant_freq', 'principal_component_1', 'principal_component_2', 'principal_component_4'**

AUC is preferred metrics for imbalanced datasets because it evaluates a model's ability to distinguish between classes across all possible thresholds, rather than focusing on raw accuracy. Since accuracy can be misleading with imbalanced data, AUC provides a more reliable performance measure by considering both true positive and false positive rates, making it insensitive to class distribution. The figure 14 shows the ROC – AUC curve for the current task. AUC value is 1 which represents the perfect classification.



14. ROC – AUC curve

The below figure 15 shows the confusion matrix for the current task. In imbalanced datasets, the confusion matrix is crucial because it reveals true and false positives and negatives, enabling more accurate evaluation than accuracy alone. This breakdown supports metrics like precision, recall, and F1 score, which highlight different aspects of performance. A good model balances these metrics, ensuring it captures the minority class effectively without excessive false positives, providing reliable, context-appropriate results for applications where each type of error carries a cost.



```
Mean Accuracy: 0.99
Mean Precision: 0.98
Mean Recall: 0.97
Mean ROC AUC: 1.00
```

15. Confusion Matrix