

ME623

DYNAMICS OF MACHINING PROCESSES

GROUP NUMBER: 3

PRESENTED BY
SHREYAS KUMAR (220103102)
SHREYA SEN (220103101)
SUDHANSU SINGH(220103105)



TOPIC:-

CHATTER DETECTION USING SUPERVISED ML ALGORITHM

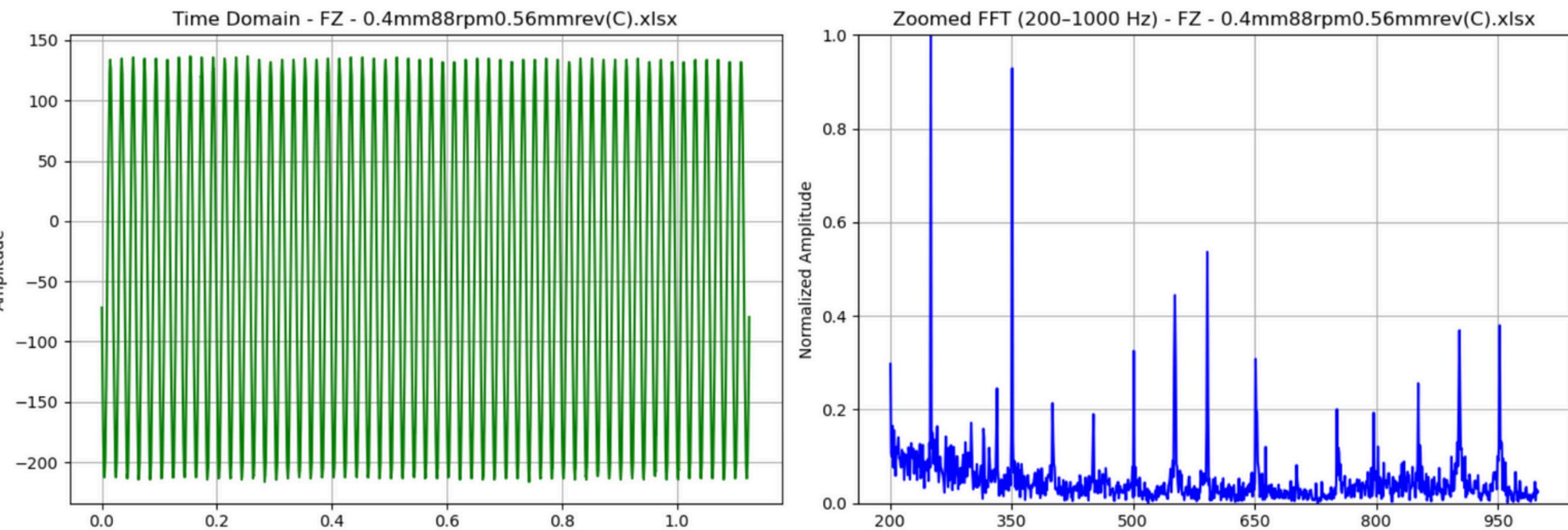
OBJECTIVE

To detect chatter during turning operations using supervised machine learning algorithm using force signal data.

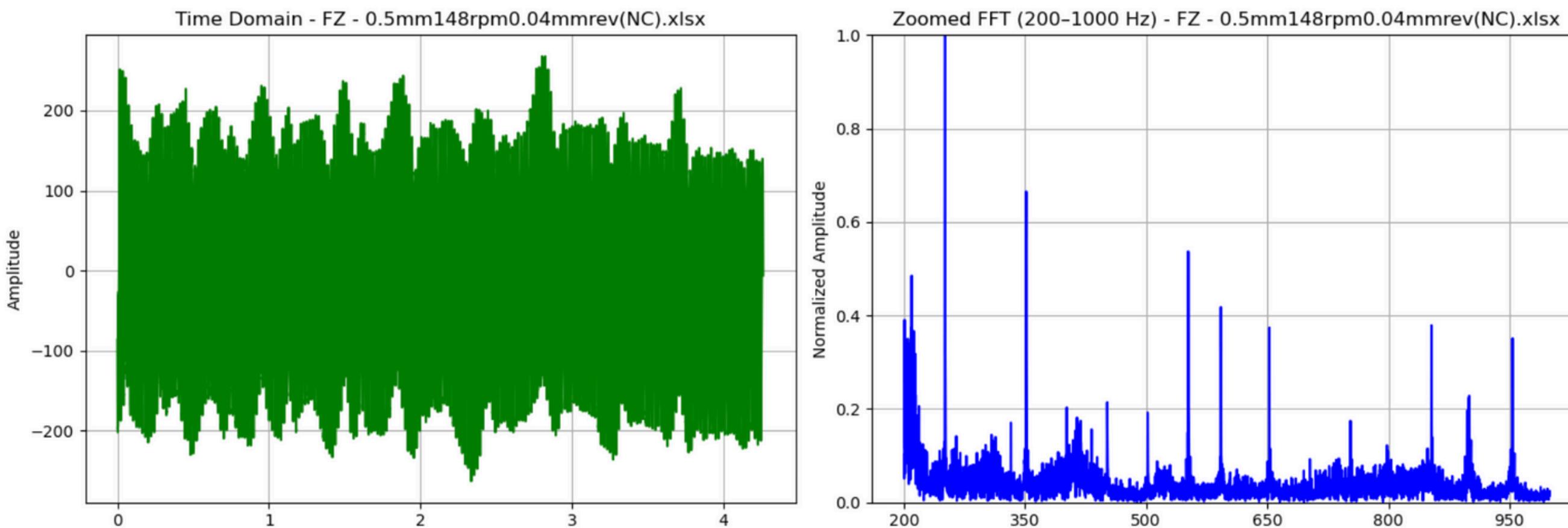
METHODOLOGY

- **Data Acquisition:** Collected force signal data by varying feed, depth of cut and rpm.
- **Feature Extraction:** Used FFT to convert time-domain force signals into frequency-domain features.
- **Preprocessing:** Applied heatmaps and PCA to remove correlated features.
- **Labelling:** Classified data into “chatter” and “non-chatter” categories.
- **Model training:** Trained using Random forest and XGBoost.
- **Evaluation:** Determined performance using precision, accuracy, and recall.
- **Prediction:** Used the model to detect chatter from test signals.

RESULTS AND DISCUSSIONS

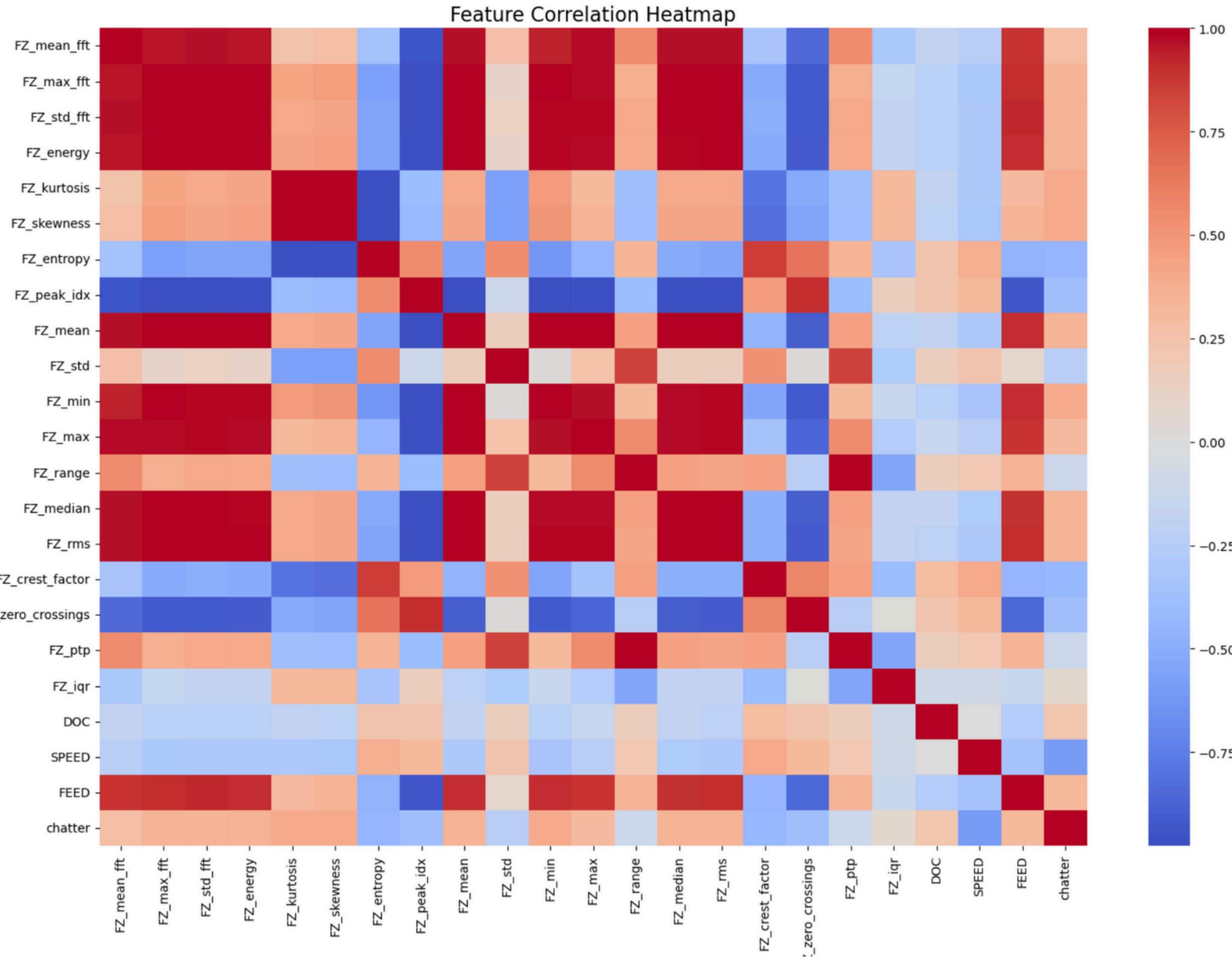


Considering first highest peak as the natural frequency in all cases, the plot with next peak above 0.8 will show chatter and for the peak<0.8 no chatter is observed.



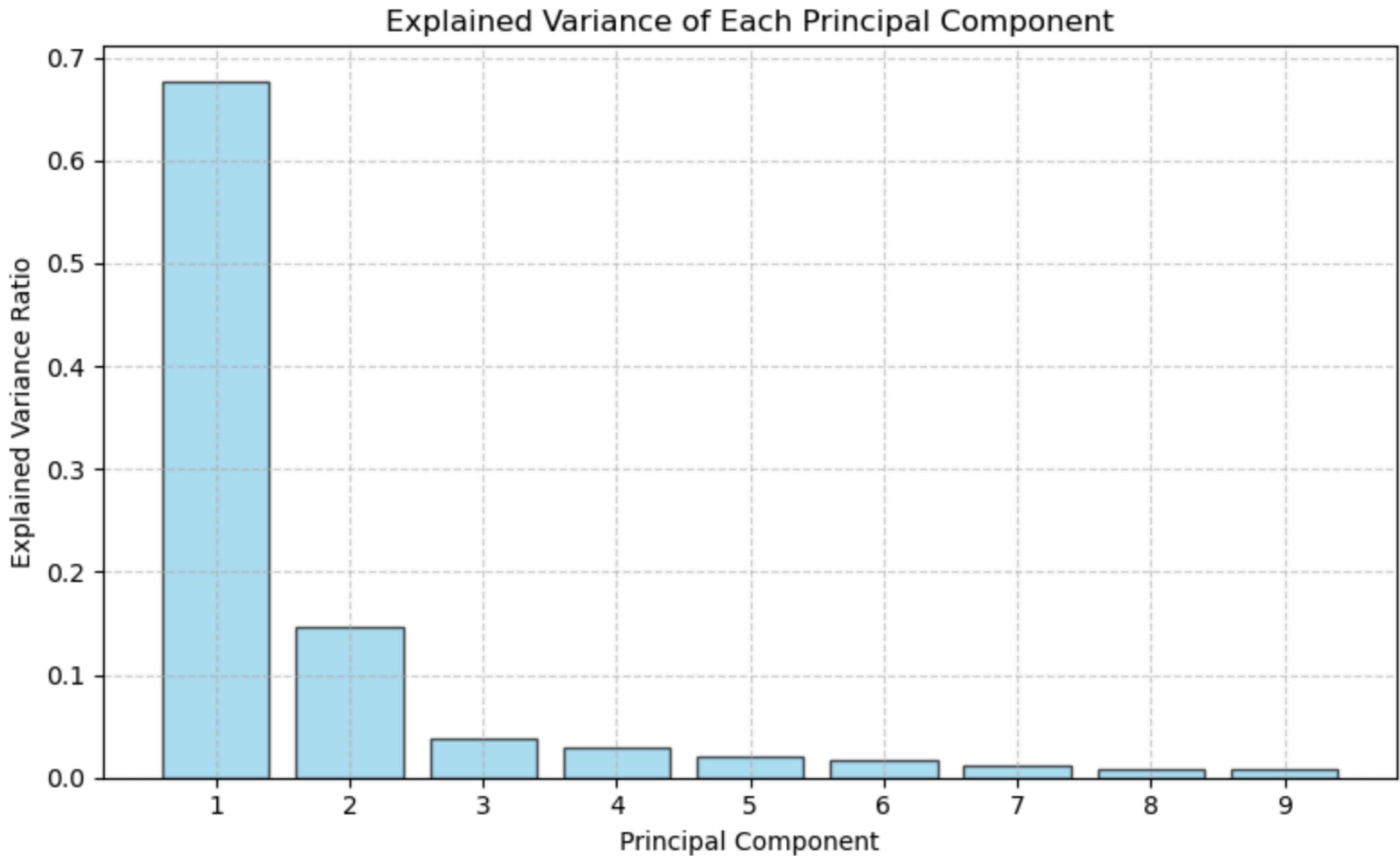
RESULTS AND DISCUSSIONS

HEATMAP TO DETERMINE CORRELATION BETWEEN ORIGINAL 22 FEATURES AND AS A RESULT UTILISING ONLY BEST 6 FEATURES



RESULTS AND DISCUSSIONS

Principal component Analysis for Correlation between original 22 features and utilizing the best 5 features.



RESULTS AND DISCUSSIONS

Confusion Matrix:
[[153 2]
 [13 129]]

Classification Report:

	precision	recall	f1-score	support
0	0.92	0.99	0.95	155
1	0.98	0.91	0.95	142
accuracy			0.95	297
macro avg	0.95	0.95	0.95	297
weighted avg	0.95	0.95	0.95	297

Train Accuracy: 0.9992
Test Accuracy: 0.9495

USING OPTION1: HEATMAP AND RANDOMFOREST



Confusion Matrix:
[[151 4]
 [3 139]]

Classification Report:

	precision	recall	f1-score	support
0	0.98	0.97	0.98	155
1	0.97	0.98	0.98	142
accuracy			0.98	297
macro avg	0.98	0.98	0.98	297
weighted avg	0.98	0.98	0.98	297

```
train_acc = best_model.score(X_train, y_train)
test_acc = best_model.score(X_test, y_test)

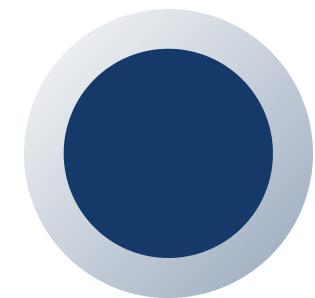
print(f"Train Accuracy: {train_acc:.4f}")
print(f"Test Accuracy: {test_acc:.4f}")
```

Train Accuracy: 1.0000
Test Accuracy: 0.9568

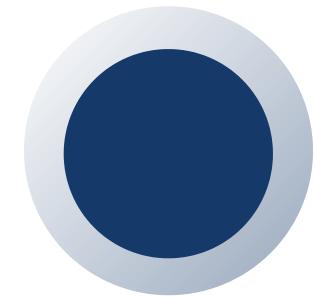
USING OPTION2: PCA AND XGBOOST



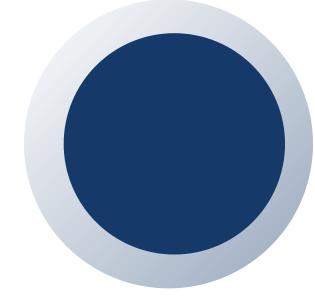
CONCLUSION



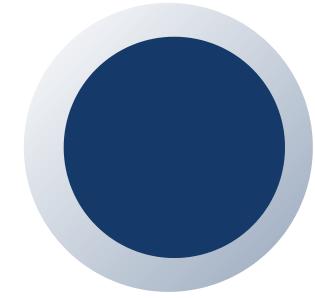
Achieved a **test accuracy** of 95.68% and 94.95% using **RandomForest** and **XGBoost** respectively.



FFT features clearly distinguished chatter vibrations, for peaks above the magnitude of 0.8 in the frequency range of 200Hz to 1000Hz, making them effective for model training.



The model generalized worked well on test data, indicating **robust feature extraction** and **low overfitting**.



Precision and Recall for chatter class is **above 90%**, showing **high reliability** in chatter identification.

-

THANK YOU!