COBEM 2023
27TH INTERNATIONAL CONGRESS OF MECHANICAL ENGENEERING

ABCM
Brazilian Society of Mechanical Sciences and Engineering
27th ABCM International Congress of Mechanical Engineering
December 04-08, 2023, Florianópolis-SC, Brazil

COB-2023-1635
Comparison of Traditional Vibration Analysis Techniques and Machine Learning Models for Bearing Fault Detection

Victor Afonso Bauler
Júlio Apolinário Cordioli
Danilo Silva
Federal University of Santa Catarina, Florianópolis - SC, 88040-900, Brazil
victor.bauler@gmail.com, julio.cordioli@ufsc.br, danilo.silva@ufsc.br

Danilo de Souza Braga
Dynamox S.A., Florianópolis - SC, 88030-909, Brazil
danilo@dynamox.net

Abstract. *Vibration analysis is a widely used technique for fault detection in rotating machinery. In recent years, machine learning techniques have been increasingly applied to this field to improve the accuracy and efficiency of fault detection. This paper compares the effectiveness of traditional vibration analysis techniques applied to rolling bearing, such as bandpass filtering and envelope analysis by means of Hilbert transform, with machine learning models, namely Support Vector Machine, Logistic Regression, and Random Forest, for bearing fault detection. The study utilizes an experimental dataset collected in the Laboratory of Vibration and Acoustics (LVA) at the Federal University of Santa Catarina (UFSC). The models are trained and tested on this dataset, and the effectiveness of each technique is evaluated based on their ability to detect bearing faults. The results of this study will contribute to the literature by providing a more objective and comprehensive evaluation of the effectiveness of different vibration analysis techniques. Furthermore, the study also explores the potential of machine learning models in improving the accuracy and efficiency of fault detection. This study provides insights into the effectiveness and limitations of each approach by comparing the performance of traditional vibration analysis techniques and machine learning models.*

Keywords: *Rolling element bearings, Envelope analysis, Vibration Analysis, Fault Detection, Machine learning models*

1. INTRODUCTION

Rotating machinery plays a crucial role in various industries like manufacturing, power generation, and transportation. By monitoring and analyzing the vibration signals produced by these machines, we can gain insights into their health condition and identify potential issues (Randall, 2021).

Rolling bearings play a critical role in maintaining motion between static and moving parts in rotating machinery, making them vital components in various industrial applications (Ahmed and Nandi, 2020). Bearing failures can lead to major breakdowns in machines, 40–90% of rotating machine failures are related to bearing faults, depending on the machine size (Bellini *et al.*, 2008).

Traditional vibration analysis techniques have proven effective in detecting faults in rotating machinery. By examining different characteristics of vibration signals, such as their frequency content, amplitude, and temporal patterns, common faults such as unbalance, misalignment, and bearing defects can be identified. Among these techniques, envelope analysis (McFadden and Smith, 1984) has emerged as a powerful tool for bearing fault detection. It enables the detection of signal modulation caused by bearing fault frequencies, which helps in early fault detection and prevention of catastrophic failures.

However, recent advancements in machine learning techniques have shown potential in enhancing fault detection in rotating machinery (Lei *et al.*, 2020). Classical machine learning models, including Logistic Regression, Support Vector Machines (SVM), and Random Forest, have distinct advantages over traditional logical rule-based systems. These models possess the ability to automatically learn complex patterns and relationships from labeled vibration data, making them well-suited for capturing intricate fault patterns and adapting to dynamic operating conditions. Leveraging large and high-dimensional fault detection datasets, machine learning models offer the promise of improved performance over time.

This study aims to investigate and compare the effectiveness of traditional vibration analysis techniques, specifically envelope analysis, and machine learning models for automatic bearing fault detection. In the context of automatic detection, a rule-based methodology referred to here as the Baseline approach is utilized as a reference point. The Baseline approach employs logical rules and predefined thresholds to identify potential faults based on vibration signal characteristics, emulating the workflow performed by a vibration analyst. By comparing the performance of the Baseline approach with that of machine learning models, this study aims to provide insights into the strengths and limitations of each approach in bearing fault detection. The findings will contribute to enhancing the reliability and effectiveness of automatic fault detection in rotating machinery.

2. METHODOLOGY

2.1 Experimental Setup

The dataset used in this study was collected at the Laboratory of Vibration and Acoustics (LVA) using the Machinery Fault Simulator (MFS) manufactured by SpectraQuest, as shown in Fig. 1. The acquisition system consisted of four accelerometers (PCB - 352C33): two sensors in the motor and one sensor in each bearing case. This study will only focus on signals obtained by the accelerometers on each bearing case, namely Drive End (DE) and Non Drive End (NDE). Additionally, an acquisition module (NI 9234) from National Instruments was utilized.

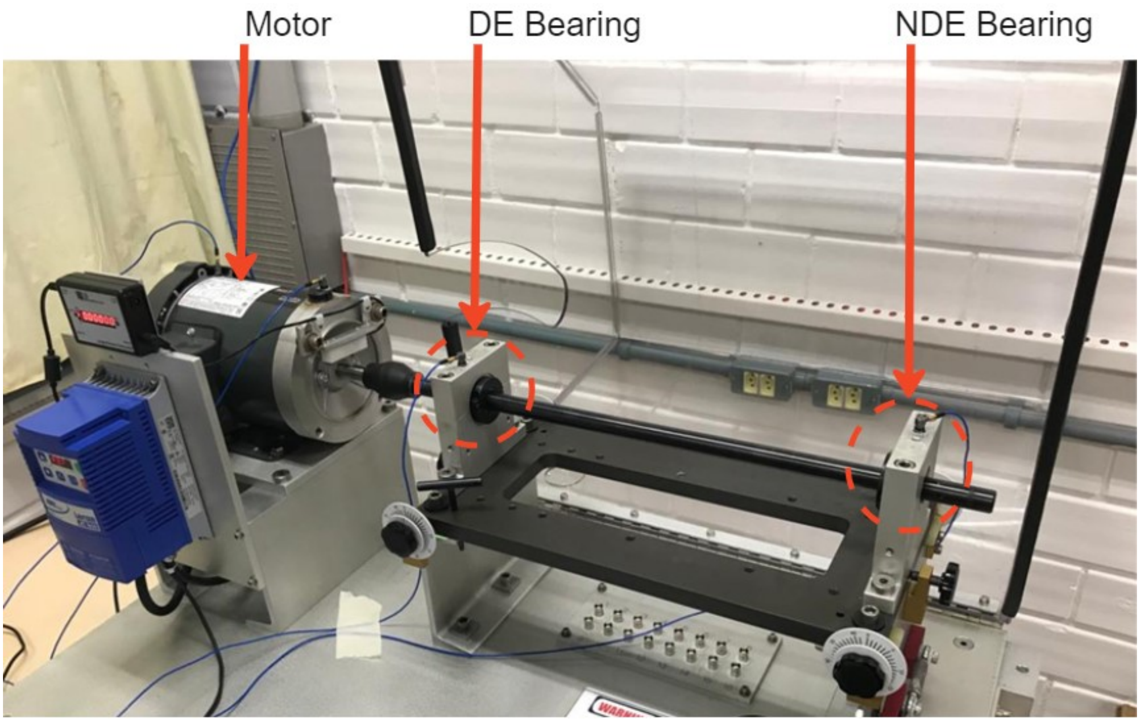

Figure 1. Accelerometers positions in SpectraQuest's Machinery Fault Simulator.

The experiments were conducted using five distinct setups, as outlined in Tab 1. The rotational speed was varied for each measurement within each setup, which covers the following Rotations Per Minute (RPM) values 100, 1000, 2000, 3000, 4000, and 5000. For setup E, the rotation speed values were slightly different, which considers 500, 1000, 2000, 3000, and 3500 RPM. Four distinct bearing defects were studied: inner race fault, outer race fault, ball fault, and combination (inner, outer, and ball) fault. The data was sampled at a frequency of 25600 Hz, and each acquisition had a duration of 30 seconds.

Table 1. Experimental setups.

Setup	Description	Number of measurements
A	Healthy condition	6
B	Bearing defect on NDE Bearing	24
C	Bearing defect on DE Bearing	24
D	Repeat setup A and B with each level of rotor unbalance	90
E	Repeat setup A and B with belt transmission connected	25

Setup A involves using healthy bearings on both the Drive End (DE) and Non-Drive End (NDE), with the rotation speed as the only variable. In setups B and C, one healthy bearing and one defective bearing are used in each measurement, with variations in both the type of defect and the rotation speed. The experimental setup D consists of three different levels of rotor unbalance. These levels are illustrated in Fig. 2, where levels 1 and 2 are associated with an unbalanced mass of 0.005 kg, while level 3 corresponds to an unbalanced mass of 0.008 kg. The Rotor is positioned in the middle of the shaft, between the two roller bearings. The E setup involves using the pulley and belts and the gearbox with minimal load, as shown in Fig. 3.

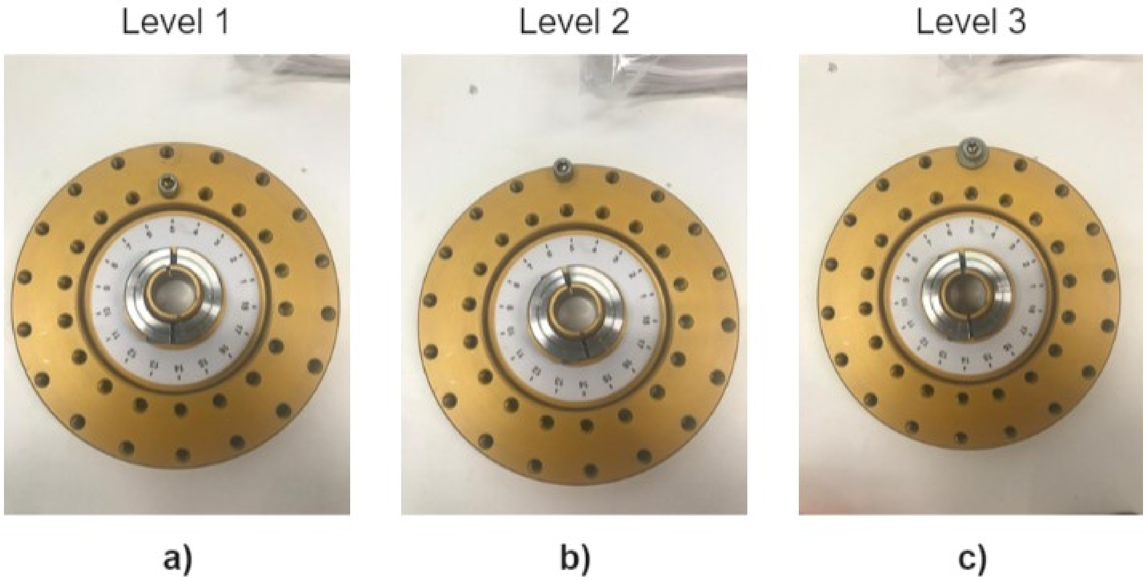

Figure 2. Rotor unbalance : a) Level 1 ; b) Level 2 ; c) Level 3.

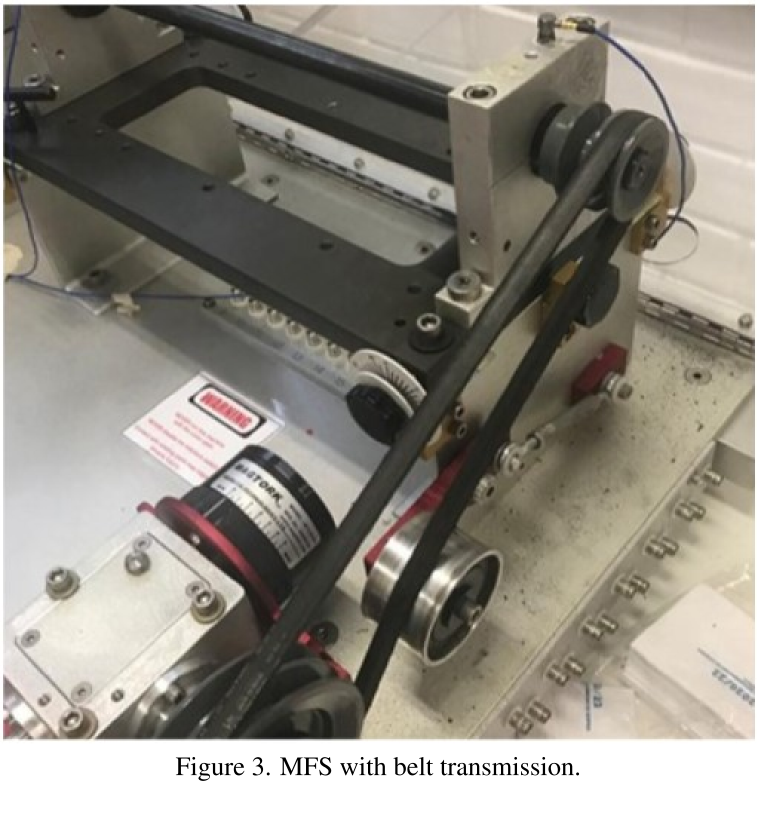

Figure 3. MFS with belt transmission.

2.2 Baseline

The Baseline model, developed as a rule-based approach, facilitates automatic fault detection and diagnosis in bearings by implementing envelope analysis. It serves as a benchmark for evaluating the performance of research models against industry-standard practices. The model emulates the workflow of a vibration analyst, encompassing a sequence of steps to analyze vibration signals and identify potential faults in the bearings.

In the initial step, a bandpass filter is applied to isolate specific frequency components relevant to bearing faults. This filter helps mitigate noise and interference (mainly related to low frequencies, i.e., other machine components), enabling subsequent analysis to focus on critical frequencies of interest. In our study, we employ four different bandpass filters in the envelope analysis for bearing fault detection. The selection of the first three envelope ranges adheres to the ISO 10-816/3 and VDI 3832 standards, which offer guidelines for machine condition monitoring and diagnostics using vibration analysis. The kurtogram (Antoni and Randall, 2006) is used to assess the kurtosis values across different frequency bands. By analyzing the kurtosis values, we can identify the frequency band with the highest kurtosis, indicating the presence of significant fault-related components. This approach allows us to determine the optimal frequency band for bearing fault detection, enhancing the accuracy and effectiveness of the analysis.

Following the signal filtering, the Hilbert transform is employed to extract the signal envelope, which represents variations in amplitude modulated over time. By applying the Fast Fourier Transform (FFT) to the envelope spectrum derived from the Hilbert transform, specific fault frequencies associated with the bearing can be identified. These fault frequencies, such as the Ball Pass Frequency Outer (BPFO), Ball Pass Frequency Inner (BPFI), Ball Spin Frequency (BSF), and Fundamental Train Frequency (FTF), manifest as modulations in the signal caused by bearing defects (Randall, 2021). This study will only focus on analyzing the BPFO, BPFI, and BSF due we are not analyzing cage faults. Mathematically, the fault frequencies can be expressed as:

$$BPFO = \frac{N_b S_h}{2}(1 - \frac{d_b}{D_p}\cos\varphi) \tag{1}$$

$$BPFI = \frac{N_b S_h}{2}(1 + \frac{d_b}{D_p}\cos\varphi) \tag{2}$$

$$BSF = \frac{D_p S_h}{2d_b}(1 - (\frac{d_b}{D_p}\cos\varphi)^2) \tag{3}$$

where N_b is the number of rolling elements, S_{sh} is the shaft speed, d_b is the rolling element diameter, D_p is the pitch diameter, and φ is the angle of the load from the radial plane. Figure 4 illustrates the image of the bearing geometry.

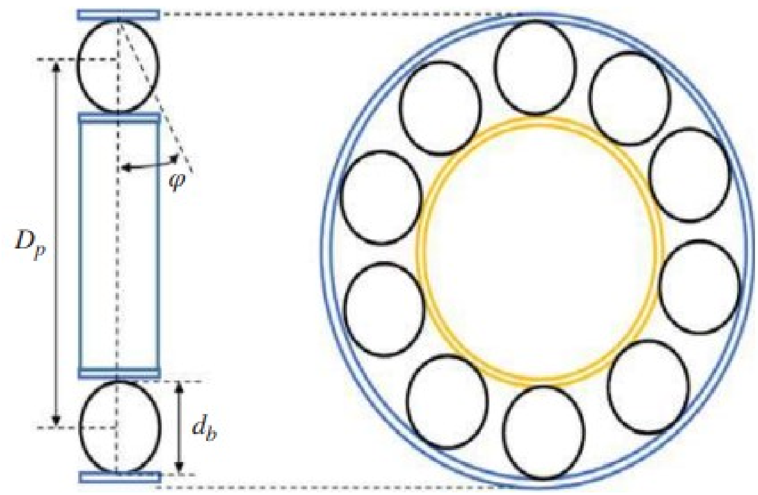

Figure 4. Rolling element bearing geometry (Ahmed and Nandi, 2020).

To automate the detection process, the Baseline model employs three autonomous binary classifiers that specialize in detecting specific fault types in bearings: inner race, outer race, and rolling elements. Each classifier focuses on analyzing harmonics at their respective fault frequencies: 1X BPFI, 2X BPFI, and 3X BPFI for inner race faults; 1X BPFO, 2X BPFO, and 3X BPFO for outer race faults; and 1X BSF, 2X BSF, and 3X BSF for rolling element faults. By evaluating the peak values within a 5% interval centered around these harmonic frequencies, the classifiers evaluate if at least two of the three evaluated peak values surpass pre-defined thresholds, specific to rotation and the frequency range determined by the bandpass filter, then the signal is classified as faulty. Figure 5 represents all steps applied on the Baseline.

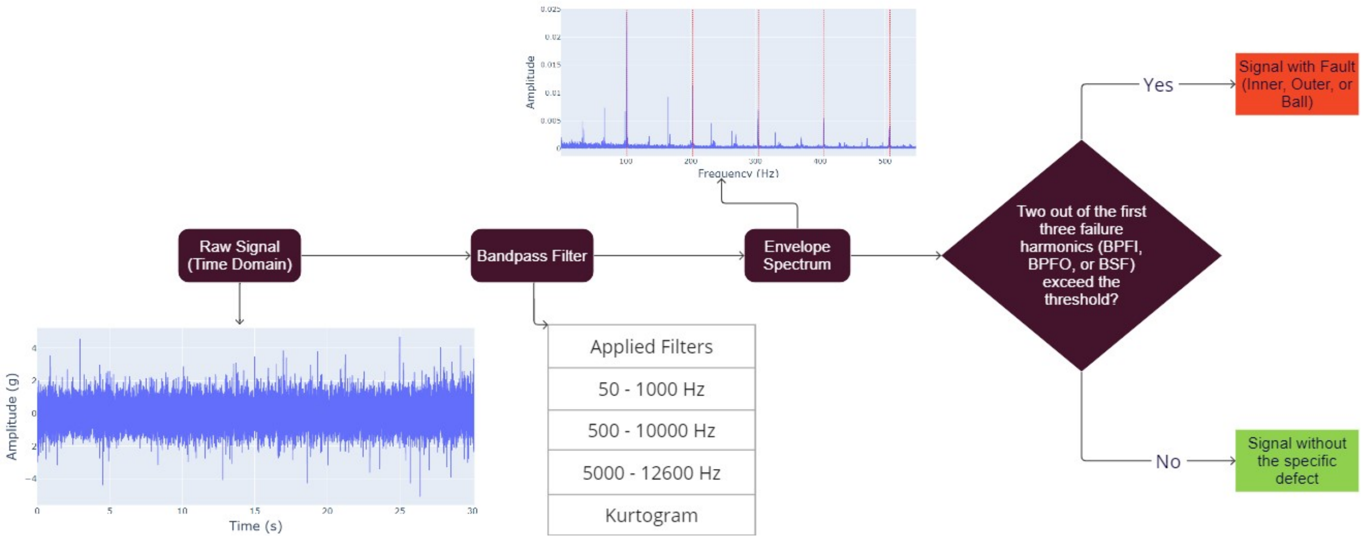

Figure 5. Baseline schema.

The threshold calculation procedure involves splitting the dataset into a training set and a test set. In the training set, the threshold is optimized to maximize the balanced accuracy, a metric that is described in detail in section 2.4. The training setup encompasses conditions A, B, and C, while the test set consists of conditions D and E.

2.3 Machine Learning Models

The machine learning models utilized in this study were implemented using scikit-learn, a widely-used machine learning library in Python (Pedregosa *et al.*, 2011). These models employed the same attributes as the Baseline model, derived from envelope analysis, with differences in the number of utilized harmonics (increased from 3 to 5) and the inclusion of RPM as an additional feature. As a result, a total of 61 features were considered, encompassing 3 distinct defects, 4 passband ranges, and 5 harmonics + RPM. The subsequent section presents the specific models used, along with details of the adopted hyperparameter optimization algorithm. Importantly, the training and testing splits employed in the machine learning models remained consistent with those of the Baseline model.

2.3.1 Logistic Regression

Logistic Regression, also known as Logit Regression, is a regression algorithm commonly employed for classification tasks. Similar to Linear Regression, Logistic Regression computes a weighted sum of the input features (Eq. 4), including a bias term. However, instead of directly outputting the result like Linear Regression, it applies the logistic function (Eq. 5) to the weighted sum. The logistic function, also known as the sigmoid function, transforms the result into a probability value between 0 and 1 (Géron, 2022).

$$\hat{p} = \sigma\left(\mathbf{w}^T\mathbf{x} + b\right) \tag{4}$$

$$\sigma(z) = \frac{1}{1+e^{-z}} \tag{5}$$

In Eq. 4, $\hat{p}$ represents the estimated probability by the logistic regression model. The parameter vector $\mathbf{w}$ contains the weights assigned to each input feature, b denotes the intercept term (also known as bias), and $\mathbf{x}$ represents the input features.

2.3.2 Support Vector Machines

Support Vector Machines (SVM) is a powerful algorithm for classification tasks that aims to find an optimal hyperplane for separating different classes (Watt *et al.*, 2020). Building upon the concepts introduced in Logistic Regression, SVM utilizes a different cost function and optimization approach, that enable the model to become non-parametric and detect non-linearly relations using a kernel function, this paper utilizes the RBF (Radial Basis Function) kernel.

2.3.3 Random Forest

Random Forest is an ensemble learning method that combines multiple decision trees to make predictions in classification and regression tasks (Géron, 2022). It utilizes bagging, which involves creating subsets of the training data and constructing decision trees independently on each subset. The final prediction is determined by aggregating the results from all the decision trees. Random Forest addresses overfitting by considering random subsets of the training data and features during tree construction, the Random Forest structure is represented on Fig. 6.

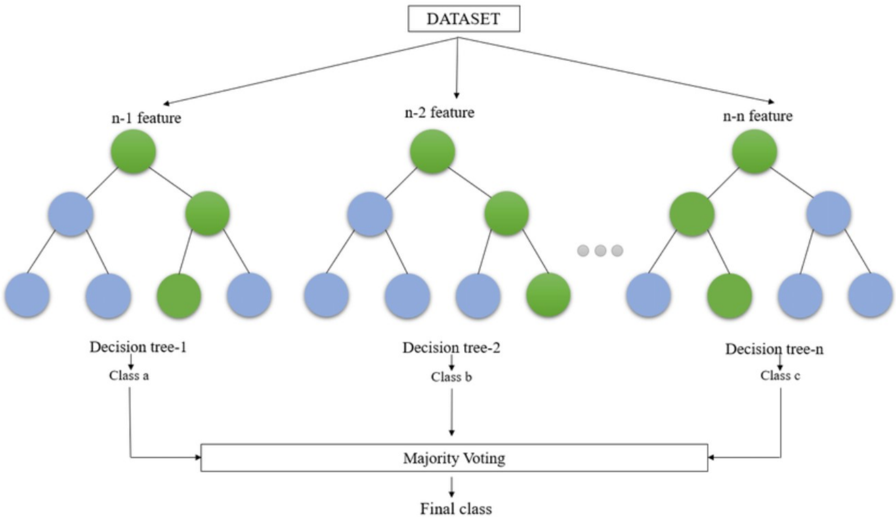

Figure 6. Random Forest structure (Sharma, 2020).

2.3.4 Hyperparameter Tuning

Hyperparameter tuning is a crucial step in machine learning model development, as it involves finding the optimal configuration of hyperparameters to maximize performance. Hyperparameters are adjustable parameters that determine the behavior and flexibility of a model, such as learning rate, regularization strength, and the number of estimators. Selecting appropriate hyperparameter values can significantly impact the model's accuracy and generalization capability.

To ensure rigorous evaluation of the models, a repeated stratified k-fold cross-validation strategy was employed on the training set. This strategy involved repeating the k-fold cross-validation process multiple times, with each repetition using a different random seed. In this study, the repeated stratified k-fold cross-validation was performed three times, with three folds in each repetition. This approach helped reduce the potential bias introduced by a single random split and provided a more comprehensive assessment of the models performance across different training and validation subsets.

To systematically search for the best hyperparameter values, the study employed the TPE (Tree-structured Parzen Estimator) (Bergstra *et al.*, 2011) algorithm implemented in the Optuna library (Akiba *et al.*, 2019). Optuna provides an efficient and automated approach for hyperparameter optimization by constructing a search space and iteratively sampling promising hyperparameter configurations based on the objective function's performance. Table 2 presents the hyperparameters tested and their corresponding search spaces for each model.

Table 2. Hyperparameters for different models.

Model	Hyperparameter	Search Space Values
Logistic Regression	C	$10^{-6} : 10^{6}$
Support Vector Machines	C	$10^{-6} : 10^{6}$
	Kernel	RBF
	Kernel coefficient (γ)	$10^{-6} : 10^{6}$
Random Forest	Max Depth	$1 : 64$
	Max Features	None, sqrt, log2
	Min Samples Split	$2 : 16$
	Min Samples Leaf	$1 : 8$
	CCP Alpha	$10^{-3} : 1$
	Min Impurity Decrease	$10^{-3} : 1$
	N Estimators	100

2.4 Evaluation Metrics

The evaluation of fault detection and diagnosis models requires the use of appropriate metrics to assess their performance. In this subsection, we discuss two key evaluation metrics employed in our study: Balanced Accuracy and the Receiver Operating Characteristic (ROC) curve.

Balanced Accuracy is a metric that takes into account class imbalance in the dataset, providing a balanced measure of overall model performance. It calculates the average accuracy across all classes (Eq. 6), ensuring that the impact of both majority and minority classes is properly considered.

$$BA = \frac{1}{2}(\frac{TP}{TP+FN} + \frac{TN}{TN+FP}) \tag{6}$$

where TP represents the number of true positives, FN represents the number of false negatives, TN represents the number of true negatives, and FP represents the number of false positives.

The ROC curve is a widely used graphical tool for evaluating binary classification models. It illustrates the trade-off between the true positive rate (TPR) (Eq. 7) and the false positive rate (FPR) (Eq. 8) at different classification thresholds. The curve's shape and position provide insights into the model's ability to correctly identify positive instances while minimizing false positives. The area under the ROC curve (AUC) quantifies the overall discriminative power of the model, with a higher AUC indicating better performance.

$$TPR = \frac{TP}{(TP+FN)} \tag{7}$$

$$FPR = \frac{FP}{(FP+TN)} \tag{8}$$

3. RESULTS AND DISCUSSION

In this section, we present the results and discussion of our fault detection and diagnosis models for bearing analysis. We evaluated two approaches: the Baseline model, which relies on rule-based techniques, and using classical machine learning models. For each approach, we analyzed their performance using key evaluation metrics such as Balanced Accuracy, TPR, FPR and ROC curves.

Machine learning models offer the advantage of having a decision function, allowing for the determination of a decision threshold in the ROC curve analysis. This threshold is critical as it determines the point at which instances are classified as positive or negative. This flexibility enables the models to optimize the trade-off between the TPR and FPR, resulting in improved performance. In contrast, the baseline model lacks this decision function, limiting it to a fixed operating point on the ROC curve. This limitation restricts the baseline model's ability to adapt the decision threshold and fine-tune the TPR and FPR, resulting in reduced flexibility and potentially impacting its overall performance on the ROC curve.

3.1 Baseline Model

The baseline approach was evaluated using 4 filter ranges for each fault type, as described on section 2.2. The achieved TPR, FPR, and Balanced Accuracy for each filter range on each fault are presented in Table 3.

For the inner fault, the best performance was observed in the filter range of 500-10000 Hz, with a TPR of 0.902 and an FPR of 0.148, resulting in a balanced accuracy of 0.877. Similarly, for the outer fault, the filter range of 5000-12600 Hz showed the highest TPR of 0.854 and a relatively lower FPR of 0.231, leading to a balanced accuracy of 0.811. However, for the ball fault, the baseline approach exhibited relatively lower performance across all filter ranges, with the filter range of 5000-12600 Hz achieving the highest TPR of 0.634.

Overall, the baseline approach demonstrated varying levels of success in detecting different fault types. While it achieved relatively high TPR values for inner and outer faults, indicating effective fault detection, the performance was relatively poorer for the Ball fault. The achieved balanced accuracy values also reflected these observations. These results provide valuable insights into the effectiveness of the baseline approach and set the foundation for further analysis and comparison with the machine learning models.

3.2 Machine Learning Models

The performance of the machine learning models was evaluated and compared to the baseline approach for fault detection. ROC curves were generated to visually assess the models performance on the Fig. 7. The baseline was represented by the filter range that achieved the highest balanced accuracy among all evaluated filter ranges. By selecting

Table 3. Baseline results.

Fault	Filter range	TPR	FPR	Balanced Accuracy
Inner	50-1000 Hz	0.780	0.349	0.716
	500-10000 Hz	0.902	0.148	0.877
	5000-12600 Hz	0.829	0.201	0.814
	Kurtogram	0.732	0.320	0.706
Outer	50-1000 Hz	0.854	0.308	0.773
	500-10000 Hz	0.829	0.290	0.770
	5000-12600 Hz	0.854	0.231	0.811
	Kurtogram	0.829	0.320	0.755
Ball	50-1000 Hz	0.610	0.574	0.518
	500-10000	0.463	0.331	0.566
	5000-12600 Hz	0.634	0.331	0.651
	Kurtogram	0.610	0.355	0.627

the most accurate filter range as the baseline, a benchmark was established for comparing the performance of the machine learning models.

For the inner fault, logistic regression exhibited superior performance compared to the baseline, outperforming the other models. Followed by SVM, which demonstrated slightly lower performance. Random Forest did not surpass the performance of the baseline. It is worth investigating the impact of hyperparameter tuning on the performance of Random Forest.

Regarding the outer fault, logistic regression emerged as the top-performing model, surpassing both the baseline and the other models in terms of fault detection. SVM also demonstrated improved performance compared to the baseline, making it a viable alternative for fault detection in this category. However, Random Forest exhibited relatively poorer performance compared to the other models, suggesting that it may not be the most suitable choice for fault detection in this specific fault type.

For the ball fault, Random Forest showcased the best performance among the evaluated models, surpassing both logistic regression and SVM. It demonstrated superior effectiveness in detecting Ball faults. Logistic regression and SVM also exhibited improved performance compared to the baseline, underscoring their efficacy in fault detection for this fault type.

In the legend of the ROC plot, logistic regression is represented by the blue curve, SVM by the orange curve, and Random Forest by the green curve. The red point symbolizes the baseline results, providing a visual reference for comparative analysis of the models performance against the baseline.

These findings highlight the potential of machine learning models, particularly logistic regression, SVM, and Random Forest, in enhancing fault detection compared to the baseline approach.

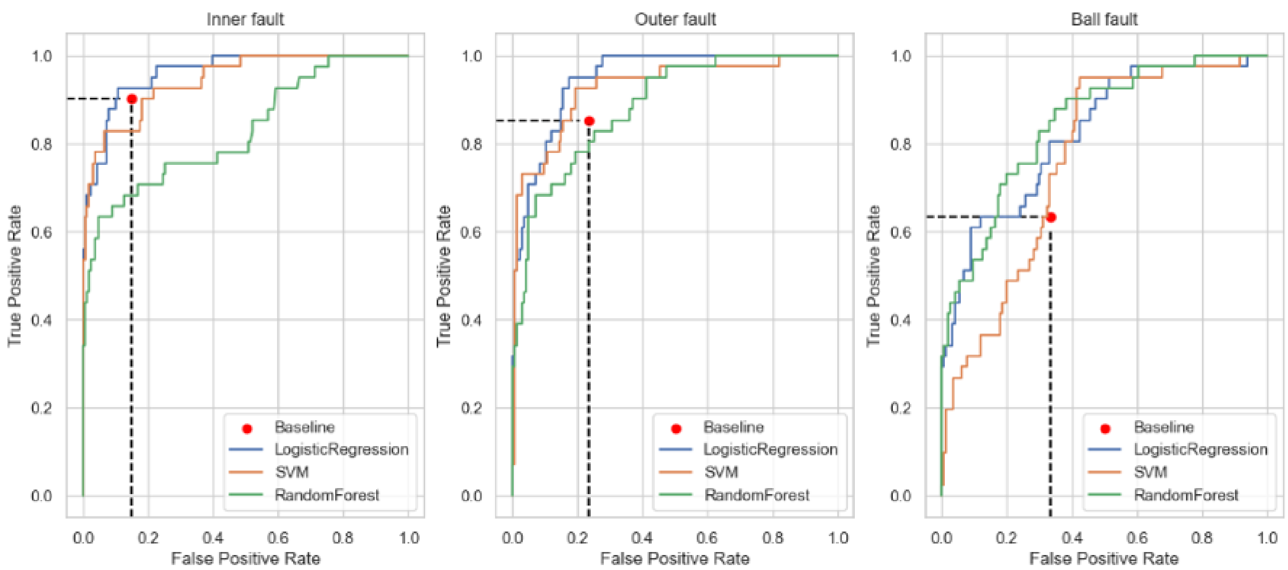

Figure 7. Models results.

4. CONCLUSION

In this study, the performance of a baseline approach and machine learning models was evaluated for fault detection in rotating machinery. The baseline approach, which utilized different filter ranges for each fault type, demonstrated varying levels of success in detecting different faults. For each fault type, the ROC curve was assessed to evaluate the true positive rate (TPR) and false positive rate (FPR) of each model. This allowed for a comprehensive performance evaluation by analyzing the ability of the models to accurately classify fault and non-fault instances across different fault types.

Moreover, the machine learning models demonstrated promising results in improving fault detection compared to the baseline. Logistic regression emerged as the best-performing model for the Inner and Outer faults, surpassing the baseline and demonstrating superior performance. SVM also showed improved performance compared to the baseline for both fault types. However, Random Forest exhibited relatively poorer performance compared to the other models.

The poor performance in detecting rolling element faults can be attributed to the limited consideration of additional fault-related attributes, such as the sidebands generated by the FTF. These sidebands, resulting from the interaction between the rolling elements and the cage, can provide valuable information for the detection and diagnosis of rolling element faults.

These findings underscore the potential of machine learning models, particularly logistic regression, SVM, and Random Forest, in enhancing fault detection compared to the baseline approach. The selection of appropriate models based on fault characteristics and requirements is crucial for achieving optimal performance. Future research should focus on further refining the models and investigating the impact of hyperparameter tuning on their performance.

Overall, this study contributes to the field of fault detection in rotating machinery by providing insights into the performance of the baseline approach and machine learning models. The findings can guide practitioners in selecting effective approaches for fault detection and improving the reliability and efficiency of rotating machinery systems.

5. REFERENCES

Ahmed, H. and Nandi, A.K., 2020. *Condition monitoring with vibration signals: Compressive sampling and learning algorithms for rotating machines*. John Wiley & Sons.

Akiba, T., Sano, S., Yanase, T., Ohta, T. and Koyama, M., 2019. "Optuna: A next-generation hyperparameter optimization framework". In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. New York, NY, USA.

Antoni, J. and Randall, R., 2006. "The spectral kurtosis: application to the vibratory surveillance and diagnostics of rotating machines". *Mechanical Systems and Signal Processing*, Vol. 20, No. 2, pp. 308–331.

Bellini, A., Immovilli, F., Rubini, R. and Tassoni, C., 2008. "Diagnosis of bearing faults of induction machines by vibration or current signals: A critical comparison". In *2008 IEEE Industry Applications Society Annual Meeting*. pp. 1–8. doi:10.1109/08IAS.2008.26.

Bergstra, J., Bardenet, R., Bengio, Y. and Kégl, B., 2011. "Algorithms for hyper-parameter optimization". In J. Shawe-Taylor, R. Zemel, P. Bartlett, F. Pereira and K. Weinberger, eds., *Advances in Neural Information Processing Systems*. Curran Associates, Inc., Vol. 24.

Géron, A., 2022. *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow*. O'Reilly Media, Inc.

Lei, Y., Yang, B., Jiang, X., Jia, F., Li, N. and Nandi, A.K., 2020. "Applications of machine learning to machine fault diagnosis: A review and roadmap". *Mechanical Systems and Signal Processing*, Vol. 138, p. 106587.

McFadden, P. and Smith, J., 1984. "Model for the vibration produced by a single point defect in a rolling element bearing". *Journal of Sound and Vibration*, Vol. 96, No. 1, pp. 69–82.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M. and Duchesnay, E., 2011. "Scikit-learn: Machine learning in Python". *Journal of Machine Learning Research*, Vol. 12, pp. 2825–2830.

Randall, R.B., 2021. *Vibration-based Condition Monitoring: Industrial, Automotive and Aerospace Applications*. Wiley, 2nd edition.

Sharma, A., 2020. "Random forest vs decision tree | which is right for you?" Analytics Vidhya, https://www.analyticsvidhya.com/blog/2020/05/decision-tree-vs-random-forest-algorithm/.

Watt, J., Borhani, R. and Katsaggelos, A.K., 2020. *Machine Learning Refined: Foundations, Algorithms, and Applications*. Cambridge University Press, 2nd edition. doi:10.1017/9781108690935.

6. RESPONSIBILITY NOTICE