

Implementation and Evaluation of Machine Learning Algorithms in Ball Bearing Fault Detection

Pavle Stepanić¹, Nedeljko Dučić², Jelena Vidaković¹, Jelena Baralić^{2*}, Marko Popović²

¹Research and Development Institute Lola L.t.d., Kneza Visaslava, 70A, 11030, Belgrade, Serbia, pavle.stepanic@li.rs, jelena.vidakovic@li.rs

²Faculty of Technical Sciences Čačak, University of Kragujevac, Svetog Save 65, 32102, Čačak, Serbia, nedeljko.ducic@ftn.kg.ac.rs, jelena.baralic@ftn.kg.ac.rs, marko.popovic@ftn.kg.ac.rs

Abstract: The subject of this research is the development of a classifier based on machine learning (ML) that is able to recognize defective and healthy ball bearings. For this purpose, vibration measurements were performed on the bearings, on a total of 196 samples. For each recorded vibration signal, a feature extraction was performed by digital processing in the time domain. The following ML algorithms were used to develop the classifier: K-nearest neighbor (KNN) and support vector machine (SVM) as well as improved versions of the aforementioned algorithms. Improved versions of the mentioned algorithms were obtained by optimizing their hyperparameters. The corresponding models of the KNN and SVM algorithms showed a high percentage of success in classification, 98.5 % and 99.5 %, respectively. By optimizing the hyperparameters, models with a maximum classification success of 100 % were achieved.

Keywords: vibration measurement, ball bearings, machine learning, fault detection

1. INTRODUCTION

Monitoring and predicting the condition of industrial systems and processes during operation has increasingly become the subject of intensive research in recent years, as it serves as the basis for the Condition-Based Maintenance concept and the Predictive Maintenance concept, which are integral concepts of Industry 4.0 [1]. The early detection of malfunctions allows operators to rectify the problem as part of regular maintenance and thus avoid additional, unplanned, and often very costly downtime due to sudden failures. Early fault detection implies a sufficiently long warning time, and thus more time for appropriate corrective action and rational planning of system repair. This directly translates to cost and time savings while increasing the productivity of the entire process.

When investigating the most common failures of electrical equipment in industrial processes and electromotor drives [2], it was found that significant production losses are caused by faults in electric motors. Various factors such as environmental influences (dust, temperature, vibrations), production or design errors, improper installation, incorrect use, wear due to long-term operation, abrasion, erosion, material aging, etc. can influence motor failure. Detection of motor components that most frequently fail varies from study to study, but three main categories stand out: stator failure,

rotor failure, and bearing failure [3], [4]. Several studies [5]-[8] have concluded that the majority of electrical machine failures are related to bearing failures.

Bearings are critical elements in machinery as they support rotating structures and much of the energy is transmitted through bearings during operation. Bearing failures lead to other major and more expensive machine failures. Therefore, bearings in most machines are regularly maintained and replaced to prevent major breakdowns. On the other hand, frequent, unnecessary disassembly and assembly of electrical machines is also a major cause of failures. The advantages of the condition-based maintenance concept for bearings, which is based on the monitoring and control of selected condition parameters (e.g., vibrations and sound) and only intervenes if the condition parameters exceed defined limit values, are therefore extremely beneficial for increasing the productivity of the industrial plant. In this way, the effective lifetime of the entire system is increased and unnecessary downtime is avoided.

As bearing failures are in most cases associated with increased vibration levels, vibration monitoring is crucial for the detection of bearing failures. However, the choice of methods for vibration monitoring and the probability of fault detection depends largely on the chosen vibration signal processing technique. Vibration signals are typically noisy

due to interference from other vibration sources. One of the key steps in acquiring fault information is to increase the signal-to-noise ratio, which is difficult to achieve without processing vibration signals [9].

From a mathematical point of view, bearing fault detection is a process in which data from the measurement space and/or characteristic features are mapped into the fault space. The mapping process actually represents a problem of pattern recognition and classification [10]-[12]. For intelligent fault diagnosis methods to be widely used in industrial machinery, the applied techniques must be general and the system must be adjustable, calibrated, and autonomous. The integration of the mentioned features is a significant challenge when implementing intelligent fault diagnosis systems in plants. In the last thirty years, with the development of computer and information technologies, various methods for fault diagnosis have been developed, which can be categorized into three groups [13]-[19]:

- statistical approach;
- application of artificial intelligence;
- model-based techniques.

Two approaches can be recognized in the application of artificial intelligence techniques: the application of classical machine learning (ML) algorithms and deep learning (DL) algorithms. Classical ML methods have been used successfully for decades to detect and categorize bearing faults. Recent developments in DL algorithms over the last ten years have demonstrated the superiority of DL-based methods in the fault feature extraction and classification. Most of the literature applying ML algorithms reports satisfactory results with classification accuracy above 90 %. To achieve even better performance in versatile operating conditions and noisy environments, DL-based methods are becoming increasingly popular to meet this requirement. The effectiveness of ML and DL algorithms in bearing fault diagnostics relies on the strong correlation of the extracted features with the underlying physical laws. Both approaches (application of ML or DL) in research are justified and depend on the problem under consideration. To optimize the selection of algorithms for bearing fault detection, the authors propose a structured approach consisting of three key considerations [20]:

Setup environment: A comprehensive analysis of the operating conditions is essential. In controlled indoor environments with fixed operating points, traditional ML techniques or frequency-based analytical models may suffice. However, in more complex scenarios with variable speeds, loads, or external disturbances (e.g., motors with variable frequency drives in electric vehicles), advanced DL techniques should be used. When operating in noisy environments with low signal-to-noise ratio (SNR), additional denoising blocks and hidden layers can improve the robustness of deep neural networks.

Sensor selection: The number and type of sensors has a significant impact on the effectiveness of fault detection. Classical ML methods usually require one or two vibration sensors close to the bearing. In contrast, DL approaches, especially convolutional neural networks (CNNs), require the

transformation of 1D time-series sensor data into a 2D format, which often requires multiple sensors or preprocessing techniques such as wavelet packet decomposition (WPD). In addition, incorporating multi-physics sensor data (e.g., acoustic emissions, stator current) can improve classification accuracy, especially under dynamic operating conditions.

Data size and quality: A thorough understanding of environmental conditions and operational variations is crucial for data collection. Similar to the setup environment considerations, data collected under simple, controlled conditions can be effectively analyzed using classical ML models. Conversely, datasets characterized by significant noise, variations in operating conditions, or abrupt changes require robust DL architectures with sufficient data augmentation and noise-resistant features.

By systematically considering these factors, engineers and researchers can make informed decisions about the optimal ML/DL approach for bearing fault diagnosis, ensuring higher accuracy and reliability in real-world applications.

The success of a model in performing the task for which it was developed is its most important characteristic. Therefore, there is a tendency to continuously improve models. Research on the application of classical ML algorithms in bearing fault diagnosis shows a trend towards the integration of advanced feature extraction methods (e.g., wavelet transforms, fast fourier transform (FFT) and optimization algorithms (e.g., genetic algorithms (GA), principal component analysis (PCA)) to improve model performance. Table 1 provides an overview of some studies of this type, with emphasis on success in performing the task.

Zhang et al. (2020) presented a valuable study in which they systematically summarized the existing literature on bearing fault diagnosis using DL algorithms [20]. Their extensive analysis provides a detailed overview of the applications of DL algorithms and their success rates, reaching about 99 % in some studies.

In this paper, the results of using ML algorithms for to detect ball bearing faults by analyzing vibration signals are presented. In contrast to the analyzed and listed research approaches (ML, DL, hybrid ML models), this study focuses on hyperparameter optimization of ML algorithms to improve classical ML models. This approach to the detection of ball bearing faults represents a novel contribution to research circles. Of particular value in this study are the results of the optimized ML model, which shows maximum performance in detecting ball bearing failures. The paper is divided into five sections. Section 2 describes the experimental part, i.e. the method of vibration signal acquisition using the developed ball bearing test bench. The digital processing of the collected data is described, which is done by extracting suitable statistical parameters in the time domain. The operating principle of the ML algorithms: K-nearest neighbor (KNN) and support vector machine (SVM) for fault detection are described in Section 3. The results of the bearing classification into healthy and defective are presented. Section 4 presents the optimization of the hyperparameters of the ML algorithms to improve the performance. Section 5 contains concluding remarks.

Table 1. Hybrid ML models.

Applied ML model	Research description
SVM with GA	A study applied SVM combined with GA to develop optimal classifiers for distinguishing healthy and faulty bearings in ASD systems, achieving 97.5 % accuracy [21].
SVM and ANN with CWT	This study explored the use of SVM and ANN alongside CWT to analyze frame vibrations during motor start-up, achieving 96.67 % accuracy with SVM and 90 % with ANN [22].
PCA and SVDD	PCA and SVDD were used to predict bearing failures, achieving 93.45 % accuracy [23].
GA-based SVM	A GA-based kernel discriminative feature analysis was combined with one-against-all multicategory SVMs (OAA MCSVMs) for fault diagnosis in low-speed bearings, achieving the highest reported accuracy of 98.66 % [24].
FEM and WPT with SVM	A hybrid approach integrating FEM, WPT, and SVM was proposed for fault classification, achieving 81 % accuracy for inner race faults and 79 % for rolling body faults [25].
FFT-based feature extraction with SVM	The frequency domain features derived from FFT were used to train an SVM model for bearing fault classification, achieving 87.35 % accuracy [26].

Note:

SVM - support vector machines
 GA - genetic algorithm
 ASD - adjustable speed drive
 ANN - artificial neural networks
 CWT - continuous wavelet transform
 PCA - principal component analysis
 SVDD - support vector data description
 FEM - finite element method
 OAA - one-against-all
 MCSVM - multicategory SVM
 WPT - wavelet packet transform
 FFT - fast fourier transform

2. EXPERIMENTAL SETUP AND DATA COLLECTION

Fig. 1 shows the system used to detect and record rolling bearing vibrations. The vibrations are recorded using two sensors mounted radially on the bearing housing (sensor 1 and sensor 2 placed at a 90° angle). The sensors are piezoelectric accelerometers manufactured by Brüel & Kjær. Low-level signals from the accelerometers are fed through preamplifiers (Charge Amplifier Type 2635, Brüel & Kjær). The outputs of the preamplifiers were connected to two channels of the acquisition and recording system (TEAC LX-10 Recording Unit). The sampling frequency per channel was set to 96 kHz. The LX-10 was connected to a PC on which the parameters of the TEAC were set during recording using special software installed on the computer (LX Navi). The vibrations were measured at a shaft rotation speed of 1518 rpm or 25.3 Hz. The collected digital signals were recorded on a PC and then processed using standard digital signal processing tools.

By recording vibration signals, a database of damaged and healthy bearings, i.e. the training set, is formed. An experimental recording of the vibration signals of 196 bearings was performed. Of the 66 undamaged bearings, 44 bearings were new, while 22 healthy bearings were already in use. The remaining 130 bearings were damaged. The vibrations of each bearing were recorded simultaneously over a period of 10 seconds at a sampling frequency of 96 kHz using two radially arranged sensors.

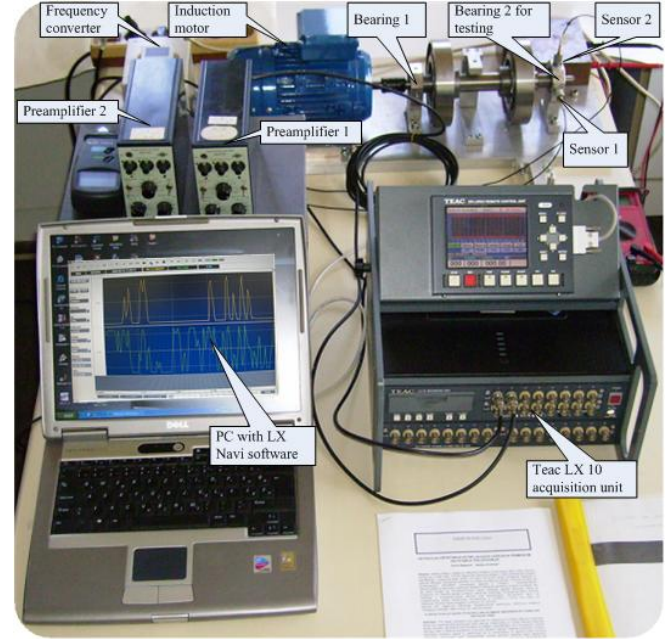


Fig. 1. The developed ball bearing test station.

Next, feature extraction was performed for each recorded vibration signal by digital processing in the time domain. The following statistical parameters were used to describe the vibration signals:

1. Arithmetic mean

$$\bar{x} = \frac{1}{N} \sum_{i=1}^N x(i)$$

2. Root mean square

$$x_{rms} = \left(\frac{1}{N} \sum_{i=1}^N x^2(i) \right)^{1/2}$$

3. Modified square mean

$$x_r = \left(\frac{1}{N} \sum_{i=1}^N |x(i)|^{1/2} \right)^2$$

4. Skewness index

$$x_{ske} = \frac{\frac{1}{N} \sum_{i=1}^N (x(i) - \bar{x})^3}{x_v^{3/2}}, \text{ where } x_v \text{ variance}$$

$$x_v = \frac{1}{N} \sum_{i=1}^N (x(i) - \bar{x})^2$$

5. Kurtosis index

$$x_{kur} = \frac{\frac{1}{N} \sum_{i=1}^N (x(i) - \bar{x})^4}{x_v^2}$$

6. C factor

$$C = \frac{PP}{x_{rms}}, \text{ where } PP \text{ is peak-to-peak value}$$

$$PP = \max_i(x(i)) - \min_i(x(i))$$

7. L factor

$$L = \frac{PP}{x_r}$$

8. S factor

$$S = \frac{x_{rms}}{\bar{x}}$$

9. I factor

$$I = \frac{PP}{\bar{x}}$$

where $x(i)$ is the vibration signal sample at the discrete time $i\Delta t$, and $\Delta t = 1/96 \text{ kHz}$.

Based on the aforementioned statistical parameters, a 9-dimensional feature vector is formed to represent the characteristics of the recorded vibration signals of bearings. The nine extracted features thus serve as inputs, while the classification into healthy bearings or bearings with faults represents the outputs for the developed ML-based algorithms.

3. MACHINE LEARNING ALGORITHM FOR CLASSIFICATION

The world of ML is vast and includes many techniques and algorithms that can be used in various applications. The present research problem demonstrates the need to select ML algorithms designed for classification that belong to the category of supervised learning, which implies the need for an adequate dataset. The data presented in Section 2 were processed using two ML algorithms: KNN and SVM.

A. K-nearest neighbor

The KNN algorithm is a popular and simple algorithm for supervised learning. KNN classifies an unknown instance by finding the k instances from the training set that are closest to this instance with respect to a selected metric, and assigns it the class that is most common among the mentioned k instances. Let X denote the matrix of instances from the training set and Y denotes the matrix of instances to be classified. These matrices are treated as row vectors x_1, x_2, \dots, x_n and y_1, y_2, \dots, y_n , with the distances between the vectors x_s and y_t defined in several different ways. Below are the relations used to calculate the distance metric:

- Euclidean distance: $d_{st}^2 = (x_s - y_t)(x_s - y_t)'$
- Minkowski distance: $d_{st} = \sqrt[p]{\sum_{j=1}^n |x_{sj} - y_{tj}|^p}$.
- Cosine distance: $d_{st} = \left(1 - \frac{x_s y_t'}{\sqrt{(x_s x_s') (y_t y_t')}}\right)$

Several variants of the KNN algorithm were used for the classification of ball bearings. Table 2 shows all variants of the KNN algorithms used and their success rate in classifying the tested ball bearings.

Table 2. The k-nearest neighbor (KNN) classification.

Model No	Model name	Distance metric	Distance weight	Number of neighbors	Classification success rate
1	Cosine KNN	Cosine	Equal	10	98.5 %
2	Coarse KNN	Euclidean	Equal	100	74 %
3	Fine KNN	Euclidean	Equal	1	97.4 %
4	Weighted KNN	Euclidean	Squared inverse	10	97.4 %
5	Medium KNN	Euclidean	Equal	1	98 %
6	Cubic KNN	Minkowski	Equal	10	98.2 %

B. Support vector machine

SVM is one of the key methods of ML. It is based on a clear geometric intuition. The SVM algorithm works by creating a decision boundary or hyperplane between the data to form different classes. The main problem of the SVM algorithm is that the hyperplane may not be an appropriate boundary between the classes. In practice, the boundary shapes may be arbitrary. Moreover, the soft margin formulation does not solve this problem, since the soft margin assumes that the hyperplane represents the shape of the boundary approximately correctly, but the data sometimes ends up on the wrong side. It may be necessary for the boundary to be circular. In this case, the straight line is clearly not a good approximation. Therefore, kernel functions are needed to determine different shapes of hyperplanes

according to the data arrangement. Kernel functions map the data to a different, often higher dimensional space, with the expectation that the classes will be easier to separate after this transformation and complex non-linear decision boundaries may be simplified to linear boundaries in the higher dimensional mapped feature space. The kernel functions used for the development of SVM classification models are given below:

- Radial basis function (RBF):

$$K(x_1, x_2) = \exp\left(-\frac{\|x_1 - x_2\|^2}{2\sigma^2}\right).$$
- Linear: $K(x_1, x_2) = x_1^T x_2$.
- Polynomial: $K(x_1, x_2) = (x_1^T x_2 + 1)^\rho$.

Three types of SVM algorithms were used for the classification of ball bearings. Table 3 shows the variants of the SVM algorithms used and their success rate in classifying the tested ball bearings.

Table 3. SVM classification.

Model No	Kernel function	Classification success rate
1	Linear	93.9 %
2	Polynomial ($\rho = 2$)	99.5 %
3	RBF	99 %

Note: RBF - radial basis function

C. Performance evaluation of machine learning algorithms

The choice of quality measures for the models depends on the type of problem to be solved and the desired results. In this case, general performance measures such as the Confusion Matrix and the receiver operating characteristic (ROC) curve, are selected for the developed models (Fig. 2 and Fig. 3).

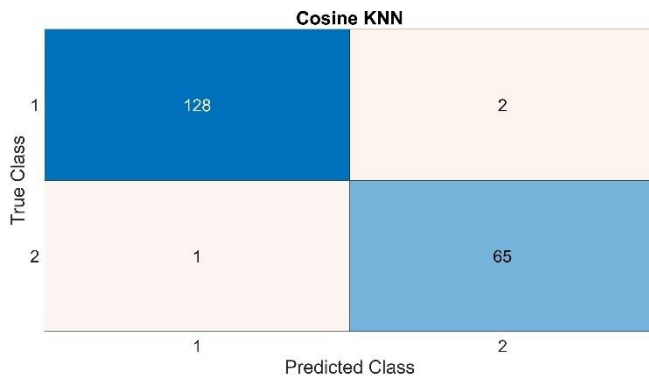


Fig. 2. Confusion matrix for the cosine KNN algorithm.

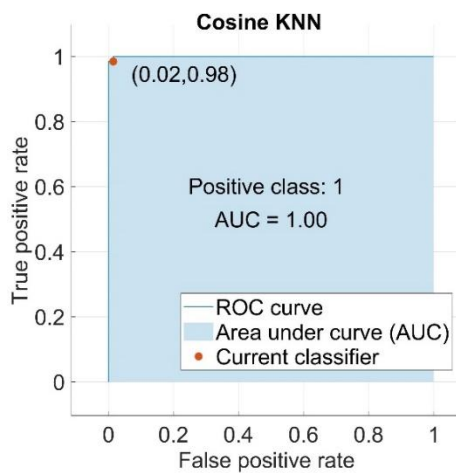


Fig. 3. ROC curve for the cosine KNN algorithm.

All KNN models achieved high classification success rates, with the highest success rate of 98.5 % achieved by the cosine KNN model. The cosine KNN model calculates the distance metric using cosine with a number of 10 neighbors. The confusion matrix criterion showed that the cosine KNN model had 128 successful and 2 unsuccessful classifications for the first class (damaged ball bearings) and 65 successful

and 1 unsuccessful classification for the second class (healthy ball bearings). The ROC curve is a metric that is additionally used as a measure of the classifier quality. Two values are calculated for each threshold: the true positive ratio (TPR) and the false positive ratio (FPR). For a given class i , the TPR is the number of outputs whose actual and predicted class is class i divided by the number of outputs whose actual class is class i . The FPR is the number of outputs whose actual class is not class i , but whose predicted class is class i , divided by the number of outputs whose actual class is not class i .

Similar to the KNN algorithms, the developed SVM models have also shown high classification success rates. The best performing SVM model is the SVM algorithm with a second-order polynomial kernel function, which achieves a success rate of 99.5 %. The aforementioned model provided only one incorrect classification, which belonged to the first class (damaged ball bearings) (Fig. 4 and Fig. 5).

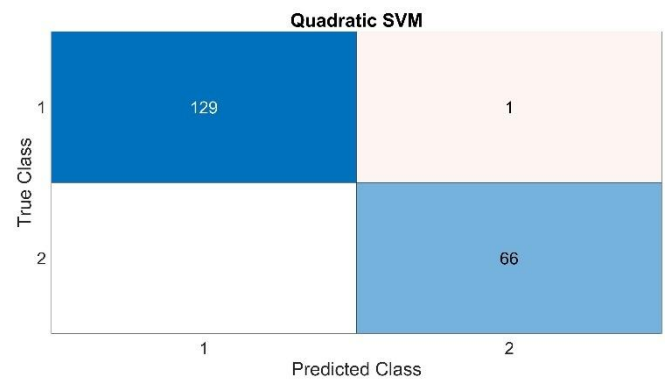


Fig. 4. Confusion matrix for the quadratic SVM algorithm.

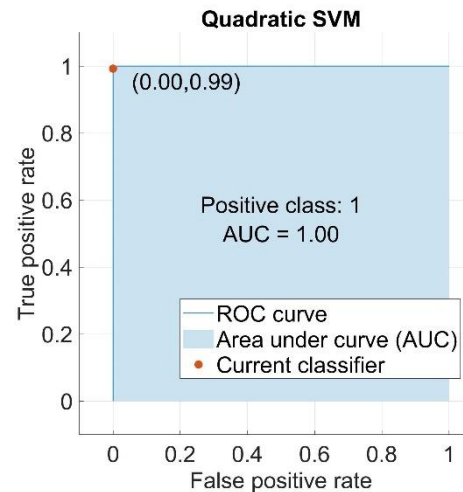


Fig. 5. ROC curve for the quadratic SVM algorithm.

4. IMPROVED MACHINE LEARNING ALGORITHM FOR CLASSIFICATION

Although the ML algorithms presented in Section 3 have shown satisfactory performance in solving the classification problem, possible improvements are considered here. ML algorithms have hyperparameters that can be adjusted to maximize the performance of the developed model. Hyperparameter optimization has become increasingly

prevalent in both scientific and commercial approaches, with the same goal: to obtain a more accurate model. The adjustment of hyperparameters in ML algorithms belongs to the category of optimization problems. Research in hyperparameter optimization has a relatively long history, spanning more than 30 years [27]-[30], and it has been shown that different configurations of hyperparameters give the best results for different datasets [28].

A. Formulation of the optimization problem

The first step in formulating the optimization problem is to define the participants in the optimization process. The ML algorithm with N hyperparameters is denoted by the variable \mathcal{A} . The domain of the N^{th} hyperparameter is Λ_N , and the entire configuration space of the hyperparameters is denoted by $\Lambda = \Lambda_1 \times \Lambda_2 \times \dots \times \Lambda_N$. The hyperparameter vector is labeled as $\lambda \in \Lambda$, and the ML algorithm containing these hyperparameters is denoted by \mathcal{A}_λ . The hyperparameters can be of different types (real values, e.g., learning rate; binary values, e.g., whether to use early stopping or not; integers, e.g., number of neighbors; categories, e.g., type of optimization algorithm). For a given dataset D , the objective of the optimization problem is defined by the relation [31].

$$\lambda^* = \underset{\mathcal{A} = \Lambda}{\operatorname{argmin}} \mathbb{E}_{(D_{\text{train}}, D_{\text{valid}}) \sim \mathcal{D}} V(\mathcal{L}, \mathcal{A}_\lambda, D_{\text{train}}, D_{\text{valid}}) \quad (1)$$

where $V(\mathcal{L}, \mathcal{A}_\lambda, D_{\text{train}}, D_{\text{valid}})$ measures the loss of a model generated by algorithm \mathcal{A} with hyperparameters λ on training data D_{train} and evaluated on validation data D_{valid} .

B. Bayesian hyperparameter optimization

Hyperparameter optimization of ML algorithms can be performed with any black-box optimization method. Global optimization algorithms are preferred due to the non-convex nature of the optimization process. Bayesian optimization is a global optimization method that is very popular in hyperparameter optimization of ML algorithms. Shahriari, Brochu et al. have presented the mechanisms of Bayesian optimization in [32], [33]. Bayesian optimization is a process that iterates with two key segments: a probabilistic surrogate model and an acquisition function. The probabilistic surrogate model represents the objective function, while the acquisition function (2) uses the predictive distribution of the probabilistic surrogate model to determine the utility of the different candidate points:

$$\mathbb{E}[\mathbb{I}(\lambda)] = (f_{\min} - \mu(\lambda))\Phi\left(\frac{f_{\min} - \mu(\lambda)}{\sigma}\right) + \sigma\phi\left(\frac{f_{\min} - \mu(\lambda)}{\sigma}\right) \quad (2)$$

where $\Phi(\cdot)$ and $\phi(\cdot)$ are the standard normal density and the standard normal distribution function, and f_{\min} is the best-observed value so far. Gaussian processes are used to generate the objective function due to their good characteristics in terms of uncertainty estimation and closed-form prediction distribution:

$$\mu(\lambda) = k_*^T K^{-1} y, \sigma^2(\lambda) = k(\lambda, \lambda) - k_*^T K^{-1} k_* \quad (3)$$

where k_* denotes the vector of covariances between λ and all previous observations, K is the covariance matrix of all previously evaluated configurations, and y are the observed function values. The quality of the Gaussian process depends exclusively on the covariance function.

C. Optimized machine learning algorithms

The hyperparameter optimization was performed for the algorithms used in Section 3, i.e. for the SVM and KNN algorithms. The optimization of both types of algorithms resulted in models with 100 % accuracy. It can be concluded that the optimized models all contained correct classifications. The optimization of the SVM algorithm was performed using the presented Bayesian hyperparameter optimization. Table 4 shows the ranges of the parameters being optimized.

Table 4. SVM hyperparameter search range.

Hyperparameter	Range
Box constraint level	0.001-1000
Kernel scale	0.001-1000
Kernel function	Gaussian, Linear, Quadratic, Cubic
Standardize data	true, false

After 78 seconds and 30 iterations, the optimization process was stopped, as shown in Fig. 6. The optimization results as well as the characteristics of the optimized SVM model can be found in Table 5.

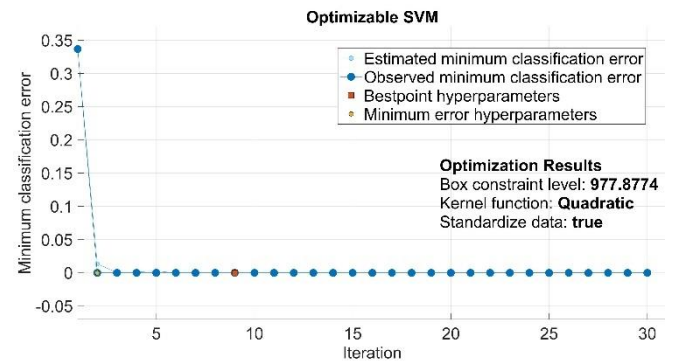


Fig. 6. The optimization flow for the SVM algorithm.

Table 5. The optimized hyperparameters of the SVM algorithm.

Hyperparameter	Value
Box constraint level	977.88
Kernel scale	1
Kernel function	Quadratic
Standardize data	true
Accuracy	100 %

The optimization of the KNN algorithm was also performed using the presented Bayesian hyperparameter optimization. Table 6 contains the ranges of the parameters being optimized.

Table 6. KNN hyperparameter search range.

Hyperparameter	Range
Number of neighbors	1-98
Distance metric	Euclidean, Cosine, Euclidean, Correlation, Chebyshev, Hamming, Minakowski, Spearman, Jaccard, City block, Mahalanobis
Distance weight	Equal, Inverse, Squared, Inverse
Standardize data	true, false

After about 35 seconds and 30 iterations, the optimization process is stopped, as shown in Fig. 7. The optimization results and the characteristics of the optimized KNN model are listed in Table 7.

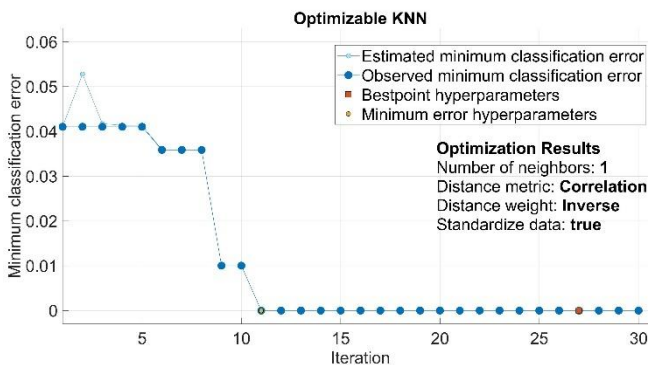


Fig. 7. The optimization flow for the KNN algorithm.

Table 7. The optimized hyperparameters of the KNN algorithm.

Hyperparameter	Range
Number of neighbors	1
Distance metric	Correlation
Distance weight	Inverse
Standardize data	true
Accuracy	100 %

5. CONCLUSION

The paper presents a novel method for fault detection in ball bearings using two types of ML algorithms: KNN and SVM. The datasets for the development of the algorithms were obtained through vibration measurements at the developed station for capturing, recording, and analyzing vibrations in ball bearings. The first part of the research on the application of said ML algorithms in ball bearing fault detection involved experimentally varying the algorithm parameters to obtain a more precise classifier model. This approach resulted in highly accurate algorithms: the KNN classifier achieved an accuracy of 98.5 % and the SVM classifier achieved an accuracy of 99.5 %. The second part of the research consisted of optimizing the hyperparameters of the two applied ML algorithms. Bayesian optimization was used as the optimization method. The result of the applied optimization was hyperparameters that led to both the KNN and SVM algorithms achieving a maximum success rate of 100 %.

The approach of applying hyperparameter optimization to ML algorithms has proven to be fully justified as the obtained classifier models achieved maximum accuracy. The methodology presented in this paper is applicable to a wide range of challenges in industrial diagnostics and predictive maintenance.

ACKNOWLEDGMENT

This research was supported by the research grants of the Serbian Ministry of Science, Technological Development and Innovations, grant No. 451-03-66/2024-03/200066 and grant No. 451-03-66/2024-03/200132.

REFERENCES

- [1] Achouch, M., Dimitrova, M., Ziane, K., Karganroudi, S. S., Dhoubi, R., Ibrahim, H., Adda, M. (2022). On predictive maintenance in Industry 4.0: Overview, models, and challenges. *Applied Sciences Review*, 12 (16), 8081. <https://doi.org/10.3390/app12168081>
- [2] Lindh, T. (2003). *On the condition monitoring of induction machines*. Doctoral Dissertation, Lappeenranta University of Technology, Lappeenranta, Finland. <https://urn.fi/URN:ISBN:951-764-843-X>
- [3] Tavner, P. J. (2008). Review of condition monitoring of rotating electrical machines. *IET Electric Power Applications*, 2 (4), 215-247. <https://doi.org/10.1049/iet-epa:20070280>
- [4] Sarma, N., Tuohy, P., Djurovic, S. (2023). Condition monitoring of rotating electrical machines. In *Encyclopedia of Electrical and Electronic Power Engineering*. Elsevier, 143-154. <https://doi.org/10.1016/b978-0-12-821204-2.00136-7>
- [5] Motor Reliability Working Group. (1985). Report of large motor reliability survey of industrial and commercial installations, Part I. *IEEE Transactions on Industry Applications*, IA-21 (4), 853-864. <https://doi.org/10.1109/TIA.1985.349532>
- [6] Motor Reliability Working Group. (1985). Report of large motor reliability survey of industrial and commercial installations, Part II. *IEEE Transactions on Industry Applications*, IA-21 (4), 865-872. <https://doi.org/10.1109/TIA.1985.349533>
- [7] Albrecht, P. F., Appiarius, J. C., McCoy, R. M., Owen, E. L., Sharma, D. K. (1986). Assessment of the reliability of motors in utility applications – updated. *IEEE Transactions on Energy Conversion*, EC-1 (1), 39-46. <https://doi.org/10.1109/TEC.1986.4765668>
- [8] Thorsen, O. V., Dalva, M. (1995). A survey of faults on induction motors in offshore oil industry, petrochemical industry, gas terminals, and oil refineries. *IEEE Transactions on Industry Applications*, 31 (5), 1186-1196. <https://doi.org/10.1109/28.464536>
- [9] Shin, K., Hammond, J. (2008). *Fundamentals of Signal Processing for Sound and Vibration Engineers*. Wiley, ISBN 978-0-470-51188-6.
- [10] Trendafilova, I. (2010). An automated procedure for detection and identification of ball bearing damage using multivariate statistics and pattern recognition. *Mechanical Systems and Signal Processing*, 24 (6), 1858-1869. <https://doi.org/10.1016/j.ymssp.2010.02.005>

- [11] Marić, D., Duspara, M., Šolić, T., Samardžić, I. (2019). Application of SVM models for classification of welded joints. *Technical Gazette*, 26 (2), 533-538. <https://doi.org/10.17559/TV-20180305095253>
- [12] Rozing, G., Duspara, M., Dudic, B., Savkovic, B. (2023). Research on the effect of load and rotation speed on resistance to combined wear of stainless steels using ANOVA analysis. *Materials*, 16 (12), 4284. <https://doi.org/10.3390/ma16124284>
- [13] Pandian, A., Ali, A. (2010). A review of recent trends in machine diagnosis and prognosis algorithms. In *2009 World Congress on Nature & Biologically Inspired Computing (NaBIC)*. IEEE. <https://doi.org/10.1109/NABIC.2009.5393625>
- [14] Liu, R., Yang, B., Zio, E., Chen, X. (2018). Artificial intelligence for fault diagnosis of rotating machinery: A review. *Mechanical Systems and Signal Processing*, 108, 33-47. <https://doi.org/10.1016/j.ymssp.2018.02.016>
- [15] Liu, X., Zhou, Q., Zhao, J., Shen, H., Xiong, X. (2019). Fault diagnosis of rotating machinery under noisy environment conditions based on a 1-D convolutional autoencoder and 1-D convolutional neural network. *Sensors*, 19 (4), 972. <https://doi.org/10.3390/s19040972>
- [16] Samanta, B., Al-Balushi, K. R. (2003). Artificial neural network based fault diagnostics of rolling element bearings using time-domain features. *Mechanical Systems and Signal Processing*, 17 (2), 317-328. <https://doi.org/10.1006/mssp.2001.1462>
- [17] Kankar, P. K., Sharma, S. C., Harsha, S. P. (2011). Fault diagnosis of ball bearings using machine learning methods. *Expert Systems with Applications*, 38 (3), 1876-1886. <https://doi.org/10.1016/j.eswa.2010.07.119>
- [18] Jenkins, C. D. (2019). *Bearing fault detection and wear estimation using machine learning*. Technical Report LA-UR-19-27700. <https://doi.org/10.2172/1557163>
- [19] Sawaqed, L. S., Alrayes, A. M. (2020). Bearing fault diagnostic using machine learning algorithms. *Progress in Artificial Intelligence*, 9 (4), 341-350. <https://doi.org/10.1007/s13748-020-00217-z>
- [20] Zhang, S., Zhang, S., Wang, B., Habetler, T. G. (2020). Deep learning algorithms for bearing fault diagnostics—A comprehensive review. *IEEE Access*, 8, 29857-29881. <https://doi.org/10.1109/ACCESS.2020.2972859>
- [21] Teotrakool, K., Devaney, M. J., Eren, L. (2008). Bearing fault detection in adjustable speed drives via a support vector machine with feature selection using a genetic algorithm. In *2008 IEEE Instrumentation and Measurement Technology Conference*. IEEE, 1129-1133. <https://doi.org/10.1109/IMTC.2008.4547208>
- [22] Konar, P., Chattopadhyay, P. (2011). Bearing fault detection of induction motor using wavelet and Support Vector Machines (SVMs). *Applied Soft Computing*, 11 (6), 4203-4211. <https://doi.org/10.1016/j.asoc.2011.03.014>
- [23] Benkedjouh, T., Medjaher, K., Zerhouni, N., Rechak, S. (2012). Fault prognostic of bearings by using support vector data description. In *2012 IEEE Conference on Prognostics and Health Management*. IEEE, 1-7. <https://doi.org/10.1109/ICPHM.2012.6299511>
- [24] Kang, M., Kim, J., Kim, J. M., Tan, A. C., Kim, E. Y., Choi, B. K. (2015). Reliable fault diagnosis for low-speed bearings using individually trained support vector machines with kernel discriminative feature analysis. *IEEE Transactions on Power Electronics*, 30 (5), 2786-2797. <https://doi.org/10.1109/TPEL.2014.2358494>
- [25] Song, W., Xiang, J. (2017). A method using numerical simulation and support vector machine to detect faults in bearings. In *2017 International Conference on Sensing, Diagnostics, Prognostics, and Control (SDPC)*. IEEE, 603-607. <https://doi.org/10.1109/SDPC.2017.118>
- [26] Pandarakone, S. E., Mizuno, Y., Nakamura, H. (2019). Evaluating the progression and orientation of scratches on outer-raceway bearing using a pattern recognition method. *IEEE Transactions on Industrial Electronics*, 66 (2), 1307-1314. <https://doi.org/10.1109/TIE.2018.2833025>
- [27] King, R. D., Feng, C., Sutherland, A. (1995). StatLog: Comparison of classification algorithms on large real-world problems. *Applied Artificial Intelligence*, 9 (3), 289-333. <https://doi.org/10.1080/08839519508945477>
- [28] Kohavi, R., John, G. H. (1995). Automatic parameter selection by minimizing estimated error. In *Machine Learning Proceedings 1995*. Morgan Kaufmann Publishers, 304-312. <https://doi.org/10.1016/B978-1-55860-377-6.50045-1>
- [29] Michie, D., Spiegelhalter, D. J., Taylor, C. C. (1994). *Machine Learning, Neural and Statistical Classification*. Prentice Hall, ISBN 978-0131063600.
- [30] Ripley, B. D. (1993). Statistical aspects of neural networks. In *Networks and Chaos - Statistical and Probabilistic Aspects*. Chapman & Hall, 40-123. ISBN 0-412-46530-2.
- [31] Hutter, F., Kotthoff, L., Vanschoren, J. (eds.) (2019). *Automated Machine Learning: Methods, Systems, Challenges*. Springer, ISBN 978-3-030-05318-5. <https://doi.org/10.1007/978-3-030-05318-5>
- [32] Shahriari, B., Swersky, K., Wang, Z., Adams, R., de Freitas, N. (2016). Taking the human out of the loop: A review of Bayesian optimization. *Proceedings of the IEEE*, 104 (1), 148-175. <https://doi.org/10.1109/JPROC.2015.2494218>
- [33] Brochu, E., Cora, V. M., De Freitas, N. (2010). A tutorial on Bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning. *arXiv*, 1012.2599. <https://doi.org/10.48550/arXiv.1012.2599>

Received April 12, 2024

Accepted March 7, 2025