26th International Congress of Mechanical Engineering
COBEM 2021
ABCM Associação Brasileira de Engenharia e Ciências Mecânicas
November 22-26, 2021, Virtual Congress, Brazil

COB-2021-0634
MACHINE LEARNING TECHNIQUES TO FAULT DETECTION IN ROTATING MACHINES

Luís Otávio Garavaso
Gregory Bregion Daniel
Kátia Lucchesi Cavalca
School of Mechanical Engineering – UNICAMP. Rua Mendeleyev, 200 - Cidade Universitária, Campinas - SP, 13083-860
otaviogaravaso@gmail.com, gbdaniel@fem.unicamp.br, katia@fem.unicamp.br

Abstract. *Rotating machines are necessary for power generation since they perform vital roles that extend from extracting resources, such as fuels, to converting kinetic energy from water and airflow into electricity for consumption. These machines, however, face different types of mechanical faults that alter the system's vibration response creating patterns known as fault signatures. Therefore, interpreting these vibration marks with adequate tools leads to proper fault identification, which improves maintenance scheduling, reduces repair time and machinery breakage. While some of these signatures can be associated to the first harmonic response, such as the rotating unbalance affecting the amplitude of the measured signal, others are still not fully understood, which turns Machine Learning into a powerful tool for fault diagnostics. This research, therefore, combines Mechanical Engineering knowledge with Machine Learning techniques to train widely used algorithms, such as Logistic Regression, Support Vector Machines, and Artificial Neural Networks, to identify through vibration data the existence of mechanical faults in rotating machines. The data is initially generated through numerical integration of equations of motion added by different levels of noise. Rotor modeling, otherwise, can be analytical, as in the case of Laval Rotor lumped parameters solution, or through the Finite Elements Method, for more complex geometries. The Support Vector Machine and the Artificial Neural Networks have shown to be reliable algorithms for rotating unbalance detection. However, it is necessary to acknowledge the Support Vector Machine's performance for requiring way less data than the Artificial Neural Networks. Finally, this work is promising to diagnostics techniques evaluation depending on the fault signature of rotating machines.*

Keywords: *Rotordynamics, Machine Learning, Support Vector Machine, Artificial Neural Network, Fault Identification*

1. INTRODUCTION

The 2019 National Energy Balance (Brasil, 2019a) evaluated the Brazilian energy matrix in 153 GW of installed power. However, the Ten-Year Energy Expansion Plan (Brasil, 2019b) foresees the system's expansion by 68 GW, representing an invested amount of R$ 2.3 trillion in ten years, being R$ 1.9 trillion in oil, natural gas, and biofuels and R$ 456 billion in electricity generation and transmission.

Resource extraction, as well as energy conversion for the final consumer, depend directly on rotating machines, such as hydraulic turbines, compressors, and reducers. Thus, system failure can result in significant economic loss and undesirable disturbances to society, so they must be identified and repaired quickly. One of the tools that can perform such an identification Machine Learning.

Machine Learning is the field of knowledge that allows computers to perform tasks without having been explicitly programmed. This definition was proposed by the American computer scientist Arthur Lee Samuel in 1959, after the development of an algorithm capable of learning to play checkers and defeating a human opponent (Samuel, 1959).

In classical programming, one can describe rules to the computer through lines of code, in which for each input, there is an output based on the machine's interpretation. With Machine Learning, on the other hand, the programmer knows the inputs and outputs, but not the translation rules. A practical example is of an autonomous car, in which you have the car's origin, route, and the desired destination. However, programming all the driving rules, considering adverse and unforeseen conditions, escapes the classic programming capabilities.

This limit can be overcome by implementing machine learning models to analyze an expressive data amount through mathematical and statistical methods. Subsequently, when the models become capable of guaranteeing reliable results, it means the algorithm learned to map inputs into outputs like an intelligent being, creating the market known as Artificial Intelligence, which will be worth US$ 100 billion in 2025, according to the Harvard Business School (Wellers *et al.*, 2017).

Machine Learning is proving to be a versatile and reliable tool, with remarkable achievements in several areas, such as fraud detection, facial recognition, and more importantly for this project, mechanical failure detection, as demonstrated by Alves *et al.* (2020) and Gecgel *et al.* (2020) which concluded that convolutional neural networks (CNNs) can be used for diagnosing the ovalization and the wear faults in hydrodynamic bearings, respectively.

Rotating components are subject to different types of failure, which alter the system vibration, causing patterns known as failure signatures. These marks' study, as a result, allow information extraction leading to machinery fault identification and turning Machine Learning into a powerful tool to diagnose system conditions.

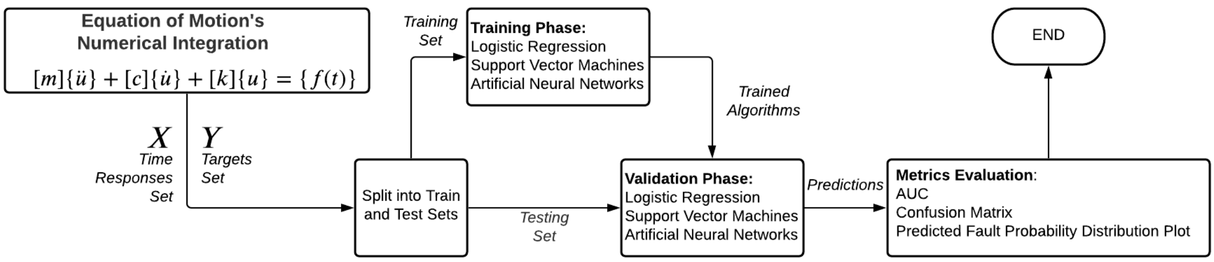
2. OBJECTIVES

This paper aims to be a preliminary study of machine learning techniques applied to rotating unbalance identification. For this reason, data generated on a Laval rotor modeled through the lumped parameters solution will be used to train three Machine Learning algorithms: the Support Vector Machines and the Artificial Neural Networks, which are widely used in fault diagnosis of rotating machinery (Liu *et al.*, 2018), and the Logistic Regression, that, given its mathematical simplicity, was chosen to evaluate if the application of more complex algorithms is justified.

The methodology follows the flowchart presented in Fig. 1, which depicts the equations of motion (EOM) getting numerically integrated, outputting time responses related to fault conditions. Next, the data is split into training and testing sets and later given to the Logistic Regression, the Support Vector Machines, and the Artificial Neural Networks algorithms for training and, finally, validation with commonly used metrics such as the area under the receiver operating characteristic curve, the confusion matrix, and the predicted fault probability distribution plot.

It must be made clear that this project does not implement the algorithms' mathematical formulation but instead use Python's open-source libraries such as *Scikit-learn* (Pedregosa *et al.*, 2011) and *Tensorflow* (Abadi *et al.*, 2015).

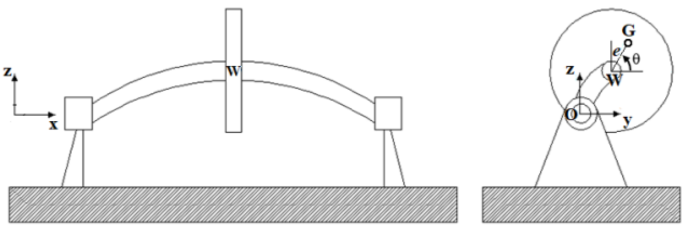
Figure 1. Machine Learning Flowchart



3. LAVAL ROTOR

According to Kramer (1993), the Laval rotor corresponds to the simplest form of a rotating machine. It consists of a massless and isotropic cylindrical shaft with a constant cross-section along its length. The shaft is on top of two ideal bearings that restrict vertical or horizontal displacements and do not resist to moments of any kind. The rotor also contains, equidistant to the bearings, a rigid and unbalanced disk with its center of mass G distant from the rotation axis W by an eccentricity e, as depicted in Fig. 2.

Figure 2. Laval or Jeffcott Rotor (Cavalca, 2020)



The Laval rotor can be modeled as a three degree of freedom (DOF) mechanism. Two of these DOFs are related to the linear displacements of the disk's geometrical center according to the O (Fig. 2) inertial reference frame in the y and z directions and represented by the u_y and u_z coordinates. The third one, on the other hand, relates to the disk's rotation on its own geometrical center and it is represented by the θ coordinate.

When the disk rotates in constant angular velocity, however, Kramer (1993) says the EOM can be written only in terms of the linear displacements DOFs, as shown in Eq. (1) and Eq. (2).

$$\ddot{u}_y m + \dot{u}_y c + k u_y = m e \Omega^2 \cos(\Omega t) \tag{1}$$

$$\ddot{u}_z m + \dot{u}_z c + k u_z = m e \Omega^2 \sin(\Omega t) - m g \tag{2}$$

Where e and m are the eccentricity and the mass of disk, respectively, and c and k are the damping and the stiffness of the cylindrical shaft, respectively. Moreover, g represents the gravitational field and the single and the double dot notations are the velocity and the acceleration of the DOFs.

4. DATA GENERATION

Equations (1) and (2) can be numerically integrated to generate time responses under different rotating unbalance levels, which then can be used to train machine learning algorithms to detect this mechanical fault.

Table 1 shows the simulated Laval rotor properties.

Table 1. Physical Properties of the Simulated Laval Rotor

Shaft	**Geometrical Properties**	**Diameter**	D	10 mm
		Length	L	800 mm
	Material Properties	**-**	-	Steel
	Calculated Properties	**Stiffness** (1)	k	9203.88 N/m
		Damping (2)	c	9.2039 N.s/m
Disk	**Geometrical Properties**	**Diameter**	D	90 mm
		Length	L	20 mm
		Unbalance	me	0.001 kg.m
	Material Properties	**-**	-	Steel
	Calculated Properties	**Volume**	V	0.1272×10^{-3} m^3
		Mass	m	0.9988 kg
Steel	**Material Properties**	**Young's Module**	E	200 GPa
		Density	ρ	7850 kg/m^3

(1) Calculated as $k = \frac{3\pi E D^4}{4L^3}$ (Kramer, 1993), (2) Calculated as $c = 10^{-3} \cdot k$

Additionally, the shaft is isotropic and the disk is centered in the machine, therefore, the mechanism's natural frequency ω_n can be calculated as 96 rad/s.

In this work data will be generated considering $\Omega = \omega_n = 96$ rad/s since the rotating unbalance fault is associated to the first harmonic response.

Furthermore, to generate N samples for the train and test sets, the unbalance me is used to form three normal distributions with $N/3$ samples each, as shown in Fig. 3, with the following properties:

$$\mathcal{N}_1(\mu = me,\, \sigma = 0.05me) \tag{3}$$

$$\mathcal{N}_2(\mu = 1.15me,\, \sigma = 0.05me) \tag{4}$$

$$\mathcal{N}_3(\mu = 0.85me,\, \sigma = 0.05me) \tag{5}$$

Where μ and σ represent the mean and the standard deviation of the normal distributions, respectively.

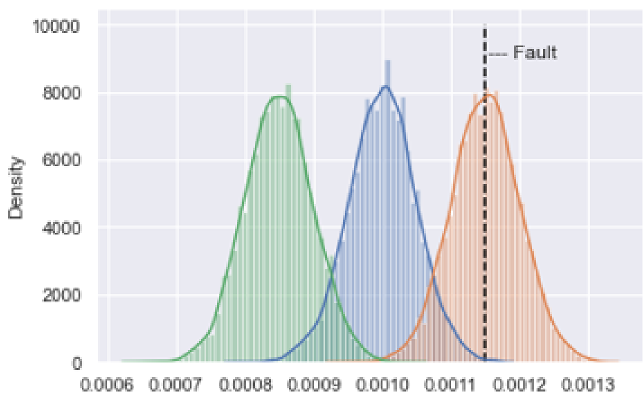


Figure 3. Normal Distributions for Unbalance Fault Definition

Next, an iterative process starts by drawing a sample k from one of the three normal distributions. The observation is then substituted in Eq. (1) and Eq. (2) for numerical integration outputting the time responses of the linear displacements related to the k sample $u_{y,k}$ and $u_{z,k}$.

The unbalance me_k is also used to determine the fault condition y_k associated with the k sample.

In this paper it is considered that the rotating unbalance relates to a faulty operation if the unbalance used to generate the time response surpasses three standard deviations above the $\mathcal{N}_1$ distribution's mean as shown in Fig. 3 and Eq. (6).

$$y_k(me_k) = \begin{cases} 0 \text{ (healthy)}, & \text{if } me_k \leq 1.15me \\ 1 \text{ (faulty)}, & \text{if } me_k > 1.15me \end{cases} \tag{6}$$

Once N samples are generated in the iterative process, the time responses and the targets sets are saved as:

$$X = \{u_{y,1} \quad u_{y,2} \quad u_{y,3} \quad \dots \quad u_{y,N}\}^T \tag{7}$$

$$Y = \{y_1 \quad y_2 \quad y_3 \quad \dots \quad y_N\}^T \tag{8}$$

Where X and Y are the time responses and the targets sets, respectively.

Next, according to Alves *et al.* (2020), SNR values in the range of 5-30 SNR are commonly added to the time responses to better simulate real machinery data, therefore a noise of 25 dB is added to the X set.

Finally, for algorithm training and validation, X and Y must be randomly split into training and test sets, leaving at least 80% of the data generated for training. Moreover, it is desirable that the proportions of 0 and 1 in the training set are about the same as in the test set.

5. ALGORITHMS

5.1 Logistic Regression

According to Geron (2019), the logistic regression is an algorithm that computes the weighted sum S_p of the input data and transforms it through a logistic function σ_m, such that:

$$S_p = \mathbf{w}^T\mathbf{x} + b = w_1x_1 + w_2x_2 + \dots + w_nx_n + b \tag{9}$$

$$\sigma_m = \frac{1}{1 + \exp(-S_p)} \tag{10}$$

Where $\mathbf{x}$ and $\mathbf{w}$ are the vector of the DOF x evaluated in time t_i and $\mathbf{w}$ the vector of the weights w_i to be determined by logistic regression, respectively. Additionally, b represents an independent term known as bias.

The prediction in a logistic regression, in turn, is made through the following relationships:

$$\hat{p}_k = \sigma_m(S_p) \tag{11}$$

$$\hat{y}_k = \begin{cases} 0, & \text{if } \hat{p}_k \leq 0.5 \\ 1, & \text{if } \hat{p}_k > 0.5 \end{cases} \tag{12}$$

Where index k is a row in the input data set and $\hat{p}$ and $\hat{y}$ are the predictions in probability and in binary, respectively.

It is observed that $\hat{p}_k$ or σ_m are limited between 0 ($S_p = -\infty$) and 1 ($S_p = \infty$), so it is understood that their results correspond to failure probabilities, which allows direct comparison with the $y_k(me_k)$ values and iterative adjustment of w and b which, in turn, is done by minimizing the following *loss function*:

$$J(w) = -\frac{1}{N}\sum_{k=1}^{N} y_k \log \hat{p}_k + (1 - y_k)\log(1 - \hat{p}_k) \tag{13}$$

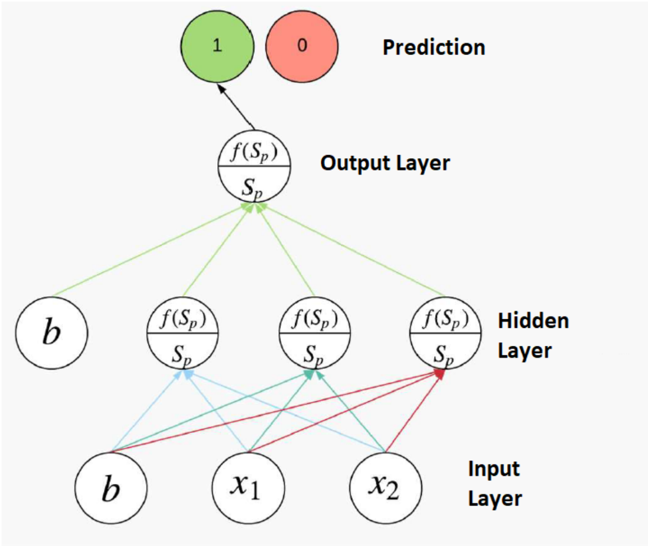
Where $J(w)$ is the *loss function* and N is the number of observations (rows) of the input data set.

The minimization of the Eq. (13) occurs by obtaining the gradients of the function J in relation to the weights w_i, which are obtained through Eq. (14):

$$\frac{\partial J}{\partial w_i} = \frac{1}{N}\sum_{k=1}^{N}(\hat{p}_k - y_k)x_i \tag{14}$$

Next, Eq. (13) and Eq. (14) are inputted to a minimization algorithm such as the Gradient Descent, which results in the best set of weights $\mathbf{w}$ to obtain the fault conditions from the input values $\mathbf{x}$.

Figure 4. Representation of an Artificial Neural Network



5.2 Artificial Neural Networks

According to Geron (2019), Artificial Neural Networks (ANNs) correspond to a mathematical model inspired by the human brain and are suitable for high complexity applications such as image classification and audio recognition.

As shown in Fig. 4, ANNs are constructed in layers starting with an input layer, passing through one or more hidden layers, and, finally, performing the prediction in the output layer.

These successive operations along the layers are performed through linear matrix operations which, after resulting in weighted sums, are transformed through non-linear functions, known as activation functions. This procedure can be described mathematically by the Eq. (15) and Eq. (16).

$$\mathbf{S}_{p,c} = \mathbf{X}_c\mathbf{W}_c + \mathbf{b}_c \tag{15}$$

$$\mathbf{A}_{c+1} = f_{c+1}(\mathbf{S}_{p,c}) \tag{16}$$

Where c is the layer's index and $\mathbf{X}$, $\mathbf{W}$ and $\mathbf{b}$ are the input, weights and biases matrices at layer c, respectively. Additionally, $\mathbf{S}_p$ is the matrix with the weighted sums at layer c that gets transformed by the non-linear activation function $f(\mathbf{S}_p)$ outputting $\mathbf{A}_c$.

Next, the layers are connected according to Eq. (17).

$$\mathbf{X}_c = \mathbf{A}_c \tag{17}$$

This paper evaluates a neural network with one input layer, two hidden layers and one output layer. Furthermore, the hidden layers use the hyperbolic tangent as the activation function ($f_2 = f_3 = \tanh$), while the output layer uses the logistic function ($f_4 = \sigma_m$). It should also be made clear that the hidden layers have 128 connection units, while the output layer has two connection units to match the number of fault conditions analyzed.

Since the output layer's activation function is the logistic function, the weights and bias optimization is also done by minimizing the Eq. (13) from the logistic regression. The ANN gradients, in contrast, are obtained through the backpropagation method (Rumelhart *et al.*, 1986) that require successive chain rules, starting from the output layer until the input layer, as depicted in Fig. 5.

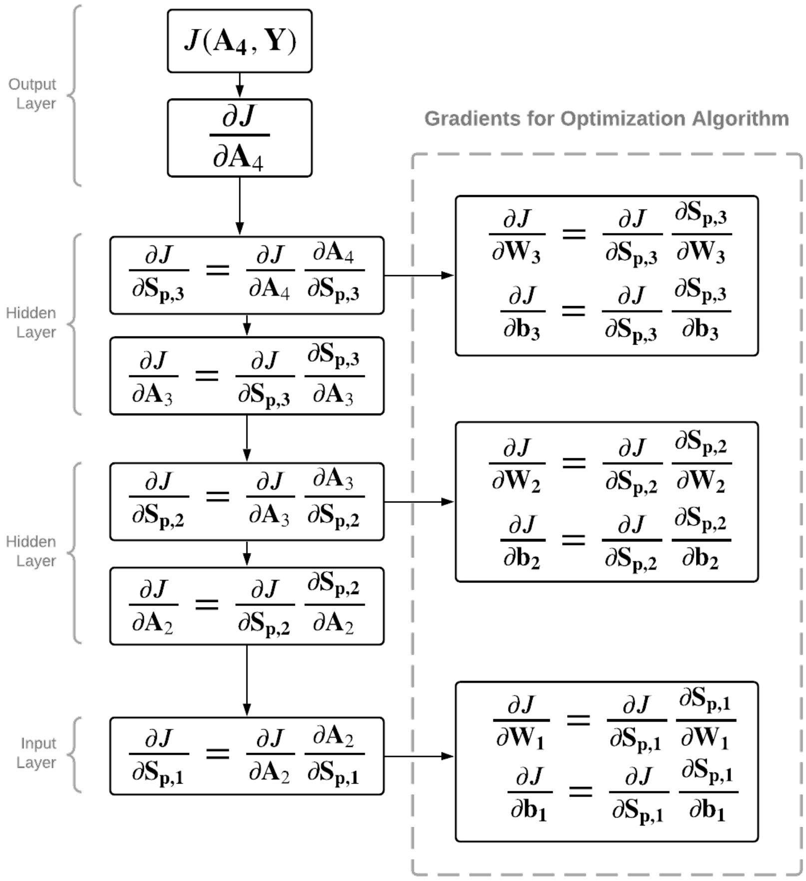
5.3 Support Vector Machines

The Support Vector Machine (SVM) is an algorithm that separates the operating conditions into two different regions in space with a hyperplane (Geron, 2019). The separation is done through the kernel trick, an operation in which each par vector of the dataset is taken to a larger dimension through specific functions capable of quantifying similarities between observations.

Unlike the logistic regression and the artificial neural networks in this paper, the SVM finds the best set of weights through a minimization problem subject to constraints. Furthermore, incorrectly classified observations will be penalized by a regularization factor during the optimization process. Hence, the optimization problem is described as in Eq. (18):

$$\begin{cases} \min_{\mathbf{w},b,\zeta} & \frac{1}{2}\mathbf{w}^T\mathbf{w} + C\sum_{k=1}^{N}\zeta_k \\ \text{subject to} & s_k(\mathbf{w}^T\phi(\mathbf{x}) + b) \geq 1 - \zeta_k \end{cases} \tag{18}$$

Figure 5. ANN Gradients According to Backpropagation Method



Where s is the sign function and ϕ is a mapping function that takes $\mathbf{x}$ to a higher-dimensional space. Additionally, ζ corresponds to how distant an observation is from its proper region and C is the penalty for the misclassified samples.

According to Geron (2019), the SVM is also part of the lagragian duality problems, which state that an optimization problem can be equivalently solved from two perspectives if the objective function and its constraints are convex and differentiable. Thus, in the case of the SVM, the dual problem corresponds to Eq. (19).

$$\begin{cases} \min_\alpha & \frac{1}{2}\alpha^T Q\alpha - d^T\alpha \\ \text{subject to} & s^T\alpha = 0 \\ \text{with} & 0 \leq \alpha_k \leq C,\ for\ k = 1, 2, 3, \ldots, N \\ & Q_{k,l} \equiv s_k s_l K(\mathbf{x}_k, \mathbf{x}_l) = s_k s_l \phi(\mathbf{x}_k)^T \phi(\mathbf{x}_l) \end{cases} \tag{19}$$

Where α_k stands for the dual coefficients, Q is a positive-definite matrix, K is the kernel function, d is a vector of ones and l is the pair index of k.

It is observed in Eq. (19) that the kernel function K corresponds to the product between the mapping functions ϕ. According to Geron (2019), this relationship is guaranteed by Mercer's Theorem, which states that the kernel function can be used instead of mapping functions just by knowing that ϕ exists, even without having knowledge of its form.

In this project the kernel function it the Gaussian Radial Basis Function (*Gaussian RBF Kernel*), defined as:

$$K(\mathbf{x}_k, \mathbf{x}_l) = \exp\left(-\gamma ||\mathbf{x}_k - \mathbf{x}_l||^2\right) \tag{20}$$

Where γ is a similarity coefficient.

Note that if $\mathbf{x}_k$ and $\mathbf{x}_l$ are distant vectors in space, $K(\mathbf{x}_k, \mathbf{x}_l) \approx 0$, and if they are close, $K(\mathbf{x}_k, \mathbf{x}_l) \approx 1$.

Finally, the prediction is made by determining whether a given vector $\mathbf{x}$ has greater similarity with vectors that indicate healthy (0) or faulty (1) operation, which mathematically can be represented by the Eq. (21).

$$\hat{y}_k(x) = \sum_{k=1}^{N} s_k \alpha_k K(x_k, x) + b \tag{21}$$

6. METRICS

The algorithms performance validation in this paper is done using three metrics: the Confusion Matrix, the Area Under the Receiver Operating Characteristic Curve, and the Predicted Fault Probability Distribution Plot.

6.1 Confusion Matrix

By definition, the confusion matrix C is such that $C_{i,j}$ is equal to the number of observations that belong to the i group and are predicted to belong to the j group (Pedregosa *et al.*, 2011).

This metric requires prediction in binary values. Thus, its construction requires a rating threshold, which is usually equal to 0.5.

Reading the confusion matrix is as follows:

- The main diagonal is evaluated, which indicates how many samples were correctly classified as 0 (negative) and 1 (positive).
- The secondary diagonal is evaluated, which indicates how many samples were incorrectly classified as 0 (negative) and 1 (positive).
- Still on the secondary diagonal, determine if the errors were mostly *False Positives* or *False Negatives*.

6.2 Area Under the Receiver Operating Characteristic Curve

Machine learning algorithms usually make binary predictions as the output's probability of being positive or equal to one. Therefore it is possible to define a threshold above which the trained algorithm should classify an input as positive (1). Likewise, if the input is mapped as a value below the limit, the sample is classified as negative (0). Consequently, different true and false positive rates are obtained for each applied threshold. By plotting these various proportions, the receiver operating characteristic (ROC) curve is obtained.

Next, the area under the ROC curve, or AUC for short, can be calculated to quantify an algorithm's ability to find a threshold that separates the positive (1) and the negative (0) samples. The AUC metric is interpreted as:

- If $AUC \approx 1$: the algorithm is highly capable of finding a fault separation threshold
- If $AUC \approx$ 0,5: the algorithm is not capable of finding a fault separation threshold

6.3 Predicted Fault Probability Distribution Plot

This statistical metric presents immediate visual information and constructs a histogram in which the x axis represents the probability of failure. Ideally, the algorithm will be able to:

- Define a fault separation threshold
- Allocate negative observations (0) on the far left of the graph
- Allocate positive observations (1) at the far right of the graph

7. RESULTS AND DISCUSSION

In this paper, 700003 rotating unbalance samples were generated according to the methodology describe above.

The size of the training and the test sets given to the algorithms are shown in Tab. 2.

Table 2. Algorithms' Data Usage

	Logistic Regression	**ANN**	**SVM**
Training	15000	630002	15000
Validation	70001	70001	70001

Note that both logistic regression and SVM required only 15000 for training while the ANN required 630002. The reason is that, even though the Logistic Regression is suitable for any dataset size, it does not show significant metrics improvement with more training data. The ANN, in contrast, is a deep learning algorithm that usually demonstrates a positive correlation between dataset size and performance improvement. Additionally, Geron (2019) says the SVM is well suited for medium-sized datasets given its computational complexity scaling quadratically with the number of training samples.

Table 3. Algorithms' Performance

		Logistic Regression	ANN	SVM
Confusion Matrix	**TP**	41845	50360	43740
	TN	19628	19070	19628
	FP	8528	13	6633
	FN	0	558	0
ROC Curve	**AUC**	0.9999	0.9999	0.9999

After training, the algorithms were validated with the same test set to allow direct comparison between them.

Table 3 contains the Confusion Matrix's and ROC Curve's results for all algorithms evaluated in this paper.

The results show that all three algorithms achieved high AUC, which indicates that they were able to find a probability threshold that separates faulty and healthy observations.

The confusion matrix's results, on the other hand, show that the ANN got fewer misclassifications. It is important to clarify, however, that most ANN mistakes were false negatives, which are highly undesirable for mechanical engineering applications considering that they can lead to catastrophic events. The logistic regression and the SVM, in contrast, made more misclassifications, but only false positives, which in the worst case scenario requires the complete stop of the machine's operation for a maintenance evaluation.

Finally, Fig. 6, Fig. 7 and Fig. 8 depict the predicted fault probability distribution plots.

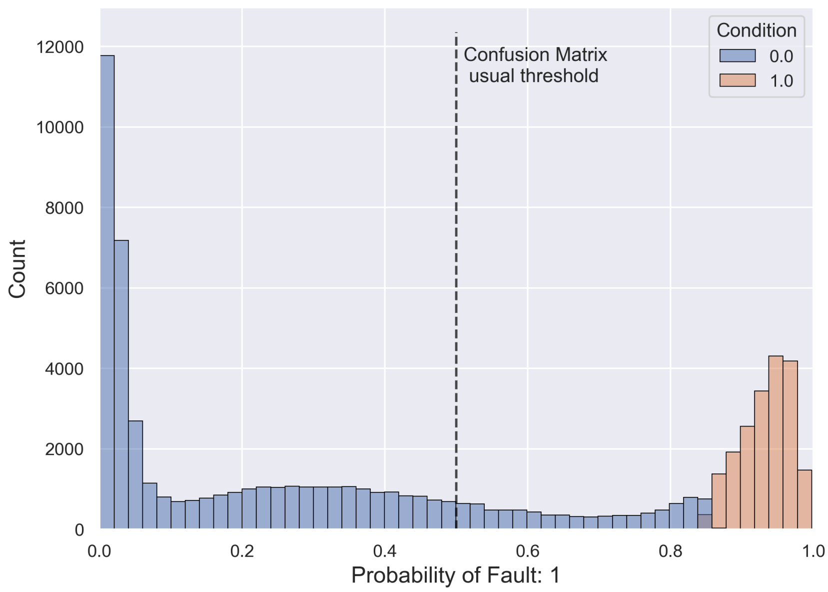


Figure 6. Logistic's Regression Predicted Fault Probability Distribution

It is clear in Fig. 7 and Fig. 8 that both the ANN and the SVM were secure about their predictions since the negative samples gather to the left while the positive ones, to the right. The logistic regression, on the contrary, proves to be inferior when compared to the other two, since it distributes the samples along the horizontal axis.

The Logistic Regression's inferiority for this problem is explained in its mathematical formulation. Since the algorithm computes a weighted sum of the input data and transforms it through a logistic function, it is clear that the Logistic Regression assumes that the dependent and independent variables are linearly correlated. In contrast, the SVM and the ANN with the kernel and the activation functions, respectively, can construct non-linear separation boundaries between the fault conditions, making them more adequate for fault detection in time series.

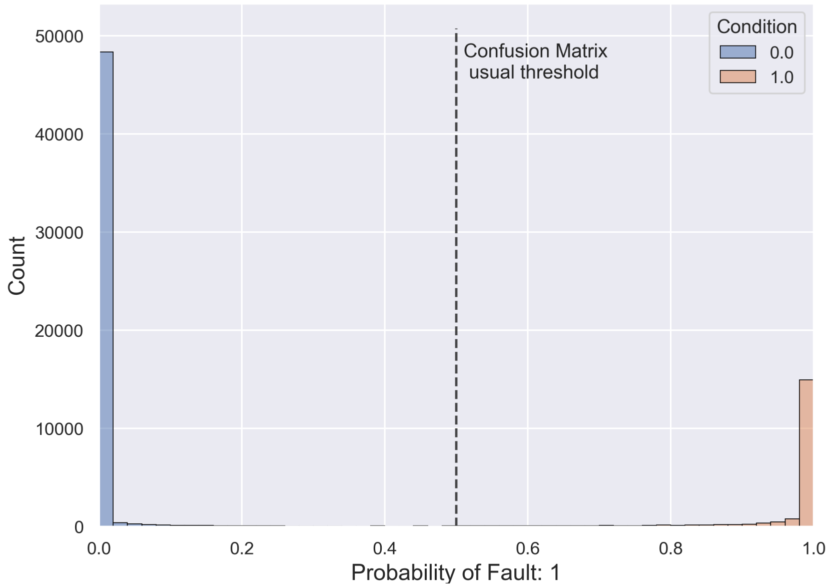


Figure 7. ANNs' Predicted Fault Probability Distribution

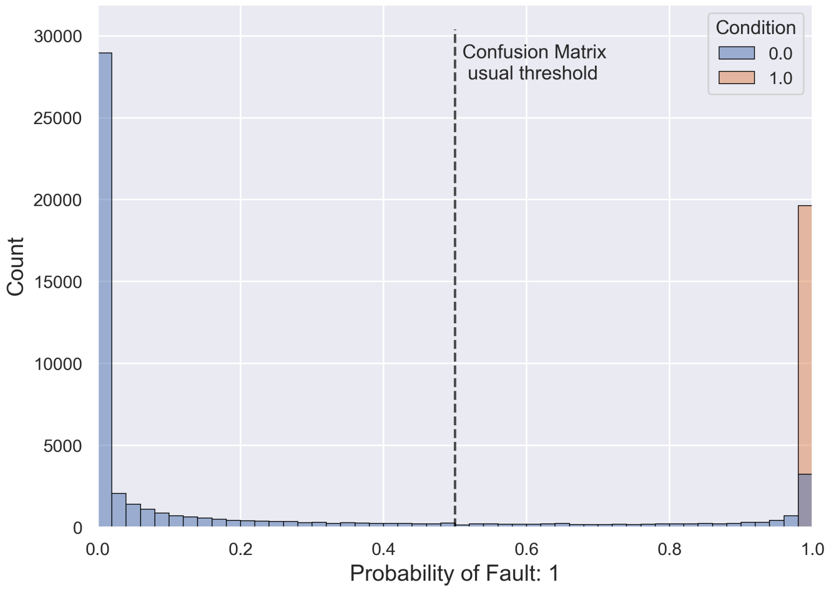


Figure 8. SVM's Predicted Fault Probability Distribution

8. CONCLUSION

This paper aims to be a preliminary study of machine learning techniques applied to rotating unbalance identification. For this reason, the Logistic Regression's, the Support Vector Machine's, and the Artificial Neural Networks' performance is evaluated on data generated from a Laval rotor modeled through the lumped parameters solution.

The methodology generates data by using a known unbalance level to create three normal distributions. Samples drawn from the distributions are then plugged in the Laval rotor equations of motions and associated with an operational condition: 1 for faulty and 0 for healthy.

Next, noise is added to the data to simulate real time responses and the datasets are split into training and test groups, which are then given to the algorithms for weights and biases optimization and, finally, for validation according to three commonly used metrics: the Confusion Matrix, the Area Under the Receiver Receiver Operating Characteristic Curve, and the Predicted Fault Probability Distribution Plot.

In conclusion, the SVM and the ANN are promising algorithms for rotating unbalance identification and are expected to be adequate for other mechanical faults. Additionally, both more complex algorithms have proven to be more reliable than the Logistic Regression, justifying their application in fault diagnosis. However, it is necessary to acknowledge the Support Vector Machine's performance for requiring considerably less data than the Artificial Neural Networks during the training phase to achieve similar results.

9. ACKNOWLEDGEMENTS

The authors thank the National Council for Scientific and Technological Development (CNPq) and the University of Campinas (UNICAMP) for infrastructural and financial support.

10. REFERENCES

Abadi, M., Agarwal, A., Barham, P., Brevdo, E. and Chen, Z., 2015. "TensorFlow: Large-scale machine learning on heterogeneous systems". URL `https://www.tensorflow.org/`. Software available from tensorflow.org.

Alves, D.S., Daniel, G.B., de Castro, H.F., Machado, T.H., Cavalca, K.L., Gecgel, O., Dias, J.P. and Ekwaro-Osire, S., 2020. "Uncertainty quantification in deep convolutional neural network diagnostics of journal bearings with ovalization fault". *Mechanism and Machine Theory*, Vol. 149, p. 103835. ISSN 0094114X. doi: 10.1016/j.mechmachtheory.2020.103835.

Brasil, 2019a. "Balanço energético nacional". Technical report, Ministério de Minas e Energia. Secretaria de Planejamento e Desenvolvimento Energético. Brasilia, DF.

Brasil, 2019b. "Plano Decenal de Expansão de Energia 2029". Technical report, Ministério de Minas e Energia. Secretaria de Planejamento e Desenvolvimento Energético. Brasilia, DF.

Cavalca, K.L., 2020. *Introduction to the Laval or Jeffcott Rotor (in portuguese)*. Laboratório de Máquinas Rotativas, LAMAR - UNICAMP, Campinas.

Gecgel, O., Dias, J.P., Ekwaro-Osire, S., Alves, D.S., Machado, T.H., Daniel, G.B., de Castro, H.F. and Cavalca, K.L., 2020. "Simulation-Driven Deep Learning Approach for Wear Diagnostics in Hydrodynamic Journal Bearings". *Journal of Tribology*, Vol. 143, No. 8. ISSN 0742-4787. doi:10.1115/1.4049067. URL `https://doi.org/10.1115/1.4049067`.

Geron, A., 2019. *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow*. O'Reilly Media, Inc, Sebastopol, CA. ISBN 978-1492032649.

Kramer, E., 1993. *Dynamics of rotors and foundations*. Springer-Verlag, Berlin New York. ISBN 978-3-662-02800-1.

Liu, R., Yang, B., Zio, E. and Chen, X., 2018. "Artificial intelligence for fault diagnosis of rotating machinery: A review". *Mechanical Systems and Signal Processing*, Vol. 108, pp. 33–47. doi:10.1016/j.ymssp.2018.02.016.

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.

Rumelhart, D.E., Hinton, G.E. and Williams, R.J., 1986. "Learning representations by back-propagating errors". *Nature*, Vol. 323, No. 6088, pp. 533–536. ISSN 1476-4687. doi:10.1038/323533a0. URL `https://doi.org/10.1038/323533a0`.

Samuel, A.L., 1959. "Some studies in machine learning using the game of checkers". *IBM Journal of Research and Development*, Vol. 3, No. 3, pp. 210–229. doi:10.1147/rd.33.0210.

Wellers, D., Elliot, T. and Noga, M., 2017. "8 Ways Machine Learning Is Improving Companies' Work Processes". URL `https://hbr.org/2017/05/8-ways-machine-learning-is-improving-companies-work-processes`.

11. RESPONSIBILITY NOTICE

The authors are solely responsible for the printed material included in this paper.