Rotating machinery fault diagnosis using similarity-based models

Felipe M. L. Ribeiro, Matheus A. Marins, Sergio L. Netto, and Eduardo A. B. da Silva

***Abstract*— This work proposes an automatic fault classifier that uses similarity-based modeling (SBM) to identify faults on rotating machines. The similarity model can be used either as an auxiliary model to generate features for a classifier or as a standalone classifier. A new approach for training the model using a prototype-selection method is investigated. Experimental results are shown for the MaFaulDa database and for the Case Western Reserve University (CWRU) bearing database. Results indicate that the proposed modifications improve the generalization power of the similarity model and of the associated classifier, achieving accuracies of 96.4% on the MaFaulDa and 98.7% on the CWRU databases.**

***Keywords*— Fault diagnosis, Condition monitoring, Feature extraction, Similarity-based modeling, Machine learning.**

I. INTRODUCTION

One application of machine learning employed in the industry is the condition-based maintenance of an equipment where one attempts to classify and predict failures. Rotating machines are an important piece of equipment used in a variety of applications, including airplanes, power turbines, oil and gas industry, and so on [1], [2]. Due to their complexity, these machines require a meticulous maintenance procedure to ensure reliability, avoiding production stops and incurring costs.

There are many approaches for detecting faults in rotating machines. Most extract features from vibration signals to assess the equipment current condition. Different features are needed to obtain useful information relevant to detect faults from the original sources over multiple conditions. These features can be classified considering their domain (time, spatial, time-spectral, or spectral) or its computation method (transform coefficients or aggregated statistics) [3], [4], [5]. As on example, the authors of [6] extract statistical and spectral features to detect failures in a machinery fault simulator (MFS) using multilayer perceptrons. In [7], the authors present a comparison of multiples classifiers under the bearing fault diagnostic task. Support vector machines (SVM) are employed to the same task in [8] and [9]. Lastly, a feature selection methods is evaluated with different classifiers in [4].

This work presents an automatic system for fault classification that uses similarity-based modeling (SBM) [10] as an auxiliary model to produce new features for a random forest classifier. Given a test sample, the SBM models return a similarity score between the sample and a set of samples, or *prototypes* [11], representing the target condition. The selection of an optimal set of prototypes is a key aspect of the SBM methodology. As such, this work compares two prototype selections methods, an adaptation of the original SBM [12] method for multiclass problems and the method proposed in [13] for $k$nn, applied on the SBM framework.

Two databases were employed to assess the models. One is the *Machinery Fault Database* (MaFaulDa) [14], comprising 1951 scenarios from a machinery fault simulator working under multiple conditions. Each scenario is monitored by six accelerometers, a tachometer and a microphone [6], [15]. The other database is the Case Western Reserve University (CWRU) bearing database [16], considered a standard reference in bearing diagnostics. This database was chosen considering that it permits comparisons to previous works [4], [7], [8], [9]. Results indicate that the proposed methodology is capable of correctly diagnosing the machine operating state, achieving an accuracy of 96.4% on the MaFaulDa database and 98.7% on the CWRU database.

This paper is organized as follows: Section II presents the SBM methodology [12], [17], [18], [19], [20], [21], including its original training phase and the new proposed approach for selecting the representative set. Section III describes the experimental methodology employed in this work to evaluate the proposed system, including details about the employed databases and the preprocessing and validation procedures. The experimental results obtained with the proposed methodology, including comparisons with other works, are discussed in Section IV. Finally, Section V provides the obtained conclusions and discusses possible future works.

II. SIMILARITY-BASED MODELING (SBM)

Similarity-based modeling (SBM) is a nonparametric modeling technique first proposed in [10] to supervise and detect equipment faults on a variety of industrial applications, including: fault diagnosis in a machinery fault simulator (MFS) [20], [18], anomaly detection in power plants [19], and modeling airplanes flight paths [17].

Given a sample at instant n, comprising M measures or features from multiples sources, represented as $\mathbf{x}_n = [x_n(1), x_n(2), \ldots, x_n(m)]^{\mathrm{T}}$, the SBM model returns the similarity between the evaluated sample and a set of samples $\mathcal{P}$ [21]. These samples, or *prototypes*, are a set of historical samples representing the target system condition. Given the prototype set $\mathcal{P}_c$ containing L samples representing the normal condition, we can arrange the prototypes in a into an $L \times M$

Program of Electrical Engineering, Universidade Federal do Rio Janeiro, Rio de Janeiro, RJ, 21941-972, Brazil. E-mails: {felipe.ribeiro, matheus.marins, eduardo, sergioln}@smt.ufrj.br. This work was partially supported by CNPq (XX/XXXXX-X).

"memory" matrix [10],

$$\mathbf{D} = \begin{bmatrix} \mathbf{x}_1 & \mathbf{x}_2 & \dots & \mathbf{x}_L \end{bmatrix}^{\mathrm{T}}. \tag{1}$$

The SBM model then attempts to represent the evaluated sample $\mathbf{x}_n$ producing an estimate $\hat{\mathbf{x}}_n$ as a linear combination of the prototype samples in $\mathbf{D}$, that is

$$\hat{\mathbf{x}}_n = \mathbf{D}^{\mathrm{T}} \frac{\mathbf{w}_n}{\|\mathbf{w}_n\|_1}. \tag{2}$$

The evaluation is made by computing the *similarity score* s_n between $\mathbf{x}_n$ and $\hat{\mathbf{x}}_n$ as

$$s_n = \mathbf{x}_n \circ \hat{\mathbf{x}}_n = s_{\mathrm{F}}\left(\mathbf{x}_n, \hat{\mathbf{x}}_n\right), \tag{3}$$

where s_{F} is a similarity function and $\circ$ represents a similarity operation. The similarity function returns a scalar in the interval $[0, 1]$, with $s_n = 1$ when the two vectors are identical, and $s_n \approx 0$ when they are very dissimilar. An example of similarity function is the one originally used in the SBM framework [20]

$$s(\mathbf{x}_i, \mathbf{x}_j) = \frac{1}{1 + \|\mathbf{x}_i - \mathbf{x}_j\|_2}. \tag{4}$$

The weight vector presented in Equation (2) is defined as

$$\mathbf{w}_n = \mathbf{G}^{-1}\mathbf{a}_n, \tag{5}$$

where

$$\mathbf{G} = \mathbf{D} \circ \mathbf{D}^{\mathrm{T}}, \tag{6}$$

and

$$\mathbf{a}_n = \mathbf{D} \circ \mathbf{x}_n, \tag{7}$$

where $\circ$ is the similarity function (Eq. (4)) that replaces the row-column inner products in the matrix multiplications.

Vector $\mathbf{a}_n$ evaluates the similarity between the current sample and each sample on matrix $\mathbf{D}$, whereas matrix $\mathbf{G}$ transforms the similarity vector $\mathbf{a}_n$ in a set of weights for each prototype. When $\mathbf{G} = \mathbf{I}$, the model is called called auto-associative kernel regression (AAKR) [22], a particular case of SBM equivalent to assuming no similarity between the samples within $\mathbf{D}$.

A. Multiclass SBM

The SBM was originally devised to detect abnormal operation conditions, which were associated with low similarity scores. To cope with multiple known conditions, we extended the original SBM framework to a multiclass formulation by defining matrices $\mathbf{D}_c$ for each operational class c. In this formulation a new input sample $\mathbf{x}_n$ is evaluated against each modeled condition as

$$\hat{\mathbf{x}}_{n,c} = \mathbf{D}_c \frac{\mathbf{w}_{n,c}}{\|\mathbf{w}_{n,c}\|_1} \tag{8}$$

with

$$\mathbf{w}_n = \left[\mathbf{D}_c \circ \mathbf{D}_c^{\mathrm{T}}\right]^{-1} \mathbf{D}_c \circ \mathbf{x}_n = \mathbf{G}_c^{-1}\mathbf{a}_{n,c}. \tag{9}$$

The input sample is then associated to the class c^* which produces the highest similarity score, that is

$$c^* = \arg\max_c \left\{s_{n,c}\right\} = \arg\max_c \left\{\mathbf{x}_n \circ \mathbf{x}_{n,c}\right\}. \tag{10}$$

B. Original SBM Training Phase

The training phase of an SBM model consists of selecting the prototype set for a target condition c. This procedure is critical to achieve the best performance, as using all the training samples would incur in a noisy and redundant representation, a high computational cost, and possible overfitting. In contrast, choosing an inadequate set $\mathcal{P}_c$ can lead to a poor representation and performance impairments, including underfitting.

The original SBM training proposition consists of an algorithm that attempts to select the minimal number of vectors that yields the same performance level as the complete set [12]. It consists of two steps:

1) First, it chooses as representatives the samples with index in the set $I = \{i_1, i_2, \dots, i_k\}$, such that

$$i \in I \quad \text{if} \quad \exists j : x_{ij} = \max_n\{x_{nj}\} \vee x_{ij} = \min_n\{x_{nj}\};$$

2) The other samples are ordered by their ℓ_2 norm in decreasing order and decimated by a factor of t. The remaining samples complement the prototype set $\mathcal{P}_c$.

In this manner, matrix $\mathbf{D}_c$ consists of samples containing the extrema of each feature (first step), and samples that have a certain difference on their ℓ_2 norm (second step). However, this last step can produce sub-optimal results, as two completely different vectors can have the same ℓ_2 norm [23]. Another critical choice of this algorithm is the decimation factor t: the number of representative samples can be expressed as $l = k + \lfloor (n-k)/t \rfloor$, which can produce a set larger than the optimal. To solve these issues, next section proposes a new approach, which transforms the prototype selection problem into a set cover problem.

C. Interpretable Prototype Selection Method

This method is based on of the prototype selection methods for interpretable classification presented in [13]. As previously described, selecting samples for $\mathbf{D}_c$ is equivalent to select a set of prototypes $\mathcal{P}_c$. Consider balls with radius τ centered in each point $\mathbf{x}_i$ from the training set $\mathcal{X}$. The best set of prototypes $\mathcal{P}_c \subseteq \mathcal{X}$ is a set of balls having the following properties [13]:

- *Property 1:* It should cover as many points from class c as possible;
- *Property 2:* It should cover as few points as possible from other classes;
- *Property 3:* It is sparse, using as few prototype as possible for a given τ;

This problem can be translated as a *set covering problem*. Given the set of points $\mathcal{X}$, the set covering problems seek the smallest subcover of $\mathcal{X}$ from the collection of sets that forms a cover of $\mathcal{X}$. If we take $B(\mathbf{x}) = \{\mathbf{x}' \in \mathbb{R}^m : d(\mathbf{x}', \mathbf{x}) < \tau\}$, which denotes the ball with radius $\tau > 0$ centered in $\mathbf{x}$ with distance d from $\mathbf{x}'$. The goal is to find the smallest subset $\mathcal{P} \subseteq \mathcal{X}$, $\mathcal{P} = \bigcup \mathcal{P}_c$, $\forall c$, such that $\{B(\mathbf{x}_i) : \mathbf{x}_i \in \mathcal{P}\}$ covers $\mathcal{X}$.

We can indicate when an instance belongs to the prototype set $\mathcal{P}$ by introducing variables α_j such that

$$\alpha_j = \begin{cases} 1, & \text{if } \mathbf{x}_j \in \mathcal{P}; \\ 0, & \text{otherwise.} \end{cases} \tag{11}$$

This problem can be described as [13]

$$\min \sum_{j=1}^{n} \alpha_j \quad \text{s.t} \quad \sum_{j:\mathbf{x}_i \in B(\mathbf{x_j})} \alpha_j \geq 1 \quad \forall \mathbf{x}_i \in \mathcal{X}, \tag{12}$$

where $\alpha_j \in \{0, 1\}$, $\forall \mathbf{x}_j \in \mathcal{X}$.

While Equation (12) represents Property 1, it does not address the remaining properties. Property 2 states that in certain cases some points from class c should be left uncovered as they would add points with labels $y \neq c$. Following [13], we adopt a *prize-collection set cover framework*, assigning a cost to each covering set, and penalties for each uncovered or incorrectly covered point. Then one finds the minimum-cost partial cover [24]. This can be described as

$$\min_{\alpha_j^{(c)}, \xi_i, \eta_i} \sum_i \xi_i + \sum_i \eta_i + \lambda \sum_{j,c} \alpha_j^{(c)}$$
$$\text{s.t.} \begin{cases} \displaystyle\sum_{j:\mathbf{x}_i \in B(\mathbf{x}_j)} \alpha_j^{(y_i)} \geq 1 - \xi_i, \quad \forall \mathbf{x}_i \in \mathcal{X}, \\ \displaystyle\sum_{\substack{j:\mathbf{x}_i \in B(\mathbf{x}_j) \\ c \neq y_i}} \alpha_j^{(c)} \leq \eta_i, \quad \forall \mathbf{x}_i \in \mathcal{X}, \\ \alpha_j^{(c)} \in \{0, 1\} \quad \forall j, i \quad \xi_i, \eta_i \geq 0 \quad \forall i, \end{cases} \tag{13}$$

where $\alpha_j^{(c)} \in \{0, 1\}$ indicates if $\mathbf{x}_j$ belongs to $\mathcal{P}_c$; ξ_i is a slack variable for the Property 1: if a training point from class c is not covered, $\xi_i = 1$; likewise, η_i counts the number of instances with $c \neq y_i$ that are within τ of $\mathbf{x}_i$; finally, $\lambda \geq 0$ is a parameter specifying the cost of adding a prototype [13].

In [13] two approaches for approximately solving this problem are discussed: one is based on linear programming relaxation with randomized rounding, and the other is a greedy approach. Here we present the latter, which is used in our prototype selection method.

Equation (13) minimizes the sum of the number of uncovered points, the number of incorrectly covered points, and the number of prototypes. We can then define a greedy algorithm which finds, at each step, the point $\mathbf{x}_j \in \mathcal{X}$ and class c for which the addition of $\mathbf{x}_j$ to $\mathcal{P}_c$ produces the maximum cost reduction. The incremental gain can be denoted by

$$\Delta L(\mathbf{x}_j, c) = \Delta\xi(\mathbf{x}_j, c) - \Delta\eta(\mathbf{x}_j, c) - \lambda \tag{14}$$

where

$$\begin{aligned} \Delta\xi(\mathbf{x}_j, c) &= \left| \mathcal{X}_c \bigcap \left(B(\mathbf{x}_j) \setminus \bigcup_{\mathbf{x}'_j \in \mathcal{P}_c} B(\mathbf{x}'_j) \right) \right|, \\ \Delta\eta(\mathbf{x}_j, c) &= \left| B(\mathbf{x}_j) \bigcap (\mathcal{X} \setminus \mathcal{X}_c) \right|. \end{aligned} \tag{15}$$

This procedure is described in Algorithm 1

Algorithm 1 Interpretable prototype selection algorithm [13]

function PROTOTYPE_SELECTION($\mathcal{X}$, $\mathcal{P}'$, τ),
  if $\mathcal{P}' = \emptyset$ **then**
    $\mathcal{P}' = \mathcal{X}$
  end if
  Start with $\mathcal{P}_c = \emptyset$ for each class c;
  while $\Delta L(\mathbf{x}^*, c^*) > 0$ **do**
    Find $(\mathbf{x}^*, c^*) = \arg\max_{(\mathbf{x}_j, c)} \Delta L(\mathbf{x}_j, c)$, $\mathbf{x}_j \in \mathcal{P}'$
    Let $\mathcal{P}_{c^*} \leftarrow \mathcal{P}_{c^*} \cup \{\mathbf{x}^*\}$
  end while
end function

III. EXPERIMENTAL METHODOLOGY

This section describes the experimental methodology used to evaluate the system performance and the proposed modifications. The proposed system, illustrated in Fig. 1, comprises three blocks: the *preprocessing* block converts the input to a new feature space; the SBM model returns the similarity between the test sample and each class; and a *classifier* that realizes the diagnosis. In this work a *random forest classifier* (RF) was employed for the last task.

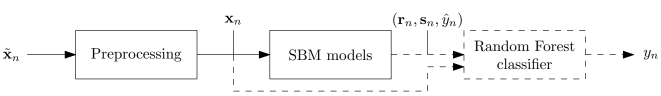

Fig. 1: Block diagram of the proposed system.

A. Database

Two databases were employed during this work to evaluate the performance of the SBM models: the MaFaulDa [14] and the CWRU bearing database [16].

1) *MaFaulDa* stands for Machinery Fault Database. It consists of 1951 scenarios acquired by eight sensors attached on a machinery fault simulator: six accelerometers, a tachometer, and a microphone. It covers the equipment normal operation and 5 faulty states: imbalanced operation, horizontal misalignment, vertical misalignment, and underhang or overhang bearing faults. This database is available for download at [14].
2) *The Case Western Reserve University* (CWRU) *bearing database* [16] consists of 161 scenarios divided in four categories, as described in [25]. Each scenario is assessed by three accelerometers: one on drive-end bearing, one on the fan-end bearing housing, and the last on the motor supporting base plate. This database is publicly available, and is widely used in the literature [4], [7], [8], [9], enabling comparisons with previous works.

B. Features

Before each sample was evaluated by the proposed system it was processed. This procedure transforms each original multivariate time-series scenario into a set of features that tries to capture the relevant information for the classification model in a compact form. This representation is also necessary to reduce the computational costs of the algorithm, reducing the original dimension of each sample.

Given the different natures of each database, they have undergone distinct preprocessing procedures, namely:

1) *MaFaulDa*: 5 Three types of features were extracted: the rotation frequency, *spectral features*, and *statistical features*:
   - The rotation frequency f_r was determined from the discrete Fourier transform of the tachometer signal, following the procedure detailed in [15], [6];
   - The spectral features correspond to magnitude of the spectrum of the other signals at the frequencies f_r, $2f_r$ and $3f_r$;
   - The statistical features computed for each signal are presented in Table I. As the signals in MaFaulDa were normalized to unit variance to reduce the dependence from the acquisition setup, features which are dependent from the variance were not considered in this case.
2) *CWRU Database*: In this case, statistical features presented on Table I were extracted, according to the procedure described in [4].

TABLE I: Statistical features taken from time (x_i) and spectral domain data (X_i) from each signal [4], [6], [15].

Time domain	
$\mu_x = \frac{1}{N}\sum_i^N x_i$	$\sigma_x^2 = \frac{1}{N}\sum_i^N (x_i - \mu_x)^2$
$H_x = -\sum_i^N P(x_i)\log P(x_i)$	$\kappa_x = \frac{1}{N}\sum_i^N \left(\frac{x_i-\mu_x}{\sigma_x}\right)^4$
$\gamma_x = \frac{1}{N}\sum_i^N \left(\frac{x_i-\mu_x}{\sigma_x}\right)^3$	$x_{\text{rms}} = \left(\frac{1}{N}\sum_i^N x_i^2\right)^{\frac{1}{2}}$
$x_{\text{sra}} = \left(\frac{1}{N}\sqrt{\lvert x_i\rvert}\right)^2$	$x_{\text{ppv}} = \max_i(x_i) - \min_i(x_i)$
$x_{\text{cf}} = \frac{\max_i(\lvert x_i\rvert)}{x_{\text{rms}}}$	$x_{\text{if}} = \frac{\max_i(\lvert x_i\rvert)}{\frac{1}{N}\sum_i^N \lvert x_i\rvert}$
$x_{\text{mf}} = \frac{\max_i(\lvert x_i\rvert)}{x_{\text{sra}}}$	$x_{\text{kf}} = \frac{\kappa_x}{x_{\text{rms}}^4}$
Spectral domain	
$\mu_X = \frac{1}{N}\sum_i^N X_i$	$X_{\text{rms}} = \left(\frac{1}{N}\sum_i^N X_i^2\right)^{\frac{1}{2}}$
$\sigma_X^2 = \frac{1}{N}\sum_i^N (X_i - \mu_X)^2$	

C. Training and Validation Methodology

This section presents the training and validation procedures used to evaluate the diagnosis system and the proposed modifications. The MaFaulDa database was randomly separated, respecting the classes distribution, in two disjoint training and test sets comprising 90% and 10% of the samples, respectively. The best set of parameters and the model performance were evaluated using a k-fold validation on the training set with $k = 10$. As performance metric to select the best model the weighted f_1-score between the classes was used. The best model was then evaluated on MaFaulDa test set and on the CWRU, by retraining the model using the same parameters in a k-fold fashion, to assess the system generalization power.

IV. Experimental Results and Discussion

A. Validation Results

This section presents the results obtained during the validation procedure. As described in the previous section, the best set of parameters for the employed models was selected using a k-fold cross-validation procedure on the MaFaulDa database. The validation considered all following options:

- Using the original SBM formulation, given in Eq. (5), or the AAKR formulation, considering $\mathbf{G} = \mathbf{I}$ in the same equation;
- The *SBM prototype selection method*, as described in Section II-B, with decimation factors $t \in [2, 21]$, or the proposed *interpretable prototype selection method*, with similarity radius $\tau \in [0.05, 1]$.

The similarity function presented in Eq. (4) was employed in all evaluated models. The best model obtained during the validation procedure was applied in the test set of each database to assess its generalization capability.

Table II presents the best model configuration in descending order of cross-validation f_1-score. One can notice that the proposed approach achieved a small but consistent increase in classification performance when compared to a model using the original SBM training phase or to a stand-alone random forest classifier.

TABLE II: Cross-validation f_1-score $(\%)$ comparing the best configuration for each model.

Model	Method	τ/t	F1-score $(\%)$
SBM+RFC	Proposed	0.953	99.24 ± 0.53
AAKR+RFC	Original	21	99.13 ± 0.65
RFC	–	–	99.08 ± 0.69

Even though the results are statistically equivalent, there are some advantages in the proposed procedure. First, the similarity score can be considered as measure of confidence in the SBM decision. Also, the residual can be used to observe how the decision was made, permitting an operator to interpret the model decision. More information can be obtained by observing the most similar representative state when assessing a decision. Lastly, the proposed modification adds value to the model, reducing its computational complexity and making the information contained on the prototypes more relevant.

B. Test Results

Table III presents the test results on each database. The results were generated using the best model obtained during the validation procedure. These results indicate that the proposed methodology is capable of generalizing for other samples as the parameters used for the CWRU were chosen on the MaFaulDa and the model still achieved higher accuracy than the one obtained in the original database.

TABLE III: Test accuracy $(\%)$ on each database.

Database	Accuracy
MaFaulDa	96.43%
CWRU	$98.7 \pm 0.76\%$

C. Comparison with Previous Works

Several other works in the literature addressed the problem of automatic classification of faults in rotating machines. The work presented in [6] employed multilayer perceptrons on the MaFaulDa database, achieving accuracy of 95.8%, inferior to the ones obtained with the proposed system presented in Table III.

For the CWRU database, even though there are many works using this database [25], it is very difficult to make a direct comparison, as most works do not present their results in a quantitative manner, only in a qualitative manner. As such, the comparison is restricted to a small set of works. In [7], kNN, naive Bayes, and SVM classifiers achieved accuracies of 98.83%, 98% and 98.97%, respectively. The SVM classifier found in [8] obtained accuracies above 98% for different rotation frequencies. The SVM and ELM classifiers using the procedure described in [9] achieved accuracies of 82.4% and 97.5%, respectively. Lastly, the kNN, SVM, and ANN classifiers using the feature selection method proposed in [4] obtained accuracies between 93% and 100%. From the previously presented results, one can conclude that the proposed SBM-based fault classifier achieves, for the CWRU database, competitive results when compared with the ones found in the literature. It is important to point out that, as demonstrated by the results over the MaFaulDa database, the proposed system is able to classify, with high accuracy, a wide range of machine faults, including misalignment and unbalanced faults.

V. Conclusion

In this work we presented an automatic fault classifier which employs similarity-based modeling to identify faults on rotating machines. The similarity model was used as a feature generator to a random forest classifier. We extend the similarity model for multiclass problems and we investigated the usage of a prototype-selection during the training procedure of the models. The system was evaluated in two databases: the MaFaulDa [14], a comprehensive database with multiple faults, and the CWRU bearing database [16], the current standard database for bearing fault diagnosis. The proposed system achieved accuracies of 96.43% on the MaFaulDa and 98.7% on the CWRU database, demonstrating the generalization power of the proposed system and competitive performance when compared with other works using the two databases.

Acknowledgments

This research was supported by CNPq and Petrobras. The authors would like to thank Dr. Kenneth A. Loparo and the Case Western Reserve University Bearing Data Center for providing the CWRU dataset for this study, and Dr. Wade A. Smith for his assistance.

References

[1] J. Liu, W. Wang, and F. Golnaraghi, “An enhanced diagnostic scheme for bearing condition monitoring,” *IEEE Transactions on Instrumentation and Measurement*, vol. 59, no. 2, pp. 309–321, 2010.

[2] P. Li, F. Kong, Q. He, and Y. Liu, “Multiscale slope feature extraction for rotating machinery fault diagnosis using wavelet analysis,” *Measurement*, vol. 46, no. 19, pp. 497–505, 2013.

[3] R. B. Randall and J. Antoni, “Rolling element bearing diagnostics – a tutorial,” *Mechanical Systems and Signal Processing*, vol. 25, no. 2, pp. 485–520, 2011.

[4] T. W. Rauber, F. de Assis Boldt, and F. M. Varejão, “Heterogeneous feature models and feature selection applied to bearing fault diagnosis,” *Mechanical Systems and Signal Processing*, vol. 62, no. 1, pp. 637–646, 2015.

[5] A. Boudiaf, A. Moussaoui, A. Dahane, and I. Atoui, “A comparative study of various methods of bearing faults diagnosis using the Case Western Reserve University Data,” *Journal of Failure Analysis and Prevention*, vol. 16, no. 2, pp. 271–284, 2016.

[6] D. Pestana-Viana, R. Zambrano-López, A. A. de Lima, T. d. M. Prego, S. L. Netto, and E. A. B. da Silva, “The influence of feature vector on the classification of mechanical faults using neural networks,” in *Proc. Latin American Symposium on Circuits and Systems*, 2016.

[7] B. Li, P. Zhang, D. Liu, S. Mi, G. Ren, and H. Tian, “Feature extraction for rolling element bearing fault diagnosis utilizing generalized S transform and two-dimensional non-negative matrix factorization,” *Journal of Sound and Vibration*, vol. 330, no. 10, pp. 2388–2399, 2011.

[8] S.-D. Wu, P.-H. Wu, C.-W. Wu, J.-J. Ding, and C.-C. Wang, “Bearing fault diagnosis based on multiscale permutation entropy and support vector machine,” *Entropy*, vol. 14, no. 8, pp. 1343–1356, 2012.

[9] Y. Li, X. Wang, and J. Wu, “Fault diagnosis of rolling bearing based on permutation entropy and extreme learning machine,” in *Proc. Chinese Control and Decision Conference*, 2016.

[10] R. M. Singer, K. C. Gross, J. P. Herzog, R. W. King, and S. Wegerich, “Model-based nuclear power plant monitoring and fault detection: Theoretical foundations,” in *International Conference on Intelligent Systems Applications to Power Systems*, July 1997.

[11] S. Garcia, J. Derrac, J. Cano, and F. Herrera, “Prototype selection for nearest neighbor classification: Taxonomy and empirical study,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 3, pp. 417–435, 2012.

[12] J. P. Herzog, S. W. Wegerich, K. C. Gross, and F. K. Bockhorst, “MSET modeling of Crystal River-3 venturi flow meters,” in *International Conference on Nuclear Engineering*, 1998.

[13] J. Bien and R. Tibshirani, “Prototype selection for interpretable classification,” *The Annals of Applied Statistics*, vol. 5, no. 4, pp. 2403–2424, 2011.

[14] “MaFaulDa - Machinery Fault Database,” http://www02.smt.ufrj.br/~offshore/mfs/, 2016, accessed March 14, 2017.

[15] A. A. de Lima, T. d. M. Prego, S. L. Netto, E. A. B. da Silva, R. H. R. Gutierrez, U. A. Monteiro, A. C. R. Troyman, F. J. d. C. Silveira, and L. Vaz, “On fault classification in rotating machines using Fourier domain features and neural networks,” in *Proc. Latin American Symposium on Circuits and Systems*, 2013.

[16] K. A. Loparo, “Bearings vibration data set, Case Western Reserve University,” http://csegroups.case.edu/bearingdatacenter/home, 2003, accessed November 25, 2017.

[17] J. Mott and M. Pipke, “Similarity-based modeling of aircraft flight paths,” in *IEEE Proc. Aerospace Conference*, vol. 3, 2004.

[18] S. W. Wegerich, “Similarity based modeling of vibration features for fault detection and identification,” *Sensor Review*, vol. 25, no. 2, pp. 114–122, 2005.

[19] F. A. Tobar, L. Yacher, R. Paredes, and M. E. Orchard, “Anomaly detection in power generation plants using similarity-based modeling and multivariate analysis,” in *Proc. American Control Conference*, vol. 3, 2011.

[20] S. W. Wegerich, A. D. Wilks, and R. M. Pipke, “Nonparametric modeling of vibration signal features for equipment health monitoring,” in *IEEE Proc. Aerospace Conference*, vol. 7, 2003.

[21] S. W. Wegerich, “Similarity-based modeling of time synchronous averaged vibration signals for machinery health monitoring,” in *IEEE Proc. Aerospace Conference*, vol. 6, 2004.

[22] J. Garvey, D. Garvey, R. Seibert, and J. W. Hines, “Validation of on-line monitoring techniques to nuclear plant data,” *Nuclear Engineering and Technology*, vol. 39, no. 2, pp. 133–142, 2006.

[23] P. Guo and N. Bai, “Wind turbine gearbox condition monitoring with aakr and moving window statistic methods,” *Energies*, vol. 4, no. 11, pp. 2077–2093, 2011.

[24] J. Könemann, O. Parekh, and D. Segev, “A unified approach to approximating partial covering problems,” *Algorithmica*, vol. 59, no. 4, pp. 489–509, 2011.

[25] W. A. Smith and R. B. Randall, “Rolling element bearing diagnostics using the Case Western Reserve University data: A benchmark study,” *Mechanical Systems and Signal Processing*, vol. 64-65, pp. 100–131, 2015.