Unsupervised Ranking of Outliers in Wind Turbines via Isolation Forest with Dictionary Learning

Sergio Martin-del-Campo[1] and Kammal Al-Kahwati[2]

[1] *EISLAB, Luleå University of Technology, Luleå, 97187, Sweden*
sergio.martindelcampo@ltu.se
kamal.alkahwati@gmail.com

Abstract

Predictive maintenance strategies for the detection of faults in wind turbines require approaches that consider the limited human resources available responsible for the final assessment of the machine. Here, we present an unsupervised framework for the ranking of wind turbines (assets) in a wind farm (fleet) based on the detection of faults in the monitored machine elements, which will help experts determine the turbine that has higher priority in further machine diagnostics. Previous work has shown that the use of sparse coding with dictionary learning enables the identification of faults in rolling element bearings. However, it has not shown how the information of the identified faults can be used in an unsupervised strategy that enables a detector to provide some sort of recommendation on how to proceed with the information of the detected faults. We describe how features derived from the sparse coding with dictionary learning method are used together with the isolation forest outlier detection algorithm to create a score for the ranking of the monitored assets. We consider scenarios where all the turbines are evaluated together and each of them individually in the creation of the ranking and we compare these results with a condition where features taken from the time-domain are considered. Sparse coding with dictionary learning together with isolation forest produces an anomaly score that can be used to rank wind turbines by their need for a maintenance action given the presence of faults in their systems resulting in an unsupervised warning system that can support the work of maintenance experts.

1. Introduction

The demand for reliable condition monitoring systems on rotating machinery for power generation is continuously increasing due to the wider use of wind power as an energy source. Simultaneously, the availability of human resources who perform the evaluation and diagnostics is not increasing at the same rate. In addition, the wind farms are typically located in remote locations where access is difficult and the transmission of data to other sites for evaluation is expensive. This situation results in a need for condition monitoring systems that are capable to detect the complex and weak signatures of the faults and can provide the information needed with the limited communication capacity. These diagnosis systems should support experts in the decision process to monitor a bearing and machine condition and act as an early warning system, while requiring a minimum of manual configuration.

Machine maintenance strategies have evolved over time to increase efficiency, while the organizations owning them evaluate whether the loss of an asset or the interruption of its operation would be critical to overall production and performance (Randall, 2011). Originally, bearings would continue to be used until failure under a strategy called corrective maintenance. Under a preventive maintenance strategy, a period of time or number of cycles interval is defined and an expert evaluates the condition at the end of this interval. The length of this interval is based on previous experience and calculations of bearing lifetime endurance. A detriment of this strategy is the length of this interval, which can be longer than some fault processes and it is increasing given the limited human resources available (Harris & Kotzalas, 2007). Condition-based maintenance or predictive maintenance aims to plan maintenance actions based on the collected information. Thus, avoiding unnecessary maintenance through fault diagnostics and estimation of remaining useful life. Under this strategy, condition-based maintenance technologies monitor the condition of a machine continuously or periodically (Jardine et al., 2006). Continuous monitoring requires the definition of a series of alarms that can be applied to the raw signals and which are typically based on time or frequency domain features. These alarms are triggered once something wrong is detected with the detection of a value out of the alarm level. Periodic monitoring can be used with more sophisticated methods derived from machine learning to diagnose the machine condition. However, because of the computational demands of these methods, the monitoring cannot be done in the raw signal and it requires the definition of an interval upon which do the evaluation. Therefore, periodic predictive maintenance faces similar challenges as preventive maintenance.

According to Dias Machado de Azevedo et al. (2016), an efficient condition monitoring system for wind turbines needs to consider the following challenges: ease of automation, ample warning time and the behavior of each turbine. Ease of automation enables scalable solutions that can be applied to a wide number of turbines. The need for ample warning time in the form of an early warning system is relevant not only for the opportunity to introduce measures that extend the lifespan of the bearing but also because replacements can take a long time before they become available, and longer stopping times are not desirable. Finally, many of the alarms that permit the continuous monitoring of the wind turbine are designed for that specific turbine under its specific environmental and operational conditions and not all of them can be translated from one asset to another.

Machine learning solutions for condition monitoring can help to alleviate the above challenges. The number of works that consider machine learning as a tool for advanced condition monitoring and early diagnosis in wind turbines is increasing, see Hossain et al. (2018) for a recent review. Many of the strategies proposed use time-domain features that are known, frequency-domain features or features derived from signal processing techniques such as the envelope or wavelet analysis. These approaches require proper selection of the features that are introduced into a machine learning algorithm that predicts the fault. The resulting constraint is a model tuned to detect a specific fault or a feature set that enables fault detection under specific conditions. Thus, wind turbine condition monitoring and their experts can benefit from an unsupervised machine learning approach that requires a minimum of human intervention in their configuration, can adapt to the environmental and operational conditions and provides an early warning. Our method aims to address these concerns with the adoption of sparse coding with dictionary learning to process the raw vibration signals and the use of the resulting model together with isolation forest for outlier detection to generate a ranking of the assets deviating the most from the fleet behavior and that might require further investigation to identify the cause of a likely fault.

Former work in the area of machine learning approaches to the condition monitoring of wind turbines aiming to improve the maintenance strategies has focused on the modeling of the wind turbine or its elements while using simulated or process data. In these approaches, preventive maintenance is carried out to reduce downtime losses rather than to enhance the resources available. Furthermore, none of these studies had used real-world raw wind turbine data in their modeling. Zhang et al. (2017) proposes a two-level maintenance threshold strategy to merge maintenance actions of wind turbines with an imperfect schedule during machine shutdown. They propose a hybrid model that determines the elements most likely to require maintenance giving their deterioration up to the time of the machine shutdown. A decision-making procedure to determine the most suitable time to perform maintenance is proposed by Ghamlouch et al. (2019). Their procedure models the deterioration of the system while introducing an uncertainty element to represent the stochastic environmental and operational conditions of the wind turbine, which results in a recommendation on the most suitable time to perform the maintenance action. An approach based on artificial neural networks that use SCADA data is proposed by Bangalore et al. (2017). Their proposed approach is capable to detect faults in the gearbox of a wind turbine but requires the definition of multiple models to handle different operational conditions. H. Liu et al. (2011) uses an unsupervised machine learning approach based on sparse coding with dictionary learning to detect the presence of faults in a bearing. This approach considers fault detection as a classification problem focusing entirely on whether a fault is present.

In this paper, we utilize an unsupervised feature learning method called sparse coding with dictionary learning on raw vibration signals from wind turbines to derive a set of features, which are in turn used on an isolation forest outlier detection method to rank wind turbines needing maintenance actions. Sparse coding with dictionary learning models a vibration signal as a linear superposition of atomic waveforms plus noise. Our interest here focuses on the fidelity of the sparse representation, which can be used to reconstruct the original vibration signal, and in the dictionary distance measure of learned dictionary as it propagates over time. Sparse coding with dictionary learning has proven useful for online monitoring (Martin-del-Campo et al., 2013) and identification of bearing characteristic frequencies Martin-del-Campo et al. (2019). The work presented here is novel because it presents a score that can be used to rank wind turbines on how much the operation is diverging from the population without incorporating prior information on the state of the machine. We observe lower scores at periods when a fault is present in a turbine than when it operates in healthy conditions. Under the presence of a fault in the rolling element bearing of the bearing, the distribution of the fidelity and dictionary distance features diverge significantly than under healthy bearing conditions. Furthermore, we compare the features derived from the sparse coding and dictionary learning to time-domain features under the isolation forest outlier detection strategy. These results indicate that the features derived from the sparse coding with dictionary learning method are suitable for the detection of outliers of a wind turbine within a wind farm and can be used together to the isolation forest method to rank the turbines most needing of a maintenance action.

2. METHODOLOGY

We present our ranking framework with an introduction of the sparse coding with dictionary learning unsupervised method, a description of used features and the background of the isolation forest algorithm. Figure 1 shows a diagram of the framework, which presents the stages that form the framework and how each stage is related to each other. The input is the raw vibration signal from a rolling element bearing in the wind turbine, and the output is a score derived from the isolation forest algorithm.

2.1. Sparse Coding with Dictionary Learning

The sparse coding with dictionary learning method generates succinct representations of signals. This means that the resulting sparse representation holds a minimum of space while still useful and informative for analysis. Here, we use a model developed by Smith & Lewicki (2006), which is inspired by the earlier work of Olshausen & Field (1997). Smith & Lewicki (2006) describes how the learned waveforms, referred to as *atoms*, from speech data mirror cochlear revcor functions. The operating hypothesis underlying our work is that features, which characterize the vibration signals originating from rotating machines, can be learned with a similar procedure. A sparse coding model represents a signal $x(t)$ as a linear superposition of noise and atomic waveforms with compact support

$$x(t) = \epsilon(t) + \sum_i a_i \phi_{m(i)}(t - \tau_i). \tag{1}$$

The atoms $\phi_m(t)$ represent shift-invariant morphological features of the input signal with a_i and τ_i specifying the amplitude and the shift (temporal position) respectively. The values of a_i and τ_i are calculated using the matching pursuit (MP) algorithm (Mallat & Zhang, 1993) and the triple $m(i), a_i, \tau_i$ represents one *atomic event*. A collection of atoms is known as a *dictionary*

$$\Phi = \{\phi_1, \cdots, \phi_M\}. \tag{2}$$

where M expresses the number of atoms in the dictionary.

The optimization of the waveforms (atoms) is carried out via an unsupervised approach using an update rule based on the gradient ascent of the approximate log data probability (Smith & Lewicki, 2006)

$$\frac{\partial}{\partial \phi_m} \log\left[p(x \mid \Phi)\right] = \frac{1}{\sigma_\epsilon^2} \sum_i a_i (x - \hat{x})_{\tau_i}, \tag{3}$$

where $(x - \hat{x})_{\tau_i}$ is the residual from the matching pursuit algorithm over the support of atom ϕ_m at time τ_i with an atom amplitude of a_i. This rule implies that the shape and length of each atom are adapted from a weighted average of the residuals of the matches identified by the matching pursuit algorithm. The sparseness of the representation is defined by the stop condition of the matching pursuit algorithm. It is important to note that the resulting sparse representation is not a linear function of the input signal because the matching pursuit algorithm is non-linear.

The optimization of the dictionary Φ is done iteratively. The procedure begins with the initialization of the dictionary. In the beginning, each atom in the dictionary has an initial length of fifty with amplitudes sampled from a Gaussian distribution plus zero padding. The matching pursuit algorithm operates as the cross-correlation of the vibration signal (residual) with all atoms in the dictionary. The maximum cross-correlation sets one event, $m(i), a_i, \tau_i$. The corresponding waveform, $a_i \phi_{m(i)}(t - \tau_i)$ is subtracted from the signal. The resulting residual is used as input in the next iteration of the matching pursuit algorithm. The procedure continues until reaching the stop condition, which is defined by the sparsity level that is the number of events per signal sample.

The main challenge and area of opportunity of this approach is learning the dictionary Φ. The result is an approach that is fundamentally different from other condition monitoring approaches like Fourier and wavelet analysis. Dictionary learning aims to learn a dictionary of atoms Φ that maximizes the expectation of the log data probability

$$\Phi = \arg\max_\Phi \langle \log\left[p(x \mid \Phi)\right] \rangle, \tag{4}$$

where

$$p(x \mid \Phi) = \int p(x \mid a, \Phi) p(a) da. \tag{5}$$

The prior of the amplitude, $p(a)$, is defined to promote a sparse representation in terms of statistically independent atoms (Olshausen & Field, 1997). The integral is approximated with the maximum a posteriori estimate resulting from the matching pursuit algorithm. The result is a learning algorithm that uses gradient ascent on the approximate log data probability defined by Eq. (3). The gradient of each atom in the dictionary is proportional to the sum of residuals corresponding to the activation of that atom. The term σ_e^2 is the variance of the residual that remains after the matching pursuit algorithm. Furthermore, we introduce a *learning rate* parameter η so that Eq. (3) is modified to

$$\Delta \phi_m = \frac{\eta}{\sigma_e^2} \sum\nolimits_{i \,:\, m = m(i)} a_i (x - \hat{x})_{\tau_i}. \tag{6}$$

The resulting adaptation rate of each atom depends on the matching-pursuit activation rate. Consequently, this implies that some atoms may adapt slowly or not at all. The motivation behind our approach is the relatively low complexity and simplicity of the algorithm.

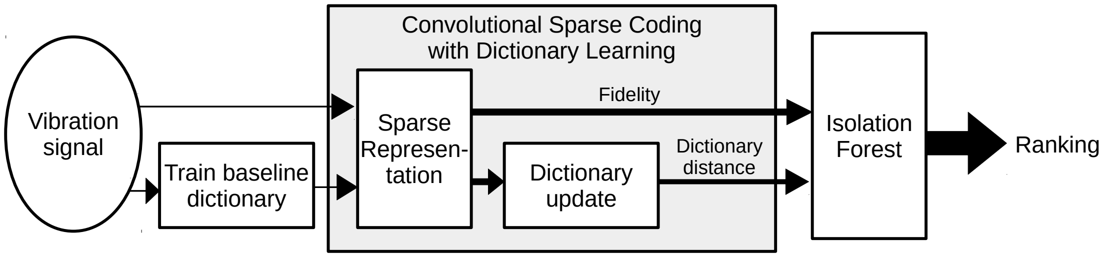


Figure 1. Diagram of our proposed ranking framework. Features derived from the Convolutional Sparse Coding with Dictionary Learning method are used with the Isolation Forest algorithm to generate the ranking.

2.2. Fidelity and Dictionary Distance

The features that we consider from the sparse coding with dictionary learning method are fidelity and dictionary distance. We define the fidelity of the sparse representation as the ratio between the sparse approximation and the signal residual

$$dB = 20log_{10}\left(\frac{\hat{x}}{\epsilon(t)}\right), \tag{7}$$

where dB is the fidelity in decibels, $\hat{x}$ is the sparse approximation of each signal segment and $\epsilon(t)$ is the residual. The residual is the difference between the input signal and the resulting sparse representation when the stop condition of the matching pursuit is fulfilled.

The dictionary distance feature quantifies the similarity of one learned dictionary at two different points in time. The first point in time is fixed and it is the initial dictionary, which is trained during a period when the machine was known to be in a healthy condition. The second point in time is the updated dictionary after the completion of the matching pursuit algorithm, as it propagates over time. Thus, the updated dictionary is compared against the initial dictionary ϕ^0 to quantify the difference between both. The procedure is repeated for each new signal segment. We quantify the difference between the two dictionaries using a distance measure proposed by Skretting & Engan (2011) and defined as

$$\beta(\Phi^k, \Phi^0) = \frac{1}{2M}\Big(\sum_{i=1}^{M}\beta(\Phi^k, \phi_i^0) + \sum_{j=1}^{M}\beta(\Phi^0, \phi_j^k)\Big), \tag{8}$$

where both dictionaries have the same number of atoms M, and the membership of each dictionary is $\phi^k \in \Phi^k$ and $\phi^0 \in \Phi^0$. The function β indicates the maximum similarity of an atom ϕ to the atoms in dictionary Φ and it is given by

$$\beta(\Phi, \phi) = \arccos \mu(\Phi, \phi). \tag{9}$$

The function μ is the mutual coherence, which is defined as Elad (2010):

$$\mu(\mathbf{\Phi}, \phi_i) = \max_{\substack{\forall \phi_j \in \Phi \\ j \neq i}} \frac{|\phi_i^T \phi_j|}{\|\phi_i\|_2 \cdot \|\phi_j\|_2}. \tag{10}$$

The unit of the dictionary distance measure is degrees.

2.3. Time-domain features

We consider three time-domain features in our comparison. First, we consider the root mean square (RMS) of each signal segment. We calculate the RMS value in the following way

$$RMS = \left(\frac{\sum_{n=1}^{N} x_n^2}{N}\right)^{\frac{1}{2}}, \tag{11}$$

where N is the total number of samples of the signal segment with values x_n. The second feature we consider is the crest factor. The crest factor is used to describe how extreme is the peak value in a signal Patidar & Soni (2013). The crest factor value is calculated using the equation

$$CF = \frac{|x_{peak}|}{RMS}, \tag{12}$$

where x_{peak} is the peak sample value. The third time-domain feature that we consider is kurtosis. Kurtosis describes the shape of the distribution of the sample values of a signal segment Patidar & Soni (2013) and it is a commonly used feature in the diagnostics of gearboxes Qu et al. (2014). We estimate the kurtosis value using

$$K = \frac{1}{\sigma_4}\left(\frac{\sum_{n=1}^{N}(x_n - \mu)^4}{N}\right), \tag{13}$$

where σ is the standard deviation and μ is the mean value of the signal segment.

2.4. Isolation Forest

The isolation forest algorithm was first proposed by F. T. Liu et al. (2008). It operates under the principle that anomalous points in a dataset are minority and as such, these points have a different attribute value. The algorithm has a linear time complexity and runs on a constant memory requirement.

Isolation forest operates by recursively partitioning the data, until a single point has been isolated. The result is that suspected anomalous points are isolated much faster than regular points. The algorithm is a tree-based ensemble method that uses a random tree structure to partition the data. The path length of the collection of these random trees is stored for the entire dataset. A shorter path length is used to indicate that the point is an outlier. The anomaly score for each point is calculated using

$$s = 0.5 - 2^{-\frac{E(h(x))}{c(n)}}, \tag{14}$$

where $E(h(x))$ is the average value of the path length $h(x)$ for data point x and $c(n)$ is the average path length of unsuccessful searches in the binary search tree under n number of nodes. The resulting anomaly score covers the range $[-0.5, 0.5]$ where a value closer to 0.5 is considered the most anomalous.

3. REAL-WORLD CASE STUDY

3.1. Description

We apply our unsupervised ranking methodology to vibration data from a real-world condition monitoring system. The source of the data is a database that collects the information from the condition monitoring system of a wind farm located in northern Sweden. The wind farm is formed by six similar wind turbines. Each wind turbine has a three-stage gearbox that includes two sequential planetary stages, followed by a helical gear stage. An accelerometer mounted on the housing of the output shaft bearing measures the raw time-domain vibration signals in the axial direction. Raw vibration signal recordings have a sampling frequency of 12.8 kHz. Each vibration signal segment is 1.28 seconds long and they are recorded at intervals of approximately 12 hours over a period of 46 consecutive months. The data is made publicly available by the Luleå University of Technology (Martin-del-Campo et al., 2018).

In the period for which the data was made available to this investigation, two out of the six wind turbines experienced a form of fault. We refer to one of these turbines as Turbine 2 and it experienced an electrical fault, where the vibration sensor was not installed properly at the beginning of operation of the wind turbine and resulted in an improper recording of the vibration data. However, the wind turbine was healthy during the electrical fault and remained to be healthy during the entirety of the investigation. The second turbine, which we refer to as Turbine 5, experienced two faults related to the bearings located in the gearbox. The first fault was an inner raceway failure on a four-point ball bearing on the output shaft that resulted in the replacement of the bearing after 1.2 years of operation. The second failure occurred in one of the cylindrical roller bearings supporting one of the planets in the first planetary gear of the gearbox. This failure required the replacement of the entire gearbox after two years of operation of the wind turbine. The remaining wind turbines were completely healthy during the entire period of our investigation. During the processing of the vibration signals with the sparse coding with dictionary learning method, we filtered signal segments that corresponded to an unloaded condition of the wind turbine. Afterwards, a baseline initial dictionary was trained using 5000 segments of one-second duration (12800 samples) from a period of time where the wind turbine operated in healthy conditions. After the baseline initial dictionary was learned, it was propagated over time using all signal segments where the wind turbine was loaded. A similar protocol was followed for all six wind turbines.

The vibration data is processed with our own implementation of matching pursuit along with the dictionary learning algorithm presented by Smith & Lewicki (2006). Each signal segment is preprocessed to have zero mean with unit variance. The stop condition of the matching pursuit algorithm is a sparsity level of 90%, which is equivalent to a data compression ratio of 0.1. Therefore, each signal segment, which is 16384 samples long, was modeled using 1600 atomic events. The dictionary contains eight atoms and during the dictionary update step, we use a step size of $\eta = 10^{-6}$. The atoms are normalized after each learning iteration. Details of the wind turbine dataset and data collection conditions, as well as method evaluation setup is available in (Martin-del-Campo et al., 2019).

The fidelity is estimated using the resulting sparse representation of the matching pursuit algorithm and the dictionary distance measure is calculated using the resulting dictionary from the dictionary update step, which is compared against the initial dictionary. Both measures are used as inputs to the isolation forest algorithm. We use a contamination factor of 0.01 in the isolation forest algorithm to further isolate each of the samples. Our interest is in the resulting anomaly score for the ranking and not in the labels that the algorithm generates. We apply the isolation forest algorithm in two different scenarios. The first scenario considers the data from all the turbines aggregated together and the isolation forest is applied to all of it. This scenario considers a situation where the data analytics can be carried out in a remote station that can collect the data from all the turbines. The second scenario considers the data from each turbine individually and the isolation forest algorithm is applied to each of them. This scenario intends to replicate a condition where the data analytics is carried out directly in the condition monitoring system of the wind turbine and the data cannot be transferred to a remote station.

The framework aims to rank the turbines in the order where further assessment for further diagnostics is required. Therefore, turbines that have some sort of imperfect should come at the top of the rank, independently of the type of problem that the turbine might have. The anomaly score resulting from the isolation forest algorithm is used to create this rank. A lower score is indicative of an outlier, which results in a guideline where the lower the anomaly score of the signal segment, the higher the ranking of the wind turbine. The feature-pair evaluated with the isolation forest algorithm include three cases: dictionary distance and fidelity; RMS and kurtosis; and crest factor and kurtosis. The feature-pair of RMS and crest factor is not considered given the high correlation between both variables. Figure 2 shows the scatter plots of the three feature cases in the scenario where the feature values of all the turbines are aggregated into a single plot. The case that uses dictionary distance and fidelity as the feature-pair is shown in Figure 2a. It can be seen that a normal condition encompasses a specific high-density region within the fidelity and dictionary distance range and values marked as outlier are outside that high-density range. Meanwhile, the spread of the values for the cases of RMS, crest factor and kurtosis is higher and outlier values tend to be at the edges of those spreads.

Table 1 presents the lowest ten anomaly scores for the three feature-pairs cases evaluated. Next to the scores, the turbine to which the score is associated is included. The table highlights the scores that correspond to signal segments where it is known the asset is faulty. Each anomaly score corresponds to different signal segments. All anomaly scores are shown for the cases of RMS and kurtosis, and crest factor and kurtosis point to turbines which were completely healthy during the period under investigation. The case that uses the dictionary distance and fidelity features shows anomaly scores corresponding to the two turbines that registered some type of fault under the period of investigation. The top ten lowest anomaly scores in the dictionary distance and fidelity case correspond to signal segments recorded in the period of one month prior to the fixing of the sensor fault in Turbine 2 and the bearing replacement in Turbine 5.

Table 1. Anomaly scores for the aggregated scenario.

Dict. distance Fidelity		**RMS Kurtosis**		**Crest Factor Kurtosis**	
Turbine	Score	Turbine	Score	Turbine	Score
2	**-0.109**	6	-0.057	4	-0.092
2	**-0.096**	3	-0.055	4	-0.091
2	**-0.082**	6	-0.053	4	-0.090
2	**-0.079**	3	-0.053	4	-0.089
5	**-0.078**	6	-0.052	4	-0.085
2	**-0.077**	4	-0.052	4	-0.080
5	**-0.077**	6	-0.051	4	-0.077
5	**-0.076**	3	-0.050	4	-0.075
5	**-0.075**	3	-0.048	4	-0.074
5	**-0.074**	6	-0.045	4	-0.071

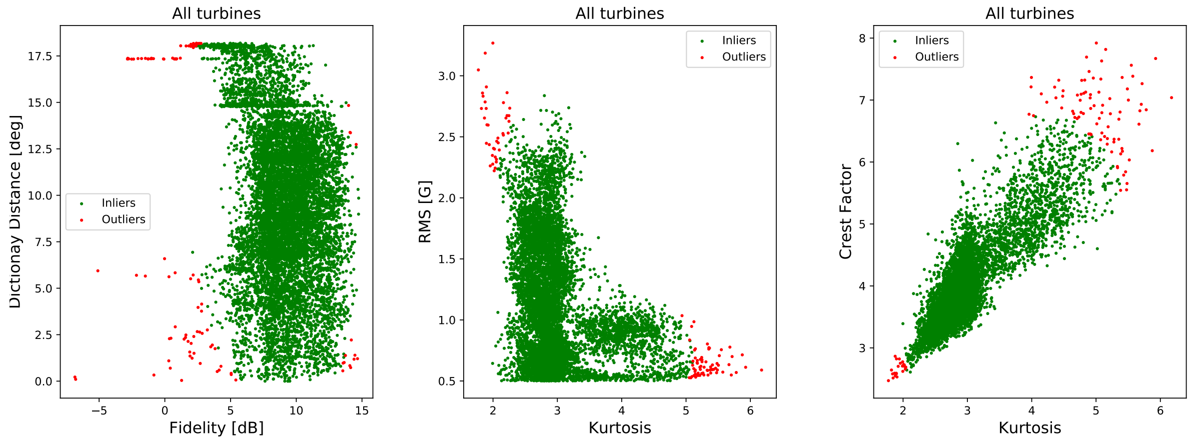
3.2. Results of the individual scenario

In the individual scenario, the isolation forest algorithm is applied to the feature values individually to each turbine. The aim remains the same, to create a turbine ranking based on the resulting anomaly scores. Figure 3 shows the scatter plots of the six turbines for the case of the dictionary distance and fidelity features. Similar to Figure 2a, the healthy signal segments of all turbines lie within a positive range of dictionary distance and fidelity values. However, it becomes evident how the outliers found in Turbine 2 and Turbine 5 are outside the positive value range, while the segments marked as outliers in the healthy turbines are within the positive range of the feature values. The feature-pair cases that use RMS and kurtosis, and, crest factor and kurtosis are shown in Figure 4 and Figure 5, respectively. In both cases, the results are different and it is not clear the presence of a high density region within which all the segment values are located. In the case shown in Figure 4, the distribution of the values for Turbine 4 has a larger spread than all the other turbines and Turbine 4 is a healthy turbine with no reported faults. The case shown in Figure 5 is a little different, the spread of the values has a similar shape across the six turbines, even if the range is not the same. In this case, it becomes easier to appreciate outliers far apart from where the main values are located in Turbine 2 and Turbine 5.

Table 2. Lowest anomaly score for each turbine.

Turbine	**Dict. distance Fidelity**	**RMS Kurtosis**	**Crest Factor Kurtosis**
1	-0.033	-0.044	-0.081
2	**-0.098**	-0.055	**-0.102**
3	-0.039	**-0.087**	-0.080
4	-0.035	-0.071	-0.060
5	**-0.054**	-0.085	**-0.127**
6	-0.043	**-0.092**	-0.064

Table 2 presents the lowest anomaly score for each turbine under the three feature sets while highlighting the lowest two scores on each feature-pair. The case that uses RMS and kurtosis as features shows the lowest anomaly score belonging to Turbine 6, followed by Turbine 3. Both of these turbines are healthy and did not show any faults during the period under investigation. The cases that use dictionary distance and fidelity and crest and kurtosis as features have a similar response. Both of these cases point towards Turbine 2 and Turbine 5 with the lowest anomaly scores and as a result a higher priority in the ranking. Both of these turbines experienced a form of fault and should have a higher ranking in the priority for further evaluation and diagnostics of the wind turbine.



(a) **Dict. distance vs. Fidelity** (b) **RMS vs. Kurtosis** (c) **Crest Factor vs. Kurtosis**

Figure 2. Scatter plots of the aggregated scenario for the selected features, which includes the data from the six wind turbines. Signal segments labeled as outliers are marked in red and segments labeled as normal in green.

4. DISCUSSION

This work focuses on creating a ranking of assets (wind turbines) within a fleet (wind farm) for further evaluation and diagnostics in an unsupervised way. In particular, we use the sparse coding with dictionary learning method together with the isolation forest algorithm to generate a set of anomaly scores, which are used to rank the wind turbines requiring further evaluation. The aim is to provide support to vibration and condition monitoring experts on the selection of the assets that need further study when the assets are most needed for further evaluation instead of defining a time or number of cycles for evaluation, which is typical in preventive maintenance strategies. We find that the framework proposed in this work highlights wind turbines with faults over healthy turbines by giving them a lower anomaly score on the period where the fault occurs. In a scenario where the data of all turbines is aggregated for the study, using dictionary distance and fidelity as features provide significantly better results than using time-domain features. In a scenario where the scores are generated individually for each wind turbine, using dictionary distance and fidelity or crest factor and kurtosis as features provide equally useful results. These results motivate further investigation of the framework in other operational conditions to determine if this unsupervised framework is scalable to different applications. An improvement where other distance measurements are considered could help in taking into consideration issues related to concept drift where the properties of the system are changing over time. Sparse coding with dictionary learning is an interesting approach to condition monitoring automation given that it enables the reduction of the amount of data to transmit without significant loss of information and requiring few assumptions about the machine or structure of the signal. This method together with isolation forest further permits detecting the presence of signal outliers, which might be representative of a fault condition in a machine element of the wind turbine. Both of these methods together permit the detection of faults in wind turbines and result in a reduction of the workload of experts whose responsibilities will be redirected into the assessment of the fault nature.

ACKNOWLEDGMENT

This work is supported by SKF via their University Technology Center at LTU. We thank Fredrik Sandin and Stephan Schnabel for the engaging discussions and valuable feedback.

REFERENCES

Bangalore, P., Letzgus, S., Karlsson, D., & Patriksson, M. (2017). An artificial neural network-based condition monitoring method for wind turbines, with application to the monitoring of the gearbox. *Wind Energy*, *20*(8).

Dias Machado de Azevedo, H., Araujo, A. M., & Bouchonneau, N. (2016). A review of wind turbine bearing condition monitoring: State of the art and challenges. *Renewable and Sustainable Energy Reviews*, *56*, 368–379.

Elad, M. (2010). *Sparse and redundant representations: from theory to applications in signal and image processing*. Springer.

Ghamlouch, H., Fouladirad, M., & Grall, A. (2019). The use of real option in condition-based maintenance scheduling for wind turbines with production and deterioration uncertainties. *Reliability Engineering & System Safety*, *188*.

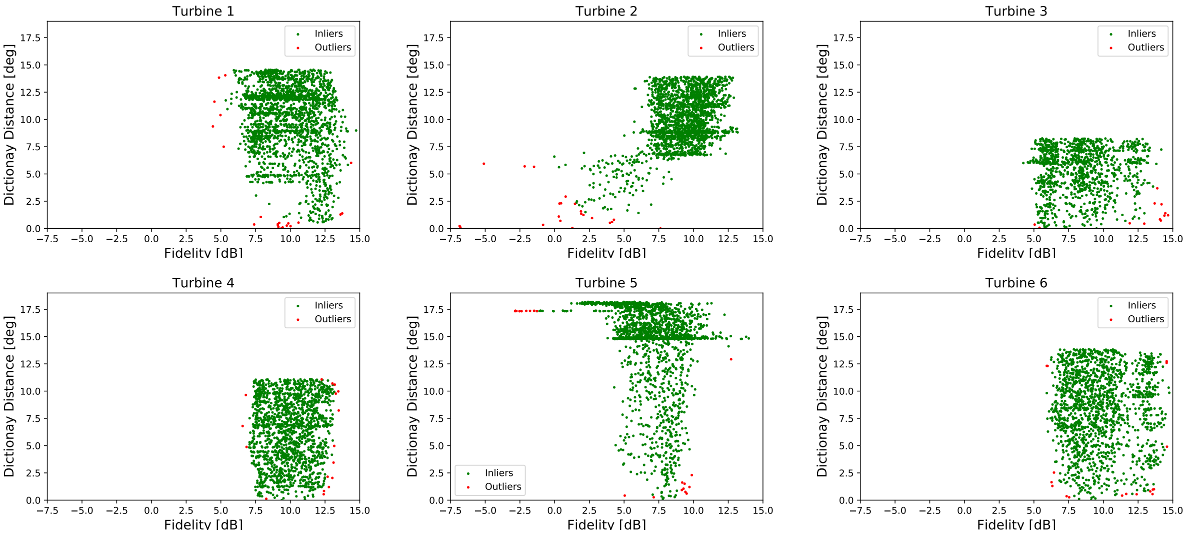


Figure 3. Individual scatter plots of all turbines for the feature-pair: **dictionary distance and fidelity**. Signal segments labeled as outliers are marked in red and segments labeled as normal in green. All plots have the same limits.

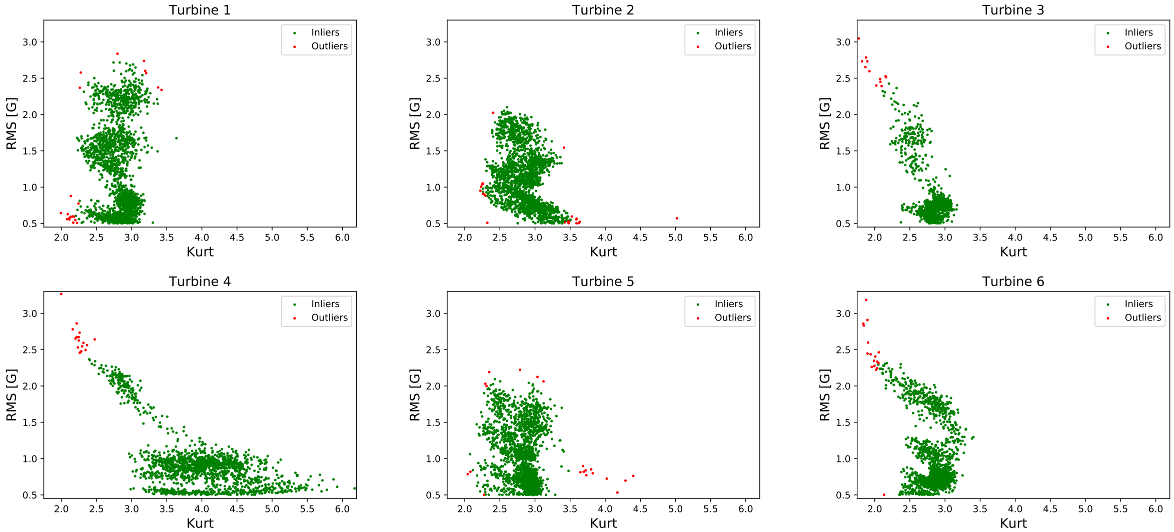


Figure 4. Individual scatter plots of all turbines for the feature-pair: **RMS and kurtosis**. Signal segments labeled as outliers are marked in red and segments labeled as normal in green. All plots have the same limits.

Harris, T. A., & Kotzalas, M. N. (2007). *Essential concepts of bearing technology* (5th ed.). Taylor and Francis.

Hossain, M. L., Abu-Siada, A., & Muyeen, S. M. (2018). Methods for advanced wind turbine condition monitoring and early diagnosis: A literature review. *Energies*, *11*(5).

Jardine, A. K., Lin, D., & Banjevic, D. (2006). A review on machinery diagnostics and prognostics implementing condition-based maintenance. *Mechanical Systems and Signal Processing*, *20*(7), 1483 - 1510.

Liu, F. T., Ting, K. M., & Zhou, Z. H. (2008). Isolation forest. In *8th IEEE international conference on data mining.*

Liu, H., Liu, C., & Huang, Y. (2011). Adaptive feature extraction using sparse coding for machinery fault diagnosis. *Mechanical Systems and Signal Processing*, *25*(2), 558-574.

Mallat, S. G., & Zhang, Z. (1993, December). Matching pursuits with time-frequency dictionaries. *IEEE Transactions on Signal Processing*, *41*(12), 3397-3415.

Martin-del-Campo, S., Albertsson, K., Nilsson, J., Eliasson, J., & Sandin, F. (2013). FPGA prototype of machine learning analog-to-feature converter for event-based suc-

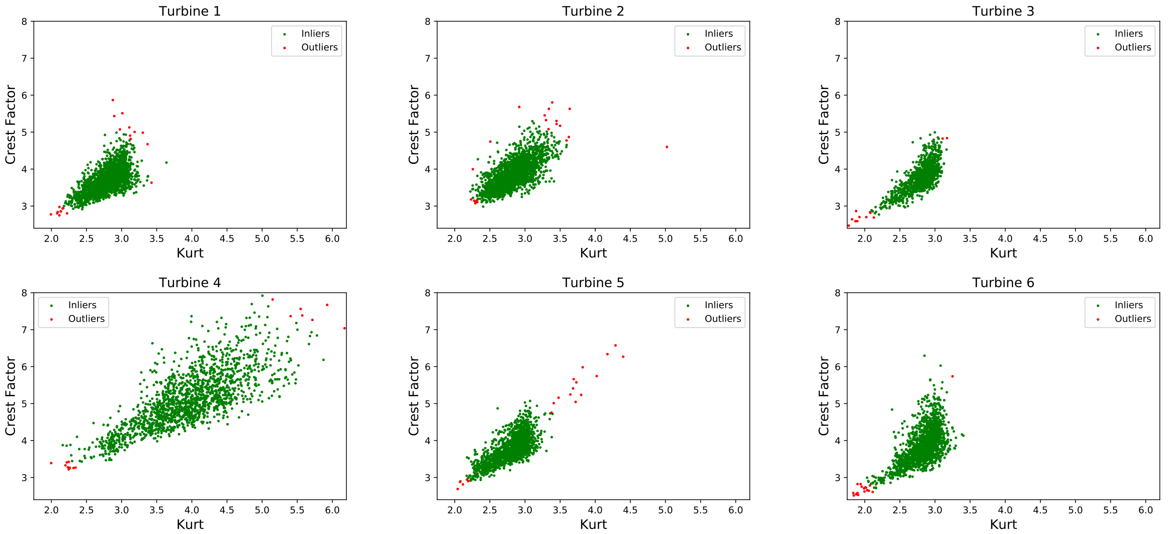


Figure 5. Individual scatter plots of all turbines for the feature-pair: **crest factor and kurtosis**. Signal segments labeled as outliers are marked in red and segments labeled as normal in green. All plots have the same limits.

cinct representation of signals. In *Machine learning for signal processing, ieee international workshop.*

Martin-del-Campo, S., Sandin, F., & Strömbergsson, D. (2018). *Dataset concerning the vibration signals from wind turbines in northern Sweden.* Luleå University of Technology. (http://urn.kb.se/resolve?urn=urn:nbn:se:ltu:diva-70730)

Martin-del-Campo, S., Sandin, F., & Strömbergsson, D. (2019). Dictionary learning approach to monitoring of wind turbine drivetrain bearings. *arXiv preprint arXiv:1902.01426*.

Olshausen, B. A., & Field, D. J. (1997). Sparse coding with an overcomplete basis set: A strategy employed by V1? *Vision Research*, *37*(23), 3311 - 3325.

Patidar, S., & Soni, P. K. (2013, May). An overview on vibration analysis techniques for diagnosis of rolling element bearings faults. *International Journal of Engineering Trends and Technology (IJETT)*, *4*(5), 1804-1809.

Qu, Y., He, D., Yoon, J., hecke, B. V., Bechhoefer, E., & Zhu, J. (2014). Gearbox tooth cut fault diagnostics using acoustic emission and vibration sensors — a comparative study. *Sensors*, *14*(1), 1372–1393.

Randall, R. B. (2011). *Vibration-based condition monitoring*. Wiley.

Skretting, K., & Engan, K. (2011). Learned dictionaries for sparse image representation: properties and results. In *Spie proceedings* (Vol. 8138, p. 81381N-81381N-14).

Smith, E. C., & Lewicki, M. S. (2006, February). Efficient auditory coding. *Nature*, *439*, 978-981.

Zhang, C., Gao, W., Guo, S., Li, Y., & Yang, T. (2017). Opportunistic maintenance for wind turbines considering imperfect, reliability-based maintenance. *Renewable Energy*, *103*, 606 - 612.

BIOGRAPHIES

Sergio Martin-del-Campo received the B.Sc. degree in Mechatronic Engineering from Monterrey Institute of Technology and Higher Education (ITESM), Guadalajara, Mexico in 2007, the M.Sc. degree in Space Science and Technology from Luleå University of Technology (LTU), Sweden and Julius Maximilians University of Würzburg, Germany in 2012, and the Ph.D. degree in Industrial Electronics from LTU in 2017.

Currently, he is a post-doc in EISLAB at LTU, working on the use of machine learning for condition monitoring. His research interests include sparse decomposition of signals and time series analysis.

Kammal Al-Kahwati received his M.Sc degree in Computer Science and Engineering from Luleå University of Technology in 2020. He is currently working as a software developer at Scania CV within CAD/vPDM solutions and customizations. Kammal also has an interest in embedded systems and data analytics.