






Article

Improved Ensemble Learning for Wind Turbine Main Bearing Fault Diagnosis

Mattia Beretta ^{1,2} , Yolanda Vidal ^{3,4} , Jose Sepulveda ² , Olga Porro ²  and Jordi Cusidó ^{2,5,*} 

¹ Unitat Transversal de Gestió de l'Àmbit de Camins (UTGAC), Universitat Politècnica de Catalunya (UPC), 08034 Barcelona, Spain; mattia.beretta@upc.edu

² SMARTIVE S.L., 08204 Sabadell, Spain; jose.sepulveda@smartive.eu (J.S.); olga.porro@smartive.eu (O.P.)

³ Control, Modeling, Identification and Applications (CoDALab), Department of Mathematics, Escola d'Enginyeria de Barcelona Est (EEBE), Campus Diagonal-Besós (CDB), Universitat Politècnica de Catalunya (UPC), Eduard Maristany, 16, 08019 Barcelona, Spain; yolanda.vidal@upc.edu

⁴ Institute of Mathematics (IMTech), Universitat Politècnica de Catalunya (UPC), Pau Gargallo 14, 08028 Barcelona, Spain

⁵ Enginyeria de Projectes i de la Construcció EPC, Universitat Politècnica de Catalunya, 08028 Barcelona, Spain

* Correspondence: jordi.cusido@smartive.eu; Tel.: +34-620-602-495

Abstract: The goal of this paper is to develop, implement, and validate a methodology for wind turbines' main bearing fault prediction based on an ensemble of an artificial neural network (normality model designed at turbine level) and an isolation forest (anomaly detection model designed at wind park level) algorithms trained only on SCADA data. The normal behavior and the anomalous samples of the wind turbines are identified and several interpretable indicators are proposed based on the predictions of these algorithms, to provide the wind park operators with understandable information with enough time to plan operations ahead and avoid unexpected costs. The stated methodology is validated in a real underproduction wind park composed by 18 wind turbines.

Keywords: fault prognosis; WT; main bearing; normality model; real SCADA data



Citation: Beretta, M.; Vidal, Y.; Sepulveda, J.; Porro, O.; Cusidó, J. Improved Ensemble Learning for Wind Turbine Main Bearing Fault Diagnosis. *Appl. Sci.* **2021**, *11*, 7523. <https://doi.org/10.3390/app11167523>

Academic Editor: Mohsen Soltani

Received: 6 July 2021

Accepted: 12 August 2021

Published: 17 August 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Global demand for energy has achieved an unprecedented level with the growing world population and the rising industrialization in developing countries. However, globally, the largest amount of energy is obtained from fossil fuels, which is closely related to the rising levels of greenhouse gases emissions. Definitely, energy transition to renewable sources is the crux of the matter to fight climate change. As stated in [1] only 10% of the world's primary energy supply is already from renewable energies, but they are steadily growing. Among renewable energy sources, wind power has been the fastest growing in the recent decades, and in 2019 it was the leading source of new capacity in Europe, the US, and Canada, and the second largest in China [2].

It is noteworthy that wind energy levelized cost of energy (LCOE) fell 39% between 2010 and 2019 [3]. However, from this LCOE, the operation and maintenance (O&M) accounts for 28.5% in land-based wind projects, and up to 34% in offshore wind projects [4]. Energy production losses due to downtime (caused by O&M of the assets), together with the costs associated to the replacement of components can scale up to millions of Euros per year in any industrial size wind park. Thus, it is of paramount importance that the wind industry moves from corrective (repairing components after they break down) and preventive maintenance (scheduled at regular intervals) to predictive maintenance (scheduled as needed based on the asset condition). Predictive, or condition-based maintenance (CBM), provides operators with an advanced warning before the actual fault occurs, allowing them to plan ahead and schedule repairs to coincide with weather or production windows to reduce costs and turbine downtime. CBM is an extensive area of research in a wide variety

of applications, such as smart manufacturing [5]. However, its application to complex systems, such as wind turbines (WTs) that work under different and varying operating and environmental conditions remains an open challenge. Furthermore, the latest CBM developments tend to use expensive specifically tailored sensors for condition monitoring (CM), which is not economically viable for turbines already under operation and even less in case they are close to reach the end of their lifespan.

The life expectancy of wind parks from the late 1990s and early 2000s surge of wind power is about to be completed. A decision on lifetime extension is complicated, but it is clear that end-of-life solutions will develop a significant market over the next five years [6]. In particular, CBM data-driven methodologies based on already available supervisory control and data acquisition (SCADA) data are a promising cost-effective solution. These SCADA data are highly variable because of the changing operational conditions, they have a low sampling time (10 min average value), are not standardized, and have not been initially designed for the particular purpose of CM but for control purposes. Therefore, it is a hard challenge to contribute CBM strategies based solely on these data [7]. However, this is an active research area, as shown by recent publications. On the one hand, much of the references found in the literature work with simulated SCADA (e.g., [8]) that can be obtained using open source simulation software and, rarely, real data as these are proprietary data from the wind park operators and are not easily available. On the other hand, when dealing with real data, normally, only one or two WTs are tested. For example, [9] where support vector machines are used to predict WT faults, and [10], where different machine learning classifiers are compared to predict generator faults. These references use real WT SCADA data from one and two WTs, respectively, and are based on supervised approaches that require historical fault data to be constructed. This is an important drawback, as obtaining labeled datasets from operational data is typically hard, it is exposed to errors, and leads to a highly unbalanced dataset. Additionally, the methodology can not be applied straightforward to wind parks where the fault of interest did not occur in the past. Thus, despite the promising performance of supervised methods, unsupervised approaches are preferred for SCADA predictive maintenance [11]. The recent works related to unsupervised SCADA based predictive maintenance can be mainly grouped in one of the following three categories: signal trending, normality models, and anomaly detection models. In the following paragraphs, a brief review of each one of these categories is given.

The approaches in the signal trending category study changes and trends in the SCADA time series. Some relevant works include [12], where monitoring of WT generators using the statistical inertia of a wind park average is proposed; and [13], where the difference between the SCADA data of each turbine with the median of the rest of turbines in the wind park is used to establish a condition vector to later apply vector autoregression Hotelling and vector autoregression multivariate exponentially weighted moving average to locate the faulty turbine. The main advantages of signal trending methods are its simplicity and interpretability. However, most of these methods can not capture complex relations among different variables, thus machine and deep learning models are needed to move beyond signal trending solutions.

Regarding data-based normality models, also called normal behavior models, their objective is to learn the relation between a set of input variables and a target variable under normal operation. Then, the difference between predictions and real measurements (for the target variable) is used to detect abnormal behavior. This normality models are well established in unsupervised SCADA based predictive maintenance, but the general trend is to use the power curve (relation between the wind intensity and the extracted power) as the target variable. It is noteworthy the work of [14] where main working parameters (e.g., the rotor speed, and the blade pitch) are used as input variables and the power is employed as the target to construct the normality model. The main advantage of these methods is their capability to learn complex relations between different sensors. On the other hand, its major drawback is a lack of interpretability.

The main bearing is the object of this analysis due to its central role in the drive train assemble. It connects the rotor to the gearbox, and it has to withstand the large torque generated by the rotor, making it prone to problems. Failure rates of 30% are reported for single main bearing and 15% for double main bearing calculated over a 20 year lifetime [15]. Within the typical wear mechanisms affecting the main bearing, micro-pitting, spalling, and smearing can be listed [16]. Main bearing failures can be anticipated through vibration analysis, but also using predictive models based on SCADA data and analysis of temperature signals, as reported in [17].

In this work, a normality model at WT level is selected, based in [18], where the target variable is selected to be the closest sensor to the component under study. In particular, as the main bearing is the component to be monitored, the main shaft temperature is used as target variable. In this work, the component under study is the main bearing because, as stated by the European Academy of Wind Energy (EAWWE) [16], the wind industry has identified main bearing failures as a critical issue in terms of increasing WT reliability and availability, as they lead to major repairs with high replacement costs and long downtime periods.

The methods in the category of anomaly detection models, also called outlier detection, seek to identify rare samples which raise suspicion by differing significantly from the majority of the data. A significant work is [19] where a comparison of three anomaly detection techniques (one-class support vector machine, isolation forest, and elliptical envelope) for WT CBM using SCADA data is realized. Isolation forest has some advantages over other approaches as it does not require a normal operation dataset, and it requires less training data than other deep learning strategies. In this work, an anomaly detection model at wind park level is proposed based on the isolation forest methodology.

Considering all the aforementioned references, in this work, a CBM strategy based on SCADA data is stated with a five-fold contribution: (i) It is an unsupervised approach, thus there is no need that the specific studied fault happened in the past to train the proposed models; (ii) It is an ensemble that combines the benefits of a WT normal behavior model with a wind park anomaly detection model; (iii) It combines different training and prediction window sizes; (iv) Fault prediction is accomplished months in advance prior to the fault, giving enough time to operators to plan ahead and schedule repairs; and (v) The validity and performance of the proposed methodology is demonstrated (tested) on a dataset covering two years and a half of operation from a real underproduction wind park composed by 18 WTs.

The rest of the article is organized as follows. Section 2 presents the available SCADA data and work order logs used in this work. Next, Section 3 states the proposed ensemble methodology for WT main bearing fault diagnosis, including a comprehensive description of the single models and their indicators. Section 4 presents the results and their discussion to interpret and describe the significance of the ensemble method in comparison to the single models by themselves. Finally, in Section 5, conclusions are drawn, and future work is proposed.

2. Data

Two information sources are used in this research: (i) SCADA operating data and (ii) work order logs. The first one is used for modeling the behavior of the turbine, whereas the second one is used uniquely to validate the results of the algorithms. Both SCADA and work order logs are typically available to wind park owners, without the need of installing new sensors nor change operating routines. Albeit, the format and content of the work order logs, being less structured and not standardized, may vary significantly from one wind park maintainer to another.

2.1. SCADA Operating Data

SCADA is made off a vast net of sensors, monitoring the state of the main components of a turbine, as well as the environmental conditions. Operating data are recorded at high frequency and successively down-sampled to reduce network usage and storage costs.

The resolution of SCADA data is usually of 10 min. To reduce information loss, due to down-sampling, not only the mean value across the aggregation period is stored, but also the standard deviation, minimum, and maximum values. An example of a typical SCADA dataset is provided in Table 1.

Table 1. Sample of SCADA data for a limited selection of turbines and signals. Each signal is stored completed with its minimum (*min*), maximum (*max*), mean (*mean*), and standard deviation (*std*) values. The frequency of the data is 10 min.

WT	Timestamp	Wind Speed [m/s]				Main Bearing Temperature [°C]			
		<i>min</i>	<i>max</i>	<i>std</i>	<i>mean</i>	<i>min</i>	<i>max</i>	<i>std</i>	<i>mean</i>
WT01	2018-01-01 00:00:00	3.8	11.8	1.369	8.346	33.0	33.0	0.000	33.000
WT02	2018-01-01 00:00:00	3.8	11.5	1.234	8.065	32.0	33.0	0.221	32.051
WT03	2018-01-01 00:00:00	5.6	11.5	0.976	8.283	29.0	30.0	0.500	29.505
WT01	2018-01-01 00:10:00	3.7	11.1	1.323	7.794	33.0	33.0	0.000	33.000
WT02	2018-01-01 00:10:00	3.5	10.3	1.152	7.178	32.0	33.0	0.386	32.182
WT03	2018-01-01 00:10:00	4.6	10.9	0.984	7.858	29.0	30.0	0.499	29.532

Operating data from a European onshore wind park, situated in Poland, is used in this research. A total of 18 turbines with nominal power of 2.3 MW is available. Approximately three and a half years of data are analyzed, starting from the beginning of 2017 to mid-2020. Data from 2017 are used to train the normality model, and the remaining two and a half years are used to test predictions. Having more than one year of data allows to control seasonal variations in environment temperature and wind-speed.

2.2. Work Orders

Wind park owners commonly keep track of the ordinary and extraordinary maintenance interventions required by the turbines. This information is typically stored in text files, where a description of the intervention is reported with a timestamp of the date in which it occurred. Sometimes detailed information, such as the material required for the intervention and other details, is provided. In this research, work order logs are used to assess the validity of the main-bearing status predictions, and it is not fed as input to the algorithms. An example of the work order logs is presented in Table 2.

Table 2. Sample of work order data.

WT	Timestamp	Component	Comments
WT11	2017-04-28 11:08:00	Gearbox Bearing	Gearbox Bearing repair
WT06	2018-06-11 08:40:00	Main Bearing	Replacing Main Bearing and Main Shaft
WT07	2019-07-17 07:40:00	Main Bearing	Inspection required, condition-based.
WT03	2020-02-25 09:10:00	Main Bearing	Main Bearing inspection, due to rate of worn

3. Methodology

3.1. Normal Behavior Model

In this subsection, the selected normal behavior model based on an artificial neural network (ANN) is comprehensively described. Note that this model is designed at a turbine level, thus each WT in the park will have its associated normality model trained with only its own historical SCADA data. First, the data preprocess is detailed to deal with out-of-range values, missing data, and sensors with different magnitudes. Second, a one-year time window is selected to define the training dataset including all operating conditions of the WT. This will ensure that the detected anomalies are not just a change in seasonality and will allow the methodology to be used across all regions of operation of the WT. Third, the ANN set-up is detailed, including a brief explanation of the optimization

and regularization methods, as well as the selection of the ANN structure. Finally, a specific purposely build indicator for the normal behavior model is stated.

3.1.1. Data Preprocess

First, variable selection to determine a set of variables that will provide the best fit for the model so that accurate predictions can be made is needed. In this work, the temperatures of the components located close to the main bearing are selected as variables of the normality model together with the ambient temperature, as it affects the temperatures of all subsystems. Additionally, the generated power and rotor speed provide information about the region of operation of the WT and, thus, they are also used as variables. In summary, the selected variables are shown in Table 3 where also the specific ranges of realistic values for each sensor are listed.

Table 3. Selected SCADA variables to develop the normal behavior model, its description, range of possible values, and units. All of them are related to the mean value over a 10-min period.

Variable	Description	Range	Units
Power	Generated real power	[0, 2000]	kW
AmbientTemp	Ambient temperature	[−5, 40]	°C
BearingCSTemp	Bearing coupling side temperature	[0, 120]	°C
BearingNCSTemp	Bearing non-coupling side temperature	[0, 120]	°C
LowSpeedShaftTemp	Low-speed shaft temperature	[0, 120]	°C
GeneratorTemp	Generator temperature	[0, 175]	°C
GearboxTemp	Gearbox temperature	[0, 120]	°C
RotorSpeed	Rotor speed	[0, 50]	rpm

Second, for each variable out-of-range values are deleted as they are associated with sensor measurement errors. Furthermore, data imputation of missing values (and deleted values of the prior step) is carried out through piecewise cubic Hermite interpolating polynomials. As it has a local smoothing property, this strategy produces more stable estimates compared to other standard approaches used for data imputation [20]. Notice that the nearest value before or after the missing values is used at the beginning and end of the dataset, respectively.

Finally, as data from various variables have varying magnitudes, the max–min normalization is used to scale the dataset.

3.1.2. Train and Test Sets

The aim of the normality model is that it is capable to cope with the various operational and environmental conditions that the WT will face, see [18]. Thus, the train and test datasets include data from all working conditions: different wind speed regions and their associated regions of operation of the WT (start up, maximize power, and limit wind power to avoid exceeding the safe electrical and mechanical loads), different year seasons, curtailment, etc. Therefore, it is noteworthy that in this work there is no filtering of the data based on specific regions of operation or seasonality. The available SCADA data, for the normality model, are divided as follows: data from 2017 are used to train the model (thus, the training dataset contains a whole year of data), and the remaining two and a half years are used to test predictions. Note that there is no validation set because Bayesian regularization is used to train the ANN.

Finally, recall that a customized normality model will be built for each wind turbine in the park. This could raise some concerns related to its computational cost. However, note that after the model is trained on one year data, this model is used for two-year predictions ahead, as will be shown in the results section. Thus, the computational cost is low in comparison to the use of a training rolling window of observations preceding the target forecast that must be retrained at each window shift of the data.

3.1.3. ANN Set Up

The ANN model structure is proposed in this section and is based on the eight selected variables, shown in Table 3. The output of the ANN is the temperature of the low-speed shaft (variable of interest) at time t , and the inputs are the following ones:

- y_1 : generated real power at time $t - 1$,
- y_2 : generated real power at time t ,
- y_3 : ambient temperature at time $t - 1$,
- y_4 : ambient temperature at time t ,
- y_5 : bearing coupling side temperature at time $t - 1$,
- y_6 : bearing coupling side temperature at time t ,
- y_7 : bearing non-coupling side temperature at time $t - 1$,
- y_8 : bearing non-coupling side temperature at time t ,
- y_9 : generator temperature at time $t - 1$,
- y_{10} : generator temperature at time t ,
- y_{11} : gearbox temperature at time $t - 1$,
- y_{12} : gearbox temperature at time t ,
- y_{13} : rotor speed at time $t - 1$,
- y_{14} : rotor speed at time t .

Thus, referring to the structure of the ANN, there are 14 inputs noted as $y_i, i = 1, \dots, 14$, there is 1 output noted as \hat{y} , and a hidden layer comprising 72 neurons. Figure 1 shows the ANN architecture.

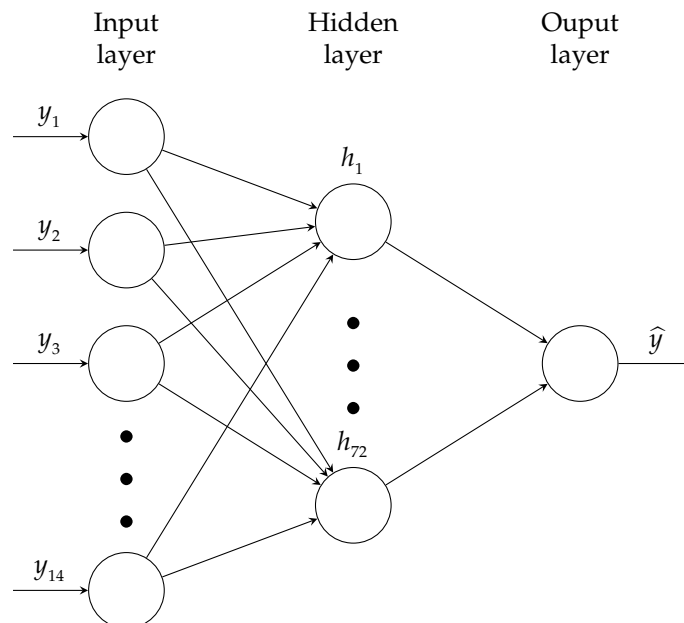


Figure 1. ANN proposed architecture. There are 14 inputs. The hidden layer is set to 72 nodes. The output layer is the estimated temperature of the low-speed shaft at time t .

The ANN is trained with the Levenberg–Marquardt optimization method combined with Bayesian regularization (to enhance the generalization capability of the model) as they are able to obtain lower mean squared errors than any other algorithms for functional approximation problems, see [21,22]. Recall that the main purpose of the WT normality model is to approximate precisely a function, namely the temperature of the low-speed shaft of that specific WT, under normal (healthy) condition. The Bayesian selection of the regularization parameters provides an optimal regularized solution, as well as insight into the effective number of parameters actually used by the ANN which is extremely useful to design the size of the network. In this work a value of 1058 effective number of parameters is obtained from a total of 1153 parameters in the proposed network (number of weights

and biases), thus the complexity of the ANN is appropriate for the used training dataset. A comprehensive description of the Levenberg–Marquardt with Bayesian regularization method can be found, for example, in [18,21,23].

Finally, note that the network uses rectified linear unit (ReLU) activation functions, and weights and biases initialization is performed using the Xavier initializer [24].

3.1.4. Normal Behavior Model Indicator

A fault indicator is needed that activates an alarm when samples not following the normality model are detected. Initially, such an indicator could be based on establishing a threshold in the residual between the estimated value (given by the ANN) and the real SCADA data. However, this will result in an unacceptably high number of false alarms because of solitary samples trespassing the threshold, rendering the method worthless. Therefore, it is critical to establish an indicator that takes into account the persistence of consecutive samples above a certain threshold.

First, a detection threshold is prescribed as $\text{threshold} = \mu + 6\sigma$, based on the mean, μ , and standard deviation, σ , of the residuals over the training data.

Second, a weekly indicator is implemented as follows:

$$\text{indicator}_{\text{normality}} = \min\left(1, \frac{n_{\text{over}}}{504}\right). \quad (1)$$

where n_{over} denotes the number of samples that had a residual value greater than the threshold that week. Note that this indicator has a range between 0 and 1. When all samples are below the threshold, the indicator value is 0. When half or more of the samples in a week are above the threshold, the indicator value is 1.

3.2. Isolation Forest

A second method is used to analyze the status of the main bearing, namely an anomaly detection algorithm: isolation forest. The objective is to complement the prediction of the neural network normality model, using an algorithm that is able to analyze the wind park as a whole. First, data pre-processing is discussed, highlighting the differences to the processing required by the neural network. Second, the testing scheme for the isolation forest is presented; the main difference with the previous algorithm is the lack of a training period. Third, a brief explanation of isolation forest is provided together with a discussion on the selection of the main parameters for the algorithm. Finally, the post-processing of the anomaly detection results is detailed, showing how to construct an indicator that captures differences within turbines.

3.2.1. Data Preprocess

In analogy to the neural-network pre-processing, a reduced selection of variables is used. Main bearing temperature is chosen, as monitoring the status of this component is the objective of the research. Additionally, ambient temperature and rotating speed of the main shaft are selected as they determine information on the context and operating status of the main bearing. A summary table of the inputs is proposed in Table 4, completed with the ranges used to filter outliers and sensor measurement errors. No data imputation is performed for missing values, instead data are down-sampled from ten minutes to hourly resolution, reducing the amount of empty timestamps and cutting computation time. Being the main bearing a massive component the reduction in data frequency is mitigated by the large thermal inertia of the bearing, thus limiting information loss.

Table 4. Selected SCADA variables used to develop the anomaly detection model, its description, range of possible values, and units. All of them are related to the mean value over a 10-min period.

Variable	Description	Range	Units
AmbientTemp	Ambient temperature	[−5, 40]	°C
MainBearingTemp	Main bearing temperature	[0, 120]	°C
RotorSpeed	Rotor speed	[0, 50]	rpm

3.2.2. Testing Scheme

The way the isolation forest algorithm is used in this research, requires no selection of a set of normal conditions to adjust the parameters of the model. Instead, data are analyzed through a rolling window of 1 month width and instances lying at the border of the data distribution are marked as anomalies. The rolling window is shifted each time by one week, effectively sweeping the entire dataset. The choice of the window length is the fruit of a compromise between computation time and the ability to capture a complete set of operating conditions of the turbines.

3.2.3. Isolation Forest Setup

Isolation forest has been introduced by Liu et al. [25] as an efficient anomaly detection algorithm based on widely known decision trees. Unlike most model-based algorithms, no normal condition training set is required. Points lying at the edges of the data distribution are to be intended as anomalies. Having this definition of anomalous point, isolation forest uses binary search trees to recursively split data. Various fully developed trees, i.e., having single instance terminal nodes, are trained. Isolation of anomalies is promoted by using random partitions when separating data, as indicated in [25]. The anomaly score used to determine the likelihood of a point to be anomalous is based on the average path length required by the trained trees to isolate the cited point from the rest of the data. Equation (2) is the canonical definition for an isolation forest anomaly score.

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

where $E(h(x))$ is the average of the path length $h(x)$ required to isolate the given point. The denominator $c(n)$ is the average path length of unsuccessful search in binary search tree. Given a distribution d , three cases are typical:

- $s \rightarrow 1 \implies x$ is anomalous
- $s < 0.5 \implies x$ is normal
- $s \approx 0.5 \forall x$ in $d \implies$ no anomalies in d

In this research, the implementation of Isolation Forest from Sklearn, for Python programming language is utilized [26]. The following parameters are set to run the algorithm:

- number of estimators = 250
- contamination = 0.1
- max samples = 0.3

The number of estimators determines the quantity of decision trees of which the isolation forest is composed. More trees lead to more reliable estimations of the path length, at the cost of higher computational time. The contamination coefficient represents the expected amount of anomalies in the data. The points in the analyzed distribution are sorted by their anomaly score and the most anomalous ones, according to the contamination value, are selected. Finally, the maximum number of samples to use to train each estimator, i.e., each decision tree, is defined by the homonymous parameter.

3.2.4. Anomaly Detection Indicator

An anomaly indicator is defined to translate the output of the isolation forest into concrete information on the WT status. The discovered anomalies are assigned to the corresponding turbines, to define which turbines have a larger percentage of anomalous points with respect to the rest of the wind park. The indicator is calculated as the ratio between the anomalous points ($n_{anomalies}$) and the total amount of records (n_{total}) of a turbine for a given period of time, see Equation (3). The idea is that if no clear differences between turbines' behaviors are present, the anomaly indicator will be similar across the wind farm. On the other hand, if a turbine is subjected to different operating conditions, its percentage of anomalies over the total number of records will be higher if compared with other turbines.

$$\text{indicator}_{\text{anomaly}} = \frac{n_{\text{anomalies}}}{n_{\text{total}}}. \quad (3)$$

3.3. Ensemble

Ensemble strategies have been widely used in the literature both in power output prediction, as well as fault diagnosis [27–29].

The output of the previous algorithms is merged into a composed indicator, with the objective to produce better predictions by overcoming the limitations of the two methods. To build a valuable ensemble, it is important to choose base indicators that are complementary and not redundant, thus a key step is to verify that the correlation of the indicators is limited. The correlation of the indicators is addressed in Section 4.3. The normality indicator provides an individual analysis of the turbines, by comparing a historical model with current conditions, whereas the anomaly indicator is based on a comparison within the different turbines of the wind farm.

Ensemble Indicator

The ensemble strategy that is used in this research does not require training of an additional meta-learner, instead the normality indicator (1) and anomaly indicator (3) are combined through a rolling windowed sum. The pseudocode to compute the ensemble indicator is given in Algorithm 1. In particular, the first step is to rank the turbines, assigning the corresponding percentile in which they fall when compared to the indicator distribution of the whole wind farm. Possible ties between turbines are managed using a ranking scheme that assigns to the tying turbines the lowest rank of the group. Being the single model indicators calculated on a weekly basis, each turbine will be assigned a percentile rank per each week. The next step consists in the application of a rolling window sum. More specifically, a window of $p = 4$ weeks size is chosen. Applying a rolling window sum allows to highlight long-lasting changes, limiting the influence of isolated peaks. Finally, a decision threshold is applied to decide whether an alarm must be or not triggered.

Algorithm 1: Computation of the ensemble indicator

Result: i_e : indicator_{ensemble}.
 Given i_a, i_n : indicator_{anomaly}, indicator_{normality} respectively;
 Set DT: decision threshold (e.g. 0.85);
 Set p : rolling window size (e.g. 4);
STEP 1: Rank turbines w.r.t. wind farm;
for w in $[w, w+1, \dots, w+p-1]$ **do**
 | **for** wt in turbines **do**
 | | $x_{[anomaly]wt,w} =$ percentile of i_a for turbine wt w.r.t. distribution of the entire windfarm during week w ;
 | | $x_{[normality]wt,w} =$ percentile of i_n for turbine wt w.r.t. distribution of the entire windfarm during week w ;
 | **end**
end
STEP 2: Calculate the ensemble ranking during the period of observation for each turbine;
for wt in turbines **do**
 | $x_{[ensemble]wt} = \sum_{w=1}^{w+p-1} x_{[anomaly]wt,w} + x_{[normality]wt,w}$;
end
 $x_{[max-ensemble]} = 2p$;
STEP 3: Calculate the ensemble indicator of a turbine;
 $i_{e,wt} = \frac{x_{[ensemble]wt}}{x_{[max-ensemble]}}$;
STEP 4: Apply decision threshold;
if $i_{e,wt} > DT$ **then**
 | Turbine requires maintenance;
else
 | No maintenance required;
end

4. Results

4.1. Test Data

The proposed methodology is validated with real SCADA data from an underproduction wind park composed by 18 wind turbines. Recall that approximately three and a half years of data are analyzed, starting from the beginning of 2017 to mid-2020. Data from 2017 are used to train the normality model, and the remaining two and a half years are used to test predictions using the work order logs. Table 5 shows the main bearing failures during the test period based on the work order logs. Three WTs in the park suffer from the main bearing failure during the test period. Note that the work order log comments are different for each case, ranging from lubrication to component replacement.

Table 5. Main bearing failures reported in the work order logs.

WT	Timestamp	Component	Comments
WT03	2020 February 25 9:10:00	Main bearing	Main Bearing inspection, due to rate of worn
WT05	2018 June 11 8:40:00	Main bearing	Replacing Main Bearing and Main Shaft
WT06	2019 July 17 7:40:00	Main bearing	Inspection required, condition-based

4.2. Performance Metrics

To evaluate the performance of the proposed methodology, the results will be analyzed via a confusion matrix based on the criteria that is described next. When the alarm is triggered within a six-month window before the failure actually occurs, the result is reported as a true positive (TP). The method in this case provides sufficient advance notice of the failure, so that the wind park operators can organize the maintenance operation while minimizing costs. When the alarm is not triggered and the WT does not have any work order, the result is documented as a true negative (TN). In this case, the method is correctly establishing that the WT is under normal operation. When the alarm is not

triggered, or it is done out of the six-month before failure window, and the WT suffers a main bearing failure, the result is reported as a false negative (FN). Finally, when the alarm is triggered but the WT does not have any work order, this result is informed as a false positive (FP).

Finally, the following performance metrics are used to analyze the results: accuracy, precision, recall, specificity, and F1 score. Their definitions are briefly recalled hereby,

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}},$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}},$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}},$$

$$\text{Specificity} = \frac{\text{TN}}{\text{TN} + \text{FP}},$$

$$\text{F1} = \frac{\text{TP}}{\text{TP} + \frac{1}{2}(\text{FN} + \text{FP})}.$$

4.3. Ensemble vs. Single Models

The different models that compound the ensemble should provide complementary information, see [17]. Thus, it is important to analyze the correlation between the indicators provided by each of the individual models: the anomaly detection model vs. the normality model. Figure 2 displays the isolation forest indicator vs. the neural network indicator for each WT in the park separately. The Pearson correlation coefficient and p -value for testing non-correlation are reported as r and p , respectively. It is shown that, in general, there is no correlation between the individual indicators (low values of the Pearson correlation coefficient are obtained), thus they are complementary in terms of their contained information. Note the particular case of WT05 where the Pearson correlation is about 0.78, which indicates that there is a moderate positive relationship between the variables, as the p -value in this case is almost zero (which indicates that the correlation coefficient is significant). The explanation of this special behavior is that WT05 suffers the only main bearing fault in the park that is correctly predicted by both indicators, thus in this case the indicators are correlated.

Figures 3–5 show the normal behavior model indicator, the anomaly detection indicator, and the ensemble indicator, respectively. In these figures, the blue stars indicate the occurrence of a main bearing fault (as recorded in the work order logs). Note that the normal behavior indicator shows high values in WT01 and WT18, which are healthy over the whole test set, but the anomaly detection indicator shows low values for these WTs. On the contrary, the anomaly indicator shows high values for WT16, which is healthy, but the normality indicator shows low values for this WT. The ensemble indicator adequately combines the information from the single indicators showing low indicator values for WT01, WT16, and WT18 as desired.

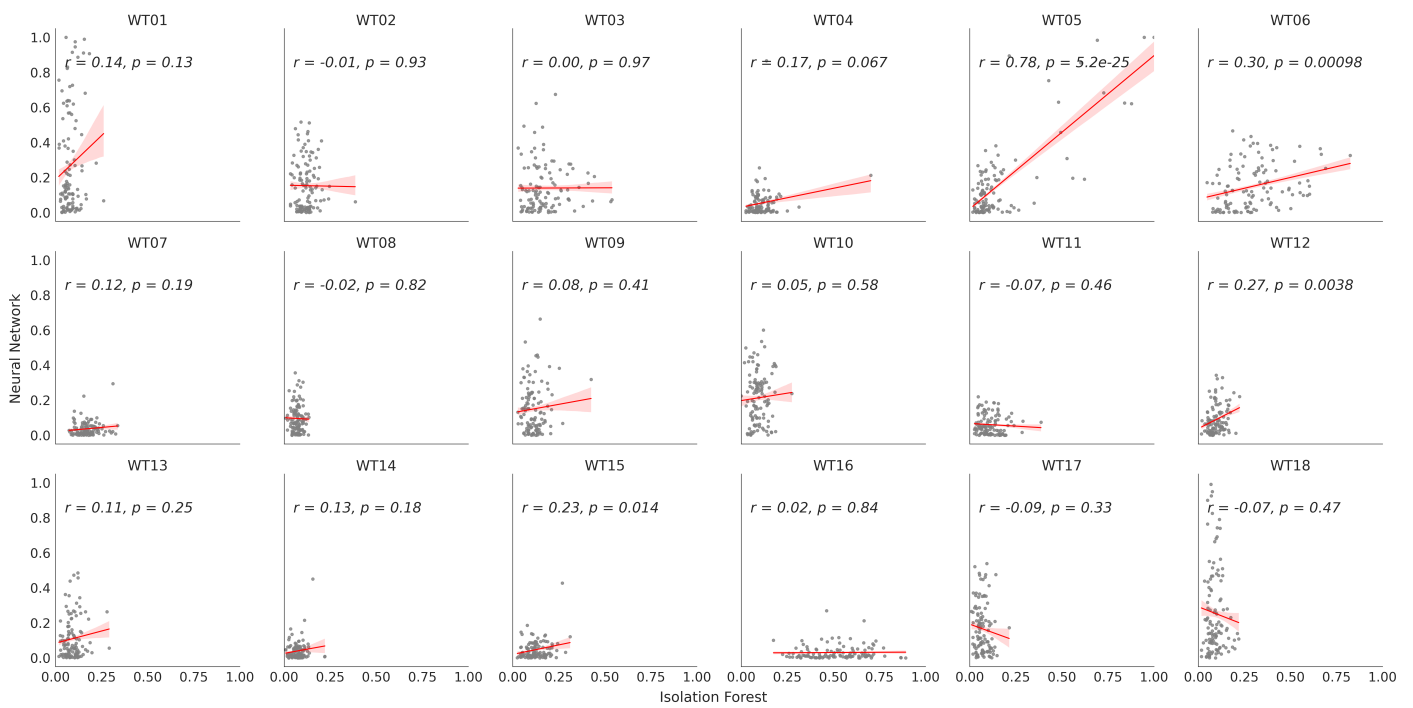


Figure 2. Scatter plot showing the isolation forest indicator vs. the neural network indicator independently for each WT. The linear regression model fit, that minimizes the squared error of the data, is plot in red. The shaded area represents the 68% confidence interval of the model. Pearson correlation coefficient and p -value for testing non-correlation are reported as r and p , respectively.

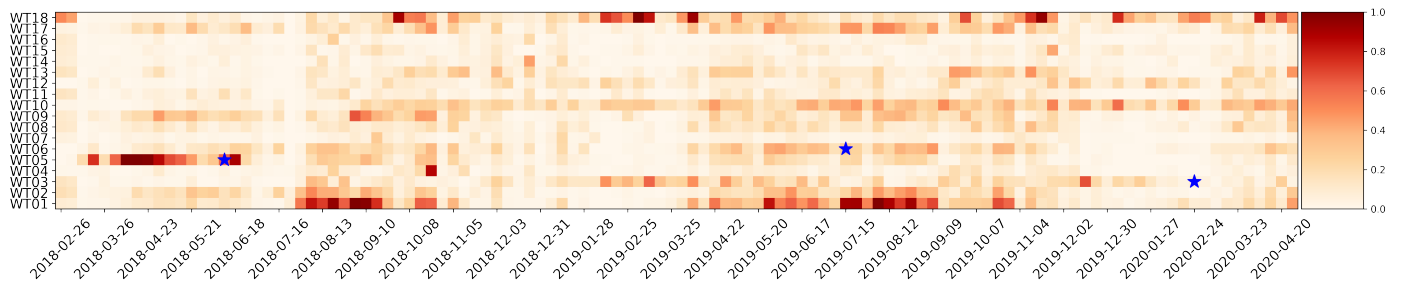


Figure 3. Main bearing health according to normal behavior model. Faults are marked with a blue star.

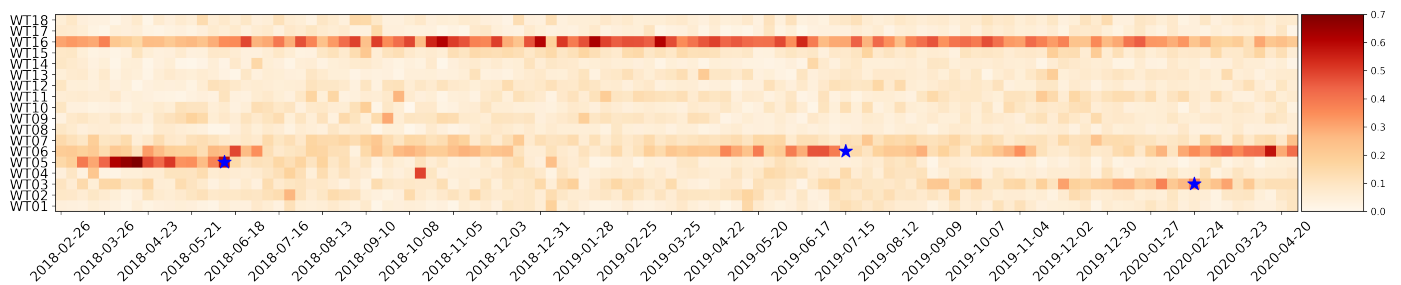


Figure 4. Main bearing health according to anomaly detection. Faults are marked with a blue star.

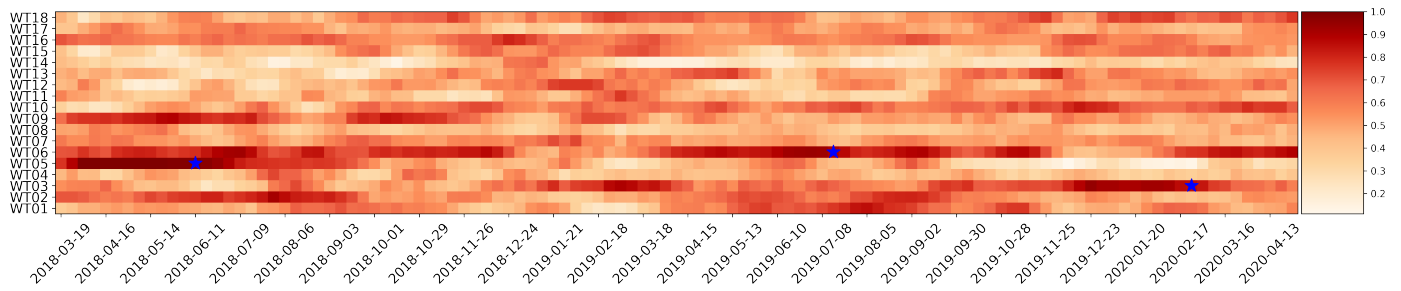


Figure 5. Main bearing health according to the proposed ensemble method. Faults are marked with a blue star.

To make a decision to trigger or not an alarm, a decision threshold is needed. Values of the indicator above the threshold will activate an alarm, and values below the threshold will be considered normal. To better visualize the results, a saturation mask is proposed in the following manner: all indicator values below the threshold are saturated to 0, and all indicator values above or equal to the threshold are saturated to 1. Figure 6 shows the ensemble indicator after applying the saturation mask with a decision threshold value of 0.85. It can be observed that the three main bearing faults in the park (highlighted with the blue stars) are warned at least 5 months in advance. On the other hand, there are false alarms at WT01, WT02, and WT09.

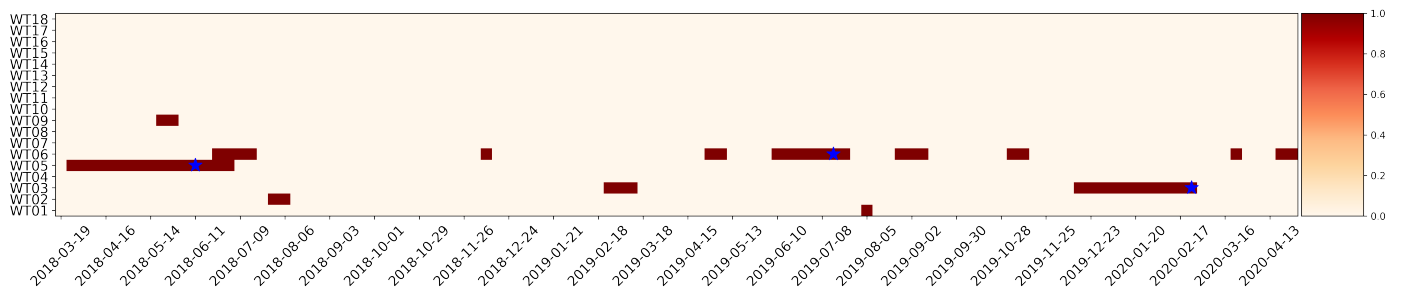


Figure 6. Saturation mask, with a decision threshold value of 0.85, used to determine whether a turbine should be reviewed or not. Faults are marked with a blue star.

The value of the decision threshold (DT) has a great impact on the final results. Figure 7 shows the DT value with respect to the different metrics. It is observed that, in general, higher values of the DT lead to better performance metrics, and it is noteworthy that for the same value of the DT the best performance metrics are clearly obtained with the ensemble method (except for low values of the threshold that have no practical relevance as they would lead to an excessive number of false alarms).

To further analyze the impact of the DT on the final results, Table 6 shows the confusion matrix information with respect to the DT value for the ensemble method. It is clear that low values of the threshold have no practical application since they lead to diagnose always as faulty the WT, since all results are positive (either TP or FP). Increasing the value of the DT diminishes the number of FP (false alarms), and consequently increases the number of TN (correctly classified healthy WTs). When the DT is equal to 0.95 only 28 instances are wrongly classified as FN (that is, faulty instances wrongly classified as healthy). When the DT is equal to 0.90 there are 58 instances wrongly classified as FP (false alarm), however, all faulty instances are correctly detected. In this case, when DT = 0.9, all faults are detected at the expense of having some false alarms. The wind park operator should compare the cost of unnecessary checks (due to false alarms) with respect to the savings of early warning of the main bearing fault to decide the best DT to be used.

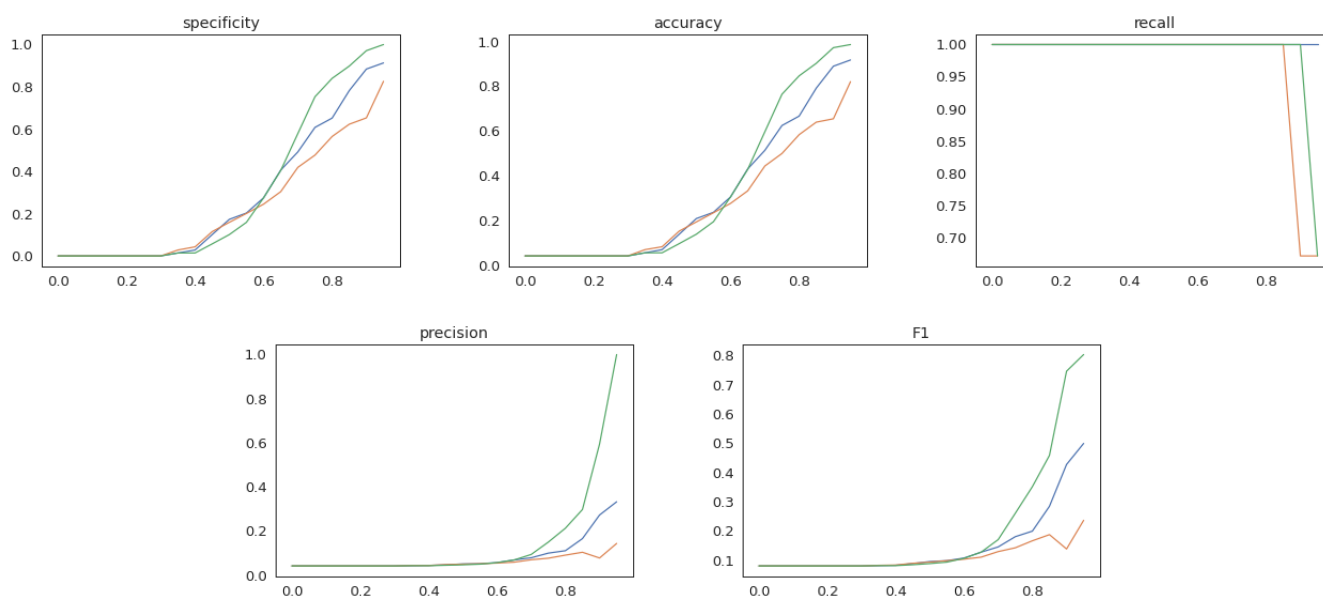


Figure 7. Decision threshold (DT) value vs. the different model indicators for the specificity (top left), accuracy (top middle), recall (top right), precision (bottom left), and F1 score (bottom right) metrics. The blue line corresponds to the anomaly indicator, the orange line to the normality indicator, and the green line to the ensemble indicator.

Table 6. Confusion matrix of the ensemble method results with respect to the decision threshold (DT).

DT	TP	FP	FN	TN
0.00	85	1967	0	0
0.05	85	1967	0	0
0.10	85	1967	0	0
0.15	85	1967	0	0
0.20	85	1967	0	0
0.25	85	1967	0	0
0.30	85	1967	0	0
0.35	85	1939	0	28
0.40	85	1939	0	28
0.45	85	1854	0	113
0.50	85	1769	0	198
0.55	85	1654	0	313
0.60	85	1427	0	540
0.65	85	1173	0	794
0.70	85	829	0	1138
0.75	85	485	0	1482
0.80	85	315	0	1652
0.85	85	201	0	1766
0.90	85	58	0	1909
0.95	57	0	28	1967

Finally, Figure 8 and Table 7 compare the performance metrics among the different single models and the ensemble model with respect to different DT values. In particular, Figure 8 shows a bar plot of the used metrics for three different specific values of DT, namely 0.6, 0.85, and 0.95. It is clear that the ensemble strategy outperforms in all cases the single methods, being outstanding the result in precision for DT=0.95 where a value of 1.0 is achieved with the ensemble but values lower than 0.5 are obtained with the normality and anomaly models by themselves. Table 7 gives a further detailed comparison of the performance metrics for values of the DT from 0 to 0.95 with increments of 0.05. Note that for values of the DT equal or bigger than 0.7 the best performance metrics are obtained with the ensemble method, clearly showing the advantage of this method with respect to the single models.

Table 7. Metrics comparison among the single and ensemble methods with respect to the decision threshold (DT).

DT	Recall			Specificity			Accuracy			Precision			F1		
	Isolation Forest	Neural Network	Ensemble	Isolation Forest	Neural Network	Ensemble	Isolation Forest	Neural Network	Ensemble	Isolation Forest	Neural Network	Ensemble	Isolation Forest	Neural Network	Ensemble
0.00	1.0	1.000	1.000	0.000	0.000	0.000	0.041	0.041	0.041	0.041	0.041	0.041	0.080	0.080	0.080
0.05	1.0	1.000	1.000	0.000	0.000	0.000	0.041	0.041	0.041	0.041	0.041	0.041	0.080	0.080	0.080
0.10	1.0	1.000	1.000	0.000	0.000	0.000	0.041	0.041	0.041	0.041	0.041	0.041	0.080	0.080	0.080
0.15	1.0	1.000	1.000	0.000	0.000	0.000	0.041	0.041	0.041	0.041	0.041	0.041	0.080	0.080	0.080
0.20	1.0	1.000	1.000	0.000	0.000	0.000	0.041	0.041	0.041	0.041	0.041	0.041	0.080	0.080	0.080
0.25	1.0	1.000	1.000	0.000	0.000	0.000	0.041	0.041	0.041	0.041	0.041	0.041	0.080	0.080	0.080
0.30	1.0	1.000	1.000	0.000	0.000	0.000	0.041	0.041	0.041	0.041	0.041	0.041	0.080	0.080	0.080
0.35	1.0	1.000	1.000	0.014	0.028	0.014	0.055	0.069	0.055	0.042	0.043	0.042	0.081	0.082	0.081
0.40	1.0	1.000	1.000	0.028	0.043	0.014	0.069	0.082	0.055	0.043	0.043	0.042	0.082	0.083	0.081
0.45	1.0	1.000	1.000	0.101	0.115	0.057	0.138	0.152	0.096	0.046	0.047	0.044	0.088	0.089	0.084
0.50	1.0	1.000	1.000	0.173	0.159	0.101	0.208	0.193	0.138	0.050	0.049	0.046	0.095	0.093	0.088
0.55	1.0	1.000	1.000	0.203	0.201	0.159	0.236	0.234	0.194	0.051	0.051	0.049	0.098	0.098	0.093
0.60	1.0	1.000	1.000	0.275	0.245	0.275	0.305	0.276	0.305	0.056	0.054	0.056	0.107	0.103	0.106
0.65	1.0	1.000	1.000	0.406	0.304	0.404	0.430	0.332	0.428	0.068	0.058	0.068	0.127	0.110	0.127
0.70	1.0	1.000	1.000	0.492	0.419	0.579	0.513	0.443	0.596	0.078	0.069	0.093	0.145	0.129	0.170
0.75	1.0	1.000	1.000	0.608	0.477	0.753	0.624	0.499	0.764	0.099	0.076	0.149	0.180	0.142	0.260
0.80	1.0	1.000	1.000	0.652	0.565	0.840	0.666	0.583	0.846	0.110	0.090	0.212	0.199	0.166	0.351
0.85	1.0	1.000	1.000	0.782	0.623	0.898	0.791	0.639	0.902	0.165	0.103	0.297	0.284	0.187	0.458
0.90	1.0	0.671	1.000	0.884	0.653	0.971	0.889	0.654	0.972	0.272	0.077	0.594	0.427	0.138	0.746
0.95	1.0	0.671	0.671	0.913	0.826	1.000	0.917	0.820	0.986	0.332	0.143	1.000	0.499	0.236	0.803

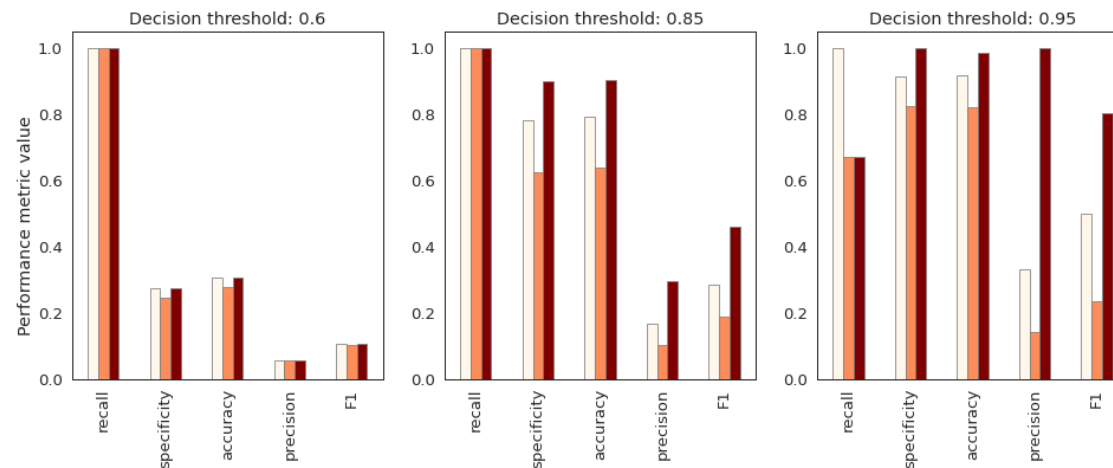


Figure 8. Performance metrics comparison for three different decision threshold values. The white bar represents the anomaly detection indicator, the orange one is for the normality indicator, and the red one for the ensemble indicator.

5. Conclusions

In this work an ensemble method for main bearing fault diagnosis has been proposed, implemented, and validated on a real under-production wind park composed by 18 WTs. The ensemble combines a normality model at WT level (i.e., a specific model is trained for each WT in the park) with an anomaly detection model at wind park level by using only SCADA data and past work order logs.

The stated ensemble method only requires healthy data to be trained (since it is not a supervised approach), and thus the methodology can be applied to any WT, even when there are no past records about the fault under study. The obtained results show that for the same value of the DT the best performance metrics are clearly obtained with the ensemble method in comparison to the single methods. Furthermore, the proposed strategy is able to warn at least 5 months in advance (giving enough time to the wind park operator to organize logistics and minimize downtime) the three main bearing faults present in the wind park during the test set, with few false alarms.

The DT has a great influence on the final results, thus it is strongly advised that the wind park operator analyzes the cost of unnecessary checks (due to false alarms) with respect to the savings of early warning of the main bearing fault to decide the best DT to be employed.

Two future work directions are envisioned. First, the extended isolation forest (EIF) presented in [30] will be incorporated to investigate the possible improvement in the results. Second, gearbox faults will be studied, as gearboxes tend to fail prematurely in WTs, and their replacement is very costly (the outage can last between a few days to months, depending on crane and parts availability). The wind park SCADA data and work order logs used in this research also contain gearbox faults, thus future work will address this type of fault, taking as starting point the ensemble method proposed in this work.

Author Contributions: All authors contributed equally to this work. All authors have read and agreed to the published version of the manuscript.

Funding: This work has been partially funded by the Spanish Agencia Estatal de Investigación (AEI) - Ministerio de Economía, Industria y Competitividad (MINECO), and the Fondo Europeo de Desarrollo Regional (FEDER) through the research project DPI2017-82930-C2-1-R; by the Generalitat de Catalunya through the research project 2017 SGR 388; by Centro para el Desarrollo Tecnológico Industrial, grant number CDTI-IDI 20191294; and by Agència de Gestió d'Ajuts Universitaris i de Recerca (AGAUR), grant number DOCTORADO AGAUR-2017-DI 004.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data required to reproduce these findings cannot be shared at this time as it is proprietary.

Acknowledgments: The authors express their gratitude and appreciation to the Smartive company (<http://smartive.eu/>, accessed on 5 July 2021), as this work would not have been possible without their support and ceded wind park data.

Conflicts of Interest: The authors declare no conflict of interest. The founding sponsors had no role in the design of the study; in the collection, analyses, or interpretation of the data; in the writing of the manuscript, and in the decision to publish the results.

References

1. Reader, G.T. Energy, Renewables Alone? In *Sustaining Resources for Tomorrow*; Springer: Berlin/Heidelberg, Germany, 2020; pp. 1–45.
2. Europe, W. *Wind Energy in Europe in 2018—Trends and Statistics*; Wind Europe: Brussels, Belgium, 2019.
3. Agency, I.R.E. *Renewable Power Generation Costs in 2019*; RENA: Abu Dhabi, United Arab Emirates, 2020.
4. Stehly, T.J.; Beiter, P.C. *2018 Cost of Wind Energy Review*; Technical Report; National Renewable Energy Lab. (NREL): Golden, CO, USA, 2020.

5. Kumar, A.; Shankar, R.; Thakur, L.S. A big data driven sustainable manufacturing framework for condition-based maintenance prediction. *J. Comput. Sci.* **2018**, *27*, 428–439. [[CrossRef](#)]
6. Ziegler, L.; Gonzalez, E.; Rubert, T.; Smolka, U.; Melero, J.J. Lifetime extension of onshore wind turbines: A review covering Germany, Spain, Denmark, and the UK. *Renew. Sustain. Energy Rev.* **2018**, *82*, 1261–1271. [[CrossRef](#)]
7. Leahy, K.; Gallagher, C.; O'Donovan, P.; O'Sullivan, D.T. Issues with data quality for wind turbine condition monitoring and reliability analyses. *Energies* **2019**, *12*, 201. [[CrossRef](#)]
8. Vidal, Y.; Pozo, F.; Tutivén, C. Wind turbine multi-fault detection and classification based on SCADA data. *Energies* **2018**, *11*, 3018. [[CrossRef](#)]
9. Leahy, K.; Hu, R.L.; Konstantakopoulos, I.C.; Spanos, C.J.; Agogino, A.M.; O'Sullivan, D.T. Diagnosing and predicting wind turbine faults from SCADA data using support vector machines. *Int. J. Progn. Health Manag.* **2018**, *9*, 1–11.
10. Zhao, Y.; Li, D.; Dong, A.; Kang, D.; Lv, Q.; Shang, L. Fault prediction and diagnosis of wind turbine generators using SCADA data. *Energies* **2017**, *10*, 1210. [[CrossRef](#)]
11. Helbing, G.; Ritter, M. Deep Learning for fault detection in wind turbines. *Renew. Sustain. Energy Rev.* **2018**, *98*, 189–198. [[CrossRef](#)]
12. Cambron, P.; Masson, C.; Tahan, A.; Pelletier, F. Control chart monitoring of wind turbine generators using the statistical inertia of a wind farm average. *Renew. Energy* **2018**, *116*, 88–98. [[CrossRef](#)]
13. Li, Y.; Wu, Z. A condition monitoring approach of multi-turbine based on VAR model at farm level. *Renew. Energy* **2020**, *166*, 66–80. [[CrossRef](#)]
14. Astolfi, D.; Castellani, F.; Lombardi, A.; Terzi, L. Multivariate SCADA Data Analysis Methods for Real-World Wind Turbine Power Curve Monitoring. *Energies* **2021**, *14*, 1105. [[CrossRef](#)]
15. Hart, E.; Turnbull, A.; Feuchtwang, J.; McMillan, D.; Golysheva, E.; Elliott, R. Wind turbine main-bearing loading and wind field characteristics. *Wind. Energy* **2019**, *22*, 1534–1547. [[CrossRef](#)]
16. Hart, E.; Clarke, B.; Nicholas, G.; Kazemi Amiri, A.; Stirling, J.; Carroll, J.; Dwyer-Joyce, R.; McDonald, A.; Long, H. A review of wind turbine main bearings: Design, operation, modelling, damage mechanisms and fault detection. *Wind. Energy Sci.* **2020**, *5*, 105–124. [[CrossRef](#)]
17. Beretta, M.; Julian, A.; Sepulveda, J.; Cusidó, J.; Porro, O. An Ensemble Learning Solution for Predictive Maintenance of Wind Turbines Main Bearing. *Sensors* **2021**, *21*, 1512. [[CrossRef](#)] [[PubMed](#)]
18. Encalada-Dávila, Á.; Puruncajas, B.; Tutivén, C.; Vidal, Y. Wind Turbine Main Bearing Fault Prognosis Based Solely on SCADA Data. *Sensors* **2021**, *21*, 2228. [[CrossRef](#)]
19. McKinnon, C.; Carroll, J.; McDonald, A.; Koukoura, S.; Infield, D.; Soraghan, C. Comparison of new anomaly detection technique for wind turbine condition monitoring using gearbox SCADA data. *Energies* **2020**, *13*, 5152. [[CrossRef](#)]
20. Zhang, Z. Missing data imputation: Focusing on single imputation. *Ann. Transl. Med.* **2016**, *4*, 9. [[CrossRef](#)]
21. Kayri, M. Predictive abilities of bayesian regularization and Levenberg–Marquardt algorithms in artificial neural networks: A comparative empirical study on social data. *Math. Comput. Appl.* **2016**, *21*, 20. [[CrossRef](#)]
22. Foresee, F.D.; Hagan, M.T. Gauss-Newton approximation to Bayesian learning. In Proceedings of the International Conference on Neural Networks (ICNN'97), IEEE, Houston, TX, USA, 9–12 June 1997; Volume 3, pp. 1930–1935
23. Bartilson, D.T.; Jang, J.; Smyth, A.W. Finite element model updating using objective-consistent sensitivity-based parameter clustering and Bayesian regularization. *Mech. Syst. Signal Process.* **2019**, *114*, 328–345. [[CrossRef](#)]
24. Glorot, X.; Bengio, Y. Understanding the difficulty of training deep feedforward neural networks. In Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics. JMLR Workshop and Conference Proceedings, Chia Laguna Resort, Sardinia, Italy, 13–15 May 2010; pp. 249–256
25. Liu, F.T.; Ting, K.M.; Zhou, Z.H. Isolation Forest. In Proceedings of the 2008 Eighth IEEE International Conference on Data Mining, Pisa, Italy, 15–19 December 2008; pp. 413–422. [[CrossRef](#)]
26. Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; et al. Scikit-learn: Machine Learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825–2830.
27. Wang, G.; Jia, R.; Liu, J.; Zhang, H. A hybrid wind power forecasting approach based on Bayesian model averaging and ensemble learning. *Renew. Energy* **2020**, *145*, 2426–2434. [[CrossRef](#)]
28. Lee, J.; Wang, W.; Harrou, F.; Sun, Y. Wind Power Prediction Using Ensemble Learning-Based Models. *IEEE Access* **2020**, *8*, 61517–61527. [[CrossRef](#)]
29. Wu, Z.; Lin, W.; Ji, Y. An integrated ensemble learning model for imbalanced fault diagnostics and prognostics. *IEEE Access* **2018**, *6*, 8394–8402. [[CrossRef](#)]
30. Hariri, S.; Kind, M.C.; Brunner, R.J. Extended Isolation Forest. *arXiv* **2018**, arXiv:1811.02141.