

ORIGINAL ARTICLE

Wind turbine fault detection and isolation robust against data imbalance using KNN

Ali Fazli | Javad Poshtan 

Department of Electrical Engineering,
Iran University of Science and
Technology, Tehran, Iran

Correspondence

Javad Poshtan, Iran University of Science
and Technology, Narmak 16846-13114,
Tehran, Iran.
Email: jposhtan@iust.ac.ir

Abstract

Due to the difficulties of system modeling, nonlinearity effects, uncertainties, and the availability of Wind Turbines (WTs) SCADA system data, data-driven Fault Detection and Isolation (FDI) methods for WTs have received increasing attention. In this paper, using the wind turbine SCADA data, an effective FDI scheme is proposed using the *K*-Nearest Neighbors (KNN) classifier. The operational data set is labeled by the status and warning data sets, and the labeled operational data set, after eliminating invalid data, feature selection, and standardization, is used for training and validation of the FDI model. Data imbalance, which is common in real data sets, does not affect the performance of the proposed method, hence there is no need for data balancing methods in this algorithm and the performance is not deteriorated by occurring false alarms. Therefore, the proposed method has provided impressive performance in FDI compared with previous research on this data set. Also, many of the fault classes addressed in this paper were not considered in previous works on this data set.

KEY WORDS

data imbalance, fault detection and isolation, KNN, SCADA data, wind turbine

1 | INTRODUCTION

Wind energy which is renewable and compatible with the environment is developing rapidly and in this regard, large-scale Wind Turbines (WTs) are used in various countries. Therefore, in recent years, the monitoring and maintenance of WTs have attracted much attention. In a harsh environment, fault detection and condition monitoring are necessary for WTs safety and reliability.¹ Fault scenarios in WTs include sensor faults, actuator faults, and system faults.² Common faults and failures in WTs and the various methods that researchers use to diagnose them have been investigated and classified in many

studies.^{1,3,4} From one perspective, fault detection and isolation (FDI) methods in WTs are divided into three main categories: model-based, data-driven, and hybrid approaches. In model-based methods, a mathematical model of the WT or its subsystem produces a simulated output based on the measured input signal. By comparing the actual output of WT with the estimated output, a residual is produced to be used for FDI.⁵ Due to the system modeling difficulties and the availability of sensors data, data-driven methods in fault detection of WTs have gained increasing interest compared with the model-based approaches.⁶ Data-driven FDI methods in WTs include acoustic emission analysis, noise analysis,

This is an open access article under the terms of the [Creative Commons Attribution](#) License, which permits use, distribution and reproduction in any medium, provided the original work is properly cited.

© 2024 The Authors. *Energy Science & Engineering* published by Society of Chemical Industry and John Wiley & Sons Ltd.

oil analysis, vibration analysis, machine learning (data mining) methods, and hybrid methods.¹ Major disadvantages of acoustic emission analysis and oil analysis methods are the need for background noise to be shielded and limitation to bearings with the closed-loop oil supply system, respectively.⁷ Vibration analysis methods^{8,9} which are effective tools for the condition monitoring and fault diagnosis of WT drivetrains,¹⁰ despite the reliability and standardization are expensive, intrusive, subject to sensor failures, and have limited performance for low-speed rotation.⁷

Machine learning algorithms are divided into two main categories: supervised learning which consists of regression-based and classification-based approaches, and unsupervised learning algorithms such as clustering.¹¹ In regression methods, which are divided into parametric and nonparametric modeling techniques, an artificial intelligence (AI) model learns the dynamics of a WT or a WT subsystem from the patterns of input and output collected from a data set,⁵ and the predicted behavior of this model is compared with the WT measured actual behavior to detect possible faults from the changes in the WT behavior.¹² On the other hand, in the classification methods, an AI model is trained to recognize the patterns (e.g., mode, location, and severity) of different WT faults from the input signals containing the faults information.⁵ The main steps of classification are data acquisition, preprocessing, equalization of the classes, feature selection and extraction, model fitting, cross-validation, and using the best model based on validation.¹¹ The amount of normal data is much more than that of abnormal data in SCADA data sets, which makes FDI models tend to be biased toward the majority class, that is, normal data, leading to poor accuracy in diagnosing faults.¹³ Several methods such as setting misclassification cost, undersampling majority class, oversampling minority class (e.g., synthetic minority oversampling technique and adaptive synthetic sampling method¹⁴), Tomek links, and cluster centers are used to overcome data imbalance challenge.¹⁵ However, the main focus is to use an appropriate method from the existing methods or develop new algorithms according to the need, because if one technique works well on an imbalanced data set, it may not work for another imbalanced data set.¹⁶

The main machine learning models already used in the wind turbine FDI problem are Artificial Neural Networks (ANNs), Support Vector Machines (SVMs), K-Nearest Neighbors (KNN), classifier fusion, and ensemble classifiers. Wind turbine SCADA data include multivariate time series that have temporal correlations within each sensor variable and spatial correlations between different sensor variables. To effectively capture

these correlations, in He et al.¹⁷ two parallel modules are used for feature extraction; first, a MultiScale Deep Echo State Network (MSDeepESN) for extracting temporal multiscale features, and second, a MultiScale Residual Network (MSResNet) that can extract spatial multiscale features. Due to the extreme data imbalance, the focal loss function is used in He et al.¹⁷ to reduce its effect on model training. In Pang et al.,¹⁸ first, a MultiKernel Fusion Convolution Neural Network (MKFCNN) is designed to extract multiscale spatial correlations of different features, and then the Long Short-term Memory (LSTM) is used for learning the temporal correlations among the learned spatial features. After applying a down-sampling method to the training data in Pang et al.¹⁸ to balance normal and faulty samples, only 210 observations from 24,108 normal samples are used for the training process. Although using temporal correlations for the training and validation of the FDI system can provide useful information, it leads to the problem of the algorithm's dependence on historical data for FDI. Using a WT real SCADA data based on the SVM learning method, a fault diagnosis and prediction model is presented in Leahy et al.,¹⁵ and despite the use of various data balancing methods, the desired accuracy is not reached because of the lack of attention to the fact that transient observations before and after fault occurrence must be removed. As an alternative to the SVM technique, the Gaussian process classifier (GPC), as a Bayesian nonparametric classification method, which gets away from any assumptions about the structural relationship between inputs and the output, provides probabilistic information about the fault types, which is valuable for making maintenance plans.¹⁹ Using an undersampling method, just 460 samples are representative of the 24,108 normal observations of the training data. Therefore, a lot of information is lost by ignoring a large number of samples in the model training process in many studies.^{15,18,19} A data-driven FDI scheme is proposed in Pashazadeh et al.²⁰ based on the fusion of the decision tree (DT), SVM, KNN, Radial Basis Function (RBF), and MultiLayer Perceptron (MLP) classifiers. Although this algorithm is robust against different operational conditions and measurement errors, according to the confusion matrix of Pashazadeh et al.,²⁰ it is biased for some minority classes toward the majority class. On the other hand, using data balancing methods may prevent faulty observations from being detected as normal samples, but it can result in false alarms according to the confusion matrices of many studies.^{15,17-19} Figure 1 shows various methods of detection and isolation of WT faults.

KNN is a nonparametric supervised learning algorithm that can be used for regression and classification

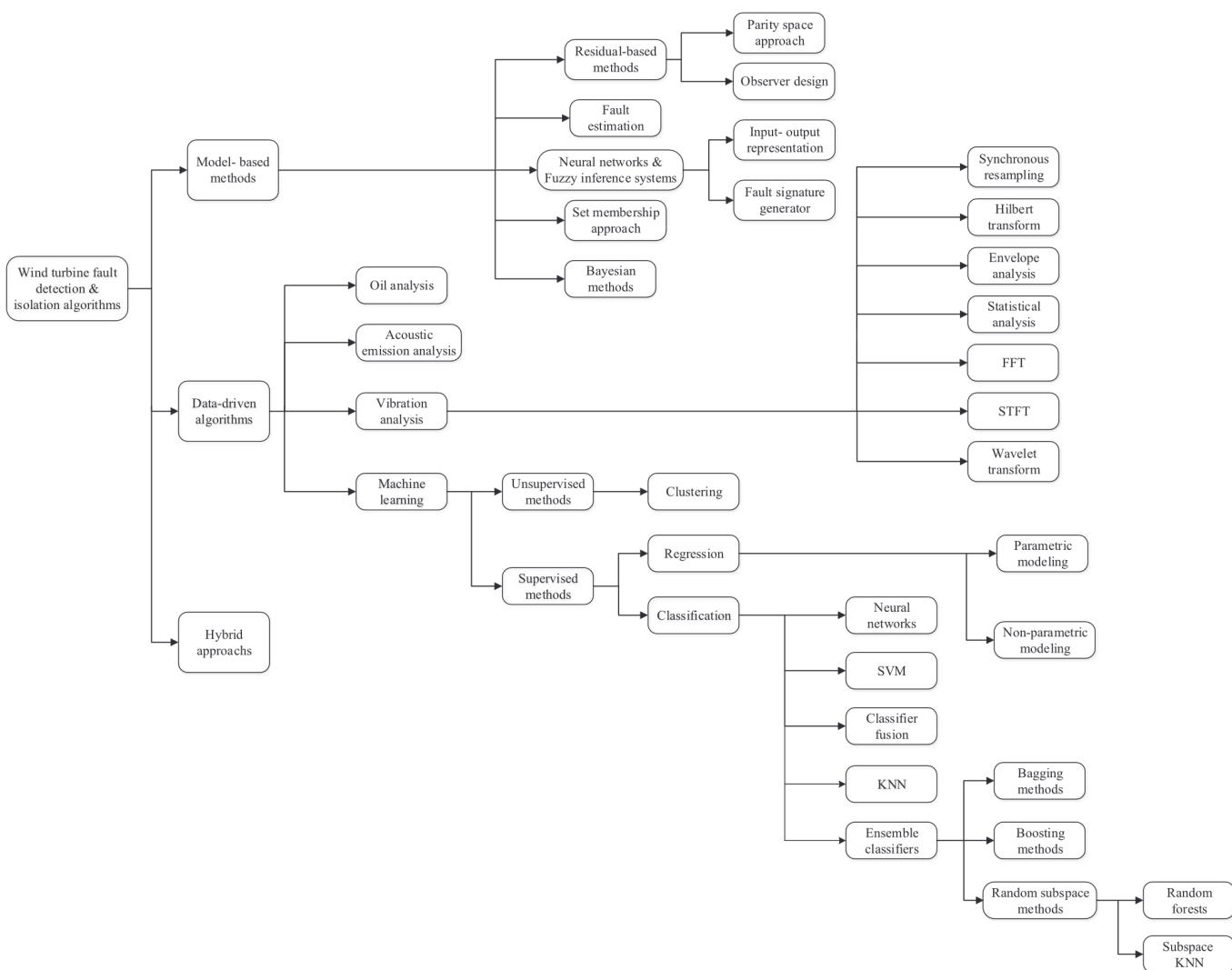


FIGURE 1 Various methods used for WT fault detection and isolation. FFT, fast Fourier transform; KNN, K-Nearest Neighbors; STFT, short-time Fourier transform; SVM, Support Vector Machine; WT, Wind Turbine.

predictions. Although KNN is a simple and accurate algorithm it has some weaknesses such as being biased toward majority observations when facing data imbalance.²¹ Several types of research have been performed to improve KNN performance against data imbalance by using oversampling,²² boosting-by-resample strategy,²³ misclassification cost,²⁴ and so forth. This algorithm has been used for fault detection in a wide area of subjects such as power systems,^{25,26} railway point systems,²⁷ nuclear power plants,²⁸ and especially WTs,^{20,29–31} however, the focus of previous FDI works is not mainly on data imbalance challenge.

In this paper, an FDI method is presented using WT real SCADA data. First, operational SCADA data are labeled using messages of the status and warning data sets. Note that each status and warning data set includes various faults, while in previous researches only some of the status data set faults have been considered. Second,

after eliminating invalid data observations, informative features are selected from the original features' set to form an appropriate feature set. Third, the features selected are standardized in such a way that each feature has zero mean and unit standard deviation, and then the pre-processed operational data set acts as the KNN input. Finally, WT fault detection and isolation is performed and the holdout validation method verifies good accuracy of the KNN method. The motivation of this paper is to present a data-driven method for wind turbine FDI so that the problem of data imbalance as an important challenge of the literature is solved using the potential of machine learning methods without using data balancing techniques. As an example, in the undersampling method only some sample observations of the majority class, as a representative of this class, are used for the training process, and obviously, these samples cannot fully represent the behavior of the no-fault (majority) class.

Using data balancing methods for handling imbalanced data sets can lead to false alarms, as it can be seen in the confusion matrices of several studies.^{15,17–19} Note that using undersampling methods makes the training data balanced but validation data (as in real practical operation) is still imbalanced, hence, this inconsistency between training and validation data results in false alarms, that is, some useful information is lost by undersampling the majority class of the training data. This is while the proposed algorithm can perform the training and validation processes without the requirement for these interface methods and with the benefit of the capacity of all data set observations. The contributions of this paper are:

- (1) The Model is robust against imbalanced data, which prevents the classifier from tending to the majority class. Imbalanced classification is one of the most important challenges of wind turbine FDI methods because faults occur infrequently compared with normal operation.
- (2) Due to the algorithm's robustness against imbalanced data, there is no need for data balancing methods. Therefore, false alarms are less likely to occur and the classifier's performance is not deteriorated.
- (3) These two advantages make the algorithm have good performance in WT fault detection and isolation.
- (4) Although the collected data set which is used for training the FDI system is offline, the algorithm can determine the labels of the new data observations online.

- (5) The proposed FDI system determines the label of the new data observation only with its current feature values, without any requirement for historical data.
- (6) Existing SCADA data are used to train and validate the algorithm, so there is no extra cost to collect and record the data.
- (7) A large number of status and warning faults are considered (in comparison with previous works on this SCADA data^{15,17–19}).
- (8) The dependence between different faults based on physical facts and the overlap of these faults' occurrence is analyzed.

The article structure is as follows: Section 2 describes the wind turbine SCADA data set. In Section 3 data preprocessing, including labeling, eliminating invalid data, feature selection, and standardization is explained. Section 4 presents the parameter selection of the wind turbine FDI model, and Section 5 includes KNN training and validation results. Finally, in Section 6 conclusion and future work are discussed. Figure 2 shows the flowchart of the proposed scheme for WT fault detection and isolation.

2 | DATA DESCRIPTION

The real SCADA data in this study come from a 3-MW direct-drive WT that supplies the required power of a major biomedical manufacturing plant located off the

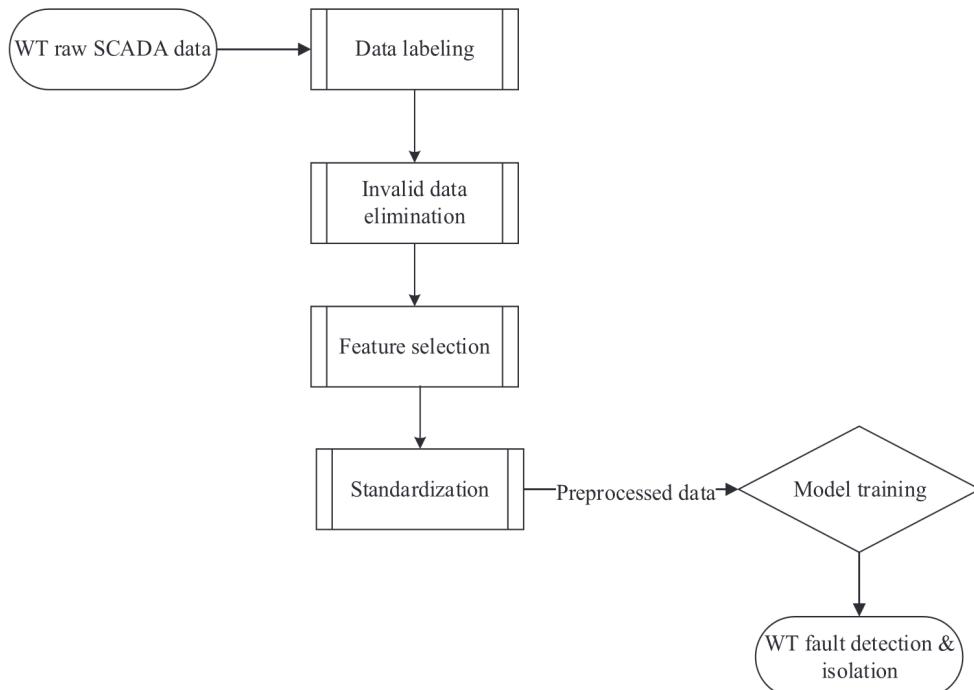


FIGURE 2 Flowchart of the proposed scheme for WT fault detection and isolation. WT, Wind Turbine.

southern coast of Ireland.^{15,17–19,32} This three-blade WT comprises common major components, including main shaft, generator, rotor, turbine blades, yaw system, pitch system, control and power electronics system, hydraulic and cooling system, and so forth. The data include observations over an 11-month period from May 2014 to April 2015 which contains three separate data sets:

- (1) Operational data including 49,027 samples with a 10-min time interval and 66 features, such as ambient parameters (e.g., wind speed and ambient temperature), power measurements (e.g., active power and reactive power), WT components recorded temperature (e.g., tower temperature, nacelle cabinet temperature, etc.), and operating conditions (e.g., nacelle position including cable twisting, operating hours).
- (2) Status data that represents the operating conditions of the WT and is divided into two sets of data: wind energy converter (WEC) and remote terminal unit (RTU). The WEC data includes status messages that are directly related to the WT itself, while the RTU data corresponds to the power control data at the point of connection to the grid, such as active and reactive power. In the status data set, a status message with a new timestamp is generated each time the status changes. Therefore, it is assumed that the turbine operates in that mode until the next status message is generated. Each turbine state has a “main status” code and a “substatus” code, where each main status code greater than zero indicates an abnormal condition and not necessarily a fault; for example, status code 2 (“lack of wind”) indicates the lack of sufficient wind for normal operation of a WT.
- (3) Warning data that corresponds to the general information about the turbine and is not directly related to the turbine operation or safety. Warning messages are about potentially developing faults in the turbine. These messages have timestamps in the same way as status messages and also have a “main warning” code and a “subwarning” code. This data is also divided into two sets of WEC data and RTU data. The WEC data includes warning messages that are directly related to the turbine itself, while the RTU data are related to the power control data at the point of connection to the grid.

3 | PREPROCESSING

Before applying the FDI model to the operational data, as its input, some steps must be taken. First, considering the messages of the status and warning data sets, the operational data set is labeled. Then some observations

must be removed to prevent feeding the model with invalid data samples. In the next step, noninformative features are removed and z -score normalization is applied to the selected set of features. Note that one of the most important preprocessing steps in classification-based FDI problems, called data balancing, is eliminated by taking advantage of using the KNN model.

3.1 | Data labeling

The WT operational data set is labeled based on the information in the status and warning data sets so that the observations during faults occurrence are distinguished from other observations. Table 1 shows status data faults along with status and substatus codes, and the symbols assigned to each fault class so that fault classification and evaluation are performed easily. Faults that have not been addressed in previous works on this data set are highlighted in green. Feeding faults refer to faults in the power feeder cables of the WT, excitation errors are mainly due to problems with the generator excitation system, malfunction air-cooling indicates problems in the air circulation and internal temperature circulation in the WT, mains failure is related to problems with mains electricity supply to the WT, and generator heating faults refer to the generator overheating. Other faults are WT cable twist, inverter overtemperature, WT tower transversal oscillation, fan-inverter malfunction, and yaw control fault. Among these, Feeding fault and Excitation error have overlaps in 95 samples. Also, 12 samples have the labels of Malfunction air cooling and Mains failure at the same time. As these overlaps cannot be ignored and include a significant proportion of the number of samples of each fault, either multilabel classification methods should be used, the overlap section should be considered a separate class, or the classes with significant overlap should be integrated. The third solution is adopted, and for this purpose, the modified classes are symbolized as EFF and AMF, respectively. In Table 2 several faults extracted from the WT warning data and used to label the operational data are shown. None of these faults have been investigated in previous researches on this data set and hence all are highlighted in green.

According to the symbols defined for each fault, Table 3 presents the number of overlapping samples between the faults defined in Tables 1 and 2. Overlaps with less than 10% of both faults' samples are negligible and not used in the training and testing of the proposed method. For example, without considering overlap samples, the classes BF and FA with 144 and 271 samples, respectively, have 49 common samples, which

TABLE 1 Status data faults.

Fault	Status code	Substatus code	Number of samples	Symbol
Generator heating: Hygrostat inverter	9	3	43	GF
Feeding fault	62	Various	260	FF
Excitation error: Overvoltage DC-link	80	21	175	EF
Timeout warn message: Malfunction air cooling	228	100	62	AF
Mains failure: Under voltage L1/start delay	60	11/2	20	MF
Cable twisted: Left/right (2–3 turns)	21	1/2	43	CF
Overtemperature: Power choke inverter 4	67	604	12	OF
Tower oscillation: Transversal oscillation	31	1	3	TF
Malfunction fan-inverter: Other control board errors system 3	26	373	3	IF
Yaw control fault: TWK NOCN: fault in cam switch mechanism	22	226	5	YF

TABLE 2 Warning data faults.

Fault	Warning code	Subwarning code	Number of samples	Symbol
Faulty yaw inverter: Rotational speed error motor x/DC-link voltage instability system y/Feedback mains contactor faulty (–)	25	Various	28	YIF
Yaw control fault: TWK NOCN installed	22	200	32	YCF
Fault blade load control: Blade load curve A/B/C too low (28)	49	193/293/393	195	BF
Fault air cooling: Generator pressure to low (7)	100	11	327	FA
Fault water cooling: Pressure too low (7)/No flow rate (7)	101	1/5	85	FWC
Fault lubrication system: Grease reservoir empty (rotor) (90)	58	1	140	LF
Tower oscillation: Longitudinal oscillation (low wind speed)	31	32	42	TLF
Malfunction fan-inverter: DC-link voltage unstable system 3/Circuit breaker nacelle fan tripped	26	316, 331, 335	26	MFI

TABLE 3 Number of the faults overlapping samples.

Overlap fault	Number of each fault samples without considering the overlap	Number of overlap samples	Symbol	Can be ignored
BF + FA	144 + 271	49	BFA	No
YIF + MFI	19 + 19	5	YIMFI	No
FA + FWC	271 + 73	5	FAWC	Yes
LF + FWC	131 + 73	7	LFWC	Yes
LF + FA	131 + 271	2	LFA	Yes
BF + YIF	144 + 19	2	BYIF	Yes
AF + MF	50 + 8	12	AMF	No
EF + FF	80 + 165	95	EFF	No

exceeds 10% of each class sample size; hence this overlap cannot be ignored. On the other hand, some faults in nature are so close together that have the same main code. Accordingly, faults that are conceptually very close

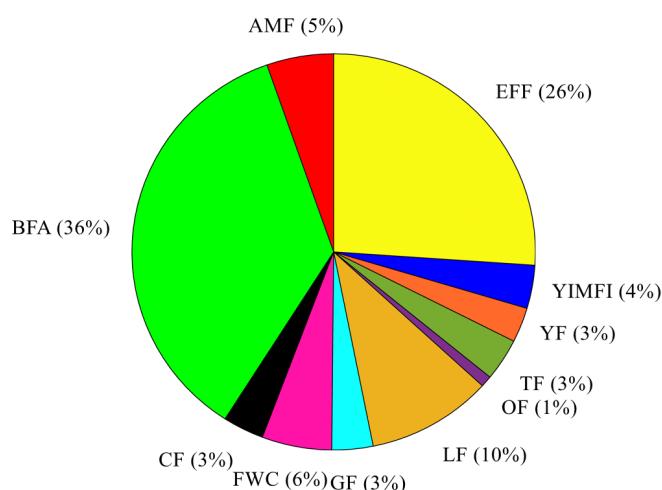
to each other can also be considered as an integrated class. Therefore, two types of class integration have been considered in this paper: first, class integration based on the number of common (overlap) samples in the

TABLE 4 Integrated classes based on the fault nature.

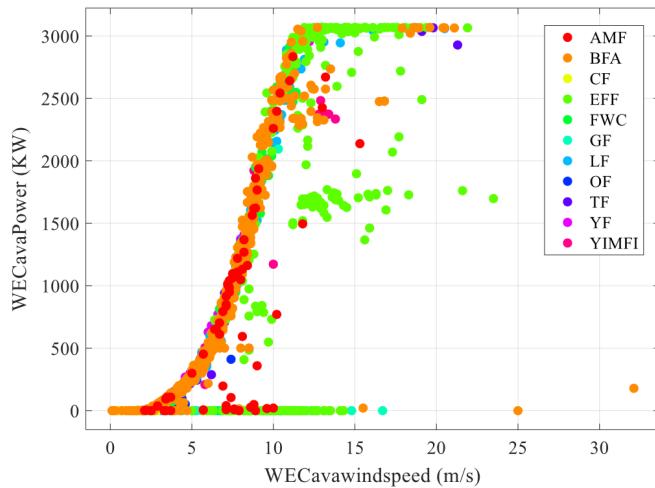
Fault 1	Number of samples	Fault 2	Number of samples	Integrated class symbol	Number of samples
YF	5	YCF	32	YF	37
TF	3	TLF	42	TF	45
IF	3	MFI	19	MFI	22

TABLE 5 Final fault labels.

Fault	Decomposition	Number of samples
BFA	BF + FA	464
YIMFI	YIF + MFI + IF	46
EFF	EF + FF	340
AMF	AF + MF	70
TF	TF + TLF	45
YF	YF + YCF	37
GF	GF	43
CF	CF	43
OF	OF	12
FWC	FWC	73
LF	LF	131

**FIGURE 3** Relative frequency of the WT's considered faults. WT, Wind Turbine.

operational data, and second, class integration based on the nature of faults. Table 4 lists the faults that are merged accordingly. Therefore, the final fault labels used for data labeling are given in Table 5 and the relative frequency of these classes is shown in Figure 3 using a pie chart. Figure 4 shows the scatter plot of the WT's

**FIGURE 4** Scatter plot of the wind turbine faulty operational data on the power curve.

faulty samples on the power curve according to labels presented in Table 5.

3.2 | Invalid data elimination

Some observations which indicate abnormal behavior of the WT are not labeled with fault labels, because many of these are not associated with a fault, for example, status code 2—"lack of wind". Some of these abnormal conditions such as warning code 230—"Power limitation (10 h)" are not considered as the no-fault class in data labeling. According to Pang et al.,¹⁸ at least 120 min before the change from the normal state and 30 min after the change to the normal state should be considered as transient state data. This is due to the fact that, unlike in simulation, in a real SCADA data set, the change from normal to abnormal performance (and vice versa) occurs with soft behavior, so the data in this transition state should not be considered as normal WT performance data. In this paper with less conservative data cleaning, at least 60 min before the change from the normal state and 20 min after the change to the normal state are considered as transient observations. Also, sometimes a fault or a group of faults occur intermittently with a

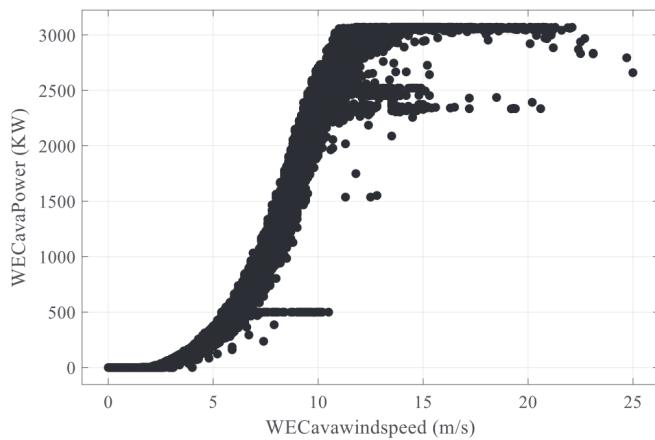


FIGURE 5 Scatter plot of the wind turbine normal operational data on the power curve.

small time interval, which indicates that the problem of the turbine is not solved between consecutive faults. Therefore after filtering invalid normal data, we have 34,472 observations as the no-fault class which is labeled with the symbol NF. Note that in many studies^{17–19} 32,056 observations are considered as normal class. This means that in this paper, the normal class is more in line with the conditions provided by the wind turbine SCADA data set. Figure 5 shows the scatter plot of the WT's normal operational data on the power curve. According to Figures 4 and 5, most observations of some fault types such as fault BFA are not separable from normal class samples on the power curve. Hence, it is important to use information within all informative data features according to Section 3.3.

3.3 | Feature selection and standardization

Some of the 66 variables of operational data must be used in the FDI process. In short, the following steps are taken to select appropriate features, resulting in the final set of features displayed in Table 6.

- ✓ Date Time is not a numeric feature and is eliminated.
- ✓ Variables such as Time, WEC: Operating Hours, WEC: Production kWh, and WEC: Production minutes that are accumulative features result in time-based FDI if considered in the training and validation process. Therefore, even if the accuracy obtained is high, it will not be reliable. For example, if a specific fault occurs in March in the training set, the test sample can only be correctly predicted if it occurs in the same month. Hence, to prevent the algorithm from time-based bias these features are not considered in the FDI process.

TABLE 6 Final SCADA variables.

No.	Variable name	No.	Variable name
1	Ava wind speed	14	Nacelle ambient temp (averaged)
2	Ava Rotation	15	Nacelle temp
3	Ava Power	16	Nacelle cabinet temp
4	Ava Nacelle's position including cable twisting	17	Main carrier temp
5	Ava Reactive Power	18	Rectifier cabinet temp
6	Ava blade angle A	19	Inverter cabinet temp (averaged)
7	Sys1 inverter cabinet temp (averaged)	20	Ambient temp
8	Spinner temp	21	Tower temp
9	Bearing temp (averaged)	22	Control cabinet temp
10	Pitch cabinet blade temp (averaged)	23	Transformer temp
11	Blade temp (averaged)	24	Inverter averages
12	Rotor temp (averaged)	25	Inverter standard deviation
13	Stator temp (averaged)	26	Status (label)

- ✓ For variables recorded as minimum, maximum, and average, it is better to use only the average because the minimum and maximum values may be affected by outliers and deteriorate the performance of the algorithm. Wind speed, Rotation, Power, and Reactive power are of this type.
- ✓ Some parameters including Sys1 inverter cabinet temp, Bearing temp, Pitch cabinet blade temp, Blade temp, Rotor temp, Stator temp, Nacelle ambient temp, and Inverter cabinet temp are recorded by more than one sensor (seven, two, three, three, two, two, two, and two, respectively). Therefore, to prevent the curse of dimensionality, which is an important challenge in the KNN algorithm, these variables are merged into one variable by averaging. For example, Front bearing temp and Rear bearing temp are variables that result in one variable (Bearing temp) by averaging.
- ✓ Some noninformative features that reduce the accuracy of the algorithm are ignored.

Numerical features must be standardized before feeding to the KNN algorithm to contribute equally to the similarity measures. Standardization (z-score normalization) is a scaling method where the values are centered around the mean with a unit standard

deviation. This means that the mean of the attribute becomes zero, and the resultant distribution has a unit standard deviation, that is,

$$X'_i = \frac{X_i - \mu_i}{\sigma_i}, \quad (1)$$

where X_i represents the i th original feature, μ_i and σ_i are its mean and standard deviation, respectively, and X'_i is the standardized i th feature for $i = 1, \dots, 25$.

4 | KNN PARAMETER SELECTION

Considering the final fault classes, presented in Table 5, the preprocessed operational data are used to determine the FDI model parameter based on the classification error. A KNN classifier is trained for different values of the parameter K . Figure 6 shows the effect of changes in this parameter on the classification error by the 10-fold validation method. According to this figure, increasing the value of parameter K results in increasing the classification error, so $K=1$ is the best choice for this classifier. Choosing this value usually can yield overfitting, hence to avoid this, the holdout validation is adopted in the procedure of the wind turbine FDI model design. Meanwhile, the value of K in the KNN algorithm should be normally chosen an odd number because the choice of an even number for K may lead to a risk of tie in the decision of which label must be assigned to a new instance. Therefore, the next option

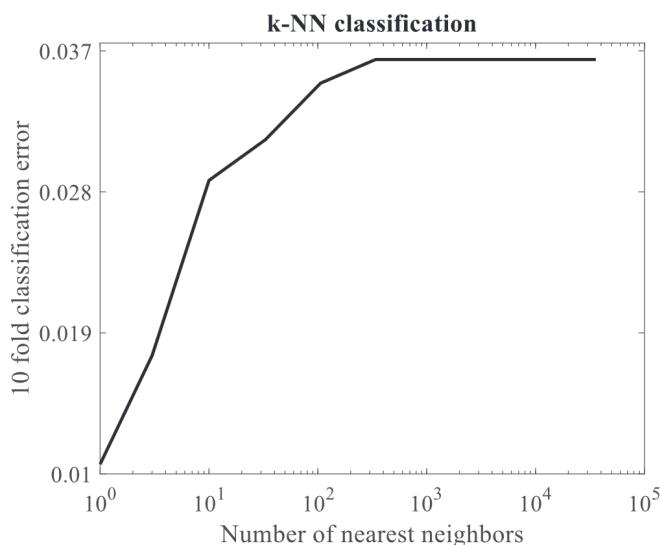


FIGURE 6 Classification error for different values of parameter K . KNN, K-Nearest Neighbor.

is $K = 3$ which is not appropriate due to the high imbalance in data, that is, some classes have so few observations, compared with the majority class, that are not rich enough to be distinguished from the majority class by the algorithm. For example, fault OF only has 12 samples (including training and validation observations), while NF has 34,472 samples, and when the KNN considers three nearest neighbors of a validation instance, there is a much higher risk of tendency to the majority class compared with the case $K = 1$. As shown in Figure 6, the KNN classifier for $K = 1$ can well classify different classes of WT imbalanced SCADA data.

Many references confirm that the performance of KNN is affected by data imbalance.^{21,24,33} Conceptually the KNN algorithm calculates the (Euclidian) distances of the training data set observations from a validation sample (i.e., to be labeled) and assigns the majority label of KNNs of the validation sample to it. Therefore, if the observations of each class (in the feature space of the data set) have little distance from each other and a small parameter K is selected, the performance of this algorithm will not be affected by data imbalance. In other words, if the distance of samples within a class is less than the distance between samples of different classes, and a small parameter K is selected, this algorithm will be robust against data imbalance. For example, as shown in Figure 7, let's assume that one class of the data set has many samples while another class has few observations, but these samples are well close to each other in the feature space. In this case, considering $K = 1$, the KNN algorithm will assign a true label to the minority class samples and will not be affected by the data imbalance. Also, according to Figure 6, by increasing the parameter K , the error reaches 0.03645, which means that the algorithm tends to the majority class: Note that $0.03645 * 35,776 = 1304$, which is exactly equivalent to the false prediction of the minority classes samples (1304 fault observations).

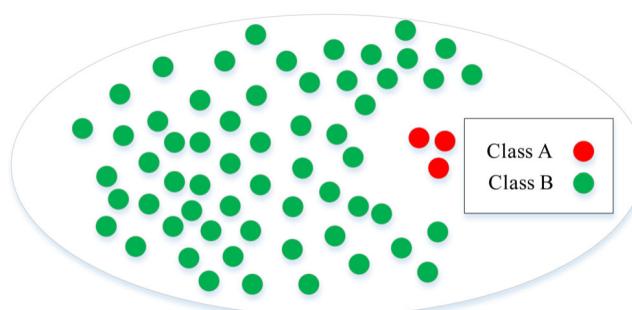


FIGURE 7 An example of minority and majority classes separable by K -Nearest Neighbor.

5 | TRAINING AND VALIDATION

After choosing suitable parameters according to the previous section's explanations, the KNN model is applied to the train data, which is 80% of the data. Therefore, 20% of the data is used as validation data, employing the holdout validation method, thus none of the validation data samples are involved in the training process. Considering the faults given in Table 5, the KNN model results in a good performance as shown in Figure 8. This confusion chart illustrates that a high percentage of the validation data samples are assigned properly to the true classes. The following parameters are used to clarify this performance:

$$TPR = \frac{TP}{P} = \frac{TP}{TP + FN}, \quad (2)$$

$$TPR_{ave} = \frac{\sum_{m=1}^C TPR_m}{C}, \quad (3)$$

$$PPV = \frac{TP}{TP + FP}, \quad (4)$$

$$PPV_{ave} = \frac{\sum_{m=1}^C PPV_m}{C}, \quad (5)$$

$$F1 = \frac{2 \times TPR \times PPV}{TPR + PPV}, \quad (6)$$

$$F1_{ave} = \frac{\sum_{m=1}^C F1_m}{C}, \quad (7)$$

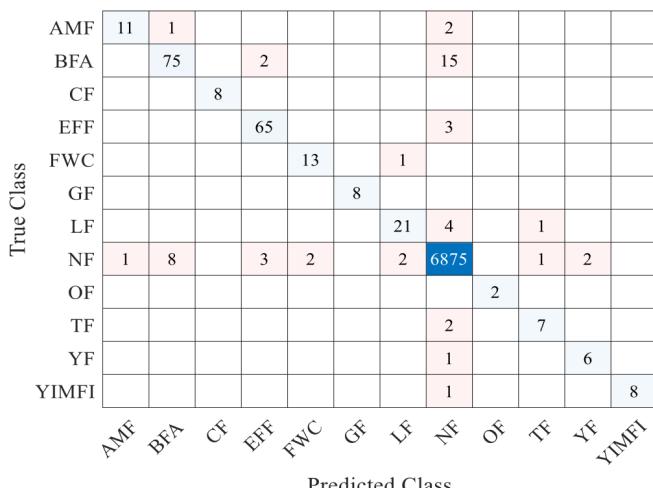


FIGURE 8 The confusion chart of the proposed model validation.

where C is the number of classes including no-fault and faulty categories, TP is the number of true predictions of the class, FN represents the number of false predictions of the class, P indicates the number of all samples of the class, and FP is the representative of the false alarms of the class. Note that average TPR in (3), average PPV in (5), and average $F1$ -score in (7), which are proposed as validation parameters in this paper, can be useful for the validation of imbalance classification. The values of TPR and PPV obtained for all classes are greater than 77.8% and 75%, respectively, according to Figure 8, confirming the good performance of the proposed model in the WT fault detection and isolation. Note that the training and validation data are chosen randomly based on the 80-20 proportions. So, each time the training and validation process takes place, the results may be slightly different. In multiple runs, however, the number of false predictions does not exceed 75, which is a good achievement compared with the previously reported methods. Figure 6 also confirms this statement because almost 0.01062 10-fold classification error is equal to $0.01062 * 0.2 * 35,776 \approx 76$ false predictions in the holdout validation. Figure 9 shows the Receiver Operating Characteristic (ROC) and Area Under Curve (AUC) of the proposed FDI system for all considered classes. According to this figure, all of the curves are in the desired area and AUC values confirm that the classification is performed well. Table 7 compares the proposed FDI model with related previous works on this data set, in terms of accuracy, the number of the fault classes considered, the number of the normal class samples, overall accuracy (the ratio of the number of the whole true predictions to the number

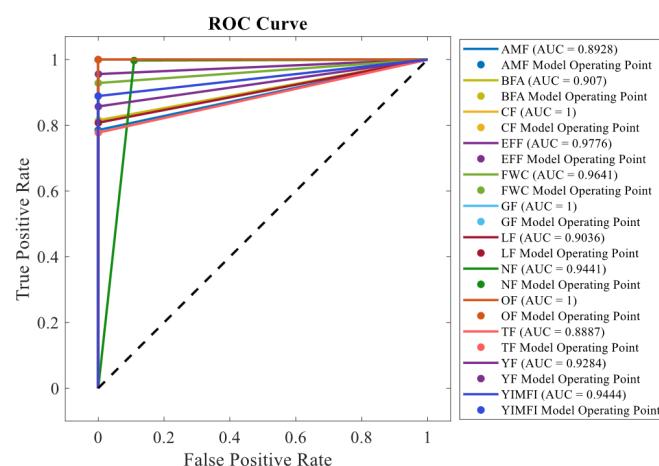


FIGURE 9 The ROC curve and the AUC values of the proposed model for all considered classes. AUC, Area Under Curve; ROC, Receiver Operating Characteristic.

TABLE 7 Comparison of the proposed FDI model with related previous works on this data set.

FDI model	Number of fault classes	Number of normal class samples	Overall accuracy (%)	Average TPR (%)	Average PPV (%)	Average F1-score (%)	The requirement for data balancing
Proposed model	11	34,472	99.27	90.12	91.70	90.77	No
STMNN ¹⁷	5	32,056	94.93	90.64	63.39	66.15	Yes
STFNN ¹⁸	5	32,056	93.84	84.28	63.74	61.98	Yes
GPC ¹⁹	5	32,056	87.90	79.00	74.49	75.83	Yes
SVM ¹⁵	4	48,503	88.67	74.99	40.98	45.02	Yes
RF ¹⁸	5	32,056	84.75	60.95	46.09	41.31	Yes
SVM ¹⁸	5	32,056	72.05	64.19	37.24	38.09	Yes
DT ¹⁸	5	32,056	80.54	62.24	36.57	37.29	Yes
GPC ¹⁸	5	32,056	91.80	79.00	41.86	47.84	Yes
CNN ¹⁸	5	32,056	86.90	62.86	32.94	37.10	Yes
LSTM ¹⁸	5	32,056	87.59	53.61	30.03	34.24	Yes
MKFCNN ¹⁸	5	32,056	91.11	57.42	40.80	42.71	Yes
CNN-LSTM ¹⁸	5	32,056	89.32	60.84	39.94	42.00	Yes
CNN-GRU ¹⁸	5	32,056	84.08	59.15	34.54	37.05	Yes
MSDeepESN ¹⁷	5	32,056	87.77	64.11	37.44	41.65	Yes
LSTM ¹⁷	5	32,056	78.22	61.87	33.03	35.64	Yes
MSResNet ¹⁷	5	32,056	91.49	65.51	50.69	49.68	Yes
CNN ¹⁷	5	32,056	82.04	56.98	28.50	31.18	Yes
MKFCNN ¹⁷	5	32,056	87.13	65.22	39.96	42.40	Yes
CNN-GRU ¹⁷	5	32,056	86.07	65.50	48.31	47.82	Yes
STFNN ¹⁷	5	32,056	93.99	84.62	52.22	57.37	Yes
SVM ¹⁹	5	32,056	81.15	62.07	71.16	62.39	Yes

Abbreviations: CNN, Convolution Neural Network; DT, decision tree; FDI, Fault Detection and Isolation; GPC, Gaussian process classifier; GRU, gated recurrent unit; LSTM, Long Short-term Memory; MKFCNN, MultiKernel Fusion Convolution Neural Network; MSDeepESN, MultiScale Deep Echo State Network; MSResNet, MultiScale Residual Network; RF, random forest; STFNN, spatio-temporal fusion neural network; STMNN, spatio-temporal multiscale neural network; SVM, Support Vector Machine.

of the whole false predictions), average TPR, average PPV, and requirement of the method to data balancing. According to the comparison made in Kadir et al.²¹ and Bao et al.,²² different algorithms including several deep learning methods such as CNN, LSTM, MKFCNN, CNN-LSTM, CNN-GRU, MSDeepESN, and MSResNet neural networks, as well as shallow machine learning techniques such as random forests, SVM, DT, and GPC are adopted for comparison on the present SCADA data set. Note that overall accuracy is not a good validation criterion for imbalance classification, because a case such as “good TPR for no-fault class and bad prediction for other classes” also can result in high overall accuracy.

6 | CONCLUSION

In this paper, the KNN model is used for WT fault detection and isolation. For preprocessing, four steps of data labeling, invalid data elimination, feature selection, and standardization are applied to the raw SCADA data set of a WT, respectively. It was shown that the proposed method is robust against the data imbalance challenge, which prevents the classifier from tending to the majority class. Due to the algorithm's robustness against imbalanced data, there is no need for data balancing methods. Therefore, false alarms are less likely to occur and the classifier's performance is not deteriorated. As a result of the above, the proposed model shows good performance

in WT fault detection and isolation. The proposed FDI system can be used for online monitoring, unlike the training step which is an offline process. This system determines the label of the new data observation only with its current feature values, without any requirement for historical data. Although using temporal correlations for the training and validation of the FDI system can provide useful information, it leads to the problem of the algorithm's dependence on historical data for FDI. Since the existing SCADA data was used to train and validate the algorithm, there is no extra cost to collect and record the data. It is worth mentioning that many faults are considered, especially from the warning data, which are not indicated in the previous related works. Finally, the dependence of the faults is analyzed based on the nature of the faults and the overlap of their occurrence. However, the proposed method performs FDI assuming the correctness of the values recorded by the sensors. Also, the algorithm will not be able to detect and isolate a new type of fault that is not defined in the training data set. In future works, the aforementioned challenges can be investigated, and predictive maintenance may be pursued by developing the existing model to predict WT faults.

ORCID

Javad Poshtan  <http://orcid.org/0000-0001-6243-0901>

REFERENCES

1. Purarjomandlangrudi A, Nourbakhsh G, Esmalifalak M, Tan A. Fault detection in wind turbine: a systematic literature review. *Wind Eng.* 2013;37(5):535-547. doi:[10.1260/0309-524X.37.5.535](https://doi.org/10.1260/0309-524X.37.5.535)
2. Yu D, Chen ZM, Xiahou KS, Li MS, Ji TY, Wu QH. A radically data-driven method for fault detection and diagnosis in wind turbines. *Int J Electr Power Energy Syst.* 2018;99:577-584. doi:[10.1016/j.ijepes.2018.01.009](https://doi.org/10.1016/j.ijepes.2018.01.009)
3. Amirat Y, Benbouzid MEH, Al-Ahmar E, Bensaker B, Turri S. A brief status on condition monitoring and fault diagnosis in wind energy conversion systems. *Renewable Sustainable Energy Rev.* 2009;13(9):2629-2636. doi:[10.1016/j.rser.2009.06.031](https://doi.org/10.1016/j.rser.2009.06.031)
4. Rezamand M, Kordestani M, Cariveau R, Ting DSK, Orchard ME, Saif M. Critical wind turbine components prognostics: a comprehensive review. *IEEE Trans Instrum Meas.* 2020;69(12):9306-9328. doi:[10.1109/TIM.2020.3030165](https://doi.org/10.1109/TIM.2020.3030165)
5. Qiao W, Lu D. A survey on wind turbine condition monitoring and fault diagnosis—part II: signals and signal processing methods. *IEEE Trans Ind Electron.* 2015;62(10):6546-6557. doi:[10.1109/TIE.2015.2422394](https://doi.org/10.1109/TIE.2015.2422394)
6. Jiang G, Xie P, He H, Yan J. Wind turbine fault detection using a denoising autoencoder with temporal information. *IEEE/ASME Trans Mechatron.* 2018;23(1):89-100. doi:[10.1109/TMECH.2017.2759301](https://doi.org/10.1109/TMECH.2017.2759301)
7. Aval SMM, Ahadi A. Wind turbine fault diagnosis techniques and related algorithms. *Int J Renewable Energy Res.* 2016;6(1):80-89.
8. He M, He D. A new hybrid deep signal processing approach for bearing fault diagnosis using vibration signals. *Neurocomputing.* 2020;396:542-555. doi:[10.1016/j.neucom.2018.12.088](https://doi.org/10.1016/j.neucom.2018.12.088)
9. Liu C, Cheng G, Liu B, Chen X. Bearing fault diagnosis method with unknown variable speed based on multi-curve extraction and selection. *Measurement.* 2020;153:107437. doi:[10.1016/j.measurement.2019.107437](https://doi.org/10.1016/j.measurement.2019.107437)
10. Teng W, Ding X, Tang S, Xu J, Shi B, Liu Y. Vibration analysis for fault detection of wind turbine drivetrains—a comprehensive investigation. *Sensors.* 2021;21(5):1686. doi:[10.3390/s21051686](https://doi.org/10.3390/s21051686)
11. Stetco A, Dinmohammadi F, Zhao X, et al. Machine learning methods for wind turbine condition monitoring: a review. *Renewable Energy.* 2019;133:620-635. doi:[10.1016/j.renene.2018.10.047](https://doi.org/10.1016/j.renene.2018.10.047)
12. Zaher A, McArthur SDJ, Infield DG, Patel Y. Online wind turbine fault detection through automated SCADA data analysis. *Wind Energy.* 2009;12(6):574-593. doi:[10.1002/we.319](https://doi.org/10.1002/we.319)
13. Chen L, Xu G, Zhang Q, Zhang X. Learning deep representation of imbalanced SCADA data for fault detection of wind turbines. *Measurement.* 2019;139:370-379. doi:[10.1016/j.measurement.2019.03.029](https://doi.org/10.1016/j.measurement.2019.03.029)
14. Burkov A, *The Hundred-Page Machine Learning Book*; 2019. doi:[10.1080/15228053.2020.1766224](https://doi.org/10.1080/15228053.2020.1766224)
15. Leahy K, Hu RL, Konstantakopoulos IC, Spanos CJ, Agogino AM, O'Sullivan DTJ. Diagnosing and predicting wind turbine faults from SCADA data using support vector machines. *Int J Prognostics Health Manage.* 2018;9(1):1-11.
16. Rout N, Mishra D, Mallick MK. Handling imbalanced data: a survey. In: Reddy M, Viswanath K, KM S, eds. *International Proceedings on Advances in Soft Computing, Intelligent Systems and Applications*. Advances in Intelligent Systems and Computing, vol 628. Springer; 2018:431-443. doi:[10.1007/978-981-10-5272-9_39](https://doi.org/10.1007/978-981-10-5272-9_39)
17. He Q, Pang Y, Jiang G, Xie P. A spatio-temporal multiscale neural network approach for wind turbine fault diagnosis with imbalanced SCADA data. *IEEE Trans Ind Inf.* 2021;17:6875-6884. doi:[10.1109/TII.2020.3041114](https://doi.org/10.1109/TII.2020.3041114)
18. Pang Y, He Q, Jiang G, Xie P. Spatio-temporal fusion neural network for multi-class fault diagnosis of wind turbines based on SCADA data. *Renewable Energy.* 2020;161:510-524. doi:[10.1016/j.renene.2020.06.154](https://doi.org/10.1016/j.renene.2020.06.154)
19. Li Y, Liu S, Shu L. Wind turbine fault diagnosis based on Gaussian process classifiers applied to operational data. *Renewable Energy.* 2019;134:357-366. doi:[10.1016/j.renene.2018.10.088](https://doi.org/10.1016/j.renene.2018.10.088)
20. Pashazadeh V, Salmasi FR, Araabi BN. Data driven sensor and actuator fault detection and isolation in wind turbine using classifier fusion. *Renewable Energy.* 2018;116:99-106. doi:[10.1016/j.renene.2017.03.051](https://doi.org/10.1016/j.renene.2017.03.051)
21. Kadir ME, Akash PS, Sharmin S, Ali AA, Shoyaib M. A proximity weighted evidential k nearest neighbor classifier for imbalanced data. In: Lauw H, Wong RW, Ntoulas A, Lim EP, Ng SK, Pan S, eds. *Advances in Knowledge Discovery and Data Mining. PAKDD 2020*. Lecture Notes in Computer Science(), vol 12085. Springer; 2020:71-83. doi:[10.1007/978-3-030-47436-2_6](https://doi.org/10.1007/978-3-030-47436-2_6)
22. Bao F, Wu Y, Li Z, Li Y, Liu L, Chen G. Effect improved for high-dimensional and unbalanced data anomaly detection model based on KNN-SMOTE-LSTM. *Complexity.* 2020;2020:1-17. doi:[10.1155/2020/9084704](https://doi.org/10.1155/2020/9084704)

23. Haixiang G, Yijing L, Yanan L, Xiao L, Jinling L. BPSO-Adaboost-KNN ensemble learning algorithm for multi-class imbalanced data classification. *Eng Appl Artif Intell.* 2016;49: 176-193. doi:[10.1016/j.engappai.2015.09.011](https://doi.org/10.1016/j.engappai.2015.09.011)
24. Zhang S. Cost-sensitive KNN classification. *Neurocomputing.* 2020;391:234-242. doi:[10.1016/j.neucom.2018.11.101](https://doi.org/10.1016/j.neucom.2018.11.101)
25. Altayef E, Anayi F, Packianather M, Benmahamed Y, Kherif O. Detection and classification of lamination faults in a 15 kVA three-phase transformer core using SVM, KNN and DT algorithms. *IEEE Access.* 2022;10:50925-50932. doi:[10.1109/ACCESS.2022.3174359](https://doi.org/10.1109/ACCESS.2022.3174359)
26. Swetapadma A, Yadav A. A novel single-ended fault location scheme for parallel transmission lines using k-nearest neighbor algorithm. *Comput Electr Eng.* 2018;69:41-53. doi:[10.1016/j.compeleceng.2018.05.024](https://doi.org/10.1016/j.compeleceng.2018.05.024)
27. Shi Z, Liu Z, Lee J. An auto-associative residual based approach for railway point system fault detection and diagnosis. *Measurement.* 2018;119:246-258. doi:[10.1016/j.measurement.2018.01.062](https://doi.org/10.1016/j.measurement.2018.01.062)
28. Naimi A, Deng J, Shimjith SR, Arul AJ. Fault detection and isolation of a pressurized water reactor based on neural network and K-nearest neighbor. *IEEE Access.* 2022;10: 17113-17121. doi:[10.1109/ACCESS.2022.3149772](https://doi.org/10.1109/ACCESS.2022.3149772)
29. Meyer A. Multi-target normal behaviour models for wind farm condition monitoring. *Appl Energy.* 2021;300:117342. doi:[10.1016/j.apenergy.2021.117342](https://doi.org/10.1016/j.apenergy.2021.117342)
30. Freeman B, Tang Y, Huang Y, VanZwieten J. Rotor blade imbalance fault detection for variable-speed marine current turbines via generator power signal analysis. *Ocean Eng.* 2021;223:108666. doi:[10.1016/j.oceaneng.2021.108666](https://doi.org/10.1016/j.oceaneng.2021.108666)
31. Dhiman H, Deb D, Muyeen SM, Kamwa I. Wind turbine gearbox anomaly detection based on adaptive threshold and twin support vector machines. *IEEE Trans Energy Convers.* 2021;36(4):3462-3469. doi:[10.1109/TEC.2021.3075897](https://doi.org/10.1109/TEC.2021.3075897)
32. Hu RL, Leahy K, Konstantakopoulos IC, Auslander DM, Spanos CJ, Agogino AM. Using domain knowledge features for wind turbine diagnostics. In: *Proceedings of the 15th IEEE International Conference on Machine Learning and Applications, Anaheim, CA, USA, 18–20 December.* 2016:300-305.
33. Kim K. Normalized class coherence change-based kNN for classification of imbalanced data. *Pattern Recogn.* 2021;120: 108126. doi:[10.1016/j.patcog.2021.108126](https://doi.org/10.1016/j.patcog.2021.108126)

How to cite this article: Fazli A, Poshtan J. Wind turbine fault detection and isolation robust against data imbalance using KNN. *Energy Sci Eng.* 2024;12:1174-1186. doi:[10.1002/ese3.1706](https://doi.org/10.1002/ese3.1706)