Applied Soft Computing 57 (2017) 427–435

Contents lists available at ScienceDirect

Applied Soft Computing

journal homepage: www.elsevier.com/locate/asoc

Full Length Article

Classification of ball bearing faults using a hybrid intelligent model

Manjeevan Seera[a,*], M.L. Dennis Wong[a,b], Asoke K. Nandi[c,d]

[a] Faculty of Engineering, Computing and Science, Swinburne University of Technology (Sarawak Campus), Sarawak, Malaysia
[b] Heriot-Watt University Malaysia, Putrajaya, Malaysia
[c] Department of Electronic and Computer Engineering, Brunel University London, Uxbridge UB8 3PH, United Kingdom
[d] The Key Laboratory of Embedded Systems and Service Computing, College of Electronic and Information Engineering, Tongji University, Shanghai, China

ARTICLE INFO

Article history:
Received 28 March 2016
Received in revised form 28 January 2017
Accepted 18 April 2017
Available online 21 April 2017

Keywords:
Condition monitoring
Ball bearing
Electrical motor
Fuzzy min-max neural network
Random forest

ABSTRACT

In this paper, classification of ball bearing faults using vibration signals is presented. A review of condition monitoring using vibration signals with various intelligent systems is first presented. A hybrid intelligent model, FMM-RF, consisting of the Fuzzy Min-Max (FMM) neural network and the Random Forest (RF) model, is proposed. A benchmark problem is tested to evaluate the practicality of the FMM-RF model. The proposed model is then applied to a real-world dataset. In both cases, power spectrum and sample entropy features are used for classification. Results from both experiments show good accuracy achieved by the proposed FMM-RF model. In addition, a set of explanatory rules in the form of a decision tree is extracted to justify the predictions. The outcomes indicate the usefulness of FMM-RF in performing classification of ball bearing faults.

1. Introduction

Condition monitoring of machinery is becoming increasingly important in modern maintenance. There is a need to reduce unscheduled downtime in order to maintain corporate competitiveness. The cost of maintenance can also be reduced by constantly monitoring the health of the machines [1]. In this way, disastrous faults that could potentially happen can be detected early, which will reduce the total downtime of the machine and entire operations. Predictive maintenance techniques have been effectively utilized in reducing unexpected machine failures [2]. One of the most commonly used predictive maintenance technology is vibration monitoring, due to the amount of machine conditions information that is provided [2].

In the past two decades, a number of researchers have reported their achievements on condition monitoring of rotating machinery. Condition monitoring in the rotating machines of the industry uses accelerometers and vibration transmitters in order to acquire data [3–5]. Once data is acquired, it is then vital to process these data. Pattern recognition is the central task in the machine condition monitoring, with various solutions reported [6–9]. It first looks at information from multitude of sources, such as transducer signals from the machine [9]. Feature extraction is then used to extract useful features from the collected information.

Selecting the right features is the key to solve accurately the classification problem, and the choice of features can greatly affect the classification performance [1]. In general, time-domain features are commonly used in machine condition monitoring. The features commonly used, but not limited to are the root mean squared (RMS) voltage, the peak voltage, the X-Y plot, and the crest factor (the ratio of the peak voltage over the RMS voltage). For more advanced methods, the vibration data is often transformed to its frequency domain equivalent, which is the Power Spectrum or FFT. With the increased computing power and digital storage in recent years, the use of waterfall diagram and discrete wavelet transform has increased.

The contributions of this paper are two-fold: the use of a hybrid intelligent model for detection and classification of real-world roller ball bearing faults as well as detailed investigations in the use of a set of power spectrum and sample entropy-based features for performing this task. For validation purposes, a well-known benchmark database is first used in the experimental works. Then, a real-world data set with new features extracted using entropy is used to further validate the data. It is worth mentioning that the hybrid intelligent model delivers a simple yet useful tree in classifying the outputs from the data.

This paper is organized as follows. A literature review on condition monitoring using vibration signals with various intelligent systems is first presented in Section 2. Details of the power spectrum and sample entropy feature extractions are presented in

* Corresponding author.
E-mail address: mseera@gmail.com (M. Seera).

http://dx.doi.org/10.1016/j.asoc.2017.04.034

Section 3. The hybrid FMM-RF model is detailed in Section 4. A benchmark study to evaluate FMM-RF effectiveness is detailed in Section 5. Applicability of FMM-RF is real-world data set is shown in Section 6. Concluding remarks are finally offered in Section 7.

2. Literature review

A literature review for condition monitoring using vibration signals with various intelligent systems is presented in this section. The empirical mode decomposition (EMD) energy entropy is used to extract features from vibration signals in [10]. Features are then selected using the intrinsic mode functions (IMFs) method, which are then fed to an artificial neural network (ANN) with back propagation (BP) to classify bearings defects [10]. Results indicate the proposed method can accurately determine bearing defects using run-to-failure vibration signals [10]. Hilbert Transform and Fast Fourier Transform are two feature extraction methods in [11] for vibration signals. The ANN-based fault estimation algorithm with the use of genetic algorithm (GA) is used for fault diagnosis of rolling bearings [11]. Improved classification results are seen from the experiments [11].

Seven decomposed signals with varying frequency range for kurtosis of bearing vibration signal is presented in [12]. A hybrid empirical mode decomposition and relevance vector machine (RVM) with artificial bee colony algorithm model is used to predict the signals [12]. Results indicate the hybrid model in [12] improves the accuracy rates as compared to RVM in kurtosis of bearing vibration signal. Remaining useful life (RUL) of bearings is of interest in [13] and [14]. A simplified fuzzy adaptive resonance theory map (SFAM) neural network is used to predict the RUL of rolling element bearings [13]. The Weibull distribution is used to fit measurements, based on the seven defined classes [13]. Experimental results in [13] indicate the reliability of the RUL prediction. A generalised function from Weibull function is used to fit measurements in [14], similar to [13]. An ANN is used for classification, with a validation mechanism used to improve ANN performance [14]. Using real-world vibration data from pump bearings, good results are achieved [14].

A total of seven bearing states from the EMD are produced in [15]. For feature reduction, the principal component analysis and linear discriminant analysis are used [15]. Classification is then carried out using the probabilistic neural network and SFAM [15]. Results show better generalization capability as compared with other methods [15]. Vibration signals from bearing are pre-processed using the de-trended fluctuation analysis and rescaled-range analysis techniques in [16]. Signals are acquired from different frequency and load conditions [16]. Using principal component analysis and ANN, the classification yielded fairly good results [16]. A hard competitive growing neural network (HC-GNN) with shrinkage learning is used in [17] for fault detection and diagnosis of small bearing faults. Wavelet transform is used in feature extraction [17]. The HC-GNN creates smaller networks compared to other networks [17]. Results on a machinery system with various small bearing faults indicate good results from the proposed network [17].

Prediction of faulty bearing conditions using the interval type-2 fuzzy neural network is detailed in [18]. A total of three different features are extracted [18]. The faulty bearings are used for validation, with results compared with those from adaptive neuro-fuzzy inference system (ANFIS) [18]. The proposed method yields better prediction accuracy as compared to ANFIS [18]. Frequency-domain features of bearing vibration signals are extracted in [19]. In identifying fault types, a sequential diagnosis technique through the partially-linearized neural network (PLNN) is done [19]. The PLNN can automatically determine fault types in rolling bearing with good accuracy rates [19]. Time-domain data is extracted from vibration signals in a rotor-bearing system in [20]. The measurements are done at five different rotating speeds [20]. For classification, a support vector machine is utilized with good classification accuracy achieved [20].

The dependent feature vector is first used in [21] for fault diagnosis of rolling element bearings. The features are then fed to a probability neural network [21]. Experimental results show the proposed method achieves an efficient accuracy in analysis the bearing faults [21]. Vibration signals from a rotor-bearing system are analyzed in [22]. The key kernels (KK) and particle swarm optimization (PSO), known as KK-PSO method is proposed for Volterra series identification in feature extraction [22]. Using simulation and real-data, results show the KK-PSO method outperforms the least square and traditional PSO method [22]. In a shaft-bearing mechanism, both vibration and current signal data are acquired [23]. The time-domain and frequency-domain parameters are extracted from both signals [23]. A multi-stage algorithm based on ANN and ANFIS model is used for classification, with results showing the proposed method is effective [23].

Various vibration conditions are extracted from drilling machines in [24]. The radial basis neural network is then used to analyze the acquired signals [24]. Compared to radial basis networks, the proposed network has better performance in adapting to real-time parameters of the drilling machines [24]. In condition diagnosis of various bearing systems, vibration signals are used as inputs in [25]. Ten statistical features are extracted from the signals. A hybrid technique combining GA with adaptive operator probabilities (AGAs) and back propagation neural networks (BPNNs), named AGAs-BPNNs is proposed [25]. Results from the experiment show the proposed AGAs-BPNNs method acquired higher classification accuracy [25].

3. Feature extraction

In the general pattern classification framework, the feature extraction is a key stage for extracting the salient information from the raw signals and reducing the dimensionality of the input vector to the classification engine. The feature extraction method chosen is dependent on the specific task. In this section, we present two different types of feature extraction method for classifying ball bearing faults: (1) the conventional power spectrum and (2) the sample entropy [26]. The former is a commonly used feature for classifying ball bearing faults and the latter was recently introduced in [27].

3.1. Power spectrum (PS)

The existence of defects in ball bearings will exhibit as high frequency spikes and other fault patterns in the vibration time series. In the frequency domain, this translates to the addition of new harmonics in the power spectrum (PS). As such, PS is often chosen for condition monitoring problems as it compactly represents the time varying time domain signal into a set of fixed length vector representing the square magnitude of the frequency components (harmonics). There are many methods for estimating the signal's power spectrum. In this work, we adopted the commonly used Welch's method to perform this task. The Welch's method is essentially a non-parametric method which compute the PS through an averaging process.

Given the time series data $X_n = \left\{x_0, x_1, x_2, x_3 \cdots, x_{N-1}\right\}$, the Welch's PS estimate can be written as:

$$\mathrm{P}_{X_i}\left(e^{j\omega}\right) = \frac{1}{KLU}\sum_{i=0}^{K-1}\left|\sum_{n=0}^{L-1} W_n x_{n+iD} e^{-jn\omega}\right|^2 \qquad (1)$$

where W_i is the windows function of Length L-1; K is the number of segments the X_n is divided into with D points overlapping between two consecutive segments; and U is a normalizing factor defined as $\frac{1}{L}\sum_{n=0}^{L-1}|w_n|^2$.

In this work, we have chosen to use Hanning Window with a window length of 1024 samples and the overlapping factor is set to 50% of the window length. Following the computation of the PS, we extract the PS values from DC to 12 KHz with 500 Hz intervals, which resulted in a vector of 25 PS features.

3.2. Sample entropy (SampEn)

Shannon's entropy is a measure of information content and more specifically it measures the level of unpredictability of a given sampled time series. As faults in the ball bearing will introduce new patterns into the vibration signals, e.g., spikes at different intervals, and broaden envelope the information content of the vibration signals will change. Therefore, it is intuitive for one to capture these changes through computing the signals' entropy values. However, it is difficult to estimate Shannon's entropy directly for a time series signal.

The approximate entropy (ApEn) is proposed in [28] for noisy and short time series, in order to estimate the rate of generating new information. In reducing the bias produced by pattern self-matching in ApEn, the Sample Entropy (SampEn) is proposed in [26] for a sampled time series data from a continuous process. This provides an accurate negative logarithm intended for ApEn. We briefly present the computation of SampEn as follows.

Given a time-series data $X_n = \{x_0, x_1, x_2, x_3 \cdots, x_{N-1}\}$ of length N as above with a sampling interval, τ, let $m << N$ be a constant, then X_n can be divided into (N–m+1) template vectors $X'_{n+t} = \{x_{n+0}, x_{n+1}, \cdots x_{n+m-1}\}$, each of length m and for all $n = 0, \ldots,$ $(N–m)$. Let $\Delta\left[X'_i, X'_j\right]$ be the Chebyshev distance function and if for any two template vectors, where $\Delta\left[X'_i, X'_j\right] < r$ then a match is recorded. The tolerance parameter, r, by convention, is set to a fraction of the standard deviation of the sequence for convenience. In general, the setting is usually a fifth of the standard deviation of the given time series.

Furthermore, let A_m denotes the number of matches of length m, and B_{m-1} denotes the number of matches of length m except at the end of the sequence. The sample entropy can be computed as [29]:

$$\mathrm{SampEn}(X_n) = -\ln\left\{\frac{A_m}{B_{m-1}}\right\} : A_m \neq 0 \text{ and } B_{m-1} \neq 0 \tag{2}$$

In the case that either A_m or B_{m-1} is zero, then,

$$\mathrm{SampEn}(X_n) = -\ln\left\{\frac{N-m}{N-m-1}\right\} : A_m = 0 \text{ or } B_{m-1} = 0 \tag{3}$$

for $m = 0$, B_{m-1} is set to $\frac{N(N-1)}{2}$.

In this paper, SampEn with $m = 0, 1, 2$ (labelled as m_0, m_1, m_2) for each of the acquired vibration signal are extracted and r was set to the recommended one-fifth of the standard deviation. Therefore, for each of the vibration signals, we have three SampEn features.

4. Hybrid intelligent model

The details of the hybrid intelligent model, FMM-RF, are outlined in the following subsections. Details of the Classification and Regression Tree (CART) are given, being a part of RF. In addition, the modifications of FMM and CART are given in the respective subsections. The procedure of the hybrid model is given as follows, in Fig. 1.

4.1. Fuzzy Min-Max

FMM uses hyperbox fuzzy sets for learning. To regulate a hyperbox size, the expansion parameter (user-defined) of $\theta \in [0,1]$ is used. The min (minimum) and max (maximum) points in a hyperbox are used in measuring how an input pattern fits in the hyperbox from a fuzzy membership function. A hyperbox fuzzy set (B_j) with V_j being the min point, W_j being the max points, and I^n being a unit hypercube is defined as follow [30]:

$$B_j = \left\{X, V_j, W_j, f\left(X, V_j, W_j\right)\right\} \forall X \in I^n \tag{4}$$

The joint fuzzy set that categorises the output class k^{th} is:

$$C_k = \bigcup_{j \in K} B_j \tag{5}$$

where hyperboxes belong to class k is denoted by K.

The learning algorithm in FMM constructs non-linear boundaries for each output class. As such, overlapping between hyperboxes is only allowed for the same class. A membership function can be computed using [30]:

$$b_j(A_h) = \frac{1}{2n}\sum_{i=1}^{n}\left[\max(0, 1 - \max\left(0, \gamma \min\left(1, a_{hi} - w_{ji}\right)\right)\right.$$
$$\left. + \max(0, 1 - \max(0, \gamma \min\left(1, v_{ji} - a_{hi}\right)\right] \tag{6}$$

where γ being the sensitivity parameter regulated the speed of membership function and A_h=(a_{h1}, a_{h2}, ., a_{hn}) is the h^{th} input pattern.

There are three node layers in FMM, consisting of the input (F_A), hidden (F_B), and output (F_C) layers. F_A corresponds to number of input dimension, F_B being the hyperbox layer, and F_C corresponding to the number of output classes. Every hyperbox set is marked with one F_B node while min to max points are contained within the connections of F_A to F_B. Connection between the nodes of F_B and F_C is:

$$u_{jk} = \begin{cases} 1 & \text{if } b_j \text{ is a hyperbox for class } C_k \\ 0 & \text{otherwise} \end{cases} \tag{7}$$

where C_k being k^{th} target class in F_C while b_j being j^{th} hidden node in F_B. A fuzzy union is done in every F_C node:

$$c_k = \max_{j=1} b_j u_{jk} \tag{8}$$

The F_C nodes can be used in two ways. The first one is the outputs used directly, which produces a soft decision, or the second one called winner-take-all where it uses a hard decision.

To integrate FMM with CART and RF, a hyperbox B_j is first tagged with CF_j, a confidence factor, which is calculated as:

$$CF_j = (1 - n)U_j + nA_j \tag{9}$$

where $n \in [0,1]$ being the weighting factor, U_j being usage of hyperbox, and A_j being accuracy of hyperbox.

The confidence factor can identify the hyperboxes that are used regularly and fairly accurate, and also those not being used regularly but highly accurate. In addition, the centroids of hyperboxes are calculated as follows, as the original FMM only contains the min and max points:

$$C_{ji}^{new} = C_{ji} + \frac{|a_{hi} - C_{ji}|}{N_j} \tag{10}$$

where C_{ji} being the centroid of hyperbox, N_j being number of contained data in hyperbox, and a_{hi} is the h-th input data.

1. Provide the input data samples
2. Initiate the FMM learning procedure
   a. Initialize the hyperbox minimum and maximum points
   b. Identify the closest hyperbox to the input pattern for expansion, if not, add a new hyperbox
   c. Check whether the expansion causes any overlaps among hyperboxes
   d. If overlaps exist, contract the hyperboxes to eliminate the overlapped regions
   e. Calculate the centroid point of the hyperbox
3. Use the generated hyperboxes as the input samples to CART
   a. Split the nodes recursively, with each node assigned to a predicted class
   b. Use the centroid point and the confidence factor for tree building
   c. Conduct bagging for RF, with the capability of random attribute selection
   d. Prune the ensemble to find a good subset of members
4. Use the majority voting scheme to combine the predictions from an ensemble of trees

Fig. 1. Procedure of the proposed FMM-RF model.

4.2. Classification and regression tree

In building a decision tree, a training data set, which consists of input data with its respective classes is needed. The data for training consists of centroids of the FMM hyperboxes (as in Eq. (6)), are partitioned into a number of smaller groups. Based on e input samples, the process of building the tree starts at the root node in which all data samples are taken into account. Splitting of tree happens when the data samples are not pure, it happens when they are not from the same class. When this happens, two leaf nodes are generated from the most notable feature from samples of data. This same tree-splitting technique is used till a full decision tree is generated.

In principle, the Gini impurity index is used to determine when tree splitting should occur, starting with the measurement of degree of impurity from samples of data, G [31]:

$$Gini(G) = 1 - \sum_i g^2(i) \tag{11}$$

where $g(i)$, where $i = 1, \ldots, e$, is the fraction (probability of instance) of the i-th input sample at node to split, in regards to all m input samples.

In measuring the goodness-of-split, p, the impurity function of every leaf node is utilized. In an ideal case, every leaf node contains data samples only from a single class. Tree-splitting stops when this occurs; else, the goodness-of-split at the spitting node (indicated as node l) subject to the i-th input sample is calculated [32]:

$$\Delta i(p, l) = i(l) - d_L\,[i(l_L)] - d_R\,[i(l_R)] \tag{12}$$

where d_L and d_R shows the data sample fraction at node l that moves to the left (d_L) and right (d_R) child nodes while $i(d_L)$ and $i(d_R)$ show the impurity measures of the left and right child nodes [32].

During tree building, it is plausible for a sample of data in taking an incorrect branch in CART. In tackling this issue, the centroid from each prototype node in FAM is given a weight, also known as confidence factor, which is computed using Eq. (9). Using this weight information, Eq. (13) replaces Eq. (11):

$$Gini(G) = 1 - \sum_i v^2(i) \tag{13}$$

where $v(i)$ is the weight of the i-th input sample at node l, $i = 1, \ldots, e$. The significance of every prototype node is shown by the confidence factor, or weight in the proposed equation.

4.3. Random forest

The random forest (RF) structure is displayed in Fig. 2. Classes are listed as k and number of trees as T [33]. The construction of RF is based on the bagging method, using random attribute selection. Using a data set (D) with tuples (t) and CART trees (k) in the ensemble, in every iteration D_i is formed using d tuples from sample replacement method [31]. The CART is then applied in growing the RF tree until it reaches its maximal size. Pruning is then done to locate a robust subset of ensemble members.

Pruning shrinks the tree by either turning branch nodes to leaf nodes or removing leaf nodes under the original branch. The cost-complexity pruning algorithm [31] is utilized, where it starts from bottom of the tree and cost-complexity at an internal node is then counted. If the sub-tree results in a smaller cost complexity, it is pruned; else it remains [31]. The majority voting method is then used in combining predictions from the ensemble, as shown in Fig. 2.

5. Experiments: benchmark

In the benchmark experiment, the test setup is made up of a 3-phase motor, a torque encoder/transducer, and a dynamometer. Different load levels were measured with the dynamometer. In acquiring the vibration signals from the motor bearings manufactured by SKF, an accelerometer was fitted on top of drive-end of motor. The vibration samples were then sampled at 12 kHz and saved using a 16-channel digital audio tape recorder. Faults in single points with diameters of 7, 14, 21, and 28 mils were inserted using electro-discharge machining. Operating conditions of normal (N), outer ring (OR) race fault, inner ring (IR) race fault, and ball fault (BF) were created at four load levels from 0 to 3 Hp.

In addition to FMM-RF, four other models, i.e. FMM, CART, RF, and FMM-CART [34] were used for comparison purposes. FMM, CART, and RF are standalone models, with their details given in Sections 4.1, 4.2, and 4.3, respectively. FMM-CART is a combination of FMM and CART, with the use of centroids and confidence factor in FMM and a modified Gini impurity index in CART. To compare the results with [35], the 5-fold cross validation method is used. A total of 10 test runs were conducted in total, with the results computed using the bootstrap method. The averages and standard deviations (StdDev) were computed with a resampling rate of 5000 for a reliable performance [36]. The experiments were run using MATLAB® R2014a on an Intel Core i5 2.60 GHz processor with 8GB of RAM.

The benchmark experiments were split into three, using Sample Entropy (SampEn) features, Power Spectrum (PS) features, and

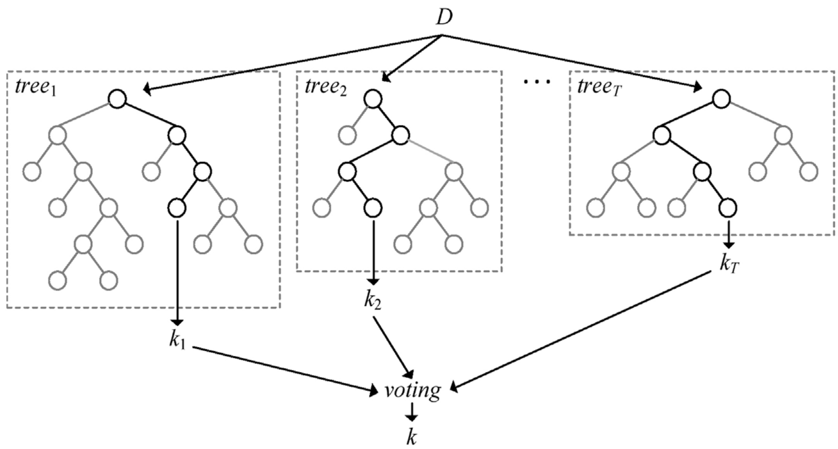

Fig. 2. The random forest structure (adopted from [33]).

Table 1
Results of benchmark experiments.

Model	Features	Accuracy (%)	StdDev	Complexity
FMM	SampEn	98.29	1.15	41 hyperboxes
	PS	94.48	3.28	169 hyperboxes
	SampEn+PS	94.65	2.47	173 hyperboxes
CART	SampEn	82.51	6.53	15 leaf nodes
	PS	88.30	3.07	12 leaf nodes
	SampEn+PS	89.21	2.25	13 leaf nodes
RF	SampEn	96.95	0.03	16 leaf nodes
	PS	97.71	0.20	13 leaf nodes
	SampEn+PS	97.98	0.32	14 leaf nodes
FMM-CART	SampEn	93.20	1.42	10 leaf nodes
	PS	95.77	0.56	8 leaf nodes
	SampEn+PS	95.82	0.51	8 leaf nodes
FMM-RF	SampEn	99.84	0.02	9 leaf nodes
	PS	99.83	0.87	7 leaf nodes
	SampEn+PS	99.89	0.53	8 leaf nodes

the combination of both sets of features. The results are shown in Table 1. It can be seen that FMM-RF acquired the highest accuracy rate at 99.89% using the combined SampEn and PS features, while CART acquired the lowest accuracy rate using SampEn features alone. FMM-RF had the least complex network while FMM had the most complex network with 173 hyperboxes. The standard deviation of FMM-RF was the lowest, at 0.02.

One of the main advantages of the hybrid intelligent model is the ability to explain its predictions using a decision tree. The decision tree is helpful for its interpretability, whereby knowledge learned can be revealed and represented in terms of a rule set to users. With reference to the decision tree for CWRU data in Fig. 3, the most important feature from FMM-RF is "f_{13}".

When the value is <0.10, the input is categorized as OR, else it the tree splits at "f_1". When the value of "f_1" is < 0.08, the input is categorized as NO, else the tree splits again. When the value of "f_{20}" is ≥ 0.62, the input is categorized as IR, else the tree takes a split at "f_9", where the tree splits into two branches. When the value is <0.36, it splits to "f_{20}", where if the value is ≥0.20, the input is categorized as IR, else it is BF. On the other hand, when the value is ≥0.36, the "m_0" is checked, where if the value is ≥0.12, the input is categorized as BF. The tree then makes it final split at "f_6", where if the value is ≥0.34, it is categorized as IR, else it is BF.

As shown in Table 2, comparison of results with [35] is shown. A total of 3 models were used, consisting of multilayer perceptron (MLP), FMM, and CART. Results in [35] were computed using a 5-fold cross validation method. The features used in [35] are different

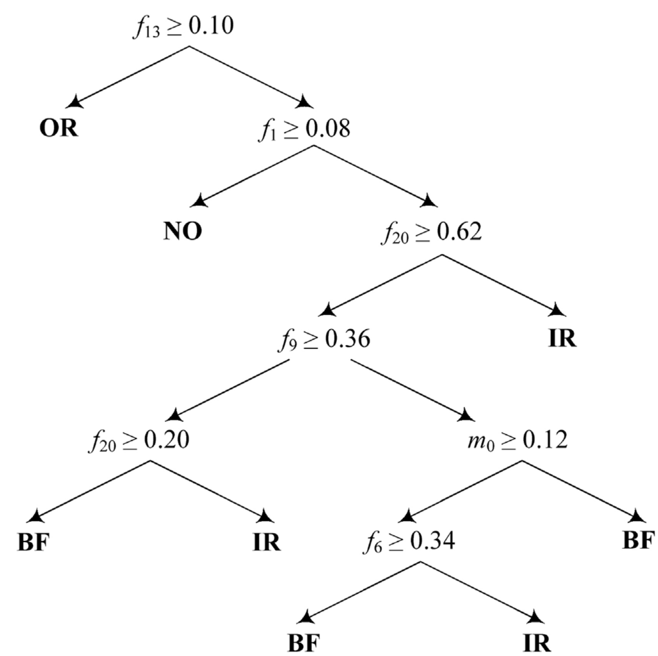

Fig. 3. Decision tree for CWRU data set using the combination of SampEn and PS features.

Table 2
Results comparison of FMM-RF with [35].

Model	Accuracy (%)	StdDev	Complexity
MLP [35]	85.23	6.86	50 hidden nodes
FMM [35]	96.37	2.87	25 hyperboxes
CART [35]	99.02	0.98	5 leaf nodes
FMM-RF	99.89	0.53	8 leaf nodes

from those in this paper, where they consisted of nine time-domain features and seven-frequency domain features. While the results of FMM-RF acquired the highest accuracy rate with the smallest standard deviation, CART [35] on the other hand had the simplest network with five leaf nodes.

6. Experiments: real-world

Real data was acquired from a small test rig [5,37,38], as depicted in Fig. 4 that emulates a running roller bearings environment. The test rig consists of a DC motor which drives the shaft through a flexible coupling. Two plummer bearing blocks then support the shaft.

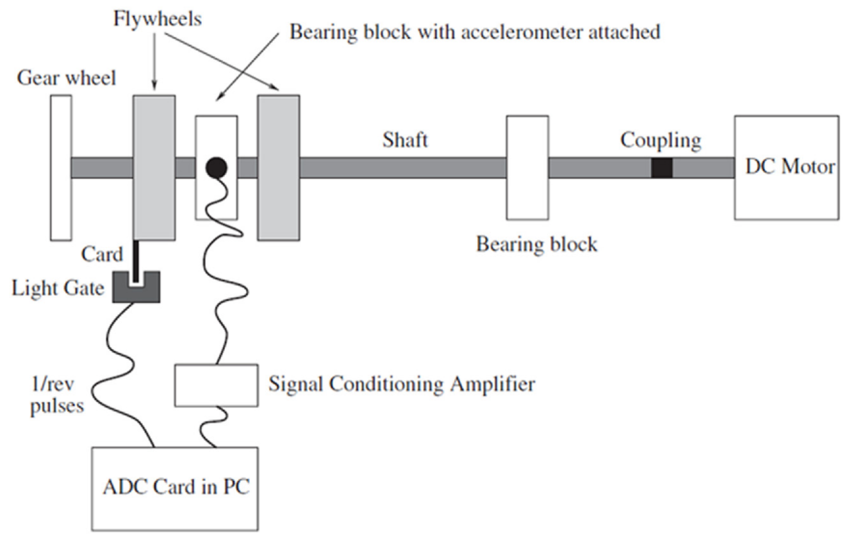

Fig. 4. The setup of the data acquisition test rig (reproduced from [5]).

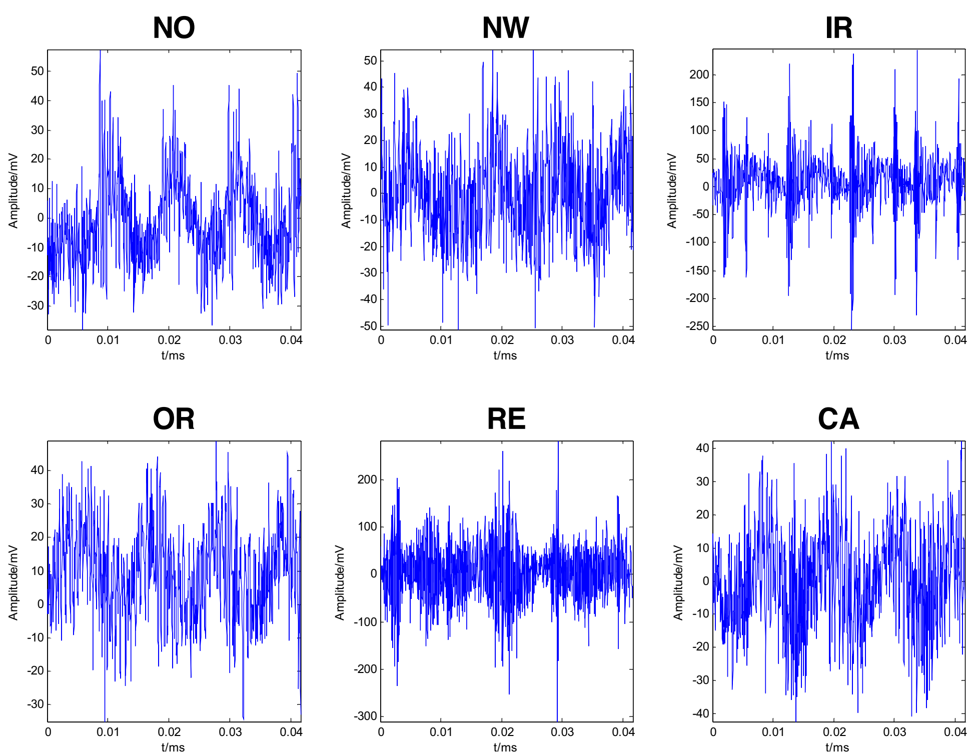

Fig. 5. Six sample vibration signals with respective fault types.

Six conditions were tested and recorded. Two normal conditions; a brand new condition (NO) and a worn but undamaged condition (NW); four fault conditions; outer race (OR), cage (CA), inner race (IR), and rolling element (RE) faults. The machine was operated at a range of speeds, from 25 to 75 rev/s, and ten time-series were taken at each speed. This resulted in 960 samples, with 160 example time-series each from the conditions. For this work, the data was acquired at sixteen different speeds which add non-linearity onto this problem.

Fig. 5 depicts sample vibration signals for the six different fault types. Depending on the fault types, the defect in the bearing modulates the vibration signals and some with distinctive spikes. Two fault conditions, inner and outer race have reasonably periodic signal as compared to the rolling element which may or may not be periodic. This depends on a number of factors which include the severity of damage to rolling element, bearing loading, and ball track within the raceway. The cage fault creates a random distortion, again depending on severity of damage and bearing loading. The feature space for these three features is shown in Fig. 6.

Similar to the benchmark experiments in Section 5, the FMM, CART, RF, and FMM-CART [34] were used for comparison purposes. To compare the results with [27], the 10-fold cross validation method is used. A total of 10 test runs were conducted in total, with the results calculated with the bootstrap method. The results of the 2-class problem are shown in Table 3. FMM-RF acquired the highest accuracy rate at 99.82% using SampEn +PS features while CART using PS only features acquired the lowest accuracy rate. FMM-RF at 5 leaf nodes had the least complex network while FMM had the

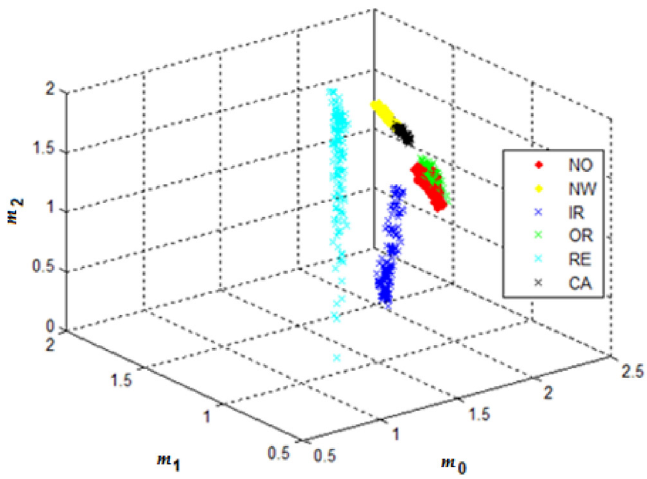

Fig. 6. Feature space for the entropic features of m_0, m_1, and m_2.

Table 4
Results of 6-class problem.

Model	Features	Accuracy (%)	StdDev	Complexity
FMM	SampEn	94.34	1.56	82 hyperboxes
	PS	95.17	2.01	75 hyperboxes
	SampEn + PS	95.18	1.21	80 hyperboxes
CART	SampEn	89.69	7.79	26 leaf nodes
	PS	89.18	5.36	19 leaf nodes
	SampEn+PS	90.21	6.22	21 leaf nodes
RF	SampEn	97.97	0.09	29 leaf nodes
	PS	96.43	0.04	21 leaf nodes
	SampEn+PS	97.75	0.03	25 leaf nodes
FMM-CART	SampEn	96.16	2.88	11 leaf nodes
	PS	95.75	2.71	8 leaf nodes
	SampEn+PS	96.02	2.01	10 leaf nodes
FMM-RF	SampEn	99.74	0.02	10 leaf nodes
	PS	99.72	0.50	6 leaf nodes
	SampEn + PS	99.81	0.41	8 leaf nodes

Table 3
Results of 2-class problem.

Model	Features	Accuracy (%)	StdDev	Complexity
FMM	SampEn	93.46	1.32	69 hyperboxes
	PS	94.23	4.02	165 hyperboxes
	SampEn+PS	94.65	3.21	171 hyperboxes
CART	SampEn	84.81	7.53	20 leaf nodes
	PS	83.96	10.51	17 leaf nodes
	SampEn+PS	84.82	6.21	18 leaf nodes
RF	SampEn	97.58	0.08	17 leaf nodes
	PS	97.31	0.31	15 leaf nodes
	SampEn+PS	97.69	0.25	13 leaf nodes
FMM-CART	SampEn	95.83	1.78	6 leaf nodes
	PS	95.75	3.87	5 leaf nodes
	SampEn+PS	95.97	1.28	6 leaf nodes
FMM-RF	SampEn	99.80	0.02	5 leaf nodes
	PS	99.82	1.78	6 leaf nodes
	SampEn+PS	99.82	0.35	7 leaf nodes

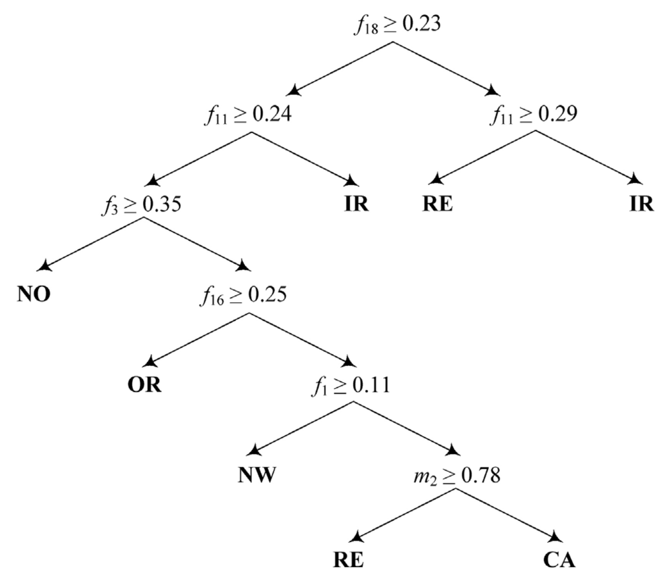

Fig. 8. Decision tree for 6-class problem using entropic features using SampEn and PS features.

most complex network with 171 hyperboxes. The standard deviation of FMM-RF was the lowest, at 0.02.

With reference to the decision tree for 2-class problem in Fig. 7, the most important feature from FMM-RF is "f_{21}". The tree splits into two main parts. When the value is $\geq$0.32, "f_{13}" is checked, where if the value is <0.62, the input is categorized as healthy, else the tree splits again to "m_0". When the value is $\geq$0.92, the input is categorized as faulty, else it is healthy. When the value of "f_{21}" is <0.32, the tree splits to "f_{18}", where if the value is $\geq$ 0.18, "f_{19}" is checked. When the value is $\geq$0.11, the input is categorized as faulty, else it is healthy. When the value is <0.18, "f_{22}" is checked, where if the value is $\geq$0.09, the input is categorized as faulty, else it is healthy.

In addition to 2-class problem, the 6-class problem is conducted, with results shown in Table 4. The same setup was used for the 2-class problem is used in this experiment. The results are similar to those of the 2-class problem, with FMM-RF acquiring the highest accuracy rate and CART the lowest. Again, FMM had the most complex network with 82 hyperboxes while FMM-RF had 6–10 leaf nodes, with FMM-CART coming in second with maximum of 11 leaf nodes. FMM-RF had the lowest standard deviation at 0.02.

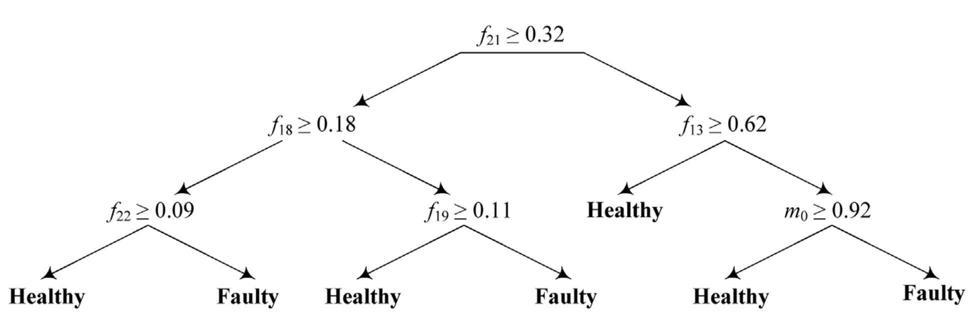

Fig. 7. Decision tree for 2-class problem using SampEn and PS features.

Table 5
Results comparison of FMM-RF with [27].

Model	Accuracy (%)	StdDev	Complexity
SVM [27]	93.50	0.50	20 hidden nodes
MLP [27]	93.70	0.21	–
FMM-RF	99.81	0.41	8 leaf nodes

With reference to the decision tree for 6-class problem in Fig. 8, the most important feature from FMM-RF is "f_{18}". The tree splits into two main branches, one on the left another on the right. When the value of "f_{18}" is $\geq$0.23, the tree splits to "f_{11}", where if the value is $\geq$0.29, the input is categorized as IR, else it is RE. When the value of "f_{18}" is <0.23, the tree splits to "f_{11}". When the value of "f_{11}" is $\geq$0.24, the input is categorized as IR, else it splits to "f_3". When "f_3" is <0.35, the input is categorized as NO, else it splits to "f_{16}". When the value of "f_{16}" is <0.25, the input is categorized as OR, else it splits again. When the value of "f_1" is <0.11, the input is categorized as NW, else the tree takes the final split. When the value of "m_2" is $\geq$0.78, the input is categorized as CA, else it is RE.

The comparisons of results are done with those from [27], as shown in Table 5. Two different models are used in [27], consisting of support vector machine (SVM) and MLP. A linear SVM classifies linearly separable input data by using a hyperplane determined through training with a set of labelled training data. SVM, as a member of the kernel machine family, can be generalised to non-linearly separable data through the use of the kernel trick. In a nutshell, the non-linearly separable data is projected to a linearly separable space through a chosen kernel before applying the usual SVM classification procedures.

On the other hand, the MLP is a classical feedforward neural network where the neurons are arranged in a two layers configuration connected through individual weights. The weights are obtained by back-propagation training algorithms. The individual neuron (perceptron) consists of multiple inputs and a non-linear output activated through an activation function. A common activation function of choice would be the hyperbolic tangent function.

During the training stage, SVM is usually slower to train than a MLP given the same dataset and it requires further adaptation for multiclass classification. However, during the classification stage, SVM is much faster than the MLP as it requires only a cosine product. In terms of prediction accuracy, SVM is reported to have superior accuracy than MLP in many literature, although the performance will also depend on the nature of the problem, data configuration and other constraints.

The 10-fold cross validation method was used to get the results in [27]. The SVM uses the radial basis function (RBF) kernel while the MLP has 20 hidden nodes. Features used in [27] consisted of the three entropy features. The results of FMM-RF are the highest, with the lowest standard deviation. With these three features, SVM [27] acquired the lowest accuracy rate, with almost 6% lower than that of FMM-RF. The FMM-RF achieved better accuracy with the same sample entropy feature set and with much reduced structure complexity and training efforts. The classification rules obtained from FMM-RF is also easily comprehensible.

7. Conclusions

The classification results of ball bearing faults using vibration signals have been presented in this paper. Various condition monitoring techniques with vibration signals using intelligent systems are detailed. The hybrid FMM-RF model has been proposed and used in the experiments, which were divided into benchmark and real-world data. Power spectrum and sample entropy features were used in the feature extraction, where important features were extracted. Both the benchmark and real-world data set showed accurate performances using the FMM-RF model. The best results of benchmark and real-world data sets were at 99.9% and 99.8% respectively. In addition to accurate results, explanatory rules from a decision tree generated by FMM-RF, which explained the results, are presented. This study does indicate the usefulness of the proposed hybrid FMM-RF model for classification of ball bearing faults.

Acknowledgements

Professor Nandi is a Distinguished Visiting Professor at Tongji University, Shanghai, China. This work was partly supported by the National Science Foundation of China grant number 61520106006 and the National Science Foundation of Shanghai grant number 16JC1401300.

References

[1] H. Guo, L.B. Jack, A.K. Nandi, Feature generation using genetic programming with application to fault classification, IEEE Trans. Syst. Man Cybernet. Part B: Cybernet. 35 (1) (2005) 89–99.
[2] Y. Zhan, C.K. Mechefske, Robust detection of gearbox deterioration using compromised autoregressive modeling and Kolmogorov–Smirnov test statistic—part I: compromised autoregressive modeling with the aid of hypothesis tests and simulation analysis, Mech. Syst. Sig. Process. 21 (5) (2007) 1953–1982.
[3] J. Mathew, R.J. Alfredson, The condition monitoring of rolling element bearings Using vibration analysis, J. Vibr. Acoust. Stress Reliab. Des. 106 (1984) 447–453.
[4] L. Zhang, L.B. Jack, A.K. Nandi, Fault detection using genetic programming, Mech. Syst. Sig. Process. 19 (2) (2005) 271–289.
[5] L. Zhang, A.K. Nandi, Fault classification using genetic programming, Mech. Syst. Sig. Process. 21 (2007) 1273–1284.
[6] M.L.D. Wong, L.B. Jack, A.K. Nandi, Modified self-organising map for automated novelty detection applied to vibration signal monitoring, Mech. Syst. Sig. Process. 20 (3) (2006) 593–610.
[7] M. Timusk, M. Lipsett, C.K. Mechefske, Fault detection using transient machine signals, Mech. Syst. Sig. Process. 22 (7) (2008) 1724–1749.
[8] A. Rojas, A.K. Nandi, Practical scheme for fast detection and classification of rolling-element bearing faults using support vector machines, Mech. Syst. Sig. Process. 20 (7) (2006) 1523–1536.
[9] M.L.D. Wong, M. Zhang, A.K. Nandi, Effects of compressed sensing on classification of bearing faults with entropic features, 2015 23rd European Signal Processing Conference (EUSIPCO) (2015) 2256–2260.
[10] J.B. Ali, N. Fnaiech, L. Saidi, B. Chebel-Morello, F. Fnaiech, Application of empirical mode decomposition and artificial neural network for automatic bearing fault diagnosis based on vibration signals, Appl. Acoust. 89 (2015) 16–27.
[11] M. Unal, M. Onat, M. Demetgul, H. Kucuk, Fault diagnosis of rolling bearings using a genetic algorithm optimized neural network, Measurement 58 (2014) 187–196.
[12] S.W. Fei, Kurtosis forecasting of bearing vibration signal based on the hybrid model of empirical mode decomposition and RVM with artificial bee colony algorithm, Expert Syst. Appl. 42 (11) (2015) 5011–5018.
[13] J.B. Ali, B. Chebel-Morello, L. Saidi, S. Malinowski, F. Fnaiech, Accurate bearing remaining useful life prediction based on Weibull distribution and artificial neural network, Mech. Syst. Sig. Process. 56 (2015) 150–172.
[14] Z. Tian, An artificial neural network method for remaining useful life prediction of equipment subject to condition monitoring, J. Intell. Manuf. 23 (2) (2012) 227–237.
[15] J.B. Ali, L. Saidi, A. Mouelhi, B. Chebel-Morello, F. Fnaiech, Linear feature selection and classification using PNN and SFAM neural networks for a nearly online diagnosis of bearing naturally progressing degradations, Eng. Appl. Artif. Intell. 42 (2015) 67–81.
[16] E.P. De Moura, C.R. Souto, A.A. Silva, M.A.S. Irmao, Evaluation of principal component analysis and neural network performance for bearing fault diagnosis from vibration signal processed by RS and DF analyses, Mech. Syst. Sig. Process. 25 (5) (2011) 1765–1772.
[17] M. Barakat, M. El Badaoui, F. Guillet, Hard competitive growing neural network for the diagnosis of small bearing faults, Mech. Syst. Sig. Process. 37 (1) (2013) 276–292.
[18] C. Chen, G. Vachtsevanos, Bearing condition prediction considering uncertainty: an interval type-2 fuzzy neural network approach, Rob. Comput. Integr. Manuf. 28 (4) (2012) 509–516.
[19] H. Wang, P. Chen, Intelligent diagnosis method for rolling element bearing faults using possibility theory and neural network, Comput. Ind. Eng. 60 (4) (2011) 511–518.
[20] S. Fatima, B. Guduri, A.R. Mohanty, V.N.A. Naikan, Transducer invariant multi-class fault classification in a rotor-bearing system using support vector machines, Measurement 58 (2014) 363–374.

[21] X. Chen, J. Zhou, J. Xiao, X. Zhang, H. Xiao, W. Zhu, W. Fu, Fault diagnosis based on dependent feature vector and probability neural network for rolling element bearings, Appl. Math. Comput. 247 (2014) 835–847.
[22] X. Xia, J. Zhou, J. Xiao, H. Xiao, A novel identification method of Volterra series in rotor-bearing system for fault diagnosis, Mech. Syst. Sig. Process. (2015), http://dx.doi.org/10.1016/j.ymssp.2015.05.006.
[23] H.M. Ertunc, H. Ocak, C. Aliustaoglu, ANN-and ANFIS-based multi-staged decision algorithm for the detection and diagnosis of bearing faults, Neural Comput. Appl. 22 (1) (2013) 435–446.
[24] İ. Eski, Vibration analysis of drilling machine using proposed artificial neural network predictors, J. Mech. Sci. Technol. 26 (10) (2012) 3037–3046.
[25] L.A. Wulandhari, A. Wibowo, M.I. Desa, Condition diagnosis of multiple bearings using adaptive operator probabilities in genetic algorithms and back propagation neural networks, Neural Comput. Appl. 26 (1) (2015) 57–65.
[26] J.S. Richman, J.R. Moorman, Physiological time-series analysis using approximate entropy and sample entropy, Am. J. Physiol. Heart Circ. Physiol. 278 (12) (2000) H2039–H2049.
[27] M.L.D. Wong, C. Liu, A.K. Nandi, Classification of ball bearing faults using entropic measures, Surveillance 7 International Conference (2013).
[28] S.M. Pincus, Approximate entropy as a measure of system complexity, Proc. Natl. Acad. Sci. U. S. A. 88 (1992) 2297–2301.
[29] G. Fele-Žorž, A faster algorithm for calculating the sample entropy, in: Proceedings of Middle-European Conference on Applied Theoretical Computer Science, Slovenia, 2013.
[30] P.K. Simpson, Fuzzy Min-Max neural networks-part 1: classification, IEEE Trans. Neural Netw. 3 (5) (1992) 776–786.
[31] J. Han, M. Kamber, J. Pei, Data Mining: Concepts and Techniques, 3rd ed., Morgan Kaufmann, MA, USA, 2012.
[32] Y. Yohannes, P. Webb, Classification and Regression Trees, International Food Policy Research Institute, USA, 1999.
[33] A. Verikas, A. Gelzinis, M. Bacauskiene, Mining data with random forests: a survey and results of new tests, Pattern Recogn. 44 (2) (2011) 330–349.
[34] M. Seera, C.P. Lim, D. Ishak, H. Singh, Offline and online fault detection and diagnosis of induction motors using a hybrid soft computing model, Appl. Soft Comput. 13 (12) (2013) 4493–4507.
[35] M. Seera, C.P. Lim, C.K. Loo, Motor fault detection and diagnosis using a hybrid FMM-CART model with online learning, J. Intell. Manuf. (2014) 1–13.
[36] B. Efron, R. Tibshirani, An Introduction to the Bootstrap, Chapman and Hall, New York, 1993.
[37] H. Guo, L.B. Jack, A.K. Namdi, Feature generation using genetic programming with application to fault classification? IEEE Trans. Syst. Man Cybernet. Part B 35 (1) (2005) 89–99.
[38] L.B. Jack, A.K. Namdi, Fault detection using support vector machines and artificial neural networks, augmented by genetic algorithms, Mech. Syst. Sig. Process. 16 (2–3) (2002) 373–390.