Dazhong Wu[1]
Department of Industrial and
Manufacturing Engineering,
National Science Foundation
Center for e-Design,
Pennsylvania State University,
University Park, PA 16802
e-mail: dxw279@psu.edu

Connor Jennings
Department of Industrial and
Manufacturing Engineering,
National Science Foundation
Center for e-Design,
Pennsylvania State University,
University Park, PA 16802
e-mail: connor@psu.edu

Janis Terpenny
Department of Industrial and
Manufacturing Engineering,
National Science Foundation
Center for e-Design,
Pennsylvania State University,
University Park, PA 16802
e-mail: jpt5311@psu.edu

Robert X. Gao
Department of Mechanical and
Aerospace Engineering,
Case Western Reserve University,
Cleveland, OH 44106
e-mail: robert.gao@case.edu

Soundar Kumara
Department of Industrial and
Manufacturing Engineering,
Pennsylvania State University,
University Park, PA 16802
e-mail: skumara@psu.edu

A Comparative Study on Machine Learning Algorithms for Smart Manufacturing: Tool Wear Prediction Using Random Forests

Manufacturers have faced an increasing need for the development of predictive models that predict mechanical failures and the remaining useful life (RUL) of manufacturing systems or components. Classical model-based or physics-based prognostics often require an in-depth physical understanding of the system of interest to develop closed-form mathematical models. However, prior knowledge of system behavior is not always available, especially for complex manufacturing systems and processes. To complement model-based prognostics, data-driven methods have been increasingly applied to machinery prognostics and maintenance management, transforming legacy manufacturing systems into smart manufacturing systems with artificial intelligence. While previous research has demonstrated the effectiveness of data-driven methods, most of these prognostic methods are based on classical machine learning techniques, such as artificial neural networks (ANNs) and support vector regression (SVR). With the rapid advancement in artificial intelligence, various machine learning algorithms have been developed and widely applied in many engineering fields. The objective of this research is to introduce a random forests (RFs)-based prognostic method for tool wear prediction as well as compare the performance of RFs with feed-forward back propagation (FFBP) ANNs and SVR. Specifically, the performance of FFBP ANNs, SVR, and RFs are compared using an experimental data collected from 315 milling tests. Experimental results have shown that RFs can generate more accurate predictions than FFBP ANNs with a single hidden layer and SVR. [DOI: 10.1115/1.4036350]

Keywords: tool wear prediction, predictive modeling, machine learning, random forests (RFs), support vector machines (SVMs), artificial neural networks (ANNs), prognostics and health management (PHM)

1 Introduction

Smart manufacturing aims to integrate big data, advanced analytics, high-performance computing, and Industrial Internet of Things (IIoT) into traditional manufacturing systems and processes to create highly customizable products with higher quality at lower costs. As opposed to traditional factories, a smart factory utilizes interoperable information and communications technologies (ICT), intelligent automation systems, and sensor networks to monitor machinery conditions, diagnose the root cause of failures, and predict the remaining useful life (RUL) of mechanical systems or components. For example, almost all engineering systems (e.g., aerospace systems, nuclear power plants, and machine tools) are subject to mechanical failures resulting from deterioration with usage and age or abnormal operating conditions [1–3]. Some of the typical failure modes include excessive load, overheating, deflection, fracture, fatigue, corrosion, and wear. The degradation and failures of engineering systems or components will often incur higher costs and lower productivity due to unexpected machine downtime. In order to increase manufacturing productivity while reducing maintenance costs, it is crucial to develop and implement an intelligent maintenance strategy that allows manufacturers to determine the condition of in-service systems in order to predict when maintenance should be performed.

Conventional maintenance strategies include reactive, preventive, and proactive maintenance [4–6]. The most basic approach to maintenance is reactive, also known as run-to-failure maintenance planning. In the reactive maintenance strategy, assets are deliberately allowed to operate until failures actually occur. The assets are maintained on an as-needed basis. One of the disadvantages of reactive maintenance is that it is difficult to anticipate the maintenance resources (e.g., manpower, tools, and replacement parts) that will be required for repairs. Preventive maintenance is often referred to as use-based maintenance. In preventive maintenance, maintenance activities are performed after a specified period of time or amount of use based on the estimated probability that the systems or components will fail in the specified time interval. Although preventive maintenance allows for more consistent and predictable maintenance schedules, more maintenance activities are needed as opposed to reactive maintenance. To improve

the efficiency and effectiveness of preventive maintenance, predictive maintenance is an alternative strategy in which maintenance actions are scheduled based on equipment performance or conditions instead of time. The objective of proactive maintenance is to determine the condition of in-service equipment and ultimately to predict the time at which a system or a component will no longer meet desired functional requirements.

The discipline that predicts health condition and remaining useful life (RUL) based on previous and current operating conditions is often referred to as prognostics and health management (PHM). Prognostic approaches fall into two categories: model-based and data-driven prognostics [7–12]. Model-based prognostics refer to approaches based on mathematical models of system behavior derived from physical laws or probability distribution. For example, model-based prognostics include methods based on Wiener and Gamma processes [13], hidden Markov models (HMMs) [14], Kalman filters [15,16], and particle filters [17–20]. One of the limitations of model-based prognostics is that an in-depth understanding of the underlying physical processes that lead to system failures is required. Another limitation is that it is assumed that underlying processes follow certain probability distributions, such as gamma or normal distributions. While probability density functions enable uncertainty quantification, distributional assumptions may not hold true in practice.

To complement model-based prognostics, data-driven prognostics refer to approaches that build predictive models using learning algorithms and large volumes of training data. For example, classical data-driven prognostics are based on autoregressive (AR) models, multivariate adaptive regression, fuzzy set theory, ANNs, and SVR. The unique benefit of data-driven methods is that an in-depth understanding of system physical behaviors is not a prerequisite. In addition, data-driven methods do not assume any underlying probability distributions which may not be practical for real-world applications. While ANNs and SVR have been applied in the area of data-driven prognostics, little research has been conducted to evaluate the performance of other machine learning algorithms [21]. Because RFs have the potential to handle a large number of input variables without variable selection and they do not overfit [22–24], we investigate the ability of RFs for the prediction of tool wear using an experimental dataset. Further, the performance of RFs is compared with that of FFBP ANNs and SVR using accuracy and training time.

The main contributions of this paper include the followings:

- Tool wear in milling operations is predicted using RFs along with cutting force, vibration, and acoustic emission (AE) signals. Experimental results have shown that the predictive model trained by RFs is very accurate. The mean squared error (MSE) on the test tool wear data is up to 7.67. The coefficient of determination (R^2) on the test tool wear data is up to 0.992. To the best of our knowledge, the random forest algorithm is applied to predict tool wear for the first time.
- The performances of ANNs, support vector machines (SVMs), and RFs are compared using an experimental dataset with respect to the accuracy of regression (e.g., MSE and R^2) and training time. While the training time for RFs is longer than that of ANNs and SVMs, the predictive model built by RFs is the most accurate for the application example.

The remainder of the paper is organized as follows: Section 2 reviews the related literature on data-driven methods for tool wear prediction. Section 3 presents the methodology for tool wear prediction using ANNs, SVMs, and RFs. Section 4 presents an experimental setup and the experimental dataset acquired from different types of sensors (e.g., cutting force sensor, vibration sensor, acoustic emission sensor) on a computer numerical control (CNC) milling machine. Section 5 presents experimental results, demonstrates the effectiveness of the three machine learning algorithms, and compares the performance of each. Section 6 provides conclusions that include a discussion of research contribution and future work.

2 Data-Driven Methods for Tool Wear Prediction

Tool wear is the most commonly observed and unavoidable phenomenon in manufacturing processes, such as drilling, milling, and turning [25–27]. The rate of tool wear is typically affected by process parameters (e.g., cutting speed and feed rate), cutting tool geometry, and properties of workpiece and tool materials. Taylor's equation for tool life expectancy [28] provides an approximation of tool wear. However, with the rapid advancement of sensing technology and increasing number of sensors equipped on modern CNC machines, it is possible to predict tool wear more accurately using various measurement data. This section presents a review of data-driven methods for tool wear prediction.

Schwabacher and Goebel [29] conducted a review of data-driven methods for prognostics. The most popular data-driven approaches to prognostics include ANNs, decision trees, and SVMs in the context of systems health management. ANNs are a family of computational models based on biological neural networks which are used to estimate complex relationships between inputs and outputs. Bukkapatnam et al. [30–32] developed effective tool wear monitoring techniques using ANNs based on features extracted from the principles of nonlinear dynamics. Özel and Karpat [33] presented a predictive modeling approach for surface roughness and tool wear for hard turning processes using ANNs. The inputs of the ANN model include workpiece hardness, cutting speed, feed rate, axial cutting length, and mean values of three force components. Experimental results have shown that the model trained by ANNs provides accurate predictions of surface roughness and tool flank wear. Palanisamy et al. [34] developed a predictive model for predicting tool flank wear in end milling operations using feed-forward back propagation (FFBP) ANNs. Experimental results have shown that the predictive model based on ANNs can make accurate predictions of tool flank wear using cutting speeds, feed rates, and depth of cut. Sanjay et al. [35] developed a model for predicting tool flank wear in drilling using ANNs. The feed rates, spindle speeds, torques, machining times, and thrust forces are used to train the ANN model. The experimental results have demonstrated that ANNs can predict tool wear accurately. Chungchoo and Saini [36] developed an online fuzzy neural network (FNN) algorithm that estimates the average width of flank wear and maximum depth of crater wear. A modified least-square backpropagation neural network was built to estimate flank and crater wear based on cutting force and acoustic emission signals. Chen and Chen [37] developed an in-process tool wear prediction system using ANNs for milling operations. A total of 100 experimental data were used for training the ANN model. The input variables include feed rate, depth of cut, and average peak cutting forces. The ANN model can predict tool wear with an error of 0.037 mm on average. Paul and Varadarajan [38] introduced a multisensor fusion model to predict tool wear in turning processes using ANNs. A regression model and an ANN were developed to fuse the cutting force, cutting temperature, and vibration signals. Experimental results showed that the coefficient of determination was 0.956 for the regression model trained by the ANN. Karayel [39] presented a neural network approach for the prediction of surface roughness in turning operations. A feed-forward back-propagation multilayer neural network was developed to train a predictive model using the data collected from 49 cutting tests. Experimental results showed that the predictive model has an average absolute error of 2.29%.

Cho et al. [40] developed an intelligent tool breakage detection system with the SVM algorithm by monitoring cutting forces and power consumption in end milling processes. Linear and polynomial kernel functions were applied in the SVM algorithm. It has been demonstrated that the predictive model built by SVMs can recognize process abnormalities in milling. Benkedjouh et al. [41] presented a method for tool wear assessment and remaining useful life prediction using SVMs. The features were extracted from cutting force, vibration, and acoustic emission signals. The experimental results have shown that SVMs can be used to estimate the

wear progression and predict RUL of cutting tools effectively. Shi and Gindy [42] introduced a predictive modeling method by combining least squares SVMs and principal component analysis (PCA). PCA was used to extract statistical features from multiple sensor signals acquired from broaching processes. Experimental results showed that the predictive model trained by SVMs was effective to predict tool wear using the features extracted by PCA.

Another data-driven method for prognostics is based on decision trees. Decision trees are a nonparametric supervised learning method used for classification and regression. The goal of decision tree learning is to create a model that predicts the value of a target variable by learning decision rules inferred from data features. A decision tree is a flowchart-like structure in which each internal node denotes a test on an attribute, each branch represents the outcome of a test, and each leaf node holds a class label. Jiaa and Dornfeld [43] proposed a decision tree-based method for the prediction of tool flank wear in a turning operation using acoustic emission and cutting force signals. The features characterizing the AE root-mean-square and cutting force signals were extracted from both time and frequency domains. The decision tree approach was demonstrated to be able to make reliable inferences and decisions on tool wear classification. Elangovan et al. [44] developed a decision tree-based algorithm for tool wear prediction using vibration signals. Ten-fold cross-validation was used to evaluate the accuracy of the predictive model created by the decision tree algorithm. The maximum classification accuracy was 87.5%. Arisoy and Özel [45] investigated the effects of machining parameters on surface microhardness and microstructure such as grain size and fractions using a random forests-based predictive modeling method along with finite element simulations. Predicted microhardness profiles and grain sizes were used to understand the effects of cutting speed, tool coating, and edge radius on the surface integrity.

In summary, the related work presented in this section builds on previous research to explore how the conditions of tool wear can be monitored as well as how tool wear can be predicted using predictive modeling. While earlier work focused on prediction of tool wear using ANNs, SVMs, and decision trees, this paper explores the potential of a new method, random forests, for tool wear prediction. Further, the performance of RFs is compared with that of ANNs and SVMs. Because RFs are an extension of decision trees, the performance of RFs is not compared with that of decision trees.

3 Methodology

This section presents the methodology for data-driven prognostics for tool wear prediction using ANNs, SVR, and RFs. The input of ANNs, SVR, and RFs is the following labeled training data:

$$D = (x_i, y_i)$$

where $x_i = (F_X, F_Y, F_Z, V_X, V_Y, V_Z, AE)$, $y_i \in \mathbb{R}$. The description of these input data can be found in Table 1.

3.1 Tool Wear Prediction Using ANNs. ANNs are a family of models inspired by biological neural networks. An ANN is

Table 1 Signal channel and data description

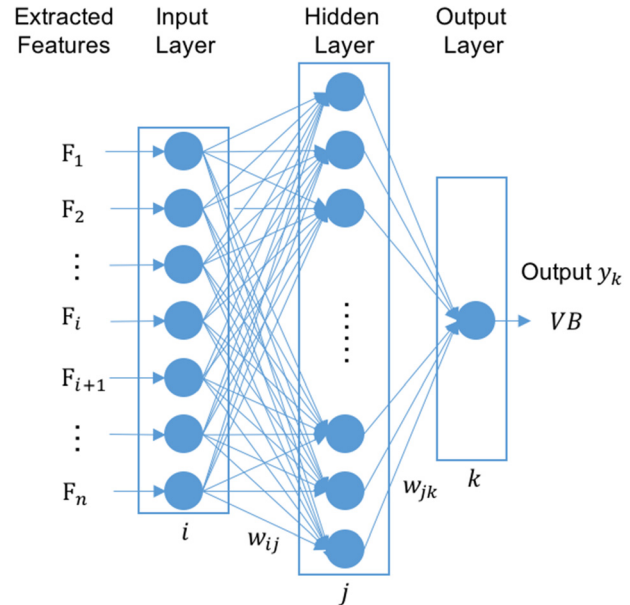| Signal channel	Data description
Channel 1	F_X: force (N) in X dimension
Channel 2	F_Y : force (N) in Y dimension
Channel 3	F_Z: force (N) in Z dimension
Channel 4	V_X: vibration (g) in X dimension
Channel 5	V_Y: vibration (g) in Y dimension
Channel 6	V_Z: vibration (g) in Z dimension
Channel 7	AE: acoustic emission (V)

Fig. 1 Tool wear prediction using a feed-forward back-propagation (FFBP) ANN

defined by three types of parameters: (1) the interconnection pattern between different layers of neurons, (2) the learning process for updating the weights of the interconnections, and (3) the activation function that converts a neuron's weighted input to its output activation. Among many types of ANNs, the feed-forward neural network is the first and the most popular ANN. Back-propagation is a learning algorithm for training ANNs in conjunction with an optimization method such as gradient descent.

Figure 1 illustrates the architecture of the FFBP ANN with a single hidden layer. In this research, the ANN has three layers, including input layer i, hidden layer j, and output layer k. Each layer consists of one or more neurons or units, represented by the circles. The flow of information is represented by the lines between the units. The first layer has input neurons which act as buffers for distributing the extracted features (i.e., F_i) from the input data (i.e., x_i). The number of the neurons in the input layer is the same as that of extracted features from input variables. Each value from the input layer is duplicated and sent to all neurons in the hidden layer. The hidden layer is used to process and connect the information from the input layer to the output layer in a forward direction. Specifically, these values entering a neuron in the hidden layer are multiplied by weights w_{ij}. Initial weights are randomly selected between 0 and 1. A neuron in the hidden layer sums up the weighted inputs and generates a single output. This value is the input of an activation function (sigmoid function) in the hidden layer f_h that converts the weighted input to the output of the neuron. Similarly, the outputs of all the neurons in the hidden layer are multiplied by weights w_{jk}. A neural in the output layer sums up the weighted inputs and generates a single value. An activation function in the output layer f_o converts the weighted input to the predicted output y_k of the ANN, which is the predicted flank wear VB. The output layer has only one neuron because there is only one response variable. The performance of ANNs depends on the topology or architecture of ANNs (i.e., the number of layers) and the number of neurons in each layer. However, there are no standard or well-accepted methods or rules for determining the number of hidden layers and neurons in each hidden layer. In this research, the single-hidden-layer ANNs with 2, 4, 8, 16, and 32 neurons in the hidden layer are selected. The termination criterion of the training algorithm is that training stops if the fit criterion (i.e., least squares) falls below 1.0×10^{-4}.
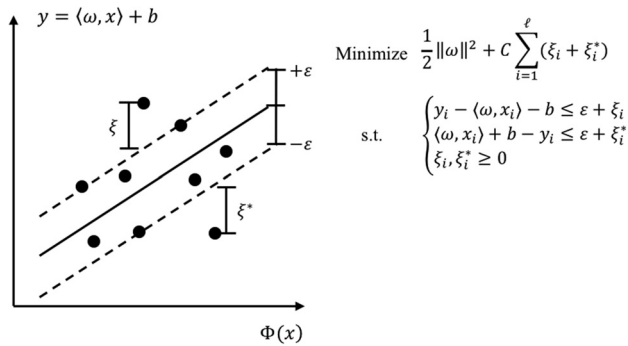
Fig. 2 Tool wear prediction using SVR

3.2 Tool Wear Prediction Using SVR. The original SVM for regression was developed by Vapnik and coworkers [46,47]. A SVM constructs a hyperplane or set of hyperplanes in a high- or infinite-dimensional space, which can be used for classification and regression.

The framework of SVR for linear cases is illustrated in Fig. 2. Formally, SVR can be formulated as a convex optimization problem

$$\text{Minimize} \quad \frac{1}{2}\|\omega\|^2 + C\sum_{i=1}^{\ell}(\xi_i + \xi_i^*)$$

$$\text{Subject to} \quad \begin{cases} y_i - \langle \omega, x_i \rangle - b \leq \varepsilon + \xi_i \\ \langle \omega, x_i \rangle + b - y_i \leq \varepsilon + \xi_i^* \\ \xi_i, \xi_i^* \geq 0 \end{cases} \quad (3.1)$$

where $\omega \in \chi, C = 1, \varepsilon = 0.1$, and $\xi_i, \xi_i^* = 0.001$. b can be computed as follows:

$$\begin{aligned} b &= y_i - \langle \omega, x_i \rangle - \varepsilon \quad \text{for } \alpha_i \in [0, C] \\ b &= y_i - \langle \omega, x_i \rangle + \varepsilon \quad \text{for } \alpha_i^* \in [0, C] \end{aligned} \quad (3.2)$$

For nonlinear SVR, the training patterns x_i can be preprocessed by a nonlinear kernel function $k(x, x') := \langle \Phi(x), \Phi(x) \rangle'$, where $\Phi(x)$ is a transformation that maps x to a high-dimensional space. These kernel functions need to satisfy the Mercer's theorem. Many kernels have been developed for various applications. The most popular kernels include polynomial, Gaussian radial basis function (RBF), and sigmoid. In many applications, a nonlinear kernel function provides better accuracy. According to the literature [32,33], the Gaussian RBF kernel is one of the most effective kernel functions used in tool wear prediction. In this research, the Gaussian RBF kernel is used to transform the input dataset $D = (x_i, y_i)$, where x_i is the input vector and y_i is the response variable (i.e., flank wear) into a new dataset in a high-dimensional space. The new dataset is linearly separable by a hyperplane in a higher-dimensional Euclidean space as illustrated in Fig. 2. The slack variables ξ_i and ξ_i^* are introduced in the instances where the constraints are infeasible. The slack variables denote the deviation from predicted values with the error of $\varepsilon = 0.1$. The RBF kernel is $k(x_i, x_j) = \exp(-((\|x_i - x_j\|^2)/2\sigma^2))$, where $\sigma^2 = 0.5$. At the optimal solution, we obtain

$$\omega = \sum_{i=1}^{\ell}(\alpha_i - \alpha_i^*)\Phi(x) \quad \text{and} \quad f(x) = \sum_{i=1}^{\ell}(\alpha_i - \alpha_i^*)k(x_i, x_j) + b \quad (3.3)$$

3.3 Tool Wear Prediction Using RFs. The random forest algorithm, developed by Breiman [22,48], is an ensemble learning method that constructs a forest of decision trees from bootstrap

samples of a training dataset. Each decision tree produces a response, given a set of predictor values. In a decision tree, each internal node represents a test on an attribute, each branch represents the outcome of the test, and each leaf node represents a class label for classification or a response for regression. A decision tree in which the response is continuous is also referred to as a regression tree. In the context of tool wear prediction, each individual decision tree in a random forest is a regression tree because tool wear describes the gradual failure of cutting tools. A comprehensive tutorial on RFs can be found in Refs. [22,48,49]. Some of the important concepts related to RFs, including bootstrap aggregating or bagging, slipping, and stopping criterion, are introduced in Secs. 3.3.1–3.3.4.

3.3.1 Bootstrap Aggregating or Bagging. Given a training dataset $D = \{(x_1, y_1), (x_2, y_2), ..., (x_N, y_N)\}$, bootstrap aggregating or bagging generates B new training datasets D_i of size N by sampling from the original training dataset D with replacement. D_i is referred to as a bootstrap sample. By sampling with replacement or bootstrapping, some observations may be repeated in each D_i. Bagging helps reduce variance and avoid overfitting. The number of regression trees B is a parameter specified by users. Typically, a few hundred to several thousand trees are used in the random forest algorithm.

3.3.2 Choosing Variables to Split On. For each of the bootstrap samples, grow an un-pruned regression tree with the following procedure: At each node, randomly sample m variables and choose the best split among those variables rather than choosing the best split among all predictors. This process is sometimes called "feature bagging." The reason why a random subset of the predictors or features is selected is because the correlation of the trees in an ordinary bootstrap sample can be reduced. For regression, the default $m = p/3$.

3.3.3 Splitting Criterion. Suppose that a partition is divided into M regions $R_1, R_2, ..., R_m$. The response is modeled as a constant c_m in each region

$$f(x) = \sum_{m=1}^{M} c_m I(x \epsilon R_m) \quad (3.4)$$

The splitting criterion at each node is to minimize the sum of squares. Therefore, the best $\widehat{c_m}$ is the average of y_i in region R_m

$$\widehat{c_m} = \text{ave}(y_i | x_i \epsilon R_m) \quad (3.5)$$

Consider a splitting variable j and split point s, and define the pair of half-planes

$$R_1(j, s) = \{X | X_j \leq s\} \quad \text{and} \quad R_2(j, s) = \{X | X_j \geq s\} \quad (3.6)$$

The splitting variable j and split point s should satisfy

$$\min_{j,s}\left[\min_{c1}\sum_{x_i \in R_1(j,s)}(y_i - c1)^2 + \min_{c2}\sum_{x_i \in R_2(j,s)}(y_i - c2)^2\right] \quad (3.7)$$

For any j and s, the inner minimization is solved by

$$\widehat{c_1} = \text{ave}(y_i | x_i \epsilon R_1(j, s)) \quad \text{and} \quad \widehat{c_2} = \text{ave}(y_i | x_i \epsilon R_2(j, s)) \quad (3.8)$$

Having found the best split, the dataset is partitioned into two resulting regions and repeat the splitting process on each of the two regions. This splitting process is repeated until a predefined stopping criterion is satisfied.

3.3.4 Stopping Criterion. Tree size is a tuning parameter governing the complexity of a model. The stopping criterion is that the splitting process proceeds until the number of records in D_i falls below a threshold, and five is used as the threshold.
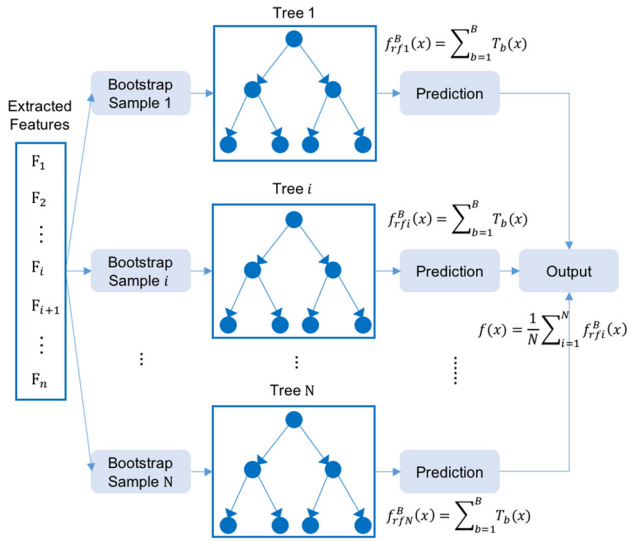
Fig. 3 Tool wear prediction using an RF

Fig. 5 Experimental setup

After B such trees $\{T_b\}_1^B$ are constructed, a prediction at a new point x can be made by averaging the predictions from all the individual B regression trees on x

$$\hat{f}_{rf}^B(x) = \frac{1}{B} \sum_{b=1}^{B} T_b(x) \qquad (3.9)$$

The random forest algorithm [48,49] for regression is as follows:

(1) Draw a bootstrap sample Z of size N from the training data.
(2) For each bootstrap sample, construct a regression tree by splitting a node into two children nodes until the stopping criterion is satisfied.
(3) Output the ensemble of trees $\{T_b\}_1^B$.
(4) Make a prediction at a new point x by aggregating the predictions of the B trees.

The framework of predicting flank wear using an RF is illustrated in Fig. 3. In this research, a random forest is constructed using $B = 500$ regression trees. Given the labeled training dataset $D = (x_i, y_i)$, a bootstrap sample of size $N = 630$ is drawn from the training dataset. For each regression tree, $m = 9$ ($m = (p/3)$, $p = 28$) variables are selected at random from the 28 variables/features. The best variable/split-point is selected among the nine variables. A regression tree progressively splits the training dataset into two child nodes: left node (with samples $<z$) and right node (with samples $\geq z$). A splitting variable and split point are selected by solving Eqs. (3.7) and (3.8). The process is applied recursively on the dataset in each child node. The splitting process stops if the number of records in a node is less than 5. An
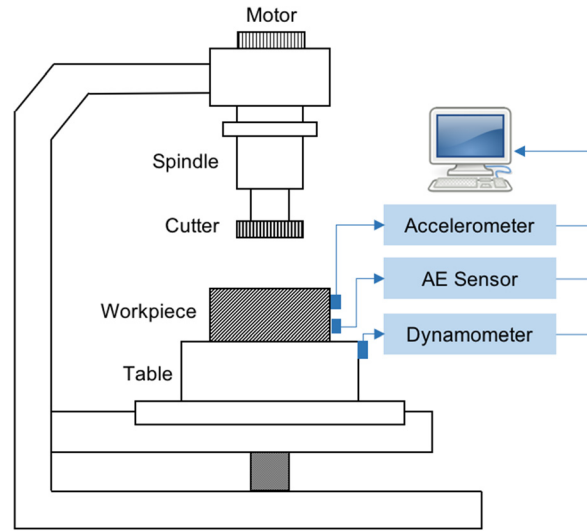
individual regression tree is built by starting at the root node of the tree, performing a sequence of tests about the predictors, and organizing the tests in a hierarchical binary tree structure as shown in Fig. 4. After 500 regression trees are constructed, a prediction at a new point can be made by averaging the predictions from all the individual binary regression trees on this point.

4 Experimental Setup

The data used in this paper were obtained from Li et al. [50]. Some details of the experiment are presented in this section. The experimental setup is shown in Fig. 5.

The cutter material and workpiece material used in the experiment are high-speed steel and stainless steel, respectively. The detailed description of the operating conditions in the dry milling operation can be found in Table 2. The spindle speed of the cutter was 10,400 RPM. The feed rate was 1555 mm/min. The Y depth of cut (radial) was 0.125 mm. The Z depth of cut (axial) was 0.2 mm.

315 cutting tests were conducted on a three-axis high-speed CNC machine (Röders Tech RFM 760). During each cutting test, seven signal channels, including cutting force, vibration, and acoustic emission data, were monitored in real-time. The sampling rate was 50 kHz/channel. Each cutting test took about 15 s. A stationary dynamometer, mounted on the table of the CNC machine, was used to measure cutting forces in three, mutually perpendicular axes (x, y, and z dimensions). Three piezo accelerometers, mounted on the workpiece, were used to measure vibration in three, mutually perpendicular axes (x, y, and z dimensions). An acoustic emission (AE) sensor, mounted on the workpiece, was
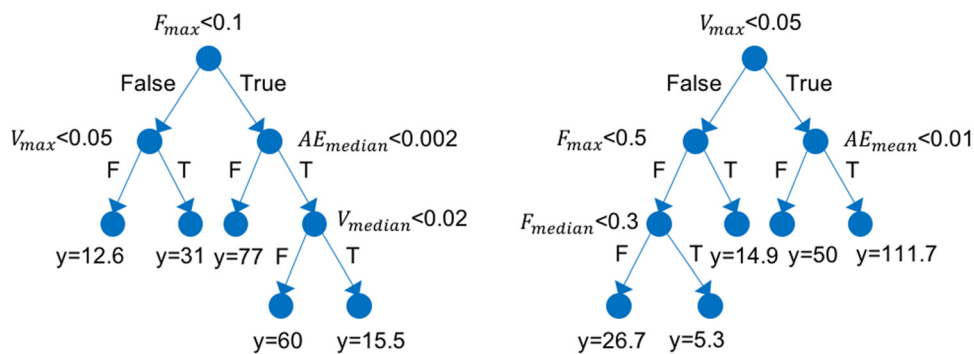

Fig. 4 Binary regression tree growing process

Parameter	Value
Spindle speed	10,400 RPM
Feed rate	1555 mm/min
Y depth of cut	0.125 mm
Z depth of cut	0.2 mm
Sampling rate	50 kHz/channel
Material	Stainless steel

Table 3 List of extracted features

Cutting force (X, Y, Z dimensions)	Vibration (X, Y, Z dimensions)	Acoustic emission
Max	Max	Max
Median	Median	Median
Mean	Mean	Mean
Standard deviation	Standard deviation	Standard deviation

Fig. 7 Comparison of observed and predicted tool wear using SVR (termination criterion: slack variable or tolerance ξ is equal to 0.001)

used to monitor a high-frequency oscillation that occurs spontaneously within metals due to crack formation or plastic deformation. Acoustic emission is caused by the release of strain energy as the microstructure of the material is rearranged. After each cutting test, the value of tool wear was measured off-line using a microscope (Leica MZ12). The total size of the condition monitoring data is about 8.67 GB.

5 Results and Discussion

In machine learning, feature extraction is an essential preprocessing step in which raw data collected from various signal channels are converted into a set of statistical features in a format supported by machine learning algorithms. The statistical features are then given as an input to a machine learning algorithm. In this experiment, the condition monitoring data were collected from (1) cutting force, (2) vibration, and (3) acoustic emission signal channels. A set of statistical features (28 features) was extracted from these signals, including maximum, median, mean, and standard deviation as listed in Table 3.
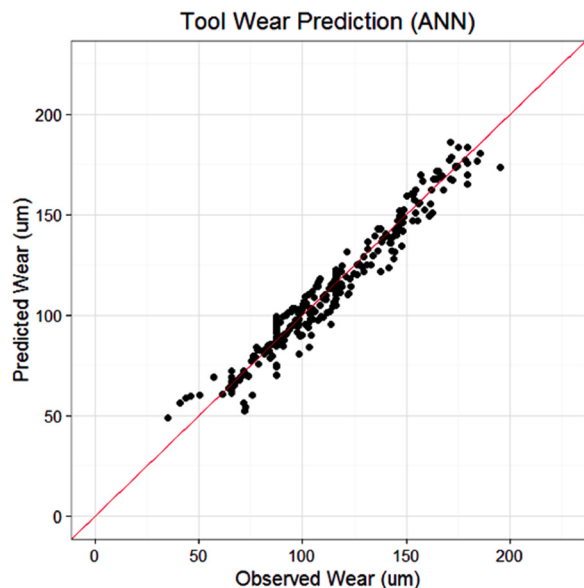
Three predictive models were developed using ANNs, SVR, and RFs, respectively. Two-thirds (2/3) of the input data (i.e., three datasets) were selected at random for model development (training). The remainder (1/3) of the input data was used for model validation (testing). Figures 6–8 show the predicted against observed tool wear values with the test dataset using ANNs, SVR, and RFs, respectively. Figure 9 shows the tool wear against time with RFs.

In addition, the performance of the three algorithms was evaluated on the test dataset using accuracy and training time. Accuracy is measured using the R^2 statistic, also referred to as the coefficient of determination, and mean squared error (MSE). In statistics, the coefficient of determination is defined as $R^2 = 1 - (\text{SSE}/\text{SST})$, where SSE is the sum of the squares of

Fig. 6 Comparison of observed and predicted tool wear using an ANN with 16 neurons in the hidden layer (termination criterion: tolerance is equal to 1.0×10^{-4})
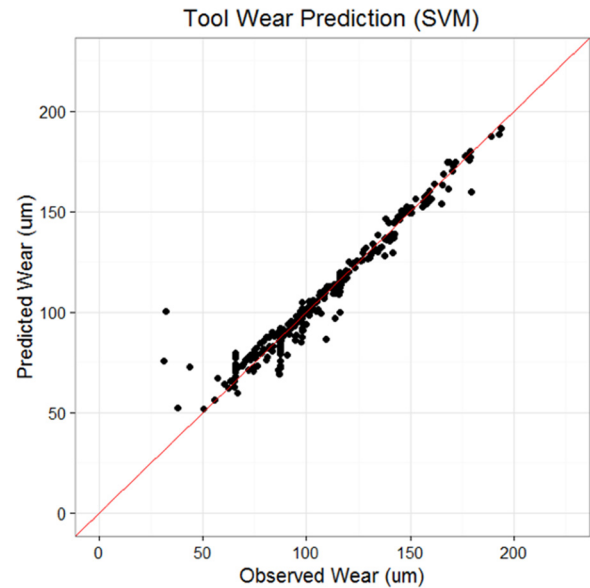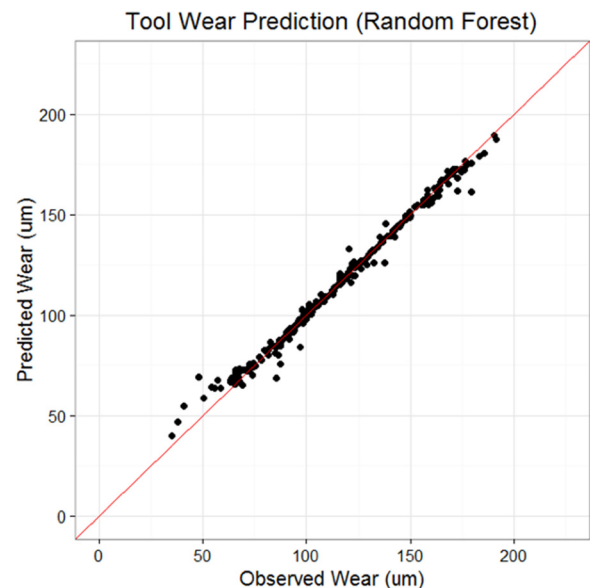

Fig. 8 Comparison of observed and predicted tool wear using RFs (termination criterion: minimum number of samples in each node is equal to 5)
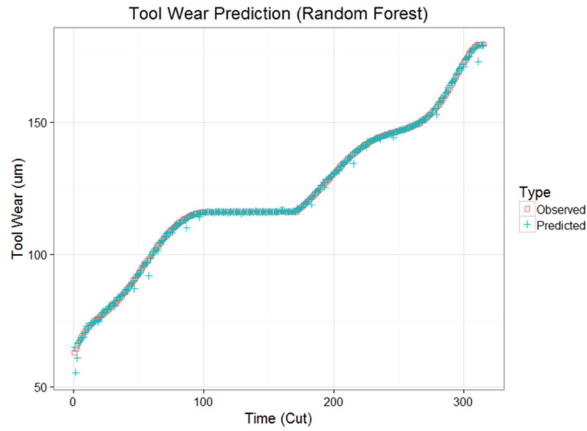
Fig. 9   Tool wear against time (cut) using RFs

Table 4   Accuracy on the test data and training time for the FFBP ANN with two neurons in the hidden layer

Training size (%)	ANN (number of neurons = 2)		
	MSE	R^2	Training time (s)
50	49.790	0.951	0.049
60	45.072	0.955	0.054
70	45.626	0.956	0.055
80	47.966	0.953	0.062
90	48.743	0.955	0.056

Table 5   Accuracy on the test data and training time for the FFBP ANN with four neurons in the hidden layer

Training size (%)	ANN (number of neurons = 4)		
	MSE	R^2	Training time (s)
50	43.428	0.958	0.122
60	51.001	0.951	0.084
70	43.645	0.958	0.093
80	45.661	0.955	0.103
90	45.058	0.958	0.118

Table 6   Accuracy on the test data and training time for the FFBP ANN with eight neurons in the hidden layer

Training size (%)	ANN (number of neurons = 8)		
	MSE	R^2	Training time (s)
50	36.810	0.964	0.167
60	34.168	0.968	0.186
70	39.795	0.961	0.202
80	44.175	0.957	0.197
90	46.634	0.954	0.234

residuals, SST is the total sum of squares. The coefficient of determination is a measure that indicates the percentage of the response variable variation that is explained by a regression model. A higher R-squared indicates that more variability is explained by the regression model. For example, an R^2 of 100% indicates that the regression model explains all the variability of the response data around its mean. In general, the higher the R-squared, the

Table 7   Accuracy on the test data and training time for the FFBP ANN with 16 neurons in the hidden layer

Training size (%)	ANN (number of neurons = 16)		
	MSE	R^2	Training time (s)
50	36.337	0.964	0.394
60	41.420	0.959	0.412
70	40.138	0.960	0.468
80	42.486	0.957	0.506
90	44.056	0.957	0.566

Table 8   Accuracy on the test data and training time for the FFBP ANN with 32 neurons in the hidden layer

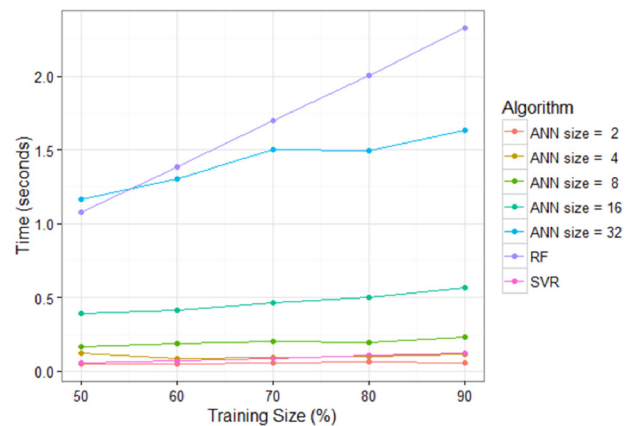| Training size (%)	ANN (number of neurons = 32)		
	MSE	R^2	Training time (s)
50	35.305	0.965	1.165
60	38.612	0.963	1.301
70	38.824	0.963	1.498
80	42.469	0.959	1.496
90	48.138	0.953	1.633

Fig. 10   Comparison of training times

better the regression model fits the data. The MSE of an estimator measures the average of the squares of the errors. The MSE is defined as $\mathrm{MSE} = (1/n)\sum_{i=1}^{n}(\widehat{y}_i - y_i)^2$, where $\widehat{y}_i$ is a predicted value, y_i is an observed value, and n is the sample size. The ANN, SVR, and RF algorithms use between 50% and 90% of the input data for model development (training) and use the remainder for model validation (testing). Because the performance of ANNs depends on the hidden layer configuration, five ANNs with a single hidden layer but different number of neurons were tested on the training dataset. Tables 4–8 list the MSE, R-squared, and training time for the ANNs with 2, 4, 8, 16, and 32 neurons. With respect to the performance of the ANN, the training time increases as the number of neurons increases. However, the increased in training time are not significant as shown in Fig. 10. In addition, while the prediction accuracy increases as the number of neurons increases, the performance is not significantly improved by adding more than eight neurons in the hidden layer as shown in Figs. 11 and 12. Tables 9 and 10 list the MSE, R-squared, and training time for SVR and RFs. While the training time for RFs is longer than that of ANNs and SVR, the predictive model built by RFs is the most accurate as shown in Figs. 10–12.
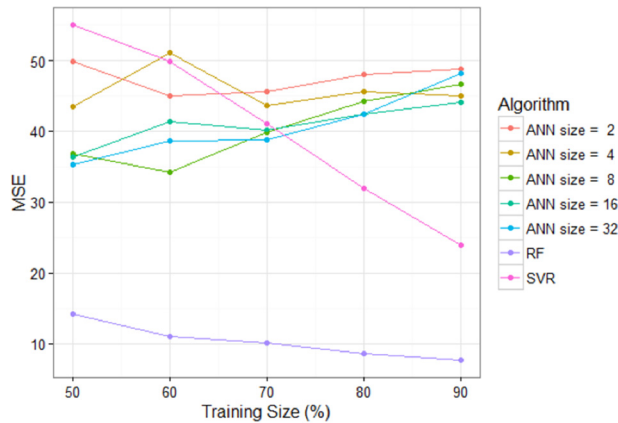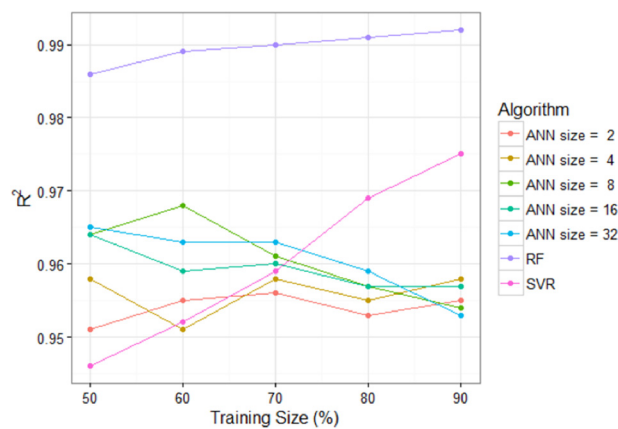
Fig. 11 Comparison of MSEs

Fig. 12 Comparison of *R*-squared errors

Table 9 Accuracy on the test data and training time for SVR with radial basis kernel

Training size (%)	SVR		
	MSE	R^2	Training time (s)
50	54.993	0.946	0.060
60	49.868	0.952	0.073
70	41.072	0.959	0.088
80	31.958	0.969	0.107
90	23.997	0.975	0.126

Table 10 Accuracy on the test data and training time for RFs

Training size (%)	RFs (500 trees)		
	MSE	R^2	Training time (s)
50	14.170	0.986	1.079
60	11.053	0.989	1.386
70	10.156	0.990	1.700
80	8.633	0.991	2.003
90	7.674	0.992	2.325

6 Conclusions and Future Work

In this paper, the prediction of tool wear in milling operations was conducted using three popular machine learning algorithms, including ANNs, SVR, and RFs. The performance of these algorithms was evaluated on the dataset collected from 315 milling tests. The performance measures include mean squared error, *R*-squared, and training time. A set of statistical features was extracted from cutting forces, vibrations, and acoustic emissions. The experimental results have shown that while the training time on the particular dataset using RFs is longer than the FFBP ANNs with a single hidden layer and SVR, RFs generate more accurate predictions than the FFBP ANNs with a single hidden layer and SVR. The main contribution of this paper is twofold: (1) we demonstrated that the predictive model trained by RFs can predict tool wear in milling processes very accurately for the first time to the best of our knowledge and (2) we compared the performance of RFs with that of FFBP ANNs and SVR, as well as observed that RFs outperform FFBP ANNs and SVR for this particular application example.

In the future, a comparison of the performance of SVR and RFs with that of other types of ANNs, such as recurrent neural networks and dynamic neural networks, will be conducted. In addition, our future work will focus on designing the parallel implementation of machine learning algorithms that can be applied to large-scale and real-time prognosis.

References

[1] Swanson, L., 2001, "Linking Maintenance Strategies to Performance," Int. J. Prod. Econ., **70**(3), pp. 237–244.
[2] Valdez-Flores, C., and Feldman, R. M., 1989, "A Survey of Preventive Maintenance Models for Stochastically Deteriorating Single-Unit Systems," Nav. Res. Logist., **36**(4), pp. 419–446.
[3] Wu, D., Terpenny, J., Zhang, L., Gao, R., and Kurfess, T., 2016, "Fog-Enabled Architecture for Data-Driven Cyber-Manufacturing Systems," ASME Paper No. MSEC2016-8559.
[4] Lee, J., 1995, "Machine Performance Monitoring and Proactive Maintenance in Computer-Integrated Manufacturing: Review and Perspective," Int. J. Comput. Integr. Manuf., **8**(5), pp. 370–380.
[5] Bevilacqua, M., and Braglia, M., 2000, "The Analytic Hierarchy Process Applied to Maintenance Strategy Selection," Reliab. Eng. Syst. Saf., **70**(1), pp. 71–83.
[6] Suh, J. H., Kumara, S. R., and Mysore, S. P., 1999, "Machinery Fault Diagnosis and Prognosis: Application of Advanced Signal Processing Techniques," CIRP Ann.-Manuf. Technol., **48**(1), pp. 317–320.
[7] Hu, C., Youn, B. D., and Kim, T., 2012, "Semi-Supervised Learning With Co-Training for Data-Driven Prognostics," IEEE Conference on Prognostics and Health Management (PHM), Denver, CO, June 18–21, pp. 1–10.
[8] Schwabacher, M., 2005, "A Survey of Data-Driven Prognostics," AIAA Paper No. 2005-7002.
[9] Byrne, G., Dornfeld, D., Inasaki, I., Ketteler, G., König, W., and Teti, R., 1995, "Tool Condition Monitoring (TCM)—The Status of Research and Industrial Application," CIRP Ann.-Manuf. Technol., **44**(2), pp. 541–567.
[10] Teti, R., Jemielniak, K., O'Donnell, G., and Dornfeld, D., 2010, "Advanced Monitoring of Machining Operations," CIRP Ann.-Manuf. Technol., **59**(2), pp. 717–739.
[11] Gao, R., Wang, L., Teti, R., Dornfeld, D., Kumara, S., Mori, M., and Helu, M., 2015, "Cloud-Enabled Prognosis for Manufacturing," CIRP Ann.-Manuf. Technol., **64**(2), pp. 749–772.
[12] Daigle, M. J., and Goebel, K., 2013, "Model-Based Prognostics With Concurrent Damage Progression Processes," IEEE Trans. Syst. Man Cybernetics: Syst., **43**(3), pp. 535–546.
[13] Si, X.-S., Wang, W., Hu, C.-H., Chen, M.-Y., and Zhou, D.-H., 2013, "A Wiener-Process-Based Degradation Model With a Recursive Filter Algorithm for Remaining Useful Life Estimation," Mech. Syst. Signal Process., **35**(1), pp. 219–237.
[14] Dong, M., and He, D., 2007, "Hidden Semi-Markov Model-Based Methodology for Multi-Sensor Equipment Health Diagnosis and Prognosis," Eur. J. Oper. Res., **178**(3), pp. 858–878.

[15] Saha, B., Goebel, K., and Christophersen, J., 2009, "Comparison of Prognostic Algorithms for Estimating Remaining Useful Life of Batteries," Trans. Inst. Meas. Control, **31**(3–4), pp. 293–308.

[16] Niaki, F. A., Michel, M., and Mears, L., 2016, "State of Health Monitoring in Machining: Extended Kalman Filter for Tool Wear Assessment in Turning of IN718 Hard-to-Machine Alloy," J. Manuf. Processes, **24**(Part 2), pp. 361–369.

[17] Orchard, M. E., and Vachtsevanos, G. J., 2009, "A Particle-Filtering Approach for On-Line Fault Diagnosis and Failure Prognosis," Trans. Inst. Meas. Control, **31**(3–4), pp. 221–246.

[18] Wang, P., and Gao, R. X., 2015, "Adaptive Resampling-Based Particle Filtering for Tool Life Prediction," J. Manuf. Syst., **37**(Part 2), pp. 528–534.

[19] Niaki, F. A., Ulutan, D., and Mears, L., 2015, "Stochastic Tool Wear Assessment in Milling Difficult to Machine Alloys," Int. J. Mechatronics Manuf. Syst., **8**(3–4), pp. 134–159.

[20] Wang, P., and Gao, R. X., 2016, "Stochastic Tool Wear Prediction for Sustainable Manufacturing," Proc. CIRP, **48**, pp. 236–241.

[21] Sick, B., 2002, "On-Line and Indirect Tool Wear Monitoring in Turning With Artificial Neural Networks: A Review of More Than a Decade of Research," Mech. Syst. Signal Process., **16**(4), pp. 487–546.

[22] Breiman, L., 2001, "Random Forests," Mach. Learn., **45**(1), pp. 5–32.

[23] Biau, G., 2012, "Analysis of a Random Forests Model," J. Mach. Learn. Res., **13**, pp. 1063–1095.

[24] Verikas, A., Gelzinis, A., and Bacauskiene, M., 2011, "Mining Data With Random Forests: A Survey and Results of New Tests," Pattern Recognit., **44**(2), pp. 330–349.

[25] Kamarthi, S., Kumara, S., and Cohen, P., 2000, "Flank Wear Estimation in Turning Through Wavelet Representation of Acoustic Emission Signals," ASME J. Manuf. Sci. Eng., **122**(1), pp. 12–19.

[26] Liang, S., and Dornfeld, D., 1989, "Tool Wear Detection Using Time Series Analysis of Acoustic Emission," J. Eng. Ind., **111**(3), pp. 199–205.

[27] Huang, Y., and Liang, S. Y., 2004, "Modeling of CBN Tool Flank Wear Progression in Finish Hard Turning," ASME J. Manuf. Sci. Eng., **126**(1), pp. 98–106.

[28] Taylor, F. W., 1907, *On the Art of Cutting Metals*, ASME, New York.

[29] Schwabacher, M., and Goebel, K., 2007, "A Survey of Artificial Intelligence for Prognostics," AAAI Fall Symposium, Arlington, VA, Nov. 9–11, pp. 107–114.

[30] Bukkapatnam, S. T., Lakhtakia, A., and Kumara, S. R., 1995, "Analysis of Sensor Signals Shows Turning on a Lathe Exhibits Low-Dimensional Chaos," Phys. Rev. E, **52**(3), p. 2375.

[31] Bukkapatnam, S. T., Kumara, S. R., and Lakhtakia, A., 2000, "Fractal Estimation of Flank Wear in Turning," ASME J. Dyn. Syst. Meas. Control, **122**(1), pp. 89–94.

[32] Bukkapatnam, S., Kumara, S., and Lakhtakia, A., 1999, "Analysis of Acoustic Emission Signals in Machining," ASME J. Manuf. Sci. Eng., **121**(4), pp. 568–576.

[33] Özel, T., and Karpat, Y., 2005, "Predictive Modeling of Surface Roughness and Tool Wear in Hard Turning Using Regression and Neural Networks," Int. J. Mach. Tools Manuf., **45**(4), pp. 467–479.

[34] Palanisamy, P., Rajendran, I., and Shanmugasundaram, S., 2008, "Prediction of Tool Wear Using Regression and ANN Models in End-Milling Operation," Int. J. Adv. Manuf. Technol., **37**(1–2), pp. 29–41.

[35] Sanjay, C., Neema, M., and Chin, C., 2005, "Modeling of Tool Wear in Drilling by Statistical Analysis and Artificial Neural Network," J. Mater. Process. Technol., **170**(3), pp. 494–500.

[36] Chungchoo, C., and Saini, D., 2002, "On-Line Tool Wear Estimation in CNC Turning Operations Using Fuzzy Neural Network Model," Int. J. Mach. Tools Manuf., **42**(1), pp. 29–40.

[37] Chen, J. C., and Chen, J. C., 2005, "An Artificial-Neural-Networks-Based In-Process Tool Wear Prediction System in Milling Operations," Int. J. Adv. Manuf. Technol., **25**(5–6), pp. 427–434.

[38] Paul, P. S., and Varadarajan, A., 2012, "A Multi-Sensor Fusion Model Based on Artificial Neural Network to Predict Tool Wear During Hard Turning," Proc. Inst. Mech. Eng., Part B, **226**(5), pp. 853–860.

[39] Karayel, D., 2009, "Prediction and Control of Surface Roughness in CNC Lathe Using Artificial Neural Network," J. Mater. Process. Technol., **209**(7), pp. 3125–3137.

[40] Cho, S., Asfour, S., Onar, A., and Kaundinya, N., 2005, "Tool Breakage Detection Using Support Vector Machine Learning in a Milling Process," Int. J. Mach. Tools Manuf., **45**(3), pp. 241–249.

[41] Benkedjouh, T., Medjaher, K., Zerhouni, N., and Rechak, S., 2015, "Health Assessment and Life Prediction of Cutting Tools Based on Support Vector Regression," J. Intell. Manuf., **26**(2), pp. 213–223.

[42] Shi, D., and Gindy, N. N., 2007, "Tool Wear Predictive Model Based on Least Squares Support Vector Machines," Mech. Syst. Signal Process., **21**(4), pp. 1799–1814.

[43] Jiaa, C. L., and Dornfeld, D. A., 1998, "A Self-Organizing Approach to the Prediction and Detection of Tool Wear," ISA Trans., **37**(4), pp. 239–255.

[44] Elangovan, M., Devasenapati, S. B., Sakthivel, N., and Ramachandran, K., 2011, "Evaluation of Expert System for Condition Monitoring of a Single Point Cutting Tool Using Principle Component Analysis and Decision Tree Algorithm," Expert Syst. Appl., **38**(4), pp. 4450–4459.

[45] Arisoy, Y. M., and Özel, T., 2015, "Machine Learning Based Predictive Modeling of Machining Induced Microhardness and Grain Size in Ti–6Al–4V Alloy," Mater. Manuf. Process., **30**(4), pp. 425–433.

[46] Cortes, C., and Vapnik, V., 1995, "Support-Vector Networks," Mach. Learn., **20**(3), pp. 273–297.

[47] Drucker, H., Burges, C. J., Kaufman, L., Smola, A., and Vapnik, V., 1997, "Support Vector Regression Machines," *Advances in Neural Information Processing Systems*, Vol. 9, pp. 155–161.

[48] Liaw, A., and Wiener, M., 2002, "Classification and Regression by Random Forest," R News, **2**(3), pp. 18–22.

[49] Friedman, J., Hastie, T., and Tibshirani, R., 2001, *The Elements of Statistical Learning, Springer Series in Statistics*, Springer, Berlin.

[50] Li, X., Lim, B., Zhou, J., Huang, S., Phua, S., Shaw, K., and Er, M., 2009, "Fuzzy Neural Network Modelling for Tool Wear Estimation in Dry Milling Operation," Annual Conference of the Prognostics and Health Management Society (PHM), San Diego, CA, Sept. 27–Oct. 1, pp. 1–11.