Machine-learning for automatic prediction of flatness deviation considering the wear of the face mill teeth

Andres Bustillo¹ · Danil Yu. Pimenov² · Mozammel Mia³ · Wojciech Kapłonek⁴

Received: 26 February 2019 / Accepted: 13 August 2020 / Published online: 3 September 2020
 © The Author(s) 2020

Abstract

The acceptance of the machined surfaces not only depends on roughness parameters but also in the flatness deviation (Δ_{fl}). Hence, before reaching the threshold of flatness deviation caused by the wear of the face mill, the tool inserts need to be changed to avoid the expected product rejection. As current CNC machines have the facility to track, in real-time, the main drive power, the present study utilizes this facility to predict the flatness deviation—with proper consideration to the amount of wear of cutting tool insert's edge. The prediction of deviation from flatness is evaluated as a regression and a classification problem, while different machine-learning techniques like Multilayer Perceptrons, Radial Basis Functions Networks, Decision Trees and Random Forest ensembles have been examined. Finally, Random Forest ensembles combined with Synthetic Minority Over-sampling Technique (SMOTE) balancing technique showed the highest performance when the flatness levels are discretized taking into account industrial requirements. The SMOTE balancing technique resulted in a very useful strategy to avoid the strong limitations that small experiment datasets produce in the accuracy of machine-learning models.

Keywords Face milling · Wear · Tool life · Tool condition monitoring · Flatness deviation · Cutting power · Random forest · SMOTE

List of symbols

CNC Computer numerical control
SMOTE Synthetic Minority Over-sampling Technique

✉ Danil Yu. Pimenov
 danil_u@rambler.ru

✉ Wojciech Kapłonek
 wojciech.kapłonek@tu.koszalin.pl

Andres Bustillo
 abustillo@ubu.es

Mozammel Mia
 m.mia19@imperial.ac.uk

- ¹ Department of Civil Engineering, Universidad de Burgos, Avda Cantabria s/n, 09006 Burgos, Spain
- ² Department of Automated Mechanical Engineering, South Ural State University, Lenin Prosp. 76, Chelyabinsk, Russia 454080
- ³ Department of Mechanical Engineering, Imperial College London, London SW7 2AZ, UK
- ⁴ Department of Production Engineering, Faculty of Mechanical Engineering, Koszalin University of Technology, Raławicka 15-16, 75-620 Koszalin, Poland

t	Tool life (processing time) (min)
T	Work cycle time (machining pass) (min)
a_p	Cutting depth (mm)
B	Cutting width (mm)
f_z	Feed per tooth (feed rate) (mm/tooth)
v_c	Cutting speed (m/min)
n	Spindle rotation speed per minute (rpm)
V_B	Tool flank wear value on the flank surface (flank wear) (mm)
P_c	Drive power primary motion (kW)
Δ_{fl}	Flatness deviation (μm)
χ_{YDeg}	Torsional angle of a machine tool-device-spindle ($^\circ$)
L	Workpiece length (mm)
B	Workpiece width (mm)
H	Workpiece height (mm)
D	Cutter diameter (mm)
γ	Rake angle ($^\circ$)
α	Back angle ($^\circ$)
λ	Angle of the main cutting edge ($^\circ$)
k_r	Major cutting edge angle ($^\circ$)
k_{r1}	Minor cutting edge angle ($^\circ$)

z	Number of teeth
k	Number of cycles in the experiment

Introduction

Artificial intelligence (AI) and its integration with computer integrated manufacturing systems have been at the forefront of manufacturing technology to cause the fourth industrial revolution (Zheng et al. 2018; Oztemel and Gursev 2020). Moreover, the advancement of data science has helped researchers to constitute even better prediction models for sensitive engineering applications such as precision manufacturing. Use of such technological integration in precision controlling of manufacturing outcomes becomes key features that industries are aiming for. Intricate and sophisticated features such as the machined surface profile i.e. roughness and flatness can be of major interest that can use the real-time power-based controlling (Xu et al. 2020). However, during machining, the flatness and surface roughness are highly influenced by the wear of the cutting tool (Kuram and Ozelik 2016; Machado and Diniz 2017). In this case, the following studies (Grzenda et al. 2012; Kovac et al. 2013; Simunovic et al. 2013; Huang et al. 2019a; Pimenov et al. 2018a) are known for modeling the roughness of face milling, including using prediction models by artificial intelligence. Grzenda et al. (2012) proposed a new strategy for improving the models of artificial intelligence used to predict the surface roughness of face milling using small data sets. Simunovic et al. (2013) presented studies of the surface roughness during face milling of an aluminum alloy. A regression model and a model based on the use of neural networks are obtained. Huang et al. (2019a) employed the Grey online modeling surface roughness monitoring (GOMSRM) system, as proposed in this study by utilizing the Grey theory GM(1, N) with bilateral best-fit method for milling operations. Kovac et al. (2013), in their study, proposed modeling the surface roughness obtained by face milling using fuzzy logic and regression analysis. Recently, Pimenov et al. (2018a) developed the surface roughness model based on AI for face milling. In their work, they monitored the augmented tool wear.

In literature, works on literature have concentrated on the study of flatness, surface roughness and other deviations; however, they did not consider the tool wear as a factor to influence the responses. As such, the appropriate compensation should be considered in modeling the flatness in face milling operation. Also, periodically, the worn cutting tool insert should be replaced—provided the maximum flatness deviation is yet to reach (Mikołajczyk et al. 2018; García-Ordás et al. 2018; Rodrigues et al. 2016). For instance, Nguyen et al. (2014) in their work studied the deliberate initial inclination of the cutter-spindle—but did not consider the tool wear. A model of shape deflection was proposed by

Denkena and Hasselberg (2015). They have not considered the tool wear too. Huang and Hoshi (2001) proposed efficient methods of reducing surface flatness error caused by the cutting heat in face milling. Badar et al. (2005) described a method of estimating flatness for determining the accuracy of the machined surfaces in face milling. Nadolny and Kapłonek (2014) presented an analysis of flatness deviations of a billet made of austenitic stainless steel X2CrNiMo17-12-2. Dobrzynski et al. (2018) in their study showed the influence of the path of the face mills on the deviation from the flatness of aluminum plates. Krishnaprasad et al. (2019) in their work studied the effect of various cutting parameters on output parameters, such as cutting force, surface roughness, flatness and material removal rate during milling of AISI 304 steel. However, some studies (Nguyen et al. 2014; Denkena and Hasselberg 2015; Huang and Hoshi 2001; Badar et al. 2005; Nadolny and Kapłonek 2014; Dobrzynski et al. 2018; Krishnaprasad et al. 2019) investigated flatness deviation without taking into account tool wear.

There are very few literature prediction models for surface flatness obtained by face milling, for example, among them (Gu et al. 1997a, b; Liu and Zou 2011; Yi et al. 2015). Gu et al. (1997a, b) showed in their work a model of plane prediction for face milling operations, taking into account cutting conditions, elastic deformations of the spindle milling cutter and workpiece. Liu and Zou (2011) predicted surface flatness errors in face milling using the traditional method. The work combines the empirical data and simulation results. Yi et al. (2015) in their work investigated and formulated a model for predicting errors in face milling using the finite element method. However, these works do not take into account the effect of tool wear on the flatness of processing, which is always present in such processes.

Studies (Davoudinejad et al. 2014; Rybicki and Kawalec 2010; Pimenov et al. 2011a, 2017, 2018b) are dedicated to machined surface deviation from flatness in face milling taking into account tool wear. Davoudinejad et al. (2014) considered critical issues such as the strategy of processing, spindle thermal transient management, and compensation for tool wear to improve the surface flatness. Rybicki and Kawalec (2010) compared the components of the deviations that occurred when changing the position of the wedge angle due to its wear, deformation of the machined system, as well as the inclination of the axis of the face mill. Pimenov et al. (2011a, 2017) presented a model that has taken into consideration axial elastic displacement of the subsystem “work piece—device—machine table” as well as elastic displacement of the subsystem “tool—device—z-head system”. Pimenov et al. (2018b) presented a model of torsional angle, χ_{YDeg} , and flatness, Δ_{fl} , with face milling and wear parameters. However, the use of this model in face milling operations requires monitoring the current amount of tool wear. This is often possible with the use of additional equip-

ment, which will not always be acceptable in production. However, even though studies (Davoudinejad et al. 2014; Rybicki and Kawalec 2010; Pimenov et al. 2011a, 2017, 2018b) focused on the deviation from flatness and accounted tool wear, they did not establish an integrated correlation of deviation from flatness, tool wear, and machining time for the purpose of controlling these parameters in automated production. However, now this is possible with artificial intelligence.

As the tool wear increases, the cutting force and power increase too (Huang et al. 2019b). Guzeev and Pimenov (2011) in their work showed a mathematical model of the force of face milling, taking into account machining parameters and tool wear. Pimenov and Guzeev (2017) and Pimenov et al. (2011b) in their works showed the influence of cutting conditions and wear on cutting force on the back surface of a tooth cutter. Promising for monitoring cutting operations is control of cutting power. Shnfir et al. (2019) investigated the influence of the mechanical milling parameters of AISI 1045 hardened steel, milling configuration, edge preparation and hardness of the processed material on machinability indicators, such as the resulting cutting force, energy consumption and wear of the side tool. Shao et al. (2004) described the cutting power model for face milling used in the strategy for updating the cutting power threshold for monitoring tool wear. Bhattacharyya et al. (2008) offered a method of continuous online assessment of tool wear during face milling based on inexpensive measurements of the current and voltage of a spindle motor. Da Silva et al. (2016) demonstrated the use of a probabilistic neural network in monitoring tool wear during milling through acoustic emission and cutting power signals. Pimenov (2015) offered a mathematical model of the power of the main drive, which can be used to control the milling conditions in the process of wear of the cutting tool. Many modern machines are equipped with such systems and can be used for online monitoring.

A very interesting prospect is to establish a complex relationship between such parameters as the deviation from flatness of surfaces obtained by face milling, the amount of tool wear and cutting power (Mori et al. 2011). The possibilities of artificial intelligence are very good for this. The prediction models created with its help can be integrated; for example, into the CNC system of the machine. This allows using these models in the course of preparation of production.

With such a complex model, it is possible to control the process of changing the deviation from flatness directly, controlling the degree of tool wear due to wear in size. In this case, the CNC machine should have a contact control sensor, for example, Renishaw. At the same time, wear control is performed between tool replacements. When tool wear reaches a point at which the deviation of the design surface from flatness cannot be achieved or tool wear values approach the maximum, forced replacement of the cutting tool is neces-

sary. In this case, the direct tool wear control method will be used.

However, not all machines are equipped with such sensors. Many CNC machines are equipped with feedback sensors that allow monitoring, for example, the drive power and, in particular, the power of the main drive. As soon as the correlation between tool wear and main drive power is established, the indicated deviation from flatness can be achieved by monitoring the current power value during processing. Whenever tool wear reaches a point at which a deviation from flatness cannot be achieved or tool wear values approach the maximum, forced replacement of the cutting tool is necessary. In this case, an indirect method of monitoring tool wear with indications of the power of the main drive will be used.

The first two options cannot be realized with the help of universal manual machines. However, by formulating appropriate relation of wear of tool with tool life may help to monitor the tool life, which in turn may result in flatness deviation. Whenever the current machining time reaches a level point at which the deviation from flatness cannot be reached or the time value approaches the maximum, forced replacement of the cutting tool is necessary. In this case, an indirect tool wear control method with the current machining time will be used.

Although the 1st and 3rd methods are noted to be used during pause mode, it is not possible to replace tools in cases of tool failure in real time. The second method does not have this disadvantage. Power changes can be monitored in real time. The above disadvantages can be bypassed by hybridizing the first method with the second or second method with the third.

However, to implement the above methods, one must have an appropriate model where the deviation of surface flatness is integrated with power of main drive provided the tool wear is taken into account. Although the experimental or analytical solutions of the surface deviation from flatness in face milling have been studied extensively, to the best of authors knowledge, no machine-learning techniques have been proposed to solve this industrial problem. Machine-learning techniques have a clear advantage from analytical models: they can provide reliable models under noisy real industrial conditions. Under laboratory conditions, a very few number of inputs are varied from one experiment to the next; there is almost no experiment repetition, the experiment conditions are carefully selected, mainly by factorial or Taguchi experiment design and any input and output are carefully measured and validated before the next experiment is performed (Benardos and Vosniakos 2003). However, under industrial conditions, most of the experiments refer to the same cutting conditions—a very broad range of parameters are varied at the same time and the data presents many empty values or very noisy values due to limited sensor's quality or sensor's location (Grzenda and Bustillo 2019). Machine-learning techniques

are especially suitable to overcome these limitations (Park and Kim 1998; Mohanraj et al. 2020). Machine-learning techniques have many capabilities: they generalize models to new conditions, reducing in this way the number of expensive experimental tests to be performed. They can extract useful information for imbalanced datasets (Bustillo and Rodriguez 2014) and reduce the number of features without losing information (Grzenda et al. 2012), while they assure the confidence of the measured data (Grzenda and Bustillo 2019). These techniques have proved its ability to create models as accurate as other traditional approaches, but more flexible for many manufacturing processes (Zhu et al. 2009; Wu et al. 2017).

These kinds of models have been, up to now, successfully used to model many cutting processes and their main outputs: from surface roughness in milling (Zain et al. 2010), drilling (Sanjay and Jyothi 2006; Bustillo and Correa 2012) and turning (Benardos and Vosniakos 2003) to cutting tool's wear in those processes (Sick 2002; Abellan-Nebot and Romero Subirón 2010). Nonetheless, the challenge is still open to validate their capabilities in the case of geometrical errors prediction, especially with small and noisy datasets.

In the light of the above literature facts, the objective of current study is to develop prediction models using machine learning methods for flatness deviation. Attempts on developing such models are taken due to the complex prevailing relationship among the tool wear, its role on altering the flatness, time of operation, and the cutting performances. It is expected that this model can be of interest to monitoring the real time flatness in automated production systems.

Materials and methods

Before making the prediction we have performed experimental study on face milling. Here, an attempt was made to make a correlation with the cutting power to the flatness deviation, tool surface wear and machining time (tool life). The work material was AISI 1045 steel that had a length of 200 mm and width of 75 mm and height of 100 mm. The used machine was a 5-axis CNC machining center Mori Seiki NMV 5000 (Mori Seki, Nagoya, Japan) as visualized in Fig. 1. The chemical composition of the used material i.e. 45 steel (analog AISI 1045 steel) is listed in Table 1. Using a Brinell intender TB 5004-03 the work hardness was established at HB190.

The actual machining runs were conducted in cooling condition. Here, the used tool inserts were Pramet 8230 (see the properties in Table 2). The tool manufacturer recommends this type of tool. Moreover, the employed levels of cutting parameters are shown in Table 3.

The data from the Mori Seiko NMV 5000, as in the form of online reading, was recorded for the main power drive. It is to be noted that the readings were taken in both the maximum

Fig. 1 General view of workspace of a 5-axis vertical machining center Mori Seki NMV 5000

value and minimum value, captured at the starting of the milling operation, middle of operation and end of the milling cycle. This is taken as in the form of percentage, where the total machine power is 22 kW. Here, $k = 6$ indicates that for each experiment the total cycle is 6.

To understand the extent of tool wear, especially at the flank surface, its macrograph was taken for each cycle. Afterward, it was processed to measure the width of flank wear of the mill tooth. This was accomplished by taking the comparison of wear with respect to dimensional ruler as shown in Fig. 2.

The measurement of deviation from flatness Δ_f was carried out on the basis of processed Renishaw (Ultra compact sensor OMP-40) touch probe readings. The readings were recorded for nine points: basic point 1, points 6 and 7 in the beginning, middle (points 2, 5 and 8) and the end (points 3, 4 and 9) of the working stroke of the mill. Here, points 3, 4, 9 and 1, 6, 7 are located at 90 mm distance from the part center; points 7, 8, 9 and 4, 5, 6 are located at 30 mm distance from the part center (see Fig. 3). The relation $t = L/f_z n$ divulged the tool life (t) for a pass of cutting. Here, L indicates the length of the workpiece, f_z shows the feed rate and n stands for the rotational speed of the spindle.

Experiments

From the above presented measurements and computations, the values of main drive power, the flank wear width, the tool life, and flatness deviation are found and presented in Table 4.

Table 1 Properties of AISI 1045 steel—chemical composition and hardness

Workpiece material	Weight percentage										Brinell hardness, HB
	C	Si	Mn	Ni	S	P	Cr	Cu	As	Fe	
ANSI 1045	0.42–0.5	0.17–0.37	0.5–0.8	to 0.25	to 0.04	to 0.035	to 0.25	to 0.25	to 0.08	≈ 97	190

Table 2 Parameters of tools

Material of the tool cutting part	Cutter diameter, D , mm	Rake angle, γ , °	Back angle, α , °	Angle of the main cutting edge, λ , °	Major cutting edge angle, k_r , °	Minor cutting edge angle, k_{r1} , °	Number of teeth, z
Coated carbide, Pramet Company (Chesterfield, UK)	100	20	8	− 5	45	45	1

Table 3 Specifications of cutting conditions

Parameters	Magnitudes
Cutting depth	1.0 mm
Cutting speed	188.5 m/min
Feed per tooth	0.2 mm
Rotation of the spindle	600 rpm
Workpiece length	200 mm
Tool life	1.67 min
Work cycle time	2.58 min

Fig. 2 Visualization of the wear of the flank surface of mill insert

Modeling

Dataset description

The data collected in Table 4 was used to generate the first dataset used in the modeling stage. This dataset includes three

Fig. 3 Scheme of control deviation from flatness Δ_f

inputs: the tool life (t), the average drive power (P_c) and the flank wear (V_B). The output is the measured deviations of flatness (Δ_f) of the milled surface. All the three inputs and the output are continuous numeric variables, while Table 5 summarizes their units, and their variation's range in the dataset; the output is highlighted in bold. As the tool's wear was measured in 35 different tool lives, the dataset is composed of 35 instances. Finally, it is interesting to remark that the output presents a high dispersion in its values, around 27% of its variation range in terms of the standard deviation: 14.0 μm (with a variation range of 52 μm). This high dispersion might be due to the dispersive behavior of the wear, one of the input parameters. The tool's wear can be treated as a very-noisy input for two reasons. Firstly, the tool's degradation is a non-linear process with strong changes depending on its remaining lifetime (Mohanraj et al. 2020). Secondly, it depends on many parameters that cannot be controlled (previous cutting conditions, microstructure of the tool and the workpiece, vibrations, etc.) under real industrial conditions, where the machines are not completely sensorized (Sick 2002). Therefore, the high dispersion of the deviations of flatness may be a challenge for machine-

Table 4 Results of the experiments: main drive power P_c , tool life t , flank wear V_B , deviation from flatness Δ_{fl}

	Drive power primary motion, P_c , kW	Tool life, t , min	Flank wear, V_B , mm	Deviation from flatness, Δ_{fl} , μm
0	1.25	0	0	8
1	1.39	1.67	0.08	10
2	1.43	3.34	0.16	12
3	1.39	5.01	0.33	10
4	1.5	6.68	0.39	8
5	1.47	8.35	0.39	8
6	1.32	10.02	0.46	8
7	1.5	11.69	0.48	12
8	1.43	13.36	0.52	10
9	1.65	15.03	0.52	10
10	1.58	16.7	0.54	8
11	1.72	18.37	0.55	10
12	1.69	20.04	0.57	12
13	1.76	21.71	0.57	10
14	1.76	23.38	0.58	12
15	1.8	25.05	0.58	12
16	1.72	26.72	0.58	12
17	1.87	28.39	0.58	10
18	1.8	30.06	0.59	10
19	1.8	31.73	0.59	10
20	1.94	33.4	0.59	10
21	2.09	35.07	0.6	14
22	2.24	36.74	0.61	18
23	2.31	38.41	0.62	26
24	2.16	40.08	0.64	26
25	2.42	41.75	0.64	24
26	2.68	43.42	0.65	24
27	2.68	45.09	0.69	28
28	2.64	46.76	0.69	40
29	2.9	48.43	0.7	28
30	2.93	50.1	0.7	24
31	2.86	51.77	0.7	32
32	2.93	53.44	0.71	54
33	3.15	55.11	0.73	52
34	3.26	56.78	0.76	60

learning models, because it can reduce their accuracy and the machine-learning techniques that can handle better the noisy nature of the tool's wear can be able to provide high accurate models.

A first analysis of the correlation between inputs at outputs was done before any machine-learning technique is applied to the dataset. Figure 4 shows the correlation of the three inputs with the output Δ_{fl} . As it would be expected if the inputs are correctly selected, it is possible to see that the three inputs have influence on the output and a pattern might be extracted from these figures. In both cases, higher values of power, wear and tool life increase the output Δ_{fl} once a certain threshold

Table 5 Input variable to the machining process and modeling along with their range and unit

Input variable	Abbreviation	Range	Units
Tool life	t	0–57.9	min
Flank wear	V_B	0–0.76	mm
Drive power	P_c	1.21–3.74	KW
Deviation from flatness	Δ_{fl}	8–60	μm

is reached. However, this pattern is noisy due to two possible reasons: (1) noise in the experimental data and (2) crossed

Fig. 4 Relationship between inputs (V_B , P_c and t) and the deviation from flatness Δ_{fl}

influence of the different inputs in the output. Both reasons can be overcome using machine-learning techniques.

Finally, it is necessary to consider the level of imbalance of the dataset. Obviously it is possible to build prediction models without taking into account this fact, but they may result with a low accuracy or, what is still worse: with a false high accuracy because the model only fits properly to the high-populated range of inputs and the output and perform very badly in the low-populated ranges. In this dataset, there is a strong imbalance for the output, Δ_{fl} . If the total variation range of the deviation of flatness Δ_{fl} is divided in 3 intervals of the same size, there are 26 instances in the lower interval [8–25.3], 6 instances in the middle interval [25.4–42.6] and 3 instances in the higher interval [42.7–60]. This fact in the dataset is very common in industrial datasets related to wear or breakage processes because the tools are designed to work without losing capabilities for a long time and then they decrease their performance very quickly (Grzenda et al. 2012; Bustillo and Rodriguez 2014). Therefore, if the whole tool's lifetime is considered, the number of measurements under bad performance conditions will be very low compared with the number of measurements under proper conditions.

One of the possible strategies to improve the accuracy of prediction models is the output discretization. This solution is only possible if the output is classified in different levels under industrial conditions, as it happens with surface roughness (Bustillo et al. 2011) or geometrical errors prediction (Maudes et al. 2017) for milling operations. In one of the modeling stages presented in Sect. "Results and discussion", discretization of the dataset is done.

The tolerance values of the flatness deviation were compared by following the GOST 24643-81 standard (Russian standard). For this standard, the accuracy of the degree and respective flatness tolerance is listed in Table 6. It is to be noted that the verification of the tolerance is done for specific nominal dimensions, and for our study the workpiece was of 200 mm, hence the range 160–250 mm is considered in Table 6.

After this discretization, the dataset is still strongly imbalanced, as might be expected because discretization should not change the nature of the dataset. There are 5 instances for level 6 (14%), 16 instances for level 7 (46%), 2 instances for level 8 (6%), 7 instances for level 9 (20%), 2 instances

Table 6 The values of flatness tolerance with respect to the accuracy of the degree, μm

Interval of nominal dimension, mm	Accuracy of the degree							
	4	5	6	7	8	9	10	11
	Flatness tolerance, μm							
160...250	5	8	12	20	30	50	80	120

for level 10 (6%) and 3 instances for level 11 (8.5%) for the deviation of flatness.

Cross validation

As previously presented, there are two possible strategies to predict geometrical deviation of a milled surface. If the surface deviations are considered continuous outputs, the prediction problem is called regression from the machine-learning perspective and the suitable techniques for this kind of problems are known as regressors. But, if the surface deviations are split in a discrete number of ordered levels, the prediction problem is called classification and the suitable techniques for this kind of problems are known as classifiers. In both cases the prediction models are trained with a dataset in a first stage, and then tested with another dataset to evaluate their accuracy and their generalization capability.

The way the dataset is used to train and test the prediction model plays a main role in its performance and can provide very different model's accuracy if the training data are considered in the validation stage (Teixidor et al. 2015). There are different ways to divide the dataset between training and test subsets: the leave-one-out technique (Hall et al. 2009), 5×2 cross validation (Kohavi 1995) and the 10×10 cross validation scheme (Kohavi 1995). Different authors have pointed out that the 10×10 cross validation scheme is the most suitable strategy for datasets of the size of the presented in this research (Sick 2002; Abellan-Nebot and Romero Subirón 2010; Teixidor et al. 2015; Maudes et al. 2017; Oleaga et al. 2018), while the leave-one-out technique is specially thought for smaller datasets (Maudes et al. 2017) and the 5×2 cross validation scheme can be the best solution for datasets of bigger size (Santos et al. 2018). In the 10×10 cross validation scheme, the dataset is randomly divided into 10 subsets of the same size. Among them, nine subsets are used to train the prediction model leaving the rest one to evaluate its performance. This process is repeated 9 times until all the 10 subsets have been used for the validation stage. This whole process is repeated 9 times more with different random divisions of the dataset in 10 subsets. In this way the accuracy 100 prediction models are built and the final accuracy is the mean value of the 100 models' accuracy (Hall et al. 2009), assuring a statistical quality of the results and their independence of the random division of the dataset for training and

validation tasks (Kohavi 1995). Besides, as all the prediction models are tested with instances that have not been used in the training stage, it was also considered that the prediction model has capability to deal with new instances.

SMOTE and metrics for imbalanced datasets

Although 10×10 cross validation is the most suitable technique to identify the most accurate machine-learning technique in this dataset due to its size (Teixidor et al. 2015; Maudes et al. 2017; Oleaga et al. 2018), it presents a serious limitation if the dataset is strongly imbalanced: any prediction model will try to adapt itself to predict better the most populated class or output range, depending if we deal with a classification or a regression problem. Therefore, the low-populated classes will be very badly predicted. Two different strategies have been developed to solve this limitation: (1) the use of techniques that balance the dataset (adding new instances of the low-populated classes or delete instances of the most populated classes) and (2) the use of metrics to evaluate the model's performance that weight the performance of the different classes or value's range to avoid the effect of the imbalance in the dataset.

The artificial balance of the dataset could only be done in one way due to the small size of the dataset: creating new instances of the low-populated classes. This is the most common strategy in industrial problems with small or medium datasets (Bustillo and Rodriguez 2014), like the one presented in this research, while the reduction of instances is mainly done with Big Data problems, like failure detection, due to the huge amount of instances of the majority class, e.g. non-failure class, in such datasets (Leevy et al. 2018; Santos et al. 2018). Synthetic Minority Over-sampling Technique (SMOTE) (Chawla et al. 2002; Fernandez et al. 2018) is a well-established method for creating new synthetic examples in the minority class in a dataset. It considers one instance of the minority class and it selects randomly one of its k nearest neighbors, where k is a parameter of the SMOTE algorithm. The new instance is created in a random point of the segment that joins the selected neighbor and the original minority-class instance. Figure 5 shows the flow chart for the SMOTE algorithm (in order to simplify the description, the pseudo-code is formulated to duplicate the number of instances in the minority class, and therefore the number of artificial instances to be generated is the same than the number of minority instances in the original data-set). It is important to highlight that SMOTE in this implementation only increase the number of instances in one class (the lowest-populated one) by duplicating it. Therefore, if many classes present a low number of instances the SMOTE algorithm should be used in more than one iteration and more than one iteration should be done on the same class if the dupli-

Fig. 5 Flow chart for the SMOTE algorithm

cation of the number of instances is not enough to balance the dataset.

The use of different metrics to evaluate the model's performance is always an open issue in machine-learning techniques and should be always oriented to the industrial use of the prediction models (Benardos and Vosniakos 2003; Abellan-Nebot and Romero Subirón 2010). For example, in the industrial detection of failures of a certain machine, a very-high accuracy of a classification model-in terms of well-classified instances in the validation subset-could be a disaster if the model is not able to predict the failure-class, because it is optimized for a training dataset where very few instances of the failure class are included. Then, the standard metrics for surface errors in milled parts, both for regression or classification problems, they will be of limited interest in this research. These indicators are mainly the Root Mean Square Error (RMSE) and the Mean Absolute Error (MAE) in the case of a continuous output. Different authors have chosen RMSE (García-Ordás et al. 2018), while others preferred MAE (Grzenda and Bustillo 2019) depending on their industrial objectives. MAE has the advantage of providing a physical measure of the variable's error while RMSE penalizes the models by considering worn predictions. One such wrong prediction can be the consideration of the square of the errors instead of the errors themselves. This fact might be of special interest because there are areas in the dataset that

are not enough populated (e.g. the high-wear area) due to the imbalance nature of the dataset; therefore, high errors might appear in the prediction of such areas. For discrete outputs, the accuracy (number of well-classified instances divided but the total amount of instances in the dataset) is selected.

However, to allow a deeper discussion of the performance of the machine-learning models, some special metrics for imbalanced datasets in the classification case were also taken into account (Hossin and Sulaiman 2015). These metrics are defined for a 2-classes classification problem, while their extension to a multiclass problem is done by weighting their value for each class and calculating the average of the weighted values for each class (Stehman 1997) (e.g. in such a way that the weighted values sum 100% for accuracy). If the class to be predicted is considered positive while the other classes are considered negative, then the prediction can be true or false: hence, we have true positives (TP), true negatives (TN), false positives (FP) and false negatives (FN). For instance, a false positive is an example of any of the negative classes that is predicted as positive. Combining these four categories, the following performance indicators can be defined (Witten et al. 2016):

$$Accuracy = (TP + TN) / (TP + TN + FP + FN)$$

$$TP\ rate = TP / (TP + FN)$$

$$FP\ rate = FP / (TN + FP)$$

$$Recall = TP\ rate$$

$$Precision = TP / (TP + FP)$$

$$F\text{-measure} = (2 \cdot Recall \cdot Precision) / (Recall + Precision)$$

Considering these definitions, small values for FP rate will indicate good performance of the prediction model, while high values are desired for the other quality indicators. The F-measure is a more complex measure and combines recall and precision. In the flatness deviation case, recall is the proportion of milled pieces correctly classified in its class while the precision is the proportion of milled pieces correctly classified in its class related to the total number of milled pieces classified in each class. Finally, the area under the ROC curve (AUC) (Fawcett 2006) is also evaluated. Many classifiers associate a confidence level (e.g. a probability) with their predictions and therefore, some thresholds can be defined to discriminate between two classes (or between one class and the rest of the classes). The FP rate and TP rate are used for each threshold. Here, the ROC curve's shape is determined by the different threshold. Reportedly, the classifier performance is indicated by the area under this curve.

Machine-learning prediction models

Weka machine-learning software has been used for testing two regressors families (Hall et al. 2009): artificial neural networks (ANNs) and regression trees due to their different

nature: while multilayer perceptrons are the most standard black-box technique (Benardos and Vosniakos 2003), regression trees can generate visual and interpretable models for the process engineer (Bustillo et al. 2016). In the case of classification problems their variation: ANNs for classification and decision trees have been used. Besides, other two methods were considered for experimentation as baseline methods: the Naïve approach and linear regression or a logistic classifier depending on the consideration of the deviation from flatness as a continuous or discrete output respectively. A linear regressor was considered because a linear fit is a very-spread fitting method in industries while it behaves very well for small variation of the input's ranges. On the other hand, the logistic classifier contributes with non-linear capabilities to model the flatness deviation.

Artificial neural networks is a very extensive family of prediction models. The most common typologies of ANNs are Multilayer Perceptrons (MLPs) and Radial-Basis Functions Networks (RBFs). Their strong capability to predict complex multivariable processes have converted them in standards in the prediction of different machining processes outputs like surface roughness (Samanta et al. 2008; Markopoulos et al. 2008; Grzenda and Bustillo 2019), cutting tool's wear (García-Ordás et al. 2017; D'Addona et al. 2017) or surface wear based on surface isotropy levels (Bustillo et al. 2018). ANNs are composed of, at least, three different layers, each of them formed by neurons or computational nodes (Yegnanarayana 2009). The first layer is called the input layer and includes a neuron for each input parameter of the process. The hidden layer could include different numbers of neurons, while each of them is connected with all the neurons of the input layer and produce its output as a weighted combination of the values of the input neurons. Finally, the output layer includes only one neuron for a regression problem with one output or one neuron for each possible class of the output in a classification task. The difference between MLPs and RBFs is located in the way the weight for each connection of the ANN is calculated. It is to be mentioned that the back propagation is used by MLPs (Yegnanarayana 2009). On the other hand, for RBFs, the radial basis function is used by RBFs (Leonard and Kramer 1991).

The nature of the ANNs makes them black-box models because it is not possible to extract direct or visual knowledge from them. Only if a certain combination of inputs is given to a trained ANN, the ANN will provide a prediction of the output. Then, the final user can use this prediction to take a decision about the cutting conditions to be tested. Therefore, 3D or 2D graphs of the predicted output in terms of different input conditions have to be built to extract useful information (Bustillo et al. 2016). As opposed to this limitation of ANNs, decision trees provide visual models of special interest for end-users (Bustillo et al. 2016). A decision tree is read from top (the root node) to bottom (the final leaf). Each decision

node opens different branches considering the value of a certain input of the process or attribute. When a final leaf is reached, it will provide a class value in the case of a decision tree (for a classification task) and a linear regression model for the output in the case of a regression tree (for a regression task). In each decision note, the attributes and their numeric thresholds are selected using the M5P algorithm (Quinlan 1996). It causes subdivision of the training data into subsets.

But the accuracy of simple regressors can be improved if ensembles of regressors are used to build prediction models. Ensembles are combinations of base regressors that combine their individual predictions in a final averaged prediction (Kuncheva 2004; Mendes-Moreira et al. 2012). In this study, Random Trees are used as base regressors to build an ensemble called Random Forest. One of the cases of M5P trees is random trees, which are effectively used when choosing solutions at each node (Witten et al. 2016), a random forest is a set of different random trees (Breiman 2001). Random forest is effective, for example, in predicting geometric errors (Maudes et al. 2017) or in prediction of chatter (Oleaga et al. 2018).

Finally, some regressors have been considered as baseline methods to allow comparison. Usually the naïve approach and a regression or logistic model are good candidates for such task (Bustillo et al. 2011; Maudes et al. 2017; Oleaga et al. 2018). In classification problems, the naïve approach always predicts the class of the most-populated class (e.g. if the most-populated class is Class 4, the naïve approach will always predict Class 4 independent of the values of the inputs); for regression problems the naïve approach predicts always the mean value of the output. Other statistical approaches couldn't be considered as baseline methods, due to the reduced size of the experimental dataset. The linear regression model is a simple but very common method to model the input–output relationship in very extensive areas of their variation range. Consequently, any machine-learning model should perform significantly better than these baseline methods if non-linear relationships and noise effects are present in the machining process under study. The selected regression methods are SimpleLogistic for classification and SimpleLinearRegression for regression.

After the selection of the machine-learning techniques to be tested on the dataset, it is necessary to perform a tuning process of their parameters to find their optimal values to achieve their best performance. In the case of decision trees and Random Forest ensembles, no parameters were tuned because the bibliography has proven their stability (Bustillo et al. 2011; Maudes et al. 2017; Oleaga et al. 2018) in similar manufacturing problems. For the other machine-learning techniques, the following parameters were adjusted:

- Learning Rate, Momentum, Hidden layer neurons for MLPs

Table 7 Tuning parameters of the MLPs and RBFs

Network type	Parameter	Range	Tested values in the first search
MLP	Momentum (M)	0.05–0.5	0.05, 0.1, 0.5
MLP	Learning rate (L)	0.01–0.5	0.01, 0.05, 0.1, 0.5
MLP	Hidden layer neurons (H)	2–4	2, 3, 4
RBF	Number of radial basis functions	2–4	2, 3, 4
RBF	Ridge	0.00005–0.05	0.00005, 0.0001, 0.0005, 0.001, 0.005, 0.01, 0.05
RBF	Tolerance	0.0001–0.01	0.0001, 0.0005, 0.001, 0.005, 0.01

- Number of radial basis functions, ridge and tolerance for RBFs

The tuning process was performed following a two-stages grid search scheme. Table 7 summarizes the tuning range and the tested values of the tuned parameters in the first tuning stage. In a grid search scheme, all the considered values for one certain parameter (e.g. the momentum for an MLP) is tested against the other parameters possible values (e.g. learning rate and number of neurons for MLPs). In the first stage, to tune the considered parameter's range, 4 values were selected for the learning rate and 3 values for the momentum and the number of hidden layers. The selected values are collected in Table 7. With this strategy, a limited number of tested MLPs ($4 \times 3 \times 3 = 36$) were built covering the whole variation range. However, this strategy will not assure that the most accurate combination of parameter's values is found if the machine-learning techniques are very sensitive to any of those parameters. Therefore, after this first search, a second search is performed around the best parameters' values with smaller variation steps. In this second stage, Momentum and Learning Rate take 4 different values around the best value identified in the first search, two of them are higher and two of them are lower (in steps of 10% of the parameter's value). The number of hidden layers was not varied because it presents a lower sensitivity. For example, if the best combination in the first stage was: $M = 0.1$, $L = 0.05$, $H = 3$, 25 MLPs were built (5×5) in a new grid search with values for the Momentum of 0.08, 0.09, 0.1, 0.11 and 0.12 and Learning Rate of 0.04, 0.045, 0.5, 0.55 and 0.6 all of them with 3 hidden layers. The prediction model with the lower RMSE error represents the best model. Therefore, this tuning process requires the construction of more than a hundred of prediction models, a fact that can be done due to the small size of the dataset because the execution time is not significant.

With this search strategy and parameters variation, every tuning process of MLPs requires the construction of 60 combinations (36 in the general search and 24 in the fine search) while the RBFs requires the construction of 129 combinations (105 in the general search and 24 in the fine search). Besides, under the 10×10 cross validation scheme, each parameter's combination means the construction of 100 pre-

Table 8 Accuracy of the prediction models for flatness deviations as a continuous output (RMSE, μm | MAE, μm | time, s)

Methods	RMSE (μm)	MAE (μm)	Time (s)
ZeroR	14.6447	11.7	0
Linear regression	6.8662	5.3545	0
M5P Tree	6.1842	4.3463	0.17
MLP	5.8207	3.9568	0.2
RBF	4.8761	3.4486	0.05
Random Forest	6.8106	4.0605	0.05

diction models. Therefore, such an extensive search, although optimized, cannot be so easily considered if huge datasets are available.

Results and discussion

Regression modelling

Firstly, the considered regressors and baseline methods were tested on the dataset considering the flatness deviation as a continuous output (original dataset). Table 8 shows the results of the considered quality indicators (RMSE and MAE) and the computational time for each tested regressor while the best models are outlined in bold. The computational time has been calculated using an Intel Core i7 6700 3.4-GHz processor with 16 GB RAM and a NVIDIA Titan Xp GPU. The best MLP configuration is found for: a learning rate of 0.04, a momentum of 0.05 and hidden layer neurons of 2. The best RBF configuration is found with 3 functions, a ridge value of 0.011 and a tolerance of 0.0004.

The quality indicators shown in Table 8 disclose the superiority of the machine-learning based techniques than the baseline methods. The corrected resampled t-test (Nadeau and Bengio 2003) with a significance level of 5% is considered to calculate the statistically significant differences. This test is used as a standard to compare the performance of two machine-learning models (Kuncheva 2004). It computes the difference between both models' performances and compares it to a certain significance level (usually 0.05 or 5%).

Fig. 6 Effect in the model's accuracy of the tuning parameters of **a** MLPs and **b** RBFs

If the difference is smaller than the significance level, it is accepted that there is a significant difference in the performance of both models (Nadeau and Bengio 2003). The RBFs perform 17–12% better than MLPs depending on the considered quality indicator. However, although RBFs perform better than MLPs, their best configuration is very unstable perhaps due to the small size of the dataset. Figure 6 shows the effect in the model's accuracy (RMSE) of the tuning parameters of the networks. X-axis is the Learning Rate (MLPs) and the Ridge (RBFs) while Y-axis is the Momentum (MLPs) and the Tolerance (RBFs) respectively. It can be quickly concluded from Fig. 6 that while MLPs have a broad area of stability in their tuning parameters, especially Momentum that plays a minor role, RBFs present a very unstable behavior and any small change in any of their two main tuning parameters, tolerance or ridge, can change significantly the model's accuracy.

Regression trees are less precise than ANNs although they have the advantage that no-tuning process in their parameters is required to build the most-accurate model. Taking into account the standard deviation of the flatness deviation ($14.01 \mu\text{m}$), the RMSE of the best model is still too high ($4.88 \mu\text{m}$) than knowledge available in literature (Bustillo et al. 2011; Maudes et al. 2017; Oleaga et al. 2018) and the relationships between inputs and output observed in Fig. 4. Two reasons might be responsible for such poor behavior of all the tested machine-learning algorithms: the small size of the dataset, 35 instances, and its imbalance. As Fig. 4 shows, there are very few instances for high values of wear, power consumption or tool life; this fact limits the accuracy of the prediction models under a 10×10 cross validation scheme. To verify this hypothesis, Fig. 7 has been plotted. It represents in X-axis the measured flatness deviation and in Y-axis the predicted flatness deviation by the RBF model (the most accurate); besides, it shows the error in the prediction for each instance in the dataset by

Fig. 7 Comparison of the real flatness deviations with the predicted ones by the RBF model

changing proportionately the cross's size. In this figure, high errors are located in the area of high flatness deviation, where a very limited number of instances are found. Although an academic solution for such a problem might be the extension of the experimental tests to get a larger dataset, this solution is not acceptable from an industrial point of view, where every new experiment means non-acceptable expenses.

If the computational time of the machine-learning models is taken into account, RBF model are the quickest machine-learning model as Table 8 summarizes, although there is a remark to be done: it is necessary to build many RBF to find the optimal values of its parameters, as it also happens with MLPs. During this stage of the research, it was necessary to test 129 combinations of RBF parameters and 60 combinations of MLPs to find the optimal values of these models using the grid search described in Sect. "Machine-learning prediction models". As opposed to this fact, regression trees and random forest ensembles require no tuning process at all

Table 9 Accuracy of the prediction models for flatness deviations as a discretized output

	Accuracy, %	TP rate	FP rate	Precision	Recall	F-measure	ROC Area
ZeroR	45.7	0.457	0.457	0.209	0.457	0.287	0.258
Logistic classifier	54.29	0.543	0.215	0.494	0.543	0.498	0.853
M5P Tree	62.86	0.629	0.114	0.643	0.629	0.631	0.837
MLP	62.86	0.629	0.188	0.478	0.629	0.538	0.826
RBF	65.71	0.657	0.180	0.504	0.657	0.569	0.858
Random Forest	65.71	0.657	0.113	0.638	0.657	0.641	0.879

and therefore only one model is built. Although the computational time is not a major issue in manufacturing problems with small datasets like the case presented in this research, might play a main role in other manufacturing problems like failure detection in online monitoring systems. Therefore, under similar accuracy levels, Random Forest or Regression trees may be preferred than ANNs models. This result might be extensible to the next section.

Discretized imbalanced modelling

After the limited success of these machine-learning models, the strategy should be changed to take advantage of the industrial point of view for this industrial problem. As already presented in Sect. "Dataset description", the accuracy of prediction models can be improved if the outputs discretization in different levels of industrial interest. The range and levels used to discretize the flatness deviation have been presented in Sect. "Dataset description". Table 9 collects the accuracy of the machine-learning models and the baseline methods, marking in bold the best results. In this case, as presented previously in Sect. "SMOTE and metrics for imbalanced datasets", the quality indicators of the prediction models are not limited to the accuracy but to other metrics of interest in the case of imbalanced dataset like TP rate, FP rate, Recall, Precision, F-measure and ROC Area. The most accurate RBFs are revealed at: 3 functions, 0.01 ridge and 0.001 tolerance. The most accurate MLPs are found at: 2 neurons, 0.3 learning rate and 0.2 momentum.

As in the continuous case, all the machine-learning methods perform clearly better than the baseline methods. Again, in this case RBFs perform better than MLPs. However, discretization improves the behavior of decision trees and therefore Random Forest ensembles are able to perform slightly better than RBFs, especially in the metrics directly thought for imbalanced datasets.

Discretized balanced modelling

Although the results presented in Table 9 are clearly better than the results obtained considering flatness deviation as a continuous output, they should be improved before the model's prediction may be reliable enough for their industrial

Fig. 8 Number of instances in each class in the different datasets

use. Therefore, the use of the SMOTE balancing technique is proposed. SMOTE has been run 7 times on the dataset, and 3 datasets have been generated: 3-steps of SMOTE, 5-steps of SMOTE and 7-steps of SMOTE. Taking into account the SMOTE implementation in Weka, the 35-instances original dataset grows to 42 instances in the 3-step dataset, to 50 instances in the 5-steps dataset and to 59 instances in the 7-steps dataset. This means that the biggest dataset, the 7-steps dataset, includes 68.5% of artificial instances, a limit that should not be increased because the final dataset would be too artificial. With these 3 levels, the low populated class has 25% of the instances than the most populated class (3-steps dataset), 31.5% (5-steps dataset) and 50% (7-steps). Although 50% relationship between the less and the most populated class cannot be considered a balanced dataset, the situation is far away better than in the original dataset. Figure 8 shows the growth of the less-populated classes in the dataset with every new application of the SMOTE algorithm. Although class 7 (16 instances) always present a significantly higher number of instances than the other classes, the 7-steps dataset at least achieves a balanced situation between the rest of the classes that varies between 8 and 10 instances each.

The modelling process was repeated with the new 3 datasets and the quality indicators of the best models are collected in Tables 10, 11 and 12. The tuning range, steps and ranges, are the same as for the regression problem. The most accurate RBFs are achieved with the following configuration: 3 functions (all cases), 0.001, 0.0001 and 0.02 ridge and 0.002, 0.003 and 0.005 tolerance for 3, 5 and 7-steps datasets respectively. The most accurate MLPs are achieved with the following configuration: 2, 2 and 4 neurons, 0.3, 0.3

Table 10 Accuracy of the prediction models for flatness deviations as a discretized output with a 3-steps extended dataset with SMOTE

	Accuracy, %	TP rate	FP rate	Precision	Recall	F-measure	ROC area
ZeroR	38.1	0.381	0.381	0.145	0.381	0.210	0.287
Logistic classifier	73.81	0.738	0.081	0.731	0.738	0.729	0.929
M5P Tree	66.66	0.667	0.104	0.623	0.667	0.643	0.844
MLP	64.29	0.643	0.146	0.471	0.643	0.543	0.893
RBF	64.29	0.643	0.125	0.563	0.643	0.587	0.911
Random Forest	78.57	0.786	0.066	0.773	0.786	0.773	0.929
Best models are outlined in bold							

Table 11 Accuracy of the prediction models for flatness deviations as a discretized output with a 5-steps extended dataset with SMOTE

	Accuracy, %	TP rate	FP rate	Precision	Recall	F-measure	ROC area
ZeroR	32.0	0.320	0.320	0.102	0.320	0.155	0.347
Logistic classifier	70.0	0.700	0.079	0.702	0.700	0.650	0.945
M5P Tree	74.0	0.740	0.066	0.736	0.740	0.737	0.899
MLP	80.0	0.800	0.057	0.738	0.800	0.753	0.922
RBF	72.0	0.720	0.061	0.749	0.720	0.679	0.903
Random Forest	80.0	0.800	0.051	0.785	0.800	0.785	0.961
Best models are outlined in bold							

Table 12 Accuracy of the prediction models for flatness deviations as a discretized output with a 7-steps extended dataset with SMOTE

	Accuracy, %	TP rate	FP rate	Precision	Recall	F-measure	ROC area
ZeroR	27.1	0.271	0.271	0.074	0.271	0.116	0.412
Logistic classifier	76.27	0.763	0.053	0.784	0.763	0.756	0.961
M5P Tree	78.0	0.780	0.046	0.797	0.780	0.783	0.896
MLP	76.27	0.763	0.045	0.765	0.763	0.735	0.943
RBF	71.19	0.712	0.065	0.750	0.712	0.679	0.928
Random Forest	86.44	0.864	0.031	0.862	0.864	0.861	0.977
Best models are outlined in bold							

and 0.2 learning rate and 0.5, 0.4 and 0.5 momentum for 3, 5 and 7-steps datasets respectively.

Some remarks should be done on the results collected in Tables 10 and 11. There is a continuous improvement in the accuracy of all the machine-learning techniques with the increment in the SMOTE level. This result is related to the main reason for the low accuracy of the machine-learning models in the original dataset: the imbalanced nature of this dataset. In the other hand ZeroR, the baseline method that predicts always the majority class, decreases its performance continuously. This result is also expected because the majority class reduces its presence in the dataset with the performance of the SMOTE technique. The other baseline method, the logistic regression, gets reasonable results, although its quality indicators are not as good as the best machine-learning algorithm for each dataset. In any case, a logistic fit seems to be able to extract the main information of the cutting process (see Fig. 4) although the noise effect reduces its accuracy. This method also improves its accuracy with the SMOTE level. Therefore, if the imbalance level is reduced, all machine-learning techniques perform better. In the case of a 5-steps SMOTE extended dataset (Table 12) MLPs achieve the same accuracy as Random Forest, the best

model. This case is especially interesting because it shows that accuracy can only be taken into account if metrics for imbalance dataset are also considered: although MLP achieve the same accuracy than Random Forest, its FP rate is worse than the FP rate of Random Forest and this fact will penalize the metrics F-Measure and ROC Area. Figures 9 and 10 show the evolution of the accuracy and the ROC Area respectively depending on the SMOTE level, where 0-level refers to the original dataset. These figures show clearly the better performance of Random Forest ensembles and the misleading effect that accuracy has if it is considered as the main quality indicator in imbalanced datasets for manufacturing problems. Besides, Fig. 10 shows that decision trees and random forest ensembles have a very smooth behavior, while ANNs have a noisy behavior, perhaps related to the effect of the tuning process of their parameters.

Summarizing, one of the main objectives of this research was to evaluate the capability of SMOTE algorithm to reduce the effect of imbalance in a real industrial dataset. But this dataset is not only imbalanced but can be also considered very noisy as one of the inputs, the tool's wear, is a very non-linear process depending on many parameters that cannot be evaluated. Therefore, this dataset becomes a challenge

Fig. 9 Evolution of the model's accuracy with the SMOTE level in the dataset

Fig. 10 Evolution of the ROC Area with the SMOTE level in the dataset

for SMOTE for its both imbalance and noisy nature. The results already discussed that SMOTE-based models provide higher accuracy (in terms of different quality measures) than other methods. Therefore, SMOTE is stable under the type of noise presented in this dataset, and it is suitable to improve the accuracy in this manufacturing process and for this quality evaluator: flatness deviation in face milling. This conclusion, that have been previously outlined for Big Data problems with different noisy structure, mainly related to human behavior (Leevy et al 2018; Fernandez et al. 2018) or biological nature (García-Pedrajas et al. 2012), can now be extended to manufacturing problems with small datasets. Due to this result, it has not been necessary to apply noise-resistant variants of SMOTE in this research (Fernandez et al. 2018).

Conclusions

This study proposes the use of different machine-learning techniques for surface deviation from flatness prediction in face milling operations under real industrial conditions. Those conditions limit the number of controlled inputs of the process and the number of different cutting conditions that are tested to reduce testing costs. Therefore, small and noisy datasets are to be expected under real industrial conditions and machine learning techniques are expected to be more accurate than experimental or analytical solutions for these tasks.

Firstly, the experiments were conducted by face milling the steel at different wear conditions. The tool's life and wear, the deviation from flatness and the power drive data were recorded.

Secondly, the prediction of the deviation from flatness is addressed as a regression problem and the deviation is considered a continuous output of the prediction's model. The considered quality indicators (RMSE and MAE) of the tested machine-learning models are statistically better than the baseline methods. RBFs perform 12–17% better than MLPs depending on the considered quality indicator, but their best configuration is very unstable. Regression trees are less precise than ANNs. In any case, the prediction's error of the best model is still too high from the industrial point of view—perhaps due to the small size of the dataset and its imbalance. The use of tool's wear, a very noisy input due to the nature of tool's degradation plays a major role in this fact.

Thirdly, the prediction of the deviation from flatness is addressed as a classification problem and the deviation is considered as a discrete output of the prediction's model taking into account industrial standards for the discretization. Besides, SMOTE is used in 3 different levels to reduce the imbalance level in the dataset. In this case, the quality indicators of the prediction models are not limited to the accuracy but to other metrics of interest in the case of imbalanced dataset like TP rate, FP rate, Recall, Precision, F-measure and ROC Area. As in the continuous case, all the machine-learning methods perform clearly better than the baseline methods. Again, in this case RBFs perform better than MLPs. However, discretization improves the behavior of decision trees and, therefore, Random Forest ensembles are able to perform clearly better than RBFs, especially in the metrics directly thought for imbalanced datasets, like F-measure or ROC area. The analysis of the effect of the SMOTE's strategy shows that there is a continuous improvement in the accuracy of all the machine-learning techniques with the increment in the SMOTE level. Decision trees and random forest ensembles have a very smooth behavior, while ANNs have a noisy behavior, perhaps related to the effect of the complex tuning process of their parameters.

Finally, although the computational time is not a major issue in manufacturing problems with small datasets like the case presented in this research, it might play a main role in other manufacturing problems like failure detection in online monitoring systems. Therefore, under similar accuracy levels, Random Forest or Regression trees may be preferred than ANNs models considering the issue that they are built quicker and no tuning process of their parameters is required. Specifically, to the dataset used in this research, Random Forest or Regression trees are quite quicker than ANNs or RBFs considering the tuning time.

Future research will be focused on the effect that new instances can have in machine-learning models already built because a solution to the small size of industrial datasets might be the construction of flexible machine-learning models able to learn from new cutting conditions once they are already implemented in real workshops. It is possible to integrate the suggestions for surface deviation from flatness prediction in face milling operations into computer-aided manufacturing systems.

Acknowledgements Special thanks to Dr. Juan J. Rodríguez from the University of Burgos for his kind-spirited and useful advice. The work was supported by Act 211 Government of the Russian Federation, Contract No. 02.A03.21.0011 and partially supported by the Project TIN2015-67534-P (MINECO/FEDER, UE) of the Ministerio de Economía Competitividad of the Spanish Government and the Project BU085P17 (JCyL/FEDER, UE) of the Junta de Castilla y León both cofinanced from European Union FEDER funds. The research was carried out within the South-Ural State University Project 5-100 from 2016 to 2020 aimed to increase the competitiveness of leading Russian universities among the world research and educational centers. The authors gratefully acknowledge the support of NVIDIA Corporation with the donation of the Titan Xp GPU used for this research.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

- Abellan-Nebot, J. V., & Romero Subirón, F. (2010). A review of machining monitoring systems based on artificial intelligence process models. *International Journal of Advanced Manufacturing Technology*, 47(1–4), 237–257. <https://doi.org/10.1504/IJMMM.2010.034486>.
- Badar, M. A., Raman, S., & Pulat, P. S. (2005). Experimental verification of manufacturing error pattern and its utilization in form tolerance sampling. *International Journal of Machine Tools and Manufacture*, 45(1), 63–73. <https://doi.org/10.1016/j.ijmachtools.2004.06.017>.
- Benardos, P. G., & Vosniakos, G.-C. (2003). Predicting surface roughness in machining: A review. *International Journal of Machine Tools and Manufacture*, 43(8), 833–844. [https://doi.org/10.1016/S0890-6955\(03\)00059-2](https://doi.org/10.1016/S0890-6955(03)00059-2).
- Bhattacharyya, P., Sengupta, D., Mukhopadhyay, S., & Chattopadhyay, A. B. (2008). On-line tool condition monitoring in face milling using current and power signals. *International Journal of Production Research*, 46(4), 1187–1201. <https://doi.org/10.1080/00207540600940288>.
- Breiman, L. (2001). Random forests. *Machine Learning*, 45(1), 5–32. <https://doi.org/10.1023/A:1010933404324>.
- Bustillo, A., & Correa, M. (2012). Using artificial intelligence to predict surface roughness in deep drilling of steel components. *Journal of Intelligent Manufacturing*, 23(5), 1893–1902. <https://doi.org/10.1007/s10845-011-0506-8>.
- Bustillo, A., Díez-Pastor, J.-F., Quintana, G., & García-Osorio, C. (2011). Avoiding neural network fine tuning by using ensemble learning: Application to ball-end milling operations. *International Journal of Advanced Manufacturing Technology*, 57(5–8), 521–532. <https://doi.org/10.1007/s00170-011-3300-z>.
- Bustillo, A., Grzenda, M., & Macukow, B. (2016). Interpreting tree-based prediction models and their data in machining processes. *Integrated Computer-Aided Engineering*, 23(4), 349–367. <https://doi.org/10.3233/ICA-160513>.
- Bustillo, A., Pimenov, D. Yu., Matuszewski, M., & Mikolajczyk, T. (2018). Using artificial intelligence models for the prediction of surface wear based on surface isotropy levels. *Robotics and Computer Integrated Manufacturing*, 53, 215–227. <https://doi.org/10.1016/j.rcim.2018.03.011>.
- Bustillo, A., & Rodríguez, J. J. (2014). Online breakage detection of multitooth tools using classifier ensembles for imbalanced data. *International Journal of Systems Science*, 45(12), 2590–2602. <https://doi.org/10.1080/00207721.2013.775378>.
- Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, 16, 321–357.
- D'Addona, D. M., Ullah, A. M. M. S., & Matarazzo, D. (2017). Tool-wear prediction and pattern-recognition using artificial neural network and DNA-based computing. *Journal of Intelligent Manufacturing*, 28(6), 1285–1301. <https://doi.org/10.1007/s10845-015-1155-0>.
- da Silva, R. H. L., da Silva, M. B., & Hassui, A. (2016). A probabilistic neural network applied in monitoring tool wear in the end milling operation via acoustic emission and cutting power signals. *Machining Science and Technology*, 20(3), 386–405. <https://doi.org/10.1080/10910344.2016.1191026>.
- Davoudinejad, A., Annoni, M., Rebaioli, L., & Semeraro, Q. (2014). Improvement of surface flatness in high precision milling. In *Conference proceedings—14th international conference of the European Society for precision engineering and nanotechnology, EUSPEN 2014* (Vol. 2, pp. 190–193).
- Denkena, B., & Hasselberg, E. (2015). Influence of the cutting tool compliance on the workpiece surface shape in face milling of workpiece compounds. *Procedia CIRP*, 31, 7–12. <https://doi.org/10.1016/j.procir.2015.03.074>.
- Dobrzynski, M., Chuchala, D., & Orlowski, K. A. (2018). The effect of alternative cutter paths on flatness deviations in the face milling of aluminum plate parts. *Journal of Machine Engineering*, 18(1), 80–87. <https://doi.org/10.5604/01.3001.0010.8825>.
- Fawcett, T. (2006). An introduction to ROC analysis. *Pattern Recognition Letters*, 27(8), 861–874. <https://doi.org/10.1016/j.patrec.2005.10.010>.
- Fernandez, A., Garcia, S., Herrera, F., & Chawla, N. V. (2018). SMOTE for learning from imbalanced data: Progress and challenges, mark-

- ing the 15-year anniversary. *Journal of Artificial Intelligence Research*, 61, 863–905. <https://doi.org/10.1613/jair.1.11192>.
- García-Ordás, M. T., Alegre, E., González-Castro, V., & Alaiz-Rodríguez, R. (2017). A computer vision approach to analyze and classify tool wear level in milling processes using shape descriptors and machine learning techniques. *International Journal of Advanced Manufacturing Technology*, 90(5–8), 1947–1961. <https://doi.org/10.1007/s00170-016-9541-0>.
- García-Ordás, M. T., Alegre-Gutiérrez, E., Alaiz-Rodríguez, R., & González-Castro, V. (2018). Tool wear monitoring using an online, automatic and low cost system based on local texture. *Mechanical Systems and Signal Processing*, 112, 98–112. <https://doi.org/10.1016/j.ymssp.2018.04.035>.
- García-Pedrajas, N., Pérez-Rodríguez, J., García-Pedrajas, M., Ortiz-Boyer, D., & Fyfe, C. (2012). Class imbalance methods for translation initiation site recognition in DNA sequences. *Knowledge-Based Systems*, 25(1), 22–34. <https://doi.org/10.1016/j.knosys.2011.05.002>.
- Grzenda, M., & Bustillo, A. (2019). Semi-supervised roughness prediction with partly unlabeled vibration data streams. *Journal of Intelligent Manufacturing*, 30(2), 933–945. <https://doi.org/10.1007/s10845-018-1413-z>.
- Grzenda, M., Bustillo, A., Quintana, G., & Ciurana, J. (2012). Improvement of surface roughness models for face milling operations through dimensionality reduction. *Integrated Computer-Aided Engineering*, 19(2), 179–197. <https://doi.org/10.3233/ICA-2012-0398>.
- Gu, F., Melkote, S. N., Kapoor, S. G., & Devor, R. E. (1997a). A model for the prediction of surface flatness in face milling. *Journal of Manufacturing Science and Engineering, Transactions of the ASME*, 119(4 PART 1), 476–484.
- Gu, F., Melkote, S. N., Kapoor, S. G., & DeVor, R. E. (1997b). Model for the prediction of surface flatness in face milling. *Journal of Manufacturing Science and Engineering, Transactions of the ASME*, 119(4), 476–484.
- Guzeev, V. I., & Pimenov, D. Yu. (2011). Cutting force in face milling with tool wear. *Russian Engineering Research*, 31(10), 989–993. <https://doi.org/10.3103/S1068798X11090139>.
- Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., & Witten, I. H. (2009). The WEKA data mining software: An update. *ACM SIGKDD Explorations Newsletter*, 11, 10–18. <https://doi.org/10.1145/1656274.1656278>.
- Hossin, M., & Sulaiman, M. N. (2015). A review on evaluation metrics for data classification evaluations. *International Journal of Data Mining & Knowledge Management Process*, 5(2), 1–11. <https://doi.org/10.5121/ijdkp.2015.5201>.
- Huang, P. B., Zhang, H.-J., & Lin, Y.-C. (2019a). Development of a Grey online modeling surface roughness monitoring system in end milling operations. *Journal of Intelligent Manufacturing*, 30(4), 1923–1936. <https://doi.org/10.1007/s10845-017-1361-z>.
- Huang, Y., & Hoshi, T. (2001). Optimization of fixture design with consideration of thermal deformation in face milling. *Journal of Manufacturing Systems*, 19(5), 332–340. [https://doi.org/10.1016/S0278-6125\(01\)89005-1](https://doi.org/10.1016/S0278-6125(01)89005-1).
- Huang, Z., Zhu, J., Lei, J., Li, X., & Tian, F. (2019b). Tool wear predicting based on multi-domain feature fusion by deep convolutional neural network in milling operations. *Journal of Intelligent Manufacturing*, 31(4), 953–966. <https://doi.org/10.1007/s10845-019-01488-7>.
- Kohavi, R. (1995). A study of cross-validation and bootstrap for accuracy estimation and model selection. In *Proceedings of the fourteenth international joint conference on artificial intelligence, Morgan Kaufmann, San Mateo* (Vol. 2 (12), pp. 1137–1143).
- Kovac, P., Rodic, D., Pucovsky, V., Savkovic, B. G., & M., (2013). Application of fuzzy logic and regression analysis for modeling surface roughness in face milling. *Journal of Intelligent Manufacturing*, 24(4), 755–762. <https://doi.org/10.1007/s10845-012-0623-z>.
- Krishnaprasad, K., Sumesh, C. S., & Ramesh, A. (2019). Numerical modeling and multi objective optimization of face milling of AISI 304 steel. *Journal of Applied and Computational Mechanics*, 5(4), 749–762. <https://doi.org/10.22055/JACM.2019.27528.1414>.
- Kuncheva, L. I. (2004). *Combining pattern classifiers: Methods and algorithms*. Hoboken, NJ: Wiley-Interscience.
- Kuram, E., & Ozelik, B. (2016). Micro-milling performance of AISI 304 stainless steel using Taguchi method and fuzzy logic modelling. *Journal of Intelligent Manufacturing*, 27(4), 817–830. <https://doi.org/10.1007/s10845-014-0916-5>.
- Leevy, J. L., Khoshgoftar, T. M., Bauder, R. A., & Seliya, N. (2018). A survey on addressing high-class imbalance in big data. *Journal of Big Data*, 5, 42. <https://doi.org/10.1186/s40537-018-0151-6>.
- Leonard, J. A., & Kramer, M. A. (1991). Radial basis function networks for classifying process faults. *IEEE Control Systems*, 11(3), 31–38. <https://doi.org/10.1109/37.75576>.
- Liu, E. A., & Zou, Q. (2011). Machined surface error analysis a face milling approach. *Journal of Advanced Manufacturing Systems*, 10(2), 293–307. <https://doi.org/10.1142/S0219686711002211>.
- Machado, A. R., & Diniz, A. E. (2017). Tool wear analysis in the machining of hardened steels. *International Journal of Advanced Manufacturing Technology*, 92(9–12), 4095–4109. <https://doi.org/10.1007/s00170-017-0455-2>.
- Markopoulos, A. P., Manolakis, D. E., & Vaxevanidis, N. M. (2008). Artificial neural network models for the prediction of surface roughness in electrical discharge machining. *Journal of Intelligent Manufacturing*, 19(3), 283–292. <https://doi.org/10.1007/s10845-008-0081-9>.
- Maudes, J., Bustillo, A., Guerra, A. J., & Ciurana, J. (2017). Random Forest ensemble prediction of stent dimensions in microfabrication processes. *International Journal of Advanced Manufacturing Technology*, 91(1–4), 879–893. <https://doi.org/10.1007/s00170-016-9695-9>.
- Mendes-Moreira, J., Soares, C., Jorge, A. M., & De Sousa, J. F. (2012). Ensemble approaches for regression: A survey. *ACM Computing Surveys*, 45(1), 10. <https://doi.org/10.1145/2379776.2379786>.
- Mikolajczyk, T., Nowicki, K., Bustillo, A., & Pimenov, D. Yu. (2018). Predicting tool life in turning operations using neural networks and image processing. *Mechanical Systems and Signal Processing*, 104, 503–513. <https://doi.org/10.1016/j.ymssp.2017.11.022>.
- Mohanraj, T., Shankar, S., Rajasekar, R., Sakthivel, N. R., & Pramanik, A. (2020). Tool condition monitoring techniques in milling process—A review. *Journal of Materials Research and Technology*, 9(1), 1032–1042. <https://doi.org/10.1016/j.jmrt.2019.10.031>.
- Mori, M., Fujishima, M., Inamasu, Y., & Oda, Y. (2011). A study on energy efficiency improvement for machine tools. *CIRP Annals Manufacturing Technology*, 60(1), 145–148. <https://doi.org/10.1016/j.cirp.2011.03.099>.
- Nadeau, C., & Bengio, Y. (2003). Inference for the generalization error. *Machine Learning*, 52(3), 239–281. <https://doi.org/10.1023/A:1024068626366>.
- Nadolny, K., & Kaplonek, W. (2014). Analysis of flatness deviations for austenitic stainless steel workpieces after efficient surface machining. *Measurement Science Review*, 14(4), 204–212. <https://doi.org/10.2478/msr-2014-0028>.
- Nguyen, H. T., Wang, H., & Hu, S. J. (2014). Modeling cutter tilt and cutter-spindle stiffness for machine condition monitoring in face milling using high-definition surface metrology. *International Journal of Advanced Manufacturing Technology*, 70(5–8), 1323–1335. <https://doi.org/10.1007/s00170-013-5347-5>.
- Oleaga, I., Pardo, C., Zulaika, J. J., & Bustillo, A. (2018). A machine-learning based solution for chatter prediction in heavy-duty milling machines. *Measurement: Journal of the International*

- Measurement Confederation, 128, 34–44. <https://doi.org/10.1016/j.measurement.2018.06.028>.
- Oztemel, E., & Gursev, S. (2020). Literature review of Industry 4.0 and related technologies. *Journal of Intelligent Manufacturing*, 31(1), 127–182. <https://doi.org/10.1007/s10845-018-1433-8>.
- Park, K. S., & Kim, S. H. (1998). Artificial intelligence approaches to determination of CNC machining parameters in manufacturing: A review. *Artificial Intelligence in Engineering*, 12(1–2), 127–134. [https://doi.org/10.1016/S0954-1810\(97\)00011-3](https://doi.org/10.1016/S0954-1810(97)00011-3).
- Pimenov, D. Yu. (2015). Mathematical modeling of power spent in face milling taking into consideration tool wear. *Journal of Friction and Wear*, 36(1), 45–48. <https://doi.org/10.3103/S1068366615010110>.
- Pimenov, D. Yu., Bustillo, A., & Mikolajczyk, T. (2018a). Artificial intelligence for automatic prediction of required surface roughness by monitoring wear on face mill teeth. *Journal of Intelligent Manufacturing*, 29(5), 1045–1061. <https://doi.org/10.1007/s10845-017-1381-8>.
- Pimenov, D. Yu., & Guzeev, V. I. (2017). Mathematical model of plowing forces to account for flank wear using FME modeling for orthogonal cutting scheme. *International Journal of Advanced Manufacturing Technology*, 89(9–12), 3149–3159. <https://doi.org/10.1007/s00170-016-9216-x>.
- Pimenov, D. Yu., Guzeev, V. I., & Koshin, A. A. (2011a). Elastic displacement of a technological system in face milling with tool wear. *Russian Engineering Research*, 31(11), 1105–1109. <https://doi.org/10.3103/S1068798X11110219>.
- Pimenov, D. Yu., Guzeev, V. I., & Koshin, A. A. (2011b). Influence of cutting conditions on the stress at tool's rear surface. *Russian Engineering Research*, 31(11), 1151–1155. <https://doi.org/10.3103/S1068798X11110207>.
- Pimenov, D. Yu., Guzeev, V. I., Krolczyk, G., Mia, M., & Wojciechowski, S. (2018b). Modeling flatness deviation in face milling considering angular movement of the machine tool system components and tool flank wear. *Precision Engineering*, 54, 327–337. <https://doi.org/10.1016/j.precisioneng.2018.07.001>.
- Pimenov, D. Yu., Guzeev, V. I., Mikolajczyk, T., & Patra, K. (2017). A study of the influence of processing parameters and tool wear on elastic displacements of the technological system under face milling. *International Journal of Advanced Manufacturing Technology*, 92(9–12), 4473–4486. <https://doi.org/10.1007/s00170-017-0516-6>.
- Quinlan, J. R. (1996). Learning decision tree classifiers. *ACM Computing Surveys*, 28(1), 71–72. <https://doi.org/10.1145/234313.234346>.
- Rodrigues, M. A., Hassui, A., Lopes da Silva, R. H., & Loureiro, D. (2016). Tool life and wear mechanisms during Alloy 625 face milling. *International Journal of Advanced Manufacturing Technology*, 85(5–8), 1439–1448. <https://doi.org/10.1007/s00170-015-8056-4>.
- Rybicki, M., & Kawalec, M. (2010). Form deviations of hot work tool steel 55NiCrMoV (52HRC) after face finish milling. *International Journal of Machining and Machinability of Materials*, 7(3–4), 176–192. <https://doi.org/10.1504/IJMMM.2010.033065>.
- Samanta, B., Ercvelles, W., & Omurtag, Y. (2008). Prediction of workpiece surface roughness using soft computing. *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, 222(10), 1221–1232. <https://doi.org/10.1243/09544054JEM1035>.
- Sanjay, C., & Jyothi, C. (2006). A study of surface roughness in drilling using mathematical analysis and neural networks. *International Journal of Advanced Manufacturing Technology*, 29(9–10), 846–852. <https://doi.org/10.1007/s00170-005-2538-8>.
- Santos, P., Maudes, J., & Bustillo, A. (2018). Identifying maximum imbalance in datasets for fault diagnosis of Gearboxes. *Journal of Intelligent Manufacturing*, 29(2), 333–351. <https://doi.org/10.1007/s10845-015-1110-0>.
- Shao, H., Wang, H. L., & Zhao, X. M. (2004). A cutting power model for tool wear monitoring in milling. *International Journal of Machine Tools and Manufacture*, 44(14), 1503–1509. <https://doi.org/10.1016/j.ijmachtools.2004.05.003>.
- Shnfir, M., Olufayo, O. A., Jomaa, W., & Songmene, V. (2019). Machinability study of hardened 1045 steel when milling with ceramic cutting inserts. *Materials*, 12(23), 3974. <https://doi.org/10.3390/ma12233974>.
- 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. *Mechanical Systems and Signal Processing*, 16(4), 487–546. <https://doi.org/10.1006/mssp.2001.1460>.
- Simunovic, G., Simunovic, K., & Saric, T. (2013). Modelling and simulation of surface roughness in face milling. *International Journal of Simulation Modelling*, 12(3), 141–153. [https://doi.org/10.2507/IJSIMM12\(3\)1.219](https://doi.org/10.2507/IJSIMM12(3)1.219).
- Stehman, S. V. (1997). Selecting and interpreting measures of thematic classification accuracy. *Remote Sensing of Environment*, 62(1), 77–89. [https://doi.org/10.1016/s0034-4257\(97\)00083-7](https://doi.org/10.1016/s0034-4257(97)00083-7).
- Teixidor, D., Grzenda, M., Bustillo, A., & Ciurana, J. (2015). Modeling pulsed laser micromachining of micro geometries using machine-learning techniques. *Journal of Intelligent Manufacturing*, 26(4), 801–814. <https://doi.org/10.1007/s10845-013-0835-x>.
- Witten, I. H., Frank, E., Hall, M. A., & Pal, C. J. (2016). Data mining: Practical machine learning tools and techniques (Book). In *Data mining: Practical machine learning tools and techniques* (pp. 1–621).
- Wu, D., Jennings, C., Terpenney, J., Gao, R. X., & Kumara, S. (2017). A comparative study on machine learning algorithms for smart manufacturing: tool wear prediction using random forests. *Journal of Manufacturing Science and Engineering, Transactions of the ASME*, 139(7), 071018. <https://doi.org/10.1115/1.4036350>.
- Xu, L., Huang, C., Li, C., Wang, J., Liu, H., & Wang, X. (2020). Estimation of tool wear and optimization of cutting parameters based on novel ANFIS-PSO method toward intelligent machining. *Journal of Intelligent Manufacturing*. <https://doi.org/10.1007/s10845-020-01559-0>.
- Yegnanarayana, B. (2009). *Artificial neural networks*. PHI Learning Pvt. Ltd.
- Yi, W., Jiang, Z., Shao, W., Han, X., & Liu, W. (2015). Error compensation of thin plate-shape part with prebending method in face milling. *Chinese Journal of Mechanical Engineering*, 28(1), 88–95. <https://doi.org/10.3901/CJME.2014.1120.171>.
- Zain, A. M., Haron, H., & Sharif, S. (2010). Prediction of surface roughness in the end milling machining using artificial neural network. *Expert Systems with Applications*, 37(2), 1755–1768. <https://doi.org/10.1016/j.eswa.2009.07.033>.
- Zheng, P., Wang, H., Sang, Z., Zhong, R. Y. E., Liu, Y., Yu, S., et al. (2018). Smart manufacturing systems for Industry 4.0: Conceptual framework, scenarios, and future perspectives. *Frontiers of Mechanical Engineering*, 13(2), 137–150. <https://doi.org/10.1007/s11465-018-0499-5>.
- Zhu, K. P., Wong, Y. S., & Hong, G. S. (2009). Wavelet analysis of sensor signals for tool condition monitoring: A review and some new results. *International Journal of Machine Tools and Manufacture*, 49(7–8), 537–553. <https://doi.org/10.1016/j.ijmachtools.2009.02.003>.