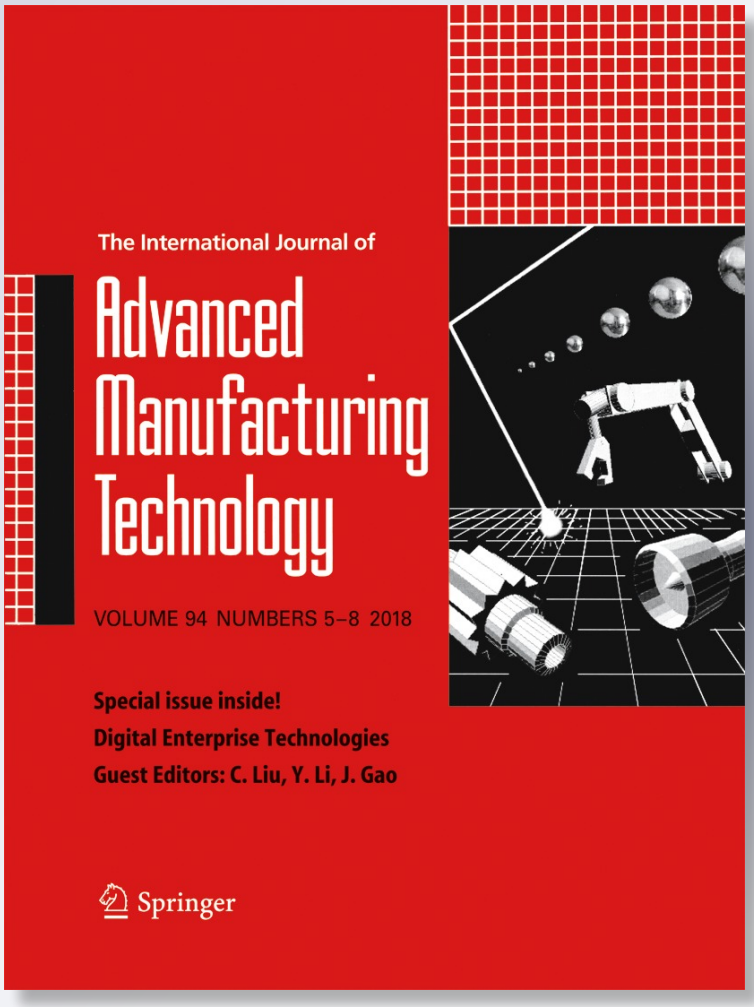
On the use of machine learning methods to predict component reliability from data-driven industrial case studies

Emanuel F. Alsina, Manuel Chica, Krzysztof Trawiński & Alberto Regattieri

The International Journal of Advanced Manufacturing Technology

ISSN 0268-3768
Volume 94
Combined 5-8

Int J Adv Manuf Technol (2018)
94:2419-2433
DOI 10.1007/s00170-017-1039-x



Springer

Int J Adv Manuf Technol (2018) 94:2419–2433
DOI 10.1007/s00170-017-1039-x

ORIGINAL ARTICLE

On the use of machine learning methods to predict component reliability from data-driven industrial case studies

Emanuel F. Alsina[1] · Manuel Chica[2,3] · Krzysztof Trawiński[4] · Alberto Regattieri[5]

Received: 12 June 2017 / Accepted: 29 August 2017 / Published online: 11 September 2017

Abstract The reliability estimation of engineered components is fundamental for many optimization policies in a production process. The main goal of this paper is to study how machine learning models can fit this reliability estimation function in comparison with traditional approaches (e.g., Weibull distribution). We use a supervised machine learning approach to predict this reliability in 19 industrial components obtained from real industries. Particularly, four diverse machine learning approaches are implemented: artificial neural networks, support vector machines, random forest, and soft computing methods. We evaluate if there is one approach that outperforms the others when predicting the reliability of all the components, analyze if machine learning models improve their performance in the presence of censored data, and finally, understand the performance impact when the number of available inputs changes. Our experimental results show the high ability of machine learning to predict the component reliability and particularly, random forest, which generally obtains high accuracy and the best results for all the cases. Experimentation confirms that all the models improve their performance when considering censored data. Finally, we show how machine learning models obtain better prediction results with respect to traditional methods when increasing the size of the time-to-failure datasets.

Keywords Reliability prediction · Machine learning · Censored data · Weibull distribution

✉ Manuel Chica
mchica@rod.com.es

Emanuel F. Alsina
emanuelfederico.alsina@unimore.it

Krzysztof Trawiński
krzysztof@novelti.io

Alberto Regattieri
alberto.regattieri@unibo.it

[1] Department of Physics, Mathematics and Informatics, University of Modena and Reggio Emilia, 41125 Modena, Italy

[2] RØD Brand Consultants, 28001 Madrid, Spain

[3] School of Electrical Engineering and Computing, The University of Newcastle, Callaghan NSW, 2308, Australia

[4] Novelti, 28012 Madrid, Spain

[5] Department of Industrial Engineering, University of Bologna, 40136 Bologna, Italy

1 Introduction

The advance of international markets and the consequent increase of global competition have led manufacturers to create more and more customizable products to reach higher customer expectations. Nowadays, manufacturers need to produce high-quality products in less time. In this competitive environment, the manufacturers' interest is to focus more than ever before on the machines' reliability [1, 2]. Then, an appropriate maintenance is essential in manufacturing systems to ensure all operating equipment in healthy condition, reduce failures, and guarantee the quality of the produced items [3]. Several additional factors have motivated this interest growth in reliability, which include the increasing of complexity and sophistication of the systems [4], and the insistence on product quality, warranty programs, safety laws, and supply chain sustainability [5]. Some of the latter factors are also influenced by the high cost of failures, their repairs, or their replacement [6].

Many possible optimization policies can apply reliability with its corresponding improvement in safety and costs. The study of the reliability is important to make important decisions to improve different aspects of the manufacturing processes [6–8]. Examples of these policies are production planning, optimal mix of maintenance policies, fault detection, effective spare part management, and warehouse optimization. Two maintenance techniques exist in the literature: time-based maintenance and condition-based maintenance [8]. For the first type of maintenance, a robust reliability analysis requires an a priori effective failure process investigation. This process is based on the operation and failure times of a generic component that can be a part, device, piece of equipment, or the whole system. Failure times, also called time-to-failure (TTF), are the times when the considered components stop working adequately.

Generally, this failure process investigation is costly as the data collection requires a lot of effort. For instance, one of the most difficult task in the industry is to obtain sufficient and reliable data available as it is likely to have very little accurate TTF data [9]. Therefore, practitioners sometimes make use of *incomplete* failure information to define the reliability of a component. This *incomplete* failure information derives from the components which are still operating at the end of the tests of the failure investigation. These conditions are known as *censored data* situations and can influence the analysis of the components' reliability [10].

The application of mathematical models is a common practice to predict the reliability of the components when TTFs are available. In most of the cases, theoretical probability distributions, fault tree analysis, and Markov models are used for the systems reliability modeling [11]. However, these reliability modeling processes often makes simplified assumptions to enable analytical or suitable numerical treatment difficult to validate. Also, several complex factors that exhibit non-linear patterns influence the study of the reliability [12]. Therefore, the analysis of the reliability cannot depend on the assumptions of independence and linearity but requires more sophisticated models to capture the complexities of the reliability behavior in a realistic way.

Recently, new machine learning models, able to capture the complexity of the reliability [13, 14], emerged above the non-parametric techniques of failure data regression. Particularly, the use of artificial neural networks (ANNs) [15, 16] and support vector machines (SVMs) [12, 17] showed outstanding results when predicting reliability from historical data analysis. In a nutshell, machine learning models are data-driven learning methods used to train software to make generalized predictions from historical data. These models have the ability to learn automatically to solve problems of different nature and with different dimensionality values, from hundreds of input features to just a few. In the last years, they have been used for the systems reliability modeling, evaluation, and prediction [3, 18, 19], even when the number of features of the problem is not high because of the difficulties of manually acquiring big datasets in the industry.

Given the importance and benefits of machine learning methods for reliability prediction, one of the principal contributions of this paper is the comparison of a wide and diverse set of machine learning models to fit the reliability of a set of industrial real cases. These cases were taken from different conditions and real industrial components. We present the machine learning model application to the failure prediction process, from the initial data collection phase to the final performance evaluation of the machine learning models to predict the reliability of the components. The main purpose is to understand if there is a single data-driven learning model able to outperform the other models under different reliability prediction conditions: from a low number of TTFs (e.g., less than 40) to a high number of them (e.g., more than 100).

Furthermore, this study explores the performance of these models in the presence of censored data. In [20], they demonstrated that neglecting censored information results in significant errors while evaluating the reliability performance of the components using the two-parameter Weibull reliability estimation. We aim to demonstrate the crucial role of this censored data also when using machine learning models and to also suggest effective methodologies to take this data into consideration during the process. To do this, we divide the experimentation into the application of the models to censored and uncensored data. First, we apply the median rank method (MRM) to the uncensored set (all the components) while the product limit estimator (PLE) is applied to just the set with censored data.

Finally, we explore the influence of the number of TTF available in the machine learning models' performance. This study is done by defining a set of instance clusters depending on their number of TTF. In this way, we can show by analyzing the experimentation the best data-driven learning model for every cluster and if the ranking of the methods changes when increasing the available TTF data. The results and conclusions that arise from the broad experimentation can be a useful contribution for decision makers in the field of maintenance engineering to have a wide view on the capabilities and limitations of machine learning methods when real industrial data is available.

Next, Section 2 presents an overview of the process of estimating the reliability function and the effects of the use of censored data during this process. Also, this section presents a study of the related literature. Sections 3 and 4 describe the machine learning models implemented for the work and the real industrial data used in our analysis, respectively. We discuss the performance results of the models in Section 5 where we examine the use of censored

data and the performance analysis when modifying the amount of available TTF data. Finally, Section 6 discusses the conclusions of the work.

2 Background

2.1 The reliability function estimation

The reliability estimation problem can be classified depending on the prediction it makes: system's ability to operate, to complete a mission, or to perform at some given circumstances. We will focus on the prediction of the analyzed components to operate without maintenance and logistic support at certain times [6, 7, 21], also called *basic reliability analysis*. Therefore, the term reliability is used to find a probability function $R(t)$ for a component to perform without failing at least until a given period of time t under stated operating conditions and by assuming that the component is new when $t = 0$. The reliability function is then a decreasing function with time t, assuming that it is equal to one at time zero, and zero at infinity time [6].

The purpose of the non-parametric estimation approach, i.e., an empirical approach, is to directly estimate the reliability function $R(t)$ from a set of failure times. In the same way, this approach can be used in maintenance and logistics support, where the data analyzed correspond to the maintenance, repair, or support task completion times [21]. The data-driven reliability function $\widehat{R}(t)$ is estimated by empirical methods when historical failure times were collected. For instance, one can use different statistical distributions to fit the learnt function $\widehat{R}(\text{TTF})$ as a continuous reliability distribution from the TTF series data. Once obtained, those distributions become available for practitioners to compute the reliability even for failure times which were not present in the range of collected data. The most common distributions used to fit the $\widehat{R}(t)$ are mainly the Weibull, exponential, and normal [6, 8]. This fitting phase can be considered as a machine learning problem to find a model to represent the failure time distribution. In fact, some works have already shown how machine learning and soft computing algorithms are suitable for solving such problems as we will review in next Section 2.3.

2.2 Complete and censored data

Given an environment of n units, we call a *complete data* situation when the failure times of all n units are available. That is, the failure time of the i^{th} unit, represented by t_i, is available for all units $t_1, t_2, ..., t_n$. Figure 1a shows a complete data situation, where during the test, all units have a failure at a given unit of time. In this situation, all the unit failures are known. However, TTF data is often based on non-trivial knowledge about the past performance of components [20]. The available dataset, usually received from industry, is not represented by a series of complete data, but composed of a series of *censored* times. Censored times mean that not all units fail during the tests, or several units fail between two data monitoring times without knowing the exact failure time. It then includes incomplete and/or missing information and in some cases, the available data is not necessarily a tracking data collection. In all the latter scenarios, *censored time* means the precise failure time is unknown, whatever the reason is.

One of the most common reasons for censoring is the fact of analyzing life test data before all units have failed [22]. Figure 1b shows a censored data situation. As it can be observed in this graph, unit 4 is still working at the end of the test. This case is called *censored on the right* as tested units are still in function at the end of a test or they have been removed from tracking before the failure [23]. Similarly, Fig. 1b shows unit 3 where time when the units fail is known, but the effective working time is not known. This fact occurs because the initial working time of the unit is unknown, called *censored on the left*. The last censored data scenario is called *interval censored data* and it is shown by unit 2 in Fig. 1b. This scenario takes place when the state of the observed unit is controlled at a time interval and a failure occurs between two inspections. It is usual to find up to 40% of the failures after the programmed inspections in complex industrial systems [24].

2.3 Existing literature on machine learning for reliability prediction

The machine learning approach has attracted considerable interest lately, mainly due to the flexible capacity and ability of these models to *understand* the complex non-linear relationships between input and output time series patterns through appropriate learning processes [25]. Machine learning models, especially artificial neural networks (ANNs) [26, 27] and support vector machines (SVMs) [28], have been used to improve the quality of the reliability prediction. One of the first works using machine learning for industrial reliability was the one of [29] which explored the accuracy of a machine learning model to study small-sample reliability data. The authors used ANNs to identify the most appropriate probability distribution which fits a set of failure data, and estimate its parameters. The results showed how ANNs perform well when the number of available TTF is small.

Similar results emerged from the work of [30] where the efficiency of simple ANNs was tested to find the cumulative failure distribution of mechanical components. The authors of the latter work showed how these ANN models can perform better than other common distributions,

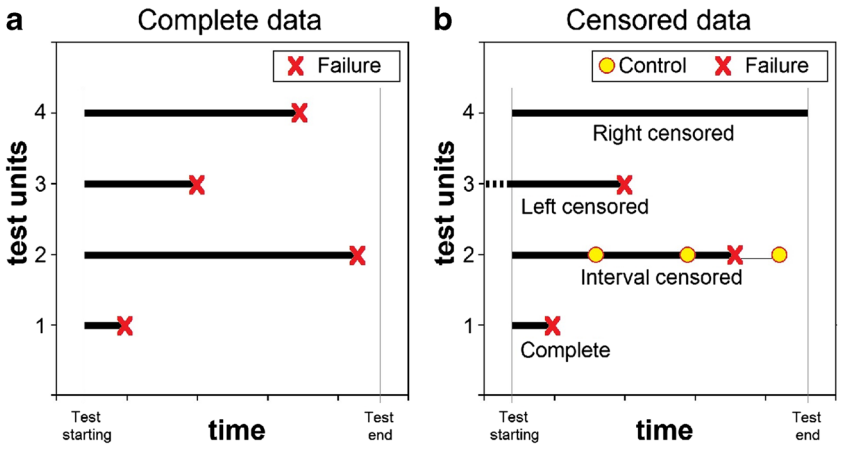


Fig. 1 Two graphs with complete (left-hand) and censored (right-hand) failure time data

especially under poor data conditions. Amjady and Ehsan [15] developed ANNs for evaluating the reliability of a power system, concretely the reliability evaluation of generating units and the transmission system. Chatterjee and Bandopadhyay [16] applied an ANN to predict the cumulative failure time of a load-haul-dump machine by showing how these models can better predict the component reliability than the auto-regressive integrated moving average model (ARIMA). The experimental results of [31] also demonstrated that ANNs outperform the ARIMA approach in terms of prediction accuracy when forecasting repairable systems. Another ANN-related study was done by [32] who predicted the reliability and failures of engine systems using ANNs. The authors confirmed how the use of ANNs provided an alternative and good prediction performance with respect to the existing models. The same Xu's dataset [32] was used in other works to test the performance of other types of ANNs [33]. The results of [33] showed the potential of using ANNs in reliability data analysis, especially for mid- to long-term predictions.

Different support vector machine (SVM) models (i.e., support vector regression (SVR) when dealing a time series fitting problem) have been developed to predict the reliability of the already commented dataset [34]. SVM performed at least equal or even better than ANNs and ARIMA. The experimental results of [35] suggested that, within the forecasting fields of systems reliability, SVM had a better forecast accuracy than ANNs. Hong and Pai [17] made the first attempt to apply a SVM model to predict engine reliability by comparing it with respect to the Duane model, ARIMA, and general regression neural networks. Finally, das Chagas Moura et al. [12] showed the SVM effectiveness in forecasting TTF and reliability of engineered components on literature case studies against two types of ANN (i.e., radial basis function and multilayer perceptron), an auto-regressive-integrated-moving average and the recurrent neural networks. The comparison demonstrated that in the analyzed cases, SVM outperforms or is comparable to other techniques.

Other machine learning approaches have been tested for the same well-known literature instances. An example is the work of [3] where authors proposed a two-stage maintenance framework using evolvable ANNs, restricted Boltzmann machine and deep belief networks. Pai and Lin [36] tested the performance of a soft computing [37] -based approach (i.e., a neural fuzzy model) against ANNs and ARIMA models. They showed that the proposed fuzzy-based model provided lower forecast errors than the other machine learning models.

In summary, all the presented studies demonstrated that machine learning models can perform well in the industrial reliability field. However, they often use one well-known case study to validate their models. Their use, therefore, appears to be ad hoc for a specific dataset and industrial configuration, and it is still unclear if there is a machine learning model able to perform generally well across different TTF datasets. Particularly, Xu's dataset [32] has a large number of TTF data points to train models.

A failure process investigation, indeed, is costly, and sometimes, it is hard to have a long available TTF series (e.g., 40), like Xu's dataset does. One key characteristic of real-world situations is they are diverse and it is not granted that the number of available TTF is of that magnitude. In addition, revised studies used the last five TTFs to test the models. But, in real-world situations, it is not always useful to know the reliability of a component at the end of its life time.

Therefore, one of the main contribution of our work is studying a wide set of possible and different real-world industrial situations with diverse types and number of available data points. Also, we aim at comparing different machine learning approaches to see if there is a common conclusion for all the studied datasets. Next, Section 3 presents these models, from ANNs to more advanced soft computing and bagging classification approaches.

3 Machine learning methods

In the next sub-sections, we present the five machine learning methods applied in the experimentation for predicting the reliability of the mechanical components from the real industrial cases. These methods are based on different paradigms and methodologies to be as diverse as possible. They belong from classical statistical regression [38] to more advanced machine learning methods such as random forests and methods based on soft computing techniques [39, 40].

3.1 Linear regression

Linear regression is a classical statistical technique for modeling the relationship between a scalar dependent variable y_i (output) and x_{ij} which is called the i^{th} input of a j^{th} independent variable (inputs) for the i^{th} sample [38]. The case of one explanatory variable is called simple linear regression. When there are more than one explanatory variable, the process is called multiple linear regression.

Least mean squares (LMS) is the standard fitting approach to adjust the parameters of the linear regression model to existing data, often used as a baseline for innumerable regression problems [38]. In fact, some studies showed that linear regression methods achieved good performance for short-term data stream forecasting in system fault prediction [41].

Linear regression is based on the minimization of the observation errors, obtained from the difference between an observed value of an instance j and the value provided by the fitted function [13, 42]. The deviation between the observed value y_i and the estimated value $y_i^* = \alpha^t x_i$ is generally regarded as the observation error $\epsilon_i = y_i - y_i^*, \forall i = 1, .., N$.

3.2 Support vector regression

In a support vector regression (SVR) model, the input examples are mapped into the high-dimensional feature space through a non-linear mapping (kernel) selected a priori [43]. SVR constructs a linear decision surface, i.e., a hyperplane, in the original input feature space. Precisely, this hyperplane is a linear decision function with maximal margin between the examples of the different categories in order to separate them as much as possible. The maximal margin is determined by the support vectors, a small amount of the training data representing the decision boundary. When the SVR model is trained with the corresponding training data, testing samples are mapped into the same feature space and their output values are predicted based on which side of the hyperplane they fall on.

SVR works on similar principles as support vector machine classification [44]. The SVR technique depends on kernel functions and permits for construction of a non-linear model without changing the explanatory variables, helping in better interpretation of the resultant model. The basic idea behind SVR is not to care about the prediction as long as the error is less than certain value (i.e., principle of maximal margin). SVR is a useful technique provides the user with high flexibility in terms of distribution of underlying variables, relationship between independent and dependent variables, and the control on the penalty term.

3.3 Random forests

The ensemble learning systems are well-recognized machine learning tools capable of obtaining better performance than a single-component model [45]. They are also able to deal with complex and high-dimensional regression and classification problems [45, 46]. They are based on the combination of the output of simple-machine learning systems, in the literature called *weak learners*, into a group of learners to get better fitting results (in regression) or better accuracy rates (in classification). Their performance strongly relies on the diversity of the weak learners as these learners correctly behave on different parts of the problem space.

Random forests (RF) is a state-of-the-art ensemble algorithm for regression and classification [47]. Its learning algorithm constructs a set of diverse decision trees during the training phase and aggregates the results provided by all the component decision trees to compute the final output during the prediction phase. When dealing with a regression problem as the reliability fitting function, RF returns the mean prediction of all its individual trees. The basis of RF is a bagging algorithm [48]. This algorithm builds a forest from a set of random trees which play the role of the weak learners of the ensemble system.

3.4 Artificial neural networks

ANNs are a family of learning algorithms inspired by biological nervous systems of animals (in particular a neural structure of the brain) and are used to estimate or predict functions that are generally unknown and can depend on a large number of inputs [49]. They are presented as systems of interconnected neurons, also called perceptrons, which are able to compute the output values of a complex system from some given inputs. Although neurons in isolation are able of performing simple computations, they need to operate as a collective to solve difficult, non-linear classification and regression problems.

Within the numerous existing ANN algorithms, we specifically use a classical multilayer perceptron algorithm

Table 1 Description of the 19 components' data: number of available TTFs and censored data times (in case they are)

	Component	TTFs	Censored times
1	cod A	73	10
2	cod B	96	10
3	cod C	20	–
4	cod D	25	10
5	cod E	21	6
6	cod F	37	–
7	cod H	22	–
8	cod I	23	4
9	cod L	47	5
10	cod M	32	4
11	cod N	71	7
12	cod O	34	–
13	cod P	11	–
14	cod Q	24	–
15	cod R	24	–
16	cod S	15	–
17	cod T	45	
18	cod U	607	18
19	cod V	282	10

(MLP) and a conjugate gradient method to train the weights of its structure [50]. MLP is a feed-forward ANN which is a directed graph constructed over multiple layers of nodes. Each layer is fully connected to the next one. A conjugate gradient method is an iterative search method that comes from the conventional numerical analysis. This method also uses a scaled conjugate gradient in order to speed up the learning process.

3.5 Soft computing regression method based on fuzzy logic

Fuzzy systems [37], which are based on fuzzy logic [51], became popular in the research community since they have the ability to deal with complex, non-linear problems, difficult to be solved by classical methods [52]. Besides, its capability of knowledge extraction and human-centric representation allows them to become more comprehensible to the users of the models (that is, they are different from classical black-box models) [53, 54].

The lack of the automatic extraction of fuzzy systems has attracted the attention of the soft computing community to incorporate learning capabilities to these kinds of systems. In consequence, a hybridization of fuzzy systems and evolutionary algorithms has become one of the most popular approaches in this field [55–57]. In general, evolutionary genetic fuzzy systems are fuzzy systems enhanced by a learning procedure coming from evolutionary computation, i.e. considering any evolutionary algorithm.

In particular, we fit the reliability function by means of a fuzzy rule learning method based on genetic programming grammar operators and simulated annealing (EFS-SA) [58]. EFS-SA is based on a fuzzy rule-based system with fuzzy *if-then* rules. The learning mechanism of this soft computing algorithm is a hybrid method formed by a simulated annealing [59] and genetic programming approach [60] to build a set of the fuzzy rules which finally generate the numerical output y_i^*. The algorithm learns the rules of the system and the fuzzy partitions from the dataset and uses a tree-based codification for genetic programming. The authors of the algorithm defined specific genetic programming grammar operators and adapted them for the simulated annealing optimization method.

4 Industrial dataset description and experimental setup

This section presents the description of the used dataset and non-parametric methods (Section 4.1) and the experimental setup with all the parameters used for comparing the algorithms (Section 4.2).

4.1 Data description

Different TTF and censored times are typical characteristics of real industrial conditions. For instance, the monitoring

Table 2 Non-parametric methods to estimate the reliability function when there is complete and censored data

Complete Data			
	Direct method	Improved Direct Method	Median Rank
$\widehat{R}(t_i)$	$\frac{n-i}{n}$	$\frac{n+1-i}{n+1}$	$\frac{n+0.7-i}{n+0.4}$
Censored Data			
	Product Limit Estimator		Kaplan-Meier Method
$\widehat{R}(t_i)$	$\left(\frac{n+1-i}{n+2-i}\right)^{\delta_i} \widehat{R}(t_{i-1})$		$\left(1-\frac{1}{n_i}\right)^{\delta_i} \widehat{R}(t_{i-1})$
δ_i = (1: if failure occurs at time t_i, 0: if censoring occurs at time t_i)			

and good maintenance policies can ensure the availability of a consistent number of TTF but, on the other hand, there are cases where only small datasets are available [61]. The goal of the datasets used in this paper is to be an extensive representation of real industrial conditions. As already commented, these datasets are diverse as contain failure data from 19 different real-world engineered components. Particularly, data comes from the information systems which supports the maintenance management of three large companies located in northern Italy, engaged in different industrial sectors. These components belong to mechanical and electrical components of technical assets for the electricity production, equipments for filling beverages, and parts of car assembly lines.

Each of the datasets has different available failure and censored times as shown in Table 1. This table has a column with the number of TTF for each type of component and another column with the number of censored times.

An estimated function $\widehat{R}(t_i)$ is used to evaluate and train the machine learning models. To do this, we use non-parametric approximation methods. Depending on the type of data (complete or censored data), different approximation methods to estimate $\widehat{R}(t_i)$ can be applied. In our study, we use the direct (DM), improved direct (IDM), and median rank (MRM) methods [10] for complete TTF data series. On the other hand, in presence of censored data, we use the product limit estimator (PLE) method, and its variation, known as the Kaplan and Meier method (KM) [62].

Table 3 An example of code E dataset which can be used as complete or censored sample

i	TTF	Censored	Complete dataset $\widehat{R}(t)$			Censored dataset $\widehat{R}(t)$	
			DM	IDM	MRM	PLE	KM
1	376	*					
2	436	*					
3	614	*					
4	830	*					
5	1135		0.957	0.958	0.970	0.962	0.960
6	1285	*					
7	1384		0.913	0.917	0.927	0.921	0.918
8	1865		0.870	0.875	0.885	0.881	0.877
9	1971		0.826	0.833	0.842	0.841	0.835
10	2027		0.783	0.792	0.799	0.801	0.793
11	2202						
12	2202						
13	2202		0.652	0.667	0.671	0.681	0.668
14	2684		0.609	0.625	0.628	0.641	0.626
15	2784	*					
16	2850		0.565	0.583	0.585	0.598	0.581
17	3112		0.522	0.542	0.543	0.556	0.537
18	3258		0.478	0.500	0.500	0.513	0.492
19	3310		0.435	0.458	0.457	0.470	0.447
20	3541		0.391	0.417	0.415	0.427	0.402
21	3674		0.348	0.375	0.372	0.385	0.358
22	3875		0.304	0.333	0.329	0.342	0.313
23	4037		0.261	0.292	0.286	0.299	0.268
24	4339		0.217	0.250	0.244	0.256	0.224
25	4506		0.174	0.208	0.201	0.214	0.179
26	5552		0.130	0.167	0.158	0.171	0.134
27	5821		0.087	0.125	0.115	0.128	0.089
28	6447		0.043	0.083	0.073	0.085	0.045
29	8286		0.000	0.042	0.030	0.043	0.000

Table 2 shows the mathematical definitions for all these non-parametric methods which estimate the reliability function.

As an example, we show in Table 3, the data series of *code E* component to understand the TTF construction and how this data can be used with complete and censored data. This engineered component E has 21 TTFs (see Table 1). During its failure process investigation, 29 TTFs were available: six of them were censored times and in three failed after 2202 time units (see times $i = \{11, 12, 15\}$ in Table 3).

4.2 Experimental setup

A k-fold cross validation technique is used to compare the accuracy of the machine learning algorithms described in Section 3. k-fold cross validation is a statistical technique which evaluates how the results of a statistical analysis will generalize to k independent subsets of the main dataset [14]. First, data is randomly split into the k subsets. At each k iteration the test set is composed of a single subset, while the training set is composed of the remaining $k-1$ subsets. This procedure is repeated k times. In this paper, we use its classical variant, tenfold cross validation (10cv) [63].

The used fitting evaluation measure is the mean squared error (MSE) which will show the reliability fitting performance of the machine learning method. MSE computes the sum of the squares of the output differences at the i^{th} time unit between the value of the approximate failure distribution and the output from the machine learning models ($\widehat{R}(t_i)$) (see Eq. 1 for the mathematical definition of the MSE). Obviously, the smaller the MSE value is, the better the machine learning method performs as MSE measures the deviation between the non-parametric estimation and the predicted values by the machine learning methods.

$$\text{MSE} = \frac{1}{n} \cdot \sum_{i=0}^{n} (R(t_i) - \widehat{R}(t_i))^2. \tag{1}$$

Table 4 Parameters of the five machine learning algorithms used in the experimentation

Algorithm	Parameters	Values
LMS	None	
SVR	– Kernel type	Radial
	– C	0.5
	– Degree	3
	– γ	0.15
	– Coefficient	0
	– Tolerance of termination criterion	0.001
	– ϵ	0.1
RF	– Number of trees	1000
	– Size of a node	1
	– Number of variables randomly sampled at each split	1
MLP	– Number of hidden nodes	10
	– Iterations (conjugate gradient)	10,000
EFS-SA	Fuzzy system parameters	
	– Number of labels	3
	– Number of rules	8
	Simulated annealing parameters	
	– δ	0.5
	– Iterations	10,000
	– Mutation probability	0.5
	– Mutation amplitude	0.1
	– Initial probability for accepting	0.5
	– Final probability for accepting	0.5
	– Individuals to be analyzed at each iteration	10

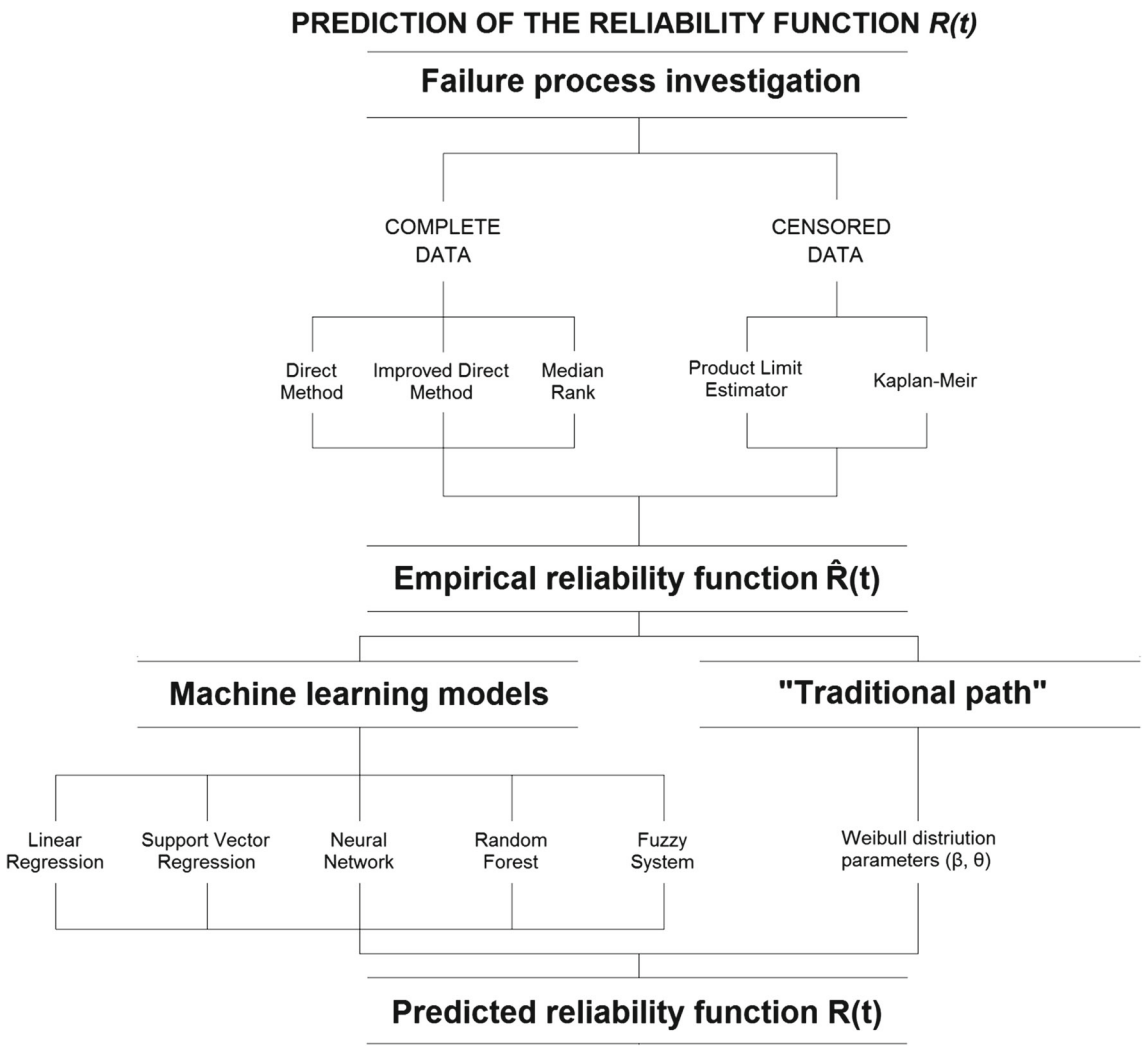


Fig. 2 Methods to estimate the reliability function $R(t)$ and the data-driven models used in the experimentation of this work

The values of the parameters for the five machine learning methods are presented in Table 4. These parameters were used during the whole experimentation and are set to the default values of the different implementations of the algorithms not to fit the methods to the problem in an unfair comparison. LMS, SVR, and RF were implemented using an R library while we used the KEEL [64] implementation for the last two methods, MLP and EFS-SA. We run all the experiments on an Intel i5-2400, 3.1 GHz processor and 4 GBytes of memory, under the Linux operating system.

Additionally, we have included the traditional two-parameter Weibull reliability estimation [10, 65] to compare the performance of the machine learning algorithms with it. The experimentation was carried out by using two non-parametric estimation functions. We used the MRM method for uncensored data and the PLE for censored data. The MRM method was applied to all the components, while the PLE method just to the components with censored data[1] (see Table 1 for checking the components with censored and complete data).

To sum up, we show in Fig. 2 a summary with the followed failure process and a list of the machine learning models used during the experimentation.

5 Reliability prediction results

In this section, we present the obtained results of the machine learning models on the dataset of 19 engineered components. We first explore the results with complete data

[1] We did not include the rest of the non-parametric estimations (i.e., DM, IDM, and KM) in the final experimentation of the paper as we did not find differences between their values.

Table 5 MSE results of the machine learning models and two-parameter Weibull reliability estimation for complete data (MRM method)

	LMS	SVR	RF	MLP	EFS-SA	Weibull
COD A	0.0015	0.0006	**0.0001**	0.0006	0.0012	0.0003
COD B	0.0006	0.0011	**0.0001**	0.0006	0.0009	0.0028
COD C	**0.0021**	0.009	0.0026	0.0023	0.0017	0.0036
COD D	**0.0009**	0.0032	0.001	0.001	0.0018	0.0011
COD E	0.0147	0.0104	0.0032	0.0029	0.0029	**0.0028**
COD F	0.0159	0.0021	**0.0006**	0.0008	0.0007	0.0021
COD H	0.0071	0.0046	0.0021	**0.0015**	0.0026	0.0044
COD I	0.0022	0.0035	0.0020	0.0014	0.0019	**0.001**
COD L	0.0053	0.001	**0.0005**	0.0015	0.0055	0.0006
COD M	0.0097	0.0055	**0.0012**	0.0051	0.0082	0.0061
COD N	0.0011	0.0009	**0.0002**	0.0008	0.0032	0.0021
COD O	0.0011	0.0017	0.001	**0.0008**	0.0009	0.0015
COD P	0.0179	0.0290	0.0075	**0.0027**	0.0063	0.0176
COD Q	0.0279	0.0093	**0.0018**	0.0179	0.0068	0.0059
COD R	0.0113	0.0089	**0.002**	0.0031	0.0027	0.0085
COD S	0.0096	0.0105	0.0035	**0.0031**	0.0017	0.0039
COD T	0.0183	0.0082	**0.0005**	0.0042	0.0048	0.0051
COD U	0.0002	0.0003	**0**	0.0001	0.0019	0.0023
COD V	0.0002	0.0003	**0**	0.0001	0.0004	0.0014
Average	0.0078	0.0058	**0.0016**	0.0027	0.0030	0.0038

(Section 5.1). Later, we do the same when data is censored (Section 5.2). Finally, we study the influence of the TTF size in the models' performance (Section 5.3).

5.1 Analysis of results with complete data

We analyze the experiments obtained using machine learning models with respect to the two-parameter Weibull reliability estimation for the MR method [6] (complete data). Table 5 presents the results obtained for all the industrial components averaged in the tenfold validation results. These values are the errors obtained by the MSE metric for the algorithms considered (lower error values indicate a better performance). Also, Table 5 shows the average error on the 19 components in the last row (highlighted by a gray filling). For each component, the best-performing result is in bold.

We can observe from these results that the majority of the machine learning models obtain better results than the classical method for the reliability prediction based on the two-parameter Weibull reliability estimation (averaged MSE value of 0.0038). The best-performing models are the RF, MLP, and EFS-SA with averaged MSE values of 0.0016, 0.0027, and 0.003, respectively. However, two machine learning models, SVR and the traditional LMS, do

Table 6 Results of the machine learning models and two-parameter Weibull reliability estimation for censored data (PLE method)

	LMS	SVR	RF	MLP	EFS-SA	Weibull
COD A	0.0011	0.0006	**0.0001**	0.0003	0.0012	0.0003
COD B	0.0005	0.0012	**0.0001**	0.0005	0.0005	0.0023
COD D	0.001	0.0026	**0.0009**	**0.0009**	**0.0009**	0.0011
COD E	0.0136	0.0093	0.003	**0.0025**	0.0033	0.0025
COD I	0.0019	0.0032	0.0019	**0.001**	0.0017	0.0009
COD L	0.0056	0.001	**0.0004**	0.0008	0.0051	0.0007
COD M	0.0095	0.0053	**0.0011**	0.0036	0.0094	0.0061
COD T	0.0206	0.0099	**0.0005**	0.0022	0.0061	0.007
COD U	0.0002	0.0003	**0**	0.0001	0.0006	0.0022
COD V	0.0002	0.0003	**0**	0.0002	0.0004	0.0013
Average	0.0054	0.0034	**0.0008**	0.0012	0.0029	0.0024

Table 7 Difference results of the machine learning and two-parameter Weibull reliability estimation models between using the PLE (censored) and the MRM (complete) methods

	LMS	SVR	RF	MLP-CG	EFS-SA	Weibull
COD A	**−0.0005**	0	0	**−0.0003**	0	0
COD B	**−0.0001**	0.0001	0	**−0.0001**	**−0.0004**	**−0.0004**
COD D	**−0.0011**	**−0.0064**	**−0.0017**	**−0.0014**	**−0.0008**	**−0.0025**
COD E	0.0127	0.0061	0.0020	0.0015	0.0015	0.0014
COD I	**−0.0128**	**−0.0072**	**−0.0012**	**−0.0019**	**−0.0012**	**−0.0019**
COD L	**−0.0103**	**−0.0011**	**−0.0002**	0	**0.0044**	**−0.0014**
COD M	0.0023	0.0007	**−0.001**	0.0021	0.0068	0.0017
COD T	0.0184	0.0063	**−0.0016**	0.0008	0.0042	0.0061
COD U	**−0.0051**	**−0.0007**	**−0.0005**	**−0.0014**	**−0.0049**	0.0016
COD V	**−0.0094**	**−0.0053**	**−0.0012**	**−0.0049**	**−0.0078**	**−0.0048**
Average	**−0.0006**	**−0.0007**	**−0.0005**	**−0.0006**	**0.0002**	0

not improve the prediction provided by the two-parameter Weibull reliability estimation.

The LMS, which is a simple regression algorithm, obtains the worst reliability prediction results on average. In contrast, RF confirms its high potential when solving this prediction problem by obtaining the lowest accuracy results on average (0.0016). Furthermore, RF individually outperforms the rest techniques in 11 out of 19 industrial component cases. The ANN variant, i.e., the MLP with a conjugate gradient algorithm, is the second best-performing model and obtains the best results in four cases.

5.2 Analysis of results with censored data

In this subsection, we focus the analysis and compare the results obtained by the machine learning algorithms and the two-parameter Weibull reliability estimation method using censored set of data (using the PLE method). Later on, we will also compare the results obtained for the censored data (the PLE method) with the results of the complete data (the MRM method), discussed in previous Section 5.1.

Table 6 presents the results obtained for censored data by showing the MSE metric values for all the algorithms considered. Again, the best result for a given component dataset is presented in bold font. The last row of the table presents the average values over all the components datasets.

The prediction results for the PLE method show a similar trend with respect to when using complete data (the MRM method shown in Section 5.1). Two machine learning approaches, the RF and MLP, obtain better accuracy results (0.0008 and 0.0012, respectively) than the traditional two-parameter Weibull reliability estimation method on average (0.0024). Again, the statistical LMS is the worst-performing approach on average (an averaged error value of 0.0054). RF turns out to be the best-performing approach both in average and individually by components. RF outperforms the other approaches in eight out of ten cases. As in the previous section, when dealing with complete data, the ANN variant, the MLP with a conjugate gradient algorithm, is the second best-performing algorithm.

In order to clearly see differences between using complete or censored data, Table 7 presents the difference

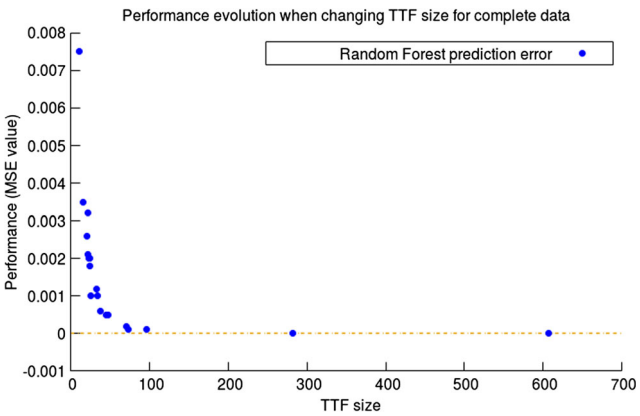


Fig. 3 Evolution of the RF model performance when increasing the number of TTF for complete data

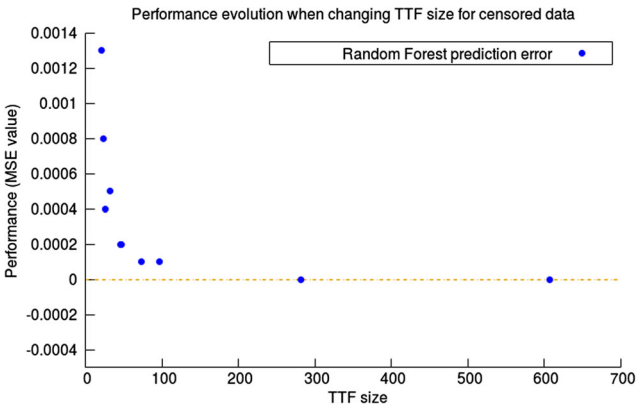


Fig. 4 Evolution of the RF model performance when increasing the number of TTF for censored data

between the results obtained for PLE (censored) and MRM (uncensored) methods for each algorithm. The negative values (in bold font) represent when machine learning models and two-parameter Weibull reliability estimation better perform for censored data. It means they obtain a lower MSE metric value for the same component when having censored data. In contrast, positive values represent the same behavior but for complete data.

In 36 out of 50 cases, we see that the models obtain better behavior when using censored data (i.e., negative values in Table 7). In six cases, there is no difference between both censored and complete data. All the machine learning models but EFS-SA obtain better results when using censored data (PLE method) than when using complete data (MRM method). The clearest case is the best-performing machine learning algorithm, RF, which almost always obtains better or equal results when using censored data. Its censored data results are only worse in the case of industrial component E.

Additionally, if we look at the averaged values (last row of Table 7), we find no difference between the accuracy of censored and complete data for the two-parameters Weibull reliability estimation. Four of the five machine learning models obtain better accuracy when using censored data on the averaged metrics but EFS-SA.

5.3 Impact of the dataset size on the models' performance

In this section, we will evaluate the impact of the number of TTF on the performance of the machine learning models. The goal is to find out if models perform better when having industrial components with a higher number of failure times available. First, we plot in Figs. 3 and 4 the performance, measured by the MSE values, of the RF model (the best-performing machine learning algorithm). Horizontal axis of both graphs is the number of TTF of the datasets while vertical axis is the MSE values obtained by the machine learning method for the different components.

We can see that the RF performance increases when the number of TTF also increases. Both graphs of Figs. 3 and 4

Table 8 Results of the components' clusters based on the dataset size (number of TTF) for the MRM method (complete data)

	TTF	MAX-MIN	LMS	SVR	RF	MLP	EFS-SA	Weibull
COD P	11	4506	0.0179	0.0290	0.0075	**0.0027**	0.0063	0.0176
COD S	15	2511	0.0096	0.0105	0.0035	0.0031	**0.0017**	0.0039
COD C	20	6065	0.0021	0.0090	0.0026	0.0023	**0.0017**	0.0036
COD E	21	7151	0.0147	0.0104	0.0032	0.0029	0.0029	**0.0028**
COD H	22	2373	0.0071	0.0046	0.0021	**0.0015**	0.0026	0.0044
COD I	23	5564	0.0022	0.0035	0.0020	0.0014	0.0019	**0.0010**
COD Q	24	3583	0.0279	0.0093	**0.0018**	0.0179	0.0068	0.0059
COD R	24	5521	0.0113	0.0089	**0.0020**	0.0031	0.0027	0.0085
Average	–	–	0.0116	0.0107	**0.0031**	0.0044	0.0033	0.0060
COD D	25	8519	0.0009	0.0032	**0.0010**	**0.0010**	0.0018	0.0011
COD M	32	5519	0.0097	0.0055	**0.0012**	0.0051	0.0082	0.0061
COD O	34	5291	0.0011	0.0017	0.0010	**0.0008**	0.0009	0.0015
COD F	37	3010	0.0159	0.0021	**0.0006**	0.0008	0.0007	0.0021
COD T	45	5800	0.0183	0.0082	**0.0005**	0.0042	0.0048	0.0051
COD L	47	4207	0.0053	0.0010	**0.0005**	0.0015	0.0055	0.0006
Average	–	–	0.0085	0.0036	**0.0008**	0.0022	0.0037	0.0028
COD N	71	3602	0.0011	0.0009	**0.0002**	0.0008	0.0032	0.0021
COD A	73	8541	0.0015	0.0006	**0.0001**	0.0006	0.0012	0.0003
COD B	96	2542	0.0006	0.0011	**0.0001**	0.0006	0.0009	0.0028
Average	–	–	0.0011	0.0009	**0.0002**	0.0007	0.0018	0.0017
COD V	282	3367	0.0002	0.0003	**0.0000**	0.0001	0.0004	0.0023
COD U	607	3423	0.0002	0.0003	**0.0000**	0.0001	0.0019	0.0014
Average	–	–	0.0002	0.0003	**0.0000**	0.0001	0.0012	0.0019

have the same performance evolution. The same behavior occurs when using the MRM method (complete data, Fig. 3) and the PLE method (censored data, Fig. 4). Therefore, there are no differences between using censored or complete data for this better performance behavior when the TTF size increases, as expected.

We will analyze the behavior of the machine learning models by defining dataset clusters to further study this scenario of dataset size variation. These clusters are created according to the size of the different available datasets. We divide all the industrial components datasets into four different clusters using TTF as the splitting criterion. The first cluster comprises components with TTF up to 25 (TTF< 25), the second one consists of the ones with TTF between 25 and 50 ($25 < TTF \leq 50$), the third one contains the components with TTF between 50 and 100 ($50 < TTF \leq 100$), and the fourth one includes the components with TTF higher than 100 (100<TTF).

Tables 8 and 9 present the results of the machine learning models and two-parameters Weibull reliability estimation for the four clusters considering the MRM and PLE methods, respectively. The values of the table show the number of TTF of each component of the clusters (second column), the difference between the maximum and minimum times of the components (third column) as well as the results of the different algorithms. The numerical values are the MSE (lower values mean a better performance).

Focusing on the first cluster, we can see that three machine learning approaches (RF, EFS-SA, and MLP) obtain lower accuracy results than the traditional two-parameters Weibull reliability estimation on average (0.0031, 0.0033, and 0.006, respectively). In the next clusters, RF gets a higher difference with respect to the other techniques. It means that RF works better when having larger datasets. For instance, RF is the best-performing method for all the components but one of the last three clusters. In other words, the higher TTF cluster, the better RF performs. The same happens for both complete (MRM method, see Table 8) and censored data (PLE method, Table 9).

Finally, one of the main important analysis from this study is that the two-parameters Weibull reliability estimation is the method with the lowest improvement when increasing the number of TTF. In fact, this reliability estimation is more competitive to deal with the prediction of the components of the first cluster (small datasets in terms of available TTFs). When increasing the number of TTFs, machine learning models such as RF and MLP present important performance differences with respect to the Weibull reliability estimation. This analysis makes our clustering study useful to better understand the need of different models depending on the life cycles of the real industrial components. Therefore, it is highly recommended to use machine learning methods when the number of TTF is relatively high. A reasonable threshold value for this decision can be 25 TTF. Differences between traditional distribution methods and machine learning models is not significant for components with low number of TTF available.

Table 9 Results of the components' clusters based on the dataset size (number of TTF) for the PLE method (censored data)

	TTF	MAX-MIN	LMS	SVR	RF	MLP	EFS-SA	Weibull
COD E	21	7151	0.0136	0.0093	0.0030	**0.0025**	0.0033	0.0025
COD I	23	5564	0.0019	0.0032	0.0019	**0.001**	0.0017	0.0009
Average	–	–	0.0078	0.0062	0.0025	**0.0018**	0.0025	0.0017
COD D	25	8519	0.0010	0.0026	**0.0009**	**0.0009**	**0.0009**	0.0011
COD M	32	5519	0.0095	0.0053	**0.0011**	0.0036	0.0094	0.0061
COD T	45	5800	0.0206	0.0099	**0.0005**	0.0022	0.0061	0.0070
COD L	47	4207	0.0056	0.0010	**0.0004**	0.0008	0.0051	0.0007
Average	–	–	0.0092	0.0047	**0.0007**	0.0019	0.0054	0.0037
COD A	73	8541	0.0011	0.0006	**0.0001**	0.0003	0.0012	0.0003
COD B	96	2542	0.0005	0.0012	**0.0001**	0.0005	0.0005	0.0023
Average	–	–	0.0008	0.0009	**0.0001**	0.0004	0.0009	0.0013
COD V	282	3367	0.0002	0.0003	**0**	0.0002	0.0004	0.0013
COD U	607	3423	0.0002	0.0003	**0**	0.0001	0.0006	0.0022
Average	–	–	0.0002	0.0003	**0.0000**	0.0002	0.0005	0.0018

6 Conclusions

Reliability prediction is fundamental in estimating engineering system performance. An effective estimation of $R(t)$ permits the appropriate application of several models of maintenance policies of components. And then, it has their significant benefits in the total cost of the ownership of a complex system (e.g., manufacture machine). In this paper, we applied five machine learning models to estimate the reliability $R(t)$ of a large dataset of mechanical components. In particular, we used a set of 19 real cases, representative of a wide range of components behavior, to try to understand the ability of the proposed models to capture the complexities of the reliability behavior by approximating the reliability function of industrial components over time $R(t)$.

This paper analyzed how machine learning models behave with respect to the two-parameters Weibull reliability estimation, the traditional method used in the industry. In particular, we concluded that the random forest (RF) algorithm is the best model as average and in the majority of the industrial components of the datasets, obtaining high accuracy values for the engineered components and without performing a calibration for the parameters' values of the machine learning methods. Additionally, the results showed that prediction models can get better results in presence of censored data than when using complete data.

We also carried out a study about the implications of having different number of available TTF. To do this, we compared the behavior of the best-performing algorithm, i.e., RF, when increasing the number of TTF, and grouped the datasets by size-based clusters. The main conclusion of this experimentation was that machine learning models obtained better results when having more TTF data. Specifically, RF clearly was the best model in all the larger components (the three last clusters). We showed that differences between the traditional two-parameter Weibull reliability estimation and machine learning models are higher when having more TTF data. Therefore, it is even more recommended to use machine learning models with respect to traditional approaches in datasets with more than 25 TTF.

Our research study showed the potential of the machine learning approach for industrial reliability analysis. For instance, our results can be also an useful contribution in the field of maintenance engineering, helping to define some threshold values which trigger certain maintenance action. Future works will focus on using more advanced machine learning models such as those using aggregation mechanisms [5]. Additionally, we could explore if the used models are able to generalize data about specific industrial components and predict their reliability under different industrial conditions.

References

1. Birolini A (2007) Reliability engineering, vol 5. Springer, Berlin
2. Shi H, Zeng J (2016) Real-time prediction of remaining useful life and preventive opportunistic maintenance strategy for multi-component systems considering stochastic dependence. Comput Ind Eng 93:192–204
3. Luo M, Yan HC, Hu B, Zhou JH, Pang CK (2015) A data-driven two-stage maintenance framework for degradation prediction in semiconductor manufacturing industries. Comput Ind Eng 85:414–422
4. Sun Y, Ma L, Mathew J (2009) Failure analysis of engineering systems with preventive maintenance and failure interactions. Comput Ind Eng 57(2):539–549
5. Rodger JA, George JA (2017) Triple bottom line accounting for optimizing natural gas sustainability: a statistical linear programming fuzzy ILOWA optimized sustainment model approach to reducing supply chain global cybersecurity vulnerability through information and communications technology. J Clean Prod 142:1931–1949
6. Ebeling CE (2010) An introduction to reliability and maintainability engineering. Waveland Press, Long Grove
7. O'Connor P, Kleyner A (2011) Practical reliability engineering. Wiley, Hoboken
8. Ahmad R, Kamaruddin S (2012) An overview of time-based and condition-based maintenance in industrial application. Comput Ind Eng 63(1):135–149
9. Crocker J, Kumar UD (2000) Age-related maintenance versus reliability centred maintenance: a case study on aero-engines. Reliab Eng Syst Saf 67(2):113–118
10. Manzini R, Regattieri A, Pham H, Ferrari E (2009) Maintenance for industrial systems. Springer Science & Business Media, Berlin
11. Nakagawa T (2006) Maintenance theory of reliability. Springer Science & Business Media, Berlin
12. das Chagas Moura M, Zio E, Lins ID, Droguett E (2011) Failure and reliability prediction by support vector machines regression of time series data. Reliab Eng Syst Saf 96(11):1527–1534
13. Bishop CM (2006) Pattern recognition and machine learning. Springer, Berlin
14. Witten IH, Frank E (2005) Data mining: practical machine learning tools and techniques, 2nd edn. Morgan Kaufmann, San Francisco
15. Amjady N, Ehsan M (1999) Evaluation of power systems reliability by an artificial neural network. IEEE Trans Power Syst 14(1):287–292
16. Chatterjee S, Bandopadhyay S (2012) Reliability estimation using a genetic algorithm-based artificial neural network: an application to a load-haul-dump machine. Expert Syst Appl 39(12):10,943–10,951
17. Hong WC, Pai PF (2006) Predicting engine reliability by support vector machines. Int J Adv Manuf Technol 28(1–2):154–161
18. Gokulachandran J, Mohandas K (2015) Comparative study of two soft computing techniques for the prediction of remaining useful life of cutting tools. J Intell Manuf 26(2):255–268
19. Wu X, Zhu Z, Fan S, Su X (2016) Failure and reliability prediction of engine systems using iterated nonlinear filters based state-space least square support vector machine method. Optik Int J Light Electron Opt 127(3):1491–1496
20. Regattieri A, Manzini R, Battini D (2010) Estimating reliability characteristics in the presence of censored data: a case study in a light commercial vehicle manufacturing system. Reliab Eng Syst Saf 95(10):1093–1102
21. Kumar UD, Crocker J, Knezevic J, El-Haram M (2012) Reliability, maintenance and logistic support: a life cycle approach. Springer, Berlin

22. Meeker WQ, Escobar LA (1998) Statistical methods for reliability data, vol 314. Wiley-Interscience, Hoboken
23. Nelson WB (2009) Accelerated testing: statistical models, test plans, and data analysis, vol 344. Wiley, Hoboken
24. Jardine AK, Tsang AH (2013) Maintenance, replacement, and reliability: theory and applications. CRC Press, Boca Raton
25. Bontempi G, Taieb SB, Le Borgne YA (2013) Machine learning strategies for time series forecasting. In: Business intelligence. Springer, Berlin, pp 62–77
26. Vanini ZS, Khorasani K, Meskin N (2014) Fault detection and isolation of a dual spool gas turbine engine using dynamic neural networks and multiple model approach. Inf Sci 259:234–251
27. Ogaji SO, Singh R (2003) Advanced engine diagnostics using artificial neural networks. Appl Soft Comput 3(3):259–271
28. Widodo A, Yang BS (2007) Support vector machine in machine condition monitoring and fault diagnosis. Mech Syst Signal Process 21(6):2560–2574
29. Liu MC, Kuo W, Sastri T (1995) An exploratory study of a neural network approach for reliability data analysis. Qual Reliab Eng Int 11(2):107–112
30. Alsina EF, Cabri G, Regattieri A (2016) A neural network approach to find the cumulative failure distribution: modeling and experimental evidence. Qual Reliab Eng Int 32(2):167–579
31. Ho S, Xie M, Goh T (2002) A comparative study of neural network and Box-Jenkins ARIMA modeling in time series prediction. Comput Ind Eng 42(2):371–375
32. Xu K, Xie M, Tang LC, Ho S (2003) Application of neural networks in forecasting engine systems reliability. Appl Soft Comput 2(4):255–268
33. Fink O, Zio E, Weidmann U (2014) Predicting component reliability and level of degradation with complex-valued neural networks. Reliab Eng Syst Saf 121:198–206
34. Lins ID, Moura MdC, Zio E, Droguett EL (2012) A particle swarm-optimized support vector machine for reliability prediction. Qual Reliab Eng Int 28(2):141–158
35. Chen KY (2007) Forecasting systems reliability based on support vector regression with genetic algorithms. Reliab Eng Syst Saf 92(4):423–432
36. Pai PF, Lin KP (2006) Application of hybrid learning neural fuzzy systems in reliability prediction. Qual Reliab Eng Int 22(2):199–211
37. Zadeh LA (1994) Fuzzy logic, neural networks, and soft computing. Commun ACM 37(3):77–84
38. Draper NR, Smith H (2014) Applied regression analysis. Wiley, Hoboken
39. Marsland S (2015) Machine learning: an algorithmic perspective. CRC Press, Boca Raton
40. Jang JSR, Sun CT, Mizutani E (1997) Neuro-fuzzy and soft computing: a computational approach to learning and machine intelligence. Prentice Hall, Upper Saddle Rive
41. Alzghoul A, Löfstrand M, Backe B (2012) Data stream forecasting for system fault prediction. Comput Ind Eng 62(4):972–978
42. Rustagi J (1994) Optimization techniques in statistics. Academic Press, Cambridge
43. Smola AJ, Schölkopf B (2004) A tutorial on support vector regression. Stat Comput 14(3):199–222
44. Cortes C, Vapnik V (1995) Support vector networks. Mach Learn 20:273–297
45. Dietterich TG (2000) Ensemble methods in machine learning. In: Multiple classifier systems. Springer, Berlin, pp 1–15
46. Kuncheva L (2001) Combining classifiers: soft computing solutions. In: Pal S, Pal A (eds) Pattern recognition. From classical to modern approaches. World Scientific, Singapore, pp 427-451
47. Breiman L (2001) Random forests. Mach Learn 45(1):5–32
48. Breiman L (1996) Bagging predictors. Mach Learn 24(2):123–140
49. Haykin S (1998) Neural networks: a comprehensive foundation, 2nd edn. Prentice Hall PTR, NJ
50. Moller F (1990) A scaled conjugate gradient algorithm for fast supervised learning. Neural Netw 6:525–533
51. Zadeh LA (1965) Fuzzy sets. Inf Control 8(3):338–353
52. Yager RR, Filev DP (1994) Essentials of fuzzy modeling and control. Wiley-Interscience, New York
53. Casillas J, Cordon O, Herrera F, Magdalena L (2003) Interpretability issues in fuzzy modeling. Springer, Berlin
54. Alonso JM, Magdalena L, González-Rodríguez G (2009) Looking for a good fuzzy system interpretability index: an experimental approach. Int J Approx Reason 51:115–134
55. Cordón O, Herrera F, Hoffmann F, Magdalena L (2001) Genetic fuzzy systems. Evolutionary tuning and learning of fuzzy knowledge bases. World Scientific, Singapore
56. Cordón O, Gomide F, Herrera F, Hoffmann F, Magdalena L (2004) Ten years of genetic fuzzy systems: current framework and new trends. Fuzzy Sets Syst 141(1):5–31
57. Herrera F (2008) Genetic fuzzy systems: taxonomy, current research trends and prospects. Evol Intell 1:27–46
58. Sánchez L, Couso I, Corrales J (2001) Combining GP operators with SA search to evolve fuzzy rule based classifiers. Inf Sci 136(1-4):175–191
59. Kirkpatrick S, Gelatt CD, Vecchi MP (1983) Optimization by simulated annealing. Sci 220(4598):671–680
60. Koza JR (1992) Genetic programming: on the programming of computers by means of natural selection, vol 1. MIT Press, Cambridge
61. Ferrari E, Pareschi A, Regattieri A, Persona A (2002) TPM: situation and procedure for a soft introduction in Italian factories. TQM Mag 14(6):350–358
62. Kaplan EL, Meier P (1958) Nonparametric estimation from incomplete observations. J Am Stat Assoc 53(282):457–481
63. Kohavi R (1995) A study of cross-validation and bootstrap for accuracy estimation and model selection. In: Proceedings of the 14th international joint conference on artificial intelligence, IJCAI'95, vol 2. Morgan Kaufmann Publishers Inc., San Francisco, pp 1137–1143
64. Alcalá-Fdez J, Sanchez L, Garcia S, del Jesus MJ, Ventura S, Garrell JM, Otero J, Romero C, Bacardit J, Rivas VM et al (2009) Keel: a software tool to assess evolutionary algorithms for data mining problems. Soft Comp 13(3):307–318
65. Wang ZM, Yang JG (2012) Numerical method for Weibull generalized renewal process and its applications in reliability analysis of nc machine tools. Comput Ind Eng 63(4):1128–1134