Available online at www.sciencedirect.com

ScienceDirect

Procedia Manufacturing 42 (2020) 41–48

Procedia MANUFACTURING

www.elsevier.com/locate/procedia

International Conference on Industry 4.0 and Smart Manufacturing (ISM 2019)

An ensemble-learning model for failure rate prediction

Braglia Marcello[a], Castellano Davide[b], Frosolini Marco[a]*, Gabbrielli Roberto[a],
Marrazzini Leonardo[a], Padellini Luca[c]

[a]*Dipartimento di Ingegneria Civile e Industriale - Università di Pisa, Largo Lucio Lazzarino 2, 56126 Pisa, Italy*
[b]*Dipartimento di Ingegneria Chimica, dei Materiali e della Produzione Industriale - Università di Napoli Federico II, Piazzale Tecchio 80, 80125 Napoli,Italy*
[c]*Dipartimento di Ingegneria dell'Informazione - Università di Pisa, Via Girolamo Caruso 16, 56122 Pisa, Italy*

* Corresponding author. Tel.: +39 050 2218139; fax: +39 050 2218140. *E-mail address:* m.frosolini@ing.unipi.it

Abstract

In the Industry 4.0 era, Preventive Maintenance (PM) is still an attractive solution to prevent breakdowns and failures and to reduce maintenance and failure costs. A PM program is part of both the Total Productive Maintenance (TPM) philosophy and the Reliability Centered Maintenance (RCM) process. A prerequisite to carry out effective PM activities is the availability of a reliable estimate of the equipment failure rate. Assessing it may be a hard task, as it requires analysing a large set of maintenance data, which includes both quantitative and qualitative variables. To this aim, it is possible to exploit advanced data analysis techniques that permit extracting information and knowledge from big datasets. This paper presents an ensemble-learning model to estimate the failure rate of equipment subject to different operating conditions. At the same time, the method permits to identify the most important working parameters affecting the failure rate. An industrial application is considered to show the potentialities and the effectiveness of the proposed method. In particular, a sample of 143 centrifugal pumps installed in an oil refinery plant is analysed.

Keywords: Industry 4.0; Ensemble learning; Preventive maintenance; Failure rate

1. Introduction

The Industry 4.0 paradigm and the related technologies are currently under the spotlight of both academicians and practitioners [1]. The motivation is that the manufacturing industry is facing the so-called "data-driven revolution". The digitalization process that converts traditional factories into smart factories has given rise to an enormous growth of data production [3]. Smart organizations must challenge to record and manage such "big data", to extract meaningful information from them by means of appropriate analytical techniques and tools.

In this Industry 4.0 era, preventive maintenance (PM) is a still an attractive solution to prevent breakdowns and failures and to reduce maintenance and failure costs [4]. A PM program is part of both the Total Productive Maintenance (TPM) philosophy and the Reliability Centred Maintenance (RCM) process [6]. The primary objective of PM is to prevent failures before they occur. Comprehensive PM programs schedule repairs, lubrication, adjustments and machine rebuilds for all critical plant machinery. In order to support the work of maintenance experts, good PM practices require that all available data regarding failures is recorded into a well-organized database. All PM programs assume that machines will degrade within a time frame specific of their peculiar working conditions. Clearly, a reliable evaluation of the equipment failure rate permits to carry out effective PM programs. On the other hand, the mode of operation and system- or plant-specific variables directly affect the operating life of each equipment. This means that the failure rate depends on several factors whose identification and

10.1016/j.promfg.2020.02.022

quantification is a challenging task [7].

As observed by [8], failure rates can be estimated from empirical data and formulas available in handbooks, manufacturers' data, industry standards, MIL-standards, and so on. If quantitative data are not available, experts can make use of their own experience to assess values. However, all these methods are characterized by many limits. Failure rate formulas provided by handbooks or regulations are complex and valid only under very special operating conditions, which are seldom satisfied in practice. Suggested failure rates by manufacturers may be conservative, which may lead to excessive maintenance. Probabilistic approaches to estimating the time to failure (e.g., proportional hazards model) appear to be too complex to be applied in most practical situations, and the required statistical tools and competencies are hardly available in industry.

To overcome these issues, some researchers have proposed practical non-parametric approaches for failure rate analysis. [9] used the classification and regression tree (CART) methodology. [10] developed an approach based on artificial neural networks (ANNs). More recently, [11] presented a multivariate data classification technique.

The present work belongs to the same stream of research, but it exploits the advances in big data analytics to tackle two important tasks required to carry out an effective PM program: (i) estimating the failure rate of equipment subject to different operating conditions and (ii) identifying the most important working parameters affecting the failure rate. These activities can evidently take advantage of analytic techniques able to treat stream of "big data" as they require analysing a large set of maintenance data collected into a computerized maintenance management system (CMMS), which includes both quantitative and qualitative variables.

In order to accomplish these tasks, this paper proposes an ensemble-learning model that combines prediction results from multiple algorithms. Although ensemble-learning methods have been used to approach a variety of different problems, and the associate literature is rather ample [12], we have not found any contribution focused on using this technique to estimate the failure rate of equipments subject to different operating conditions and to discriminate the working parameters affecting the failure rate. Hence, the present work aims to fill this gap. Note that we do attempt to develop neither a mathematical relation between failure rate and operating conditions nor lifetime distributions. Rather, our intention is to provide an easy-to-use approach that can be implemented in practice by means of a free software. Specifically, an industrial application is considered to show the potentialities and the effectiveness of the proposed method. In particular, a sample of 143 centrifugal pumps installed in an oil refinery plant is analysed.

2. Basic Concepts of Ensemble Modelling

Ensemble modelling refers to the use of multiple learning algorithms to obtain better predictive capabilities than those obtained from any of the basic, constituting learning methods. Ensemble model works better when the original models present low correlation [13]. This happens because each one of the different algorithms may better contribute with its own strengths. Typically, in predictive modelling and in data analytics in general, a single model is used on a given data sample. The dataset can be large and rich, without missing values and errors. However, the model often presents biases, high variability and inaccuracies that affect the overall reliability of its conclusions. Often, this is because some algorithms, though extremely powerful under given hypotheses and circumstances, suffer from the presence of previously unseen examples within the studied datasets. The same effect is introduced by outliers and rare values. On the contrary, an ensemble investigates the dataset using all its constituent algorithms, allowing each one of them to support the others in the case of dubious outcomes. A famous example is represented by the random forest of trees [15]. This algorithm builds numerous decision trees while training and gives, as the output, a single class that is the mode of the corresponding classes of the individual trees. Doing so, the method avoids the known overfitting behaviour of the original trees.

Building an ensemble requires building different models and combining their estimates. The building stage may be accomplished, for instance, changing weights, data values, control parameters, variable subsets, or partitions of the input datasets. Bagging, short for Bootstrap Aggregating model, [13] uses the training dataset to build different decision trees and, finally, takes the majority vote or the average of their estimates. Random Forests [15] add stochastic components to increase diversity among the trees being combined. AdaBoost [16] builds models iteratively, changing case weights, and uses the weighted sum of the estimates. Good ensembles should present both accuracy and simplicity. However, to reach higher accuracy, models tend to become extremely complex. While doing so, they are exposed to the risk of overfitting and poor generalization capabilities. Regularization techniques have been introduced to reduce the complexity of the model fitting procedures and have shown over time to allow for extremely effective ensemble models [17].

3. Case study

In this section, a case study is presented in order to illustrate the creation of the ensemble model proposed in this work which is used to estimate the failure rate of a set of centrifugal pumps subject to different operating conditions and to identify the most important parameters affecting the failure rate.

3.1. Brief overview of the refinery plant

The industrial application concerns several centrifugal pumps installed in an oil refinery plant. The plant performs the entire petrochemical cycle: crude oil supply, refinery process, and distribution of finished products. The transformation process adopts a "medium-high" conversion type, operating through the adoption of thermal process. Fig. 1 provides the oil refinery-processing scheme.

The refinery is characterized by processing and service systems which occupy a surface of nearly 650,000 m^2 with 3,000 km of piping. The plant has a storage capacity of more than 1,500,000 m^3, an annual production capability of about 390,000 tons of oil, and an oil tanker receiving capability up to 400,000 tons displacement. A closed-loop water system capable of delivering 700 m^3/h of water and a fire system able to bring in up to 3,000 m^3/h of seawater are included in the plant. An integrated combined cycle plant assures the necessary power supply capable of about 280 MW, which operates by burning a synthesis gas obtained in the refining cycle.

After a primary distillation phase, which is known as *topping*, the materials flow into two separate distillation units:

- The atmospheric distillation unit (i.e., unifining) treats light fractions by separating petrol from liquefied petroleum gas (LPG). Then, petrol undergoes two further transformation phases (i.e., platforming and isomerization) required to increase the octane number and to eliminate aromatic compounds.
- The vacuum unit treats the middle distillation fraction (mainly kerosene) and feeds a desulphurization process.

Finally, the heavy fraction and all distillation residuals are processed by means of *thermal cracking* and *visbreaking*. These treatments are required to improve oil conversion rate and to increase the overall production of lighter products.

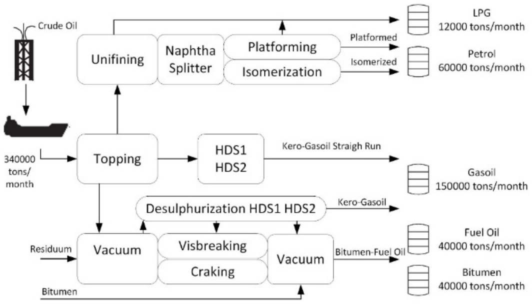


Fig. 1. Oil refinery processing scheme.

3.2. Structure

The analysis considered 143 centrifugal pumps installed in the oil refinery plant described in Section 2.1. These pumps were monitored over a period of 18 months. In this period, operating time, failures and maintenance tasks were recorded in a standard computerized maintenance management system (CMMS). Table 1 shows a sample of the records included into the CMMS. Within the database, ten potential predictors were identified. While the name of some variables is self-explanatory, some others require a brief description. "Plant type" defines the part of the refinery plant where the pump operates. "Soot" identifies the solid carbon-based particles present in the fluid, which typically have disruptive actions on the seals. Finally, we clustered the seal type into four categories: single-seal (S), dual-seal (DS), lip-seal (SL), and tandem-seal (T).

The net operating time (NOT) included in Table 1 is defined as the cumulated functioning time, from the start up until the last observed failure. In formulas, the net operating time of the *j*th pump, NOT_j, can be expressed as follows:

$$NOT_j = \sum_{i=1}^{f_j} \left(FT_{ij} - CT_{(i-1)j}\right) \cong \sum_{i=1}^{f_j} \left(OT_{ij} - CT_{(i-1)j}\right. \tag{1}$$

where f_j is the number of observed failures for the *j*th pump, FT_{ij} is the failure time of failure *i* on pump *j*, OT_{ij} and CT_{ij} are the opening time and the closing time, respectively, of the maintenance order regarding failure *i* on pump *j*.

The exact values of FT_{ij} were not available in the database. Hence, we replaced FT_{ij} with OT_{ij} in the evaluation of the net operating time. This approximation can be justified according to the following argument. Reactive maintenance is the first option, while condition maintenance applies only to the pumps that do not operate in active redundancy. However, even in this case, when an operating threshold limit is trespassed (e.g., excessive vibration and/or leakage) and a maintenance order is issued, the remaining useful life of the pump can be considered nearly null. Hence, the approximation $FT_{ij} \cong OT_{ij}$ can be assumed reasonable, and the MTBF is finally given by:

$$\text{MTBF} = \frac{\text{Net Operating Time}}{\text{Number of failures recorded}}$$

where the net operating time is obtained according to Eq. (1).

3.3. Ensemble model for failures analysis

To apply ensemble modelling to the pumps' dataset, the stacked generalization and the bagging methods were used. One of the most interesting aspects of stacking is that it may be used to combine models of different types. The most important goal of stacking is, possibly, the reduction of bias. To begin with, the algorithm splits the training set into two separate subsets and trains several learners on the first subset. The remaining data are used to test and validate the model but, instead of using a monolithic approach in which the best learner becomes the winner, the outcomes of all the models

Table 1. Example of data available in the CMMS.

	Potential predictors											**Failure modes and MTBF**						
Code	*Plant type*	*Service/ fluid*	*Seal type*	*Soot*	*Nominal capacity [m³/h]*	*Nominal head [m]*	*Nominal power [kW]*	*Fluid temp. [°C]*	*Cinematic viscosity [cSt]*	*Density [kg/m³]*	*Fluid leakages*	*Irregular working*	*Vibrations*	*Mechanical failures*	*Electrical failures*	*Total number of failures*	*Net operating time*	*MTBF [h]*
P1001	TOP1000	Crude oil	S	No	530	237	394.5	18	42	870	4	2	0	1	3	10	13127	1313
P1002	TOP1000	Residuum	T	Yes	231	173.5	127	360	1.17	801	4	0	1	0	0	5	12283	2457
P1003	TOP1000	Petroleum	S	No	63	117	22.5	187	0.46	670	8	0	0	0	0	8	15918	1990
…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…
P1010	TOP1000	Condensed water	S	No	7.7	180	27.8	38	0.69	992.6	2	2	0	0	0	4	11560	2890
P1011	TOP1000	Preheated gasoil	S	No	90	59	16	156	0.88	760	12	0	0	1	0	13	27817	2139
P1012	TOP1000	Acid water	S	No	28	190	43.4	16	1.11	998.5	1	0	1	0	2	4	11067	2767
…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…

are mixed, possibly in a nonlinear fashion, to get the most of all the algorithms. The weight used to mix the various models are obtained by means of a supplementary algorithm (usually a logistic regression process), that compares the outcomes with the inputs of all the used models. Therefore, the key point lies in the fact that all the models are compared and judged on subsets of data that were not used to create them.

Bagging is primarily used to decrease the variance of the prediction/classification. It works generating supplementary data for training the models starting from the original dataset. To do so, it opportunely combines data with repetitions to produce multisets of the same cardinality as the original data. Obviously, this method cannot improve the predictive capability just increasing the size of the training set, but it strongly decreases the variance, narrowly tuning the prediction itself.

The two models have been built by means of the educational licensed release of the RapidMiner Studio 9® software package. It is a data science software platform that provides an integrated environment for data preparation, machine learning and predictive analytics. It also supports all steps of the learning process (Fig. 2), following the cross-industry standard process for data mining (CRISP-DM), including data preparation, results visualization, model validation and optimization within an easy and user-friendly graphic interface.

To begin with, the original data have been saved as a comma separated value (CSV) file and have been imported into RapidMiner® workbench. The case study and the corresponding characteristics and aspects were all well-known from previous research activities and therefore, with respect to steps 1 and 2 of the CRISP-DM process, no additional information was necessary.

Referring to step 3, namely the data preparation, the dataset has been cleansed and some attributes (columns), that resulted incomplete or clearly wrong, have been removed from the dataset. The only significant modification to the original dataset was in the computation of the MTBF, expressed in hours. Indeed, some pumps had no failures over relatively long time periods. In such cases, it was decided to use a high numerical value to correctly represent the infinity. On the other hand, some pumps presented no failure at all, but their operating time was very short and, therefore, there was a great deal of uncertainty on their MTBF value. These pumps have been characterized by a MTBF equal to -1 and have been later removed from the analysis.

As a result, it was possible to pass immediately to step 4 and start to model the whole ensemble process. First, it was necessary to change the attribute role of the column "Code" from "regular" to "id", meaning that this field should be considered only for identification purposes, without using it for classification/prediction. Then, the attribute "MTBF", has been used as the "label" or, in other words, the goal of the analysis. These two activities have been performed by means of a "Set Role" operator, as shown in Fig. 3. RapidMiner® changes the background colours of the corresponding columns to visually communicate their new state.

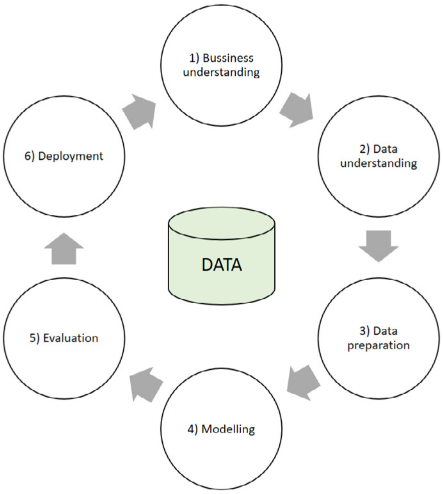
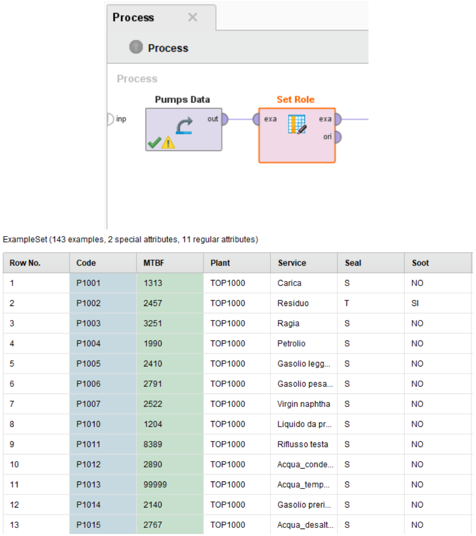


Figure 2. The CRISP-DM process.



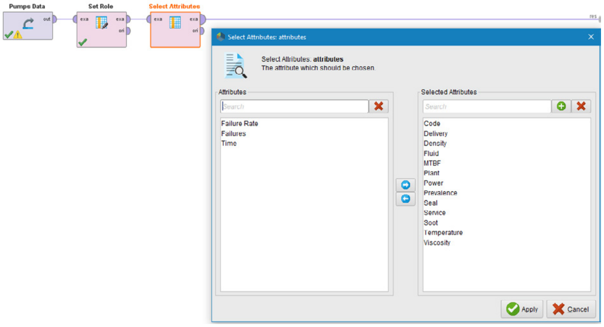
ExampleSet (143 examples, 2 special attributes, 11 regular attributes)

Row No.	Code	MTBF	Plant	Service	Seal	Soot
1	P1001	1313	TOP1000	Carica	S	NO
2	P1002	2457	TOP1000	Residuo	T	SI
3	P1003	3251	TOP1000	Ragia	S	NO
4	P1004	1990	TOP1000	Petrolio	S	NO
5	P1005	2410	TOP1000	Gasolio legg...	S	NO
6	P1006	2791	TOP1000	Gasolio pesa...	S	NO
7	P1007	2522	TOP1000	Virgin naphtha	S	NO
8	P1010	1204	TOP1000	Liquido da pr...	S	NO
9	P1011	8389	TOP1000	Riflusso testa	S	NO
10	P1012	2890	TOP1000	Acqua_conde...	S	NO
11	P1013	99999	TOP1000	Acqua_temp...	S	NO
12	P1014	2140	TOP1000	Gasolio preri...	S	NO
13	P1015	2767	TOP1000	Acqua_desalt...	S	NO

Figure 3. Setting roles.

Successively, using the "Select Attributes" operator, a subset of the available columns has been chosen for the analysis. In particular, the fields "Failures", "Failure Rate" and "Time" have been removed due to the fact that the "MTBF" attribute is clearly given by their combination (Fig. 4).

As stated, a filter has been introduced to remove those examples that showed a "MTBF" equal to -1. Thus, the dataset was reduced to 130 records, with 2 special and 11 regular attributes. Additionally, a "Discretize" operator has been adopted to arrange the label field (MTBF) into homogeneous ranges. Indeed, this operator discretizes the selected numerical attributes to nominal attributes. All numerical values are mapped to the defined classes according to the values specified by the user. Both for comparing purposes and because they proved to be representative and coherent, the ranges are maintained similar to those already used in the previous analyses. Briefly, there are 5 different ranges, as summarised in Table 2.



Select Attributes: attributes

Select Attributes: **attributes**
The attribute which should be chosen.

Attributes

Search

Failure Rate
Failures
Time

Selected Attributes

Search

Code
Delivery
Density
Fluid
MTBF
Plant
Power
Prevalence
Seal
Service
Soot
Temperature
Viscosity

Apply
Cancel

Figure 4. Attribute selection.

Table 2. Example of data available in the CMMS.

MTBF range	**Label**
Up to 2500 hours	VERY LOW
2500 to 4500 hours	LOW
4500 to 9000 hours	MODERATE
9000 to 16000 hours	HIGH
Above 16000 hours	VERY HIGH

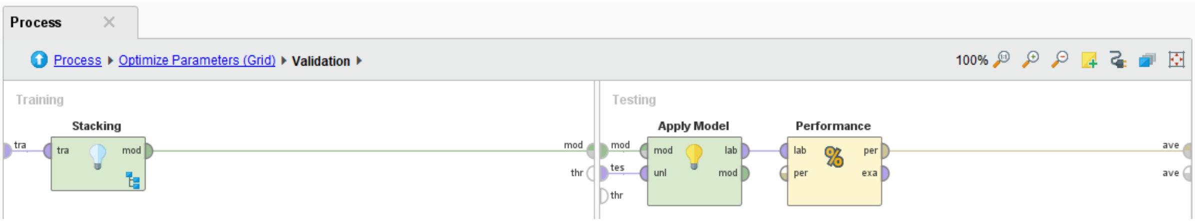


Figure 5. Learning and validation structures.

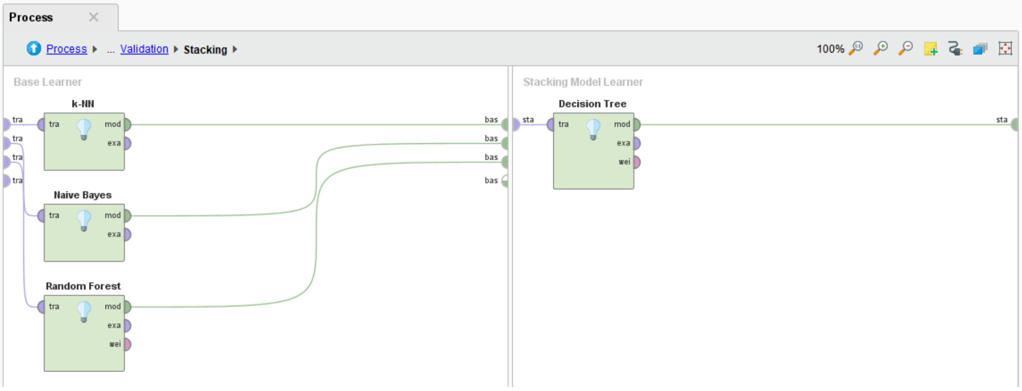


Figure 6. Stacking model structure.

At this point, the dataset is partitioned into two different samples, to leave the 40% of data unseen by the learner and available for a later validation. To this aim, the well-known stratified sampling algorithm is used. It builds random subsets and ensures that the class distribution in the subsets is the same as in the whole dataset.

Following, the ensemble model was built as follows: first, an "Optimise parameters (Grid)" was introduced to perform multiple optimizations on the various algorithms (these settings are reported in detail in Table 2). This is a nested operator that executes all given sub-processes for all combinations of the selected values of the parameters. Finally, it gives as an output the optimal parameters and, at the same time, it sets and runs the model. Within the nested operator a "Split validation" algorithm is used to separate the available examples into two different datasets, respectively the training and the testing sets. Instead of defining the splitting ratio as a static value, it is passed as a parameter to optimization algorithm. In particular, it varies in the interval 50-80%.

These datasets are then passed to the actual learner. As shown in Fig. 5, the learning (training) and validation (testing) structures, within the "Split validation" operator, are constituted respectively by the "Stacking" algorithm, that returns the optimised model, and a testing structure that applies it to the testing dataset and measures the overall performance.

The "Stacking" operator is also a nested operator. It contains two separated sub-processes (Fig. 6), namely the "Base learner" and the "Stacking model learner". The "Base learner" is the operator where all the simple learning algorithms are trained and evaluated with respect to their performance. Then, the "Stacking model learner" decides how to mix their characteristics, strengths and weaknesses to build the final ensemble model.

The "Base learner" includes three algorithms: k-nearest neighbours (k-NN), naïve Bayes, and random forest. The "Stacking model learner" makes use of decision trees. All these are widely known methods in machine learning [18]:

- the *k*-NN algorithm is a non-parametric method for classification, whose output is a class membership;
- naïve Bayes is a probabilistic model-based algorithm that relies on Bayes' theorem with independence assumptions between the features;

- random forests are an ensemble learning method which builds a multitude of decision trees at training time and gives as output the class that is (typically) the mode of the classes of the individual trees;
- decision tree is non-parametric method used to go from observations about an item to conclusions about the item's target value.

Finally, the ensemble model is used to perform a validation cycle.

This activity involves using those samples (40% of the entire dataset) that had been split in a previous step. The "Optimise parameter" operator is used to evaluate the optimal configuration acting on the parameters reported in Table 3.

Running the ensemble model yields the following results. To begin with, the optimal values for the above-mentioned parameters are summarised in Table 4.

Table 3. Optimization parameters.

Operator	Parameter	Range
Validation	Split ratio	0.5 to 0.8
Random forest	Number of trees	10 to 100
Decision Tree	Criterion	Gain ratio, Information gain, Gini index, Accuracy
Decision Tree	Min. leaf size	2 to 6
Decision Tree	Min. size for split	4 to 12

Table 4. Optimal values.

Operator	Parameter	Range
Validation	Split ratio	0.79
Random forest	Number of trees	40
Decision Tree	Criterion	Accuracy
Decision Tree	Min. leaf size	4
Decision Tree	Min. size for split	8

Table 5. Ensemble model confusion matrix.

Accuracy: 96.15%

	True VERY LOW	True LOW	True MODERATE	True HIGH	True VERY HIGH	Class precision
Pred. VERY LOW	9	0	0	0	0	100%
Pred. LOW	0	5	0	0	0	100%
Pred. MODERATE	0	0	5	0	1	83.33%
Pred. HIGH	0	0	0	3	0	100%
Pred. VERY HIGH	0	0	0	0	3	100%
Class recall	100%	100%	100%	100%	75%	

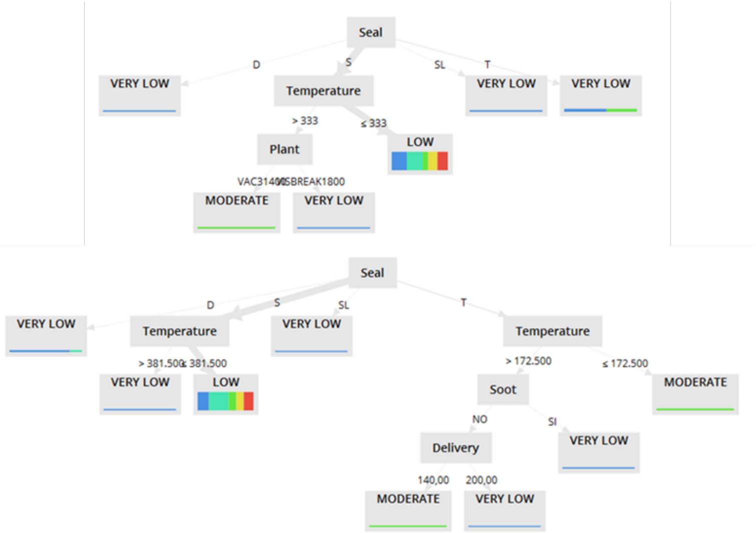


Figure 7. Random forest trees with uncertainty on a LOW label.

Applying such parameters, the ensemble model reached a significant accuracy value of 96.15%. The corresponding confusion matrix is given in Table 5. It clearly shows that only one example over 26 within the testing dataset has been incorrectly classified, whereas all the other results are exact. With the aim of showing the improved capabilities of the ensemble model two trees evaluated by the random forest algorithm have been reported in Fig. 7, where it is evident that the algorithm shows some uncertainty with respect to some examples belonging the label LOW.

On the contrary, the ensemble model is able to overcome this doubt and show a very good capability of discriminating and correctly classifying the majority of the examples (Fig. 8). Even more interestingly, the system evaluates the prediction and the confidence for each row, giving the analyst a very powerful tool to investigate the outcomes. As an example, in Fig. 9, some rows are reported along with the true MTBF value, the prediction and its confidence level.

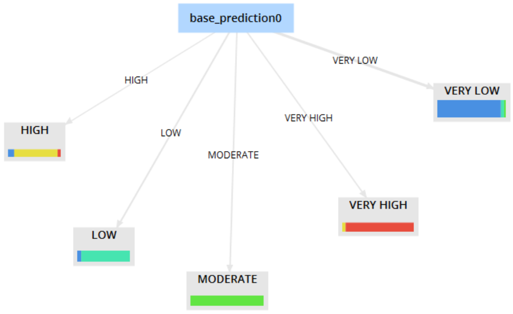
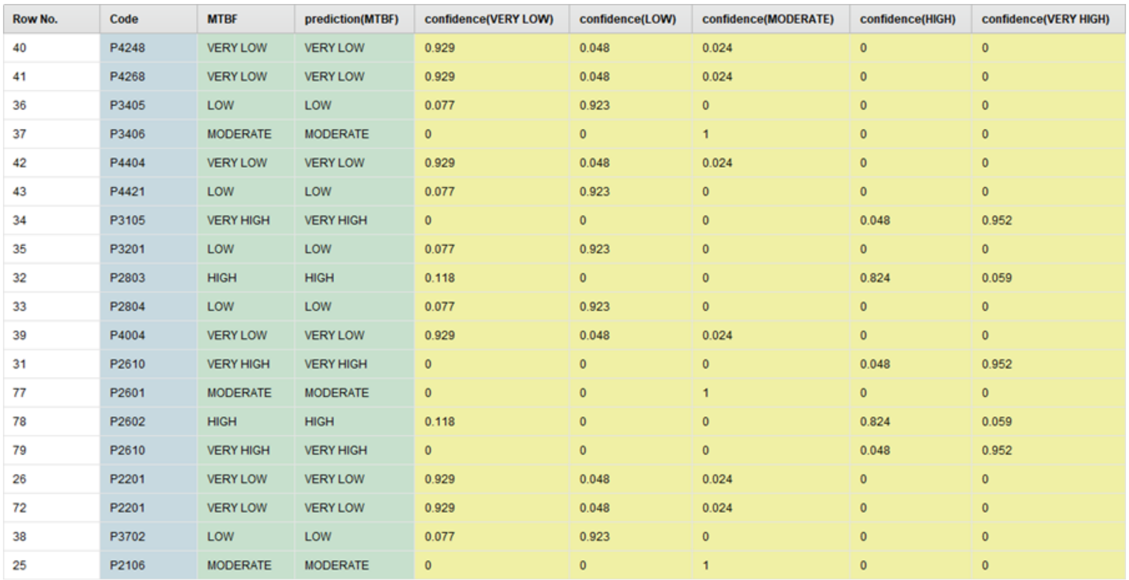


Figure 8. Correctly classified examples (by the ensemble model).

4. Conclusions

The paper presented an innovative framework based on ensemble learning model that, combining results from multiple algorithms, makes it possible to classify items in terms of the MTBF. Specifically, the framework is an ex-post analysis that, starting from "big data" recorded in a CMMS, which includes both quantitative and qualitative variables, tries to give a good estimation of the MTBF of installed equipments. The novel approach is characterized by several peculiar advantages: (i) it exploits the advances in big data analytics to estimate the failure rate of equipments subject to different operating conditions; (ii) it permits to discriminate the working parameters affecting the failure rate; and (iii) it obtains better predictive capabilities than those obtained from any of the basic, constituting learning methods. Finally, the effectiveness and the usefulness of the novel analysis have been demonstrated using an industrial case study about 143 centrifugal pumps installed in an oil refinery plant. As proven by the results, the ensemble reached a significant accuracy value of 96.15%, giving the analyst a very powerful tool to investigate the true MTBF value. Clearly, due to the extreme variability in the operating conditions, results are site specific, unless they are used exclusively to identify the most critical factors in the failure mechanisms of specific equipment. This means that the model, though effective and able to provide extremely good results, cannot be used "as-is" to estimate the MTBF of the equipment installed in a different industrial context. Indeed, it must be reconfigured every time, according to the novel industrial setting in which it is applied. In brief, it requires a new dataset, large and rich enough, possibly without missing values and errors, that will be use during the training stage. If this is the case, thanks to the large number of available algorithms that can be used within the ensemble model, a configuration will be certainly found that provides a good degree of generalization and that is therefore able to correctly estimate the MTBF.



Row No.	Code	MTBF	prediction(MTBF)	confidence(VERY LOW)	confidence(LOW)	confidence(MODERATE)	confidence(HIGH)	confidence(VERY HIGH)
40	P4248	VERY LOW	VERY LOW	0.929	0.048	0.024	0	0
41	P4268	VERY LOW	VERY LOW	0.929	0.048	0.024	0	0
36	P3405	LOW	LOW	0.077	0.923	0	0	0
37	P3406	MODERATE	MODERATE	0	0	1	0	0
42	P4404	VERY LOW	VERY LOW	0.929	0.048	0.024	0	0
43	P4421	LOW	LOW	0.077	0.923	0	0	0
34	P3105	VERY HIGH	VERY HIGH	0	0	0	0.048	0.952
35	P3201	LOW	LOW	0.077	0.923	0	0	0
32	P2803	HIGH	HIGH	0.118	0	0	0.824	0.059
33	P2804	LOW	LOW	0.077	0.923	0	0	0
39	P4004	VERY LOW	VERY LOW	0.929	0.048	0.024	0	0
31	P2610	VERY HIGH	VERY HIGH	0	0	0	0.048	0.952
77	P2601	MODERATE	MODERATE	0	0	1	0	0
78	P2602	HIGH	HIGH	0.118	0	0	0.824	0.059
79	P2610	VERY HIGH	VERY HIGH	0	0	0	0.048	0.952
26	P2201	VERY LOW	VERY LOW	0.929	0.048	0.024	0	0
72	P2201	VERY LOW	VERY LOW	0.929	0.048	0.024	0	0
38	P3702	LOW	LOW	0.077	0.923	0	0	0
25	P2106	MODERATE	MODERATE	0	0	1	0	0

Figure 9. Results of the analysis (example).

References

[1] Bokrantz J, Skoogh A, Berlin C, Stahre J. Maintenance in digitalised manufacturing: Delphi-based scenarios for 2030. International Journal of Production Economics 2017; 191: 154–169.

[2] Mourtzis D, Fotia S, Boli N, Pittaro P. Product-service system (PSS) complexity metrics within mass customization and Industry 4.0 environment. The International Journal of Advanced Manufacturing Technology 2018; 97(1-4): 1–13.

[3] Bumblauskas D, Gemmill D, Igou A, Anzengruber J. Smart Maintenance Decision Support Systems (SMDSS) based on corporate big data analytics. Expert Systems with Applications 2017; 90: 303–317.

[4] Duan C, Deng C, Wang B. Optimal multi-level condition-based maintenance policy for multi-unit systems under economic dependence. The International Journal of Advanced Manufacturing Technology 2017; 91(9-12): 4299-4312.

[5] Zandieh M, Joreir-Ahmadi MN, Fadaei-Rafsanjani A. Buffer allocation problem and preventive maintenance planning in non-homogenous unreliable production lines. The International Journal of Advanced Manufacturing Technology 2017; 91(5-8): 2581-2593.

[6] Agustiady TK, Cudney EA. Total Productive Maintenance: Strategies and Implementation Guide. Boca Raton, FL: CRC Press; 2015.

[7] O'Connor PDT, Kleyner A. Practical Reliability Engineering (5th ed.). New York, NY: John Wiley & Sons; 2012.

[8] Shalev DM, Tiran J. Condition-based fault tree analysis (CBFTA): A new method for improved fault tree analysis (FTA), reliability and safety calculations. Reliability Engineering and System Safety 2007; 92(9): 1231–1241.

[9] Bevilacqua M, Braglia M, Montanari R. The classification and regression tree approach to pump failure rate analysis. Reliability Engineering and System Safety 2003; 79(1): 59–67.

[10] Bevilacqua M, Braglia M, Frosolini M, Montanari R. Failure rate prediction with artificial neural networks. Journal of Quality in Maintenance Engineering 2005; 11(3): 279–294.

[11] Braglia M, Carmignani G, Frosolini M, Zammori F. Data classification and MTBF prediction with a multivariate analysis approach. Reliability Engineering and System Safety 2012; 97(1): 27–35.

[12] Sagi O, Rokach L. Ensemble learning: A survey. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery 2018; 8(4): e1249.

[13] Breiman L. Bagging predictors. Machine Learning 1996; 26(2), 123–140.

[14] Elder JF. A review of Machine Learning, Neural and Statistical Classification. Journal of The American Statistical Association 1996; 91(433): 436–437.

[15] Breiman L. Random Forests, random features. Berkeley; 1999.

[16] Freund Y, Shapire RE. Experiments with a New Boosting Algorithm. In Machine Learning: Proceedings of the Thirteenth International Conference; 1996.

[17] Tibshirani R. Regression shrinkage and selection via the lasso. Journal of the Royal Statistic Society B. 1996; 58(1): 267–288.

[18] Ramasubramanian K, Singh A. Machine Learning Using R. Apress, New York, NY; 2016.

Authors Biography

Marcello Braglia graduated (with distinction) in 1988 in Electronic Engineering at Politecnico di Milano. Since 1995, he has been a Researcher in Mechanical Technology and Production Systems at the Università degli Studi di Brescia. Since 1998, he has been employed as a researcher and since 2002, as a Full Professor, in Industrial Plants at the Università di Pisa. His research activities mainly concern maintenance, reliability, production planning, lean production, logistics and statistical quality control. He is the author of about 180 technical papers published in national and international journals and conference proceedings. He is a member of ANIMP (National Association on Industrial Plants) and AIDI (National Association of Academicians on Industrial Plants).

Davide Castellano graduated in 2010 in Management Engineering at Università di Pisa. In 2015, he obtained a PhD in Mechanical Engineering at Università di Pisa with specialisation in Operations Management. In 2015, he was a Research Fellow at Università di Pisa. At present, he is a Research Fellow at Università degli Studi di Napoli "Federico II". His research activity mainly concerns maintenance, reliability, production management, logistics, and inventory management. He is the author of more than 30 technical papers published in international journals and conference proceedings.

Marco Frosolini graduated in Mechanical Engineering at the Università di Pisa and obtained a PhD in Mechanical Engineering in 2005. He is currently Associate Professor in Industrial Plants within the same University. His research activities mainly concern equipment maintenance and reliability, production planning, logistics and project management. He is also interested in industrial information systems and Industry 4.0. He is author of more than 30 technical papers, published in national and international journals and conference proceedings. He is a member of AIDI (National Association of Academicians on Industrial Plants).

Roberto Gabbrielli graduated with honours in Mechanical Engineering, specializing in Energy, at the University of Pisa (Italy). He has a PhD in Energy Power Systems and is Associate Professor in Industrial Systems Engineering at the Department of Civil and Industrial Engineering of the University of Pisa (Italy). His current research concerns production planning and control, development of decision support systems for industrial investments, occupational safety, energy saving, energy storage and the reduction of the environmental impact in industrial systems. He authored more than 25 papers published in international scientific journals.

Leonardo Marrazzini graduated (110/110) in 2016 in Mechanical Engineering at the University of Pisa. In the same year, he began his PhD studies in Industrial Plants within the same university. His research activities mainly concern the adaptation of Lean Manufacturing principles to engineering-to-order (ETO) production environments. His research focuses on models and techniques to support various company operations.

Luca Padellini graduated in 2018 in Mechanical Engineering at the University of Pisa. In the same year, he began his PhD studies in Smart Industry within the same University. His research activities mainly concern the digitalization of the supply chain in the Tuscan Fashion district. His research focuses on models and techniques able to improve the communication and the transfer of information between the members of the supply chain.