



Ubiquitous Health Technology Management (uHTM)

Using Machine Learning Algorithms to Support Predictive Health Technology Management Programs

Rafael Peixoto¹ · Reginaldo Soares Filho¹ · Juliano Martins¹ · Renato Garcia¹

Received: 23 October 2020 / Revised: 15 April 2021 / Accepted: 17 April 2021 / Published online: 26 April 2021
© Escola Politécnica - Universidade de São Paulo 2021

Abstract

The COVID-19 pandemic increased the need for distributed and ubiquitous health technology management. The eminent risk of Sars-CoV-2 contamination when visiting a health care establishment requires an efficient allocation of the technical team. The equipment problems should be quickly identified and fixed to keep the facility working at its full condition. This article presents a solution to perform remote real-time analysis of primary health care technology behavior, detecting and diagnosing the failures to create predictive maintenance plans. The project uses feature engineering to adapt regular machine learning algorithms to multiclass classification of time series data. The methodology was applied to a dental air compressor. It includes data collection, analysis, and exhibition. The model verified the IBM Watson and the Microsoft Azure Machine Learning Studio with the algorithms of neural networks, logistic regression, decision jungle, and decision forest, which was the most suitable one. The transformation performed in the data considered the influence of time in the read values to obtain a more efficient result in the platform. The solution integrated data collected by the sensors with the cloud using an Internet of Things architecture, a web service, and python scripts to exhibit the outcomes on the computer screen. Therefore, the model performs notification and identification of health technology failures, supporting the decision-making process of ubiquitous management in clinical engineering.

Keywords Clinical engineering · Machine learning · Ubiquitous health technology management · Remote supervision · Dental air compressor

1 Introduction

Driven by the necessity of safer proceedings, more reliable diagnosis, and cost reduction, clinical engineering, once associated mainly with the maintenance of medical equipment, has expanded its fields of operation (Garcia et al. 2011; Zambuto 2004). Besides, the supervision of technology as part of predictive maintenance is financially justifiable in systems where failures have serious impacts on the whole technological process. It can be used in situations of

randomness, repetitiveness, and dangers caused by faults in the technology (Mobley 2002).

Among the systems used in equipment management, there are machine learning (ML) algorithms. These computation methods comprehend the relation between data and information through the generalization of examples. Therefore, they are not explicitly programmed to develop a task, but they learn from experience. These tools can be used to solve classification problems. They are indicated to work with large amounts of data processing and with the confirmation of conditions yet unknown (Awad and Khanna 2015). The creation of a machine learning model can be performed with programming languages or with a Software as a Service (SaaS) platform (Idoine et al. 2018).

The importance of an Internet of Things (IoT) architecture in medical equipment management is widely known (Maktoubian and Ansari 2019; Çoban et al. 2018), yet there are few examples of machine learning applied to their predictive maintenance. The industry has used Support

✉ Rafael Peixoto
rafael.peixoto@grad.ufsc.br;
rafaeldossantospeixoto@gmail.com

¹ Biomedical Engineering Institute (IEB-UFSC), Federal University of Santa Catarina, Mail Box 5138, Campus Universitário, Florianópolis, Brazil

Vector Machine (SVM) algorithms to detect failures in reciprocating compressor valves based on vibration information (Ren et al. 2005). In health care, SVM and vibration data have also been applied to predictive maintenance of an Immunoassay Analyzer, identifying the belt slippage of the metering arm failure (Shamayleh et al. 2020). In both cases, the algorithms were restricted to detect only a single failure type in the whole system. A multiclass failure classification allows the creation of maintenance plans to indicate the tools and mechanical parts required to fix the equipment, reducing the number of visits to the health facility.

The Brazilian primary health care establishments are geographically distributed to follow the population density distribution. The facilities are not close to each other and may have different types of equipment to be monitored. Therefore, the technical team needs to travel significant distances to manage them. A remote decision-making support system with trend indicators would allow the prediction of conditions to coordinate the team mobility. Furthermore, the failure classification would avoid visits designed to malfunctioning identification.

The objective of the present work is to provide a platform to detect and classify health equipment failures applying machine learning. It integrates the data collected by the sensors with the cloud using an IoT architecture to perform remote real-time analysis. The results of this application belong to the decision category as a stage of cognitive analysis, in which cognitive tools are used to construct predetermined actions to support the decision-making process in technology management regarding ubiquitous structures of clinical engineering (Garcia et al. 2018).

This article evaluates models created using two software as a service (SaaS) platforms: the IBM Watson and the Microsoft Azure Machine Learning Studio. It also examines how the number of variables, feature engineering, and the number of classes influence the performance of the algorithms available in such high-level interfaces. The paper proposes an alternative to classifying time series data with general machine learning models, such as decision forest, instead of more complex algorithms, such as recurrent neural networks.

2 Materials and Methods

The development of this work uses a SaaS ML platform as a cognitive tool. Some cognitive analysis services were verified: Amazon AWS Studio, IBM Watson, Google Cloud, and

Microsoft Azure Machine Learning Studio, electing the last one due to its flexibility, higher number of classification algorithms, and method for model validation (Idoine et al. 2018). The Microsoft Azure Machine Learning Studio (MAMLS) presents blocks with functions, algorithms, and data processing that can be linked to obtain the desired pipeline.

The creation of an ML algorithm is divided into seven steps: data collection, preprocessing, transformation, training, testing, application of reinforcement learning, and execution (Awad and Khanna 2015). However, the procedure executed in this proposal has adaptations for the implementation in the ubiquitous management of health technology using the MAMLS platform.

The Fig. 1 presents the data flow in the solution. The data are received from a collector installed in the equipment, transferred to a data hub, sent to the database, downloaded by a computer, sent to the cloud through a web service, and returned to the computer for exhibition in real-time.

2.1 Data Collection

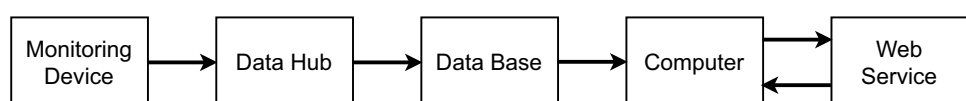
The selection of the dental air compressor as the device to be monitored was made in the operational category, a previous step in this study, performed by the Clinical Engineering area of the Biomedical Engineering Institute of the Federal University of Santa Catarina (IEB-UFSC). It was supported by failure analysis tools to identify and define the equipment and its parameters to be investigated (Soares et al. 2020). To extend this methodology to other equipment, the process performed in the article should be repeated, determining the faults, electing the values to be measured, and performing the steps for the creation of the ML algorithm.

The failures observed, corresponding to 89% of all existing failures reported in information systems for dental technologies, are the following: air leak through the hose, push-in connector, and regulator; defect in the piston valves, rings, gasket and cylinder head; leaks through the air extractor; leaks through the coil; and locked piston (Soares et al. 2020). They are categorized into the following behaviors:

- *Small, medium, and large leak*: air leak through the hose, push-in connector, regulator, air extractor, and coil.
- *Worn out rings (no rings)*: defect in the piston valves, rings, gasket, and cylinder head.
- *Motor failure*: locked piston and defective capacitor.

Based on the deficiencies described, the variables chosen to be monitored were current, voltage, equipment

Fig. 1 Data flow in the technological solution



temperature, environment temperature, environment humidity, and pressure in the compressed air system. However, after the observation of some test results, the environment temperature and humidity were removed since they were mostly constant.

A data collector is installed in each piece of equipment to read the values from the sensors. Since there are multiple devices in a health facility, a data concentrator gathers the information from all collectors and sends it to a database through the wireless network.

The data representing the normal behavior were obtained from a health facility in Florianópolis (SC - Brazil). A clinical engineering specialist monitored the device over three months while the information was collected. The data representing each defect were controllably created by simulating such conditions in the IEB-UFSC laboratory. For each routine, different runs were made to avoid the influence of environmental conditions, such as the local temperature and period of the day, and to create a robust machine learning model.

The collector module records the data from the sensors every two seconds. Due to the effects of some simulations in the compressor, the amount of data in each class is different.

2.2 Data Preprocessing

Since the values are collected using different sensors, there needs to be a standardization. The data from the simulations are joined into one file and aligned by the time of collection. Then, the commas representing the decimals are replaced by dots, the columns are renamed, and the unities are regulated. The information collected when the equipment was off is deleted. The unities indicators are removed and the missing values are filtered. Furthermore, a column indicating the simulated behavior is added.

A Python script is created for the data preprocessing stage. The code uses the *pandas* library due to its efficiency, wide use, and open-source, the *psycopg2* library to communicate with the SQL database, and the *datetime* library to deal with values in date format. Thus, the data are formatted, cleaned, filtered, divided by simulation, and classified accordingly.

2.3 Data Transformation

It is possible to understand the equipment behavior as time functions of the variables collected by the sensors. Consequently, the task of classifying the health care technology behavior consists in the identification of the category its curve belongs to.

There is no previous knowledge about the parameter with higher influence in the operation, however, it is known that the values change differently as time passes, depending on the behavior of the equipment. So, an analysis of the data at a single moment is not enough to solve the problem. To work around this issue, it is essential to create a variable that relates the measures to time.

Since the MAMLS platform does not offer algorithms to analyze time series information, such as recurrent neural networks, feature engineering was needed. Transformations derived the original data to create more significant characteristics.

First, the operation attribute (on or off) is added to the data. The feature is defined based on the current value, when it is low (less than 1A) the device is off, otherwise, it is on.

Using the operation element, a column is created to store the elapsed time since the last change in functioning. Hence, it is possible to obtain how long the equipment has been working. This feature is useful to classify air leaks, when the technology takes more time to turn off.

The elapsed time brings forward the equivalent time between the actual moment and the one in which the device was turned on/off, however, it does not present a measure relative to subsequent data. Therefore, to relate the actual value with the one before, two features are built: the temperature rate and the pressure rate. This solution is effective because it represents how the function evolves as time passes. For example, if the temperature rate is a large positive number, the temperature increases quickly; if it is zero, the temperature is constant; if the rate is a small negative, the temperature decreases slowly. These attributes are created by subtracting the value of the actual data by the past one and dividing them by the time between the samples. This way, as the time difference tends to zero, the value approaches the slope (inclination) of the tangent line at the point.

Consequently, it is possible to obtain an approximation of the derivative by the Euler Backward Method (Equation 1), where $f(k)$ is the value of the function at instant k and Dt is the time difference between instants k and $k-1$.

$$f'(k) = \frac{f(k) - f(k-1)}{Dt} \quad (1)$$

Thus, the parameters that present the relationship between data in distinct moments are constructed, solving the problem of using the MAMLS platform for time series data analysis.

2.4 Model Training

The data sent to the MAMLS platform are randomly divided into two sets, one for training, corresponding to 70%, and

the other for testing, 30%. The platform has four multiclass classification algorithms: decision forest, which classifies by combining different decision trees (Tong et al. 2003); decision jungle, an extension of the decision forest algorithm (Shotton et al. 2013); logistic regression, classification using the logistic curve (Kleinbaum et al. 2002); and neural networks, which uses layers of artificial neurons for classification (Guresen and Kayakutlu 2011). The parameters for training each algorithm are the ones set as default by the platform.

To obtain the best model, all the algorithms available in the MAMLS platform were compared (Fig. 2). Then, the best one was integrated into the solution.

The study also verified the IBM Watson platform. It has a model creation assistant, a high-level interface, and only one algorithm for multiclass classification. It does not allow the connection of blocks to create a pipeline. Instead, a menu where one can select the desired purpose encapsulates the whole process. The platform has an exclusive page for manually testing one data at a time. Besides the lack of algorithm customization, the IBM Watson did not perform as expected (Fig. 3) and it was not considered in future evaluations.

2.5 Model Testing

The algorithm evaluation creates a confusion matrix that presents the predicted categories against their true classes with the values for the true positives (TP), false positives (FP), true negatives (TN), and false negatives (FN). So, a matrix with all data in the main diagonal reflects an effective model.

To have a single score value that is easier to compare, some evaluation metrics can be derived from the confusion matrix. One of them is the accuracy (Acc), which corresponds to the fraction of the total instances that are correctly classified (Equation 2) (Hand and Christen 2018).

$$Acc = \frac{TP + TN}{TP + FP + TN + FN} \quad (2)$$

Besides, the values of precision and recall can be also obtained: precision (P) is the ratio between the values that were correctly classified as positive and values that were predicted to be positive (Equation 3); recall (R) is the rate of all positive values that were correctly labeled (Equation 4) (Hand and Christen 2018).

$$P = \frac{TP}{TP + FP} \quad (3)$$

$$R = \frac{TP}{TP + FN} \quad (4)$$

The F1 score is used because the accuracy measure is not adequate for classification problems with imbalanced classes (Schütze et al. 2008). For binary classification, this metric calculates the harmonic means of precision and recall, however for multiclass classification, the F1-score is calculated as the arithmetic mean over the harmonic means, (Equation 5), where K is the number of classes (Opitz and Burst 2019).

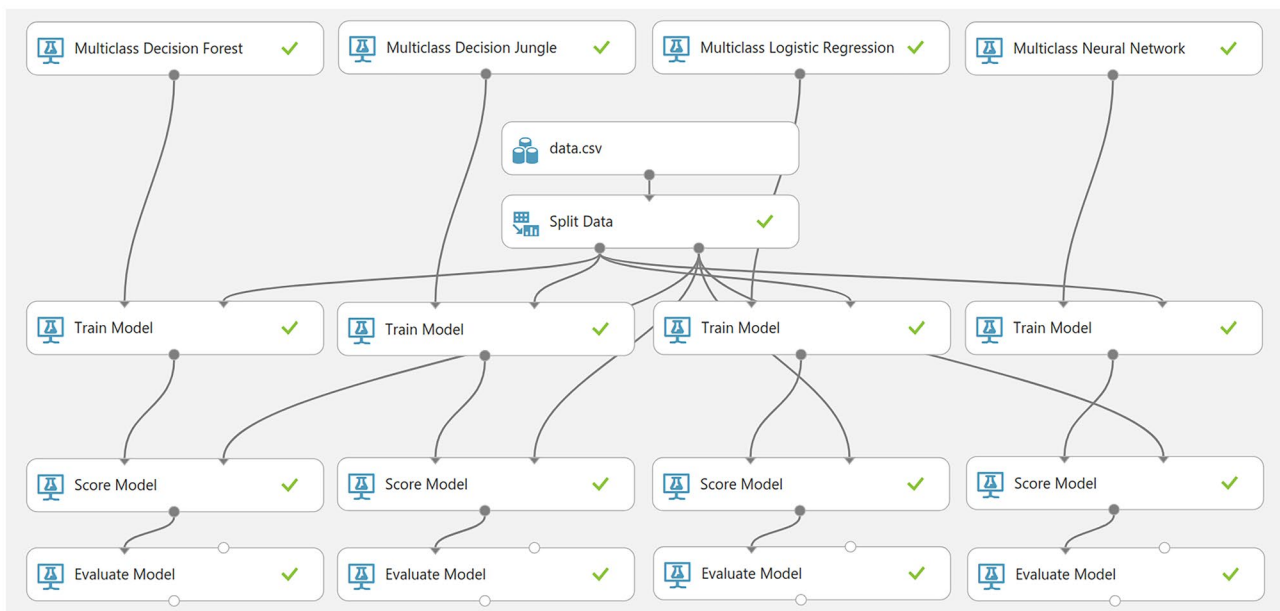


Fig. 2 Microsoft Azure Machine Learning Studio model

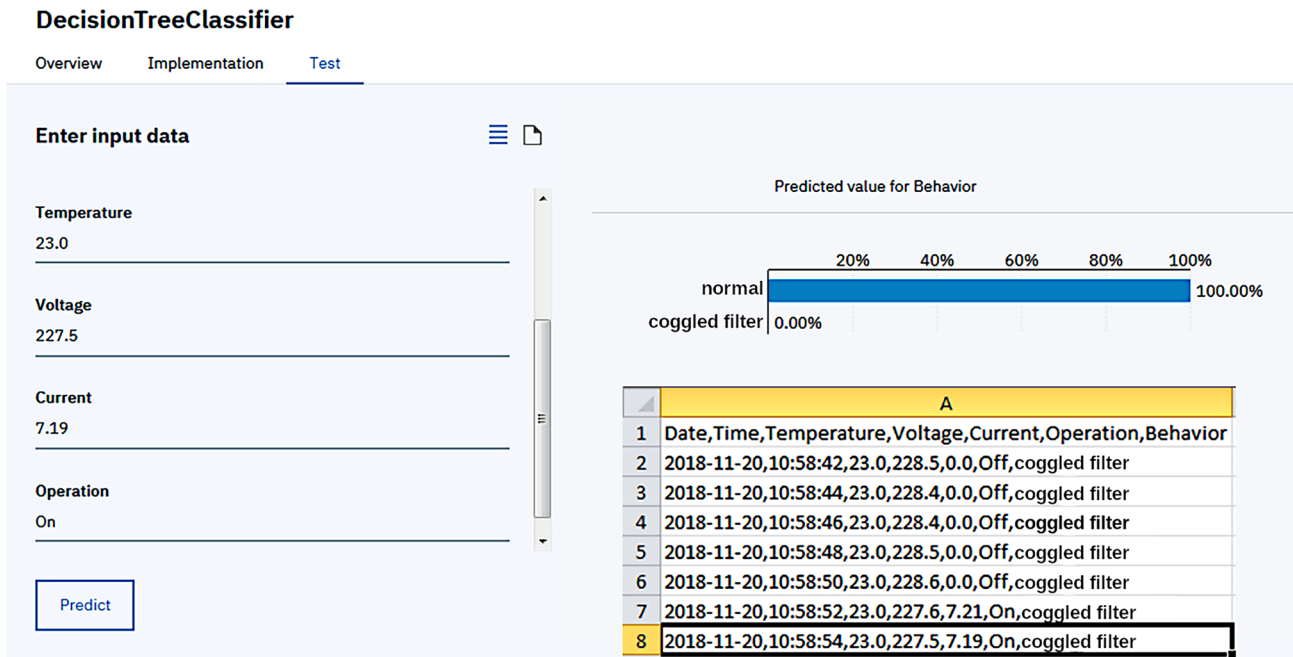


Fig. 3 IBM Watson result and comparison

$$F_1 = \frac{1}{K} \sum_{k=1}^K \frac{2P_k R_k}{P_k + R_k} \tag{5}$$

In case the test results are not appropriate, the proceedings should be repeated, adding more data, more features, modifying the transformations or changing the training parameters. Besides the model evaluation performed in 30% of the collected data, online tests conducted in the laboratory of the IEB-UFSC simulated some behaviors and verified that the results corresponded to the ones presented in the confusion matrix.

2.6 Execution

To apply the remote classification in real-time, the raw data collected by the device is sent to a local database. A Python script is created to transform it following the standards applied in the training procedure and send it to the cloud using a web service to be analyzed by the model. The response from the MAMLS platform is treated and exhibited on the computer screen.

3 Results

An initial test simulated only the problems of worn-out rings and clogged filter. In this case, no transformations were performed, only preprocessing. Besides, this model included the measurements of environmental humidity and temperature, but

not of compressed air pressure. The development also tested the four algorithms: decision forest (Fig. 4), decision jungle (Fig. 5), logistic regression (Fig. 6) e neural network (Fig. 7).

Then, the model contemplated the measurements of pressure, but not the ones of environmental humidity and temperature since these values were mostly constant during the

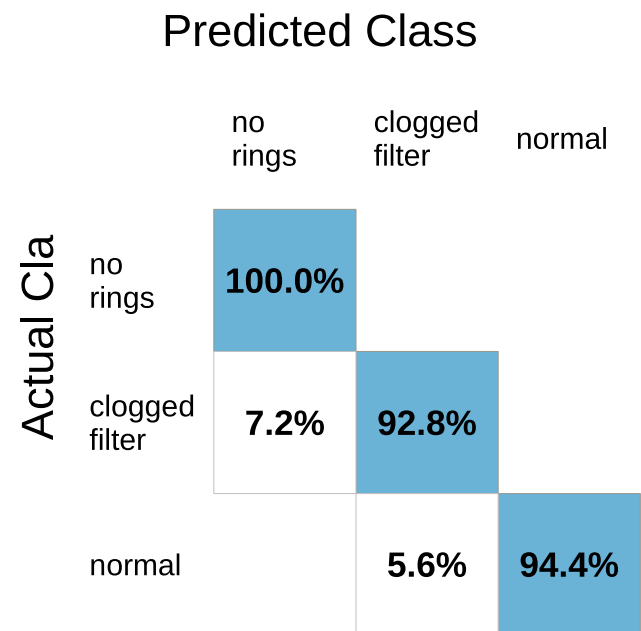


Fig. 4 Confusion matrix of the decision forest algorithm development version

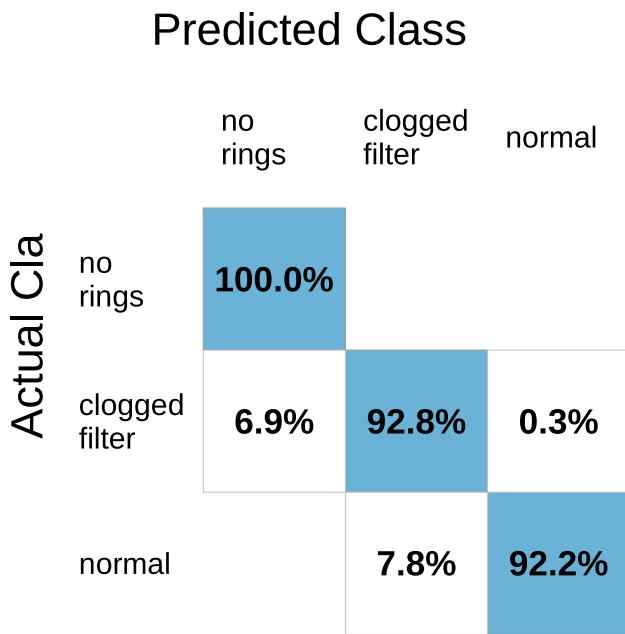


Fig. 5 Confusion matrix of the decision jungle algorithm development version

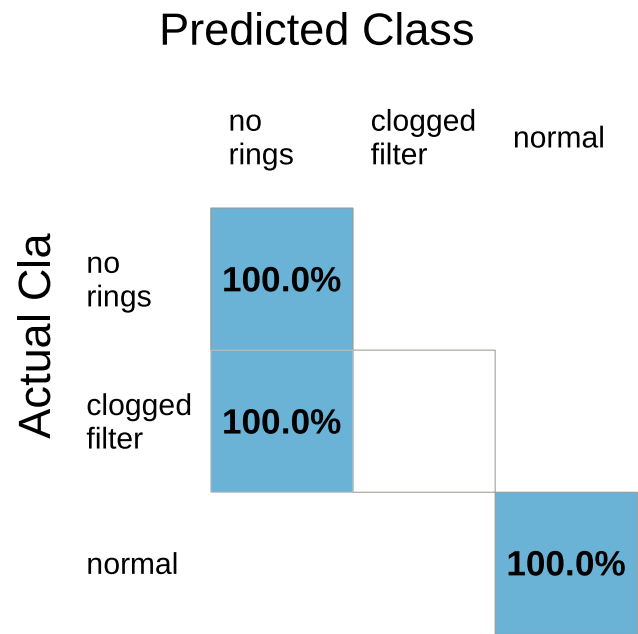


Fig. 7 Confusion matrix of the neural network algorithm development version

simulation. This version considered the derived values to relate the data to time: operation, elapsed time, temperature rate, and pressure rate. Besides, it included the problems from all categories.

The confusion matrix for the decision forest algorithm (Fig. 8) shows that it did most of the classification correctly.

The majority of the values were presented in the main diagonal, except for the *motor failure* one, which was classified as *normal* in 33% of the cases.

The decision jungle algorithm shows similar behavior to the decision forest model (Fig. 9), yet the *motor failure* data are not classified correctly because in 66.7% of the cases they are classified as *no rings*.

The logistic regression (Fig. 10) and the neural network (Fig. 11) algorithms do not present good results because there is considerable dispersion in classification with some focus on the *normal* behavior.

The accuracy and F1-score metrics for each algorithm are shown in Table 1.

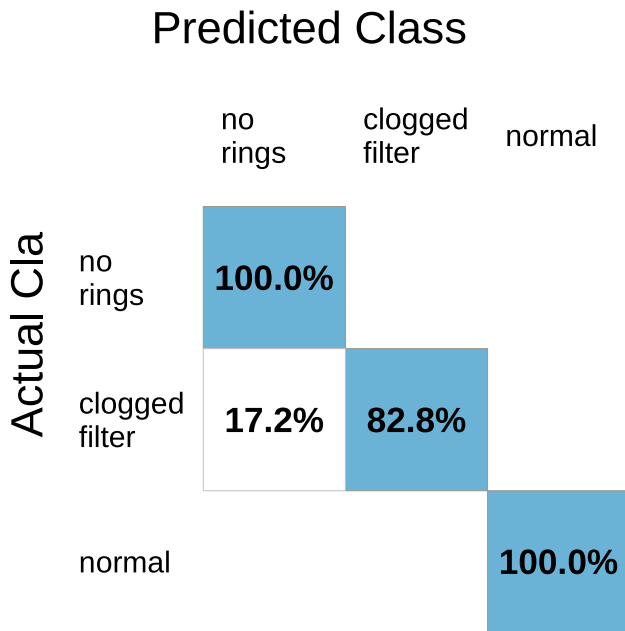
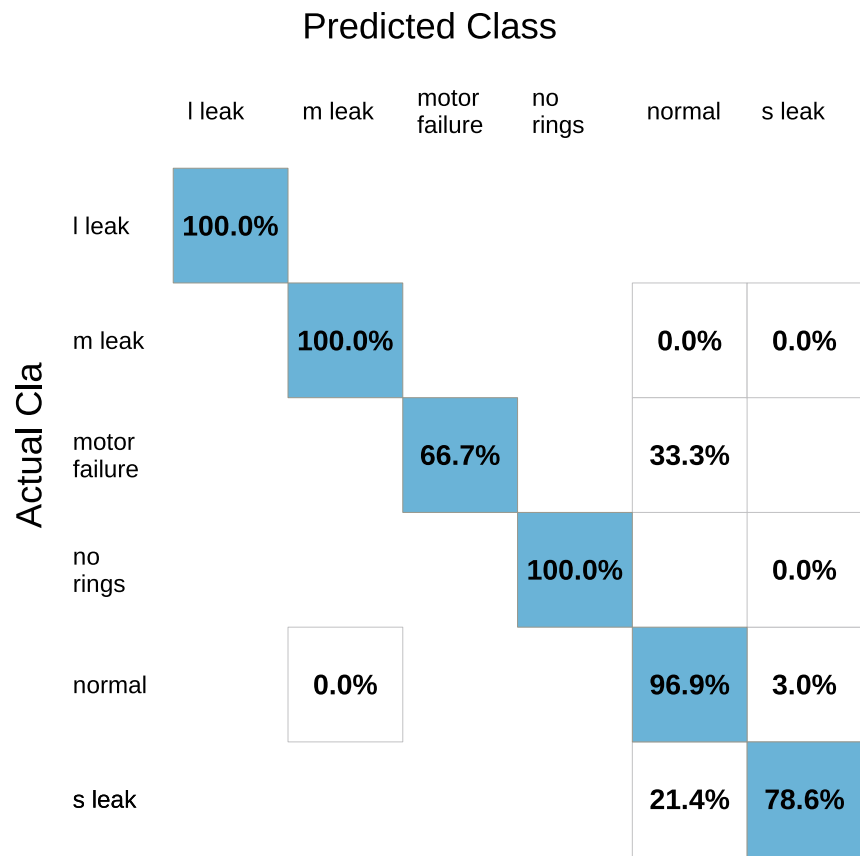


Fig. 6 Confusion matrix of the logistic regression algorithm development version

4 Discussion

The high-level of the Microsoft platform allows fast model creation, but it also comes with some caveats. During training, it considered the information from the date column in the classification. Thus, data from different behaviors, but simulated in periods close to each other, were classified the same way and the date column had to be removed. The tests considering just the raw data of the normal, worn-out rings, and clogged filter behaviors had adequate results for all algorithms but the neural networks. However, as the number of problems increased, the model could not classify them properly, indicating the limitations of the machine learning studio to classify time series data.

Fig. 8 Confusion matrix of the decision forest algorithm



The model increased in performance after implementing feature engineering and creating the elements of operation, elapsed time, temperature rate, and pressure rate. One minor issue is that some of the small leak values were classified as normal by the decision forest and decision jungle algorithms. A small leak (Fig. 12) does not present a significant influence on the equipment behavior. During the simulations, the air compressor did not work constantly, but in real cases, when the technology is used during the whole day, this could cause a larger leak and the algorithm would detect it properly. **Fig. 12** Small leak

Another condition noticed in the decision forest algorithm is that some data from *motor failure* were wrongly classified (Ling and Sheng 2010). A possible explanation is the effects of imbalanced classes (skewed data). In other words, there are much more data corresponding to the other classes, mainly the *normal* class, than the *motor failure* one. It is difficult to collect data of this category because the simulations of a stuck piston or defective capacitor automatically trigger the residual circuit breaker as a result of overcurrent in the electric power system, shutting down the motor.

Since the acquisition of more data seems impractical, the solution should come from the algorithm. There are techniques that could be used to create artificial data for the imbalanced class. However, some of them require

adaptations for multiclass problems (He et al. 2008). Thus, this study only used the data obtained in the simulations.

Analyzing the results, the decision forest is the chosen model because it joins different decision tree classifiers, training each one in a different part of the data, and combining their results. It mitigates some of the errors from the decision tree model, such as those caused by class imbalance (Rokach 2016). The results validate that this method is more suitable for the classification of applications where the importance of each variable is initially unknown (Oza and Tumer 2008).

Furthermore, the algorithms of decision forest and decision jungle had similar accuracies, yet their F1-scores were distinct, showing that the F1-score is a better evaluation metric for classification (Schütze et al. 2008). This can be explained because the F1-score for the *motor failure* class is zero for all the algorithms except for the decision forest one.

However, the F1-score has the disadvantage of attributing the same importance for precision and recall independently of their classes, while this should be an aspect defined by the problem (Hand and Christen 2018). For example, the classes of *small leaking* and *motor failure* had false negatives, that is, sometimes the behavior was classified as *normal* even though it was not. This fact has consequences in primary health care management since the non-notification

Fig. 9 Confusion matrix of the decision jungle algorithm

		Predicted Class					
		l leak	m leak	motor failure	no rings	normal	s leak
Actual Cla	l leak	100.0%					
	m leak		100.0%			0.0%	0.0%
	motor failure				66.7%	33.3%	
	no rings		0.0%		100.0%		0.0%
	normal		0.0%			99.4%	0.6%
	s leak	0.0%	0.0%			45.8%	54.1%

Fig. 10 Confusion matrix of the logistic regression algorithm

		Predicted Class						
		l leak	m leak	motor failure	no rings	normal	s leak	missing
Actual Cla	l leak	8.7%	25.1%		16.3%	47.1%	2.8%	0.0%
	m leak	8.1%	26.3%		16.2%	46.6%	2.8%	0.0%
	motor failure				33.3%	66.7%		
	no rings	8.4%	24.9%		17.0%	46.9%	2.9%	0.0%
	normal	8.1%	25.3%		15.9%	47.9%	2.8%	0.0%
	s leak	8.2%	25.5%		16.1%	47.1%	3.1%	
	missing							

Fig. 11 Confusion matrix of the neural network algorithm

		Predicted Class						
		l leak	m leak	motor failure	no rings	normal	s leak	missing
Actual Cla	l leak	8.8%	25.2%	0.0%	16.3%	43.0%	6.8%	0.0%
	m leak	8.1%	26.4%		16.2%	42.1%	7.2%	0.0%
	motor failure				33.3%	66.7%		
	no rings	8.4%	24.9%		16.9%	42.5%	7.2%	0.0%
	normal	8.1%	25.3%		15.9%	43.6%	7.1%	0.0%
	s leak	8.2%	25.5%	0.0%	16.0%	42.5%	7.8%	
	missing							

Table 1 Evaluation metrics

Algorithm	Accuracy	F1-score
Decision Forest	96.61%	92.98%
Decision Jungle	95.4%	77.34%
Logistic Regression	31.1%	16.75%
Neural Network	29.68%	19.11%



Fig. 12 Small leak

of this event may cause the equipment to deteriorate or even break. On the other hand, the incorrect notification of normal behavior as problematic would waste time and resources. Therefore, to make changes to the weights of precision and recall, a study is suggested on the impacts of false positives and false negatives in clinical engineering ubiquitous management.

5 Conclusion

The cognitive analysis platform can support clinical engineering in the decision-making process by creating plans for predictive maintenance, increasing the reliability and the safety of primary health care system users.

The application of the platform as a tool for ubiquitous management of medical technologies enables the constant supervision of the equipment, improving the quality of health assistance. The amount of time and resources needed to manage health devices can be reduced by the remote real-time analysis provided by this technological solution, causing the decision-making process to be faster, more efficient, and based on evidence. Furthermore, the monitoring allows the collection of data for the future creation of a predictive maintenance model based on the occurrence of failures.

The remote analysis system can help to coordinate the management team's actions by creating maintenance plans based on the failures found in the equipment. The workers would only need to visit the health establishment to fix the technology when a problem was reported and classified. There would be no appointments to identify failures and elect the tools and mechanical parts required to repair them. Hence, it would reduce the number of visits to the health facilities and the risk of contracting Sars-CoV-2, increasing the team's safety.

The development of the solution indicated the importance of feature engineering for the multiclass classification of time series data using regular ML algorithms. The transformations of operation, elapsed time, and time derivatives were crucial to improve the performance of the model as the number of classes increased. Besides, the measurement of pressure in the compressed air system proved to be essential, while the ones of environment temperature and humidity did not.

Finally, the methodology applied in the dental air compressor had an accuracy of 96.91% and an F1-score of 92.98% for the decision forest algorithm. Thus, it proved to be efficient in failure detection and classification, as shown in Fig. 8. Hence, this procedure could be extended to other primary health care technologies, including those with higher costs, risks, and impacts in hospitals' budgets, by repeating the process to identify the monitored parameters (Soares et al. 2020) and the steps for the ML model creation (Awad and Khanna 2015).

Acknowledgements I am grateful for my family and friends who supported me during the development of this work.

Funding Information No funding was received to assist with the preparation of this manuscript.

Availability of data and material The data was submitted as attachment.

Code availability The code was submitted as attachment..

Declarations

Conflicts of interest The authors declare that they have no conflict of interest.

References

- Awad M, Khanna R (2015) Efficient learning machines: theories, concepts, and applications for engineers and system designers. Apress, Berkeley, CA <https://books.google.com.br/books?id=Bk4nCgAAQBAJ>
- Çoban S, Gökçalp MO, Gökçalp E, Eren, PE, Koçyiğit A (2018) Predictive maintenance in healthcare services with big data technologies. In: 2018 IEEE 11th Conference on Service-Oriented Computing and Applications (SOCA), pp. 93–98. <https://doi.org/10.1109/SOCA.2018.00021>
- Garcia R, Pezzolla G, Soares Filho R, Martins J (2018) Health technology ubiquitous management model for primary health care. In: Fourth WHO Global Forum on Medical Devices Report. Visakhapatnam, India
- Garcia S, Santos R, de Avelar P, Zaniboni R, Garcia R (2011) Health care technology management applied to public primary care health. In: 2011 Pan American Health Care Exchanges, pp. 250–253. <https://doi.org/10.1109/PAHCE.2011.5871898>
- Guresen E, Kayakutlu G (2011) Definition of artificial neural networks with comparison to other networks. *Proc Comp Sci* 3:426–433. <https://doi.org/10.1016/j.procs.2010.12.071>
- Hand D, Christen P (2018) A note on using the f-measure for evaluating record linkage algorithms. *Statistics and Computing* 28(3), 539–547 <https://doi.org/10.1007/s11222-017-9746-6>, <https://app.dimensions.ai/details/publication/pub.1084928040> and <http://spiral.imperial.ac.uk/bitstream/10044/1/46235/2/stco-d-16-00349-final.pdf>
- He H, Bai Y, Garcia EA, Li S (2008) Adasyn: Adaptive synthetic sampling approach for imbalanced learning. In: 2008 IEEE international joint conference on neural networks (IEEE world congress on computational intelligence), pp. 1322–1328
- Idoine C, Krensky P, Brethenoux E, Hare J, Sicular S, Vashisth S (2018) Magic quadrant for data science and machine-learning platforms. Gartner, Inc
- Kleinbaum DG, Dietz K, Gail M, Klein M, Klein M (2002) Logistic Regression: A Self-Learning Text. Springer, New York
- Ling CX, Sheng VS (2010) Encyclopedia of Machine Learning. Springer US, Boston, MA. https://doi.org/10.1007/978-0-387-30164-8_110
- Maktoubian J, Ansari K (2019) An iot architecture for preventive maintenance of medical devices in healthcare organizations. *Health Technol* 9(3):233–243. <https://doi.org/10.1007/s12553-018-00286-0>
- Mobley R (2002) An Introduction to Predictive Maintenance, second edition edn. Butterworth-Heinemann, Burlington (2002). <https://doi.org/10.1016/B978-075067531-4/50002-6>, <http://www.sciencedirect.com/science/article/pii/B9780750675314500026>
- Opitz J, Burst S (2019) Macro f1 and macro f1. arXiv e-prints p., <https://ui.adsabs.harvard.edu/abs/2019arXiv1911033470>
- Oza N, Tumer K (2008) Classifier ensembles: Select real-world applications. *Inform Fusion* 9(1):4–20. <https://doi.org/10.1016/j.inffus.2007.07.002>
- Ren Q, Ma X, Miao G (2005) Application of support vector machines in reciprocating compressor valve fault diagnosis. In: Advances in Natural Computation, pp. 81–84. Berlin, Heidelberg https://doi.org/10.1007/11539117_13
- Rokach L (2016) Decision forest: Twenty years of research. *Inform Fusion* 27:111–125. <https://doi.org/10.1016/j.inffus.2015.06.005>
- Schütze H, Manning CD, Raghavan P (2008) Introduction to Information Retrieval. Cambridge University Press, Cambridge
- Shamayleh A, Awad M, Farhat J (2020) Iot based predictive maintenance management of medical equipment. *J Med Sys* 44(4):72. <https://doi.org/10.1007/s10916-020-1534-8>
- Shotton J, Sharp T, Kohli P, Nowozin S, Winn J, Criminisi A (2013) Decision jungles: Compact and rich models for classification. In: *Adv Neural Inf Process Syst* 26, pp. 234–242. Red Hook
- Soares Filho R, Martins J, Garcia R (2020) Methodology for defining ubiquitous management indicators in primary health care. In: VIII Latin American Conference on Biomedical Engineering and XLII National Conference on Biomedical Engineering, pp. 1298–1305. Cham https://doi.org/10.1007/978-3-030-30648-9_167

Tong W, Hong H, Fang H, Xie Q, Perkins R (2003) Decision forest: Combining the predictions of multiple independent decision tree models. *J Chem Info Comp Sci* 43(2):525–531. <https://doi.org/10.1021/ci020058s>

Zambuto RP (2004) Introduction to Clinical Engineering, pp. 1–2. *Bio-medical Engineering*. Academic Press, Burlington <https://www.sciencedirect.com/science/article/pii/B9780122265709500028>

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.