SUPERVISED LEARNING BASED APPROACH FOR TURBOFAN ENGINES FAILURE PROGNOSIS WITHIN THE BELIEF FUNCTION FRAMEWORK

Rachid Hamza and Safa Ben Ayed
CESI-LINEACT. UR7527, Arras 62000, France

ABSTRACT

Recent developments in maintenance modeling, powered by data-driven approaches such as Machine Learning (ML), have enabled a wide range of applications. For example, industrial systems coming with a huge operating database make ML an ideal candidate for their predictive maintenance (PdM). PdM is the process of predicting malfunctions using data from equipment monitoring and process performance measurements. Indeed, PdM and ML have developed a very strong connection. However, it is not always easy or straightforward to perform effective predictive maintenance for several reasons such as imperfect data. Therefore, we aim, during this paper, to manage uncertainty and/or imprecision during learning, using an evidential supervised learning approach built on a powerful framework called the belief function theory. This research work is applied on NASA's C-MAPSS dataset for turbofan engines failure prognosis.

KEYWORDS

Predictive Maintenance, Machine Learning, Belief Function Theory, Uncertainty

1. INTRODUCTION

Today, a new revolution is emerging with "The Fourth Industrial Revolution", a term also known as "Industry 4.0". It is primarily merging automation with advanced manufacturing to reduce direct human effort and resources. The diffusion of new digital technologies using connected objects, the Internet of Things, cloud, and artificial intelligence have led to the development of a new concept of maintenance known worldwide under the name intelligent predictive maintenance for Industry 4.0. Nowadays, Prognostic and Health Management (PHM) systems are some of the main protagonists of the Industry revolution. PHM is a research area with multiple methodologies and functions as a decision support tool that aims at minimizing maintenance costs and predicting when a failure could occur by the assessment, prognosis, diagnosis, and health management of engineered systems. The core of PHM is failure prognostic. It refers specifically to the phase involved with predicting future behavior and the system's useful lifetime left in terms of the current operating state and the scheduling of required maintenance actions to maintain system health. This useful lifetime is often called the remaining useful life (RUL) and is defined as the length from the current time and operating state to the end of the useful life. The notice of pending equipment failure allows for sufficient lead time so that necessary decisions, personnel, equipment, and spare parts can be organized and deployed, thus minimizing equipment downtime and repair costs. By leveraging RUL estimation, industries, such as aerospace, maritime, and energy, can improve maintenance schedules to avoid catastrophic failures and, consequently, save lives and costs. The industry has also to assure that its asset utilization is optimum by guaranteeing timely, but not premature. Furthermore, this practice promotes sustainability as the use of spare parts is optimum and no useful life is wasted. There are different recent works that have been used in literature for predictive maintenance and reliability engineering. Some of these approaches are Markov chains (Tierney, 1996), Petri nets (Chaouiya, 2007), fault tree analysis (Lee, et al., 1985), and analytic hierarchy process (Vaidya & Kumar, 2006). In addition, quantitative methods have been proposed using heuristic methods (Kleining & Witt, 2001), simulation techniques (Rivera-Gómez, et al., 2020), and analytical methods (Angius, et al., 2018). Among approaches used for PdM, machine learning-based ones are considered to be the most suitable approaches because they can handle high-dimensional problems that consist of hundreds or thousands of variables such as voltages, flows, and currents (Susto, et al., 2014). There exist two main categories of machine learning-based techniques for PdM. The first one is supervised approaches where the failure information exists in the dataset. The second one is unsupervised approaches, where there is only process information, and no failure-related information exists. KNN generally depends on unsupervised learning procedures where distances between classes in the neighborhood play an important role in PHM. However, most often, the data are imperfect and uncertain. Indeed, the source providing the information may be insecure or prone to making mistakes or giving intentionally incorrect information. To tackle this problem, there are several theories that can be used such as probability theory, fuzzy set theory (Zadeh, et al., 1996), possibility theory (Dubois & Prade, 1995), and belief function theory (Dempster, 1967; Shafer, 1976). In this paper, we will manage this uncertainty within the framework of the belief function theory, which has been applied in different fields (Ben Ayed, et al, 2022). For it, the paper is organized as follows: Section 2 gives an overview of the basic concepts regarding the belief function theory. Section 3 gives an overview of the data analysis, including the approach used in the experimental verification to analyze the residuals. Section 4 gives the experimental results on the real use case where the evidential machine learning based approach for PdM has been implemented and tested.

2. THE BELIEF FUNCTION THEORY: BASIC CONCEPTS

The belief functions theory, also known by the evidence theory or the Dempster-Shafer theory (Dempster, 1967; Shafer, 1976), is an ideal tool for modeling uncertainty and imprecision. This theory has been interpreted differently using several models such as the Transferable Belief Model (TBM) (Smets, 1998). Consider a set Ω called the frame of discernment, containing the elementary and exclusive hypotheses of a given problem. The key point of this theory is the basic belief assignment (*bba*) which is noted $m: 2^{\Omega} \rightarrow [0,1]$ as a belief function that satisfies the following normalization condition:

$$\sum_{X \subseteq \Omega} m(X) = 1$$

A bba allows for the assignment of elementary beliefs on different combinations of hypotheses. The uncertainty about a hypothesis $A \subseteq \Omega$ is modeled by the value of the bba: the higher the bba is (close to 1), the more confidence there is in *A*. However, if the value of bba is low (close to 0), it means that few pieces of evidence support *A*. The hypothesis *A* is called focal element if $m(A) > 0$. Other belief functions can be defined such as the *credibility function (Bel)* and the *plausibility function (Pl)*. They are defined, respectively, as follows:

$$Bel(A) = \sum_{\emptyset \neq B \subseteq A} m(B) \text{ and } Pl(A) = \sum_{A \cap B \neq \emptyset} m(B).$$

Given m_1 and m_2 deux *bbas* on Ω induced from two independent sources of information. They could be combined to yield a new bba m_{12} using, for instance, the Dempster's rule of combination which is identified by the orthogonal sum $\oplus$ and is defined such that:

$$m_{12}(\emptyset) = (m_1 \oplus m_2)(\emptyset) = 0$$

and

$$m_{12}(A) = (m1 \oplus m_2)(A) = \frac{\sum_{B,C \subseteq \Omega; B \cap C = A} m_1(B).m_2(C)}{1-\kappa} \quad with\ A \subseteq \Omega; A \neq \emptyset$$

where κ presents the conflict degree and is defined such as: $\kappa = \sum_{B \cap C = \emptyset} m_1(B).m_2(C)$

Finally, to make a decision within the belief function theory, several solutions have been proposed. For instance, the Pignistic Probability (*BetP)*, which is offered by Smet's TBM model (Smets, 1998), is considered as a powerful way for decision making. To do, the event $\omega \in \Omega$ having the highest value of BetP, according to the following equation, will be selected:

$$BetP(\omega) = \sum_{\omega \in A} \frac{m(A)}{|A|} \quad \forall \omega \in \Omega$$

3. EVIDENTIAL KNN BASED APPROACH FOR TURBOFAN ENGINES FAILURE PROGNOSIS

This section presents a case study of the proposed method using NASA's C-MAPSS dataset for engine performance degradation tracking and RUL prediction. The dataset is generated from a C-MAPSS commercial turbofan engine simulator (Dean, et al., 2012). The role of prognostics is particularly important in the context of aircraft engine health condition assessment due to the high cost associated with in-flight malfunctions, maintenance-related delays and cancellations, increased fuel consumption, and especially, potential loss of human lives. The main contribution of this work is to build a RUL prediction model using an evidential KNN. The proposed RUL prediction process is shown in Figure 1, and the procedure in each step is detailed in the following subsections.

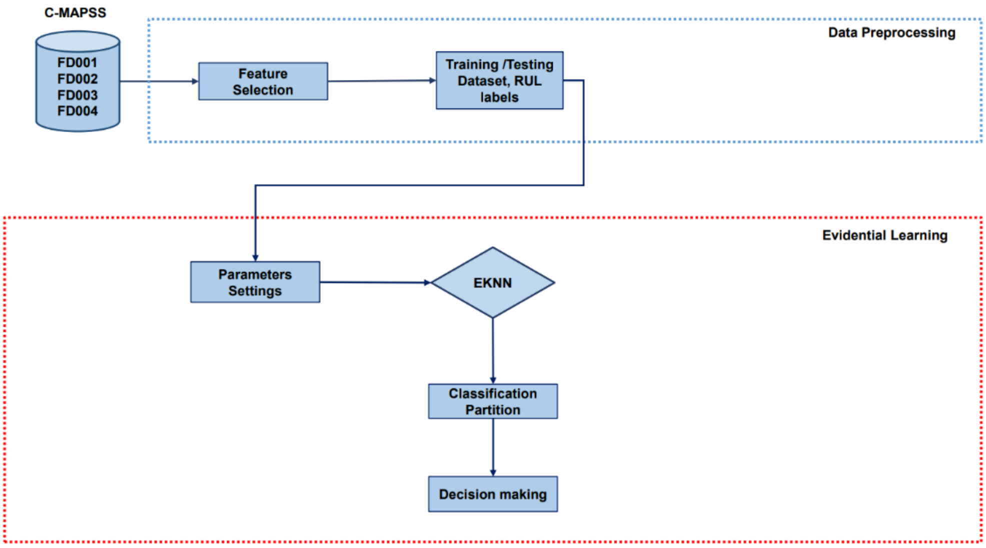

Figure 1. Flowchart of the proposed approach

3.1 Commercial Modular Areo Propulsion System

Prognostics estimates remaining useful component life. The remaining useful life estimates are in units of time. Commercial Modular Aero Propulsion System Simulation (C-MAPSS) (Abhinav, et al., 2008) datasets were generated to allow the development and benchmarking of various prognostic approaches. C-MAPSS is a tool for simulating a realistic large commercial turbofan engine. Figure 2 is a simplified diagram of the simulated turbojet engine showing the main components, such as the low-pressure compressor (LPC), the high-pressure compressor (HPC), the fan, and the combustor.

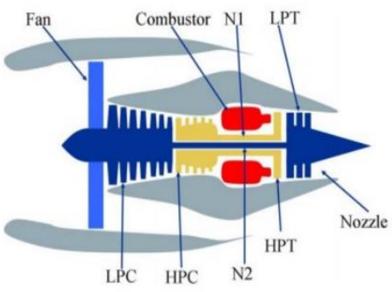

Figure 2. Simplified diagram of a simulated turbojet in C-MAPSS

The C-MAPSS datasets pose several challenges, in particular, management of high variability due to sensor noise, effects of operating conditions, and presence of multiple simultaneous fault modes are some factors that have a great impact on the generalization capabilities of prognostics algorithms. Four datasets have been generated which are called turbofan datasets and consist of training and test data. A learning instance represents the degradation of a turbojet engine seen by 21 sensors. At each operational cycle of an engine, one observation is logged in the form of a multivariate observation comprising the measurements of 21 sensors, three operational parameters, a timer feature, and the engine ID. The C-MAPSS data have been split into four sub-datasets (FD001, FD002, FD003, FD004). Each dataset can operate under different operating conditions and the system failure can be caused by two components: the turbine and the compressor. Thus, FD001 and FD003 operate under the same conditions although FD003 includes engines whose failure could be caused by either of the two mentioned components. Then, FD002 operates under 6 operating conditions as does FD004, while in FD004, as in FD003, the failure conditions cover both turbine and compressor failure as shown in Table 1.

Table 1. Information on the C-MAPSS dataset

Sub-Dataset names	**FD001**	**FD002**	**FD003**	**FD004**
Turbofan for train	100	260	100	249
Fault Modes	HPC	HPC	HPC, and Fan	HPC, and Fan
Operating condition	1	6	1	6

3.2 Data Pre-Processing

Pre-processing data is intended to transform the raw data into a format that is easier and more effective to use for future processing steps. For each subset, the 21 sensors record different data, and the data range between different sensors varies significantly. Feature Selection (FS) is the process of reducing the number of input variables when developing a predictive model, this step aims to select the best feature in the data set. FS helps understand the data, reduces computing needs, and improves performance. In fact, from analyzing the statistical properties of each feature, it was observed that not all sensor measurements provide useful information for RUL prediction. For example, the standard deviation of sensors 1, 5, 6, 10, 16, 18, and 19 is statistically insignificant in FD001, therefore, the variables will be excluded from the rest of the study. In addition, unprocessed data affected the performance of predicting the engine RUL, so it is essential to standardize the data. Data normalization is an essential step in data pre-processing. Normalization can improve the training time because all data used in training have the same scale, in the range of 0 and 1. For this purpose, the following formula is implemented, where X_{max} andX_{min} represent, respectively, the maximum and the minimum and values of each feature:

$$X_{\text{norm}} = \frac{X - X_{\text{min}}}{X_{\text{max}} - X_{\text{min}}}$$

Before proceeding with a complicated task such as RUL estimation, a simpler task such as turbojet health classification can show the complexity of the problem. The first and last 50 cycles of each unit are considered healthy and faulty respectively. We have chosen to delimit the classes by 50 cycles to have balanced data between the three classes.

3.3 Evidential Learning on the C-MAPSS Dataset

In order to manage uncertainty during the supervised learning step of our proposal, we perform the Evidential k-Nearest Neighbor (EK-NN) (Denoeux, 1995) that is a pattern classification method based on the belief function theory. Contrary to the hard KNN classification method, EK-NN offers, as output, a credal classification of data instances, which presents a richer information content. According to EK-NN method, the frame of discernment *Ω* contains the set of *N* possible classes. In the frame of C-MAPSS dataset classification, we define Ω as the set containing three classes such as: $\Omega = \{C1, C2, C3\}$, with $C1, C2, C3$ referring, respectively, to "Normal", "Moderate degradation", "Near-to-failure degradation". The first and last 50 cycles of each unit are considered as a criterion for determining the engine's Near-to-failure degradation and Normal status, respectively.

Let X_i be the set of all the n objects within the C-MAPSS dataset, defined as $X_i = \{X_1, X_2, \dots, X_n\}$, and X is a new object to be classified. Classifying X means assigning it to one class in Ω which is done through handling pieces of evidence to manage uncertainty regarding that assignment. We note, also, $N_K(X)$ as the ensemble of the K-Nearest Neighbors of X.

3.3.1 The EK-NN Method

Based on the selected training set of C-MAPSS dataset, the evidential k-NN method classifies every new instance X that should be assigned to one class of the selected neighbors modeled by $N_K(X)$. However, EK-NN models the knowledge regarding the assignment of each neighbor X_j to its class $C_q \in \Omega$ through a piece of evidence. For this reason, the cardinality of $N_K(X)$ identifies the number of bba functions to be handled, where each function supports a number of hypotheses towards the class of the object X to be classified. These pieces of evidence are built in function to the distance between X and every neighbor X_j (e.g., the euclidean distance). For every $X_j \in N_K(X)$, the knowledge that $L_j = C_q$ is seen as a piece of evidence that boosts our belief regarding the assigning of X to C_q. Since we cannot be totally certain towards this piece of evidence, the Dempster-Shafer formalism allows to model it by saying that only some part of our belief is committed to C_q. To do, EK-NN uses a function, called the simple support function, where the bba has only one focal element aside the frame of discernment Ω. It is, therefore, defined as follows:

$$m_{X,X_j}(\{C_q\}) = \alpha$$

$$m_{X,X_j}(\{\Omega\}) = 1 - \alpha$$

$$m_{X,X_j}(\{C\}) = 0 \qquad \forall C \in 2^{\Omega}\{\Omega, C_q\}$$

where α is defined in what follows in function to the euclidean distance between X and its neighbor X_j denoted by $d(X, X_j)$, a constant α_0 fixed to 0.95, and a positive parameter $\gamma_q$assigned to each class C_q:

$$\alpha = \alpha_0 exp^{-(\gamma_q^2 \text{ d(X,Xj)}^2)}$$

Once the K different bbas are generated by the EK-NN, they will be combined using the Dempster's rule of combination such that:

$$m_X = m_{X,X_1} \oplus m_{X,X_2} \oplus \dots \oplus m_{X,X_K}$$

3.3.2 The Credal Classification Partition and Decision Making

As a result of its strategy, EK-NN provides a credal classification partition that allows each object to be assigned, with a degree of belief, not only to all the classes but also to the total ignorance defined by Ω. In our context, this classification partition M presents an $n \times 4$ matrix; The rows of M are the instances in the C-MAPSS dataset and the columns refer respectively to $m_{X_j}(C1)$, $m_{X_j}(C2)$, $m_{X_j}(C3)$, and $m_{X_j}(\Omega)$, which makes the sum of all the columns' values, in every row, is equal to 1. To make decisions regarding the class of each object, we need to move on from the credal level to the pignistic level. To do so, the aforementioned pignistic probability method has been used.

4. EXPERIMENTAL ANALYSIS

In this article, the experiments are carried out on C-MAPSS data, which are assumed to be uncertain and imperfect. Classic RUL predictions do not take into account the uncertainty of the data, therefore the aim of this experimentation is to use evidential KNN to predict RUL and compare the performance with the existing machine learning models in the literature. For each prediction model, we obtain results from the RULs that we will compare to know the best model.

The evclass package in the R environment currently contains functions for the evidential K-nearest neighbor (EK-NN) rule (Denoeux, 1995; Zouhal, & Denoeux, 1998]). In contrast with classical statistical classifiers, evidential classifiers quantify the uncertainty of the classification using Dempster-Shafer mass functions. The main functions are *EkNNinit*, *EkNNfit*, and *EkNNval* for the initialization, training, and evaluation of the EK-NN classifier.

4.1 Parameters' Setting

The principle of the evidential K-nearest neighbor (EK-NN) classifier is explained in Section 3.3.1, and the optimization of the parameters of this model is presented in (Zouhal & Denoeux, 1998). Here, we focus on the practical application of this method using the functions implemented in *evclass* (Denoeux, 2017). The EK-NN classifier is implemented through three functions: *EkNNinit* for initialization, *EkNNfit* for training, and *EkNNval* for testing. Let us initialize the classifier with $K = 45$ neighbors. EkNNval classified instances in a test set using the EV-KNN classifier. EkNNval is used in the following form:

$$\text{EkNNval (xtrain, ytrain, } x \text{ tst }, K, ytst = y \text{ test, param } = EkNNfit)$$

The used parameters are:

- *Xtrain*: Matrix of size $ntrain \times d$, containing the values of the *d* attributes for the training data.
- *Ytrain*: Vector of class labels for the training data.
- *Xtst*: Matrix of size $ntst \times d$ x d, containing the values of the *d* attributes for the test data.
- *K*: Number of neighbors.
- *Ytest*: Vector of class labels for the test data.
- *Param*: Parameters, as returned by EkNNfit.

4.2 Evaluation Criteria

In this study, precision, recall, and F1 are used to evaluate the performance of the proposed model of RUL estimation. The confusion matrix is a cross table between the actual values and the predictions. Precision is the accuracy of positive predictions calculated by:

$$\text{Precision } = \frac{\text{TP}}{\text{TP} + \text{ FP}}$$

where TP is True Positives and FP is False Positives.

Recall is the fraction of correctly identified positives calculated by:

$$\text{Recall } = \frac{\text{TP}}{\text{TP} + \text{ FN}}$$

where FN is False Negatives.

F1 Score is a metric for comparing two classifiers. It is obtained by finding the harmonic mean of precision

$$\text{F1} = 2 * \frac{\text{Precision} * \text{Recall}}{\text{Precision} + \text{ Recall}}$$

and recall.

The support is the number of occurrences of each class.

The confusion matrix for the dataset FD001 indicates that 299 instances of class *Near-to-failure degradation* status were correctly predicted on the support of 998, 206 were wrongly predicted as Moderate degradation status, while the class Normal turbofan engine has only 89 wrongly classified instances on the support of 1011. This appears reasonable considering the difference in the number of instances. It is expected that the classifier mistakes class Moderate degradation status for class *Near-to-failure degradation status* as the environmental measurements or both classes do not differ significantly.

4.3 Results and Discussion

Overall, the models achieved accurate results for the prediction of the RUL of aircraft engines in the testing set. Based on the three evaluation criteria, Tables 2, 3, 4 and 5 present the performance of the trained models on the testing set for the four aforementioned datasets.

Table 2. Obtained results in terms of Precision, Recall, F1, and Support for FD001

Average	**Precision**	**Recall**	**F1**	**Support**
KNN	0.70458	0.7	0.7022	4 127
EV-KNN	**0.70471**	0.7	0.**7031**	4 127

Table 3. Obtained results in terms of Precision, Recall, F1, and Support for FD003

Average	**Precision**	**Recall**	**F1**	**Support**
KNN	0.74406	**0.73**	0.**7369**	4 944
EV-KNN	**0.74459**	0.71	0.7268	4 944

Tables 2 and 3 show the classification results of KNN and EV-KNN, when applied on the FD001 and FD003 datasets. The number of predicted instances is 4127 and 4944, respectively. In addition, we note that results are somehow competitive, with some benefits offered by EV-KNN. For instance, the EV-KNN classifies positive classes more accurately with a precision of 0.7047 for FD001 and 0.74459 for FD003, while KNN follows closely but achieves higher recall and F1 score on FD003.

Table 4. Obtained results in terms of Precision, Recall, F1, and Support for FD002

Average	**Precision**	**Recall**	**F1**	**Support**
KNN	**0.70710**	0.7	0.7034	10 752
EV-KNN	0.70705	0.7	**0.7035**	10 752

Table 5. Obtained results in terms of Precision, Recall, F1, and Support for FD004

Average	**Precision**	**Recall**	**F1**	**Support**
KNN	**0.73506**	0.69	0.7118	12 250
EV-KNN	0.73497	**0.7**	**0.7146**	12 250

As mentioned in Tables 4 and 5, EV-KNN and KNN are applied to FD002 and FD004, which contain a higher number of observations. The number of predicted instances is 10752 for FD002 and 12 250 for FD004, which provides more instances to accurately evaluate the results. We observe that in a different operation condition than in FD001 and FD003, the EV-KNN did not manage to achieve higher precision values than KNN. This might be due to the more complex behavior of the sensor measurements.

Finally, we remark that results show that the evidential based predictions can provide better performance in predicting the RUL. It has a slightly better performance compared to the hard KNN. However, the precision does not significantly outperform the KNN model. The evidential approach showed higher performance on smaller datasets such as FD001 and FD003, whereas on FD002 and FD004, the task is more complex, and the prediction is affected by the dataset type. In all predictions, the performance of the models on FD003 outperforms their performance on the rest of the datasets.

5. CONCLUSION

This article has implemented a supervised learning-based approach for turbofan engines failure prognosis and described how to manage the uncertainty within this data to develop and test prognostic algorithms. An aerodynamic propulsion system simulator, C-MAPSS, was used in this study. With the assumption that the data are uncertain and imperfect, the use of the evidential KNN model seems more effective than the hard KNN model according to the carried out experiments. The evidential technique has the potential to deal with complex sensor readings, and therefore avoid poor decision-making, which can lead to devastating consequences. However, the prediction aspect requires improvement and careful feature engineering by looking at similarities across similar units and identifying faulty sensor readings. In future studies, the predictive models should be improved by investigating more complex machine learning techniques merged with evidential theories, on larger datasets of different aerospace engine reading scenarios.

REFERENCES

Abhinav, S., Member IEEE, Kai, G., Do, S., Member, IEEE, Neil. E, Member IEEE, (2008). Damage Propagation Modeling for Aircraft Engine Run-to-Failure Simulation.

Angius, A., Colledani, M., & Yemane, A. (2018). Impact of condition based maintenance policies on the service level of multi-stage manufacturing systems. Control. Eng. Pract.76, 65–78.

Ben Ayed, S., Elouedi, Z., & Lefevre, E. (2022). CIMMEP: constrained integrated method for CBR maintenance based on evidential policies. *Applied Intelligence*, *52*(6), 6939-6954.

Chaouiya, C. (2007). Petri net modelling of biological networks. Briefings in bioinformatics, 8(4), 210-219.

Dean, K. F., Jonathan, A. D., & Jonathan, S. L. (2012). User's guide for the commercial modular aero-propulsion system simulation (C-MAPSS). Tech. rep.

Dempster, A. P. (1967) Upper and lower probabilities induced by a multivalued mapping. The annals of mathematical statistics, 325-339.

Denoeux, T. (1995). K-nearest neighbor classification rule based on Dempster-Shafer theory. IEEE Trans. Syst. Man Cybern. 25(5), 804–813.

Denoeux, T. (2017). Evidential Distance-Based Classification.

Dubois, D., & Prade, H. (1995). Possibility theory as a basis for qualitative decision theory. In IJCAI (Vol. 95), 1924-1930.

Kleining, G., & Witt, H. (2001). Discovery as basic methodology of qualitative and quantitative research. In Forum Qualitative Sozialforschung/Forum: Qualitative Social Research (Vol. 2, No. 1).

Lee, W. S., Grosh, D. L., Tillman, F. A., & Lie, C. H. (1985). Fault tree analysis, methods, and applications: a review. IEEE transactions on reliability, 34(3), 194-203.

Rivera-Gómez, H., Gharbi, A., KennÉ, J.P., & Montaño-Arango, O. Corona-Armenta, J.R. Joint optimization of production and maintenance strategies considering a dynamic sampling strategy for a deteriorating system. Comput. Ind. Eng. 2020, 140, 106273.

Shafer, G. (1976). A mathematical theory of evidence. Vol. 1. Princeton: Princeton university press.

Smets, P. (1998). The transferable belief model for quantified belief representation. In Quantified Representation of Uncertainty and Imprecision, 267-301. Springer, Dordrecht.

Susto, G. A., Schirru, A., Pampuri, S., McLoone, S., & Beghi, A. (2014). Machine learning for predictive maintenance: A multiple classifier approach. IEEE Trans. Ind. Inform., 11, 812–820.

Tierney, L. (1996). Introduction to general state-space Markov chain theory. Markov chain Monte Carlo in practice, 59-74.

Vaidya, O. S., & Kumar, S. (2006). Analytic hierarchy process: An overview of applications. European Journal of operational research, 169(1), 1-29.

Zadeh, L. A., Klir, G. J., & Yuan, B. (1996). Fuzzy sets, fuzzy logic, and fuzzy systems: selected papers (Vol. 6). World Scientific.

Zouhal, L. M., & Denoeux, T. (1998). An evidence-theoretic k-NN rule with parameter optimization. IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), 28(2), 263-271.