



HAL
open science

On a Method to Measure Supervised Multiclass Model's Interpretability: Application to Degradation Diagnosis (Short Paper)

Charles-Maxime Gauriat, Yannick Pencolé, Pauline Ribot, Gregory Brouillet

► To cite this version:

Charles-Maxime Gauriat, Yannick Pencolé, Pauline Ribot, Gregory Brouillet. On a Method to Measure Supervised Multiclass Model's Interpretability: Application to Degradation Diagnosis (Short Paper). 35th International Conference on Principles of Diagnosis and Resilient Systems (DX 2024), Nov 2024, Vienna, Austria. 10.4230/OASIS.DX.2024.27 . hal-04862630

HAL Id: hal-04862630

<https://laas.hal.science/hal-04862630v1>

Submitted on 3 Jan 2025

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License

On a method to measure supervised multiclass model’s interpretability: application to degradation diagnosis

Charles-Maxime Gauriat, Yannick Pencolé, Pauline Ribot, Gregory Brouillet

November 11, 2024

Abstract

In an industrial maintenance context, degradation diagnosis is the problem of determining the current level of degradation of operating machines based on measurements. With the emergence of Machine Learning techniques, such a problem can now be solved by training a degradation model offline and by using it online. While such models are more and more accurate and performant, they are often black-box and their decisions are therefore not interpretable for human maintenance operators. On the contrary, interpretable ML models are able to provide explanations for the model’s decisions and consequently improves the confidence of the human operator about the maintenance decision based on these models. This paper proposes a new method to quantitatively measure the interpretability of such models that is agnostic (no assumption about the class of models) and that is applied on degradation models. The proposed method requires that the decision maker sets up some high level parameters in order to measure the interpretability of the models and then can decide whether the obtained models are satisfactory or not. The method is formally defined and is fully illustrated on a decision tree degradation model and a model trained with a recent neural network architecture called Multiclass Neural Additive Model.

1 Introduction

Condition monitoring plays an inevitable role in the safety of any industrial system, especially when it comes to the sensitive parts of machines, like the bearings in rotating machinery, which are prone to faults. Fault diagnosis of rotating machinery is a technique of fault detection, isolation and identification, which can be used as an assistance for system maintenance. As stated for instance in [9], there are three basic tasks of fault diagnosis in such machines: determining whether the equipment is normal or not; finding the incipient failure and its reason; predicting the trend of fault development. Therefore, essentially, fault diagnosis can be regarded here as a pattern recognition problem that aims at determining the current degradation state of the rotating machinery based on the available set of measurements and its possible future trend (prognostics). As a powerful pattern recognition tool, Artificial Intelligence (AI) and especially

Machine Learning (ML) has attracted great attention from many researchers and shows promise in rotating machinery fault recognition applications [3, 2].

While the main reason for using ML techniques is usually the models' performance (accuracy score, computational speed), the question about the ability of humans to understand them is of great importance. The interpretability of models is essential as soon as these models are effectively used in practice to solve problems and provide decisions for humans and/or for their businesses [16].

In our industrial context, such models are used to solve maintenance decision problems. Maintenance consists in optimally deciding when to replace a component in a system (like a machine tool) so that the system is always operating properly and manufacturing waste is prevented. To get such a maintenance strategy, the objective is to add relevant sensors in the system, to acquire time series at operating time and use a degradation model to check the current health of every component based on these time series. Amongst these ML methods, Neural Network approaches are more and more used as they can handle large and complex computations to produce efficient models. However, even if already proposed ML methods greatly improve the degradation diagnosis of such equipments, degradation models obtained by deep learning techniques are known to be black-boxes, meaning that they cannot be open to understand their decisions as interpretable. Some previous works have reached a certain level of explainability like [11] but these results do not give the full insight about how interpretable the effective model's choices are. However, it is important for a human operator to understand how an algorithmic model determines a maintenance decision with respect to human-interpretable physical laws and quantities: *how and why* such a model plans the decision. In fact, by providing the physical reason why the model decides about a degradation level, model's interpretability not only provides a diagnosis of the equipment but also the confidence needed by a human operator towards trained models [12].

In this paper, we address the problem of how to effectively and quantitatively measure the interpretability of a model that results from a Multiclass Supervised Learning problem and apply this framework to degradation model of rotating machinery. The proposed method is agnostic in the sense that it does not rely on a specific ML technique. We actually aim at applying it to different ML techniques that are currently used to solve the maintenance problem (i.e. Decision Tree (DTs), Multi-Layer Perceptrons (MLPs), Neural Additive Model for multi-class supervised learning (MNAMs) [5]). We also propose that the interpretability measure is parameterized, and the initialization of these parameters is the responsibility of the human decision maker.

The paper is organized as follows. Section 2 first discusses the notion of interpretability with respect to the notion of local/global explainability in models. Section 3 describes a first degradation model as a simple decision tree model that will be used throughout this paper as an illustration of the concepts that we introduce to measure interpretability. Section 4 formally defines the way to score interpretability. Finally, Section 5 discusses the way to apply the interpretability scoring framework to a second degradation model that has been trained with our own recent neural network architecture: Multiclass Neural Additive Models [5].

2 About interpretable models

Lately, the Explainable Artificial Intelligence (XAI) community has been using various terms referring to the comprehension of machine learning models: interpretability, explainability, intelligibility or even comprehensibility [13]. As the vision behind these concepts seems to be fuzzy and does not refer to a monolithic concept so far [8], we have decided in this paper to propose the following definitions for explainability and interpretability.

Explainability of a machine learning model is based on the ability to obtain rules that highlight the relations between attributes and predictions. These rules are conditional functions like: "IF $A > 1$ and $B < 4$ THEN PREDICTION P" with two features A and B in this example. Based on these rules, two types of explainability can be defined: local explainability and global explainability.

Local explainability means the ability of identifying, for a given prediction, the rule that shows the relation between the feature and that specific prediction. The rule explains why a particular prediction was made based on the features of the instance in question. Local explainability is inherent in Decision Trees (DT) because the rule corresponding to a specific prediction can be extracted from the leaf node where the prediction concludes.

For other methods such as MLP, it can be difficult to extract such a rule because of the complex interactions between features and the non-linearity of the model. To address this challenge, various post-prediction processing methods have been developed. For instance, these include extracting rules from the weights and activations of a MLP [4]. Additionally, widely-used methods, such as Local Interpretable Model-agnostic Explanations (LIME) [15] and SHapley Additive exPlanation (SHAP) [11], can be used to approximate the rules linking features to predictions. However, all these methods only provide approximations of the rules and, therefore, do not always accurately reflect the model's decision-making process.

Global explainability is a stronger concept than local explainability, achieved by extracting the entire set of rules used by a model. This allows for a general understanding of how the model operates and how it will make decisions for different instances. Whatever the predictions the model will perform in the future, they can be explained by these feature-based rules. Global explainability ensures the algorithmic transparency of the models because of this set of rules; it becomes possible to determine the classification of any instance without explicitly running it through the model. DT are inherently globally explainable. A DT rule is defined by a branch of the tree and the thresholds defined at each node along the branch. All rules can be displayed, allowing a human to manually predict outcomes by following these rules. In contrast, Multi-Layer Perceptrons (MLP) do not provide precise model rules because a fully connected network is highly complex. Therefore, MLPs are not globally explainable.

Interpretability is the model's ability to be understood by humans. This is achieved by keeping the explanation as minimalist as possible. The fewer the number of rules and the fewer the number of features used within to make a prediction, the higher the interpretability will be. According to our definitions, interpretability implies global explainability. Interpretability is bringing up together concepts as simulatability, decomposability from [8] and comprehensibility from [13] which are about maximizing the human comprehension using the minimum set of rules. Any type of model may lack of interpretability.

For instance, a DT may have a high number of rules or be excessively deep. Unlike global explainability, interpretability is more of a quantitative property that should be assessed or scored.

3 Running Example

The way to score model’s interpretability will be illustrated all along this paper with the following example that is a multi-class supervised learning problem. This example has been selected as it is based on real data, and it is simple.

3.1 Experimental setup

In Bosch-Rodez, we set up an experimental platform that records time series on a set of spindles (critical parts of machine tools under maintenance). Figure 1 illustrates this experimental platform. To initiate the experiment and get the training measured data, an available set of five spindles has been installed on the test-bed in a closed chamber that simulates the real operating conditions. Each spindle is connected with cooling oil inlet and outlet for cooling down the spindle (lubrication of the bearings). Then a VSA 005 sensor (the accelerometer) has been screwed on the front part of the spindle as it would be inside a machine tool using this type of spindle at operating time. This vibration sensor is thus located between the pair of bearings of the spindle. It measures the vertical acceleration just on the top of these bearings. The cable of the VSA sensor is connected to an IFM VSE 100 module that contains the software allowing to record the vibratory signals, a module that is used at operating time. The objective is to acquire data at 9K RPM so that frequency measurements are within the range of the available VSA sensor.

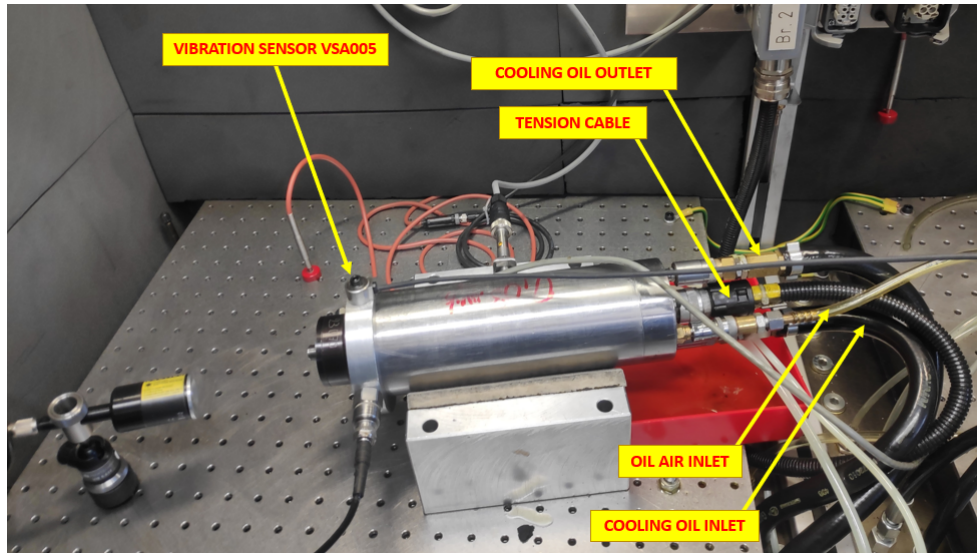


Figure 1: Experimental test-bed.

These time series consist of vibration data. The degradation state of each spindle that is measured is known by expertise and leads to five classes $\mathcal{C} = \{1, \dots, 5\}$ that represents the degradation of a spindle: from Class 1 that characterises new spindles to Class 5 that characterises worn-out and failing spindles. In this context, the objective of maintenance is to be able to determine at operating time what is the effective class of the measured spindle. Especially if the operating spindle is in Class 4, its replacement would be necessary as it is about to fail.

3.2 Degradation model

Our objective is then to learn from the time series a degradation model M that is able to predict the degradation class $c \in \mathcal{C}$ of a spindle. To do so, available raw time series have been converted to spectrograms (short fast Fourier transforms) so that each time series is represented by a set of 409 frequency amplitudes (from 0Hz to 9985Hz with a resolution of 24.414Hz). We denote by X the available dataset, each individual $x \in X$ is then composed of 409 features. In the following, $fa(x, f)$ will denote the frequency amplitude at the frequency f in the individual x . Experts were able to label any individual x with an ageing class denoted $\ell(x) \in \mathcal{C}$.

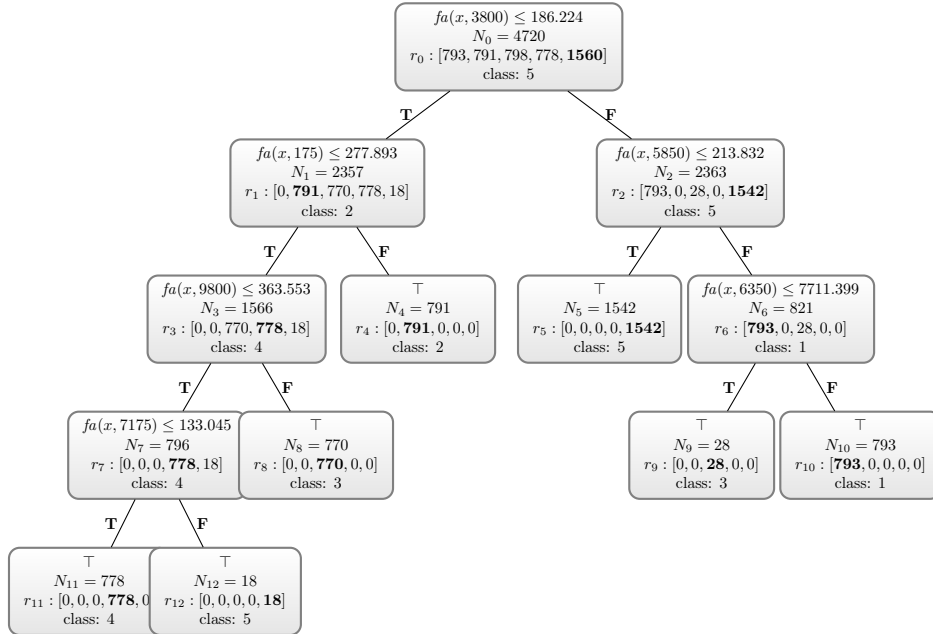


Figure 2: Degradation diagnostic model of a spindle based on vibration frequencies.

Figure 2 presents the degradation model M that has been trained on these labeled data. It is a DT that can be used to assist the maintenance decision. For instance, if at operating time, a new individual x is available and if $fa(x, 3800) \leq 186.224$ and $fa(x, 175) > 277.893$, the model M predicts the spindle is currently

in class 2 and no replacement is required yet. Each node is also associated with the following information: N_i is the number of individuals from X that are covered by the node, and a vector of numbers that shows the distribution of these individuals with respect to their label. For instance the root node covers 4720 individuals (that is the size of X), 793 individuals are labeled 1, 791 are labeled 2, etc.

While this DT has been trained on real data, it must be however noticed that this model is not satisfactory yet in terms of performance and accuracy as the available dataset X is still limited and more recordings are required to enrich the dataset X . This model is however satisfactory for the sake of illustration of the way to measure its interpretability.¹

4 Rule-based interpretability score

This section formally introduces a method to score the interpretability of a model that is globally explainable. For the sake of illustration, this method is applied to the DT that is detailed in Section 3. As detailed in Section 2, a model M is globally explainable if there exists a set of feature-based rules that can be used to provide an explanation for any prediction.

In the following, A denotes the set of available attributes of the problem. An individual is generally denoted $x = (x_1, \dots, x_{|A|})$ where x_i is the value of x associated with the i^{th} attribute of A . The space of every possible individuals is denoted \mathcal{X} and the available dataset for training the model is denoted X , obviously $X \subseteq \mathcal{X}$. Let B denote a subset of A , \mathcal{X}_B denotes the projection of \mathcal{X} on the attributes in B .

Definition 1 (Rule) *Let B be a subset of features A , a rule r over B on a class $c \in \mathcal{C}$ is a Boolean function*

$$\begin{aligned} r : \mathcal{X}_B &\rightarrow \{\mathbf{T}, \mathbf{F}\} \\ x_B &\mapsto r(x_B). \end{aligned}$$

The purpose of a rule r is to assert whether the prediction of an individual $x \in \mathcal{X}$ is c or not *according to the rule*. Such a rule can be used as an explanation of why the prediction of individual x given by the underlying model M is class c :

M predicts class c for individual x because $r(x)$ is true.

As opposed to a model, a rule r is usually defined over a subset of features B . This subset is called the *support* of rule r . In the following, $r(x)$ denotes the effective prediction $r(x_B)$ where B is the support of r .

In Figure 2, each node of the DT is associated with a rule. For instance, consider the node associated with rule r_3 . This node is associated with class 4 as it is the class of the largest set of individuals from the training dataset X covered by this node (i.e. 778 individuals over 1566). Rule r_3 associated with this node informally states that any individual whose frequency amplitude at 3800Hz is smaller than 186.22 and frequency amplitude at 175Hz is smaller than 277.893 should be of class 4. The support of r_3 only consists of two attributes

¹Interpretability does imply accuracy and reciprocally.

(frequency amplitude at 3800Hz and frequency amplitude at 175Hz) over 409 attributes. Rule r_3 on class 4 is formally defined as follows:

$$r_3(x) \text{ iff } fa(x, 3800) \leq 186.224 \wedge fa(x, 175) \leq 277.893$$

Another example is rule r_5 on class 5 that is associated with a leaf node of the DT. The support of rule r_5 is the frequency amplitudes at 3800Hz and 5850Hz, formally:

$$r_5(x) \text{ iff } fa(x, 3800) > 186.224 \wedge fa(x, 5850) \leq 213.832$$

Note that the root node of the tree is also associated with a rule r_0 on class 5. The support of r_0 is empty and $\forall x \in \mathcal{X}, r_0(x)$.

Definition 2 (Rule coverage) Let r be a rule on a class c , the coverage of the rule r over a dataset X is:

$$cvr(r) = \{x \in X, r(x)\}. \quad (1)$$

The correct coverage of the rule r over a dataset X is:

$$ccvr(r) = cvr(r) \cap \{x \in X, \ell(x) = c\}. \quad (2)$$

Two rules r_1 and r_2 are disjoint if:

$$cvr(r_1) \cap cvr(r_2) = \emptyset. \quad (3)$$

Back to Figure 2, the dataset X is composed of $N_0 = 4720$ individuals and rule r_0 covers all of them ($|cvr(r_0)| = N_0$). Generally speaking, for every rule r_i in Figure 2, $|cvr(r_i)| = N_i$. Any node of the tree displays the size of the associated set $ccvr(r)$ in bold: for example, $|ccvr(r_1)| = 791$ (among 2357 individuals covered by r_1 , only 791 are labeled with class 2). Disjoint rules in a DT are rules that do not belong to the same branch. For instance, rules r_{12} and r_5 on class 5 are disjoint, while r_2 and r_5 are not (rule r_2 subsumes r_5).

Definition 3 ((τ, ε)-rule) Let $\tau \in [0, 1]$, $\varepsilon \in [0, 1]$, a (τ, ε)-rule on a class c over a dataset X is rule r such that:

$$\frac{|ccvr(r)|}{|\{x \in X, \ell(x) = c\}|} \geq \tau \quad (4)$$

$$\frac{|cvr(r) \setminus ccvr(r)|}{|cvr(r)|} \leq \varepsilon \quad (5)$$

Parameter τ is a *minimal coverage rate* that is selected by the decision maker before the extraction of a set of interpretable rules. A rule r with an effective coverage rate lower than τ is considered by the decision maker not enough significant as it covers too few individuals from the dataset. Rules with low coverage rate might in fact be due to overfitting and could not be considered as interpretable. Parameter ε is the *maximal error rate*. This error rate is also selected by the decision maker before the extraction of a set of interpretable rules. If a rule on a class c has a low error rate ε , it means that the explanation provided by the rule for class c is given with a high level of confidence $1 - \varepsilon$. Individuals misclassified by the rule and included in the error

rate can be considered anomalies or noise if this rate is low. This is valid within the context of the dataset used to train the model, as it reflects the model’s perspective based on the data it has seen.

Here are a few examples based on Figure 2. Rule r_2 is on class 5, $|ccvr(r_2)|$ is 1542. The number of individuals in X labelled with class 5 is 1560 (see details in node r_0). So the coverage rate of r_2 is $(1542/1560) = 98.89\%$. Looking now at the error rate of r_2 , it is given by $(2363 - 1542)/2363 = 34.74\%$. Therefore, by Definition 3, rule r_2 is a (τ, ε) -rule for any couple (τ, ε) such that $\tau \in [0, 0.9889]$ and $\varepsilon \in [0, 0.3474]$. While r_2 is covering most individuals of class 5 in X , explaining that an individual $x \in X$ is of class 5 by rule r_2 (i.e. $r_2(x)$ is true) is 34.74% erroneous. As a second example, now let us have a look at rule r_8 on class 3. Its coverage is $770/798 = 96.49\%$ and the error rate is 0%. Any individual $x \in X$ such that $r_8(x)$ is of class 3, $r_8(x)$ explains the prediction of x without error. Moreover, 96.49% of the individuals of class 3 in X can be explained by r_8 . All the results are presented in Table 1.

Table 1: Couples $(\tau_{max}, \varepsilon_{min})$ for every rule in the decision tree of Figure 2.

Rules	r_0	r_1	r_2	r_3	r_4	r_5	r_6
Class	5	2	5	4	2	5	1
τ_{max} (%)	100	100	98.89	96.49	100	98.89	100
ε_{min} (%)	66.94	66.44	34.74	50.31	0	0	3.41

Rules	r_7	r_8	r_9	r_{10}	r_{11}	r_{12}
Class	4	3	3	1	4	5
τ_{max} (%)	100	96.49	3.51	100	100	1.1
ε_{min} (%)	2.26	0	0	0	0	0

We propose to score the interpretability of a model M based on the following definition.

Definition 4 ((T, P, τ, ε)-interpretability) A model M is $(T, P, \tau, \varepsilon)$ -interpretable for a class $c \in \mathcal{C}$ if there exists a set of rules $\mathcal{I} = \{r_1, \dots, r_k\}$ on class c from M such that:

- Rule r_i is a (τ, ε) -rule, for any $i \in \{1, \dots, k\}$
- $k \leq P$
- Any pair of distinct rules in \mathcal{I} is disjoint
-

$$T \leq \frac{|\bigcup_{i=1}^k ccvr(r_i)|}{|\{x \in X, \ell(x) = c\}|}$$

A model M is $(T, P, \tau, \varepsilon)$ -interpretable if it is $(T, P, \tau, \varepsilon)$ -interpretable for every class $c \in \mathcal{C}$.

To score the interpretability of a model M , the decision maker must set up four parameters and extracts the rules according to these parameter settings:

1. Parameter T is the global covering rate. This parameter ensures that the set of extracted rules covers a minimal amount of individuals in X labeled with the same class c . The higher T is, the more chance is that an new individual $x \in \mathcal{X}$ such that $M(x) = c$ can be explained by one of these extracted rules, hence a better interpretability.
2. Parameter P is the maximal number of rules to extract. The lower P is, the more interpretable the model is (for a given set of parameters T, τ, ε) as the set of extracted rules are then more concise.
3. Parameter τ , as explained above, is a minimal coverage rate for a rule to be part of the selection. If parameter τ is too low, the decision maker accepts to select rules that cover very few individuals with regards to the dataset X . A rule with a τ that is low may be trustful due for instance to overfitting problems.
4. Parameter ε is the maximal error rate. The higher ε is, the more error-prone, the selected rules will be.

As a first example, let us consider that $P = 1$, $T = 98\%$, $\tau = 80\%$ and $\varepsilon = 0\%$. With these settings, the decision maker is expecting that the model is highly interpretable by looking for one rule that covers most of the individuals without any error. In Figure 2, there exists such a rule, it is rule r_4 on class 2. Therefore, the DT is highly interpretable for class 2. For any individual $x \in \mathcal{X}$ such that the model predicts class 2, it can also provide the following explanation:

$$r_4(x) \text{ iff } fa(x, 3800) \leq 186.224 \wedge fa(x, 175) > 277.893 \quad (6)$$

In the context of maintenance, it simply means that the operator does not require any maintenance on the spindle as it is still in class 2 and the explanation provided by the model for this degradation class is that the current frequency amplitude at 3800Hz is lower than 186.224 (the spindle is not new) and the current frequency amplitude at 175Hz is greater than 277.893 (the spindle has not yet reached class ≥ 3). The model is as highly interpretable for class 3. The rule that covers most of the individuals of class 3 is rule r_8 but its covering rate is below $T = 98\%$, there is no rule that can be selected for class 3 based on the previous settings. The model is $(96\%, 1, 80\%, 0)$ -interpretable for class 3, it is also $(100\%, 2, 3\%, 0\%)$ -interpretable: by selecting rules r_8 and r_9 , there is a full coverage, and it is not error-prone, however, the selection of r_9 requires a low minimal covering rate which might be considered as a suspicious explanation.

Looking now at the interpretability of the model from a global viewpoint, by setting $P = 1$, $T = 96\%$, $\tau = 80\%$ and $\varepsilon = 0\%$, it can be noticed one rule can be extracted from the tree for each class $c \in \mathcal{C}$, namely: $1 \rightarrow r_{10}$, $2 \rightarrow r_4$, $3 \rightarrow r_8$, $4 \rightarrow r_{11}$, $5 \rightarrow r_5$. Therefore, the model is $(96\%, 1, 80\%, 0)$ -interpretable. However, if the decision maker looks for rules with the maximal coverage 100%, then it minimally enforces $P = 2$ and $\tau = 2\%$, the same model is therefore

(100%, 2, 1%, 0)-interpretable (the extracted rules are then all the rules associated with the leaf nodes of the tree).

As this DT has been trained on a small experimental dataset, it has excellent performance due very likely to overfitting. Most of the time, rules are error-prone, hence the high interpretability of this model. Suppose now for the sake of illustration that nodes associated to rules r_7, r_8, r_{11}, r_{12} are not present in this tree. Then the model would be (100%, 1, 100%, 51%)-interpretable on class 4. The only rule that could be used as an explanation for class 4 would likely fail to do so by providing an explanation for individuals that are actually not in class 4, hence a low interpretability of the model for class 4.

5 How to score the interpretability of a Multi-class Neural Additive Model?

5.1 Multi-class Neural Additive Models

There are plenty of supervised learning methods available at different scales that aim at learning interpretable models such as Linear Models, Decision Trees or Generalized Additive Models (GAMs) [10]. However, more complex and performant methods such as Multi Layer Perceptron (MLP) are usually required to learn accurate degradation models for maintenance. However, MLP are still considered as black box models so far, as the interactions between the hidden layers of the model cannot be interpreted. Recently, to overcome this issue in such models, Neural Additive Networks (NAMs) [1] have been introduced. This supervised method proposes to use the concepts of GAMs applied to neural network structures. To date, NAMs have been able to solve supervised task like regression problems or binary classification problems [14]. We have recently proposed the Multi-class Neural Additive Model (MNAM) as an extended version of the NAM algorithm for multi-class classification and applied it to solve a predictive maintenance problem [5]. The objective of this extension is to improve the interpretability of the models while keeping the benefit of a Neural Network architecture (performance, accuracy, ...). The MNAM architecture is presented in Figure 3 and briefly described here below (for more details see [5]).

A MNAM architecture is made of $|A|$ feature networks. Each feature network S_{nn_i} is composed of its input x_i , a structure H_i made up of successive hidden layers defined during the model design phase, and its output $f_i(x_i)$. The H_i structure can be composed of several hidden layers, generally made up of regular units, using a ReLU activation function. One of the problems with S_{nn} that have only one input feature is that they often struggle to approximate 1D sharp jump functions with the regular unit and a ReLU activation function on the first layer of H_i . To solve this issue, a new hidden unit, called EXp-centered-Unit (ExU), has been introduced [1] and is preferably placed in the first layer of the H_i structure. It can learn and adjust the weight parameters in logarithmic space. Each new hidden unit using an activation function σ compute $h(x)$ as follows:

$$h(x) = \sigma(e^w(x - b)). \quad (7)$$

In case that x may have negative values, these first layer units can also be

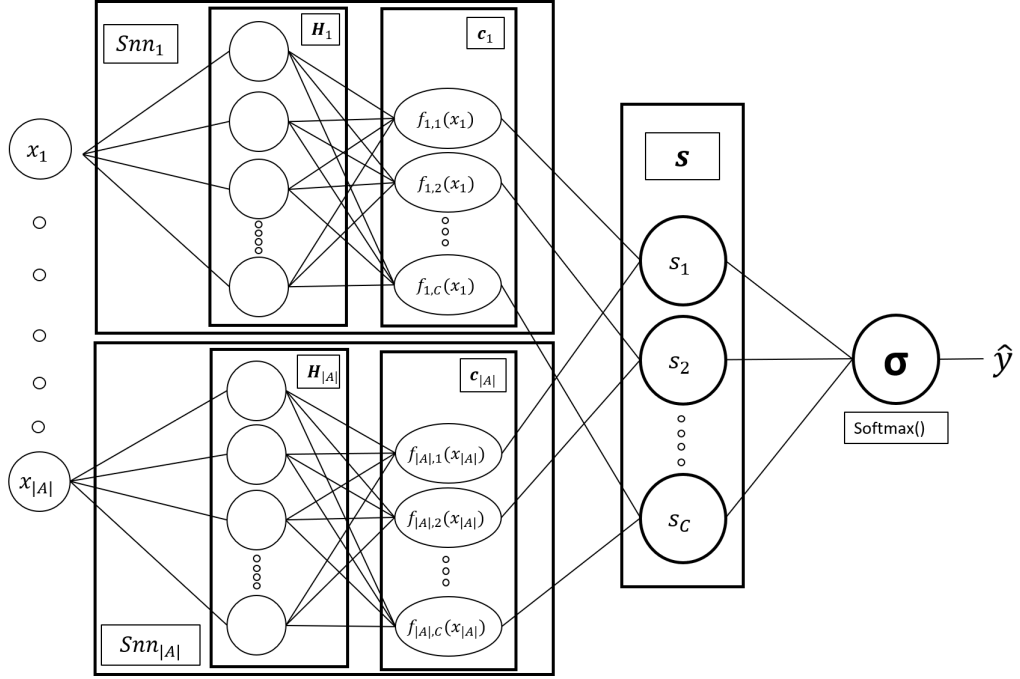


Figure 3: MNAM architecture.

replaced by ExpDive hidden units as proposed in [7]. A ExpDive hidden unit is defined by:

$$h(x) = \sigma((e^w - e^{-w}) \times (x - b)). \quad (8)$$

The structure of each S_{nn_i} is composed of one output $f_{i,c}(x_i)$ for each class c involved in the classification problem. The layer s of the MNAM architecture then gathers the outputs of the feature networks for each class c as a sum of these outputs:

$$s_i(x) = \sum_{i=1}^{|A|} f_{i,1}(x_i) + \beta_i \quad (9)$$

where β_i is a bias. Finally, since the network now has C outputs, a softmax function is applied to transform the layer s into a probability distribution that produces the MNAM output \hat{y} :

$$\hat{y} = \sigma_{softmax}([s_i(x)]_{i \in \{1, \dots, C\}}) = \left[\frac{e^{s_i(x)}}{\sum_{j=1}^C e^{s_j(x)}} \right]_{i \in \{1, \dots, C\}} \quad (10)$$

5.2 MNAM interpretability

MNAMs are glass-box models defined in [5] which use a methodology belonging to the family of Generalized Additive Models [6] known for their ability to

capture linear and non-linear relations between features and predictions while remaining globally explainable. The global explainability of MNAM mainly rely on the existence of so-called *shape functions* that can be computed once the model has been trained. They represent the exact description of the model decision process for all features [1]. The shape function of a feature a_i for the class $j \in \{1, \dots, C\}$ is given by the plot of all predictions from dataset X , that is, for any individual $x = (x_1, \dots, x_C) \in X$, the plot of $(x_i, f_{i,j}(x_i))$.

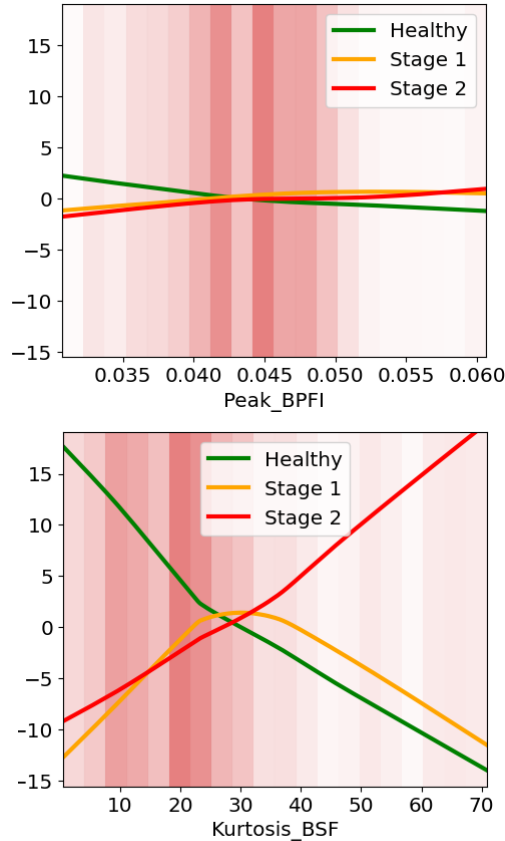


Figure 4: Example of two shape functions extracted from a Multi-class Neural Additive Model showing how the model classifies individuals with respect to the given features (namely Peak_BPFI, Kurtosis_BSF) [5]

Figure 4 illustrates a selection of two shape functions from a MNAM model that has been trained on a similar problem [5] as the one detailed in Section 3 (degradation diagnosis of bearings amongst the class *healthy*, *stage 1*, *stage 2*). The main difference here is that instead of spectrograms of frequency amplitudes as inputs, the selected features are statistics over a set of predefined frequency values that have a physical meaning (BSF: Ball Spin Frequency, BPFI: Ball Pass Frequency Inner, ...). This figure shows how the trained model actually classifies any individual of the dataset X with respect to its Kurtosis_BSF feature ($kbsf$) and its Peak_BPFI feature ($pbpfi$). The shape functions for Kurtosis_BSF

feature clearly defines a rule between the kurtosis and the degradation of the spindle bearings: for a Kurtosis_BSF within the range $[0, 25]$, it predicts a healthy stage, for a range in $[25,32]$ the stage 1 and for $[32,72]$ only stage 2. In this example, this rule is simple enough for a human to understand the model’s decision and is therefore interpretable. Similarly, as the shape functions associated with Peak_BPFI feature are all of them flat, it is pretty intuitive that the interpretation of the bearing’s degradation does not rely on this feature.

In the aim of measuring the interpretability of such a model with the framework that is defined in Section 4, rules as defined in Definition 1 must be extracted from the shape functions. Consider for the sake of simplicity, that the diagnosis machine learning problem defined here above is only based on both features $A = \{pbpfi, kbsf\}$. For any individual $x \in X$, according to the MNAM model, a possible way to design a rule r on class $c \in \mathcal{C} = \{healthy, stage1, stage2\}$ is as follows:

$$r(x) = \{pbpfi_c(x_{pbpfi}) + kbsf_c(x_{kbsf}) = \max_{c' \in \mathcal{C}} (pbpfi_{c'}(x_{pbpfi}) + kbsf_{c'}(x_{kbsf}))\}. \quad (11)$$

Intuitively speaking, such a rule r asserts that an individual x is in class c iff the sum of the shape functions for x for every feature in A is greater than any sum of the shape functions for x for another class c' (it is always the maximum by definition of the MNAM architecture, see Eq (10)).

It must be noticed that the support of such a rule is the entire set A . But the design of rules from the shape functions might be more complex to obtain finer rules with a partial feature support. For instance here, as soon as $x_{kbsf} > 3.0$, x_{pbpfi} is insignificant, so the design of a rule r only based on feature $kbsf$ is possible.

6 Conclusions and perspectives

In this paper, we have redefined the concepts of local and global explainability using feature-based rules derived from a machine learning model to diagnose system degradation. While global explainability is an absolute measure, the interpretability of each model is inherently subjective, posing challenges in scoring and comparing models across different methodologies. We introduced a novel approach to quantify interpretability using a framework based on four parameters defined by the decision-maker, referred to as $(T, P, \tau, \varepsilon)$ -interpretability. Our results demonstrate that this rule extraction methodology can be extended to other techniques, such as MNAM, due to its additive structure.

This work paves the way for several future research directions. Our immediate focus will be on enhancing the algorithm to incorporate the entire rule set from a model to accurately determine its $(T, P, \tau, \varepsilon)$ -interpretability score. Subsequently, we aim to generalize rule extraction across diverse machine learning methods to enable a universally applicable interpretability scoring system, independent of the underlying method.

References

- [1] Rishabh Agarwal, Levi Melnick, Nicholas Frosst, Xuezhou Zhang, Ben Lengerich, Rich Caruana, and Geoffrey E Hinton. Neural additive models: Interpretable machine learning with neural nets. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 4699–4711. Curran Associates, Inc., 2021.
- [2] Maryam Ahang, Mostafa Abbasi, Todd Charter, and Homayoun Najjaran. Condition monitoring with incomplete data: An integrated variational autoencoder and distance metric framework. In *IEEE 20th International Conference on Automation Science and Engineering*, Bari, Italy, 2024.
- [3] Xiaohan Chen, Beike Zhang, and Dong Gao. Bearing fault diagnosis base on multi-scale cnn and lstm model. *Journal of Intelligent Manufacturing*, 32:971–987, 2021. doi:<https://doi.org/10.1007/s10845-020-01600-2>.
- [4] Victor Contreras, Niccolo Marini, Lora Fanda, Gaetano Manzo, Yazan Mualla, Jean-Paul Calbimonte, Michael Schumacher, and Davide Calvaresi. A dexire for extracting propositional rules from neural networks via binarization. *Electronics*, 11(24), 2022. URL: <https://www.mdpi.com/2079-9292/11/24/4171>, doi:10.3390/electronics11244171.
- [5] Charles-Maxime Gauriat, Yannick Pencolé, Pauline Ribot, and Gregory Brouillet. Multi-class Neural Additive Models : An Interpretable Supervised Learning Method for Gearbox Degradation Detection. In *2024 IEEE International Conference on Prognostics and Health Management*, Spokane , WA, United States, June 2024. URL: <https://hal.science/hal-04562531>.
- [6] Trevor Hastie and Robert Tibshirani. Generalized Additive Models. *Statistical Science*, 1(3):297 – 310, 1986. doi:10.1214/ss/1177013604.
- [7] Minkyu Kim, Hyun-Soo Choi, and Jinho Kim. Higher-order neural additive models: An interpretable machine learning model with feature interactions, 09 2022. doi:10.48550/arXiv.2209.15409.
- [8] Zachary C. Lipton. The mythos of model interpretability: In machine learning, the concept of interpretability is both important and slippery. *Queue*, 16(3):31–57, jun 2018. doi:10.1145/3236386.3241340.
- [9] Ruonan Liu, Boyuan Yang, Enrico Zio, and Xuefeng Chen. Artificial intelligence for fault diagnosis of rotating machinery: A review. *Mechanical Systems and Signal Processing*, 108:33–47, 2018. URL: <https://www.sciencedirect.com/science/article/pii/S0888327018300748>, doi: <https://doi.org/10.1016/j.ymssp.2018.02.016>.
- [10] Yin Lou, Rich Caruana, and Johannes Gehrke. Intelligible models for classification and regression. In *Knowledge Discovery and Data Mining*, 2012. URL: <https://api.semanticscholar.org/CorpusID:7715182>.
- [11] Scott M. Lundberg and Su-In Lee. A unified approach to interpreting model predictions. *CoRR*, abs/1705.07874, 2017. arXiv:1705.07874.

- [12] Ricards Marcinkevics and Julia Vogt. Interpretability and explainability: A machine learning zoo mini-tour, 12 2020.
- [13] Joao Marques-Silva and Alexey Ignatiev. No silver bullet: interpretable ml models must be explained. *Frontiers in artificial intelligence*, 6:1128212, 04 2023. doi:10.3389/frai.2023.1128212.
- [14] Samad Moslehi, Hossein Mahjub, Maryam Farhadian, Ali Soltanian, and Mojgan Mamani. Interpretable generalized neural additive models for mortality prediction of covid-19 hospitalized patients in hamadan, iran. *BMC Medical Research Methodology*, 22, 12 2022. doi:10.1186/s12874-022-01827-y.
- [15] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. "why should i trust you?": Explaining the predictions of any classifier, 2016. arXiv: 1602.04938.
- [16] Cynthia Rudin. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature machine intelligence*, 1(5):206–215, 2019.