A Note on the Interpretability of Machine Learning Algorithms

By **Dominique GUÉGAN**
University Paris 1 Panthéon-Sorbonne and University Ca'Foscari in Venezia

Introduction

Following OECD (2016) "Artificial Intelligence (AI) is the ability of machines and systems to gain and apply knowledge and to carry out intelligent behavior". Applications range from education to social welfare, to energy and the environment. Advanced developments in the field of Machine Learning (ML) imply that machines will be able to learn from their experience and make their own decisions, without further input from humans, beyond initial design of the machine. Already, machines have surpassed the ability of humans to perform certain functions, such as image recognition and other intelligence-related tasks.

At the same time, generation of large volumes of data and the creation of centralized data repositories promise to drive growth across all sectors of society including agriculture, industry, banking, resource allocation, public health, education, and poverty reduction. Specifically, these data can be used to determine relationships, predict behaviors and outcomes, and establish dependencies between correlated variables. Algorithms like those developed through ML are used to generate automated outcomes using these data and improve the performance of algorithm-driven tasks, promoting improved business operations, management, and productivity as well as improved consumer-driven tasks.

Despite the growth of the benefits engendered by AI and the development of ML, some fears appear in the society revealed in particular by the press. Indeed, looking at it we observe that the influence and the importance of AI appears through disturbing titles like "software uses across the country to predict future criminals, is biased against black people", or "if you are not a white male, AI's use in healthcare could be dangerous for you", or "algorithms are making the same mistakes assessing credit scores that humans did a century ago", etc. Thus, even if AI creates advantages in the day life, it questions several issues, in particular a main question is: are the predictions provided by AI biased? A common idea is that the softwares are supposed to make policy more fair and accountable, but a huge literature shows that the predictions can be unfair, and coming from society expectations are very high, (Hardt *et al.,* 2016), (Zafar *et al.,* 2017), (Agarwal *et al.,* 2018) or Berg *et al.* (2018) with their works on discrimination, and also Miller (2019) among others.

In many cases, the reasons evocated to use AI consider that they can allocate resources with higher precision, can reduce the role of human instincts and prejudices, but in counterpart perpetuate biases against certain groups (for instance in case of racial profling). The opportunities associated to the use of AI concern higher accuracy, effectiveness, lower cost, higher effciency, preventing human biases and prejudices, transparency, consistency, more equal access to opportunities and resources. But at the same time we observe unfairness, unequal allocation of benefit or harm, opaqueness, inexplicability, unequal representation, invasion of privacy. Thus, it is important to focus on technical solutions to enhance fairness, explainability and accountability for ML systems, because if technical tools are useful, they are not sufficient.

Another reason of the necessity to understand how AI works refers to the potential impact that the European Union's new General Data Protection Regulation will have on the routine use of machine learning algorithms. It takes effect as law across the EU in May 2018 and has, in particular, the objective to restrict automated individual decision-making (that is, algorithms that make decisions based predictors user-level) which can "significantly affect" users. The law has created a "right to explanation" whereby a user can ask for an explanation of an algorithmic decision that was made about them. Interesting discussions on this subject can be found in Goodman & Flaxman (2017) and Whatcher *et al.* (2017) for instance.

Thus, three important topics emerge for acceptation of the use of ML by the scientist, industrial and regulatory communities. They concern issues of fairness, explainability, and accountability. The sources of unfairness are (i) data unfairness, (ii) algorithmic unfairness, (iii) impact unfairness. In this paper we are interesting by the question of interpretability. Indeed the concept of opacity seems to be at the very heart of new concerns about 'algorithms' among legal scholars, social scientists and engineers. Using the data as inputs, the algorithms produce an output (a classification i.e. whether to give an applicant a loan, or whether to tag an email as spam) or predictions. The output of the algorithm rarely does one have any concrete sense of **'how'** or **'why'** a particular decision has been arrived at from inputs. Additionally, the inputs themselves may be entirely unknown or known only partially. The question naturally arises, what are the reasons for this state of not knowing? Following recent researches we discuss some solutions related to these questions.

In order to provide global insight on the subject we begin to recall some definitions of fairness and provide some references on the subject in Section two. Section three is devoted to the definition of interpretability. In Section four we propose solutions for the interpretability of the algorithms. Section five focuses on two local solutions whose interest could be determinant for regulations. Section six gives an idea of the future possible ways to measure the interpretability of the algorithms. Section seven concludes.

Unfairness

Applications of fair machine learning, in the literature, concerns recidivism prediction, automated hiring, and face recognition (among others), where fairness can be understood, at least partially, in terms of well-defined quantitative metrics. However it has recently been shown that algorithms trained with biased data have resulted in algorithmic discrimination, in particular the statistical methods used in the US judicial system, pointing to the bias against African-American accused, considering that African-American accused were more likely to be wrongly labeled as higher risk of recidivism (Wadsworth *et al.,* 2018).

Thus, significant effort in the fair machine learning community has focused on the development of statistical definitions of fairness (Hardt *et al.,* 2016; Berk *et al.,* 2018) and algorithmic methods (Agarwal, 2018; Kusner *et al.,* 2017). The first notion of fairness which was introduced is "statistical parity", called also "group fairness" or "demographic parity" which equalizes outcomes across protected and non protected groups. Demographic parity requires that a decision is independent of a protected attribute meaning that membership in a protected class should have no correlation with the decision. Nevertheless this approach can create highly undesirable decision: for instance if the protected attribute is gender, one might incarcerate women who pose no public safety risk so that the same proportions of men and women are released on probation (Dwork *et al.,* 2012).

In order to avoid the limitation of the previous definition the "equalized odds" with respect to a protected attribute was introduced: the predictor and the protected attribute are independent conditionally on the output (Hardt *et al.,* 2016). An unfairness metric, which is defined in terms of

misclassification rates, has been introduced by Zafaz *et al.* (2017) called "disparate mistreatment". The authors call a decision-making process to be suffering from disparate mistreatment with respect to a given sensitive attribute (*e.g.,* race) if the mis-classification rates differ for groups of people having different values of that sensitive attribute (*e.g.,* black or white). Fermanian and Guégan (2020) provide updates on the subject.

To avoid unfairness in the ML process two stategies have been developed. The pre-process training ensures fairness of any learned model eliminating any sources of unfairness in the data before the algorithm is formulated. A major problem with this approach is that interaction effects (*e.g.,* with race and gender) containing information leading to unfairness are not removed unless they are explicitly included in the residualizing regression even if all of the additive contaminants are removed (Zemel *et al.,* 2013; Berk *et al.,* 2018; Lu *et al.,* 2016). In the post-processing training, after the algorithm is applied its performance is adjusted for instance by random reassignment of the class label previously assigned by the algorithm to make it fair (Feldman *et al.,* 2015; Hardt *et al.,* 2016). For instance a decision tree learner can be changed in splitting its criterion and pruning its strategy by using a novel leaf re-labeling approach after training in order to satisfy fairness constraints, (Kamiran *et al.,* 2010; Zliobaite, 2015; Agarwal *et al.,* 2019; Chzhen *et al.,* 2020).

However, it has been largely documented that simply removing certain variables from a model does not ensure predictions that are, in effect, uncorrelated to those variables. For example, if a certain geographic region has a high number of low income or minority residents, an algorithm that employs geographic data to determine loan eligibility is likely to produce results that are, in effect, informed by race and income (Hardt *et al.,* 2016). As unfairness is also a part of the non interpretability of the algorithms we question now the notion of interpretability and introduce some solutions.

A tentative to define interpretability

The need for explaining the decisions of expert systems was discussed as early as the 1970's. Nevertheless the definition associated to interpretability is not fixed. Following Biran and Cotton (2017), we can summarize most of the discussions around this concept as: "A key component of an artificially intelligent system is the ability to explain the decisions, recommendations, predictions or actions made by it and the process through which they are made." This ability can concern (i) the interpretability associated to justification which explains **'why'** one obtains such result, or (ii) the interpretability associated to explanation which corresponds to **'how'** we got this result. In that latter case, we focus on post hoc explanations. Interpretability can also refer to transparency which is the opposite of opacity or 'black-boxness', including the knowledge of the entire model, the knowledge of individual components such as parameters, and the knowledge of the training algorithm (Burrell, 2016; Doshi-Velez and Kim, 2017; Lipton, 2018).

Going through the literature, we see that there is no consensus on the definition of interpretability. It seems reasonnable to consider on the one hand the **explicability** which specifies **how** the model works, on the other hand **understanding** of the predictions **why**, which is a matter of **interpretation.** Discussing these two issues is important; indeed, in some application domains users need to understand the system's recommendations enough to legally explain the reason for the decisions. For instance in the medical domain, if a doctor makes a decision (say, recommends surgery) based on the prediction of a classification model and that leads to major harm to the patient, the doctor should understand the reason for the model's predictions in order to defend her/his decisions in court if she/he is sued for medical negligence. Legal requirements are also common in credit scoring applications, where a bank often has the legal obligation of explaining why a customer was denied credit.

Nevertheless even if the GDPR recitals state that a data subject has the right to "an explanation of the decision reached after algorithmic assessment," this requirement prompts the question: what does it mean, and what is required to explain an algorithm's decision? In fact a legally binding right to explanation does not exist in the GDPR, and we can consider that the right would only apply in limited cases: when, for instance, a negative decision was solely automated and had legal or other similar significant effects. Thus, explaining the functionality of complex algorithmic decision-making systems and their rationale in specific cases is important even if it is a technically challenging problem. A black box predictor is a data-mining and machine-learning obscure model, whose internals are either unknown to the observer or they are known but uninterpretable by humans. An explanation has to be an "interface" between machines and a decision maker that is comprehensible to humans (Whatcher, 2017).

In the state of the art a small set of models are considered easily understandable and interpretable for humans: decision tree, rules, linear models. A decision system based on a decision tree exploits a graph structured like a tree and composed of internal nodes representing tests on features or attributes (*e.g.,* whether a variable has a value lower than, equals to or greater than a threshold) and leaf nodes representing a class label. A decision tree can be linearized into a set of decision rules with the if-then form for instance.

Another set of approaches adopted to provide explanations are linear models. This can be done by considering and visualizing the features importance, *i.e.,* both the sign and the magnitude of the contribution of the attributes for a given prediction. If the contribution of an attribute-value is positive, then it contributes by increasing the model's output. Instead, if the sign is negative then the attribute-value decreases the output of the model. An intrinsic problem that linear models have when used for explanation is that when the model does not optimally fit the training data, it may use spurious features to optimize the error, and these features may be very hard to interpret for a human. Nevertheless these models can be used to interpret more complex models like Support Vector Machine, Deep Neural Networks or Convolution Neural Networks for example. We can distinguish two approaches the global or local approaches to make the algorithms more interpretable.

Does it exist interpretable solutions?

When a model is completely interpretable we are able to understand the whole logic of a model and follow the entire reasoning leading to all the different possible outcomes. In this case, we are speaking about global interpretability. At the contrary the local interpretability corresponds to the situation in which it is possible to understand only the reasons for a specific decision: only the single prediction/decision is interpretable. In the first case we face the "black box explanation problem" which consists in providing a global explanation of the black box model through an interpretable and transparent model: it **explains the model** (**how** it works). The second case concerns "the outcome explanation problem" which consists in providing an **explanation for the outcome** of the black box (**Why** this result). In the former case the solutions mimic the black box, and in the latter case the solutions provide a predictor which is locally interpretable.

We precise now the formal framework used in the rest of the paper. Let be X the vector of original inputs, $X \in \boldsymbol{X} \subset R^d$, we denote $X' \in \boldsymbol{X}$ a vector for its interpretable representation. The classifier assimilated to a blackbox need to be explained is designated by $b : \boldsymbol{X} \rightarrow R$ and $Y = b(X)$, shorthand for $Y = \{b(x) \mid x \in \boldsymbol{X}\}$, $b(X)$ is a black box predictor, whose internals are either unknown to the observer or they are known but uninterpretable by humans. Y is commonly called the output. The objective is to find an interpretable classifier $c : \boldsymbol{X} \rightarrow R$, $c \in C$, where C is a class of potentially interpretable models such as linear models, decision trees, or rule lists, and let Ω(C) be a measure

of the complexity (as opposed to interpretability) of c (deph of a tree, number of non-zeros in a linear model, etc). Thus c is an interpretable predictor yielding a decision c(X) = Y which can be given a symbolic interpretation comprehensible by a human, *i.e.*, for which a global or a local explanation is available.

Working with supervised ML, one uses a trained data set to train the predictor b, and a test data set D^T to evaluate its performance : let $\hat{Y}$ the outcome using the training data set, and $(X,\hat{Y}) \in D^T$. The objective is to verify how matches $\hat{Y}$ and Y=b(X). The interpretable predictor c need to be as close as possible to b in the sense that c(X) = b(X), for $(X,\hat{Y}) \in D^T$ and to mimic the results obtained through b.

A **global** interpretable model c_g is such that c_g = f (b,X), for some strategy f(.,.), and is derived from b and a subset of $X \in \boldsymbol{X}$. A **local** interpretable predictor is defined as c_l = f(b, X) derived using b and a **neighborhood** of x. The objective is to identify the function f. For simplicity, an input will be designated by x in the following. We introduce several strategies f permitting to determine global c_g or local c_l solutions and then propose specific solutions for interpretability or explainability.

Global approach

For instance, explaining neural networks with decision trees is a global approach. In that case, to build f a sort of "prototype" is generated for each target class in Y by using genetic programming to query the trained neural network b; the input variables X are exploited for constraining the prototypes; then the best prototypes are selecting for inducing the learning of the decision tree c_g. This approach leads to get more understandable and smaller decision trees starting from smaller data sets. Since 1996 single tree approximations for NN have been developed, Craven *et al.* (1996). New approaches are detailed in Guidotti et al. (2018).

Other approaches use decision rules. The solution f lies on a fair documentation of the process by inserting knowledge into neural networks, extracting rules from trained NNs, and using them to refine existing rules. Same kind of processes using rules-based classifiers have also been developed for tree ensemble or Support Vector Machines. All these solutions are not generalizable because they are strongly dependent on the black box b and on the specific type of decision rules c_g. This is a limitation of the method, thus **agnostic** methods have been developed with the objective to be ajustable for **all models**. The Generalized Additive Models (GAM) approach proposes a global solution based on splines functions, as bagged and boosted ensembles trees that chooses the number of leaves adaptively. If we denote c_j(x) the output obtained from tree j, j = 2, … , M, for classification purpose, the output b of the additive tree model is a weighted sum of all the tree outputs:ω

$$b(x) = \sum_{j=1)}^{M} \omega_j c_j\,(x)$$

where $\omega_j \in R$ is the weight associated to tree j. In their paper Lou *et al.* (2017) provide agnostic tools, see also Freitas (2014), and Ribeiro *et al.* (2016a).

Local approach

The global approach is also relatively limited due to the complexity of the models to explain, thus recent research have focused on specific local explanation providing explainability for the prediction. A common method is based on visualization. The technique is used to explain CNN process when they recognize images. The method is based on salient mask which is a part of an image or a sentence in a text. The explanation c_l of the prediction is provided through a visualization of the area of an image for instance. A technique consists in assigning a relevance score for each layer backpropagating the effect of a decision on a certain image up to the input level. The function f used to extract the local explanation c_l is always not generalisable and strictly tied with the convolutional neural network.

Thus, **agnostic** approach has also been developed for the local approach. The agnostic solutions proposed for the outcome explanation problem implements function f such that **any type** of black box b can be explained. All these approaches are generalizable by definition and return a comprehensible local predictor c_l. The more popular technique uses again the concept of additive models to weight the importance of the features of the input dataset. It provides a graphical explanation of the decision process by visualizing the feature importance for the decisions, the capability to speculate on the effect of changes to the data, and the capability, wherever possible, to drill down and audit the source of the evidence, some examples are in Poulin *et al.* (2006).

A way to find f providing a local interpretable predictor c_l is known as the Local Interpretable Model-agnostic Explanations (LIME) approach. LIME approach for f does not depend on the type of data, nor on the type of black box b to be opened, nor on a particular type of comprehensible local predictor c_l, thus LIME is model-**agnostic** in its philosophy. The main intuition of LIME is that the explanation may be derived locally from the inputs generated randomly in the neighborhood of the input x to be explained, and weighted according to their proximity to it. Linear models as comprehensible local predictor c_l are considered returning the importance of the features, and as black box b, classifiers like decision trees, logistic regression, nearest neighbors, SVM, and random forest are usually tested. A weak point of this approach is the required transformation of any type of data in a binary format (Ribeiro, 2016a). Other local interpretability methods have been developed like the counter factual explanations, but with a different objective (Ribeiro, 2016b). We detail now these two main local approaches.

Local interpretable solutions

Coming back to the previous discussion it can exist a trade-off between the performance of the model and the effort required to interpret it - especially in complex domains like text and image analysis, where the input space is very large. In these contexts, accuracy is usually sacrificed for models that are enough compact and transparent to be comprehensible by humans. To try to associate accuracy and transparency local approaches can be considered and they will permit to answer to the objective assignated to this paper: '**why'** and '**how'** a decision is taken using ML algorithms.

LIME method

The Local Interpretable Model-agnostic Explanations (LIME) method interprets individual model predictions based on locally approximating the model around a given prediction. LIME refers to simplified inputs x' as "interpretable inputs," and introduce a mapping $x = hx\ (x')$ which converts a binary vector of interpretable inputs into the original input space. Different types of h_x mappings are used for different input spaces. For bag of words text features, h_x converts a vector of 1's or 0's (present or not) into the original word count if the simplified input is one, or zero if the simplified input is zero. For images, h_x treats the image as a set of super pixels; it then maps 1 to leaving the super pixel as its original value and 0 to replacing the super pixel with an average of neighboring pixels (this is meant to represent being missing).

The important point concerning this approach is that it is based on local accuracy meaning that when approximating the original model classifier b for a specific input x, local accuracy requires the explanation model c_l to at least match the output of b for the simplified input x' (which corresponds to the original input x).

If we denote l a measure of how unfaithful c_l is in approximating b in the locality around x', measured by a local kernel π_x, we need to minimize l(b, c_l, π_x) while having Ω(C) be low enough to be interpretable by humans. The explanation f(x') produced by LIME is obtained solving

$$f(x') = argmin_{cl \in C}\ l(b, c_l, \pi_x) + \Omega(C).$$

If we use an additive model as explanation model for b in the framework proposed by LIME where a mapping x = $h_x(x')$ converts a binary vector of interpretable inputs into the original input space, then the local approximation c_i tries to ensure that $c_i(z') \approx b(h_x(z'))$ whatever z' ≈ x' (note that $h_x(x') = x$ even though x' may contain less information than x because h_x is specific to the current input x).

An explanation solution that is linear function of binary variables is provided by:

$$c_i(z') = \varphi_0 + \sum_{(1=i)}^{M} \varphi_i z_i$$

where $z' \in [0,1]^M$, M is the number of simplified input variables, and $\varphi_0 \in R$. This explanation solution attributes an effect φ_i to each input, and summing the effects of all input attributions approximate the output b(x) of the original model. Some applications are provided in Ribeiro (2016a). This approach provides a local answer for the interpretation or the understanding of the model b ('**how**'): it tries to illustrate the way by which the predictions have been provided.

Unconditional counterfactual explanation method

This approach provides a way to understand how a given decision has been obtained, and can provide grounds to contest it, and advice on how the data input can change his or her behaviour or situation to possibly receive a desired decision (*e.g.* loan approval) in the future. This knowledge can be associated to the works implied in the fairness of machine learning development. Suppose that you were denied a loan because your annual income was 30,000 euros. If your income had been 45,000 euros, you would have been offered a loan? Here the statement of decision is followed by a counterfactual, or statement of how the world would have to be different for a desirable outcome to occur. The counterfactual approach aims to create the smallest possible change to obtain a desirable result. The difficulty is to have knowledge of the relevance of all the factors at play and of their possible change. Hence the idea of having a counterfactual explanation which modifies the values from which we start as little as possible. Thus, the underlying idea is quite simple: the idea is to find a neighboor from the input which provides a different prediction with the same classifier.

As before denote X the input, Y the output, and the classifier b_ω trained by finding the optimal set of weights ω_i that minimises an objective loss function l(.) over a set of input data X, then the objective is to compute:

$$argmin_\omega\ l(b_\omega(x), y) + \sigma(\omega)$$

where σ is a regularizer over the weights. The idea is to find a counterfactual x' as close to the original point x as possible such that b_ω (x') is equal to a new target y'. We can find x' by holding ω fixed and minimizing the related objective:

$$argmin_{x'}\ max_\lambda\ \lambda(b_\omega(x') - y')^2 + d(x,x')$$

where d(,) is a distance function that measures how far the counterfactual x' and the original data point x are from one another. In practice, maximisation over λ is done by iteratively solving for x' and increasing λ until a sufficiently close solution is found. The choice of the distance d is important, that of λ is less so. Depending on the data, the distance could be the L^1 or L^2 norm, the Manhattan distance weighted by the inverse median absolute deviation. As local minima are a concern, one can initialize each run with different random values for x' and select as counterfactual the best minimizer of the previous equation. These different minima can be used as a diverse set of multiple counterfactuals.

Thus, with this approach, the original classifier does not change, the inputs are concerned in the sense that for a given classifier they determine the prediction. Thus the method tries to answer to the question of **why** we got these predictions. In that sense counterfactuals represent an easy first step that balances transparency, explainability, and accountability with other interests such as minimising the regulatory burden on business interest or preserving the privacy of others,

while potentially increasing public acceptance of automatic decisions. Thus, it may prove a highly useful mechanism to meet the explicit requirements and background aims of the GDPR, but it is important to have in mind that this approach may be the target of specific attacks such as those known as 'adversarial attacks'.

How to compute explanations of model prediction?

In the previous paragraph we provided some strategies to interpret ML algorithms and their predictions without quantifying the approximations which have been proposed. Some works tend to answer to this problematic.

Global feature attribution is represented in the literature by several methods: gain, split count, and feature permutation. Gain is the total reduction of loss or impurity contributed by all splits for a given feature. Though its motivation is largely heuristic, gain is widely used as the basis for feature selection methods (Friedman *et al.,* 2011). Split Count consists in simply to count how many times a feature is used to split. Since feature splits are chosen to be the most informative, this can represent a feature's importance (Chen & Guestrin, 2016). With permutation one randomly permutes the values of a feature in the test set and then observes the change in the model's error. If a feature's value is important then permuting it should create a large increase in the model's error (Auret & Aldrich, 2011). Two other measures have been also developed: the SHAP measures and the Quantitative Input Influence measures (QII).

SHAP values for LIME approach

The ability to correctly interpret a prediction model's output is extremely important but the solutions are not simple and it exist several strategies. These strategies engender appropriate user trust, provide insight into how a model may be improved, and support understanding of the process being modeled. In some applications, simple models (*e.g.*, linear models) are often preferred for their ease of interpretation, even if they may be less accurate than complex ones. It could be interesting to compare the accuracy of the 'interpretable' built when blackbox models are used. It is the objective of the SHapley Additive exPlanation (SHAP) values which are based on a unification of ideas from game theory (Strumbelj & Kononenko, 2014), and local explanations (Ribeiro, 2016a).

We have seen previously that the more understanding solutions are based on families of additive models. In the case of the local approximation proposed by the LIME method, the solution is provided by equation introduced in subsection 5-1. The explanation model $c_l\ (x')$ matches the original model b(x) when x = $h_x(x')$, and φ_0 = b(h_x(0)) represents the model output with all simplified inputs toggled off (*e.g.* missing). As soon as an additive interpretable model is built, the SHAP values characterize the additive inputs of this interpretable model. For instance if the additive model reduces to the very simple model:

$$c_l(x) = b(x) = \sum_{(j=1)}^{M} \omega_j x_j + a$$

The SHAP values are equal to φ_0 (x) = a and φ_j(x) = ω_j (x_j- E[x_j]). The exact computation of SHAP values is challenging, some examples can be found in Lundberg and Lee (2017) and Strumbelj and Kononenko (2014).

In case of a more general additive interpretable model like the model introduced in subsection 5-1, it has been shown that this model presents three interesting properties: local accuracy, missingness, and consistency: the local accuracy states that the sum of the feature attributions is equal to the output of the function we are seeking to explain; missingness states that features that are already missing (such that z= 0) are attributed no importance; consistency states that changing a model so a feature has a larger impact on the model will never decrease the attribution assigned to that

feature. The evaluation of the effect of missing features has on a model c_I is done through the use of the function h_x evaluating b(h_x(z)) and calculate the effect of observing or not observing a feature (by setting z= 1 or z = 0).

To compute the SHAP values for the approximation done through the linear model, we define $b_x(S)$ = b(h_x(z')) = E[b(x) | x_S] where S is the set of non zero indexes in z', and E[b(x) |x_S] is the expected value of the function conditioned on the inputs variables x_S belonging to S. SHAP values attribute φ_j values to each variable:

$$\varphi_j = \sum_{(SCN-\{i\})} \frac{|S|!(M - |S| - 1)!/M!}{M!} \; [b_x(S \cup \{i\}) - b_x(S)]$$

Where N is the set of all inputs.

The computational complexity of SHAP values have been extensively studied and some ways to reduce its computational times are discussed in Lundberg *et al.* (2018), Chen *et al.* (2019), and Erion *et al.* (2019).

QII and Counterfactual approach

The Quantitative Input Influence measures (QII) model the difference in the quantity of interest when the system operates over two related input distributions: the real distribution and a hypothetical (or counterfactual) distribution that is constructed from the real distribution in a specific way to account for correlations among inputs. Specifically, if we are interested in measuring the influence of an input on a quantity of interest of the system behavior, we construct the hypothetical distribution by retaining the marginal distribution over all other inputs and sampling the input of interest from its prior distribution. This choice breaks the correlations between this input and all other inputs and thus lets measure the influence of this input on the quantity of interest, independently of other correlated inputs.

Using the same notations as before. As QII quantifies the use of an input for individual outcomes, this quantity is defined for a particular individual. Denote x this individual and c the classifier retains in fine, the quantity E[c(.) = 1 | X=x] represents the expectation of the classifier c evaluating to 1 for the individual x. The influence measure, when the positive classification is the objective is computed as

$$\text{QII}(x_i) = \text{E}[c(X) = 1 \mid X=x] - \text{E}[c(XU_{xi})) = 1 \mid X=x]$$

where the random variable XU_{xi} corresponds to a randomized intervention on input x which is replaced with a random sample x_i. Thus, we have switched between the original distribution, represented by the random variable X, and the intervened distribution represented by XU_{xi}, Datta *et al.* (2016) for some applications.

As an example consider an analyst who asks: "What is the influence of the input gender on positive classification for women?" If it observes that 20% of women are approved according to his classifier, then, he replaces every woman's field for gender with a random value, and if he notices that the number of women approved does not change, this means that an intervention on the gender variable seems not causing a significant change in the classification outcome. Now the analyst can repeat the same process with 'weight lifting ability' variable and if the results show a 20% increase in women's hiring, therefore he can conclude that for this classifier, the variable 'weight lifting ability' has more influence on positive classification for women than gender. By breaking correlations between gender and weight lifting ability, can be a way to establish a causal relationship between the outcome of the classifier and the inputs. These facts are interesting and need to be more developed to verify what the correlations are really suppressed, and to understand the causal relationship between classifier's output and inputs.

Conclusion

The governance of decision making is an important task with developement of AI in industies and banking system. The requirement for explanation, a requirement codified through risk management in traditional sectors of industry and by the rules of certain professions (medicine, law), is also present in the AI sphere, where certain aspects are covered by legislation (for instance RGPD law).

To explain an algorithm enabling its users to understand what it does, with enough details and arguments to instill trust is a difficult task. The global and local solutions discussed here provided very interesting pistes, in particular the local counter-factual approach which could be a method from which the regulator could draw inspiration to verify the accountability of the algorithms. But many risks exist: one of them related to the interpretability of the models is known under the name of adversarial attacks and its study is in full expansion, we do not discuss it in this paper being beyond the objective of it, nevertheless some recent and interesting references are Dylan *et al.* (2018), Kim and Malde (2020), and Slack *et al.* (2020), for a review Bogroff and Guégan (2019).

In summary, for an algorithm to be explainable, its principles must be sufficiently documented to be comprehensible to all users; the transition from algorithm to code, then the execution of the program, must be formally verified. Ultimately, the explainability of an algorithm relies on rigorous methods, but also on a body of unformalized knowledge shared between human beings. As a result, a compromise has to be found between learning capacities and explainability. This compromise needs to be evaluated in relation to the field of application: while explainability is not in principle essential in applications such as games, it is crucial once the interests, rights or safety of people are concerned. From a social point of view people do not ask why an event P happened, but rather why event P happened instead of some event Q (Papernot, 2018; Alvarez-Melis, 2018).

It exist several works coming from the legal literature which propose procedures for the transparency of the source code in case of auditing purpose but also for the understanding of the users, making software verification, fairness random choices, disclosing commitments. They suggest the developers to publish in advance commitments explaining how the systems do without disclosing how those systems work up front. These procedures are complementary to the previous analysis and preaches for the need of platforms and algorithms to be evaluated (compliance, fairness, trustworthiness, neutrality, transparency…). This will contribute to good algorithm governance. In a complete paper, Kroll *et al.* (2016) provide solutions to make accountable machine learning algorithms, recommandations are also done in Cerna (2018), Linkov *et al.* (2018). Rules, regulation and governance are addressed in Barocas *et al.* (2013).

Ultimately, the role of humans must be preserved throughout the process of explicability, which requires significant expertise from those in charge of systems using AI in decision-making processes.

References

AGARWAL A., BEYGELZIMER A., DUDIK M., LANGFORD J. & WALLACH H. (2018), "A reductions approach to fair classification", arXiv preprint arXiv:1803.02453.

BIRAN O. & COTTON C. (2017), “Explanation and justification in machine learning: A survey”, in IJCAI-17 workshop on explainable AI (XAI), Vol. 8, No. 1, pp. 8-13.

BAROCAS S., HOOD S. & ZIEWITZ M. (2013), “Governing algorithms: A provocation piece”, Available at SSRN 2245322.

BERK R., HEIDARI H., JABBARI S., KEARNS M. & ROTH A. (2018), "Fairness in criminal justice risk assessments: The state of the art. Sociological Methods and Research", online publication, https://doi.org/10.1177/0049124118782533 %0049124118782533.

BOGROFF A. & GUEGAN D. (2019), "Artificial Intelligence, Data, Ethics An Holistic Approach for Risks and Regulation", University Ca'Foscari of Venice, Dept. of Economics Research Paper Series, (19).

BURRELL J. (2016), "How the machine 'thinks': Understanding opacity in machine learning algorithms", *Big Data and Society,* 3(1), 2053951715622512.

Cerna Collectif (2018), "Research Ethics in Machine Learning" [Research Report] CERNA; ALLISTENE, pp.51, hal-01724307.

CHEN H., LUNDBERG S. & LEE S. I. (2019), "Explaining Models by Propagating Shapley Values of Local Components", arXiv preprint arXiv:1911.11888.

CHZHEN E., DENIS C., HEBIRI M., ONETO L. & PONTIL M. (2020), "Fair Regression via Plug-in Estimator and Recalibration With Statistical Guarantees", hal-02501190, https://hal.archives-ouvertes.fr/hal-02501190.

CRAVEN M. W. (1996), "Extracting comprehensible models from trained neural networks", University of Wisconsin-Madison Department of Computer Sciences.

DATTA A., SEN S. & ZICK Y. (2016), "Algorithmic transparency via quantitative input influence: Theory and experiments with learning systems", in 2016 IEEE symposium on security and privacy (SP), pp. 598-617, IEEE.

DOSHI-VELEZ F. & KIM B. (2017), "Towards a rigorous science of interpretable machine learning", arXiv preprint arXiv:1702.08608.

DWORK C., HARDT Y., PITASSI T., REINGOLD O. & ZEMEL R. (2012), "Fairness Through Awareness", *Proceedings* of the 3rd Innovations of Theoretical Computer Science, 214 -226.

DYLAN H., GIOE D. V. & GOODMAN M. S. (2018), "The CIA and the pursuit of Security", Middle East, 1948, 63.

ERION G., JANIZEK J. D., STURMFELS P., LUNDBERG S., LEE S.-I. (2019), "Learning explainable models using attribution priors", arXiv preprint arXiv:1906.10670.

FELDMAN M., FRIEDLER S. A., MOELLER J., SCHEIDEGGER C. & VENKATASUBRAMANIAN S. (2015), "Certifying and removing disparate impact", *Proceedings* of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 259-268.

FERMANIAN J.-D. & GUEGAN D. (2020), "Fair learning with bagging", Working paper.

FREITAS A. A. (2014), "Comprehensible classification models: A position paper", SIGKDD Explor. Newsl., 15(1), 1–10, ISSN 1931-0145.

GOODMAN B. & FLAXMAN S. (2017), "European Union regulations on algorithmic decision-making and a 'right to explanation'", *AI magazine,* 38(3), 50-57.

GUIDOTTI R., MONREALE A., RUGGIERI S., TURINI F., GIANNOTTI F. & PEDRESCHI D. (2018), "A survey of methods for explaining black box models", ACM computing surveys (CSUR), 51(5), 1-42.

HARDT M., PRICE E. & SREBRO N. (2016), "Equality of opportunity in supervised learning", Advances in neural information processing systems, 3315-3323.

KAMIRAN F. & CALDERS T. (2012), "Data preprocessing techniques for classification without discrimination", *Knowledge and Information Systems,* 33(1): 1-33.

KIM H. & MALDE K. (2020), "Proper measure for adversarial robustness", arXiv preprint arXiv:2005.02540.

KROLL J. A., BAROCAS S., FELTEN E. W., REIDENBERG J. R., ROBINSON D. G. & YU H. (2016), "Accountable algorithms", U. Pa. L. Rev., 165, 633.

KUSNER M. J., LOFTUS J., RUSSELL C. & SILVA R. (2017), "Counterfactual fairness", Advances in Neural Information Processing Systems, (NeurIPS 2017), 4066-4076.

LINKOV I., TRUMP B. D., POINSATTE-JONES K. & FLORIN M. V. (2018), "Governance strategies for a sustainable digital world", *Sustainability,* 10(2), 440.

LIPTON Z. C. (2018), "The mythos of model interpretability", *Queue,* 16(3), 31-57.

LOU Y., CARUANA R. & GEHRKE J. (2012), "Intelligible models for classification and regression", in *Proceedings* of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining (pp. 150-158).

LU Q., CUI Z., CHEN Y. & CHEN X. (2017), "Extracting optimal actionable plans from additive tree models", *Frontiers of Compututational Science,* 11(1): 1-15.

LUNDBERG S. M. & LEE S. I. (2017), "A unified approach to interpreting model predictions", in Advances in neural information processing systems, pp. 4765-4774.

LUNDBERG S. M., ERION G. G. & LEE S. I. (2018), "Consistent individualized feature attribution for tree ensembles". arXiv preprint arXiv:1802.03888.

MELIS D. A. & JAAKKOLA T. (2018), "Towards robust interpretability with self-explaining neural networks", in Advances in Neural Information Processing Systems, pp. 7775-7784.

MILLER T. (2019), "Explanation in artificial intelligence: Insights from the social sciences", *Artificial Intelligence,* 267, 1-38.

Organization for Economic Co-Operation and Development (OECD), (2016), "OECD Science, Technology, and Innovation Outlook 2016", OECD Publishing: Paris, France, Chapter 2, pp. 110–111.

PAPERNOT N. (2018), "A Marauder's Map of Security and Privacy in Machine Learning", arXiv preprint arXiv:1811.01134.

POULIN B., EISNER R., SZAFRON D., LU P., GREINER R., WISHART D. S. & ANVIK J. *et al.* (2006), "Visual explanation of evidence with additive classifiers", in *Proceedings* of the National Conference on Artificial Intelligence, Vol. 21, No. 2, p. 1822, Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press.

RIBEIRO M. T., SINGH S. & GUESTRIN C. (2016a), "Why should I trust you? Explaining the predictions of any classifier", in *Proceedings* of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining, pp. 1135-1144.

RIBEIRO M. T., SINGH S. & GUESTRIN C. (2016b), "Model-agnostic interpretability of machine learning", arXiv preprint arXiv:1606.05386.

SLACK D., HILGARD S., JIA E., SINGH S. & LAKKARAJU H. (2020), "Fooling lime and shap: Adversarial attacks on post hoc explanation methods", in *Proceedings* of the AAAI/ACM Conference on AI, Ethics, and Society, pp. 180-186.

ŠTRUMBELJ E. & KONONENKO I. (2014), "Explaining prediction models and individual predictions with feature contributions", *Knowledge and information systems,* 41(3), 647-665.

WADSWORTH C., VERA F. & PIECH C. (2018), "Achieving fairness through adversarial learning: an application to recidivism prediction", arXiv preprint arXiv:1807.00199.

WACHTER S., MITTELSTADT B. & RUSSELL C. (2017), "Counterfactual explanations without opening the black box: Automated decisions and the GDPR", *Harv. JL and Tech.,* 31, 841.

ZAFAR M. B., VALERA I., GOMEZ RODRIGUEZ M. & GUMMADI K. P. (2017), "Fairness beyond disparate treatment and disparate impact: Learning classification without disparate mistreatment", in *Proceedings* of the 26th international conference on world wide web, pp. 1171-1180, April.

ZEMEL R., WU Y., SWERSKY K., PITASSI T. & DWORK C. (2013), "Learning fair representations", International Conference on Machine Learning, 325-333.

ZLIOBAITE I. (2015), "On the relation between accuracy and fairness in binary classification", arXiv preprint arXiv:1505.05723.