International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 08 Issue: 10 | Oct 2021 www.irjet.net p-ISSN: 2395-0072

A Review on Advances in Machine Learning Interpretability

Kruthi N Raj

Final Year B.E. Student, Department of Computer Science and Engineering, Bangalore Institute of Technology, Bengaluru, India

---

Abstract - *Whilst AI systems provide astonishingly good and accurate results, it is difficult to comprehend why and how an algorithm arrived at its conclusions, giving rise to the black-box notion in the field. Interpretable Machine Learning is no longer a novelty; it is a need. As the adoption of AI and ML for decision support grows, so does the need to understand the internal logic and functioning of these complex black boxes that even experts cannot fully comprehend. New laws are being implemented in many sectors, necessitating understanding and trusting AI and ML systems, not to mention other key reasons such as being able to correctly transfer learning into a wider knowledge base or societies becoming algorithmic-driven, among many others. The rising popularity of AI has emphasised the black-box idea, leading in a significant surge in research towards explainable AI as well as the need to understand and trust the operation of these systems. It is an active research area in both industry and academia. This article aims to provide an overview of the developments in many areas and concepts related to 'Interpretable AI.'*

Key Words: **Machine Learning; Artificial Intelligence; Interpretability; Explainability; Black-Box; White-Box; XAI**

1. INTRODUCTION

The most basic definition of ‘interpretable' is that that can be understood, explained, and accounted for. Although there is no mathematical definition of Machine Learning (ML), Miller [1] provides a more non-mathematical definition: ‘Interpretability is the degree to which a human can understand the cause of a decision.' Another suitable description is provided by Kim et al.[2], who state that ‘interpretability is the degree to which a human can consistently predict the model's result.' Understanding what influences decisions, explaining the decisions made by an algorithm, determining the patterns/rules/features learnt by an algorithm, being critical of the outcomes, and investigating the unknown of an algorithm are all aspects of interpretability.

Interpretability is one of four major issues covered by trustworthy and ethical AI, the others being fairness, robustness, and security. In comparison to a model with poor interpretability, a model with high interpretability makes its decisions easier to understand. Machine Learning interpretability research, albeit being a minor part of overall ML research, is extremely significant and absolutely necessary. The increased use of ML in different areas has created a growing need for decision support systems to be explainable.

When analysing how a particular ML model works, it is typical to hear the phrase "and then some magic happens." This is known as the AI black box, in which algorithms generate predictions but the underlying cause is ambiguous and untraceable. The inherent complexity that bestows incredible prediction skills on machine learning algorithms also makes the results difficult to comprehend. Understanding and trusting models and their predictions is a fundamental requirement for any good science, as well as a number of highly regulated and crucial industries such as healthcare, law, banking, and insurance.

Some ML models work in a straightforward and easy-to-understand manner, and each prediction can be justified. These are referred to as white-box models. However, as a result of recent advances in ML and AI, models have become extremely complex, such as complex deep neural networks and ensembles of various models. These complicated models are known as black-box models. Finding a balance between interpretability and accuracy due to the distinction between black-box and white-box models is a significant challenge for data scientists and other business leaders. The extraordinary predictive abilities of black box models stem from their complex structure, which also makes them difficult to understand and trust. The algorithms contained within the black box models are difficult to trace and interpret and they do not provide a clear explanation for why a particular prediction was made. Due to their opacity and difficulty in interpretation, these models may only provide a probability as a possible explanation for their decisions. There is no one-to-one relationship between input features and model parameters, and often combinations of multiple models with many parameters affect prediction. They require massive amounts of data to achieve high accuracy. It is difficult to determine what they learned from those data sets and which data points have a greater influence on the outcome than others.

Due to various factors, understanding the process and the results of these approaches is extremely challenging. It's also challenging to determine if we can rely on the models and make sound judgments when we use them. What if they learn the models incorrect information? What if they aren't ready fro decision making in the real word ? Misrepresentation, oversimplification, and overfitting are all the possibilities of such systems. As a result, analysing and comprehending these models has to be given top priority. In this article, I reviewed recent advancements in many areas and concepts relevant to AI interpretability.

2. BLACK-BOX (OPAQUE) AI

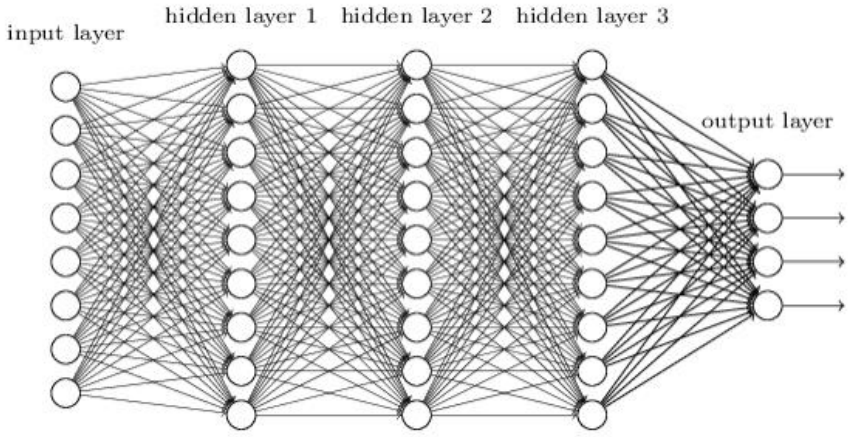

Figure 1: A fully connected neural network

Rudin categorises black-box AI systems into two types: complicated functions that are difficult for humans to understand and proprietary functionalities [4].

One of the simplest neural networks is seen in Figure 1 as a small, fully connected neural network. However, understanding many essential aspects of its operation, such as the function each neuron plays or the input features that contribute to the model output, is difficult. They are a network of interconnected variables with several layers. As the network grows in size, it becomes hard to track how the millions of parameters result in a decision. A neural network's judgement cannot be precisely deconstructed. As a result, this model, as well as any other model with comparable non-interpretability, is referred to as a black-box model. Support vector machines and ensemble approaches like boosting and random forest are further examples. Models of this type are included in the first group of Rudin's categorisation [9]. These models often include complicated transformations, several predictors, and parameters, making it difficult to visualise and comprehend the underlying workings. Despite their lack of transparency, these models are quite accurate.

The proprietary algorithms are the second type. These are the systems that businesses conceal for a variety of reasons, such as intellectual property protection or to avoid system hacking. The AI engineer of these systems may be familiar with their inner workings, but the general public who uses these systems is not. Google Search's ranking algorithm, Amazon's recommendation system, and Facebook's Newsfeed are all examples of such systems. However, when utilised in crucial areas such as healthcare and law, such as systems used to give down prison sentences, determine credit ratings, or make treatment choices in hospitals, make these sort of systems more risky.

Despite their superior performance, black-box models have numerous drawbacks. The first disadvantage is the previously mentioned lack of explainability, both internally inside an organisation utilising the system and externally to consumers and regulators demanding explanations for why a decision was made (e.g. black-box algorithm that erroneously cut medical coverage to long-time patients). The second disadvantage of black-box models is that there may be a few hidden issues affecting the output such as overfit, false correlations, or "garbage in / garbage out", that are hard to detect owing to a lack of knowledge of the black-box model's processes. Another disadvantage of not devoting enough time to comprehending reality outside the black-box model is that it produces a "comprehension debt" that must be repaid over time through difficulties in maintaining performance, unanticipated impacts such as individuals manipulating the system, or potential injustice. Finally, black-box models can accumulate technical debt over time, requiring the model to be evaluated and retrained more frequently as data drifts, as the model may rely on spurious and non-causal connections that soon evaporate, ultimately pushing up OPEX costs.

3. NEED FOR INTERPRETABLE AI

Different groups of individuals demand interpretability for a variety of reasons, including:

1. Data scientists who want to create high-accuracy models. They want to understand every aspect of their use case in order to discover the best model for their problem and ways to enhance that model. They also attempt to extract insights from the model in order to explain their results to their intended audience.

2. End users who would like to know why a model makes a certain prediction; interpretability is a way to fulfil human curiosity and learning [25]. Furthermore, they want to ensure that the forecasts are fair and that they are not adversely affected by the predicted decisions, which is only feasible if they can determine "why" a model made certain predictions. They want to be able to put their confidence in these models.

3. Regulators and policymakers who strive to make systems fair, transparent, and unbiased in order to safeguard associated rights and protect consumers. Also, their concerns are growing with the inevitable growth in the adoption of ML and AI into our life.

The reasons for needing interpretability may differ for various groups of people, but they all seek common features of AI systems: fairness, explainability, robustness, and security, which are the essential foundations of trustworthy and ethical AI.

Instances Proving the Importance of AI Interpretability

Over the years, various reported cases related to the problems in decision making of ML systems have only led us to the problem being these systems are not interpretable.Although ML is already supporting high-stakes decisions, lack of transparency and accountability for the decisions made makes these systems prone to errors in various situations and already have had severe consequences in different domains showing to us the importance Interpretability holds. A few such instances include cases of people incorrectly denied parole [73], incorrect bail decisions leading to the release of potentially dangerous criminals, pollution models stating that dangerous situations are safe [74], and more incidents in other domains, such as healthcare and finance [75]. There is also evidence that incorrect modeling assumptions were, at least, partially responsible for the mortgage crisis [72]. The worst part of such situations is that wronged people remain with little recourse to argue, and most of the entities behind these decision support systems cannot explicitly determine how these decisions were made due to the lack of transparency of the same systems. When AI was used in the criminal law realm, lots of AI tended to make harsher judgements of black people compared to white people. Having an explainable AI would have made it much clearer that the AI was being driven by the race of a person.

Benefits of AI Interpretability

Interpretable AI can help make recommendation algorithms from large tech businesses more transparent, resulting in greater understanding which can help customers identify potentially hazardous rabbit holes [10]. If done right, explainable AI can provide us with peace of mind. Interpretability is the key to satisfying

human curiosity curiosity[25]. When opaque machine learning models that simply make predictions without providing explanations are employed in research scientific discoveries are entirely buried[10]. Interpretability also helps to search for meaning in the world [25]. The decisions made by ML systems have a significant impact on people's lives, emphasising the need of understanding the process that led to the conclusion, especially when expectations differ from forecasts. For example, if a loan application is denied by an ML system, the applicant will almost definitely want to know at least the major reasons for the rejection (or what needs to be changed).

Interpretability also contributes to the development of trust and social acceptance, both of which are required for the broad use of AI and ML machines and algorithms.People ascribe beliefs and intentions to abstract things, according to Heider and Simmel [27]. As a result, it stands to reason that humans will be more willing to trust ML models if their judgments are interpretable. This is also supported by Ribeiro et al. [28], who claim that "if users do not trust a model or a prediction, they will not use it." Thus, interpretability contributes to increased human confidence and adoption of ML systems. The explanations might help manage social relationships. The explainer impacts the recipient's behaviours, emotions, and beliefs through generating a common meaning of something. To be more precise, in order for a machine to properly connect with humans, it may need to change people's emotions and ideas through persuasion in order for them to achieve their intended objective [10].

Safety is provided through interpretability [25]. ML models are verified, audited, and debugged through interpretability, which enhances their safety, especially in areas where mistakes might have significant consequences. To avoid self-driving cars colliding with cyclists, for example, an interpretation of the decision-making process can show us the features used by systems for bicycle recognition and for example, if it is shown to the two wheels that lead to the identification of the bicycles, this explanation helps to think about edge circumstances, such as when wheels are covered by side bags [10]. Through debugging and auditing, interpretability also allows for the discovery of incorrect model behaviour. the source of the inaccuracy can be identified using an explanation of the incorrect prediction and hence guides us through repairing the system and, as a result, enhance its safety. For example, in a husky vs. wolf image classifier, interpretability allows us to determine the cause of a misclassification, the reason being the model learnt to detect wolves using snow as a feature [10].

Explaining an ML model's judgments, according to Doshi-Velez et al. [7], gives a means to evaluate the desiderata of ML systems, such as fairness, privacy, and trust. Interpretability permits, for example, the discovery of bias that ML models learnt from data or as a result of incorrect parameterization caused by the issue definition's incompleteness. For example, a bank's ML model's primary aim is to give loans only to those who would eventually repay them; nevertheless, the bank must not only decrease loan defaults but also refrain from discriminating on the basis of specific demographics [10]. Interpretability also allows models to becomes the source of knowledge. The model's interpretability enables the extraction of this additional knowledge from data [10].

4. WHEN IS INTERPRETABILITY REQUIRED, AND WHEN IS IT NOT?

Doshi-Velez and Kim [7] provide an answer to the question, "Why don't we just trust an ML model and ignore why it made a specific decision when it's giving acceptable predicted performance?" "The problem is that a single metric, such as classification accuracy, is an incomplete description of most real-world tasks," they write. However, it is crucial to remember that certain circumstances do not entail high-stakes decision impact and may not necessitate explaining predictions. Although the relevance and necessity of interpretability have been explicitly emphasised, ML models employed in such circumstances that do not involve high-stake decision impact are simply required to offer highly accurate prediction performance. Only knowing what a system's predictions are is necessary in the aforementioned scenarios. However, in the majority of cases involving high-stake decision impacts, simply knowing what the system's predictions are is insufficient since it only partially addresses the problem, and it is critical to understand why the system made its predictions. There is a rising trend in the deployment of ML systems to help in making decisions that have a significant impact on human lives, society and in healthcare, financial services, and other highly regulated areas that have high-stake decision impact thus increasing the need for ML interpretability.

5. METHODS AND TECHNIQUES FOR INTERPRETABILITY OF ML MODELS

5.1. INTERPRETABILITY IN MODELING STAGES

1. **Interpretability in pre-modeling stage(interpretability of model inputs) -** Before selecting and developing models, it is critical thoroughly grasp the data using Exploratory Data Analysis (EDA) [29]. Various techniques ranging from classic descriptive statistics to data visualization methods, including Principal Component Analysis (PCA) [30] and t-SNE [31] (t-Distributed Stochastic Neighbor Embedding), and clustering methods, such as k-means [32] and MMD-critic [26] (Maximum Mean Discrepancy) can be utilised ti better comprehend and interpret the data. Having a knowledge of the data also allows us to select important and meaningful features, which allows us to describe the input-output connection after modelling.

2. **Interpretability in modeling stage -** The ML models are divided into two groups based on essential features such as simplicity, transparency, and explainability: black-box (opaque) models and white-box (transparent) models. The type of model selected, the number of features, and the depth of the model chosen determine interpretability at this stage. Interpretability in this stage is achieved by using inherently interpretable models.

3. Interpretability in post-modeling stage(post hoc interpretability) - Interpreting model predictions allows us to examine the interactions between input features and predicted outputs. In this phase, interpretability provides us with a deeper knowledge of many elements connected to features, such as the most essential characteristics for a model, how these features impact predictions, how each of them contributes to the prediction, and how sensitive the model is to specific features.

5.2. CONCERNS SURROUNDING INHERENTLY INTERPRETABLE AND POST HOC APPROACH

Inherently interpretable models are the ones that can explain themselves. The introduction of constraints on the model, such as sparsity, monotonicity, causality, or physical constraints derived from domain knowledge, might result in models that are inherently interpretable [4]. Intrinsic interpretability, also known as transparency, provides an answer to the issue of how the model operates [9]. Post-hoc (post-model) interpretability refers to explanation approaches that are used after the model has been trained. A majority of the research done on ML interpretability deals with developing tools to help explain the predictions and decisions of a model. Cynthia Rudin argues that this approach of post-hoc interpretability where ML system decisions are explained after they gave been done can have dangerous implications and the focus of research should shift from post-hoc interpretability towards breaking open the black-box i.e developing interpretable and transparent models. "Rather than trying to create models that are inherently interpretable, there has been a recent explosion of work on 'explainable ML', where a second (post hoc) model is created to explain the first black box model. This is problematic. Explanations are often not reliable," Rudin writes [4]. The post-hoc approaches can cause to significant harm to the society as well as to critical domains having high-stake decision impact such as healthcare and criminal justice [20]. According to Rudin, developers should choose models that are inherently interpretable and provide their own explanations and that, in contrary to what some AI researchers believe, interpretable ML models can produce results just as accurate results as the deep black-box models. Lipton [9] argues that post-hoc interpretability answers the question "what else can the model tell us?"

5.3. MODEL-SPECIFIC VS. MODEL-AGNOSTIC TECHNIQUES

A crucial component to our knowledge of interpretability is the understanding of interpretability techniques' dependency on models.

Model-specific techniques - Model-specific interpretation techniques are confined to a particular algorithm type or class since they are dependent on the internals of a certain model [10]. For example, the interpretation of weights in a linear model is a model-specific interpretation, similar to the tree interpreter approach, which can only be used to decision trees or the variable significance output from Random Forest. Models are interpreted intrinsically using model-specific interpretation techniques and are potentially more accurate.[66]

Model-agnostic techniques - On the other hand, model-agnostic techniques are post hoc in nature. They are applied to any ML model (black-box or not) after they are trained. The inner workings of any model such as the weights or structural information can't be accessed by these techniques [10]. Instead, They analyse input features and output predictions. Although these methods are applied after training thus allow for interpreting models without sacrificing the predictive performance [9], they rely on surrogate models and other approximations that may lead to degradation in the accuracy of the provided explanations. This technique includes Partial Dependence (PD) plots, Individual Conditional Expectation (ICE) plots and Local Interpretable Model-Agnostic Explanations (LIME).

Pre-model	–	–
In-model	Intrinsic	Model-specific
Post-model	Post hoc	Model-agnostic

Table 1 : Association between interpretability methods and techniques

6. TYPES OF EXPLANATION FROM ML INTERPRETABILITY TECHNIQUES

The results of interpretability approaches, i.e. the type of explanation produced, can be used to differentiate them [70]. Although there are many more techniques to provide explanation, such as rule sets, question-answering, or natural language explanations, the types listed below account for great majority of existing types of explanations from ML interpretability techniques.

1. **Feature summary -** Some explanation techniques include summary statistics for each feature. For example, they may be a single numbers for each feature indicating it's significance . Majority of the time these statistics are visualised, with a

few of them making little sense without it, such as partial dependency plots, which are not intuitive when provided in tabular style.

2. **Model internals -** his explanation type is suited to models that are intrinsically interpretable. Some interpretability approaches include model internals as well as summary statistics, such as linear model weights. The interpretability techniques that output only the model internals are model-specific techniques.

3. **Data point -** To achieve interpretability, these approaches generate data points. These example-based approaches necessitate that the output data points themselves be meaningful and interpretable. This approach works well for photos and texts but is less helpful for tabular data with hundreds of features, for example.

4. **Surrogate intrinsically interpretable model -** Another approach to understanding black-box models (globally or locally) using an intrinsically interpretable model is approximation. As a result, the interpretation of the surrogate model will provide information about the original model.

7. SCOPE OF INTERPRETABILITY

Each step of an algorithm training a prediction model can be evaluated in terms of transparency or interpretability. Understanding the entire model trained on a global scale and being able to look closely into the local parts of the data and predictions are equally important trading to global and local interpretability. From the algorithm level, interpretability levels are as stated below [67] -

1. **Algorithm transparency -**

Algorithm transparency is concerned with how the algorithm works, how it builds a model from data, and what kinds of associations it can learn from it. It has nothing to do with the final model learnt or how individual predictions are generated. Method transparency necessitates just knowledge of the algorithm, not of the data or learnt model, because it answers the question "how does the algorithm build the model?" [10]. The ordinary least squares technique is one such example. The emphasis of ML interpretability, on the other hand, is on the models themselves, rather than the algorithms that generate them.

2. **Global model interpretability -**

On a holistic level - A model is interpretable, according to Lipton [9], if the whole model can be understood at once. A trained model, an understanding of the algorithm, and data are necessary to explain the global model output. This level of interpretability deal with understanding how a models decisions are made based on a comprehensive view of the data characteristics and each of the learnt components, such as weights and parameters. In other words, global model interpretability refers to the ability to comprehend the distribution of prediction output depending on characteristics, therefore addressing the issue of how the trained model makes predictions. As a result, it is extremely difficult to achieve in practise [10]. For this reason, it is very difficult to achieve in practice [10]. Any model with more than 5 parameters or weights is unlikely to fit into the average human's short-term memory, especially given that humans can only process about 7 cognitive entities at once [34]. Furthermore, the fact that individuals can only see three dimensions at a time makes this sort of interpretability even more difficult to achieve. Honegger [19] contends that a model must be basic enough to satisfy this requirement.

On a modular level - While global, holistic model interpretability is typically out of reach, at least some models can be understood on a modular basis. The interpretable portions of linear models, for example, are the weights; for decision trees, the interpretable parts are the splits (features and cut-off values) and leaf node predictions. This provides an answer to the question, "How do different elements of the model influence predictions?" [10]. As a result, models with highly engineered, anonymous, or opaque characteristics do not meet this requirement [19]. It is also worth mentioning that interpreting a single weight in a linear model, for example, is interlocked with all other weights, implying that this form of interpretability often does not account for, for example, feature interaction. According to Lipton [9], this concept is known as decomposability, and it means that the inputs used to train a model must be interpretable itself. This is referred to as intelligibility by Lou et al. [106], who argue that a Generative Additive Model (GAM) meets it.

3. **Local Model Interpretability -**

For single prediction - In order to explain a single prediction, the basic approach is to focus on a single occurrence and try to understand how the model arrived at its forecast. This can be accomplished by approximating a tiny region of interest in

a black box model with a simpler interpretable model. This surrogate model, while not giving an optimum answer, is a pretty acceptable approximation that retains interpretability [21]. The logic for this is that, locally, the prediction may simply depend linearly or monotonously on some features rather than having a complicated dependency on them. This indicates that local explanations can be more correct than global explanations [10].

For a Group of Predictions - There are essentially two ways to explain a group of predictions: use global techniques and consider the group of predictions of interest as if it were the whole dataset, or perform local methods on each prediction individually, aggregating and combining these explanations afterwards [9,10].

8. INTERPRETABLE MODELS

Using algorithms capable of constructing interpretable models [10], such as linear' regression, logistic regression, and decision trees, is one of the simplest ways to achieve interpretability. These are global interpretable models on a modular level with meaningful features that use this information to explain predictions. The below mentioned features [10] can be used to select the interpretable model best suited for the situation among the several options available. Interpretable models has at least one of the following characteristics:

1. **Linearity -** Models with a linear relationship between feature and target values.

2. **Monotonicity -** Monotonicity is useful for interpretation because it makes the relationship between some features and the target easier to understand by ensuring that a specific input feature and the target outcome always go in the same direction across the entire feature domain, i.e., when the feature value increases, it always leads to an increase or always leads to a decrease in the target outcome.

3. **Interaction -** Interactions between features to predict the target outcome are naturally included in some ML models. Feature engineering can be used to manually create interaction features in any type of model. While too many or too complex interactions will decrease interpretability, interactions can improve predictive performance .

ALGORITHM	LINEAR	MONOTONE	INTERACTION	TASK
Linear Regression	Yes	Yes	No	Regression
Logistic Regression	No	Yes	No	Classification
Decision Trees	No	Some	Yes	Classification, Regression
RuleFit	Yes	No	Yes	Classification, Regression
Naive Bayes	No	Yes	No	Classification

Table 2: Interpretable models comparison

9. SCALE OF INTERPRETABILITY

High interpretability - linear , monotonic functions. Traditional regression algorithms that are “linear and monotonic” produce the most interpretable models. They allow for a change in a function's output at a specified rate in response to a change in any input variable, but only in one direction, i.e. either a reduction or an increase, and at a magnitude indicated by a predefined coefficient. Explanatory approaches like LIME make use of these features as well.

2. **Medium interpretability - nonlinear, monotonic functions.** The majority of ML models are nonlinear, with some being monotone in relation to any given independent variable. These “nonlinear, monotonic functions” include algorithms such as logistic regression, decision trees, and Naive Bayes. Changes in any input variable cause a one-way change in the function output, despite the fact that there is no single coefficient to reflect this change. Such functions create reason codes and relative variable significance measures that may be understood and used in regulated applications.

3. **Low interpretability - nonlinear, non monotonic functions.**The vast majority of ML algorithms generate “nonlinear and non monotonic functions,” which are notoriously difficult to interpret since the output might change in any direction

at any rate in response to any change in any input variable. Standard interpretability metrics give relative variable significance, thus many such spatial approaches must be coupled to properly interpret them.

Data scientists used monotonicity constraints in their models to address the black-box problem of their models, whereas algorithms generated nonlinear, non monotonic, non polynomial, and noncontinuous functions that approximated the connection between independent and dependent variables in a data set. Another option was to adopt a linear model, even if it meant sacrificing some accuracy. However, these models could not give sufficient levels of predictability and transparency, resulting in the development of a new model interpretability concept: white-box models.

10. INTERPRETABILITY EXPLANATION METHODS AND EVALUATION

10.1. MODEL-SPECIFIC

Post-hoc explanation techniques produce explanations by utilising intrinsic features of specific types of models. A significant disadvantage of adopting this sort of method is that it restricts model choice to specific model classes. Despite being typical black-box models, Deep Neural Networks are extensively utilised due to their high prediction ability. For DNNs, many model-specific techniques have been developed. There are numerous well-known examples of such DNN explanation methods [3], the majority of which are used in computer vision [60], such as guided backpropagation [61], integrated gradients [62], SmoothGrad saliency maps [63], Grad-CAM [64], and, more recently, testing with Concept Activation Vectors (TCAV) [65].

Knowledge distillation is a sort of post-hoc model-specific explanation approach that involves extracting knowledge from a complicated model and transferring it to a simpler model (which can be from a completely different class of models). This can be accomplished, for example, by model compression [43], tree regularisation [44], or even by combining model compression with dimension reduction [45]. This sort of technique research has been going on for a while [46], but it has lately expanded in tandem with the ML interpretability area [47–49].

Moreover, the increasing interest in model-specific explanation methods that focus on specific applications can be seen, for example, in the recent 2019 Conference on Computer Vision and Pattern Recognition (CVPR), which featured an workshop on explainable AI [8]. There are CVPR 2019 papers with a great focus on interpretability and explainability for computer vision, such as explainability methods for graph CNNs [50], interpretable and fine-grained visual explanations for CNNs [51], interpreting CNNs via Decision Trees [52], and learning to explain with complemental examples [53].

10.2. MODEL-AGNOSTIC

Although model-specific explanation techniques may be more beneficial in some situations, the model-agnostic approach has the benefit of being totally independent of the model, making these models completely reusable in totally new use cases where the prediction model is likewise different. These techniques are post-hoc, with some being example-based and returning existing or new data points. Table 3 summarises the present model-agnostic method, its breadth, and explanation type.

Explanation Method	Scope	Explanation Type
Partial Dependence Plot [142]	Global	Feature summary
Individual Condition Expectation [143]	Global/Local	Feature summary
Accumulated Local Effects Plot [144]	Global	Feature summary
Feature Interaction [145]	Global	Feature summary
Feature Importance [146]	Global/Local	Feature summary
Local Surrogate Model [98]	Local	Surrogate interpretable model
Shapley Values [147]	Local	Feature summary
BreakDown [148]	Local	Feature summary
Anchors [149]	Local	Feature summary
Counterfactual Explanations [87]	Local	Data point (new)
Prototypes and Criticisms [96]	Global	Data point (existing)
Influence Functions [150]	Global/Local	Data point (existing)

Table 3: Comparison of model-agnostic explanation methods

10.3. CHOOSING THE "RIGHT" EXPLAINABILITY METHOD

When selecting the best model-agnostic explanation technique, three factors must be considered:

1. Requirement for whole-model logic interpretation or just reasoning for particular decisions. A global or local approach is chosen based on this.

2. Time constraint. If a quick judgement must be made (for example, evaluating the response of a natural disaster management system), a brief and easy-to-understand explanation is preferable.

3. The level of skill of the user. Depending on their job, users of the prediction model may have varying levels of domain expertise and task experience (they could be decision-makers, scientists, engineers, etc.). While domain specialists may appreciate a comprehensive, complex explanation, others may prefer a simple explanation.

10.4. INTERPRETABILITY EVALUATION METHODS

When it comes to assessing interpretability, there is a lot of uncertainty, with no consensus on the best approach to utilise [10, 59]. Existing research, on the other hand, has focused on developing techniques for assessing interpretability. Doshi-Velez and Kim [7] offer three primary techniques for assessing interpretability:

1. **Application-grounded evaluation (end task) -** End-user assessment is required. The experiments are carried out in the context of a real-world application, and they are tested and assessed by an end-user who is also a domain expert. A reasonable starting point for this is how well a human would explain the identical decision [10]. For example, the best approach to assess an interpretation on identifying a certain ailment is for the doctor to do diagnostics.

2. **Human-grounded evaluation (simple task) -** Refers to carrying out smaller human–subject investigations that keep the essence of the goal application intact. These trials do not necessitate domain specialists, but rather laypeople. Experiments are less expensive, and it is easier to recruit testers because domain knowledge is not necessary. Humans, for example, are given with two explanations and must select the one that they believe is of greater quality.

3. **Functionally grounded evaluation (proxy task) -** There is no need for human experiments in this sort of assessment. Instead, some formal notion of interpretability, such as the depth of a decision tree, serves as a proxy for evaluating explanation quality. Model sparsity or uncertainty might also be used as proxies [70]. This works best when the class of model being utilised has already been reviewed in a human-level evaluation by someone else. This approach is far less expensive than the previous two, but the problem is determining which proxies to employ.

The most appropriate technique is application-grounded evaluation, which analyses interpretability in the end aim with end users. However, this technique is expensive, makes it difficult to compare findings across domains, and necessitates the use of domain-specific testers, who are difficult to locate. On the other side, functionally grounded evaluation does not require any human experimentation, and the stated proxies for this evaluation are generally equivalent across domains. However, the results of functionally grounded evaluation have limited validity since the proxies that may be established are not genuine indicators of interpretability and there is no human input. Human-grounded evaluation is an intermediate solution that has a lower cost than application-grounded evaluation but a higher validity than functionally grounded evaluation—the results are based on human feedback but do not take into account the domain in which the assessed interpretability would be applied.

10.5. GOALS OF INTERPRETABILITY

According to Rüping et al. [21], interpretability is comprised of three interconnected and frequently competing goals:

1. **Accuracy** - Refers to the actual relationship between the explanation method's supplied explanation and the forecast from the ML model [59]. If this goal is not met, the explanation is rendered worthless since it is not true to the prediction it seeks to explain.

2. **Understandability -** Is concerned with the ease with which an explanation is grasped by the spectator. This objective is critical because, no matter how precise an explanation is, it is meaningless if it is not understood [59].

3. **Efficiency -** Reflects the amount of time required for a user to comprehend the explanation. Without this requirement, it is obvious that nearly every model, given an unlimited length of time, is interpretable [59]. As a result, an explanation should be understood in a limited, ideally brief period of time.

This means that high interpretability would be defined as an explanation that is accurate to the data and the model, intelligible by the typical observer, and graspable in a short period of time [29]. Nonetheless, as Rüping et al. [21] point out, there is generally a trade-off between these goals; for example, the more precise an explanation is, the less comprehensible it becomes.

11. EXPLAINABILITY VS INTERPRETABILITY

Although the terms 'explainable AI' and 'interpretable AI' are sometimes used interchangeably, there is a significant distinction between the two. Interpretable AI are algorithms that create models whose decision-making process can be interpreted. It is possible to track the path that the input data takes as it goes through the model. For example, decision trees that assign coefficients to each of their input data's characteristics. Figure 2 illustrates a trace of an input data's journey through the decision tree.

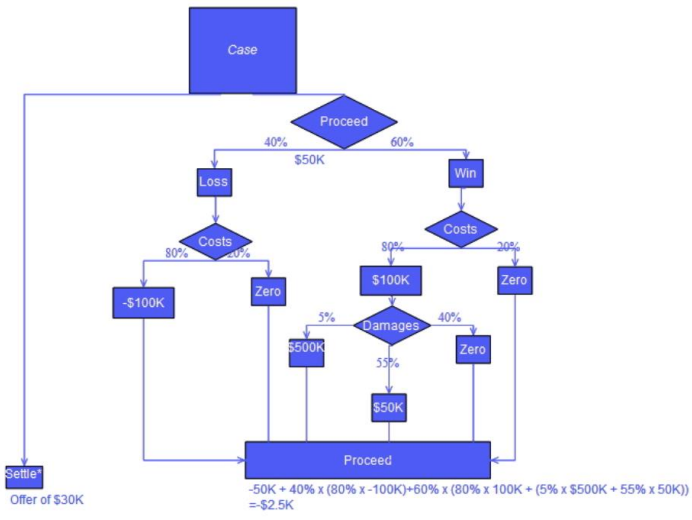

Figure 2: A clear explanation of a decision tree's result

Explainable AI, on the other hand, refers to the tools needed to comprehend the logic and operation of algorithms whose operation cannot be understood. Deep learning–based image classifier judgments, for example, are interpreted using models that generate saliency maps that emphasise the pixels in the input picture that contributed to the output. However, these explanation models do not dissect the models' internal logic. In her study [4], Rudin says, "Explanation here refers to a knowledge of how a model works, as opposed to an explanation of how the world operates."

12. PROBLEMS WITH AI EXPLAINABILITY

Without delving into the core logic, explainability techniques often illustrate how a change in the model's input impacts the input. Like in case of an image classifier, a minor modification to the input pixel values is performed to see how it impacts the categorisation. Based on these observations, heat maps displaying the features most relevant to the AI system are presented. According to Rudin [4], explainability approaches do not always give insights into how the black-box AI model operates."Explanation models do not always attempt to mimic the calculations made by the original model. Rather than producing explanations that are faithful to the original model, they show trends in how predictions are related to the features," Rudin writes.

She discusses a research of a black-box recidivism AI system that identified software that was racially biassed. While a linear model based on race was used to describe the AI's activities, the recidivism system under consideration was a sophisticated, nonlinear AI system. This study provided light on the need of transparency in AI systems that make critical decisions, but it did not provide a detailed description of how the system worked. There might have been many more problematic connections in the system that the research overlooked. Saliency maps for computer vision systems are also an evidence to the flaws of AI explanation approach. The majority of these approaches will only show which areas of an image prompted an image classifier to predict a specific label but does not give enough information in how the AI system processed the data to arrive at that label. For example, in Figure 3, the saliency maps for the "Siberian husky" and "transverse flute" are strangely similar. Although the classifier is clearly focused on the correct region of the husky image, there is no sign that it is identifying the correct characteristics.

Poor explanation techniques, according to Rudin, make troubleshooting a black-box difficult. Rudin observes that explainability approaches not only do not address the problem of analysing the too sophisticated black-box AI, but rather worsen it by providing us two systems to troubleshoot: the original AI model and the explainability tool.

Figure 3: Saliency map explanation that does not provide accurate explanation of the inner logic of black-box model [4]

13. THE AI ACCURACY-INTERPRETABILITY TRADEOFF

Kuhn and Johnson [71] stated that "Unfortunately, the predictive models that are most powerful are usually the least interpretable". Data scientists are suspected of frequently putting more focus on the forecast accuracy of their models than on comprehending the inner logic and process within the model that led to the prediction. A common misconception in the AI world is that there is a compromise between accuracy and interpretability. Deep neural networks, for example, are extensively employed for a variety of use cases because they give extremely high accuracy and flexibility in compression compared to other ML models, but this comes at the expense of interpretability.

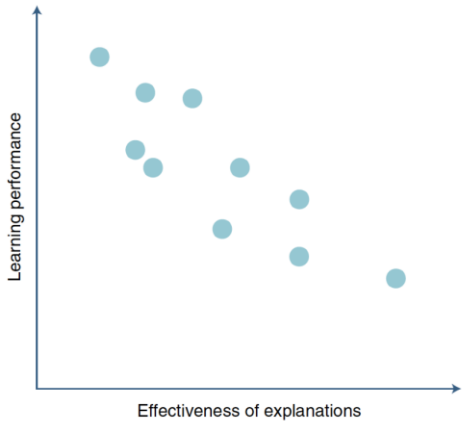

Figure 4: A graph of performance vs. explainability of ML models

Data scientists are said to place a greater emphasis on It is often assumed that data scientists prefer complicated models to obtain higher accuracy, which dramatically reduces interpretability. However, the tradeoff between accuracy and interpretability, according to Rudin, relies on the domain, the type of data available, and the intended outcomes. After preprocessing, there is frequently no substantial difference in performance between more complicated classifiers and more simpler classifiers when evaluating issues with structured data with relevant characteristics. She also asserts that in some situations, the interpretability given by simple models is more useful than the minimal accuracy gained by complicated black-box models. "In those cases, the accuracy/interpretability trade-off is reversed—more interpretability leads to better overall accuracy, not worse," she writes. This is often the case in crucial fields such as medicine. Data scientists are using the "bigger is better" attitude. They choose to train deep learning models with a large number of layers on huge datasets while achieving interpretability through the use of interpretable AI approaches in certain areas. "The belief that there is always a trade-off between accuracy and interpretability has led many researchers to forgo the attempt to produce an interpretable model. This problem is compounded by the fact that researchers are now trained in deep learning, but not in interpretable ML," Rudin writes.

14. MERGING WHITE-BOX INTERPRETABILITY AND BLACK-BOX ACCURACY

Although black-box models have visible input-output correlations, their underlying workings remain incomprehensible. White-box models, on the other hand, are interpretable, allowing users to understand the inner workings, but give lesser accuracy when compared to black-box models. Every model should ideally be explainable, clear, and extremely accurate. In the real world, there is a need for both accuracy and explainability in key fields such as healthcare and law, where comprehending the process that led to a judgement is critical. Although this is a complicated and difficult undertaking, one approach is to build systems with the interpretability of white-box models and performance similar to, if not better than, white-box models. Grey-box models are created by combining black-box and white-box models [68]. This method seeks to develop an ensemble of whole-box and black-box models, incorporating the benefits of each and resulting in a more efficient model. Grey-box models are a combination of ML black-box models such as neural networks and white-box models such as logistic regression. The following are some instances of grey-box approaches found in the literature:

1. Grau et al. [70] proposed a grey-box ensemble using a self-labeled approach for semi-supervised classification problems. guide the comparatively data expensive white-box component with the results from the more accurate black-box part. The proposed technique showed good performance on partially labelled datasets.

2. LIME [15] can be considered a Grey-Box post-hoc interpretability model which aims to explain the predictions of a Black-Box model by training a White-Box on a local generated dataset.

3. Interpretable mimic learning [19] is a Grey-Box intrinsic interpretation method which trains an accurate and complex Black-Box model and transfers its knowledge to a simpler interpretable White-Box model in order to acquire the benefits of both models for developing an accurate and interpretable Grey-Box model. In other words, the target values of the White-Box model, are the output predictions of the Black-Box model. The idea was borrowed from knowledge distillation [20] and mimic learning [21]. In this approach, the Black-Box eliminates possible noise and errors in the original training data which could lead to the reduction of the accuracy performance for a White-Box model. This Grey-Box model can have improved performance, comparing to a single White-Box model trained on the original data, being at the same time

interpretable since the output predictor is a White-Box model and thus the model falls in the intrinsic interpretation category. The obvious disadvantage of this method is the same with all intrinsic interpretation models and this is the lower accuracy comparing to post-hoc models, since the performance of these models is bounded by the performance of their output predictor which is a White-Box model.

4. Pintelas et al. presented an ensemble black box model, which employed a Black-Box model to supplement a small initial labelled dataset with the algorithm's most confident predictions from a large unlabelled dataset, and then used this enhanced dataset as a training set for a White-Box model. They combined the classification accuracy of the black-box with the interpretability of the white-box to create an ensemble that was both accurate and interpretable. The trained white-box model was used as the final predictor, as opposed to the self-training framework. This ensemble model has inherent interpretability since its output predictor is a White-Box, which is interpretable by definition.

15. PYTHON LIBRARIES TO INTERPRET ML MODELS

1. **ELI5 -** ELI5 is an abbreviation that stands for "Explain like I'm a 5-year-old." This well titled Python package can explain the majority of machine learning models. There are two approaches to interpreting a machine learning model: 1) Global Interpretation: Examine a model's parameters to determine how the model operates on a global level. 2) Local Interpretation: Examine a particular prediction and discover the elements that contribute to that prediction. One of the features of the ELI5 library is that it already supports popular libraries like as scikit-learn, XGBoost, Keras, and others. ELI5 may also be used for text data! It includes a TextExplainer module for explaining text categorisation models. Another standout feature is the formatter module, which allows us to generate HTML, JSON, or even Pandas data-frame versions of our explanation. This makes it simple to incorporate the explanation into our machine learning pipeline.

2. **LIME -** The creators of LIME state that trust is the most important factor to consider when developing a machine learning model. The ultimate aim is to make judgments based on these predictions, which is where people join in. The goal of LIME (Local Interpretable Model-Agnostic Explanations) is to explain why a prediction was made. Using the same example, if a machine learning model predicts that a film will be a blockbuster, LIME highlights the aspects of the film that will make it a smash success. Features such as genre and actor may help the film go well, while others such as running time, director, and so on may work against it. LIME's developers describe four key requirements for explanations that must be met: 1. **Understandable**: The explanation must be understandable to the target population. 2. **Local fidelity**: The explanation must explain how the model operates for specific predictions. 3. **Model-independent**: The approach should be capable of explaining any model. 4. **Global perspective**: When discussing the model, the model as a whole should be taken into account.These are good criteria to employ, and we can apply them to the interpretability of machine learning models in general.

3. **SHAP -** The Python library 'SHapley Additive exPlanations,' often known as the SHAP library, is one of the most used libraries for machine learning interpretability. The SHAP library, which is targeted at explaining individual predictions, is built on Shapley values. Shapley values are taken from Game Theory, in which each characteristic in our data is a player, with the prediction as the final prize. Shapley values teach us how to divide a prize equitably among the participants based on the award. SHAP's biggest feature is that it has a specific module for tree-based models. Given the popularity of tree-based models in hackathons and industry, this module performs rapid calculations, even when dependent features are present.

Yellowbrick - Yellowbrick is built on the scikit-learn and matplotlib libraries. This makes it compatible with the vast majority of scikit-learn models. We may even utilise the same parameters from our machine learning models (based on scikit-learn, of course). Yellowbrick makes use of the notion of 'Visualisers.' Visualizers are a set of tools that allow us to show the characteristics in our data while taking individual datapoints into account. Consider it a dashboard for all of your features. Yellowbrick's primary Visualisers are as follows: 1. **Rank Features**: To see specific features and their relationships to other features, rank them. 2. **RadViz Visualizer**: To display the separability of classes. 3. **Parallel Coordinates**: To see the target class's distribution in relation to other characteristics. 4. **PCA Projection**: Using Principal Component Analysis to see combined/higher dimensions (PCA) 5. **Manifold Visualization**: Using manifold learning to visualise data (like t-SNE) 6. **Direct Data Visualization/Joint Plot Visualizer**: To depict the link between separate characteristics and the target variable.

4. **ALIBI -** Alibi is a Python open-source package that uses instance-wise explanations of predictions to make predictions (instance, in this case, means individual data-points). Depending on the sort of data we're dealing with, this library contains a variety of explainers. The library provides several sorts of Explainer models based on various methods. The library is intended solely for black-box models. To utilise the library, you essentially need the model predictions in the end.

This is very handy when we don't want to mess with our machine learning model's process. Alibi includes a slew of additional helpful libraries as part of its dependencies, including scikit-learn, Pandas, spaCy, TensorFlow, and many more. This makes it very valuable for deep learning models.

5. **LUCID -** As deep learning becomes more prevalent, the necessity to explain these deep learning models becomes critical. However, given the enormous number of characteristics we must deal with, this can be extremely difficult. By offering tools for visualising neural networks, the Lucid library seeks to address this vacuum. It also enables the visualisation of neural networks without any prior preparation. Modelzoo is a component that comes preloaded with a variety of deep learning models. It allows to experiment with many settings, such as the effect of utilising modified data, displaying certain channels, and so on. However, there are a few of caveats: 1. At the moment, they only support TensorFlow 1.0. 2. Despite the fact that the notebooks extensively cover each example, there is no comprehensive documentation source like the libraries mentioned above.

16. CONCLUSION

The interpretability issue is domain-specific [21,22], and there is no universal definition [4]. When it comes to ML interpretability, the application domain and problem use case are critical in determining its meaning. Every day, new methods for making ML models interpretable are developed. With the application domain, problem use case, time constraints, and target audience in mind, the interpretability approach should be chosen. Comparing different explanation strategies among the numerous available, with new ones being introduced often, is challenging. More research is needed to develop metrics to properly compare explanation approaches. More study on establishing measures to better compare explanation techniques is required. There has also been a lot of study towards building post-hoc models. However, these models do not fully serve the process and can be harmful, necessitating more research into building inherently interpretable models. Grey-box models should be viewed as a possible option in crucial domains where a tradeoff between accuracy and interpretability is not practicable, and additional study on this idea is required. As of present, model-agnostic techniques provide the best explanation to back-box models among the existing ways since they take into account the issue domain, use case, and user type.

REFERENCES

1. Miller, Tim. “Explanation in artificial intelligence: Insights from the social sciences.” arXiv Preprint arXiv:1706.07269. (2017).

2. Kim, Been, Rajiv Khanna, and Oluwasanmi O. Koyejo. “Examples are not enough, learn to criticize! Criticism for interpretability.” Advances in Neural Information Processing Systems (2016).

3. Montavon, G.; Lapuschkin, S.; Binder, A.; Samek, W.; Müller, K.R. Explaining nonlinear classification decisions with deep Taylor decomposition. Pattern Recognit. 2017, 65, 211–222.

4. Rudin, C. Please Stop Explaining Black Box Models for High Stakes Decisions. arXiv 2018, arXiv:1811.10154.

5. Van Melle, W.; Shortliffe, E.H.; Buchanan, B.G. EMYCIN: A knowledge engineer’s tool for constructing rule-based expert systems. In Rule-Based Expert Systems: The MYCIN Experiments of the Stanford Heuristic Programming Project; Addison-Wesley Reading: Boston, MA, USA, 1984; pp. 302–313.

6. Fahner, G. Developing Transparent Credit Risk Scorecards More Effectively: An Explainable Artificial Intelligence Approach. Data Anal. 2018, 2018, 17.

7. Doshi-Velez, F.; Kim, B. Towards a rigorous science of interpretable machine learning. arXiv 2017, arXiv:1702.08608.

8. CVPR. CVPR 19—Workshop on Explainable AI. Available online: https://explainai.net/ (accessed on 12 July 2019).

9. Lipton, Z.C. The mythos of model interpretability. arXiv 2016, arXiv:1606.03490.

10. Molnar, C. Interpretable Machine Learning. 2019. Available online: https://christophm.github.io/ interpretable-ml-book/ (accessed on 22 January 2019).

11. Temizer, S.; Kochenderfer, M.; Kaelbling, L.; Lozano-Pérez, T.; Kuchar, J. Collision avoidance for unmanned aircraft using Markov decision processes. In Proceedings of the AIAA Guidance, Navigation, and Control Conference, Toronto, ON, Canada, 2–5 August 2010; p. 8040.

12. Wexler, R. When a computer program keeps you in jail: How computers are harming criminal justice. New York Times, 13 June 2017.

13. McGough, M. How Bad Is Sacramento's Air, Exactly? Google Results Appear at Odds with Reality, Some Say. 2018. Available online: https://www.sacbee.com/news/state/california/fires/article216227775.html (accessed on 18 January 2019).

14. Varshney, K.R.; Alemzadeh, H. On the safety of machine learning: Cyber-physical systems, decision sciences, and data products. Big Data 2017, 5, 246–255.

15. Donnelly, C.; Embrechts, P. The devil is in the tails: Actuarial mathematics and the subprime mortgage crisis. ASTIN Bull. J. IAA 2010, 40, 1–33.

16. Angwin, J.; Larson, J.; Mattu, S.; Kirchner, L. Machine Bias. 2016. Available online: https://www.propublica.org/article/machine-bias-risk-assessments-in-criminal-sentencing (accessed on 18 January 2019).

17. Tan, S.; Caruana, R.; Hooker, G.; Lou, Y. Detecting bias in black-box models using transparent model distillation. arXiv 2017, arXiv:1710.06169.

18. Doshi-Velez, F.; Kortz, M.; Budish, R.; Bavitz, C.; Gershman, S.; O'Brien, D.; Schieber, S.; Waldo, J.; Weinberger, D.; Wood, A. Accountability of AI under the law: The role of explanation. arXiv 2017, arXiv:1711.01134.

19. Honegger, M. Shedding Light on Black Box Machine Learning Algorithms: Development of an Axiomatic Framework to Assess the Quality of Methods that Explain Individual Predictions. arXiv 2018, arXiv:1808.05054.

20. O'Neil, C. Weapons of Math Destruction: How Big Data Increases Inequality and Threatens Democracy; Broadway Books: Portland, OR, USA, 2017.

21. Rüping, S. Learning Interpretable Models. Ph.D. Thesis, University of Dortmund, Dortmund, Germany, 2006.

22. Freitas, A.A. Comprehensible classification models: a position paper. ACM SIGKDD Explor. Newslett. 2014, 15, 1–10.

23. Case, N. How To Become A Centaur. J. Design Sci. 2018.

24. Varshney, K.R.; Khanduri, P.; Sharma, P.; Zhang, S.; Varshney, P.K. Why Interpretability in Machine Learning? An Answer Using Distributed Detection and Data Fusion Theory. arXiv 2018, arXiv:1806.09710.

25. Miller, T. Explanation in Artificial Intelligence: Insights from the social sciences. Artif. Intell. 2018, 267, 1–38.

26. Kim, B.; Khanna, R.; Koyejo, O.O. Examples are not enough, learn to criticize! Criticism for interpretability. In Advances in Neural Information Processing Systems; MIT Press: Cambridge, MA, USA, 2016; pp. 2280–2288.

27. Heider, F.; Simmel, M. An experimental study of apparent behavior. Am. J. Psychol. 1944, 57, 243–259.

28. Ribeiro, M.T.; Singh, S.; Guestrin, C. "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, San Francisco, CA, USA, 13–17 August 2016; pp. 1135–1144.

29. Tukey,J.W. Exploratory Data Analysis; Pearson:London,UK,1977;Volume2.

30. Olliffe,I.Principalcomponentanalysis.InInternationalEncyclopediaofStatisticalScience;Springer:Berlin,Germany, 2011; pp. 1094–1096.

31. Maaten,L.v.d.;Hinton,G. Visualizing data using t-SNE. J.Mach.Learn.Res.2008,9,2579–2605.

32. Hartigan, J.A.; Wong, M.A. Algorithm AS 136: A k-means clustering algorithm. J. R. Stat. Soc. Ser. C (Appl. Stat.) 1979, 28, 100–108.

33. Google People + AI Research (PAIR). Facets—Visualization for ML Datasets. Available online: https://pair-code.github.io/facets/ (accessed on 12 July 2019).

34. Cowan, N. The magical mystery four: How is working memory capacity limited, and why? Curr. Dir. Psychol. Sci. 2010, 19, 51–57.

35. Lou, Y.; Caruana, R.; Gehrke, J. Intelligible models for classification and regression. In Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Beijing, China, 12–16 August 2012; pp. 150–158.

36. Lakkaraju,H.;Bach,S.H.;Leskovec,J. Interpretable decision sets: A joint frame work for description and prediction. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, 13–17 August 2016; pp. 1675–1684.

37. Rudzin´ski,F. A multi-objective genetic optimization of interpretability-oriented fuzzy rule-based classifiers. Appl. Soft Comput. 2016, 38, 118–133.

38. Angelino, E.; Larus-Stone, N.; Alabi, D.; Seltzer, M.; Rudin, C. Learning certifiably optimal rule lists. In Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Halifax, NS, Canada, 13–17 August 2017; pp. 35–44.

39. Dash, S.; Günlük, O.; Wei, D. Boolean Decision Rules via Column Generation. arXiv 2018, arXiv:1805.09901.

40. Yang, H.; Rudin, C.; Seltzer, M. Scalable Bayesian rule lists. In Proceedings of the 34th International Conference on Machine Learning, Sydney, Australia, 6–11 August 2017; Volume 70, pp. 3921–3930.

41. Rudin, C.; Ustun, B. Optimized Scoring Systems: Toward Trust in Machine Learning for Healthcare and Criminal Justice. Interfaces 2018, 48, 449–466.

42. Wang,T.;Rudin,C.;Doshi-Velez,F.;Liu,Y.;Klampfl,E.;MacNeille,P. A bayesian framework for learning rule sets for interpretable classification. J. Mach. Learn. Res. 2017, 18, 2357–2393.

43. Polino, A.; Pascanu, R.; Alistarh, D. Model compression via distillation and quantization. arXiv 2018, arXiv:1802.05668.

44. Wu,M.;Hughes,M.C.;Parbhoo,S.;Zazzi,M.;Roth,V.;Doshi-Velez,F. Beyond sparsity: Tree regularization of deep models for interpretability. In Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, New Orleans, LA, USA, 2–7 February 2018.

45. Xu,K.;Park,D.H.;Yi,C.;Sutton,C. Interpreting Deep Classifier by Visual Distillation of Dark Knowledge. arXiv 2018, arXiv:1803.04042.

46. Hinton, G.; Vinyals, O.; Dean, J. Distilling the knowledge in a neural network. arXiv 2015, arXiv:1503.02531.

47. Murdoch, W.J.; Szlam, A. Automatic rule extraction from long short term memory networks. arXiv 2017, arXiv:1702.02540.

48. Frosst, N.; Hinton, G. Distilling a neural network into a soft decision tree. arXiv 2017, arXiv:1711.09784.

49. Bastani, O.; Kim, C.; Bastani, H. Interpreting blackbox models via model extraction. arXiv 2017, arXiv:1705.08504.

50. Pope, P.E.; Kolouri, S.; Rostami, M.; Martin, C.E.; Hoffmann, H. Explainability Methods for Graph Convolutional Neural Networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 16–20 June 2019.

51. Wagner,J.;Kohler,J.M.;Gindele,T.;Hetzel,L.;Wiedemer,J.T.;Behnke,S. Interpretable and Fine-Grained Visual Explanations for Convolutional Neural Networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 16–20 June 2019.

52. Zhang, Q.; Yang, Y.; Ma, H.; Wu, Y.N. Interpreting CNNs via Decision Trees. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 16–20 June 2019.

53. Kanehira, A.; Harada, T. Learning to Explain With Complemental Examples. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 16–20 June 2019.

54. Kim,B.;Rudin,C.;Shah,J.A.The bayesian case model: A generative approach for case-based reasoning and prototype classification. In Advances in Neural Information Processing Systems; MIT Press: Cambridge, MA, USA, 2014; pp. 1952–1960.

55. Ross, A.; Lage, I.; Doshi-Velez, F. The neural lasso: Local linear sparsity for interpretable explanations. In Proceedings of the Workshop on Transparent and Interpretable Machine Learning in Safety Critical Environments, 31st Conference on Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017.

56. Lage, I.; Ross, A.S.; Kim, B.; Gershman, S.J.; Doshi-Velez, F. Human-in-the-Loop Interpretability Prior. arXiv 2018, arXiv:1805.11571.

57. Lee,M.;He,X.;Yih,W.t.;Gao,J.;Deng,L.;Smolensky,P. Reasoning in vector space:An exploratory study of question answering. arXiv 2015, arXiv:1511.06426.

58. Palangi, H.; Smolensky, P.; He, X.; Deng, L. Question-answering with grammatically-interpretable representations. In Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, New Orleans, LA, USA, 2–7 February 2018.

59. Bibal, A.; Frénay, B. Interpretability of machine learning models and representations: An introduction. In Proceedings of the 24th European Symposium on Artificial Neural Networks ESANN, Bruges, Belgium, 27–29 April 2016; pp. 77–82.

60. Kindermans,P.J.;Schütt,K.T.;Alber,M.;Müller,K.R.; Erhan, D.;Kim,B.; Dähne,S. Learning how to explain neural networks: PatternNet and PatternAttribution. arXiv 2017, arXiv:1705.05598.

61. Springenberg,J.T.;Dosovitskiy,A.;Brox,T.;Riedmiller,M. Striving for simplicity: The all convolutional net. arXiv 2014, arXiv:1412.6806.

62. Sundararajan, M.; Taly, A.; Yan, Q. Axiomatic attribution for deep networks. arXiv 2017, arXiv:1703.01365.

63. Smilkov,D.;Thorat,N.;Kim,B.;Viégas,F.;Wattenberg,M. Smoothgrad:removing noise by adding noise. arXiv 2017, arXiv:1706.03825.

64. Selvaraju,R.R.;Cogswell,M.;Das,A.;Vedantam,R.;Parikh,D.;Batra,D.Grad-cam: Visual explanations from deep networks via gradient-based localization. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; pp. 618–626.

65. Kim,B.;Wattenberg,M.;Gilmer,J.;Cai,C.;Wexler,J.;Viegas,F.;Sayres,R. Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (tcav). In Proceedings of the International Conference on Machine Learning, Stockholm, Sweden, 10–15 July 2018; pp. 2673–2682.

66. Hall, Patrick and Gill, Navdeep. An Introduction to Machine Learning Interpretability. O'Reilly Media, 2018.

67. Molnar, Christoph. "Interpretable machine learning. A Guide for Making Black Box Models Explainable", 2019.

68. Bohlin, T.P. Practical Grey-Box Process Identification: Theory and Applications; Springer Science & Business Media: Berlin/Heidelberg, Germany, 2006.

69. Emmanuel Pintelas et al. "Algorithms | Free Full-Text | A Grey-Box Ensemble Model Exploiting Black-Box Accuracy And White-Box Intrinsic Interpretability | HTML." MDPI, Www.mdpi.com, 5 January, 2020.

70. Grau, I.; Sengupta, D.; Garcia, M.; Nowe, A. Grey-Box Model: An ensemble approach for addressing semi-supervised classification problems. In Proceedings of the 25th Belgian-Dutch Conference on Machine Learning BENELEARN, Kortrijk, Belgium, 12–13 September 2016.

71. Kuhn, M.; Johnson, K. Applied Predictive Modeling; Springer: New York, NY, USA, 2013; Volume 26.

72. Ribeiro, M.T.; Singh, S.; Guestrin, C. 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, San Francisco, CA, USA, 13–17 August 2016; ACM: New York, NY, USA, 2016; pp. 1135–1144.

73. Grau, I.; Sengupta, D.; Garcia, M.; Nowe, A. Grey-Box Model: An ensemble approach for addressing semi-supervised classification problems. In Proceedings of the 25th Belgian-Dutch Conference on Machine Learning BENELEARN, Kortrijk, Belgium, 12–13 September 2016.

74. Bohlin, T.P. Practical Grey-Box Process Identification: Theory and Applications; Springer Science & Business Media: Berlin/Heidelberg, Germany, 2006. Molnar, C. Interpretable Machine Learning: A Guide for Making Black Box Models Explainable (eBook); Lulu: Morrisville, NC, USA, 2019.