Received: 17 February 2022 | Revised: 9 February 2023 | Accepted: 10 February 2023

DOI: 10.1002/widm.1493

OVERVIEW

WIREs DATA MINING AND KNOWLEDGE DISCOVERY WILEY

Interpretable and explainable machine learning: A methods-centric overview with concrete examples

Ričards Marcinkevičs | **Julia E. Vogt**

Department of Computer Science, ETH Zurich, Zurich, Switzerland

Correspondence
Ričards Marcinkevičs, Department of Computer Science, ETH Zurich, Zurich, Switzerland.
Email: ricards.marcinkevics@inf.ethz.ch

Funding information
Schweizerischer Nationalfonds zur Förderung der Wissenschaftlichen Forschung, Grant/Award Number: 320038189096

Edited by: Mehmed Kantardzic, Associate Editor and Witold Pedrycz, Editor-in-Chief

Abstract

Interpretability and explainability are crucial for machine learning (ML) and statistical applications in medicine, economics, law, and natural sciences and form an essential principle for ML model design and development. Although interpretability and explainability have escaped a precise and universal definition, many models and techniques motivated by these properties have been developed over the last 30 years, with the focus currently shifting toward deep learning. We will consider concrete examples of state-of-the-art, including specially tailored rule-based, sparse, and additive classification models, interpretable representation learning, and methods for explaining black-box models post hoc. The discussion will emphasize the need for and relevance of interpretability and explainability, the divide between them, and the inductive biases behind the presented "zoo" of interpretable models and explanation methods.

This article is categorized under:

Fundamental Concepts of Data and Knowledge > Explainable AI
Technologies > Machine Learning
Commercial, Legal, and Ethical Issues > Social Considerations

KEYWORDS

explainability, interpretability, machine learning, neural networks

1 | INTRODUCTION: INTERPRETABILITY, EXPLAINABILITY, AND INTELLIGIBILITY

Interpretable and explainable machine learning (ML) techniques emerge from a need to design *intelligible* machine learning systems, that is, ones that can be comprehended by a human mind, and to understand and explain predictions made by *opaque* models, such as deep neural networks (Goodfellow et al., 2016) or gradient boosting machines (Friedman, 2001; Mason et al., 1999). The early research on interpretable machine learning dates back to the 1990s (Rudin, 2019). It often does not refer to terms like "*interpretability*" or "*explainability*," not to mention that many classical statistical models can be deemed interpretable.

In general, there is no agreement within the ML community on the definition of *interpretability* and the *task of interpretation* (Doshi-Velez & Kim, 2017; Lipton, 2018). For example, Doshi-Velez and Kim (2017) define interpretability of ML systems as "*the ability to explain or to present in understandable terms to a human*." This definition lacks mathematical rigor (Lipton, 2018). Nevertheless, the notion of interpretability often depends on the domain of application (Rudin, 2019) and the target *explainee* (Carvalho et al., 2019), that is, the recipient of interpretations and explanations. Therefore, an all-purpose definition might be infeasible (Rudin, 2019) or unnecessary. Other terms that are synonymous with interpretability and also appear in the ML literature are "*intelligibility*" (Caruana et al., 2015; Lou et al., 2012) and "*understandability*" (Lipton, 2018). These concepts are often used interchangeably.

Yet another term prevalent in the literature is "*explainability*," giving rise to the direction of *explainable artificial intelligence* (XAI) (Gunning & Aha, 2019). This concept is closely tied with interpretability; and many authors do not differentiate between the two (Carvalho et al., 2019). Doshi-Velez and Kim (2017) provide a definition of *explanation* that originates from psychology: "*explanations are ... the currency in which we exchange beliefs*." Rudin (2019) draws a clear line between interpretable and explainable ML: *interpretable ML* focuses on designing models that are *inherently interpretable*, whereas *explainable ML* tries to provide post hoc explanations for existing *black-box* models, that is, models that are incomprehensible to humans or are proprietary (Rudin, 2019). Lipton (2018) stresses the difference in questions the two families of techniques try to address: interpretability raises the question "*How does the model work?*," whereas explanation methods try to answer "*What else can the model tell me?*"

1.1 | Purpose of the review

This review is intended for a general machine learning audience interested in exploring the problems of interpretation and explanation beyond the logistic regression model or random forest variable importance. It is not an exhaustive literature survey but rather an overview with a selection of *concrete*, comprehensively studied examples that represent different research directions. We will address the following questions throughout this review:

1. What is the difference between interpretable and explainable ML?
2. In what settings is it desirable for an ML model to be interpretable or to be explained?
3. How can the interpretability and explainability be assessed in practice?
4. What inductive biases are characteristic of interpretable models and explanation methods?

The material presented in this overview is partially based on the literature review from the article by Marcinkevics and Vogt (2021), although the current work covers a much broader range of topics and has been updated with more recent references.

1.2 | Related work and our contribution

To date, interpretable and explainable machine learning form an established subfield with its own research questions and directions. There exist numerous thorough review papers tackling the topic. Many reviews can be categorized into four groups briefly summarized below. (i) Some provide a relatively nontechnical and general introduction to the fundamental problems, concepts, and research questions and directions, for example, see works by Carvalho et al. (2019), Barredo Arrieta et al. (2020), or Molnar (2020). (ii) Others view interpretability and explainability from a novel or unusual perspective or provide opinions on the progress, challenges, and future directions. For instance, Ghassemi et al. (2021) overview healthcare applications of explainability techniques and their failure cases and argue that XAI is unlikely to address the real needs of practitioners. (iii) Another category of reviews focuses on a restricted class of models or a family of methods. For example, Verma et al. (2020) discuss only counterfactual explanation methods, and Puiutta and Veith (2020) specifically survey reinforcement learning techniques. (iv) Last but not least, some reviews discuss the use of interpretable and explainable ML in a particular application area, for example, genomics (Watson, 2021) or robotics (Anjomshoae et al., 2019). Table 1 lists a nonexhaustive manual selection of the review articles from the four categories mentioned above.

In contrast, while starting with a broad introduction to the topic and basic concepts, this review explores interpretability and explainability via concrete, comprehensively studied examples of the latest models, methods, and their

TABLE 1 Overview of manually collected review papers on interpretable and explainable ML or related topics.

High-level concepts	Perspectives	Restricted methods scope	Applications
Barredo Arrieta et al. (2020); Molnar (2020); Linardatos et al. (2020); Guidotti et al. (2019); Murdoch et al. (2019); Carvalho et al. (2019); Du et al. (2019); Adadi and Berrada (2018)	Ghassemi et al. (2021); Rudin et al. (2022); Confalonieri et al. (2021); Emmert-Streib et al. (2020); Roscher et al. (2020); Byrne (2019); Miller (2019); Holzinger et al. (2019)	Verma et al. (2020); Burkart and Huber (2021); Puiutta and Veith (2020); Moraffah et al. (2020); Seeliger et al. (2019); Gilpin et al. (2018); Chakraborty et al. (2017); Otte (2013)	Watson (2021); Tjoa and Guan (2021); Zhang and Chen (2020); Stiglic et al. (2020); Azodi et al. (2020); Anjomshoae et al. (2019)

Note: Some of these reviews focus on high-level concepts and provide a relatively nontechnical and brief introduction to a wide range of research questions and directions. Other papers approach the topic from a novel perspective or express an opinionated assessment of the field's progress and challenges. Some review articles focus exclusively on specific model classes and method families, whereas others discuss interpretability and explainability in the context of a particular application domain. Note that some articles might fit into multiple categories.

typical inductive biases. We provide an intuitive explanation for many techniques but do not shy away from examining equations and definitions behind them hands-on. At the same time, we attempt to give the reader a well-rounded overview of the various lines of methodological work. The models and methods discussed later were chosen as representative of the current state of the field.

1.3 | Organization of the paper

In the remainder of this review, we discuss a need for interpretable and explainable machine learning techniques, giving examples from several application domains (Section 2). We provide an overview of the evaluation methods for interpretability and explainability (Section 3). We then outline a taxonomy of the techniques for interpretable (Section 4.2) and explainable (Section 4.3) ML with concrete examples of several recent developments. Finally, Section 5 contains concluding remarks.

2 | MOTIVATION AND RELEVANCE

It is natural to question the utility of interpretable and explainable ML, especially given a widespread belief that a trade-off exists between accuracy and interpretability (Rudin, 2019; Semenova et al., 2019). Therefore, it is sensible to ask "*Why would a designer of an ML system consider sacrificing performance for the sake of transparency?*" First, it is important to note that there are many cases when interpretability is not necessary, particularly when the studied problem is well-known, well-understood, and does not have substantial consequences (Doshi-Velez & Kim, 2017), for example, mail sorting, movie recommendation, and so forth. Second, the perceived accuracy–interpretability trade-off may not necessarily apply to all datasets and prediction problems (Rudin, 2019).

Arguably, the commonest motivation behind interpretability and explainability is developing user trust (Doshi-Velez & Kim, 2017; Lipton, 2018). Lipton (2018) decomposes trust into knowing "*how often a model is right*" and "*for which examples it is right*." Sometimes we might want to gain a more profound intuition about the model's behavior. In that case, an ability to interpret or explain could be another prerequisite for a trustable ML system. However, this ability alone is not sufficient (Rudin et al., 2022), since it is not a substitute for accurate and reliable predictions.

In practice, interpretability and explainability are typically most useful when auditing ML systems and confirming auxiliary desiderata beyond predictive performance (Carvalho et al., 2019; Doshi-Velez & Kim, 2017; Lipton, 2018). From a legal perspective, interpretable and explainable ML is concordant with the EU General Data Protection Regulation (GDPR) (Voigt & von dem Bussche, 2017) that states data subjects' *right to an explanation* of algorithmic decisions and the *right to be informed*. It is worth mentioning that the GDPR does not prohibit black-box predictive models and that the right to an explanation is not legally binding (Carvalho et al., 2019; Wachter et al., 2017). This, however, as Wachter et al. (2017) note, does not undermine the social and ethical value of providing interpretations and

TABLE 2 A few interpretable and explainable machine learning use cases with concrete application examples.

Use case	Reference	Model/method	Description
User trust	Ustun and Rudin (2015)	Supersparse linear integer models (Section 4.2.2)	Linear models with integral coefficients are introduced to learn data-driven medical risk scores amenable to clinicians
Causality	Fujii et al. (2021)	Self-explaining neural networks (Section 4.2.8)	An interpretable neural network model is introduced to discover causal structure in time series data representing animal trajectories
Scientific discovery	Udrescu and Tegmark (2020)	Symbolic regression (Section 4.2.10)	A method is introduced to extract succinct mathematical expressions explaining physics-based data
Debugging	Caruana et al. (2015)	Generalized additive models (Section 4.2.3)	Generalized additive models are used to predict pneumonia risk and readmission and mitigate spurious correlations
Fairness	Larson et al. (2016)	Surrogate models (Section 4.3.1)	Surrogate models are used to identify racial bias in the proprietary COMPAS software for recidivism prediction

explanations. Below we discuss several goals attainable with interpretability and explainability that are commonly cited in the literature. Table 2 shows a few concrete examples of the considered use cases.

One could leverage an interpretable model or an explanation method to generate hypotheses about causal relationships among the observed variables in the data (Lipton, 2018). In those cases, it is often desirable for the model or explanation to pick up cause–effect relationships (Carvalho et al., 2019) rather than spurious associations. Such formulation of interpretability is ambitious and inherently requires solving the problem of observational causal discovery (Nogueira et al., 2022). Some authors even go further and suggest that genuinely interpretable machine learning should provide *causal* interpretations and explanations of the data (G. Xu et al., 2020).

On a related note, interpretability and explainability can be instrumental in exploratory data analysis and scientific discovery (Doshi-Velez & Kim, 2017). For example, interpretable support vector machines have been used for discovering unknown physics in materials science (K. Liu et al., 2021). In quantum chemistry, neural networks allowed for an analytical differentiable representation of the quantum mechanical wavefunction (Sch✓tt et al., 2019). In computational linguistics, Pimentel et al. (2019) have leveraged NLP models alongside an information-theoretic approach to quantify the relationship between word forms and meanings. These are just a few examples of emerging machine-learning-assisted scientific discovery. A comprehensive survey by Raghu and Schmidt (2020) contains many more scientific deep learning applications.

"Good" ML models should be resistant to noisy inputs and domain shifts. Interpretations and explanations can be instrumental in designing reliable, robust, and transferable models (Carvalho et al., 2019; Doshi-Velez & Kim, 2017; Lipton, 2018). For instance, an iconic example wherein interpretability facilitated model "debugging" in that regard is discussed by Caruana et al. (2015), who have used generalized additive models for pneumonia risk prediction, exhibited, and alleviated unwanted confounding in the dataset. Another noteworthy example is the *Manifold*—an in-house visualization and debugging tool for ML models developed at Uber (Carvalho et al., 2019; L. Li & Wang, 2019).

When ML algorithms are incorporated into decision-making, for example, social, economic, or medical, and use sensitive personal data, we have to scrutinize their fairness (Barocas et al., 2019; Dignum, 2019) and privacy (Papernot et al., 2018). Interpretations and explanations can be instrumental in exposing demographic disparities and reliance on sensitive information in ML models (Carvalho et al., 2019; Doshi-Velez & Kim, 2017; Lipton, 2018) by making them readily auditable. For example, using explanation methods, the ProPublica analysis of the Correctional Offender Management Profiling for Alternative Sanctions (COMPAS) recidivism model (Larson et al., 2016; Rudin, 2019) has revealed that the COMPAS might be racially biased.

In summary, interpretability and explainability, although not necessary in many straightforward applications, become instrumental when the problem definition is incomplete and in the presence of additional desiderata, such as trust, causality, or fairness. These principles can be helpful to both the specialists designing predictive models and end-users who want to obtain a more profound intuition about the behavior of an ML system. In practice, there exists a plethora of techniques, ranging from specially tailored interpretable neural network architectures to out-of-the-box model-agnostic explanation methods. According to Bhatt et al. (2020), who have conducted interviews with 50 data scientists and practitioners from 30 different organizations, the choice of an interpretable model or an explanation technique for a specific use case should depend on the identified stakeholders' needs and expectations regarding interpretability and explainability.

3 | EVALUATION OF INTERPRETABILITY AND EXPLAINABILITY

Despite the abundance of methodological research, literature on evaluation approaches and metrics for interpretable and explainable ML is still relatively scarce (Carvalho et al., 2019). There appear to be no uniform, well-established standards for qualitative or quantitative evaluation, likely due to the lack of an all-purpose definition of interpretable and explainable ML and the diversity and subjectivity of the desiderata and principles investigated in the literature. Nauta et al. (2022) provide the most comprehensive survey to date of qualitative and quantitative methods. This section outlines one popular classification of the evaluation criteria, due to Doshi-Velez and Kim (2017), that is concordant with much of the current literature. Examples of how these evaluation methods could be implemented in practice are provided in Table 3.

3.1 | Application-grounded evaluation

Application-grounded evaluation requires evaluating a method or a model on an exact task with human experts representing the target audience. For example, the best way of evaluating an explainable ML-based decision support system for medical diagnosis would be to ask doctors to diagnose diseases assisted by the system and compare their performance to a reasonable baseline. Similar evaluation methods are widely adopted, for example, in the field of human–computer interaction (MacDonald & Atwood, 2013) and, arguably, if implemented correctly, provide the strongest evidence of success. A study by Jesus et al. (2021) is an excellent example of application-grounded evaluation: the authors evaluate several explanation methods for fraud detection based on transaction data. They measure the accuracy and time of decisions by fraud analysts assisted by different explanations and compare versus the decisions based purely on the raw data and black-box model predictions.

TABLE 3 A taxonomy of evaluation approaches for interpretable and explainable machine learning due to Doshi-Velez and Kim (2017).

Evaluation approach	Requires a user study?	Cost	Specificity	Example
Application-grounded	Yes	High	High	Jesus et al. (2021) evaluate explanation methods for the real-world task of fraud detection by measuring the accuracy and time of decisions by fraud analysts assisted by explanations
Human-grounded	Yes	Medium	Medium	Ribeiro et al. (2016) evaluate the proposed explanation method based on the ability of subjects recruited at *Amazon Mechanical Turk* to choose the best text classification model
Functionally-grounded	No	Low	Low	Shrikumar et al. (2017) measure the decrease in image classification accuracy after masking the features identified as important by the proposed explanation method

Note: Approaches are characterized in terms of their cost, specificity with respect to the end-task and user, and the requirement of user studies. The last column contains examples from the recent literature.

3.2 | Human-grounded evaluation

Human-grounded evaluation can be viewed as a relaxed version of the application-grounded evaluation. It requires conducting experiments with human users performing, possibly, a simplified task reminiscent of the target application. For instance, Ribeiro et al. (2016) evaluate the proposed explanation technique using the human-grounded approach. They recruit human subjects on *Amazon Mechanical Turk* (Paolacci et al., 2010) and compare their ability to choose the best text classification model based on explanations provided by the proposed method versus baseline techniques. Notably, the recruited subjects are not experts in the subject area of texts, and the task is merely a proxy for the end-goal of the ML system. Needless to say, although human-grounded evaluation is cheaper than the application-grounded approach, its results inevitably lead to less specific and insightful conclusions.

3.3 | Functionally-grounded evaluation

Last but not least, functionally-grounded evaluation is, arguably, most appropriate for early feasibility studies and is the simplest to implement since it requires no human subject experiments. These methods use some formal mathematical definition of interpretability or explainability as a proxy measure. For example, Shrikumar et al. (2017) perform an experiment evaluating different explanation methods for image classification based on the decrease in the classification accuracy on the MNIST dataset (LeCun et al., 2010) after masking features identified as important by an explanation method. Another example is the dataset of *Kandinsky Patterns* and accompanying challenges introduced by M✓ller and Holzinger (2021): in brief, challenges comprise classifying simple visual patterns in controllable synthetic image datasets while producing explanations in a specific format, for example, natural language. Similarly, by extending CLEVR dataset (Johnson et al., 2017) for visual question answering, Arras et al. (2020) release the CLEVR-XAI benchmark for neural network explanation methods. While such evaluation approaches are compelling and can be implemented entirely in silico, their insights are often limited by the subjectivity of the proxy measure chosen and the simplicity of the toy datasets used.

The evaluation of interpretability and explainability in ML models largely remains an open problem. Interpretable and explainable ML research still often relies on anecdotal or subjective evidence; for instance, Nauta et al. (2022) observe that only 58% of the papers surveyed by them evaluate their models and methods quantitatively, and mere 22% conduct a user study. Performing large-scale experiments with human subjects, identifying and systematizing good proxy metrics, developing rigorous criteria and desiderata for evaluation are all essential for the advancement of the whole field.

4 | INTERPRETABLE MODELS AND EXPLANATION METHODS

Now that we have established that interpretability and explainability of ML models are essential in certain settings and how these properties are evaluated, the reader might be left wondering how interpretability of a model is achieved in practice or how the predictions of a black-box model could be explained? The following sections discuss several state-of-the-art interpretable and explainable ML methods. The selection of works does not comprise an exhaustive survey of the literature. Instead, it is meant to illustrate the commonest properties and inductive biases behind interpretable models and explanation methods using concrete instances.

Figure 1 provides a roadmap for the remainder of this section, compiling some of the most salient characteristics of interpretable and explainable ML identified in the previous literature (Carvalho et al., 2019; Doshi-Velez & Kim, 2017; Lipton, 2018; Molnar, 2020). These include the model class, scale at which interpretations or explanations are produced, agnosticism with respect to the black-box model, and actionability. Tables 5 and 8 outline the properties of the concrete techniques further. These properties will be defined and discussed in detail throughout the section.

4.1 | Notation and preliminaries

This review primarily focuses on interpretability and explainability in the context of supervised learning for classification and regression tasks. However, some sections will discuss unsupervised learning scenarios, such as unsupervised representation learning (Section 4.2.11). Table 4 introduces mathematical notation used throughout the section unless

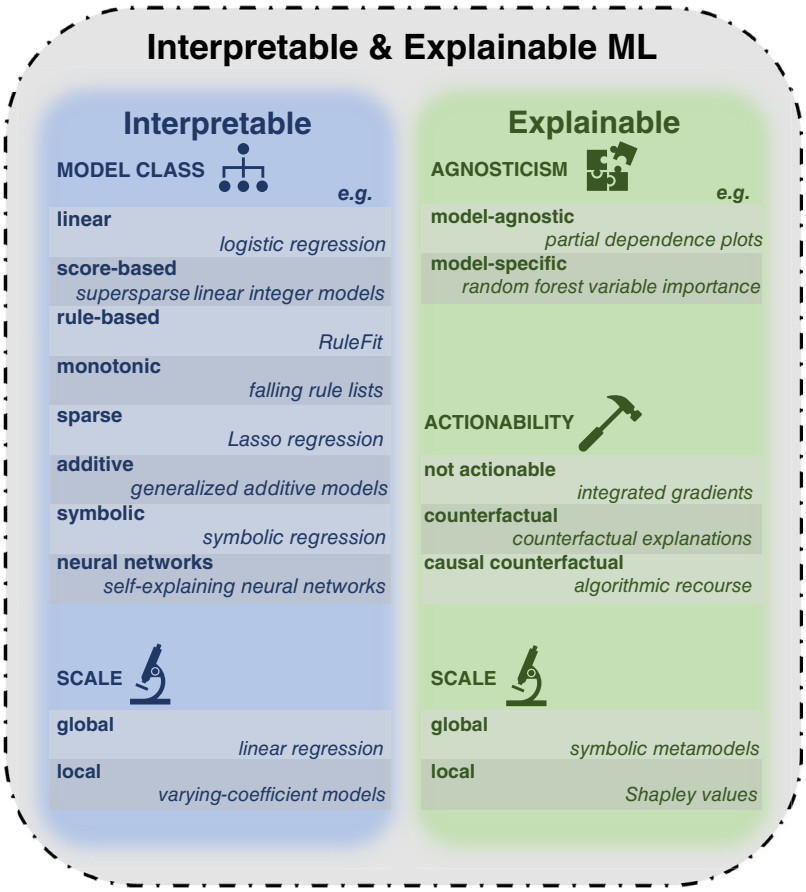


FIGURE 1 Roadmap for the review of interpretable models and explanation methods based on a compilation of salient characteristics identified in the literature (Carvalho et al., 2019; Doshi-Velez & Kim, 2017; Lipton, 2018; Molnar, 2020). Concrete examples for each property are shown in italic.

TABLE 4 Mathematical notation used throughout the remainder of this review.

Symbol	Explanation
p	Number of features in tabular data
N	Number of data points in the training set
$\boldsymbol{x}_i$	Feature vector of the i-th data point
y_i	Label of the i-th data point
x_j	j-th feature
$x_{i,j}$	j-th feature of the i-th data point
$f(\cdot)$	A model
$\boldsymbol{\beta}$	Coefficient vector in a linear model
β_j	Coefficient of the j-th feature
$\boldsymbol{\theta}$	Model parameters
$\boldsymbol{W}$	Weight matrix
$\boldsymbol{W}_{i,:}$	i-th row of a matrix
$\boldsymbol{W}_{:,j}$	j-th column of a matrix
$\mathcal{L}(\cdot)$	Loss function

TABLE 5 Properties of several reviewed interpretable models.

Model	Scale	Linear	Sparse	Additive	Monotonic	Unstructured data
F. Wang and Rudin (2015b): Falling rule lists	•		✓		✓	
Ustun and Rudin (2015, 2017): Supersparse linear integer models	•	✓	✓	✓		
Hastie and Tibshirani (1986): Generalized additive models (GAMs)	•			✓		
Caruana et al. (2015): GAMs plus interactions	•			∼		
Sparse additive models						
Ravikumar et al. (2007): Sparse additive models	•		✓	✓		
Feng and Simon (2017): Sparse-input NNs	•		✓			✓
DeepPINK						
Lu et al. (2018): DeepPINK	•		✓			✓
Hastie and Tibshirani (1993): Varying-coefficient models	⊙					
Al-Shedivat et al. (2020): Contextual explanation networks	⊙					✓
Alvarez-Melis and Jaakkola (2018): Self-explaining NNs	⊙					✓
Schwab et al. (2019): Attentive mixtures of experts	⊙					✓
Koh et al. (2020)	•	∼				✓
Udrescu and Tegmark (2020): Symbolic regression	•	∼	∼	∼	∼	

Note: "•" and "⊙" denote globally- and locally-interpretable models, respectively. "✓" denotes that a property (columns) is satisfied by a technique (rows). "∼" denotes that a property either holds partially or that a model could be easily extended to satisfy the property.
Abbreviations: DeepPINK, deep feature selection using paired-input nonlinear knockoffs; GAM, generalized additive model; NN, neural network.

TABLE 6 A falling rule list from predicting the probability of appendicitis in pediatric patients based on tabular data comprising clinical, laboratory, scoring, and ultrasonography variables.

#	Rule	Probability
1.	Surrounding tissue reaction = *yes* AND Age $\notin [9.3,11.5]$	0.96
2.	Surrounding tissue reaction = *yes* AND Dysuria = *no*	0.94
3.	Pathological lymph nodes = *no* AND Appendix on Ultrasound = *yes*	0.72
4.	Peritonitis = *local* AND Erythrocytes in Urine < 3.0	0.60
5.	C-reactive protein $\in [7.0,31.75]$ AND Alvarado Score $\in [7,10]$	0.60

Note: The risk decreases monotonically across the list.

specified otherwise. For supervised learning, we assume given a training dataset of N data points $\mathcal{D} = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^N$, consisting of features $\boldsymbol{x}_i \in \mathcal{X}$ and labels $y_i \in \mathcal{Y}$. For tabular data, features are given by a p-dimensional vector $\boldsymbol{x}_i \in \mathbb{R}^p$. We use $f(\cdot)$ to refer to a classification or regression model, which may be interpretable or black-box, fitted on the training data. In the unsupervised learning scenario, we assume a dataset of unlabeled points $\mathcal{D} = \{\boldsymbol{x}_i\}_{i=1}^N$.

Throughout this section, we will occasionally provide examples of different techniques applied to a simple dataset comprising clinical, laboratory, scoring, ultrasound variables, and ultrasound images acquired from a cohort of pediatric patients admitted to the hospital with suspected appendicitis (Marcinkevics et al., 2021; Roig Aparicio et al., 2021). The underlying problem of this dataset is binary classification—the prediction of the patient's diagnosis (*appendicitis* vs. *no appendicitis*). The data analysis was approved by the University of Regensburg institutional review board (Ethikkommission der Universität Regensburg, no. 18-1063-101). The dataset is publicly available at https://github.com/i6092467/pediatric-appendicitis-ml.

4.2 | Interpretable models

Interpretable models, sometimes also referred to as "white-" or "gray-boxes," are usually constrained and structured to reflect physical constraints, monotonicity, additivity, causality, sparsity, or other desirable properties (Carvalho

TABLE 7 A supersparse linear integer model for predicting the risk of appendicitis in pediatric patients based on tabular data comprising clinical, laboratory, scoring, and ultrasonography variables (Roig Aparicio et al., 2021).

Condition	Score
Peritonitis = *generalized*	6
Appendix diameter = 9–17 mm	6
Appendix diameter = 5.9–9.0 mm	5
Appendix on ultrasound = *yes*	4
Peritonitis = *local*	2

Note: Every feature has an integral coefficient attached to it.

et al., 2019; Rudin, 2019). Some researchers have even argued that interpretable supervised machine learning can be viewed as an instance of constrained empirical risk minimization (Dziugaite et al., 2020). The choice of properties depends on the particular application and the end-user. For example, Lipton (2018) notes that a high-dimensional linear model is not more interpretable than a very compact neural network. In contrast, a *sparse* linear model is comprehensible and easy to visualize. Therefore, two desirable characteristics are (i) simulatability and (ii) decomposability (Lipton, 2018), that is, (i) a model must be comprehensible in a limited amount of time, and (ii) its inputs and parameters should be intuitively meaningful. Table 5 contains concrete examples of machine learning models that fall into this broad category.

4.2.1 | Rule-based models

Rule-based classification algorithms have been known for a long time. One could argue that these well-established techniques *are* intrinsically interpretable. While single if-then rules are indeed readily comprehensible, inductive logic programming (De Raedt, 1999), for instance, yields an unordered set of rules; on the other hand, decision trees (Loh, 2011) are not monotonic and, thus, require additional mental effort. Several rule-based classification approaches have been introduced with interpretability in mind. Some examples include repeated incremental pruning to produce error reduction (RIPPER) (Cohen, 1995), which keeps the number of rules small, RuleFit (Friedman & Popescu, 2008), which induces rules from a sparse linear model with pairwise interactions, and falling rule lists (FRL) (F. Wang & Rudin, 2015b), which prioritize monotonicity across the induced rules. Herein, we will focus on FRLs more closely as an illustrative example.

FRLs (F. Wang & Rudin, 2015b) are binary classifiers motivated by the wide adoption of risk scores and risk stratification systems in healthcare. A falling rule list is a list of if-then rules such that (i) during classification rules have to be applied in the order given by the list, and (ii) the probability of the positive class is monotonically decreasing within the list. Table 6 provides an example of an FRL for predicting the risk of appendicitis in pediatric patients, learnt from a small publicly available tabular dataset (Section 4.1) (Marcinkevics et al., 2021; Roig Aparicio et al., 2021). Notably, the rules use simple discretized features, and the risk decreases monotonically throughout the list. The constrained format of FRLs makes them more understandable than decision trees and is natural for practical decision-making in a clinical setting.

In practice, FRLs can be learnt using a Bayesian modeling approach, wherein monotonicity and sparsity constraints are encoded in the prior distribution. The simulated annealing procedure is used to sample from the posterior distribution and obtain the MAP estimator. C. Chen and Rudin (2018) further relax the original optimization problem of learning FRLs by introducing *softly* falling rule lists. Rather than having hard monotonicity constraints, the authors add a non-monotonicity penalty term to the loss function. Such formulation is better suited to noisy real-world datasets, where sparse and strictly monotonic solutions might be less performant. Another noteworthy extension is *causal* falling rule lists (F. Wang & Rudin, 2015a) that leverage FRLs to estimate treatment effects in the potential outcomes framework (Rubin, 2005).

4.2.2 | Score-based models

Another class of interpretable binary classification models, likewise motivated by medical risk scoring, is supersparse linear integer models (SLIM), introduced by Ustun and Rudin (2015). SLIMs allow learning data-driven risk scores that

are reminiscent of conventional medical scoring systems, such as APACHE (Knaus et al., 1985) or SOFA (Vincent et al., 1996). In contrast to FRLs, whose key focus is monotonicity, SLIMs represent *sparse* decision boundaries, that is, relying on a limited number of features. Moreover, interpretability in SLIMs is additionally facilitated by learning a *linear* scoring function with *integral* coefficients.

Roughly, SLIMs require solving the following optimization problem (refer to the original paper by Ustun and Rudin (2015) for the complete formulation):

$$\min_{\boldsymbol{\beta}} \frac{1}{N}\sum_{i=1}^{N} \mathbf{1}_{\{y_i \boldsymbol{\beta}^T \boldsymbol{x}_i \leq 0\}} + \lambda_0 \|\boldsymbol{\beta}\|_0 + \lambda_1 \|\boldsymbol{\beta}\|_1, \text{ s.t.} \boldsymbol{\beta} \in \mathscr{B}, \tag{1}$$

where $\boldsymbol{\beta} \in \mathscr{B}$ is an integer-valued coefficient vector with $\mathscr{B} = \{L, L+1, ..., U-1, U\}^p$, $L, U \in \mathbb{Z}$, $L < U$, and $\mathbf{1}_{\{\cdot\}}$ denotes an indicator function. Notably, the original features, if continuous, have to be discretized and encoded as binary-valued factors. The integer linear program (ILP) defined by Equation (1) enjoys the advantages of *directly* minimizing the 0–1 loss and the ℓ_0 penalty instead of convex surrogate measures commonly adopted in statistics and machine learning literature, cf. Zou and Hastie (2005). Table 7 contains an example of a scoring system learnt from tabular data using SLIM for predicting the risk of pediatric appendicitis in children. In practice, the risk for an individual described by features $\boldsymbol{x}_i$ is quantified by $\boldsymbol{\beta}^T \boldsymbol{x}_i$, that is, by the sum of the coefficients corresponding to the applicable conditions.

In addition to the theoretical guarantees, the ILP formulation above has the benefit of easily incorporating and enforcing additional constraints beyond integrality and sparsity, for example, introducing desirable "either-or" or "if-then" conditions on features or preferences for (not) using certain variables. Ustun and Rudin (2015) also introduce a range of extensions of SLIMs. Particularly noteworthy are *personalized* SLIMs with varying scoring rules for individual data points. The authors also present rule-based adaptations. Further algorithmic improvements are made by Ustun and Rudin (2017).

4.2.3 | Generalized additive models

As mentioned before, decomposability is a desirable property of interpretable ML models (Lipton, 2018). One class of "decomposable" functions is additively separable functions (Segal, 1994). We say that a function $f(x_1, x_2, ..., x_p)$ is additively separable if we can rewrite it as a sum of univariate terms: $f(x_1, x_2, ..., x_p) = \sum_{j=1}^{p} u_j(x_j)$. Hastie and Tibshirani (1986) introduce the class of *generalized additive models* (GAM) that rely on this additivity property. In particular, for p features, a GAM is given by

$$g(y) = \sum_{j=1}^{p} s_j(x_j), \tag{2}$$

where $g(\cdot)$ is a link function, $s_j(\cdot)$ are smooth functions, often referred to as shape functions (Lou et al., 2012). GAMs are an extension of the linear model that preserves the additivity but allows introducing nonlinearities in individual variables by choosing appropriate shape functions. This model class ignores interactions between variables. Therefore, the influence of each feature is easily comprehensible and can be visualized by plotting the corresponding shape function $s_j(\cdot)$. Figure 2 depicts shape functions for two continuously-valued features in a GAM for classification, fitted on a tabular dataset.

Lou et al. (2012) conduct extensive experimental comparison among different methods for fitting GAMs and choices of $s_j(\cdot)$. They consider least squares, gradient boosting, and backfitting approaches. In addition to the standard use of spline-based shape functions (Hastie & Tibshirani, 1986), Lou et al. (2012) consider single, bagged, boosted, and boosted bagged decision trees. Building on the work by Lou et al. (2012), Caruana et al. (2015) propose a simple yet more performant extension by including two-way interaction terms referred to as generalized additive models plus interactions (GA^{2}M):

$$g(y) = \sum_{j=1}^{p} s_j(x_j) + \sum_{j=1}^{p} \sum_{\substack{k=1 \\ k \neq j}}^{p} s_{j,k}(x_j, x_k), \tag{3}$$

TABLE 8 Properties of reviewed explanation techniques.

Method	Scale	Attribution	Agnostic	Reference-free	Contrastive	Diverse	Causal	Unstructured data
Ribeiro et al. (2016): LIME	⊙, •	✓	✓	✓				✓
Shrikumar et al. (2017): DeepLIFT	⊙	✓						✓
Lundberg and Lee (2017): SHAP	⊙, •	✓	✓	✓				✓
Sundararajan et al. (2017): Integrated gradients	⊙	✓						✓
Erion et al. (2021): Expected gradients	⊙	✓		✓				✓
Kim et al. (2018): Testing with CAVs	•			✓				✓
Schrouff et al. (2021): Integrated CS	⊙							✓
Alaa and van der Schaar (2019): Symbolic metamodels	•	✓	✓	✓				
Wachter et al. (2017): CF explanations	⊙			✓	✓			∼
Mothilal et al. (2020): Diverse CF explanations	⊙			✓	✓	✓		∼
Karimi, Barthe, Balle, and Valera (2020): MACE	⊙		✓	✓	✓			∼
Mahajan et al. (2019): VAE-based CF explanations	⊙		✓	✓	✓	∼	∼	✓
S. Liu et al. (2019): GAN-based CF explanations	⊙		✓	✓	✓	∼		✓
Chang et al. (2019): FIDO CF saliency maps	⊙	✓	✓	✓	∼			✓
Algorithmic recourse								
Karimi, von Kügelgen, Schölkopf, & Valera (2020)	⊙			✓	✓		✓	

Note: "•" and "⊙" denote global and local explanation methods, respectively. "✓" denotes that a property (columns) is satisfied by a technique (rows). "∼" denotes that a property either holds partially or that a method could be easily extended to satisfy the property.

Abbreviations: CAV, concept activation vector; CF, counterfactual; CS, contextual sensitivity; DeepLIFT, deep learning important features; FIDO, fill-in the dropout; GAN, generative adversarial network; LIME, local interpretable model-agnostic explanations; MACE, model-agnostic counterfactual explanations; SHAP, Shapley additive explanations; VAE, variational autoencoder.

where $s_{j,k}(x_j, x_k)$ are pairwise interaction terms. Pairwise interactions are still intelligible since they can be visualized using simple plots, for example, heat maps. Another noteworthy extension of GAMs is sparse additive models (SpAM), proposed by Ravikumar et al. (2007). SpAMs combine the ideas of Hastie and Tibshirani (1986) with sparse linear modeling for high-dimensional regression problems (Zou & Hastie, 2005). In addition to shape functions $s_j(\cdot)$, the authors introduce a weight vector $\boldsymbol{b} \in \mathbb{R}^p$ multiplied with outputs of $s_j(\cdot)$ and penalize its norm in the loss function. In this way, SpAMs rely on a *sparse* subset of features while still preserving the additive structure of the classical GAMs. To further improve the scalability and performance of GAMs on large datasets and their modularity, Agarwal et al. (2021) and Chang et al. (2021) introduce neural generalized additive models that rely on neural networks as building blocks. Specialized cases of this model class address specific modeling tasks or induce additional inductive biases, for example, application to survival analysis by Utkin et al. (2022) or, similar to SpAMs, *sparse* neural additive models proposed by S. Xu et al. (2022). Another line of work focuses on improving interactivity and actionability of GAMs via scalable and accessible visual diagnostics and editing programs (Fasiolo et al., 2019; Hohman et al., 2019; Z. J. Wang et al., 2021).

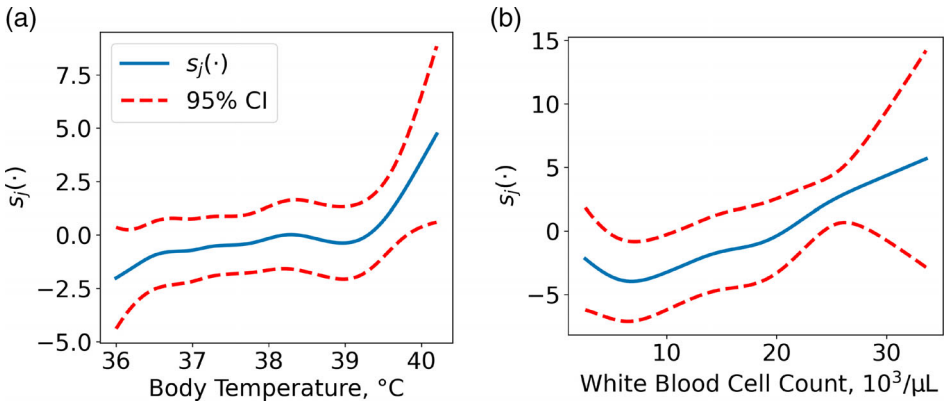


FIGURE 2 Example of a visual interpretation of a generalized additive model (GAM) for predicting the risk of appendicitis among pediatric patients based on tabular data comprising clinical, laboratory, scoring, and ultrasonography variables. The plots depict shape functions $s_j(\cdot)$ for two features present in the dataset: (a) body temperature and (b) white blood cell count. Functions are plotted as solid lines, 95% confidence intervals (CI) are plotted as dashed lines. Observe that the GAM predicts higher probabilities of appendicitis in children with higher body temperatures and white blood cell counts. The plots were generated using pyGAM library (Servén & Brummitt, 2018).

4.2.4 | Sparse-input neural networks

In many high-dimensional regression and classification problems, for example, genomic data analysis (Lucas et al., 2006) and social network modeling (Ravazzi et al., 2018), sparsity is an important inductive bias that allows producing parsimonious interpretable models. We have already mentioned sparsity as a desirable property when describing supersparse linear integer models (Section 4.2.2). However, the predictive performance of SLIMs could be limited by their assumption of linearity. Recently, there have been renewed efforts in leveraging sparsity-inducing regularization to understand and control the behavior of neural networks models (Feng & Simon, 2017; Khanna & Tan, 2020; Lu et al., 2018; Tank et al., 2021; Valdes et al., 2021). Significant advantages of neural networks are their ability to model complex nonlinear relationships and their scalability to large datasets and unstructured data types, such as text and images.

Feng and Simon (2017) provide a thorough theoretical and empirical analysis of sparse-input neural networks (SPINN) in the context of $p \gg N$ problems (Hastie et al., 2009). SPINNs are fully connected neural networks characterized by sparse weights in the input layer and are trained by minimizing the following loss function:

$$\min_{\boldsymbol{\theta}} \frac{1}{N}\sum_{i=1}^{N} \mathscr{L}(y_i, f_{\boldsymbol{\theta}}(\boldsymbol{x}_i)) + \lambda_0 \sum_{a=2}^{L} \left\| \boldsymbol{W}^{(a)} \right\|_2^2 + \lambda \sum_{j=1}^{p} \Omega_\alpha\left(\boldsymbol{W}_{\cdot j}^{(1)}\right), \tag{4}$$

where $\boldsymbol{\theta} = \left\{\boldsymbol{W}^{(a)}\right\}_{a=1}^{L}$ are weight matrices in layers $1 \le a \le L$; $\boldsymbol{W}_{\cdot j}^{(1)}$ refers to the j-th column of the input layer weight matrix; and $\Omega_\alpha(\boldsymbol{\beta}) = (1-\alpha)\|\boldsymbol{\beta}\|_1 + \alpha\|\boldsymbol{\beta}\|_2$ with $\alpha \in (0,1)$ is the sparse group Lasso penalty (Simon et al., 2013). Here, parameter α controls the trade-off between the element-wise Lasso and the group Lasso penalties. Simply put, this penalty ensures that all input weights corresponding to a single feature are shrunk together, allowing for feature selection in the style of the classical Lasso (Figure 3).

Feng and Simon (2017) prove probabilistic, finite-sample generalization guarantees for this model class and demonstrate performance gains empirically for high-dimensional data with higher-order interactions compared to other nonparametric models. Tank et al. (2021) leverage similar penalties in the context of autoregressive time series modeling. Khanna and Tan (2020) apply the approach to a more advanced neural network architecture, namely, the long short-term memory (Hochreiter & Schmidhuber, 1997).

4.2.5 | Knockoff features

One could argue that the ultimate goal of SPINNs (Feng & Simon, 2017) is "deep" feature selection. In a similar vein, Lu et al. (2018) propose another solution—deep feature selection using paired-input nonlinear knockoffs (DeepPINK),

which leverages knockoff filters (Barber & Candès, 2015) to facilitate interpretability and sparsity in deep neural networks at the input level. A compelling advantage of this technique over SPINNs is that it controls for the false discovery rate (FDR) (Benjamini & Hochberg, 1995) when selecting significant features. Knockoff filters were initially proposed by Barber and Candès (2015). In brief, knockoff filters are a variable selection procedure that controls the FDR exactly in a linear model in finite sample settings, whenever there are at least as many observations as features. The key idea is to construct knockoff features mimicking the dependency structure of the original features; augment the original dataset with knockoffs; and compare statistics for each original feature and its knockoff, for example, the absolute value of the regression coefficient. In this way, variables with genuine signals can be identified while controlling for the FDR.

Candès et al. (2018) propose "*model-X*" knockoffs which rely on the assumption that the joint distribution of variables is known, without assuming *anything* about the distribution of the output conditional on features. Their significant limitation is that the generation of knockoff features is based on a known multivariate Gaussian distribution. Jordon et al. (2019) alleviate this issue by introducing the KnockoffGAN—a generative adversarial network (GAN) (Goodfellow et al., 2014) for knockoff generation capable of producing more complex dependency structures. Along similar lines, following the model-*X* framework, Romano et al. (2019) propose constructing knockoff features with deep generative models utilizing the maximum mean discrepancy (MMD) (Y. Li et al., 2015).

4.2.6 | Varying-coefficient models

While GAMs (Section 4.2.3) generalize the linear model by allowing for nonlinearities in individual features, varying-coefficient models (VCM), proposed by Hastie and Tibshirani (1993), offer a different sort of generalization. In a VCM, variable coefficients vary smoothly with so-called "effect modifiers"—additional, potentially exogenous, variables $r_1, r_2, ..., r_p$:

$$g(y) = \beta_0 + \sum_{j=1}^{p} x_j \beta_j(r_j), \tag{5}$$

wherein $\beta_j(\cdot)$ is a smooth function corresponding to the varying coefficient of the j-th feature. The choice of variables $r_1, ..., r_p$ depends on a particular application. Note that r_j may coincide with features x_j or correspond to some additional attributes. For example, in dynamical systems, time can be a single effect modifier, producing time-varying coefficients.

Notably, VCMs are only *locally* interpretable since coefficients vary across data points, that is, interpretations may differ wildly between different instances. By contrast, all of the model classes described before are *globally* interpretable: the insights gained from inspecting model parameters are equally applicable to all data points. While the interpretation of locally interpretable is far more cumbersome, this trade-off may be necessary in pursuit of a more *flexible* model with *personalized* interpretations. Several more recent interpretable ML models (Al-Shedivat et al., 2020; Alvarez-Melis & Jaakkola, 2018) discussed in the following sections bear a striking resemblance to VCMs: they essentially generalize this

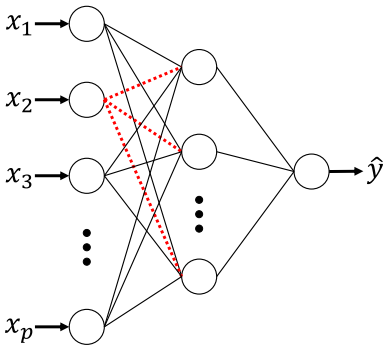


FIGURE 3 A graphical representation of a fully connected neural network with p input variables $x_1, ..., x_p$. Input weights corresponding to the feature x_2 are shown as bold dashed lines. If x_2 were not useful for predicting the output, a sparse-input network would shrink all of its input weights $\boldsymbol{W}^{(1)}_{\cdot,2}$ toward 0 thus, deselecting x_2 completely.

conventional statistical framework to unstructured data types, such as images, by parameterizing $\beta_j(\cdot)$ with neural networks.

4.2.7 | Contextual explanation networks

We have seen before how sparsity could be introduced in fully connected neural networks and feature selection could be performed (Sections 4.2.4 and 4.2.5). Nevertheless, quantifying and explaining the *contributions* of individual inputs to the predictions in neural networks is not straightforward due to entangled interactions in downstream layers of a network (Guerguiev et al., 2017; Tank et al., 2021). While there has been a substantial effort and progress in explaining neural network models post hoc (Section 4.3.1), that is, after training, a few lines of research instead focus on building *interpretable* neural network architectures whose structure is decomposable and whose parameters can be interpreted directly to produce *local* explanations.

For instance, Al-Shedivat et al. (2020) introduce contextual explanation networks (CEN)—a class of neural network architectures that jointly predict and explain their predictions without requiring additional model introspection. CENs can be defined as deep probabilistic models for learning the conditional distribution of the output variables $\mathbf{P}_{\boldsymbol{w}}(\boldsymbol{Y} \mid \boldsymbol{x}, \boldsymbol{c})$, parameterized by $\boldsymbol{w}$, where $\boldsymbol{c} \in \mathcal{C}$ are *context* variables observed in addition to the features $\boldsymbol{x} \in \mathcal{X}$ and $\boldsymbol{y} \in \mathcal{Y}$ are outputs, to be predicted given $\boldsymbol{x}$ and $\boldsymbol{c}$. The probabilistic model is then specified by

$$\begin{aligned} \boldsymbol{y} &\sim \boldsymbol{P}(\boldsymbol{Y}|\boldsymbol{x},\boldsymbol{\theta}), \\ \boldsymbol{\theta} &\sim \boldsymbol{P}_{\boldsymbol{w}}(\boldsymbol{\theta}|\boldsymbol{c}), \\ \mathbf{P}_{\boldsymbol{w}}(\boldsymbol{Y}|\boldsymbol{x},\boldsymbol{c}) &= \int_{\boldsymbol{\vartheta} \in \Theta} \mathbf{P}(\boldsymbol{Y}|\boldsymbol{x},\boldsymbol{\vartheta}) \mathbf{P}_{\boldsymbol{w}}(\boldsymbol{\vartheta}|\boldsymbol{c}) d\boldsymbol{\vartheta}, \end{aligned} \tag{6}$$

where $\mathbf{P}(\boldsymbol{Y}|\boldsymbol{x},\boldsymbol{\theta})$ is a predictive model parameterized by $\boldsymbol{\theta}$ that explicitly relates features to the outputs. Thus, parameters $\boldsymbol{\theta} \in \Theta$ can be seen as an explanation of a model's prediction that is specific to the context given by variables $\boldsymbol{c}$. For example, Al-Shedivat et al. (2020) consider the problem of poverty prediction based on categorical variables from living standards measurement surveys and the context given by satellite images. In practice, $\mathbf{P}_{\boldsymbol{w}}(\boldsymbol{\theta}|\boldsymbol{c})$ is replaced with an encoder neural network and the predictive distribution $\mathbf{P}(\boldsymbol{Y}|\boldsymbol{x},\boldsymbol{\theta})$ is parameterized by an interpretable function, for example, a linear model $f_{\boldsymbol{\theta}}(\boldsymbol{x}) = \mathrm{softmax}(\boldsymbol{\theta}^T \boldsymbol{x})$.

CENs are closely related to VCMs (Hastie & Tibshirani, 1993). In fact, they can be seen as a special case: context variables $\boldsymbol{c}$ (Equation 6) are the effect modifiers for features $\boldsymbol{x}$ (cf. Equation 5). The principal contribution of CENs is to cast the VCMs into a probabilistic framework and parameterize coefficients with neural networks. The authors demonstrate the efficacy of their approach on classification and survival analysis tasks. They show that CENs are still interpretable in datasets with noisy features where post hoc explanation techniques (Section 4.3) are often inconsistent and misleading.

4.2.8 | Self-explaining neural networks

Another class of functions related to VCMs was introduced by Alvarez-Melis and Jaakkola (2018). Similarly to Al-Shedivat et al. (2020), the authors develop an intrinsically interpretable neural network model that allows disentangling contributions of individual features or basis concepts. Self-explaining neural networks (SENN) (Alvarez-Melis & Jaakkola, 2018) are motivated by (i) explicitness, (ii) faithfulness, and (iii) stability properties—three desiderata for interpretability. The authors claim that SENNs are (i) explicit because their explanations are "*immediate*" and "*understandable*," (ii) faithful because explanations reflect the ground truth relationship between the basis concepts and outputs, and (iii) stable because their explanations are consistent for similar data points.

SENNs act like a simple model locally but can be highly complex and nonlinear globally. In their most basic form, SENNs are given by

$$f(\boldsymbol{x}) = \boldsymbol{\theta}(\boldsymbol{x})^T \boldsymbol{x}, \tag{7}$$

where $\boldsymbol{\theta}(\cdot)$ is a neural network with p outputs referred to as generalized coefficients. Without further restrictions, the model in Equation (7) is not more interpretable than a classical multilayer neural network. Therefore, SENNs are encouraged to be locally linear: it needs to hold that $\nabla_{\boldsymbol{x}} f(\boldsymbol{x}) \approx \boldsymbol{\theta}(\boldsymbol{x}_0)$ for all $\boldsymbol{x}$ in the neighborhood of $\boldsymbol{x}_0$. Under this constraint, individual components of $\boldsymbol{q}(\boldsymbol{x})$ act as interpretable and adaptive regression coefficients.

Further extensions of the model in Equation (7) are possible. For example, instead of raw features one can introduce basis concepts $\boldsymbol{h}(\boldsymbol{x}) : \mathbb{R}^p \to \mathbb{R}^k$ and use them alongside the generalized coefficients: $f(\boldsymbol{x}) = \boldsymbol{\theta}(\boldsymbol{x})^T \boldsymbol{h}(\boldsymbol{x})$. Furthermore, depending on the ground task, some generalized interpretable link function $g(\cdot)$ can be used, resulting in the refined model definition below (see Figure 4 for a schematic visualization):

$$f(\boldsymbol{x}) = g\big(\theta(\boldsymbol{x})_1 h(\boldsymbol{x})_1, ..., \theta(\boldsymbol{x})_k h(\boldsymbol{x})_k\big), \tag{8}$$

where $z_j = \theta(\boldsymbol{x})_j h(\boldsymbol{x})_j$ is the influence score, or importance, of the j-th concept for data point $\boldsymbol{x}$.

In practice, a SENN in Equation (8) is trained by minimizing the following gradient-regularized loss function that balances performance with interpretability:

$$\mathscr{L}_{\boldsymbol{y}}(f(\boldsymbol{x}), \boldsymbol{y}) + \lambda \mathscr{L}_{\boldsymbol{\theta}}(f(\boldsymbol{x})), \tag{9}$$

where $\mathscr{L}_{\boldsymbol{y}}(f(\boldsymbol{x}), \boldsymbol{y})$ is a loss term for the ground classification or regression task, for example, the mean squared error or the cross entropy; $\lambda > 0$ is a regularization parameter; and $\mathscr{L}_{\boldsymbol{\theta}}(f(\boldsymbol{x}))$ is the gradient penalty:

$$\mathscr{L}_{\boldsymbol{\theta}}(f(\boldsymbol{x})) = \big\| \nabla_{\boldsymbol{x}} f(\boldsymbol{x}) - \boldsymbol{\theta}(\boldsymbol{x})^T \boldsymbol{J}_{\boldsymbol{x}}^{\boldsymbol{h}}(\boldsymbol{x}) \big\|_2, \tag{10}$$

where $\boldsymbol{J}_{\boldsymbol{x}}^{\boldsymbol{h}}$ is the Jacobian of $\boldsymbol{h}(\cdot)$ w.r.t. $\boldsymbol{x}$. Alvarez-Melis and Jaakkola (2018) postulate further desirable properties that SENNs should satisfy:

1. The link function $g(\cdot)$ is monotonic and additively separable in its arguments (see Sections 4.2.1 and 4.2.3).
2. The link function is influenced by all of the basis concepts, that is, for all $1 \le i \le k$, $\frac{\partial g}{\partial z_i} > 0$ where $z_i = \theta(\boldsymbol{x})_i h(\boldsymbol{x})_i$.
3. 3. Generalized coefficients $\boldsymbol{\theta}(\cdot)$ are locally difference-bounded by $\boldsymbol{h}(\cdot)$, that is, for every $\boldsymbol{x}_0$, there exist $\delta > 0$ and $L \in \mathbb{R}$ such that if $\|\boldsymbol{x} - \boldsymbol{x}_0\|_2 < \delta$, then $\|\boldsymbol{\theta}(\boldsymbol{x}) - \boldsymbol{\theta}(\boldsymbol{x}_0)\|_2 \le L \|\boldsymbol{h}(\boldsymbol{x}) - \boldsymbol{h}(\boldsymbol{x}_0)\|_2$.
4. Basis concepts $\{h(\boldsymbol{x})_i\}_{i=1}^k$ are interpretable representations of $\boldsymbol{x}$.
5. The number of concepts k is small.

In addition, the authors emphasize three guiding criteria for choosing interpretable basis concepts: (i) fidelity—representations should contain relevant context information; (ii) diversity—concepts used to represent inputs should be few and nonoverlapping; and (iii) grounding—concepts should be immediately understandable to a human. Moreover, they demonstrate how such representations can be learnt using autoencoder neural networks in an end-to-end manner in conjunction with the SENN model.

The class of functions described by the assumptions above is quite broad, for example, generalized linear models (Nelder & Wedderburn, 1972) and the nearest neighbor classifier satisfy these. Nevertheless, the advantage of SENNs stems from the richness of neural network architectures that could be used for functions $\boldsymbol{\theta}(\cdot)$ and $\boldsymbol{h}(\cdot)$. Like CENs (Section 4.2.7), SENNs are closely related to varying-coefficient models (Section 4.2.6). The main difference is that in SENNs, regressors themselves act as effect modifiers and that the framework is augmented with interpretable basis concepts, defined on top of the raw inputs. Notably, all of the three model classes (Al-Shedivat et al., 2020; Alvarez-Melis & Jaakkola, 2018; Hastie & Tibshirani, 1993) described so far hold a promise of local interpretability while providing room for predictively powerful models.

4.2.9 | Attentive mixtures of experts

In natural language processing, the attention mechanism (Vaswani et al., 2017) has become a powerful tool for exploring relationships between inputs and outputs of deep neural networks and is utilized for interpretability and

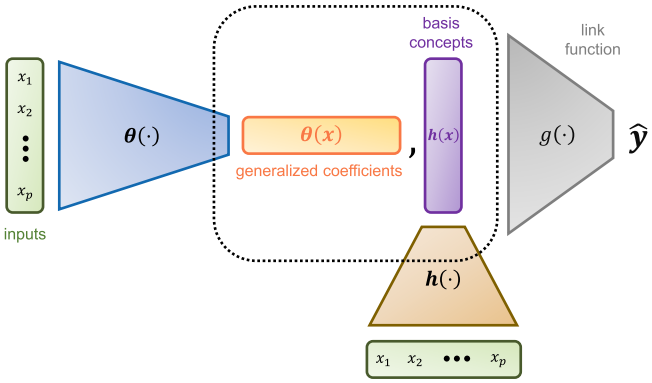


FIGURE 4 A schematic depiction of a self-explaining neural network model. Input variables $x_1,...,x_p$ are mapped to generalized coefficients and interpretable basis concepts by neural networks $\boldsymbol{\theta}(\cdot)$ and $\boldsymbol{h}(\cdot)$, respectively. Generalized coefficients and basis concepts are then combined by an interpretable link function $g(\cdot)$ into a predicted value $\widehat{\boldsymbol{y}}$.

performance. Nevertheless, several works have criticized the naïve use of attention for model interpretation (Jain & Wallace, 2019; Serrano & Smith, 2019), showing that it is often uncorrelated with gradient information and other natural feature importance measures. Some works have focused on improving the attention mechanism, particularly for interpretability, for example, models by Nauta et al. (2019) and Schwab et al. (2019). Schwab et al. (2019) propose a mixture of experts model with attentive gates for learning feature importance values alongside predictions. They introduce an auxiliary objective to mitigate the shortcomings mentioned earlier.

The attentive mixture of experts (AME) is a neural network model comprising several connected "experts," that is, subnetworks. The AME is given by the following equations:

$$\begin{aligned} f(\boldsymbol{x}) &= \sum_{j=1}^{p} \underbrace{G_j(\boldsymbol{h}_{all})}_{a_j} \underbrace{E_j(x_j)}_{c_j}, \\ \boldsymbol{h}_{all} &= \left[h_1, c_1, h_2, c_2, ..., h_p, c_p\right], \\ a_j &= \frac{\exp\left(\boldsymbol{u}_j^T \boldsymbol{u}_{s,j}\right)}{\sum_{k=1}^{p} \exp\left(\boldsymbol{u}_k^T \boldsymbol{u}_{s,k}\right)}, \qquad \text{for } 1 \le j \le p, \\ \boldsymbol{u}_j &= \sigma\left(\boldsymbol{W}_j \boldsymbol{h}_{all} + b_j\right), \qquad \text{for } 1 \le j \le p, \end{aligned} \tag{11}$$

where $c_j = E_j(x_j)$ is the output of the j-th expert subnetwork given the input variable x_j; a_j is the output of the j-th attentive gating network $G_j(\cdot)$ quantifying the importance of the j-th feature; h_j denotes a hidden representation from the j-th expert subnetwork; vector $\boldsymbol{u}_j$ is a projected representation of $\boldsymbol{h}_{all}$; vector $\boldsymbol{u}_{s,j}$ is a per-expert learnable context vector; and $\sigma(\cdot)$ is a nonlinear activation function. Notably, the architecture specified by Equation (11) allows disentangling contributions of individual features to predictions, using attentive gating and per-feature subnetworks.

The AME is trained end-to-end by minimizing a loss function augmented with an auxiliary objective. The auxiliary objective encourages the importance score a_j to reflect the decrease in error associated with the contribution of the j-th expert, that is, the j-th feature, similar in spirit to the definition of the RF variable importance (Breiman, 2001):

$$\Delta\varepsilon_{\boldsymbol{x},j} = \varepsilon_{\boldsymbol{x}\setminus\{j\}} - \varepsilon_{\boldsymbol{x}}, \quad \text{for } 1 \le j \le p, \tag{12}$$

where $\varepsilon_{\boldsymbol{x}\setminus\{j\}}$ and $\varepsilon_{\boldsymbol{x}}$ denote the prediction error of the model without the j-th feature and of the full model, respectively. The error difference above can be normalized to produce

$$\omega_{\boldsymbol{x},j} = \frac{\Delta\varepsilon_{\boldsymbol{x},j}}{\sum_{k=1}^{p} \Delta\varepsilon_{\boldsymbol{x},k}}, \quad \text{for } 1 \le j \le p. \tag{13}$$

Ideally, attentive gating network outputs should be correlated with the feature importance measure defined in Equation (13). Therefore, Schwab et al. (2019) introduce an auxiliary term into the loss function for training the AME. In particular, the authors minimize the discrepancy between normalized error differences (Equation 13) and the outputs of the attentive gating networks:

$$\mathscr{L}_{\text{aux}} = \frac{1}{N}\sum_{i=1}^{N} D(\boldsymbol{\omega}_{\boldsymbol{x}_i}, \boldsymbol{a}_{\boldsymbol{x}_i}), \tag{14}$$

where $D(\cdot,\cdot)$ is some discrepancy measure, for example, the Kullback–Leibler divergence; $\boldsymbol{\omega}_{\boldsymbol{x}_i}$ and $\boldsymbol{a}_{\boldsymbol{x}_i}$ are normalized error differences and the attentive gating network outputs, respectively, for the i-th data point.

The attentive mixture of experts successfully overcomes limitations of the naïvely trained attention mechanism by introducing regularization terms into the loss function which forces learning importance scores that reflect the increase in prediction error from removing a feature. Next to CENs and SENNs (Sections 4.2.7 and 4.2.8), AMEs are yet another class of locally interpretable neural network architectures which produce individualized explanations for each data point at prediction time. However, the relationship between predictions and explanations is, arguably, more opaque within the AMEs than the other model classes discussed so far.

4.2.10 | Symbolic regression

Most interpretable models tend to learn some numerical measure of feature importance that can be visualized and interpreted either globally or locally. By contrast, as its name suggests, symbolic regression tries to provide a *symbolic* interpretation of the data. More formally, symbolic regression is a problem of inferring an *analytic* form for an unknown function that can only be queried (Amir Haeri et al., 2017; Udrescu & Tegmark, 2020). Although this problem emerged long before the relatively recent interest in explainable and interpretable ML (McKay, 1995), symbolic regression perfectly fits the broad category of interpretable models. While neural networks and many other models do provide analytic representations, symbolic regression usually seeks *parsimonious* equations, for example, making further restrictions to polynomial forms. Thus, symbolic regression can be leveraged to learn interpretable functional relationships from raw data (Jin et al., 2019). For example, when regressing y on features $\boldsymbol{x}$, a mathematical expression $f(\boldsymbol{x}) = x_1 + 2\cos(x_2) + \exp(x_3) + 0.1$ could be a candidate solution for symbolic regression. The optimization problem behind symbolic regression can be formalized as follows:

$$\min_{f \in \mathfrak{F}} \frac{1}{N}\sum_{i=1}^{N} \mathscr{L}(f(\boldsymbol{x}_i), y_i), \tag{15}$$

where $\mathfrak{F}$ is a set of *succinct* mathematical expressions and $\mathscr{L}(\cdot,\cdot)$ is the loss function for the ground regression or classification task. In practice, symbolic regression often reduces to combinatorial optimization. Therefore, some conventional approaches include genetic programming (Amir Haeri et al., 2017; McKay, 1995) and simulated annealing (Stinstra et al., 2007). Petersen et al. (2021) and Biggio et al. (2021) have proposed neural-network-based solutions to the problem. Namely, Biggio et al. (2021) predict symbolic expressions using a Transformer model pretrained on a large-scale corpus of procedurally generated input dataset and symbolic equation pairs. Last but not least, we remark that symbolic regression is also helpful for explaining the behavior of black-box machine learning models post hoc, for example, a neural network could be approximated by a symbolic surrogate model (Section 4.3.3).

4.2.11 | Interpretable representation learning

In the previous sections, we have considered interpretability exclusively in supervised learning and at the level of raw input variables. Sometimes we might want to learn low-dimensional embeddings, or representations, in an unsupervised, weakly- or semi-supervised setting instead of exploring purely discriminative relationships among raw variables. Representations can be helpful for several, usually unknown at the time of representation learning,

downstream applications. A desirable property targeted by some of the representation learning techniques is interpretability. Similarly to the classification and regression settings, interpretability is usually attained by enforcing some constraints on representations.

Disentanglement is one such constraint: in disentangled representations, separate sets of dimensions are uniquely correlated with salient, semantically meaningful features. In addition to interpretability, disentanglement facilitates the controllable generation of synthetic data. Recently, many deep generative models have been used for disentangled representation learning—for example, X. Chen et al. (2016) demonstrate experimentally that their InfoGAN model, an information-theoretic extension of GANs, learns disentangled representations from image data. Similar results have been attained with variational autoencoders (VAE) (Higgins et al., 2016; Kingma & Welling, 2014; Kingma & Welling, 2019) by introducing statistical independence constraints on embedding dimensions via a factorizing prior distribution. In theory, learning identifiable disentangled representations in a completely unsupervised manner is fundamentally impossible (Locatello et al., 2019). However, the latter result does not diminish the utility of disentangled representation learning by injecting inductive biases and implicit or explicit forms of supervision. For instance, Adel et al. (2018) and Taeb et al. (2022) consider semi-supervised variants of the VAE wherein conditioning the representation on some side information or label helps the disentanglement and interpretability of the generative model. Tschannen et al. (2018) provide a thorough overview of noteworthy advancements and inductive biases in autoencoder-based representation learning. Many of the approaches discussed by them strive toward some form of interpretability, although often do not state that explicitly.

Disentanglement is not the only approach to interpretable representation learning. Several lines of work have focused on introducing additional supervision to learn representations reflecting high-level concepts useful in classification or regression (Z. Chen et al., 2020; Koh et al., 2020; Marcos et al., 2021). One such approach is concept bottleneck models (N. Kumar et al., 2009; Lampert et al., 2009), recently reexplored by Koh et al. (2020). As opposed to the deep generative models discussed above, concept bottlenecks perform supervised learning. For predicting the output y based on features $\boldsymbol{x} \in \mathbb{R}^p$, a bottleneck model is given by $f(g(\boldsymbol{x}))$, where $g : \mathbb{R}^p \to \mathbb{R}^k$ and $f : \mathbb{R}^k \to \mathbb{R}$. Here, $f(\cdot)$ relies entirely on interpretable concepts $\boldsymbol{c} = g(\boldsymbol{x})$, and $g(\cdot)$ is learnt in a supervised manner using additional concept labels. For instance, consider classifying bird images into species based on a set of visual traits defined by ornithologists. Koh et al. (2020) explore a range of strategies for training such models and propose to parameterize $f(\cdot)$ and $g(\cdot)$ by deep neural networks. A significant advantage of a concept bottleneck $f(g(\boldsymbol{x}))$ over a block-box $\widetilde{f}(\boldsymbol{x})$ is that at prediction time, an expert end-user, for example, a medical doctor, can intervene on incorrectly inferred concepts. However, the applicability of concept bottleneck models is limited to areas and tasks where vast domain knowledge is available and where experts can cheaply label instances.

To summarize, the problem of interpretability in representation and, more broadly, unsupervised learning is still under-explored, despite a growing body of research. As seen from the previous sections, many techniques focus exclusively on classification and regression tasks, while deep clustering, generative modeling, and representation learning have attracted comparatively less attention. With the emergence of new socially consequential application domains, a need for interpretable unsupervised learning techniques is becoming apparent.

4.3 | Explanation methods

We now turn toward a completely different family of methods. According to Rudin (2019), *explainable* ML focuses on introspection for existing black-box models, for instance, by training a simpler surrogate model post hoc. As seen before, explanations can take various forms: textual, visual, symbolic, and so forth. Even a data point from the training set or a synthetic data point can serve as an explanation (Lipton, 2018). An explanation can be global, that is, characterizing the whole dataset, or local, that is, explaining individual classification or regression outcomes (Carvalho et al., 2019; Molnar, 2020). It can be model-specific, that is, capable of explaining only a specific class of models, or model-agnostic, that is, applicable to an arbitrary model. Carvalho et al. (2019) discuss several desirable properties of an explanation technique: (i) faithfulness—an explanation should be faithful to the original black-box model, that is, an explanation should in some way accurately predict the behavior of the black-box; (ii) consistency and stability—explanations for different models tackling the same task should be consistent, and explanations for similar data points should be similar; (iii) comprehensibility—the end-user should be able to comprehend explanations easily; (iv) certainty and novelty—explanations should convey (un-)certainty about predictions and should warn the end-user if the data point considered is "*far away*" from the support of the training set; (v) representativeness—explanations

should "*cover*" training data evenly, particularly for prototype-based explanation techniques (Kim et al., 2014, 2016). Table 8 further expands on the list of salient characteristics and presents the explanation methods overviewed in the following sections.

4.3.1 | Attribution methods

Arguably, the family of explanation techniques that is used most frequently in practical applications are attribution methods. Sundararajan et al. (2017) define an attribution as follows: for a function $f:\mathbb{R}^p \to [0,1]$, representing a black-box binary classifier, and an input $\boldsymbol{x} \in \mathbb{R}^p$, an attribution for $\boldsymbol{x}$ with respect to some reference, also referred to as baseline, $\boldsymbol{x}_0$ (some of the attribution techniques do not require a reference sample $\boldsymbol{x}_0$) is given by $A_f(\boldsymbol{x},\boldsymbol{x}_0) = \left(a_1 \cdots a_p\right)^T$, wherein a_j quantifies the "contribution" of the feature x_j to the prediction made by the model $f(\cdot)$ for data point $\boldsymbol{x}$. We will adhere to the definition and notation above throughout this section. The use of attributions as a model diagnostic has become ubiquitous in ML applications (Arcadu et al., 2019; Kelley et al., 2018; Y. Liu et al., 2020; Parsa et al., 2020), especially for image data, since attributions can be readily visualized as a heat map and are helpful for understanding on which input regions the model "concentrates." Figure 5 shows an example of an attribution heat map for a deep neural network classification model trained to predict appendicitis in children based on ultrasound images.

Most recent attribution techniques focus specifically on explaining deep neural network models, and many of them implicitly or explicitly rely on gradient information to produce attributions. Ancona et al. (2019) distinguish two different categories of attribution methods: (i) sensitivity-based methods quantify how strongly the output of the model $f(\cdot)$ changes if an input variable is perturbed, whereas (ii) salience-based methods quantify marginal effects of features on the output of $f(\cdot)$ compared to some baseline, for example, the same input but with the feature of interest masked or removed. Below we describe a few archetypal examples of attribution techniques.

Lime

Local interpretable model-agnostic explanations (LIME), introduced by Ribeiro et al. (2016), seek interpretable data representations that are faithful to the given black-box classifier $f(\cdot)$. The authors define an explanation $\xi(\cdot)$ for a data point $\boldsymbol{x}$ as follows:

$$\xi(\boldsymbol{x}) = \arg\min_{g \in \mathfrak{G}} \mathscr{L}(f,g,\pi_{\boldsymbol{x}}) + \Omega(g), \tag{16}$$

where $\mathfrak{G}$ is a class of surrogate models used for explaining the black-box; $\mathscr{L}(\cdot,\cdot,\cdot)$ is the fidelity function quantifying the loss for $g(\boldsymbol{x})$ approximating $f(\cdot)$ within the neighborhood of $\mathbf{x}$ given by $\pi_{\boldsymbol{x}}$; and $\Omega(\cdot)$ is a model complexity penalty. Essentially, $\mathscr{L}(\cdot,\cdot,\cdot)$ is a locality-aware loss function and, in practice, can be minimized in a model-agnostic manner, that is, regardless of the model class of the original black-box. Usually, $\mathfrak{G}$ is chosen to be a constrained class of intrinsically interpretable models (Section 4.2), for example, linear models or GAMs. Put simply, LIME trains many interpretable surrogate models to approximate a black-box model $f(\cdot)$ *locally*. During training, instances are sampled around each data point $\boldsymbol{x}_i$ weighted by $\pi_{\boldsymbol{x}_i}$. In addition to local explanations given by $\xi(\cdot)$, Ribeiro et al. (2016) introduce a procedure for obtaining a global understanding of the model $f(\cdot)$: given a limited budget, their algorithm picks several explanations based on greedy submodular optimization (Krause & Golovin, 2014) and aggregates them into global variable importances, similar to the random forest feature importance (Breiman, 2001).

DeepLIFT

Shrikumar et al. (2017) introduce an efficient method for disentangling contributions of inputs in a neural network—*deep l*earning *i*mportant *f*eatures (DeepLIFT). Compared to LIME, DeepLIFT is not model-agnostic since it is explicitly tailored to neural networks; it also requires a reference, or baseline, data point. While in natural images an all-black image is typically used as a baseline input, the choice of a reference might not be so trivial for more specialized datasets and could affect the attribution (Srinivas & Fleuret, 2019).

Let t denote the activation of neuron of interest, usually one of the output neurons, and let $\eta_1,\eta_2,...,\eta_K$ be intermediate neurons, potentially, from several layers, that suffice to compute t. Let $\Delta t = t - t_0$ be the difference between t and a reference output t_0. We then seek to assign contribution scores $C_{\Delta\eta_i\Delta t}$ so that they satisfy the so-called summation-to-delta property:

$$\sum_{i=1}^{K} C_{\Delta\eta_i \Delta t} = \Delta t. \tag{17}$$

An intuitive interpretation of the equation above is that $C_{\Delta\eta_i\Delta t}$ is the amount of "blame" for the difference in outputs assigned to a difference in the activation of the i-th intermediate neuron. Since neurons $\eta_1, \eta_2, ..., \eta_K$ suffice to compute t, differences in their activations $\Delta\eta_1, \Delta\eta_2, ..., \Delta\eta_K$ should suffice to explain the difference Δt. Notably, $C_{\Delta\eta_i\Delta t}$ need not be 0 when $\frac{\partial t}{\partial \eta_i} = 0$ and, thus, can yield insights very different from those of gradient-based measures.

By analogy to the partial derivative, Shrikumar et al. (2017) define a multiplier as follows:

$$m_{\Delta\eta_i \Delta t} = \frac{C_{\Delta\eta_i \Delta t}}{\Delta t}. \tag{18}$$

In practice, we may not be necessarily interested in contributions of hidden units $\eta_1, \eta_2, ..., \eta_K$. Therefore, the authors instead consider the following definition of multipliers for input features, which is consistent with the summation-to-delta property (Equation 17):

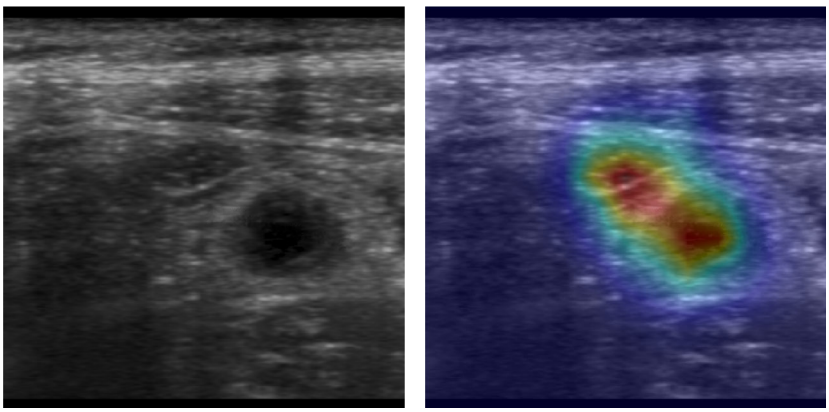
$$m_{\Delta x_i \Delta t} = \sum_j m_{\Delta x_i \Delta \eta_j} m_{\Delta \eta_j \Delta t}, \tag{19}$$

Equation (19) is informally referred to as the chain rule for multipliers. The authors propose several propagation rules for computing $C_{\Delta\eta_i\Delta t}$, which alongside the summation-to-delta and chain rule properties are then used to compute $m_{\Delta x_i \Delta t}$. The choice of propagation rules is not set in stone, and more complex or specialized neural network architectures require adaptations to the original DeepLIFT approach.

SHAP

A framework of *Sh*apley *a*dditive ex*p*lanations (SHAP) (Lundberg & Lee, 2017) builds on Shapley regression values (Lipovetsky & Conklin, 2001) inspired by the game-theoretic concept of Shapley values (Hart, 1989). For the j-th feature, the Shapley regression value at data point $\boldsymbol{x}$ is given by

$$\phi_j(\boldsymbol{x}) = \sum_{\mathcal{S} \subseteq \mathcal{F} \setminus \{j\}} \frac{|\mathcal{S}|!(|\mathcal{F}| - |\mathcal{S}| - 1)!}{|\mathcal{F}|!} \left\{ f_{\mathcal{S}\cup\{j\}}\left(\boldsymbol{x}_{\mathcal{S}\cup\{j\}}\right) - f_{\mathcal{S}}(\boldsymbol{x}_{\mathcal{S}}) \right\}, \tag{20}$$



(a) Rawimage (b) Attribution

FIGURE 5 An example of attribution in medical image classification. (a) A raw appendix ultrasound image from a pediatric patient admitted to a hospital with suspected appendicitis. (b) The corresponding attribution map, overlaid with the raw image, produced using the GradCam method (Selvaraju et al., 2017) for a deep neural network classifier predicting patients' diagnoses. Red color denotes higher attribution values, that is, higher "importance" of pixels, whereas blue color denotes lower values. According to the attribution map, the classifier concentrates on the region around the appendix.

where $\mathcal{F} = \{1,...,p\}$ corresponds to the set of *all* input variables; $\boldsymbol{x}_S$ is a feature vector composed of the components of $\boldsymbol{x}$ that are in $S \subseteq \mathcal{F}$; and $f_S(\cdot)$ is a model trained only on the features from the set S. Intuitively, $\phi_j(\boldsymbol{x})$ quantifies the change in the output of the model resulting from adding the j-th variable to the set of features. Since there are exponentially many subsets of $\mathcal{F} \setminus \{j\}$, in practice, Equation (20) does not have to be evaluated exactly and can be approximated by sampling subsets randomly.

Lundberg and Lee (2017) propose a model-agnostic kernel approximation of Shapley regression values described above. There also exist model-specific implementations of SHAP, for example, for decision trees and gradient boosted decision trees (Lundberg et al., 2020). A compelling advantage of SHAP is the generality of its formulation and elegant connections to statistical regression models and cooperative game theory. Moreover, both LIME and DeepLIFT described before are special cases of the SHAP framework that resort to model-specific approximations of Equation (20).

Follow-up work has explored other explanation methods derived from the concept of Shapley value and cooperative game theory, for example, integrated gradients (Sundararajan et al., 2017), Shapley values for individual neurons (Ghorbani & Zou, 2020), or the least core (Yan & Procaccia, 2021), based on a different solution concept. Rozemberczki et al. (2022) provide an in-depth overview of the cooperative game theory and numerous applications of the Shapley value in machine learning.

Integrated gradients

Sundararajan et al. (2017) introduce another attribution method—integrated gradients (IG). They are motivated by the two following axioms. The sensitivity axiom posits that (i) if an input differs from a baseline in one feature and has a prediction outcome different from the baseline, then the differing variable should be assigned a nonzero attribution and that (ii) if the black-box model $f(\cdot)$ is constant in some variable, then this variable should be given zero attribution. The implementation invariance axiom states that attributions should be identical for two functionally equivalent black-box models. Integrated gradients satisfy point (i) of sensitivity and the implementation invariance.

For the data point $\boldsymbol{x}$, the j-th variable, and baseline $\boldsymbol{x}_0$, the integrated gradient is given by

$$\mathrm{IG}_j^f(\boldsymbol{x}) = (x_j - x_{0,j}) \int_{\alpha=0}^{1} \frac{\partial f(\boldsymbol{x}_0 + \alpha[\boldsymbol{x} - \boldsymbol{x}_0])}{\partial x_j} d\alpha. \tag{21}$$

Observe that $\mathrm{IG}^f(\boldsymbol{x})$ is an integral of gradients along the straight path between $\boldsymbol{x}$ and $\boldsymbol{x}_0$. Similarly to DeppLIFT, integrated gradients defined above satisfy the completeness property: if $f(\cdot)$ is differentiable almost everywhere, $\sum_{j=1}^{p} \mathrm{IG}_j^f(\boldsymbol{x}) = f(\boldsymbol{x}) - f(\boldsymbol{x}_0)$.

Equation (21) can be generalized further by considering a non-straight path between $\boldsymbol{x}$ and $\boldsymbol{x}_0$. Path integrated gradients are then defined for specified paths $\gamma = \left(\gamma_1,...,\gamma_p\right) : [0,1] \to \mathbb{R}^p$ as

$$\mathrm{IG}_j^{f,\gamma}(\boldsymbol{x}) = \int_{\alpha=0}^{1} \frac{\partial f(\gamma(\alpha))}{\partial \gamma_j(\alpha)} \frac{\partial \gamma_j(\alpha)}{\partial \alpha} d\alpha. \tag{22}$$

Path integrated gradients are the *unique* attribution measure that fulfills points (i) and (ii) of sensitivity, implementation invariance, and completeness. Similarly to SHAP, path integrated gradients are rooted in cooperative game theory and correspond to a generalization of Shapley values proposed by Aumann and Shapley (1974) in the context of infinite games.

Among more recent developments, Erion et al. (2021) introduce *expected* gradients (EG), which require fewer hyperparameters than the measure in Equation (21):

$$\mathrm{EG}_j^f(\boldsymbol{x}) = \mathbb{E}_{\boldsymbol{x}_0 \sim \mathcal{D}, \alpha \sim \mathcal{U}(0,1)} \left[(x_j - x_{0,j}) \times \frac{\partial f(\boldsymbol{x}_0 + \alpha[\boldsymbol{x} - \boldsymbol{x}_0])}{\partial x_j} \right], \tag{23}$$

where $\mathcal{D}$ is the reference distribution, for example, $\boldsymbol{x}_0$ could be sampled from the training dataset with replacement, and $\mathcal{U}(0,1)$ is the uniform distribution on the interval $[0,1]$. Observe that rather than using a single reference $\boldsymbol{x}_0$, EG samples multiple references and approximates the integral as expectation. Moreover, Erion et al. (2021) investigate

incorporating attributions into the training process by imposing a prior on the expected gradients of the neural network. Attribution priors (Erion et al., 2021; Ross et al., 2017) facilitate the use of post hoc explanations, such as EG, to make the neural network more *interpretable*, thus, building a connection with approaches described in Section 4.2.

Explaining individual neurons

Bau et al. (2020) investigate the role of individual neurons in discriminative and generative deep networks and demonstrate that a sparse subset of the network's units often contributes the most to the output. Such insights facilitate a better understanding of how representation learning occurs and how high-level concepts emerge within a neural network. Several recent attribution measures have focused on providing more "fine-grained" explanations. In particular, some measures attempt to quantify the importance of individual feature detectors within a neural network corresponding to individual neurons, *aka* units, or whole channels or filters in convolutional networks (Dhamdhere et al., 2019; Leino et al., 2018; Nam et al., 2020; Srinivas & Fleuret, 2019). As a concrete example, Ghorbani and Zou (2020) propose Shapley-value-based importance for individual neurons.

Further remarks

Although attribution methods have become a well-established research topic, their general applicability and usefulness have been scrutinized (Kim et al., 2018; I. E. Kumar et al., 2020; Rudin, 2019). For instance, Rudin (2019) argues that explanation and, especially, attribution methods cannot be entirely faithful to the original black-box model and that attributions do not provide any information about *how* the model works; they instead tell us what the model looks at. I. E. Kumar et al. (2020) criticize Shapley-value-based explanations, such as described above, for their reliance on the additivity axiom (Hart, 1989) and lack of human-groundedness and contrastiveness. Through experiments on semi-synthetic datasets and a user study, Adebayo et al. (2022) demonstrate that post hoc explanation methods, in general, and particularly attributions, often fail to detect spurious correlation captured by the black-box model being explained. Thus, while attribution techniques are an easy-to-use and understand model diagnostic, their effectiveness is limited by the scope of their definitions and assumptions.

4.3.2 | Concept-based explanations

Explanation methods described so far mainly focused on elucidating the relationship between the input variables and the network's output. Arguably, explanations expressed w.r.t. the input space are not always straightforward. For example, individual pixels in an attribution map (Section 4.3.1) are meaningless. The user must associate the map with larger, semantically meaningful regions in the image to make sense of the attribution. Moreover, sometimes attribution methods might fail to explain the relationship clearly, for example, consider the case where the ground-truth explanation for classification is the object's color. One way to address such limitations is to explain the model's predictions in terms of high-level, human-understandable concepts, similar to the concept bottlenecks (Section 4.2.11). For instance, for the medical image in Figure 5, high-level concepts explaining the classification might be the visibility and diameter of the inflamed appendix.

Kim et al. (2018) propose quantitative testing with concept activation vectors (TCAV)—a method for quantifying the influence of a high-level concept on the representations learnt by a neural network post hoc. Let us consider the following decomposition of k-th output unit of a neural network classifier given by $f_k(\boldsymbol{x}) = g_k(\boldsymbol{h}^l(\boldsymbol{x}))$, where $\boldsymbol{h}^l : \mathcal{X} \to \mathbb{R}^{d_l}$ refers to the activation vector of the l-th layer. Given a binary concept $C \in \{0,1\}$, for the layer l of the neural network $f(\cdot)$, input $\boldsymbol{x} \in \mathcal{X}$, and class $y = k$, the conceptual sensitivity (CS) is defined as

$$\mathrm{CS}^f_{C,k,l}(\boldsymbol{x}) = \frac{\partial g_k(\boldsymbol{h}^l(\boldsymbol{x}))}{\partial \boldsymbol{v}^l_C} = \nabla g_k(\boldsymbol{h}^l(\boldsymbol{x}))^T \boldsymbol{v}^l_C, \tag{24}$$

where $\boldsymbol{v}^l_C \in \mathbb{R}^{d_l}$ is a concept activation vector (CAV)—a unit-norm vector orthogonal to the linear decision boundary of the classifier in the output space of $\boldsymbol{h}^l(\cdot)$ trained to differentiate between the categories of the concept C. Notably, to compute the CAV and evaluate conceptual sensitivity, a sample of data points labeled w.r.t. C is required. Consequently, conceptual sensitivities can be aggregated into the TCAV score given by

$$\mathrm{TCAV}^{f}_{C,k,l} = \frac{|\boldsymbol{x} \in \mathcal{S}_k : \mathrm{CS}^{f}_{C,k,l}(\boldsymbol{x}) > 0|}{|\mathcal{S}_k|}, \tag{25}$$

where $\mathcal{S}_k$ is a set of inputs belonging to the k-th class. Intuitively, $\mathrm{TCAV}^{f}_{C,k,l}$ quantifies the proportion of inputs from class k for which the activations of the l-th layer of $f(\cdot)$ are positively influenced by the concept C. The statistic in Equation (25) can be then used for hypothesis testing, for example, to decide if a specific concept has a *significant* influence on the network $f(\cdot)$.

TCAV score (Equation 25) only provides a global explanation, that is, it indicates if concept C influences the classifier across the entire dataset. Schrouff et al. (2021) combine the definition of the conceptual sensitivity (Equation 24) with the integrated gradients (Equation 21) to produce *local* concept-based explanations, referred to as integrated conceptual sensitivity (ICS):

$$\mathrm{ICS}^{f}_{C,k,l}(\boldsymbol{x}) = \left(\boldsymbol{h}^{l}(\boldsymbol{x}) - \boldsymbol{h}_0\right)^{T} \int_{\alpha=0}^{1} \nabla_{\boldsymbol{v}^{l}_{C}} g_k\left(\boldsymbol{h}_0 + \alpha\left[\boldsymbol{h}^{l}(\boldsymbol{x}) - \boldsymbol{h}_0\right]\right) d\alpha, \tag{26}$$

where $\boldsymbol{h}_0 \in \mathbb{R}^{d_l}$ denotes a reference activation vector. Note that, unlike the integrated gradients, ICS performs integration in the activation space of the network and uses the directional derivative, similar to the TCAV (cf. Equation 24). Another limitation of the TCAV is that, similar to concept bottlenecks, the concepts of interest have to be known, and the dataset must be at least partially labeled w.r.t. the concepts. To this end, some works have focused on the *automatic* discovery of concepts from neural network activations, for example, methods introduced by Ghorbani et al. (2019) and Yeh et al. (2020).

4.3.3 | Symbolic metamodels

Section 4.2.10 described symbolic regression as an approach to learning interpretable mathematical expressions from raw data. Similarly to linear models and GAMs in LIME (Section 4.3.1), symbolic regression can be used for surrogate modeling of already learnt opaque predictive models (Alaa & van der Schaar, 2019; Crabbe et al., 2020). For instance, Alaa and van der Schaar (2019) propose an elegant parameterization of the symbolic regression problem (cf. Equation 15) that allows for optimization by gradient descent, in contrast to genetic programming and simulated annealing approaches that search through a discrete solution space.

According to Alaa and van der Schaar (2019), symbolic metamodeling reduces to the following:

$$\min_{g \in \mathfrak{G}} \mathscr{L}(g, f) = \min_{g \in \mathfrak{G}} \int_{\mathcal{X}} (g(\boldsymbol{x}) - f(\boldsymbol{x}))^2 d\boldsymbol{x}, \tag{27}$$

where $\mathscr{L}(\cdot,\cdot)$ is a metamodeling loss, and $\mathfrak{G}$ is a class of succinct mathematical expressions that serve as a surrogate for the black-box model $f(\cdot)$. The authors introduce a parameterization of $\mathfrak{G}$ that makes the optimization problem in Equation (27) "easier": given parameterization $\mathfrak{G} = \{G(\cdot\,;\,\theta) : \theta \in \Theta\}$, the problem above becomes $\min_{\theta \in \Theta} \mathscr{L}(G(\cdot\,;\,\theta), f(\cdot))$.

By Kolmogorov–Arnold superposition theorem (Arnold, 1957; Kolmogorov, 1956), assuming data points $\boldsymbol{x} \in \mathbb{R}^p$, surrogate model $g(\cdot)$ can be rewritten in the following form:

$$g(\boldsymbol{x}) = \sum_{i=0}^{r} g^{\mathrm{out}}_i \left(\sum_{j=1}^{p} g^{\mathrm{in}}_{i,j}(x_j) \right), \tag{28}$$

where $g^{\mathrm{in}}_{i,j}(\cdot)$ and $g^{\mathrm{out}}_i(\cdot)$ are continuous basis functions. Equation (28) encapsulates a fairly broad class of functions. For instance, if $r = 1$, $g(\cdot)$ becomes a GAM (Section 4.2.3). Most importantly, the representation above yields parameterization $G(\boldsymbol{x};\,\theta) = G\left(\boldsymbol{x};\,\left\{g^{\mathrm{in}}_{i,j}\right\}_{i,j}, \left\{g^{\mathrm{out}}_i\right\}_i\right)$. The basis functions themselves can be parameterized by representing them as Meijer G-functions (Meijer, 1936) that are closed under differentiation. The closure property allows searching through $\mathfrak{G}$ efficiently using the gradient descent procedure.

Symbolic metamodeling (Alaa & van der Schaar, 2019; Crabbe et al., 2020) alongside symbolic regression (Jin et al., 2019; Stinstra et al., 2007; Udrescu & Tegmark, 2020) is a compelling alternative to attribution methods (Section 4.3.1), especially when we seek a parsimonious analytical representation of a black-box function. The parameterization proposed by Alaa and van der Schaar (2019) is a helpful reformulation of the problem that benefits from the recent advances in automatic differentiation.

4.3.4 | Counterfactual explanations

In some applications, it might be of paramount importance to provide human-friendly explanations (Carvalho et al., 2019) that are understandable to a broad nonspecialist audience. Techniques discussed so far mainly addressed the question "*Why this prediction was made?*" By contrast, counterfactual explanations try to answer the question "*Why was this prediction made instead of another?*" These techniques produce contrastive and actionable local explanations that can be helpful in a wide range of real-world settings, for example, when suggesting lifestyle changes to a patient to reduce her risks or providing reasons for the low creditworthiness of a company.

Wachter et al. (2017) formalize counterfactual explanations in the context of ML. To find a counterfactual explanation $\boldsymbol{x}'$ for a data point $(\boldsymbol{x},y)$ and a black-box model $f(\cdot)$, the authors propose solving the following optimization problem:

$$\min_{\boldsymbol{x}' \in \mathcal{X}} d(\boldsymbol{x},\boldsymbol{x}') + \lambda \mathscr{L}(f(\boldsymbol{x}'),y'), \tag{29}$$

where $d(\cdot,\cdot)$ is an appropriate distance function; y' is chosen to be meaningfully different from y, for example, y' could represent a desirable classification outcome; the loss $\mathscr{L}(f(\boldsymbol{x}'),y')$ quantifies how "different" the model's output is for $\boldsymbol{x}'$ from the y' chosen, for example, one could use MSE for regression or hinge loss for classification; λ is a parameter controlling the slackness on the constraint $f(\boldsymbol{x}') = y'$. The problem above is loosely reminiscent of generating adversarial perturbations (Moosavi-Dezfooli et al., 2017): perturbations to the original data point $\boldsymbol{x}$ are encouraged to be sparse by penalizing $d(\boldsymbol{x},\boldsymbol{x}')$.

Mothilal et al. (2020) extend the framework above to *multiple diverse* counterfactual explanations. In particular, for a data point $\boldsymbol{x}$, explanations $\boldsymbol{c}_1,\ldots,\boldsymbol{c}_K$ are found by solving the optimization problem below:

$$\min_{\boldsymbol{c}_1,\ldots,\boldsymbol{c}_K} \frac{1}{K}\sum_{k=1}^{K} \mathscr{L}(f(\boldsymbol{c}_k),y') + \frac{\lambda_1}{K}\sum_{k=1}^{K} d(\boldsymbol{x},\boldsymbol{c}_k) - \lambda_2 \det(\boldsymbol{S}), \tag{30}$$

where $S_{k,l} = \frac{1}{1+d(\boldsymbol{c}_k,\boldsymbol{c}_l)}$, and, thus, the term $\det(\boldsymbol{S})$ quantifies diversity among explanations. In addition, Mothilal et al. (2020) propose an array of quantitative evaluation metrics for counterfactual explanation techniques, such as (i) validity quantifying how many of the proposed explanations are actual counterfactuals; (ii) proximity measuring the "closeness" of explanations to the original data point; (iii) sparsity quantifying how sparse the perturbations of $\boldsymbol{x}$ are; and (iv) diversity evaluating how diverse the proposed explanations are.

Counterfactual explanation methods above rely on the gradient descent and, thus, assume that the black-box model $f(\cdot)$ is differentiable. Karimi, Barthe, Balle, and Valera (2020) generalize this framework, introducing model-agnostic counterfactual explanations (MACE). They transform the original optimization problem into a sequence of Boolean satisfiability problems and leverage powerful satisfiability modulo theory solvers to solve these. A significant advantage of MACE is its complete agnosticism to the choice of the black-box model $f(\cdot)$ or distance function $d(\cdot,\cdot)$ and its ability to incorporate additional plausibility constraints that allow injecting domain-specific knowledge.

The problem of counterfactual explanation naturally admits generative modeling as an approach to producing counterfactuals. Recently, several papers have utilized deep generative models (Chang et al., 2019; S. Liu et al., 2019; Mahajan et al., 2019) to solve problems similar to the ones considered by Wachter et al. (2017) and Mothilal et al. (2020). Chang et al. (2019) introduce fill-in the dropout (FIDO) saliency maps based on counterfactual generation with masking for explaining image classifiers. S. Liu et al. (2019) leverage GANs to generate minimal change counterfactual examples for image classifiers. Last but not least, Mahajan et al. (2019) propose a VAE-based counterfactual generative model that focuses on feasibility and preservation of causal constraints with regularization derived from a structural causal model (Pearl, 2010).

Another perspective on counterfactual explanations is algorithmic recourse, surveyed in detail by Karimi, Barthe, Schölkopf, and Valera (2020). Algorithmic recourse focuses on explaining the decisions and recommending further actions to "*individuals who are unfavourably treated by automated decision-making systems*" (Karimi, Barthe, Schölkopf, & Valera, 2020). Karimi, Schölkopf, and Valera (2020) criticize counterfactual explanations for the lack of actionability and provide a causal perspective of algorithmic recourse by considering *interventions* instead of explanations. To avoid infeasible or costly recommendations resulting from naïve counterfactuals, Karimi, Schölkopf, and Valera (2020) propose finding minimal cost structural interventions resulting in a favorable outcome. While this approach certainly offers an exciting and, possibly, more user-centered perspective, the core limitation of algorithmic recourse is the unrealistic assumption of a known causal structure (Karimi, Schölkopf, & Valera, 2020; Karimi, von Kügelgen, Schölkopf, & Valera, 2020).

5 | CONCLUDING REMARKS

Interpretable and explainable machine learning is still a young and active research area. With the recent rapid advances in designing highly performant predictive models and the inevitable infusion of machine learning into different application domains, algorithmic decision-making will have far-reaching consequences. Therefore, algorithms need to be understood and trusted by human end-users. In this overview, we surveyed interpretable machine learning models and explanation methods, described the goals, desiderata, and inductive biases behind these techniques, motivated their relevance in several fields of application, illustrated possible use cases, and discussed their evaluation.

Although a lack of universal and rigorous definitions for interpretability or explainability may seem like an impediment, it might be impossible or even harmful to define interpretability due to the sheer breadth of contexts and applications that call for it. Nevertheless, interpretable and explainable ML could benefit from better empirical research practices like most developing research areas, as many works still rely on purely qualitative or even anecdotal evidence. The development of standardized evaluation criteria and benchmarks could make research efforts reproducible and more focused. Last but not least, meaningful adaptations of the discussed methods to "real-world" machine learning systems and data analysis problems largely remain a matter for the future. For widespread and fruitful use of interpretable and explainable ML, stakeholders need to be involved in the discussion. Interdisciplinary collaboration on equal terms between machine learning researchers and stakeholders from application domains, such as medicine, natural sciences, and law, is the next logical step in the evolution of interpretable and explainable ML.

AUTHOR CONTRIBUTIONS

Ričards Marcinkevičs: Conceptualization (equal); investigation (lead); methodology (lead); software (lead); visualization (lead); writing – original draft (lead); writing – review and editing (supporting). **Julia E. Vogt:** Conceptualization (equal); investigation (supporting); methodology (supporting); project administration (lead); resources (lead); supervision (lead); writing – original draft (supporting); writing – review and editing (lead).

ACKNOWLEDGMENTS

The authors thank Pedro Roig Aparicio and David Niederberger for their help with preparing figures and tables. They are also grateful to Dr. Patricia Reis Wolfertstetter and Dr. Christian Knorr for sharing appendix ultrasound images. Open access funding provided by Eidgenossische Technische Hochschule Zurich.

FUNDING INFORMATION

Ričards Marcinkevičs is supported by the SNSF grant #320038189096.

CONFLICT OF INTEREST STATEMENT

The authors have declared no conflicts of interest for this article.

DATA AVAILABILITY STATEMENT

Data sharing is not applicable to this article as no new data were created or analyzed in this study.

ORCID

Ričards Marcinkevičs https://orcid.org/0000-0001-8901-5062

RELATED WIREs ARTICLES

Causability and explainability of artificial intelligence in medicine
Interpretability of machine learning-based prediction models in healthcare
A historical perspective of explainable artificial intelligence

REFERENCES

Adadi, A., & Berrada, M. (2018). Peeking inside the black-box: A survey on explainable artificial intelligence (XAI). *IEEE Access*, *6*, 52138–52160. https://doi.org/10.1109/access.2018.2870052

Adebayo, J., Muelly, M., Abelson, H., & Kim, B. (2022). Post hoc explanations may be ineffective for detecting unknown spurious correlation. In *International conference on learning representations*. OpenReview.net.

Adel, T., Ghahramani, Z., & Weller, A. (2018). Discovering interpretable representations for both deep generative and discriminative models. In *Proceedings of the 35th international conference on machine learning* (Vol. 80, pp. 50–59). PMLR.

Agarwal, R., Melnick, L., Frosst, N., Zhang, X., Lengerich, B., Caruana, R., & Hinton, G. E. (2021). Neural additive models: Interpretable machine learning with neural nets. In *Advances in neural information processing systems* (Vol. 34, pp. 4699–4711). Curran Associates.

Alaa, A. M., & van der Schaar, M. (2019). Demystifying black-box models with symbolic metamodels. In *Advances in neural information processing systems 32* (pp. 11304–11314). Curran Associates.

Al-Shedivat, M., Dubey, A., & Xing, E. (2020). Contextual explanation networks. *Journal of Machine Learning Research*, *21*(194), 1–44.

Alvarez-Melis, D., & Jaakkola, T. S. (2018). Towards robust interpretability with self-explaining neural networks. In *Advances in neural information processing systems 31: Annual conference on neural information processing systems* (pp. 7786–7795). Curran Associates.

Amir Haeri, M., Ebadzadeh, M. M., & Folino, G. (2017). Statistical genetic programming for symbolic regression. *Applied Soft Computing*, *60*, 447–469. https://doi.org/10.1016/j.asoc.2017.06.050

Ancona, M., Ceolini, E., Öztireli, C., & Gross, M. (2019). Gradient-based attribution methods. In W. Samek, G. Montavon, A. Vedaldi, L. K. Hansen, & K.-R. Müller (Eds.), *Explainable AI: Interpreting, explaining and visualizing deep learning* (pp. 169–191). Springer International Publishing. https://doi.org/10.1007/978-3-030-28954-6

Anjomshoae, S., Najjar, A., Calvaresi, D., & Främling, K. (2019). Explainable agents and robots: Results from a systematic literature review. In *Proceedings of the 18th international conference on autonomous agents and multiagent systems* (pp. 1078–1088). International Foundation for Autonomous Agents and Multiagent Systems.

Arcadu, F., Benmansour, F., Maunz, A., Willis, J., Haskova, Z., & Prunotto, M. (2019). Deep learning algorithm predicts diabetic retinopathy progression in individual patients. *npj Digital Medicine*, *2*(1), 92. https://doi.org/10.1038/s41746-019-0172-3

Arnold, V. I. (1957). On functions of three variables. In *Proceedings of the USSR Academy of Sciences* (pp. 679–681). USSR Academy of Sciences.

Arras, L., Osman, A., & Samek, W. (2020). *Ground truth evaluation of neural network explanations with CLEVR-XAI*. arXiv:2003.072e58.

Aumann, R., & Shapley, L. (1974). *Values of non-atomic games*. Princeton University Press. https://doi.org/10.1515/9781400867080

Azodi, C. B., Tang, J., & Shiu, S.-H. (2020). Opening the black box: Interpretable machine learning for geneticists. *Trends in Genetics*, *36*(6), 442–455. https://doi.org/10.1016/j.tig.2020.03.005

Barber, R. F., & Candès, E. J. (2015). Controlling the false discovery rate via knockoffs. *The Annals of Statistics*, *43*(5), 2055–2085. https://doi.org/10.1214/15-AOS1337

Barocas, S., Hardt, M., & Narayanan, A. (2019). *Fairness and machine learning*. http://www.fairmlbook.org

Barredo Arrieta, A., Díaz-Rodríguez, N., Del Ser, J., Bennetot, A., Tabik, S., Barbado, A., Garcia, S., Gil-Lopez, S., Molina, D., Benjamins, R., Chatila, R., & Herrera, F. (2020). Explainable artificial intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI. *Information Fusion*, *58*, 82–115. https://doi.org/10.1016/j.inffus.2019.12.012

Bau, D., Zhu, J.-Y., Strobelt, H., Lapedriza, A., Zhou, B., & Torralba, A. (2020). Understanding the role of individual units in a deep neural network. *Proceedings of the National Academy of Sciences*, *117*(48), 30071–30078. https://doi.org/10.1073/pnas.1907375117

Benjamini, Y., & Hochberg, Y. (1995). Controlling the false discovery rate: A practical and powerful approach to multiple testing. *Journal of the Royal Statistical Society: Series B (Methodological)*, *57*(1), 289–300. https://doi.org/10.1111/j.2517-6161.1995.tb02031.x

Bhatt, U., Xiang, A., Sharma, S., Weller, A., Taly, A., Jia, Y., Ghosh, J., Puri, R., Moura, J. M. F., & Eckersley, P. (2020). Explainable machine learning in deployment. In *Proceedings of the 2020 conference on fairness, accountability, and transparency* (pp. 648–657). Association for Computing Machinery. https://doi.org/10.1145/3351095.3375624

Biggio, L., Bendinelli, T., Neitz, A., Lucchi, A., & Parascandolo, G. (2021). Neural symbolic regression that scales. In *Proceedings of the 38th international conference on machine learning* (Vol. 139, pp. 936–945). PMLR.

Breiman, L. (2001). Random forests. *Machine Learning*, *45*(1), 5–32. https://doi.org/10.1023/A:1010933404324

Burkart, N., & Huber, M. F. (2021). A survey on the explainability of supervised machine learning. *Journal of Artificial Intelligence Research*, *70*, 245–317. https://doi.org/10.1613/jair.1.12228

Byrne, R. M. J. (2019). Counterfactuals in explainable artificial intelligence (XAI): Evidence from human reasoning. In *Proceedings of the twenty-eighth international joint conference on artificial intelligence*. International Joint Conferences on Artificial Intelligence Organization. https://doi.org/10.24963/ijcai.2019/876

Candès, E., Fan, Y., Janson, L., & Lv, J. (2018). Panning for gold: 'Model-X' knockoffs for high dimensional controlled variable selection. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, *80*(3), 551–577. https://doi.org/10.1111/rssb.12265

Caruana, R., Lou, Y., Gehrke, J., Koch, P., Sturm, M., & Elhadad, N. (2015). Intelligible models for healthcare: Predicting pneumonia risk and hospital 30-day readmission. In *Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 1721–1730). Association for Computing Machinery. https://doi.org/10.1145/2783258.2788613

Carvalho, D. V., Pereira, E. M., & Cardoso, J. S. (2019). Machine learning interpretability: A survey on methods and metrics. *Electronics*, *8*(8), 832. https://doi.org/10.3390/electronics8080832

Chakraborty, S., Tomsett, R., Raghavendra, R., Harborne, D., Alzantot, M., Cerutti, F., Srivastava, M., Preece, A., Julier, S., Rao, R. M., Kelley, T. D., Braines, D., Sensoy, M., Willis, C. J., & Gurram, P. (2017). Interpretability of deep learning models: A survey of results. In *IEEE SmartWorld, ubiquitous intelligence & computing, advanced & trusted computed, scalable computing & communications, cloud & big data computing, internet of people and smart city innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI)*. IEEE. https://doi.org/10.1109/uic-atc.2017.8397411

Chang, C.-H., Caruana, R., & Goldenberg, A. (2021). *NODE-GAM: Neural generalized additive model for interpretable deep learning*. arXiv: 2106.01613.

Chang, C.-H., Creager, E., Goldenberg, A., & Duvenaud, D. (2019). Explaining image classifiers by counterfactual generation. In *International conference on learning representations, ICLR 2019*. OpenReview.net.

Chen, C., & Rudin, C. (2018). An optimization approach to learning falling rule lists. In *International conference on artificial intelligence and statistics, AISTATS 2018* (Vol. 84, pp. 604–612). PMLR.

Chen, X., Duan, Y., Houthooft, R., Schulman, J., Sutskever, I., & Abbeel, P. (2016). InfoGAN: Interpretable representation learning by information maximizing generative adversarial nets. In *Proceedings of the 30th international conference on neural information processing systems* (pp. 2180–2188). Curran Associates.

Chen, Z., Bei, Y., & Rudin, C. (2020). Concept whitening for interpretable image recognition. *Nature Machine Intelligence*, *2*(12), 772–782. https://doi.org/10.1038/s42256-020-00265-z

Cohen, W. W. (1995). Fast effective rule induction. In *Machine learning proceedings 1995* (pp. 115–123). Morgan Kaufmann. https://doi.org/10.1016/B978-1-55860-377-6.50023-2

Confalonieri, R., Coba, L., Wagner, B., & Besold, T. R. (2021). A historical perspective of explainable artificial intelligence. *WIREs Data Mining and Knowledge Discovery*, *11*(1), e1391. https://doi.org/10.1002/widm.1391

Crabbe, J., Zhang, Y., Zame, W., & van der Schaar, M. (2020). Learning outside the black-box: The pursuit of interpretable models. In *Advances in neural information processing systems* (Vol. 33, pp. 17838–17849). Curran Associates.

De Raedt, L. (1999). A perspective on inductive logic programming. In *The logic programming paradigm* (pp. 335–346). Springer.

Dhamdhere, K., Sundararajan, M., & Yan, Q. (2019). How important is a neuron. In *International conference on learning representations*. OpenReview.net.

Dignum, V. (2019). *Responsible artificial intelligence*. Springer International Publishing. https://doi.org/10.1007/978-3-030-30371-6

Doshi-Velez, F., & Kim, B. (2017). *Towards a rigorous science of interpretable machine learning*. arXiv:1702.08608.

Du, M., Liu, N., & Hu, X. (2019). Techniques for interpretable machine learning. *Communications of the ACM*, *63*(1), 68–77. https://doi.org/10.1145/3359786

Dziugaite, G. K., Ben-David, S., & Roy, D. M. (2020). *Enforcing interpretability and its statistical impacts: Trade-offs between accuracy and interpretability*. arXiv:2010.13764.

Emmert-Streib, F., Yli-Harja, O., & Dehmer, M. (2020). Explainable artificial intelligence and machine learning: A reality rooted perspective. *WIREs Data Mining and Knowledge Discovery*, *10*(6), e1368. https://doi.org/10.1002/widm.1368

Erion, G., Janizek, J. D., Sturmfels, P., Lundberg, S. M., & Lee, S.-I. (2021). Improving performance of deep learning models with axiomatic attribution priors and expected gradients. *Nature Machine Intelligence*, *3*(7), 620–631. https://doi.org/10.1038/s42256-021-00343-w

Fasiolo, M., Nedellec, R., Goude, Y., & Wood, S. N. (2019). Scalable visualization methods for modern generalized additive models. *Journal of Computational and Graphical Statistics*, *29*(1), 78–86. https://doi.org/10.1080/10618600.2019.1629942

Feng, J., & Simon, N. (2017). *Sparse-input neural networks for high-dimensional nonparametric regression and classification*. arXiv: 1711.07592.

Friedman, J. H. (2001). Greedy function approximation: A gradient boosting machine. *The Annals of Statistics*, *29*(5), 1189–1232. https://doi.org/10.1214/aos/1013203451

Friedman, J. H., & Popescu, B. E. (2008). Predictive learning via rule ensembles. *The Annals of Applied Statistics*, *2*(3), 916–954. https://doi.org/10.1214/07-AOAS148

Fujii, K., Takeishi, N., Tsutsui, K., Fujioka, E., Nishiumi, N., Tanaka, R., Fukushiro, M., Ide, K., Kohno, H., Yoda, K., Takahashi, S., Hiryu, S., & Kawahara, Y. (2021). Learning interaction rules from multi-animal trajectories via augmented behavioral models. *Advances in Neural Information Processing Systems*, *34*, 11108–11122.

Ghassemi, M., Oakden-Rayner, L., & Beam, A. L. (2021). The false hope of current approaches to explainable artificial intelligence in health care. *The Lancet Digital Health*, *3*(11), e745–e750. https://doi.org/10.1016/s2589-7500(21)00208-9

Ghorbani, A., Wexler, J., Zou, J. Y., & Kim, B. (2019). Towards automatic concept-based explanations. In *Advances in neural information processing systems* (Vol. 32). Curran Associates.

Ghorbani, A., & Zou, J. Y. (2020). Neuron Shapley: Discovering the responsible neurons. In *Advances in neural information processing systems* (Vol. 33, pp. 5922–5932). Curran Associates.

Gilpin, L. H., Bau, D., Yuan, B. Z., Bajwa, A., Specter, M., & Kagal, L. (2018). Explaining explanations: An overview of interpretability of machine learning. In *2018 IEEE 5th international conference on data science and advanced analytics (DSAA)*. IEEE. https://doi.org/10.1109/dsaa.2018.00018

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. MIT Press. http://www.deeplearningbook.org

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., & Bengio, Y. (2014). Generative adversarial nets. In *Advances in neural information processing systems* (Vol. 27, pp. 2672–2680). Curran Associates.

Guerguiev, J., Lillicrap, T. P., & Richards, B. A. (2017). Towards deep learning with segregated dendrites. *eLife*, *6*, e22901. https://doi.org/10.7554/eLife.22901

Guidotti, R., Monreale, A., Ruggieri, S., Turini, F., Giannotti, F., & Pedreschi, D. (2019). A survey of methods for explaining black box models. *ACM Computing Surveys*, *51*(5), 1–42. https://doi.org/10.1145/3236009

Gunning, D., & Aha, D. (2019). DARPA's explainable artificial intelligence (XAI) program. *AI Magazine*, *40*(2), 44–58. https://doi.org/10.1609/aimag.v40i2.2850

Hart, S. (1989). Shapley value. In J. Eatwell, M. Milgate, & P. Newman (Eds.), *Game theory* (pp. 210–216). Palgrave Macmillan UK. https://doi.org/10.1007/978-1-349-20181-5

Hastie, T., & Tibshirani, R. (1986). Generalized additive models. *Statistical Science*, *1*(3), 297–310. https://doi.org/10.1214/ss/1177013604

Hastie, T., & Tibshirani, R. (1993). Varying-coefficient models. *Journal of the Royal Statistical Society. Series B (Methodological)*, *55*(4), 757–796. https://doi.org/10.1111/j.2517-6161.1993.tb01939.x

Hastie, T., Tibshirani, R., & Friedman, J. (2009). High-dimensional problems: $p \gg N$. In *The elements of statistical learning* (pp. 649–698). Springer. https://doi.org/10.1007/978-0-387-84858-7

Higgins, I., Matthey, L., Pal, A., Burgess, C., Glorot, X., Botvinick, M., Mohamed, S., & Lerchner, A. (2016). β-VAE: Learning basic visual concepts with a constrained variational framework. In *International conference on learning representations*. OpenReview.net.

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, *9*(8), 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735

Hohman, F., Srinivasan, A., & Drucker, S. M. (2019). TeleGam: Combining visualization and verbalization for interpretable machine learning. In *IEEE visualization conference (VIS)*. IEEE. https://doi.org/10.1109/visual.2019.8933695

Holzinger, A., Langs, G., Denk, H., Zatloukal, K., & Müller, H. (2019). Causability and explainability of artificial intelligence in medicine. *WIREs Data Mining and Knowledge Discovery*, *9*(4), e1312. https://doi.org/10.1002/widm.1312

Jain, S., & Wallace, B. C. (2019). Attention is not explanation. In *Proceedings of the 2019 conference of the North American chapter of the Association for Computational Linguistics: Human language technologies* (pp. 3543–3556). Association for Computational Linguistics. https://doi.org/10.18653/v1/N19-1357

Jesus, S., Belém, C., Balayan, V., Bento, J., Saleiro, P., Bizarro, P., & Gama, J. (2021). How can I choose an explainer? An application-grounded evaluation of post-hoc explanations. In *Proceedings of the 2021 ACM conference on fairness, accountability, and transparency* (pp. 805–815). Association for Computing Machinery. https://doi.org/10.1145/3442188.3445941

Jin, Y., Fu, W., Kang, J., Guo, J., & Guo, J. (2019). *Bayesian symbolic regression*. arXiv:1910.08892.

Johnson, J., Hariharan, B., van der Maaten, L., Fei-Fei, L., Zitnick, C. L., & Girshick, R. B. (2017). CLEVR: A diagnostic dataset for compositional language and elementary visual reasoning. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)* (pp. 1988–1997). IEEE. https://doi.org/10.1109/CVPR.2017.215

Jordon, J., Yoon, J., & van der Schaar, M. (2019). KnockoffGAN: Generating knockoffs for feature selection using generative adversarial networks. In *International conference on learning representations*. OpenReview.net

Karimi, A.-H., Barthe, G., Balle, B., & Valera, I. (2020). Model-agnostic counterfactual explanations for consequential decisions. In *The 23rd international conference on artificial intelligence and statistics, AISTATS* (Vol. 108, pp. 895–905). PMLR.

Karimi, A.-H., Barthe, G., Schölkopf, B., & Valera, I. (2020). *A survey of algorithmic recourse: Definitions, formulations, solutions, and prospects*. arXiv:2010.04050.

Karimi, A.-H., Schölkopf, B., & Valera, I. (2020). *Algorithmic recourse: From counterfactual explanations to interventions*. arXiv:2002.06278.

Karimi, A.-H., von Kügelgen, J., Schölkopf, B., & Valera, I. (2020). Algorithmic recourse under imperfect causal knowledge: A probabilistic approach. In *34th conference on neural information processing systems (NeurIPS)*. Curran Associates.

Kelley, D. R., Reshef, Y. A., Bileschi, M., Belanger, D., McLean, C. Y., & Snoek, J. (2018). Sequential regulatory activity prediction across chromosomes with convolutional neural networks. *Genome Research*, *28*(5), 739–750. https://doi.org/10.1101/gr.227819.117

Khanna, S., & Tan, V. Y. F. (2020). Economy statistical recurrent units for inferring nonlinear Granger causality. In *8th international conference on learning representations*. OpenReview.net.

Kim, B., Khanna, R., & Koyejo, O. (2016). Examples are not enough, learn to criticize! Criticism for interpretability. In *Proceedings of the 30th international conference on neural information processing systems* (pp. 2288–2296). Curran Associates.

Kim, B., Rudin, C., & Shah, J. (2014). The Bayesian case model: A generative approach for case-based reasoning and prototype classification. In *Proceedings of the 27th international conference on neural information processing systems* (pp. 1952–1960). MIT Press.

Kim, B., Wattenberg, M., Gilmer, J., Cai, C., Wexler, J., Viegas, F., & Sayres, R. (2018). Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (TCAV). In *Proceedings of the 35th international conference on machine learning* (Vol. 80, pp. 2668–2677). PMLR.

Kingma, D. P., & Welling, M. (2014). Auto-encoding variational Bayes. In *2nd international conference on learning representations*. ICLR.

Kingma, D. P., & Welling, M. (2019). An introduction to variational autoencoders. *Foundations and Trends in Machine Learning*, *12*(4), 307–392. https://doi.org/10.1561/2200000056

Knaus, W. A., Draper, E. A., Wagner, D. P., & Zimmerman, J. E. (1985). APACHE II: A severity of disease classification system. *Critical Care Medicine*, *13*(10), 818–829.

Koh, P. W., Nguyen, T., Tang, Y. S., Mussmann, S., Pierson, E., Kim, B., & Liang, P. (2020). Concept bottleneck models. In *Proceedings of the 37th international conference on machine learning* (Vol. 119, pp. 5338–5348). PMLR.

Kolmogorov, A. N. (1956). On the representation of continuous functions of several variables by superpositions of continuous functions of a smaller number of variables. In *Proceedings of the USSR Academy of Sciences* (pp. 179–182). USSR Academy of Sciences.

Krause, A., & Golovin, D. (2014). Submodular function maximization. In L. Bordeaux, Y. Hamadi, & P. Kohli (Eds.), *Tractability: Practical approaches to hard problems* (pp. 71–104). Cambridge University Press. https://doi.org/10.1017/CBO9781139177801.004

Kumar, I. E., Venkatasubramanian, S., Scheidegger, C., & Friedler, S. (2020). Problems with Shapley-value-based explanations as feature importance measures. In *Proceedings of the 37th international conference on machine learning* (Vol. 119, pp. 5491–5500). PMLR.

Kumar, N., Berg, A. C., Belhumeur, P. N., & Nayar, S. K. (2009). Attribute and simile classifiers for face verification. In *12th international conference on computer vision*. IEEE. https://doi.org/10.1109/iccv.2009.5459250

Lampert, C. H., Nickisch, H., & Harmeling, S. (2009). Learning to detect unseen object classes by between-class attribute transfer. In *Conference on computer vision and pattern recognition*. IEEE. https://doi.org/10.1109/cvpr.2009.5206594

Larson, J., Mattu, S., Kirchner, L., & Angwin, J. (2016). *How we analyzed the COMPAS recidivism algorithm*. https://www.propublica.org/article/how-we-analyzed-the-compas-recidivism-algorithm

LeCun, Y., Cortes, C., & Burges, C. (2010). *MNIST handwritten digit database*. ATT Labs. http://yann.lecun.com/exdb/mnist

Leino, K., Sen, S., Datta, A., Fredrikson, M., & Li, L. (2018). Influence-directed explanations for deep convolutional networks. In *IEEE international test conference (ITC)*. IEEE. https://doi.org/10.1109/test.2018.8624792

Li, L., & Wang, Y. (2019). *Manifold: A model-agnostic visual debugging tool for machine learning at Uber*. https://eng.uber.com/manifold/

Li, Y., Swersky, K., & Zemel, R. (2015). Generative moment matching networks. In *Proceedings of the 32nd international conference on machine learning* (Vol. 37, pp. 1718–1727). PMLR.

Linardatos, P., Papastefanopoulos, V., & Kotsiantis, S. (2020). Explainable AI: A review of machine learning interpretability methods. *Entropy*, *23*(1), 18. https://doi.org/10.3390/e23010018

Lipovetsky, S., & Conklin, M. (2001). Analysis of regression in game theory approach. *Applied Stochastic Models in Business and Industry*, *17*(4), 319–330. https://doi.org/10.1002/asmb.446

Lipton, Z. C. (2018). The mythos of model interpretability. *Queue*, *16*(3), 31–57. https://doi.org/10.1145/3236386.3241340

Liu, K., Sadoune, N., Rao, N., Greitemann, J., & Pollet, L. (2021). Revealing the phase diagram of Kitaev materials by machine learning: Cooperation and competition between spin liquids. *Physical Review Research*, *3*(2), 023016. https://doi.org/10.1103/physrevresearch.3.023016

Liu, S., Kailkhura, B., Loveland, D., & Han, Y. (2019). Generative counterfactual introspection for explainable deep learning. In *2019 IEEE global conference on signal and information processing (GlobalSIP)*. IEEE. https://doi.org/10.1109/globalsip45357.2019.8969491

Liu, Y., Jain, A., Eng, C., Way, D. H., Lee, K., Bui, P., Kanada, K., de Oliveira Marinho, G., Gallegos, J., Gabriele, S., Gupta, V., Singh, N., Natarajan, V., Hofmann-Wellenhof, R., Corrado, G. S., Peng, L. H., Webster, D. R., Ai, D., Huang, S. J., & Coz, D. (2020). A deep learning system for differential diagnosis of skin diseases. *Nature Medicine*, *26*(6), 900–908. https://doi.org/10.1038/s41591-020-0842-3

Locatello, F., Bauer, S., Lucic, M., Raetsch, G., Gelly, S., Schölkopf, B., & Bachem, O. (2019). Challenging common assumptions in the unsupervised learning of disentangled representations. In *Proceedings of the 36th international conference on machine learning* (Vol. 97, pp. 4114–4124). PMLR.

Loh, W.-Y. (2011). Classification and regression trees. *WIREs Data Mining and Knowledge Discovery*, *1*(1), 14–23. https://doi.org/10.1002/widm.8

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). Association for Computing Machinery. https://doi.org/10.1145/2339530.2339556

Lu, Y. Y., Fan, Y., Lv, J., & Noble, W. S. (2018). DeepPINK: Reproducible feature selection in deep neural networks. In *Proceedings of the 32nd international conference on neural information processing systems* (pp. 8690–8700). Curran Associates.

Lucas, J., Carvalho, C., Wang, Q., Bild, A., Nevins, J. R., West, M., & Vannucci, M. (2006). Sparse statistical modelling in gene expression genomics. In *Bayesian inference for gene expression and proteomics* (pp. 155–176). Cambridge University Press. https://doi.org/10.1017/CBO9780511584589.009

Lundberg, S. M., Erion, G., Chen, H., DeGrave, A., Prutkin, J. M., Nair, B., Katz, R., Himmelfarb, J., Bansal, N., & Lee, S.-I. (2020). From local explanations to global understanding with explainable AI for trees. *Nature Machine Intelligence*, *2*(1), 56–67. https://doi.org/10.1038/s42256-019-0138-9

Lundberg, S. M., & Lee, S.-I. (2017). A unified approach to interpreting model predictions. In *Advances in neural information processing systems 30* (pp. 4765–4774). Curran Associates.

MacDonald, C. M., & Atwood, M. E. (2013). Changing perspectives on evaluation in HCI: Past, present, and future. In *CHI'13 extended abstracts on human factors in computing systems* (pp. 1969–1978). ACM. https://doi.org/10.1145/2468356.2468714

Mahajan, D., Tan, C., & Sharma, A. (2019). *Preserving causal constraints in counterfactual explanations for machine learning classifiers*. arXiv: 1912.03277.

Marcinkevics, R., Reis Wolfertstetter, P., Wellmann, S., Knorr, C., & Vogt, J. E. (2021). Using machine learning to predict the diagnosis, management and severity of pediatric appendicitis. *Frontiers in Pediatrics*, *9*, 662183. https://doi.org/10.3389/fped.2021.662183

Marcinkevics, R., & Vogt, J. E. (2021). Interpretable models for Granger causality using self-explaining neural networks. In *9th international conference on learning representations, ICLR 2021*. OpenReview.net.

Marcos, D., Fong, R., Lobry, S., Flamary, R., Courty, N., & Tuia, D. (2021). Contextual semantic interpretability. In *Computer vision—ACCV 2020* (pp. 351–368). Springer International Publishing. https://doi.org/10.1007/978-3-030-69538-522

Mason, L., Baxter, J., Bartlett, P., & Frean, M. (1999). Boosting algorithms as gradient descent. In *Proceedings of the 12th international conference on neural information processing systems* (pp. 512–518). MIT Press.

McKay, B. (1995). Using a tree structured genetic algorithm to perform symbolic regression. In *1st international conference on genetic algorithms in engineering systems: Innovations and applications (GALESIA)*. IEEE. https://doi.org/10.1049/cp:19951096

Meijer, C. S. (1936). Über Whittakersche bzw. Besselsche Funktionen und deren Produkte. In *Nieuw archief voor wiskunde* (pp. 10–39). Centrum Wiskunde & Informatica.

Miller, T. (2019). Explanation in artificial intelligence: Insights from the social sciences. *Artificial Intelligence*, *267*, 1–38. https://doi.org/10.1016/j.artint.2018.07.007

Molnar, C. (2020). Interpretable machine learning. In *A guide for making black box models explainable*. Independently published. https://christophm.github.io/interpretable-ml-book/

Moosavi-Dezfooli, S.-M., Fawzi, A., Fawzi, O., & Frossard, P. (2017). Universal adversarial perturbations. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*. IEEE.

Moraffah, R., Karami, M., Guo, R., Raglin, A., & Liu, H. (2020). Causal interpretability for machine learning—problems, methods and evaluation. *ACM SIGKDD Explorations Newsletter*, *22*(1), 18–33. https://doi.org/10.1145/3400051.3400058

Mothilal, R. K., Sharma, A., & Tan, C. (2020). Explaining machine learning classifiers through diverse counterfactual explanations. In *Proceedings of the 2020 conference on fairness, accountability, and transparency* (pp. 607–617). Association for Computing Machinery. https://doi.org/10.1145/3351095.3372850

Müller, H., & Holzinger, A. (2021). Kandinsky patterns. *Artificial Intelligence*, *300*, 103546. https://doi.org/10.1016/j.artint.2021.103546

Murdoch, W. J., Singh, C., Kumbier, K., Abbasi-Asl, R., & Yu, B. (2019). Definitions, methods, and applications in interpretable machine learning. *Proceedings of the National Academy of Sciences*, *116*(44), 22071–22080. https://doi.org/10.1073/pnas.1900654116

Nam, W.-J., Gur, S., Choi, J., Wolf, L., & Lee, S.-W. (2020). Relative attributing propagation: Interpreting the comparative contributions of individual units in deep neural networks. *Proceedings of the AAAI Conference on Artificial Intelligence*, *34*(3), 2501–2508.

Nauta, M., Bucur, D., & Seifert, C. (2019). Causal discovery with attention-based convolutional neural networks. *Machine Learning and Knowledge Extraction*, *1*, 312–340. https://doi.org/10.3390/make1010019

Nauta, M., Trienes, J., Pathak, S., Nguyen, E., Peters, M., Schmitt, Y., Schlötterer, J., van Keulen, M., & Seifert, C. (2022). *From anecdotal evidence to quantitative evaluation methods: A systematic review on evaluating explainable AI*. arXiv:2201.08164.

Nelder, J. A., & Wedderburn, R. W. M. (1972). Generalized linear models. *Journal of the Royal Statistical Society: Series A (General)*, *135*(3), 370–384. https://doi.org/10.2307/2344614

Nogueira, A. R., Pugnana, A., Ruggieri, S., Pedreschi, D., & Gama, J. (2022). Methods and tools for causal discovery and causal inference. *WIREs Data Mining and Knowledge Discovery.*, *12*, e1449. https://doi.org/10.1002/widm.1449

Otte, C. (2013). Safe and interpretable machine learning: A methodological review. In *Computational intelligence in intelligent data analysis* (pp. 111–122). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-32378-28

Paolacci, G., Chandler, J., & Ipeirotis, P. G. (2010). Running experiments on Amazon mechanical turk. *Judgment and Decision Making*, *5*(5), 411–419.

Papernot, N., McDaniel, P., Sinha, A., & Wellman, M. P. (2018). SoK: Security and privacy in machine learning. In *2018 IEEE European symposium on security and privacy (EuroS&p)*. IEEE. https://doi.org/10.1109/eurosp.2018.00035

Parsa, A. B., Movahedi, A., Taghipour, H., Derrible, S., & Mohammadian, A. K. (2020). Toward safer highways, application of XGBoost and SHAP for real-time accident detection and feature analysis. *Accident Analysis & Prevention*, *136*, 105405. https://doi.org/10.1016/j.aap.2019.105405

Pearl, J. (2010). An introduction to causal inference. *The. International Journal of Biostatistics*, *6*(2), 7. https://doi.org/10.2202/1557-4679.1203

Petersen, B. K., Larma, M. L., Mundhenk, T. N., Santiago, C. P., Kim, S. K., & Kim, J. T. (2021). Deep symbolic regression: Recovering mathematical expressions from data via risk-seeking policy gradients. In *International conference on learning representations*. OpenReview.net.

Pimentel, T., McCarthy, A. D., Blasi, D., Roark, B., & Cotterell, R. (2019). Meaning to form: Measuring systematicity as information. In *Proceedings of the 57th annual meeting of the Association for Computational Linguistics*. Association for Computational Linguistics. doi:10.18653/v1/p19-1171

Puiutta, E., & Veith, E. M. S. P. (2020). Explainable reinforcement learning: A survey. In *Lecture notes in computer science* (pp. 77–95). Springer International Publishing. https://doi.org/10.1007/978-3-030-57321-85

Raghu, M., & Schmidt, E. (2020). *A survey of deep learning for scientific discovery*. arXiv:2003.11755.

Ravazzi, C., Tempo, R., & Dabbene, F. (2018). Learning influence structure in sparse social networks. *IEEE Transactions on Control of Network Systems*, *5*(4), 1976–1986. https://doi.org/10.1109/TCNS.2017.2781367

Ravikumar, P., Liu, H., Lafferty, J., & Wasserman, L. (2007). SpAM: Sparse additive models. In *Advances in neural information processing systems*. Curran Associates.

Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). "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). Association for Computing Machinery. https://doi.org/10.1145/2939672.2939778

Roig Aparicio, P., Marcinkevičs, R., Reis Wolfertstetter, P., Wellmann, S., Knorr, C., & Vogt, J. E. (2021). Learning medical risk scores for pediatric appendicitis. In *20th IEEE international conference on machine learning and applications (ICMLA)* (pp. 1507–1512). IEEE https://doi.org/10.1109/ICMLA52953.2021.00243

Romano, Y., Sesia, M., & Candès, E. (2019). Deep knockoffs. *Journal of the American Statistical Association*, *115*, 1861–1872. https://doi.org/10.1080/01621459.2019.1660174

Roscher, R., Bohn, B., Duarte, M. F., & Garcke, J. (2020). Explainable machine learning for scientific insights and discoveries. *IEEE Access*, *8*, 42200–42216. https://doi.org/10.1109/access.2020.2976199

Ross, A. S., Hughes, M. C., & Doshi-Velez, F. (2017). Right for the right reasons: Training differentiable models by constraining their explanations. In *Proceedings of the 26th international joint conference on artificial intelligence*. International Joint Conferences on Artificial Intelligence Organization.https://doi.org/10.24963/ijcai.2017/371

Rozemberczki, B., Watson, L., Bayer, P., Yang, H.-T., Kiss, O., Nilsson, S., & Sarkar, R. (2022). *The Shapley value in machine learning*. arXiv: 2202.05594.

Rubin, D. B. (2005). Causal inference using potential outcomes: Design, modeling, decisions. *Journal of the American Statistical Association*, *100*(469), 322–331. https://doi.org/10.1198/016214504000001880

Rudin, C. (2019). Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature Machine Intelligence*, *1*(5), 206–215. https://doi.org/10.1038/s42256-019-0048-x

Rudin, C., Chen, C., Chen, Z., Huang, H., Semenova, L., & Zhong, C. (2022). Interpretable machine learning: Fundamental principles and 10 grand challenges. *Statistics Surveys*, *16*, 1–85 https://doi.org/10.1214/21-ss133

Schrouff, J., Baur, S., Hou, S., Mincu, D., Loreaux, E., Blanes, R., Wexler, J., Karthikesalingam, A., & Kim, B. (2021). *Best of both worlds: Local and global explanations with human-understandable concepts*. arXiv:2106.08641.

Schütt, K. T., Gastegger, M., Tkatchenko, A., Müller, K.-R., & Maurer, R. J. (2019). Unifying machine learning and quantum chemistry with a deep neural network for molecular wavefunctions. *Nature Communications*, *10*(1), 5024. https://doi.org/10.1038/s41467-019-12875-2

Schwab, P., Miladinovic, D., & Karlen, W. (2019). Granger-causal attentive mixtures of experts: Learning important features with neural networks. *Proceedings of the AAAI Conference on Artificial Intelligence*, *33*(1), 4846–4853.

Seeliger, A., Pfaff, M., & Krcmar, H. (2019). *Semantic web technologies for explainable machine learning models: A literature review*. PROFILES/SEMEXISWC.

Segal, U. (1994). A sufficient condition for additively separable functions. *Journal of Mathematical Economics*, *23*(3), 295–303. https://doi.org/10.1016/0304-4068(94)90009-4

Selvaraju, R. R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., & Batra, D. (2017). Grad-CAM: Visual explanations from deep networks via gradient-based localization. In *IEEE international conference on computer vision (ICCV)*. IEEE. https://doi.org/10.1109/iccv.2017.74

Semenova, L., Rudin, C., & Parr, R. (2019). A study in Rashomon curves and volumes: *A new perspective on generalization and model simplicity in machine learning*. arXiv:1908.01755.

Serrano, S., & Smith, N. A. (2019). Is attention interpretable. In *Proceedings of the 57th annual meeting of the Association for Computational Linguistics* (pp. 2931–2951). Association for Computational Linguistics. doi:10.18653/v1/P19-1282

Servén, D., & Brummitt, C. (2018). *pyGAM: Generalized additive models in Python*. Zenodo. https://doi.org/10.5281/zenodo.1208723

Shrikumar, A., Greenside, P., & Kundaje, A. (2017). Learning important features through propagating activation differences. In *Proceedings of the 34th international conference on machine learning* (pp. 3145–3153). JMLR.

Simon, N., Friedman, J., Hastie, T., & Tibshirani, R. (2013). A sparse-group Lasso. *Journal of Computational and Graphical Statistics*, *22*(2), 231–245. https://doi.org/10.1080/10618600.2012.681250

Srinivas, S., & Fleuret, F. (2019). Full-gradient representation for neural network visualization. In *Advances in neural information processing systems* (pp. 4124–4133). Curran Associates.

Stiglic, G., Kocbek, P., Fijacko, N., Zitnik, M., Verbert, K., & Cilar, L. (2020). Interpretability of machine learning-based prediction models in healthcare. *WIREs Data Mining and Knowledge Discovery*, *10*(5), e1379. https://doi.org/10.1002/widm.1379

Stinstra, E., Rennen, G., & Teeuwen, G. (2007). Metamodeling by symbolic regression and Pareto simulated annealing. *Structural and Multidisciplinary Optimization*, *35*(4), 315–326. https://doi.org/10.1007/s00158-007-0132-4

Sundararajan, M., Taly, A., & Yan, Q. (2017). Axiomatic attribution for deep networks. In *Proceedings of the 34th international conference on machine learning* (Vol. 70, pp. 3319–3328). JMLR.

Taeb, A., Ruggeri, N., Schnuck, C., & Yang, F. (2022). *Provable concept learning for interpretable predictions using variational inference*. arXiv: 2204.00492.

Tank, A., Covert, I., Foti, N., Shojaie, A., & Fox, E. B. (2021). Neural Granger causality. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *44*, 1–4279. https://doi.org/10.1109/TPAMI.2021.3065601

Tjoa, E., & Guan, C. (2021). A survey on explainable artificial intelligence (XAI): Toward medical XAI. *IEEE Transactions on Neural Networks and Learning Systems*, *32*(11), 4793–4813. https://doi.org/10.1109/tnnls.2020.3027314

Tschannen, M., Bachem, O., & Lucic, M. (2018). *Recent advances in autoencoder-based representation learning*. arXiv:1812.05069.

Udrescu, S.-M., & Tegmark, M. (2020). AI Feynman: A physics-inspired method for symbolic regression. *Science Advances*, *6*(16), eaay2631. https://doi.org/10.1126/sciadv.aay2631

Ustun, B., & Rudin, C. (2015). Supersparse linear integer models for optimized medical scoring systems. *Machine Learning*, *102*(3), 349–391. https://doi.org/10.1007/s10994-015-5528-6

Ustun, B., & Rudin, C. (2017). Optimized risk scores. In *Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 1125–1134). Association for Computing Machinery. https://doi.org/10.1145/3097983.3098161

Utkin, L. V., Satyukov, E. D., & Konstantinov, A. V. (2022). SurvNAM: The machine learning survival model explanation. *Neural Networks*, *147*, 81–102. https://doi.org/10.1016/j.neunet.2021.12.015

Valdes, G., Arbelo, W., Interian, Y., & Friedman, J. H. (2021). *Lockout: Sparse regularization of neural networks*. arXiv:2107.07160.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. (2017). Attention is all you need. In *Advances in neural information processing systems 30* (pp. 5998–6008). Curran Associates.

Verma, S., Dickerson, J., & Hines, K. (2020). *Counterfactual explanations for machine learning: A review*. arXiv:2010.10596.

Vincent, J. L., Moreno, R., Takala, J., Willatts, S., De Mendonça, A., Bruining, H., Reinhart, C. K., Suter, P. M., & Thijs, L. G. (1996). The SOFA (sepsisrelated organ failure assessment) score to describe organ dysfunction/failure. On behalf of the working group on sepsis-related problems of the European Society of Intensive Care Medicine. *Intensive Care Medicine*, *22*(7), 707–710. https://doi.org/10.1007/bf01709751

Voigt, P., & von dem Bussche, A. (2017). *The EU general data protection regulation (GDPR)*. Springer International Publishing. https://doi.org/10.1007/978-3-319-57959-7

Wachter, S., Mittelstadt, B., & Russell, C. (2017). Counterfactual explanations without opening the black box: Automated decisions and the GDPR. *SSRN Electronic Journal*., *31*, 2018. https://doi.org/10.2139/ssrn.3063289

Wang, F., & Rudin, C. (2015a). *Causal falling rule lists*. arXiv:1510.05189.

Wang, F., & Rudin, C. (2015b). Falling rule lists. In *Proceedings of the eighteenth international conference on artificial intelligence and statistics* (Vol. 38, pp. 1013–1022). PMLR.

Wang, Z. J., Kale, A., Nori, H., Stella, P., Nunnally, M., Chau, D. H., Vorvoreanu, M., Vaughan, J. W., & Caruana, R. (2021). *GAM changer: Editing generalized additive models with interactive visualization*. arXiv:2112.03245.

Watson, D. S. (2021). Interpretable machine learning for genomics. *Human Genetics*, *141*, 1499–1513. https://doi.org/10.1007/s00439-021-02387-9

Xu, G., Duong, T. D., Li, Q., Liu, S., & Wang, X. (2020). *Causality learning: A new perspective for interpretable machine learning*. arXiv:2006.16789.

Xu, S., Bu, Z., Chaudhari, P., & Barnett, I. J. (2022). *Sparse neural additive model: Interpretable deep learning with feature selection via group sparsity*. arXiv:2202.12482.

Yan, T., & Procaccia, A. D. (2021). If you like Shapley then you'll love the core. *Proceedings of the AAAI Conference on Artificial Intelligence*, *35*(6), 5751–5759.

Yeh, C.-K., Kim, B., Arik, S., Li, C.-L., Pfister, T., & Ravikumar, P. (2020). On completeness-aware concept-based explanations in deep neural networks. In *Advances in neural information processing systems* (Vol. 33, pp. 20554–20565). Curran Associates.

Zhang, Y., & Chen, X. (2020). Explainable recommendation: A survey and new perspectives. *Foundations and Trends in Information Retrieval*, *14*(1), 1–101. https://doi.org/10.1561/1500000066

Zou, H., & Hastie, T. (2005). Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, *67*(2), 301–320. https://doi.org/10.1111/j.1467-9868.2005.00503.x

How to cite this article: Marcinkevičs, R., & Vogt, J. E. (2023). Interpretable and explainable machine learning: A methods-centric overview with concrete examples. *WIREs Data Mining and Knowledge Discovery*, *13*(3), e1493. https://doi.org/10.1002/widm.1493