Explaining Explainability: Interpretable machine learning for the behavioral sciences

Brendan Kennedy[1], Nils Karl Reimer[2], and Morteza Dehghani[1,2]

[1]Department of Computer Science, University of Southern California

[2]Department of Psychology, University of Southern California

Author Note

Correspondence regarding this article should be addressed to Brendan Kennedy, btkenned@usc.edu, 362 S. McClintock Ave, Los Angeles, CA 90089-161. This research was sponsored by NSF CAREER BCS-1846531 to MD.

Abstract

Predictive data modeling is a critical practice for the behavioral sciences; however, it is under-practiced in part due to the incorrect view that machine learning (ML) models are "black boxes," unable to be used for inferential purposes. In this work, we present an argument for the adoption of techniques from interpretable Machine Learning (ML) by behavioral scientists. Our argument is structured around the dispelling of three misconceptions, or myths, about interpretability. First, while ML models' interpretability is often viewed dichotomously, being either interpretable (e.g., linear regression) or "black boxes" (e.g., neural networks), the reality is far more nuanced, affected by multiple factors which should jointly affect model choice. Second, we challenge the idea that interpretability is a necessary trade-off for predictive accuracy, reviewing recent methods from the field which are able to both model complex phenomena and expose the mechanism by which phenomena are related. And third, we present *post hoc explanation*, a recent approach that applies additional methods to black box models, countering the belief that black box models are inherently unusable for the behavioral sciences.

Keywords: Machine Learning; Interpretability; Explanation; Prediction; Natural Language Processing

Explaining Explainability: Interpretable machine learning for the behavioral sciences

Explaining and predicting human behavior are both fundamental goals of the behavioral sciences. And yet, behavioral scientists have arguably neglected prediction in favor of explanation (Yarkoni & Westfall, 2017). To explain behavior means to understand it, establish its causal antecedents, and connect it to broader theoretical frameworks. Historically, this goal has been associated with an approach to statistical modeling that Breiman (2001b) describes as the *data modeling culture*: A scientist who seeks to explain how one or more measured or manipulated (independent) variables relate to one or more other (dependent) variables will start by assuming that the mechanism underlying these relationships can be approximated with a stochastic model (e.g., a linear model). The scientist will then fit such a model to the data and, if the model fits the observed data well, interpret its parameters as reflecting the mechanism explaining the relationships between variables. While the estimated parameters can be used to predict unobserved data, prediction is not the criterion by which the scientist chooses a statistical model.

Advances in machine learning (ML) have enabled scientists and practitioners to make far more accurate predictions than had hitherto been possible. To predict behavior means to make accurate predictions for future or previously unobserved behavior. Historically, this goal has been associated with an approach to statistical modeling that Breiman (2001b) describes as the *algorithmic modeling culture*: A scientist who seeks to predict one set of variables from another set of variables will start by assuming that the mechanism underlying the associations between variables is complex and unknown. Rather than relying on in-sample inferential statistics, the scientist will use cross-validation and other methods to estimate the out-of-sample prediction accuracy of various models (e.g., random forests or neural networks) and choose whichever makes the most accurate predictions.

With some exceptions (e.g., Joel et al., 2020), psychological scientists have eschewed algorithmic, prediction-first approaches to studying human behavior. Behavioral science would, however, benefit from adopting prediction-focused methods. Prediction is not only a better indicator of a model's true fit to the data than in-sample statistics (Yarkoni & Westfall, 2017), but can also be a goal in itself (Rocca & Yarkoni, 2020), allowing analysts and practitioners to generate predictions for unseen data. However, even accepting these benefits, explanation and prediction are not merely different approaches to data modeling but two distinct *cultures* (Breiman, 2001b). Behavioral scientists are not expected to trade in one for the other; indeed, the details of their coordination are still being negotiated (see Hofman et al., 2021). Fundamental to this discussion is the way in which predictive models are perceived, used, and selected; specifically, behavioral scientists operate under the assumption that predictive methods (i.e., ML models) are necessarily opaque, and that theory-building cannot occur within a purely predictive lens. If these assumptions were to be challenged, it becomes easier for the benefits of prediction (e.g., out-of-sample generalization, practical utility) to be paired with the goals of theory-oriented inference.

As behavioral scientists have started recognizing the importance of prediction, researchers in ML have started recognizing the importance of explanation. Interpretable ML (Doshi-Velez & Kim, 2017), otherwise known as "explainability" research or "explainable artificial intelligence" (Arrieta et al., 2020), refers to the growing movement among computational researchers to develop trustworthy, transparent, and fair algorithms. While algorithmic explanation is not a novel idea (see Confalonieri, Coba, Wagner, & Besold, 2021), it has certainly been spurred on by the concerns raised by applications of black box models in sensitive domains, such as healthcare or policing. Though mostly developed for technological purposes, these methods have begun to be used in the natural and physical sciences to model and understand complex phenomena (Roscher, Bohn, Duarte, & Garcke, 2020).

In this paper, we introduce the reader to ideas and methods from interpretable ML, whereby we outline an algorithmic modeling paradigm that is capable of both making accurate predictions and producing explanations of human behavior. We structure this article around three misconceptions about interpretable ML. First, we counter the idea that a model is either interpretable or a black box, discussing recent theoretical work in ML defining interpretability along multiple dimensions. Next, we show that predictive accuracy is not necessarily a trade-off for interpretability; in doing so, we review methods for jointly achieving increased accuracy and interpretability. And lastly, we show that so-called "black box" model, contrary to common belief, can indeed be used for theory-building, specifically by employing what is called "post hoc explanation."

Myth 1: Models are either interpretable or black boxes

The view of ML methods as being exclusively for prediction, and not induction or hypothesis testing, begins with a misunderstanding of interpretability. The dominant view is that a model is either interpretable (e.g., a linear regression) or a black box (e.g., a random forest ensemble).

We begin to deconstruct this view by defining what is meant by a "black box." This metaphor, implying that the model in question provides only predictions and no insight into the mechanisms producing those predictions, is rooted in behaviorist ideas about the human brain (Card, 2017) and is a common view of machine learning. However, most experts today would argue against this view. Notably, Lipton (2018) challenged that, for example, a logistic regression is always more interpretable than a neural network, on account of possible complicating factors such as independent variables being irrelevant to the respective theory. One might adjust one's conception of interpretability as being more of a spectrum, with some methods (and configurations) satisfying constraints to varying degrees. In summary, it is less that predictive methods are categorically uninterpretable, but rather that interpretability is a nuanced construct.

Definitions of Interpretability

In a critical development in formal treatment of the interpretability of ML models, Lipton (2018) defined three dimensions of interpretability: *simulatability*, *decomposability*, and *algorithmic transparency*. Most models satisfy some, but not all, interpretability dimensions, encouraging us to select models based on the interpretability needed for the given modeling task.

Simulatability

A model is *simulatable* if it can be mentally simulated. For example, a human presented with a linear regression with three variables and a single outcome can simulate the prediction of the model for new data (i.e., by multiplying coefficients by feature values and summing them). One of the dimensions of a model that simulatability captures is *scale*. A linear regression of 1000 predictors is not simulatable, for the simple reason that the human mind cannot replicate a model of such scale; similarly, a decision tree is simulatable only if the number of nodes can be mentally traced. On the other hand, a neural network can be simulatable at smaller scales, provided that the input and parameters of the network (e.g., a 1-layer perceptron classifier with a small number of "hidden units") are represented in a few dimensions.

Decomposability

Decomposability is the ability of a model to be reduced (or "decomposed") to interpretable units, both in terms of modeling components (e.g., a node in a decision tree) but also to inputs (e.g., independent variables). For example, suppose that a large set of variables (denoted as the matrix X) is transformed using Principle Components Analysis (PCA), and the reduced feature matrix (denoted as X_{PCA}) is used in a supervised model to predict some target label y. Unless the principle components (i.e., the dimensions of X_{PCA}) are analyzed and given labels by the researcher, no predictive

model based on X_{PCA} can be said to be decomposable, regardless of whether it is a linear regression or a complex neural network.

Decomposability of input variables is important when considering the task of induction from a model. Induction, the process of establishing new, general knowledge from observations, is commonly approached through "transparent" models such as linear regression. Linear regression is one method often used for induction, and this can be seen from the standpoint of decomposability. If independent variables can be mapped to theoretical constructs, then relationships among variables in the model can translated to relationships among theoretical constructs. Furthermore, decomposability implies that the components of a model can be individually inspected and assigned meaning. For example, the coefficients in a regression represent the association between the dependent variable and the independent variables. In contrast, a method lacking decomposability, such as a neural network, cannot directly support induction, as information is "distributed" across model components, with no one value being available that indicates association between meaningful variables.

Algorithmic transparency

Algorithmic transparency refers to models' having intelligible training processes. The prototypical algorithmically transparent optimization process is least squares regression, in which it is demonstrably clear that model coefficients are derived through the least squares formulation. Similarly, most algorithms for learning tree structures (e.g., the CART algorithm; Breiman, Friedman, Stone, & Olshen, 1984) follow interpretable processes for selecting the next variable on which to split the decision tree, as well as halting criteria for fitting the model. However, many model fitting processes are not algorithmically transparent; of note, neural networks are fit using a method called "stochastic gradient descent" (SGD). SGD is effective at achieving optimal model parameters for complex models — e.g., a neural network with millions of parameters

structured in multiple connected layers — but, because of their randomized nature, are entirely inaccessible to intuitive inspection.

In summary, recent theorizing has shown that interpretability is not a binary attribute but instead depends on whether a model is simulatable, decomposable, and algorithmically transparent. Whereas the previous dichotomy between interpretable models and black boxes suggests selecting the interpretable model every time, extracting knowledge from models is in fact never straight-forward. While conventional methods like linear regression are the default for many data analysts, it is common for those methods to lack decomposability (e.g., through correlation among independent variables) and simulatability. And while this does not motivate the blind adoption of neural networks into every model of data, it does motivate a reexamination of the modeling toolbox. In the next sections, we describe recent advances in ML and related fields in terms of improving model interpretability, as well as developing methods to extract information from opaque models.

Myth 2: Accurate models are necessarily black boxes

When speaking of algorithmic models, there is an assumed trade-off between interpretability and accuracy. Models are inherently restricted to two categories: high-accuracy models that are black boxes (e.g., neural networks), or low-accuracy models that are interpretable (e.g., linear regression). Perhaps this trade-off was valid at one point, when predictive algorithms were first being introduced in ML; however, building algorithms capable of both accuracy and interpretation have been an active area in ML for decades, and interest in such models has exploded in the past few years. Arrieta et al. (2020) described how designing “transparent boxes” — metaphorically opposing the “black box” — offers analysts the ability to achieve maximal predictive generalization while maintaining a higher degree of interpretability.

Here, we survey how interpretability is achieved by altering models’ learning

procedure and structure. By providing mathematical detail for these models, we believe that the technical challenges of interpretability can be appreciated. In addition, understanding how recent research has produced interpretable models capable of learning general relationships from data indicates the limitations of traditional interpretable methods. For example, while decision trees are interpretable at face-value, it is difficult to fit them to high-dimensional or large datasets.

Neural Networks for Detecting Non-Linear Effects

Neural networks are renowned for their ability to represent and learn complex functions between independent and dependent variables. In the behavioral sciences, we might envision their being used for modeling large-scale observations of behavior, high-dimensional fMRI sequences, or language; however, in their typical formulation they fail in some of the most basic metrics for interpretability. It can be argued that neural networks lack simulatability (assuming a large enough model in terms of features and modeling components), decomposability, and algorithmic transparency, functioning purely for prediction. This is why recent research has aimed to improve the interpretability of neural networks, specifically their decomposability, by fusing them together with interpretable models like Generalized Additive Models. In this way, non-linear relationships among variables can be captured and interpreted through traditional analysis of main effects and interactions.

The first notable method using the approach is the “Explainable Neural Network” (xNN; Vaughan, Sudjianto, Brahimi, Chen, & Nair, 2018; Yang, Zhang, & Sudjianto, 2020), which builds on the Additive Index Models (AIM; Ruan & Yuan, 2010). AIM applies multiple transforms to the independent variables, with each transform function selecting (“indexing”) a subset of variables to consider. With the xNN, the same logic applies, only each transform is a separate neural network. The xNN applies multiple neural networks to the dataset, with each sub-network selecting different variables to

include, and learning separate non-linear transformations of those variables. The interpretation of variable importance from an xNN can be performed in a variety of ways; for example, Vaughan et al. (2018) were able to identify, from a trained xNN model, which variables were selected by each sub-network, as well as the aggregated importance of each independent variable across those sub-networks in which the variable was selected. This is essentially extracting non-linear interactions from a dataset, with each sub-network capturing an interaction.

The Adaptable Explainable Neural Network (AxNN; Chen, Vaughan, Nair, & Sudjianto, 2020) goes further than the xNN in terms of being able to identify non-linear main effects and interactions among variables. The AxNN combines the xNN with Generalized Additive Models (Hastie & Tibshirani, 1986). Each independent variable being fed to its own sub-network, allowing main effects to be recovered from the outputs of the corresponding sub-network. Following the estimation of main effects, xNN, which has the advantage of learning variable selection for multiple sub-networks, is applied in order to detect interactions. Alternative, similar approaches to detecting interactions with neural networks include the Neural Interaction Transparency framework (Tsang, Cheng, & Liu, 2018), though Chen et al. (2020) note that the lack of separation between main effect and interaction can lead to spurious interactions being detected.

Predictions Based on Similarity to Training Observations

Deep Weighted Averaging Classifiers (DWAC; Card, Zhang, & Smith, 2019) are inspired by non-parametric kernel regression, in which the learning objective is formulated as a weighted sum of the training data rather than a manipulation of feature representations. Typically, a statistical classifier predicts based on the feature values of a single instance (and computations applied to those feature values, e.g., a linear transform of features in least squares regression). In contrast, DWAC predicts based on

the new instance's similarity to the entirety of the training data:

$$P(y = k|\boldsymbol{x}) = \frac{\sum_{i=1}^{t} \mathbb{1}(y_i = k) w(\boldsymbol{h}, \boldsymbol{h}_i)}{\sum_{j=1}^{t} w(\boldsymbol{h}, \boldsymbol{h}_j)} \tag{1}$$

with w being a similarity function, such as cosine similarity, that maps two vectors to a scalar similarity value, and $\boldsymbol{h}$ being an encoded (i.e., embedded) $\boldsymbol{x}$. In other words, for a new observation $\boldsymbol{x}$, the probability that $\boldsymbol{x}$ is class k is proportional to the cumulative weight of all the instances in the training data that were class k. Here, weight of each instance is determined by the similarity to $\boldsymbol{x}$.

Training classifiers in this way is both useful and desirable. It is epistemically desirable to explain a prediction *precisely* in terms of previous observations. In this way, the researcher must still do the work of interpreting those similar training instances, but the classification algorithm itself becomes transparent, and the dependence on training data is readily apparent. Second, DWAC can be used to detect if new data are out of distribution — specifically, by virtue of the fact that few training observations are similar to the new observation. Often, classifiers will output a probability estimate on new data without reporting uncertainty. In this case, uncertainty is quantified by familiarity with respect to the training data.

Improved Learning for Rule-Based Methods

One of the starting points for understanding the accuracy–interpretability trade-off was that interpretable models fail to generalize well (i.e., predict accurately outside of the training distribution). For example, a decision tree is intuitive to understand, but decision trees are unable to fit high-dimensional complex datasets without over-fitting (Breiman, 2001a). In the face of this difficulty, recent work has focused on improving rule-based learning algorithms' ability to avoid over-fitting or, in other words, to learn accurate models of data that also perform well on out-of-sample data. Two examples of such approaches are an algorithm for learning interpretable rule

ensembles and a new method for learning a decision tree from data that optimizes performance with respect to size constraints.

Mita, Papotti, Filippone, and Michiardi (2020) proposed "Learning Interpretable Boolean Rule Ensembles" (LIBRE), which are accurate particularly in imbalanced datasets and are easily aggregated into a final set of (interpretable) rules. LIBRE operates on Boolean feature sets, such that an instance $\boldsymbol{x} \in \{0,1\}^p$ consists of p binary variables, the target is a binary variable ($y \in \{0,1\}$), and the goal is to learn a function that separates positive and negative instances of y in the binary feature space of $\boldsymbol{x}$. In the first step, "weak" learners output sets of rules, each of which takes the form

$$\text{IF} \quad \boldsymbol{x}_1 == 1 \quad \text{OR} \quad \boldsymbol{x}_3 == 4 \quad \text{THEN} \quad y \leftarrow 1 \tag{2}$$

Like random forests and other ensemble techniques, LIBRE fits many of these weak learners on random subsets of the data. However, unlike previous ensemble methods that apply voting or other aggregation to the set of weak learners, LIBRE aggregates rule sets by first taking the union of "boundary instances" (the antecedent of Eq. 2) from all weak learners and efficiently selecting the most informative (with respect to differentiating between y values) antecedents from the combined list of rules. In summary, LIBRE uses ensemble techniques over random subsets of the data to establish a pool of informative Boolean rules, then selects among these rules to derive the most predictive boundary.

In addition to interpretable, accurate decision lists, recent research has addressed the problem of learning accurate and simulatable trees. Specifically, recently, Günlük, Kalagnanam, Li, Menickelly, and Scheinberg (2021) formulated learning a binary decision tree using an integer programming (constraint satisfaction) method. In contrast to typical decision tree algorithms, for example CART (Breiman et al., 1984), these optimal decision trees achieve provably good performance with respect to size

constraints (e.g., maximizing accuracy while keeping the tree size to a constant value) and do not rely on ensemble techniques that generate many small trees via CART.

Myth 3: Black boxes cannot be interpreted

There is a pejorative connotation associated with the "black box" label. Without model transparency, we are blind to possible faults, such as a data-driven algorithm for cropping images detecting White faces more accurately than Black faces (see Chowdhury, 2021), and are fundamentally unable to acquire explanatory insight into behavioral phenomena. In the previous section, we explained how high-powered predictive algorithms are not necessarily black boxes, and that new algorithms have been proposed for highly predictive inferential purposes. But in doing so, we implicitly reaffirmed the unwelcome status of black boxes. In this section, we explain that even though an algorithm can be labeled a black box, this label does not need to be pejorative. Rather, black boxes merely motivate a new strategy for extracting information from models trained to predict complex phenomena. In the literature, the strategy of "post hoc explanation" is to understand and explain a black box model, often without accessing more than the inputs and the outputs to a model.

In Table 1 we list the methods selected for review in this section. For broader coverage of and greater detail on methods, the reader is recommended recent surveys including Montavon, Samek, and Müller (2018), Arrieta et al. (2020), and Guidotti et al. (2018).

Key Concepts in Post Hoc Explanation

Before detailing the methods available in the post hoc toolbox, we discuss several high-level concepts.

[1] Requires particular implementation per model, but is not reliant on a specific model architecture

Table 1
Post hoc algorithms surveyed in this section.

	Type	Description
	Sensitivity Analysis	
Shapley Additive Explanation (SHAP; Lundberg & Lee, 2017)	Local, Model-agnostic	Feature importance across combinations of other features
Sampling and Occlusion (SOC; Jin, Wei, Du, Xue, & Ren, 2019)	Local, Model Agnostic[1]	Word and phrase importance from text model
	Model Simplification	
Born Again Trees (Vidal & Schiffer, 2020)	Global, Tree-specific	Tree ensemble → Tree
Companion Rule Lists (Pan, Wang, & Hara, 2020)	Global, Model-agnostic	Black box → rule-list
CXPlain (Schwab & Karlen, 2019)	Global, Model-agnostic	Feature relevance using causal objective to approximate black box predictions
	Instance-Based Approaches	
Instance influence	Both, Model-specific	Single training observations with high relevance to predictions
Counterfactual explanations	Local, Model-agnostic	Minimal change to input in order to change model prediction

Local and global explanations

Popularized by works such as Ribeiro, Singh, and Guestrin (2016) and Lundberg and Lee (2017), *local explanation* involves justifying or interrogating single predictions by a model. Common use-cases of local explanation include medical practitioners being able to understand why a particular automated diagnosis was made. Beyond the specific cases in which insight into individual predictions is useful for decision-making, local

explanation yields a useful decomposition of complex models, with potentially multiple sets of explanations being offered depending on the specific input. A non-linear model by its nature cannot be summarized by a single set of feature importance scores[2] — this is the nature of non-linearity. Instead, non-linear models can be seen as the combination of multiple interpretable models.

Local explanation is somewhat of a departure from interpretation in common regression practices, which we might call *global explanation*. In global explanations, the goal is to determine the overall effect of an independent variable on the dependent variable. To understand the contrast between global and local, we can consider the goal of understanding the predictions of a "sentiment classifier," a common NLP model used to predict whether a text contains positive or negative affect.In a linear sentiment model, independent variables (e.g., word frequencies) are assigned coefficients, or weights, which indicate the association between the word and the sentiment variable. Predicting the sentiment for a new sentence relies on the same coefficients each time, hence the model is "global." However, a non-linear sentiment classifier, such as a neural network, might weight words differently depending on the other words in the sentence. For example, the presence of the word "happy" can be used to predict positive sentiment, but when preceded by the word "not," and accompanied by the word "depressed" later in the sentence. Thus, how are we to assign a single coefficient to the word "happy"?

One way to understand local explanation is that complex phenomena require complex explanations. Instead, a more local focus can show how the relationship between dependent and independent variables can change dramatically for different observations, and that complex models can be deconstructed into multiple local explanations rather than a single global one.

---

[2] Feature importance is a term used in ML research, and is analogous to the interpretation of regression model coefficients.

Model-agnostic explanations

An explanation is model-agnostic if it does not rely on knowledge of the to-be-explained model, but rather only input–output pairs (i.e., the feature matrix X and the predicted values y). Model-agnostic explanations have the advantages of accessibility and portability — explanation as a practice is less viable if a separate method has to be developed for each model or model family — but are somewhat limited due to their lack of access to the internal states of models.

Faithfulness of explanations to models

An issue with the post hoc approach is the necessarily approximate nature of prying open the black box. There is rarely a guarantee that what can be extract from a black box model is a true reflection of the model. In fact, recent work has demonstrated that an explanation can be convincing, or plausible, yet be inaccurate in terms of representing the model (Jacovi & Goldberg, 2020). An explanation's *faithfulness* to the original model comprises its correspondence to the original model's representation of the data (e.g., a decision tree represents information in terms of nodes and branches from nodes, while a regression represents information in terms of coefficients) and its ability to reproduce the original predictions. No explanation is going to be perfectly faithful to the original, but it is expected to measure and report metrics of faithfulness (e.g., similarity of explanation model predictions to black box model predictions).

Sensitivity Analysis via Input Perturbation

One approach to local explanation is *sensitivity analysis*, often involving *input perturbations*. Sensitivity analysis in the context of explainability refers to the process of modifying the inputs to a model and observing changes in prediction (see Figure 1).

Shapley Additive Explanations (SHAP; Lundberg & Lee, 2017), which are inspired by game theory, consider the information gain (i.e., increase in predicted value or

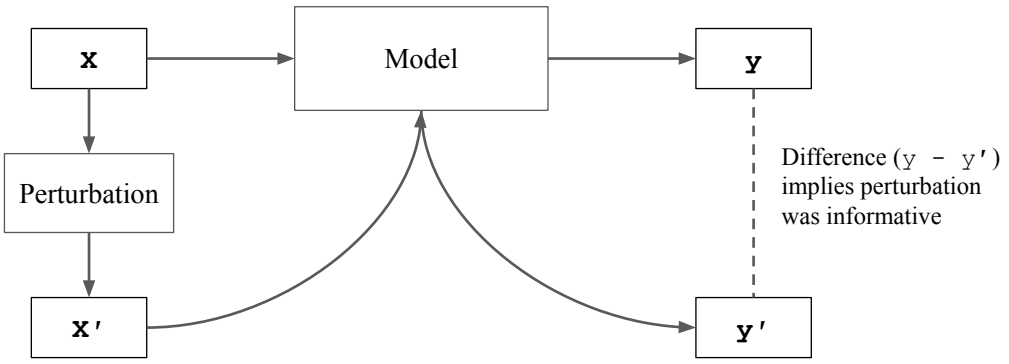

Figure 1
Perturbing inputs, more generally known as "sensitivity analysis," is a local explanation technique whereby a given input to a model is "perturbed" (e.g., modifying one feature) and changes in model output are used for explanation.

probability) of including a given variable over all possible permutations of the other variables. Perturbation here refers to systematically measuring prediction changes over every combination (or unique selection) of features. For example, with variables x_1, x_2, and x_3, the effect of x_1 on the model's prediction y is equivalent to the difference in predicted value when one adds x_1 to $\{\}$, $\{x_2\}$, $\{x_3\}$, and $\{x_2, x_3\}$. Intuitively, if x_1 is important for predicting y, then it will, on average, add information to every configuration of the other variables.

Specialized methods have been proposed for sensitivity analysis of models of language, as language is different from many other "tabular" data formats given that it is sequential and each observation is a different length. Jin et al. (2019) describe a "Sampling and Occlusion" (SOC) method for explaining predictions from a text classification model. Similarly to SHAP, SOC examines changes in the model's predictions if one "occludes" the considered word or phrase in a sentence. In an example of the use-case of such an algorithm, Kennedy, Jin, Davani, Dehghani, and Ren (2020) extracted feature relevance estimates of a large language model for the task of hate speech classification. Aggregating local word- and phrase-level explanations across predictions, SOC revealed significant bias towards social group identifiers (e.g., "black,"

"gay," etc.), a phenomenon known in the bias and fairness community to affect models trained from imbalanced data (Wiegand, Ruppenhofer, & Kleinbauer, 2019).

Model Simplification

One common approach to global explanation is *model simplification*, whereby a complex, black box model is repackaged algorithmically into a more transparent model that is still capable of similar predictions. This is typically done by training the secondary model on the predictions of the primary model. For example, a multi-layer neural network could be simplified by training a decision tree to mimic its outputs.

Model simplification is motivated by the difficulty in jointly learning and representing complex functions from data, and attempts to separate the two tasks. Intuitively, models which are best able to represent complex, non-linear processes — e.g., decision trees, rule lists — would be the ideal option for studying complex phenomena. But despite models such as decision trees having this desirable representational capacity, they are hard to train (see Bastani, Kim, & Bastani, 2017, p. 1). A common strategy in the past has been to learn many, smaller trees, achieving generalization via pooling of information (i.e., random forests); however, this ensemble approach has the known issue of a lack of interpretability (Breiman, 2001b). Rather than develop improved models that possess both the ability to clearly represent information and learn it from data, a growing trend in interpretable ML has been to first train a black box model (capable of learning) and subsequently training a secondary model (capable of representing) using the first model to guide learning.

Simplifying to rule-based models

Complex models can be simplified to rule-based models, such as decision trees. "Born again" trees (Vidal & Schiffer, 2020) are generated from an ensemble of decision trees (e.g., random forest), with the goal of recapturing the relationship learned by a random forest or gradient boosting trees in a minimally-sized tree. Similarly, Pan et al.

(2020) proposed an interactive hybrid model, "Companion Rule Lists", which involves re-training a rule list (analogous to a decision tree but without branching hierarchies) on both the original data $(\boldsymbol{X}, Y)$ and the black box model predictions $(\boldsymbol{X}, \hat{Y})$. If presented with a black box prediction, an analyst can query the companion rule list for a similar prediction as the original model but in rule-list form.

Simplification for feature importance

In addition to simplifying to a rule-based model, one can also apply simplification in order to produce feature importance estimates. Schwab and Karlen (2019), following Granger (1969), proposed to formulate model simplification as a global sensitivity analysis. They assume that a causal relationship $\boldsymbol{x}_i \rightarrow \hat{y}$ between random variables $\boldsymbol{x}_i$ and $\hat{y}$ exists if we are better able to predict $\hat{y}$ using all available information than if the information apart from $\boldsymbol{x}_i$ had been used. The primary quantity of interest in this approach, named "CXPlain," is the difference in the model $\boldsymbol{f}(\cdot)$'s prediction when including $\boldsymbol{x}_i$ and when excluding $\boldsymbol{x}_i$ (e.g., by replacing with zeros or the mean of the variable). Subsequently, estimates of importance are produced by training any supervised learning model — in this case, a multi-layer perceptron neural network — to approximate the prediction differences, and confidence intervals for testing significance are produced using bootstrap subsampling of the original dataset, reestimating prediction differences, and retraining the network.

Explaining via Example and Counterfactual

A pitfall of using black box models is that one relies too much on model performance versus actually understanding how the model is representing data. The post hoc explanation methods discussed thus far circumvent this pitfall by measuring feature importance or by simplifying a complex model into an interpretable one. But perhaps the most intuitive method for understanding a model post hoc is to extract examples and counterfactuals.

Examples and counterfactuals (alterations of an instance's features such that the model prediction changes) provide insight into how a model represents interesting or important instances in the dataset, and potentially the "boundaries" between classes in a classifier. This instance-level approach ensures that analysts' intuitions about model behavior (e.g., "This piece of text should be classified as hate speech," "These two instances should be similar," etc.) correspond to actual model behavior. Previous work has applied instance-based explanations specifically to address the gap between the workings of a model and the user's understanding of the model (Martens & Provost, 2014).

Prototypes and important instances

One type of example-based approach is to score observations based on how well they exemplify a class concept, or to what degree single observations influence models. In the specific context of neural networks, Montavon et al. (2018)'s described "prototype" generation, which estimates the exact values of the input to a model that maximize the probability of the target class or value. A particular method in this approach is to measure the bias of a model — unfair treatment or representation of protected social groups — in terms of the effect of including particular training observations (Brunet, Alkalay-Houlihan, Anderson, & Zemel, 2019).

Counterfactual generation for explanation

A counterfactual can indicate what changes to a given instance are needed in order to change the prediction. In sensitivity analysis, SHAP (for example) could be applied to a classifier trained to predict whether a loan applicant ought to be granted a loan (Mothilal, Sharma, & Tan, 2020), in order to measure the influence of a given feature (e.g., "age" or "loan history") on a "No" prediction. In contrast, counterfactual explanation seeks to determine, for a given loan applicant, the minimal changes to their variables that would change the classifier's prediction for that applicant from "No" to

"Yes."

Counterfactual generation and evaluation, beyond the specific techniques described above, describe a general approach to model interpretation. Behavioral scientists can use prototype methods to identify key instances, potentially identifying when a model has become biased or reliant on outliers. Counterfactual explanations methods can be used to determine how models represent class boundaries; in an analysis of partisanship in a social network, for example, counterfactuals could correspond to the minimal changes to a user "node" such that a classifier changes its prediction from one political party to another. Critically, this information is difficult to access from global feature relevance.

Remaining Issues

Below, we discuss the obstacles impeding the full realization of interpretable ML in the behavioral sciences, and how future work can help to overcome them.

The Technical Gap

Like any statistical technique, a method is as good as its implementation. Given the recency of most machine learning methods reviewed in this paper, there is little in the way of convenient programming options for implementing most techniques. While most techniques that we reviewed are open source, it is another matter to be readily accessible through.

Another factor limiting accessibility to behavioral scientists is the fact that other stakeholders hold greater sway — e.g., machine learning engineers, computer vision specialists, etc. — and thus many resources are embedded within deep learning programming frameworks.

Theoretical Integration using New Interpretive Modalities

That interpretable ML is useful for the behavioral sciences should be apparent at this point; however, there are unresolved questions as to how theory-building can be accomplished with this new toolbox. The data modeling culture — testing hypotheses by measuring an a priori model's fit to data — carries with it procedural norms and practices used to conduct inferences about general constructs from particular measurements. With new explanation algorithms, there will also come the challenges of using new interpretive modalities, which cannot be simply plugged in to the existing methodological ecosystem. Rather, new norms and practices (e.g., strategies to test hypotheses with decision lists) will emerge through trial and error. However, as of yet these "tips and tricks" for operating the interpretable ML toolbox are undetermined.

The Slow Development of Explanatory Modeling in ML

The motivation of most explainability research is fundamentally different from how the behavioral sciences would use explainability methods. The motivations of ML are largely having to do with technological production (i.e., predictive performance), and explainability metrics, methods, and applications are developed within this assumption. Thus, it is a challenge to adapt interpretable ML research to behavioral data and research questions. For example, much of the interpretable ML literature is structured around decision-making ("What information is a driverless car using to make driving decisions?"), a common goal of autonomous systems but not of studies of human behavior. While the present work attempts to address this by explaining the interpretable ML toolbox, more collaboration between behavioral scientists and ML and NLP researchers can give rise to the identification of new use-cases and applications, new requirements for algorithms described, and more interpretable ML literature disseminated to interested researchers.

References

Arrieta, A. B., Díaz-Rodríguez, N., Del Ser, J., Bennetot, A., Tabik, S., Barbado, A., … others (2020). Explainable artificial intelligence (xai): Concepts, taxonomies, opportunities and challenges toward responsible ai. *Information Fusion*, *58*, 82–115.

Bastani, O., Kim, C., & Bastani, H. (2017). Interpreting blackbox models via model extraction. *arXiv preprint arXiv:1705.08504*.

Breiman, L. (2001a). Random forests. *Machine learning*, *45*(1), 5–32.

Breiman, L. (2001b). Statistical modeling: The two cultures (with comments and a rejoinder by the author). *Statistical science*, *16*(3), 199–231.

Breiman, L., Friedman, J., Stone, C. J., & Olshen, R. A. (1984). *Classification and regression trees*. CRC press.

Brunet, M.-E., Alkalay-Houlihan, C., Anderson, A., & Zemel, R. (2019). Understanding the origins of bias in word embeddings. In *International conference on machine learning* (pp. 803–811).

Card, D. (2017). *The 'black box' metaphor in machine learning*. https://towardsdatascience.com/the-black-box-metaphor-in-machine-learning-4e57a3a1d2b0. Towards Data Science. (Accessed: 04/20/2021)

Card, D., Zhang, M., & Smith, N. A. (2019). Deep weighted averaging classifiers. In *Proceedings of the conference on fairness, accountability, and transparency* (pp. 369–378).

Chen, J., Vaughan, J., Nair, V., & Sudjianto, A. (2020). Adaptive explainable neural networks (axnns). *Available at SSRN 3569318*.

Chowdhury, R. (2021). *Sharing learnings about our image cropping algorithm*. https://blog.twitter.com/engineering/en_us/topics/insights/2021/sharing-learnings-about-our-image-cropping-algorithm. (Accessed:

2021-06-18)

Confalonieri, R., Coba, L., Wagner, B., & Besold, T. R. (2021). A historical perspective of explainable artificial intelligence. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, *11*(1), e1391.

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

Granger, C. W. (1969). Investigating causal relations by econometric models and cross-spectral methods. *Econometrica: journal of the Econometric Society*, 424–438.

Guidotti, R., Monreale, A., Ruggieri, S., Turini, F., Giannotti, F., & Pedreschi, D. (2018). A survey of methods for explaining black box models. *ACM computing surveys (CSUR)*, *51*(5), 1–42.

Günlük, O., Kalagnanam, J., Li, M., Menickelly, M., & Scheinberg, K. (2021). Optimal decision trees for categorical data via integer programming. *Journal of Global Optimization*, 1–28.

Hastie, T., & Tibshirani, R. (1986). Generalized Additive Models. *Statistical Science*, *1*(3), 297 – 310. Retrieved from `https://doi.org/10.1214/ss/1177013604` doi: 10.1214/ss/1177013604

Hofman, J. M., Watts, D. J., Athey, S., Garip, F., Griffiths, T. L., Kleinberg, J., … others (2021). Integrating explanation and prediction in computational social science. *Nature*, 1–8.

Jacovi, A., & Goldberg, Y. (2020). Towards faithfully interpretable nlp systems: How should we define and evaluate faithfulness? In *Proceedings of the 58th annual meeting of the association for computational linguistics* (pp. 4198–4205).

Jin, X., Wei, Z., Du, J., Xue, X., & Ren, X. (2019). Towards hierarchical importance attribution: Explaining compositional semantics for neural sequence models. In *International conference on learning representations*.

Joel, S., Eastwick, P. W., Allison, C. J., Arriaga, X. B., Baker, Z. G., Bar-Kalifa, E., … others (2020). Machine learning uncovers the most robust self-report predictors of relationship quality across 43 longitudinal couples studies. *Proceedings of the National Academy of Sciences*, *117*(32), 19061–19071.

Kennedy, B., Jin, X., Davani, A. M., Dehghani, M., & Ren, X. (2020). Contextualizing hate speech classifiers with post-hoc explanation. In *Proceedings of the 58th annual meeting of the association for computational linguistics* (pp. 5435–5442).

Lipton, Z. C. (2018). The mythos of model interpretability: In machine learning, the concept of interpretability is both important and slippery. *Queue*, *16*(3), 31–57.

Lundberg, S. M., & Lee, S.-I. (2017). A unified approach to interpreting model predictions. In *Proceedings of the 31st international conference on neural information processing systems* (pp. 4768–4777).

Martens, D., & Provost, F. (2014). Explaining data-driven document classifications. *Mis Quarterly*, *38*(1), 73–100.

Mita, G., Papotti, P., Filippone, M., & Michiardi, P. (2020). Libre: Learning interpretable boolean rule ensembles. In *International conference on artificial intelligence and statistics* (pp. 245–255).

Montavon, G., Samek, W., & Müller, K.-R. (2018). Methods for interpreting and understanding deep neural networks. *Digital Signal Processing*, *73*, 1–15.

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).

Pan, D., Wang, T., & Hara, S. (2020). Interpretable companions for black-box models. In *International conference on artificial intelligence and statistics* (pp. 2444–2454).

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).

Rocca, R., & Yarkoni, T. (2020). *Putting psychology to the test: Rethinking model evaluation through benchmarking and prediction*.

Roscher, R., Bohn, B., Duarte, M. F., & Garcke, J. (2020). Explainable machine learning for scientific insights and discoveries. *IEEE Access*, *8*, 42200–42216.

Ruan, L., & Yuan, M. (2010). Dimension reduction and parameter estimation for additive index models. *Statistics and its Interface*, *3*(4), 493–499.

Schwab, P., & Karlen, W. (2019). Cxplain: Causal explanations for model interpretation under uncertainty. *arXiv preprint arXiv:1910.12336*.

Tsang, M., Cheng, D., & Liu, Y. (2018). Detecting statistical interactions from neural network weights. In *International conference on learning representations*.

Vaughan, J., Sudjianto, A., Brahimi, E., Chen, J., & Nair, V. N. (2018). Explainable neural networks based on additive index models. *arXiv preprint arXiv:1806.01933*.

Vidal, T., & Schiffer, M. (2020). Born-again tree ensembles. In *International conference on machine learning* (pp. 9743–9753).

Wiegand, M., Ruppenhofer, J., & Kleinbauer, T. (2019). Detection of abusive language: the problem of biased datasets. In *Proceedings of the 2019 conference of the north american chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)* (pp. 602–608).

Yang, Z., Zhang, A., & Sudjianto, A. (2020). Enhancing explainability of neural networks through architecture constraints. *IEEE Transactions on Neural Networks and Learning Systems*.

Yarkoni, T., & Westfall, J. (2017). Choosing prediction over explanation in psychology: Lessons from machine learning. *Perspectives on Psychological Science*, *12*(6), 1100–1122.