arXiv:2011.14878v2 [cs.LG] 13 May 2022

Explaining by Removing: A Unified Framework for Model Explanation

Ian C. Covert ICOVERT@CS.WASHINGTON.EDU
Paul G. Allen School of Computer Science & Engineering
University of Washington
Seattle, WA 98195, USA

Scott Lundberg SCOTT.LUNDBERG@MICROSOFT.COM
Microsoft Research
Microsoft Corporation
Redmond, WA 98052, USA

Su-In Lee SUINLEE@CS.WASHINGTON.EDU
Paul G. Allen School of Computer Science & Engineering
University of Washington
Seattle, WA 98195, USA

Abstract

Researchers have proposed a wide variety of model explanation approaches, but it remains unclear how most methods are related or when one method is preferable to another. We describe a new unified class of methods, *removal-based explanations*, that are based on the principle of simulating feature removal to quantify each feature's influence. These methods vary in several respects, so we develop a framework that characterizes each method along three dimensions: 1) how the method removes features, 2) what model behavior the method explains, and 3) how the method summarizes each feature's influence. Our framework unifies 26 existing methods, including several of the most widely used approaches: SHAP, LIME, Meaningful Perturbations, and permutation tests. This newly understood class of explanation methods has rich connections that we examine using tools that have been largely overlooked by the explainability literature. To anchor removal-based explanations in cognitive psychology, we show that feature removal is a simple application of subtractive counterfactual reasoning. Ideas from cooperative game theory shed light on the relationships and trade-offs among different methods, and we derive conditions under which all removal-based explanations have information-theoretic interpretations. Through this analysis, we develop a unified framework that helps practitioners better understand model explanation tools, and that offers a strong theoretical foundation upon which future explainability research can build.

Keywords: Model explanation, interpretability, information theory, cooperative game theory, psychology

©2021 Ian Covert, Scott Lundberg and Su-In Lee.

License: CC-BY 4.0, see https://creativecommons.org/licenses/by/4.0/.

1. Introduction

The proliferation of black-box models has made machine learning (ML) explainability increasingly important, and researchers have now proposed a variety of model explanation approaches (Zeiler and Fergus, 2014; Ribeiro et al., 2016; Lundberg and Lee, 2017). Despite progress in the field, the relationships and trade-offs among these methods have not been rigorously investigated, and researchers have not always formalized their fundamental ideas about how to interpret models (Lipton, 2018). This makes the interpretability literature difficult to navigate and raises questions about whether existing methods relate to human processes for explaining complex decisions (Miller et al., 2017; Miller, 2019).

Here, we present a comprehensive new framework that unifies a substantial portion of the model explanation literature. Our framework is based on the observation that many methods can be understood as *simulating feature removal* to quantify each feature's influence on a model. The intuition behind these methods is similar (depicted in Figure 1), but each one takes a slightly different approach to the removal operation: some replace features with neutral values (Zeiler and Fergus, 2014; Petsiuk et al., 2018), others marginalize over a distribution of values (Strobl et al., 2008; Lundberg and Lee, 2017), and still others train separate models for each subset of features (Lipovetsky and Conklin, 2001; Štrumbelj et al., 2009). These methods also vary in other respects, as we describe below.

We refer to this class of approaches as *removal-based explanations* and identify 26[1] existing methods that rely on the feature removal principle, including several of the most widely used methods (SHAP, LIME, Meaningful Perturbations, permutation tests). We then develop a framework that shows how each method arises from various combinations of three choices: 1) how the method removes features from the model, 2) what model behavior the method analyzes, and 3) how the method summarizes each feature's influence on the model. By characterizing each method in terms of three precise mathematical choices, we are able to systematize their shared elements and show that they all rely on the same fundamental approach—feature removal.

The model explanation field has grown significantly in the past decade, and we take a broader view of the literature than existing unification theories. Our framework's flexibility lets us establish links between disparate classes of methods (e.g., computer vision-focused methods, global methods, game-theoretic methods, feature selection methods) and show that the literature is more interconnected than previously recognized. Exposing these relationships makes the literature more coherent, and it simplifies the process of reasoning about the benefits of each method by showing that their differences often amount to minor, interchangeable design choices.

To better understand the unique advantages of each method, we thoroughly analyze our framework's theoretical foundation by examining its connections with related fields. In particular, we find that cognitive psychology, cooperative game theory and information theory are intimately connected to removal-based explanations and help shed light on the trade-offs between different approaches. The extent of these links is perhaps surprising, because few methods explicitly reference these related fields.

Our approach yields many new results and provides a strong theoretical foundation for understanding existing methods and guiding future work. Our contributions include:

1. This total count does not include minor variations on the approaches we identified.

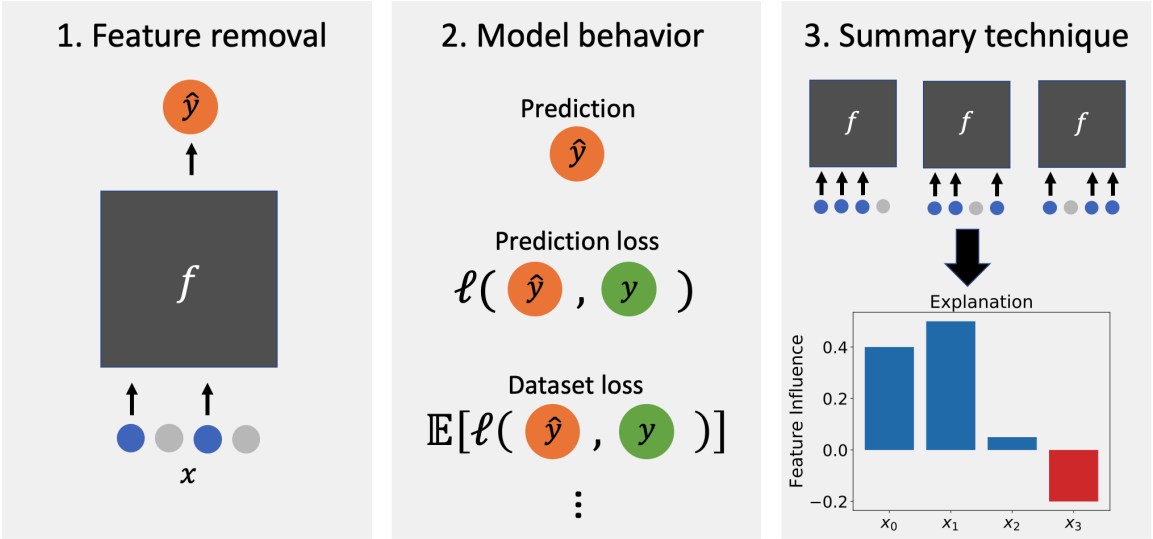

Figure 1: A unified framework for *removal-based explanations*. Each method is determined by three choices: how it removes features, what model behavior it analyzes, and how it summarizes feature influence.

1. We present a **unified framework** that characterizes 26 existing explanation methods and formalizes a new class of **removal-based explanations**. The framework integrates classes of methods that may have previously appeared disjoint, including local and global approaches as well as feature attribution and feature selection methods. In our experiments, we develop and compare 60+ new explanation approaches by mixing and matching the choices that methods make in each dimension of our framework.

2. We develop **new mathematical tools** to represent different approaches for removing features from ML models. Then, by incorporating an underlying data distribution, we argue that marginalizing out features using their conditional distribution is the only approach that is consistent with standard probability axioms. Finally, we prove that several alternative choices approximate this approach and that this approach gives all removal-based explanations an **information-theoretic interpretation**.

3. We demonstrate that **all removal-based explanations are implicitly tied to cooperative game theory**, and we leverage decades of game theory research to highlight advantages of the Shapley value over alternative allocation strategies. Building on these findings, we also show that several feature attribution techniques are generalized by LIME (Ribeiro et al., 2016), and that several feature selection techniques are generalized by the Masking Model approach (Dabkowski and Gal, 2017).

4. We consult social science research to understand the intuition behind feature removal as an approach to model explanation. We find that feature removal is a simple application of **subtractive counterfactual reasoning**, or, equivalently, of Mill's **method of difference** from the philosophy of scientific induction (Mill, 1884).

The paper is organized as follows. We begin with background on the model explanation problem and a review of prior work (Section 2), and we then give an overview of our framework (Section 3). We then present our framework in detail, describing how methods remove features (Section 4), formalizing the model behaviors analyzed by each method (Section 5), and examining each method's approach to summarizing feature influence (Section 6). We next explore connections to related fields. First, we describe our framework's relationship with cooperative game theory (Section 7). Next, we prove that under certain conditions, removal-based explanations analyze the information communicated by each feature (Section 8). Then, we refer to the psychology literature to establish a cognitive basis for removal-based explanations (Section 9). In our experiments, we provide empirical comparisons between existing methods and new combinations of existing approaches, as well as a discussion of our framework's implications for common explainability metrics (Section 10). Finally, we recap and conclude our discussion (Section 11).

2. Background

Here, we introduce the model explanation problem and briefly review existing approaches and related unification theories.

2.1 Preliminaries

Consider a supervised ML model f that is used to predict a response variable $Y \in \mathcal{Y}$ using the input $X = (X_1, X_2, \ldots, X_d)$, where each X_i represents an individual feature, such as a patient's age. We use uppercase symbols (e.g., X) to denote random variables and lowercase ones (e.g., x) to denote their values. We also use $\mathcal{X}$ to denote the domain of the full feature vector X and $\mathcal{X}_i$ to denote the domain of each feature X_i. Finally, $x_S \equiv \{x_i : i \in S\}$ denotes a subset of features for $S \subseteq D \equiv \{1, 2, \ldots d\}$, and $\bar{S} \equiv D \setminus S$ represents a set's complement.

ML interpretability broadly aims to provide insight into how models make predictions. This is particularly important when f is a complex model, such as a neural network or a decision forest. The most active area of recent research is *local interpretability*, which explains individual predictions, such as an individual patient diagnosis (e.g., Ribeiro et al., 2016; Lundberg and Lee, 2017; Sundararajan et al., 2017); in contrast, *global interpretability* explains the model's behavior across the entire dataset (e.g., Breiman, 2001; Owen, 2014; Covert et al., 2020). Both problems are usually addressed using *feature attribution*, where scores are assigned to quantify each feature's influence. However, recent work has also proposed the strategy of *local feature selection* (Chen et al., 2018a), and other papers have introduced methods to isolate sets of relevant features (Zhou et al., 2014; Fong and Vedaldi, 2017; Dabkowski and Gal, 2017).

Whether the aim is local or global interpretability, explaining the inner workings of complex models is fundamentally difficult, so it is no surprise that researchers continue to devise new approaches. Commonly cited categories of approaches include perturbation-based methods (e.g., Zeiler and Fergus, 2014; Lundberg and Lee, 2017), gradient-based methods (e.g., Simonyan et al., 2013; Montavon et al., 2017; Sundararajan et al., 2017; Selvaraju et al., 2017), more general propagation-based methods (e.g., Springenberg et al., 2014; Bach et al., 2015; Kindermans et al., 2017; Zhang et al., 2018), and inherently inter-

pretable models (e.g., Zhou et al., 2016; Rudin, 2019). However, these categories refer to loose collections of approaches that seldom share a precise mechanism.

Among the various approaches in the literature, many methods generate explanations by considering some class of perturbation to the input and the corresponding impact on the model's predictions. Certain methods consider infinitesimal perturbations by calculating gradients[2] (Simonyan et al., 2013; Sundararajan et al., 2017; Smilkov et al., 2017; Erion et al., 2019; Xu et al., 2020), but there are many possible input perturbations (Zeiler and Fergus, 2014; Ribeiro et al., 2016; Fong and Vedaldi, 2017; Lundberg and Lee, 2017). Our work is based on the observation that many perturbation strategies can be understood as simulating feature removal.

2.2 Related work

Prior work has made solid progress in exposing connections among disparate explanation methods. Lundberg and Lee (2017) proposed the unifying framework of *additive feature attribution methods* and showed that LIME, DeepLIFT, LRP and QII are all related to SHAP (Bach et al., 2015; Ribeiro et al., 2016; Shrikumar et al., 2016; Datta et al., 2016). Similarly, Ancona et al. (2017) showed that Grad * Input, DeepLIFT, LRP and Integrated Gradients can all be understood as modified gradient backpropagations. Most recently, Covert et al. (2020) showed that several global explanation methods can be viewed as *additive importance measures*, including permutation tests, Shapley Net Effects, feature ablation and SAGE (Breiman, 2001; Lipovetsky and Conklin, 2001; Lei et al., 2018).

Relative to prior work, the unification we propose is considerably broader but nonetheless precise. By focusing on the common mechanism of removing features from a model, we encompass far more methods, including both local and global ones. We also provide a considerably richer theoretical analysis by exploring underlying connections with cooperative game theory, information theory and cognitive psychology.

As we describe below, our framework characterizes methods along three dimensions. The choice of how to remove features has been considered by many works (Lundberg and Lee, 2017; Chang et al., 2018; Janzing et al., 2019; Sundararajan and Najmi, 2019; Merrick and Taly, 2019; Aas et al., 2019; Hooker and Mentch, 2019; Agarwal and Nguyen, 2019; Frye et al., 2020). However, the choice of what model behavior to analyze has been considered explicitly by only a few works (Lundberg et al., 2020; Covert et al., 2020), as has the choice of how to summarize each feature's influence based on a cooperative game (Štrumbelj et al., 2009; Datta et al., 2016; Lundberg and Lee, 2017; Frye et al., 2019; Covert et al., 2020). To our knowledge, ours is the first work to consider all three dimensions simultaneously and discuss them under a single unified framework.

Besides the methods that we focus on, there are also many methods that do not rely on the feature removal principle. We direct readers to survey articles for a broader overview of the literature (Adadi and Berrada, 2018; Guidotti et al., 2018).

3. Removal-Based Explanations

We now introduce our framework and briefly describe the methods it unifies.

2. A view also mentioned by Bhatt et al. (2020), for example.

Table 1: Choices made by existing removal-based explanations.

Method	Removal	Behavior	Summary
IME (2009)	Separate models	Prediction	Shapley value
IME (2010)	Marginalize (uniform)	Prediction	Shapley value
QII	Marginalize (marginals product)	Prediction	Shapley value
SHAP	Marginalize (conditional/marginal)	Prediction	Shapley value
KernelSHAP	Marginalize (marginal)	Prediction	Shapley value
TreeSHAP	Tree distribution	Prediction	Shapley value
LossSHAP	Marginalize (conditional)	Prediction loss	Shapley value
SAGE	Marginalize (conditional)	Dataset loss (label)	Shapley value
Shapley Net Effects	Separate models (linear)	Dataset loss (label)	Shapley value
SPVIM	Separate models	Dataset loss (label)	Shapley value
Shapley Effects	Marginalize (conditional)	Dataset loss (output)	Shapley value
Permutation Test	Marginalize (marginal)	Dataset loss (label)	Remove individual
Conditional Perm. Test	Marginalize (conditional)	Dataset loss (label)	Remove individual
Feature Ablation (LOCO)	Separate models	Dataset loss (label)	Remove individual
Univariate Predictors	Separate models	Dataset loss (label)	Include individual
L2X	Surrogate	Prediction loss (output)	High-value subset
REAL-X	Surrogate	Prediction loss (output)	High-value subset
INVASE	Missingness during training	Prediction mean loss	High-value subset
LIME (Images)	Default values	Prediction	Additive model
LIME (Tabular)	Marginalize (replacement dist.)	Prediction	Additive model
PredDiff	Marginalize (conditional)	Prediction	Remove individual
Occlusion	Zeros	Prediction	Remove individual
CXPlain	Zeros	Prediction loss	Remove individual
RISE	Zeros	Prediction	Mean when included
MM	Default values	Prediction	Partitioned subsets
MIR	Extend pixel values	Prediction	High-value subset
MP	Blurring	Prediction	Low-value subset
EP	Blurring	Prediction	High-value subset
FIDO-CA	Generative model	Prediction	High-value subset

3.1 A unified framework

We develop a unified model explanation framework by connecting methods that define each feature's influence through the impact of removing it from a model. This principle describes a substantial portion of the explainability literature: we find that 26 existing methods rely on this mechanism, including many of the most widely used approaches (Breiman, 2001; Ribeiro et al., 2016; Fong and Vedaldi, 2017; Lundberg and Lee, 2017). Several works have described their methods as either *removing*, *ignoring* or *deleting* information from a model, but our work is the first to precisely characterize this approach and document its use throughout the model explanation field.

The methods that we identify all remove groups of features from the model, but, beyond that, they take a diverse set of approaches. For example, LIME fits a linear model to an interpretable representation of the input (Ribeiro et al., 2016), L2X selects the most informative features for a single example (Chen et al., 2018a), and Shapley Effects examines how much of the model's variance is explained by each feature (Owen, 2014). Perhaps surprisingly, the differences between these methods are easy to systematize because they are all based on removing discrete sets of features.

As our main contribution, we introduce a framework that shows how these methods can be specified using only three choices.

Definition 1 Removal-based explanations *are model explanations that quantify the impact of removing groups of features from the model. These methods are determined by three choices:*

1. *(Feature removal) How the method removes features from the model (e.g. by setting them to default values, or by marginalizing over a distribution of values)*

2. *(Model behavior) What model behavior the method analyzes (e.g., the probability of the true class, or the model loss)*

3. *(Summary technique) How the method summarizes each feature's impact on the model (e.g., by removing a feature individually, or by calculating Shapley values)*

These three dimensions are independent of one another (i.e., any combination of choices is possible), but all three are necessary to fully specify a removal-based explanation. The first two choices, feature removal and model behavior, allow us to probe how a model's predictions change when given access to arbitrary subsets of features—including the behavior with all features, or with one or more features removed. Then, because there are an exponential number of feature combinations to consider, a summary technique is required to condense this information into a human-interpretable explanation, typically using either attribution scores or a subset of highly influential features.

As we show in the following sections, each dimension of the framework is represented by a specific mathematical choice. This precise yet flexible framework allows us to unify disparate classes of explanation methods, and, by unraveling each method's choices, offers a step towards a better understanding of the literature.

3.2 Overview of existing approaches

We now outline our findings, which we present in more detail in the remainder of the paper. In particular, we preview how existing methods fit into our framework and highlight groups of methods that appear closely related in light of our feature removal perspective.

Table 1 lists the methods unified by our framework, with acronyms and the original works introduced in Section 4. These methods represent diverse parts of the interpretability literature, including global interpretability methods (Breiman, 2001; Owen, 2014; Covert et al., 2020), computer vision-focused methods (Zeiler and Fergus, 2014; Zhou et al., 2014; Fong and Vedaldi, 2017; Petsiuk et al., 2018), game-theoretic methods (Štrumbelj and Kononenko, 2010; Datta et al., 2016; Lundberg and Lee, 2017) and feature selection methods (Chen et al., 2018a; Yoon et al., 2018; Fong et al., 2019). They are all unified by their reliance on feature removal, and they can be described concisely via their three choices within our framework.

Disentangling the details of each method shows that many approaches share one or more of the same choices. For example, most methods choose to explain individual predictions (the model behavior), and the Shapley value (Shapley, 1953) is the most popular summary technique. These common choices reveal that methods sometimes differ along only one or two dimensions, making it easier to reason about the trade-offs among approaches that might otherwise be viewed as monolithic algorithms.

Figure 2: Visual depiction of the space of removal-based explanations.

To further highlight similarities among these methods, we visually depict the space of removal-based explanations in Figure 2. Visualizing our framework reveals several regions in the space of methods that are crowded (e.g., methods that marginalize out removed features with their conditional distribution and that calculate Shapley values), while certain methods are relatively unique and spatially isolated (e.g., RISE; LIME for tabular data; INVASE). Empty positions in the grid reveal opportunities to develop new methods; in our experiments, we explore these possibilities by partially filling out the space of removal-based explanations (Section 10).

Table 2: Common combinations of choices in existing methods. Check marks (✓) indicate choices that are identical between methods.

Removal	Behavior	Summary	Methods
	✓	✓	IME, QII, SHAP, KernelSHAP, TreeSHAP
✓		✓	SHAP, LossSHAP, SAGE, Shapley Effects
✓	✓		Occlusion, LIME (images), MM, RISE
	✓	✓	Feature ablation (LOCO), permutation tests, conditional permutation tests
✓	✓		Univariate predictors, feature ablation (LOCO), Shapley Net Effects, SPVIM
	✓	✓	SAGE, Shapley Net Effects, SPVIM
✓	✓		SAGE, conditional permutation tests
✓		✓	Shapley Net Effects, SPVIM, IME (2009)
✓		✓	Occlusion, CXPlain
	✓	✓	Occlusion, PredDiff
✓		✓	Conditional permutation tests, PredDiff
✓	✓		SHAP, PredDiff
✓	✓		MP, EP
	✓	✓	EP, FIDO-CA
✓	✓	✓	L2X, REAL-X[3]
✓	✓	✓	Shapley Net Effects, SPVIM[4]

Finally, Table 2 shows groups of methods that differ in only one dimension of the framework. These methods are neighbors in the space of explanation methods (Figure 2), and it is noteworthy how many instances of neighboring methods exist in the literature. Certain methods even have neighbors along every dimension of the framework (e.g., SHAP, SAGE, Occlusion, PredDiff, conditional permutation tests), reflecting how intimately connected the field has become. In the remainder of the paper, after analyzing the choices made by each method, we consider perspectives from related fields that help reason about the conceptual and computational advantages that arise from the sometimes subtle differences between methods.

3. Although they share all three choices, L2X and REAL-X generate different explanations due to REAL-X's modified surrogate model training approach (Appendix A).
4. SPVIM generalizes Shapley Net Effects to black-box models by using an efficient Shapley value estimation technique (Appendix A).

4. Feature Removal

Here, we begin presenting our framework for removal-based explanations in detail. We first define the mathematical tools necessary to remove features from ML models, and we then examine how existing explanation methods remove features.

4.1 Functions on feature subsets

The principle behind removal-based explanations is to remove groups of features to understand their impact on a model, but since most models require all the features to make predictions, removing a feature is more complicated than simply not giving the model access to it. Most ML models make predictions given a specific set of features $X = (X_1, \ldots, X_d)$, and mathematically, these models are functions are of the form $f : \mathcal{X} \mapsto \mathcal{Y}$, where we use $\mathcal{F}$ to denote the set of all such possible mappings.

To remove features from a model, or to make predictions given a subset of features, we require a different mathematical object than $f \in \mathcal{F}$. Instead of functions with domain $\mathcal{X}$, we consider functions with domain $\mathcal{X} \times \mathcal{P}(D)$, where $\mathcal{P}(D)$ denotes the power set of $D \equiv \{1, \ldots, d\}$. To ensure invariance to the held out features, these functions must depend only on features specified by the subset $S \in \mathcal{P}(D)$, so we formalize *subset functions* as follows.

Definition 2 A **subset function** *is a mapping of the form*

$$F : \mathcal{X} \times \mathcal{P}(D) \mapsto \mathcal{Y}$$

that is invariant to the dimensions that are not in the specified subset. That is, we have $F(x, S) = F(x', S)$ *for all* (x, x', S) *such that* $x_S = x'_S$*. We define* $F(x_S) \equiv F(x, S)$ *for convenience because the held out values* $x_{\bar{S}}$ *are not used by* F*.*

A subset function's invariance property is crucial to ensure that only the specified feature values determine the function's output, while guaranteeing that the other feature values do not matter. Another way of viewing subset functions is that they provide an approach to accommodate missing data. While we use $\mathcal{F}$ to represent standard prediction functions, we use $\mathfrak{F}$ to denote the set of all possible subset functions.

We introduce subset functions here because they help conceptualize how different methods remove features from ML models. Removal-based explanations typically begin with an existing model $f \in \mathcal{F}$, and in order to quantify each feature's influence, they must establish a convention for removing it from the model. A natural approach is to define a subset function $F \in \mathfrak{F}$ based on the original model f. To formalize this idea, we define a model's *subset extension* as follows.

Definition 3 *An* **subset extension** *of a model* $f \in \mathcal{F}$ *is a subset function* $F \in \mathfrak{F}$ *that agrees with* f *in the presence of all features. That is, the model* f *and its subset extension* F *must satisfy*

$$F(x) = f(x) \quad \forall x \in \mathcal{X}.$$

As we show next, specifying a subset function $F \in \mathfrak{F}$, often as a subset extension of an existing model $f \in \mathcal{F}$, is the first step towards defining a removal-based explanation.

4.2 Removing features from machine learning models

Existing methods have devised numerous ways to evaluate models while withholding groups of features. Although certain methods use different terminology to describe their approaches (e.g., deleting information, ignoring features, using neutral values, etc.), the goal of all these methods is to measure a feature's influence through the impact of removing it from the model. Most proposed techniques can be understood as subset extensions $F \in \mathfrak{F}$ of an existing model $f \in \mathcal{F}$ (Definition 3).

The various approaches used in existing work (see Appendix A for more details) include:

- (Zeros) Occlusion (Zeiler and Fergus, 2014), RISE (Petsiuk et al., 2018) and CXPlain (Schwab and Karlen, 2019) remove features simply by setting them to zero:

$$F(x_S) = f(x_S, 0). \tag{1}$$

- (Default values) LIME for image data (Ribeiro et al., 2016) and the Masking Model method (MM, Dabkowski and Gal, 2017) remove features by setting them to user-defined default values (e.g., gray pixels for images). Given default values $r \in \mathcal{X}$, these methods calculate

$$F(x_S) = f(x_S, r_{\bar{S}}). \tag{2}$$

  Sometimes referred to as the *baseline* method (Sundararajan and Najmi, 2019), this is a generalization of the previous approach, and in some cases features may be given different default values (e.g., their mean).

- (Extend pixel values) Minimal image representation (MIR, Zhou et al., 2014) removes features from images by extending the values of neighboring pixels. This effect is achieved through a gradient-space manipulation.

- (Blurring) Meaningful Perturbations (MP, Fong and Vedaldi, 2017) and Extremal Perturbations (EP, Fong et al., 2019) remove features from images by blurring them with a Gaussian kernel. This approach is *not* a subset extension of f because the blurred image retains dependence on the removed features. Blurring fails to remove large, low frequency objects (e.g., mountains), but it provides an approximate way to remove information from images.

- (Generative model) FIDO-CA (Chang et al., 2018) removes features by replacing them with samples from a conditional generative model (e.g. Yu et al., 2018). The held out features are drawn from a generative model represented by $p_G(X_{\bar{S}} \mid X_S)$, or $\tilde{x}_{\bar{S}} \sim p_G(X_{\bar{S}} \mid X_S)$, and predictions are made as follows:

$$F(x_S) = f(x_S, \tilde{x}_{\bar{S}}). \tag{3}$$

- (Marginalize with conditional) SHAP (Lundberg and Lee, 2017), LossSHAP (Lundberg et al., 2020) and SAGE (Covert et al., 2020) present a strategy for removing features by marginalizing them out using their conditional distribution $p(X_{\bar{S}} \mid X_S = x_S)$:

$$F(x_S) = \mathbb{E}\big[f(X) \mid X_S = x_S\big]. \tag{4}$$

  This approach is computationally challenging in practice, but recent work tries to achieve close approximations (Aas et al., 2019, 2021; Frye et al., 2020). Shapley Effects (Owen, 2014) implicitly uses this convention to analyze function sensitivity, while conditional permutation tests (Strobl et al., 2008) and Prediction Difference Analysis (PredDiff, Zintgraf et al., 2017) propose simple approximations, with the latter conditioning only on groups of bordering pixels.

- (Marginalize with marginal) KernelSHAP (a practical implementation of SHAP) removes features by marginalizing them out using their joint marginal distribution $p(X_{\bar{S}})$:

$$F(x_S) = \mathbb{E}\big[f(x_S, X_{\bar{S}})\big]. \tag{5}$$

  This is the default behavior in SHAP's implementation,[5] and recent work discusses its potential benefits over conditional marginalization (Janzing et al., 2019). Permutation tests (Breiman, 2001) use this approach to remove individual features.

- (Marginalize with product of marginals) Quantitative Input Influence (QII, Datta et al., 2016) removes held out features by marginalizing them out using the product of the marginal distributions $p(X_i)$:

$$F(x_S) = \mathbb{E}_{\prod_{i \in D} p(X_i)}\big[f(x_S, X_{\bar{S}})\big]. \tag{6}$$

- (Marginalize with uniform) The updated version of the Interactions Method for Explanation (IME, Štrumbelj and Kononenko, 2010) removes features by marginalizing them out with a uniform distribution over the feature space. If we let $u_i(X_i)$ denote a uniform distribution over $\mathcal{X}_i$ (with extremal values defining the boundaries for continuous features), then features are removed as follows:

$$F(x_S) = \mathbb{E}_{\prod_{i \in D} u_i(X_i)}\big[f(x_S, X_{\bar{S}})\big]. \tag{7}$$

- (Marginalize with replacement distributions) LIME for tabular data replaces features with independent draws from *replacement distributions* (our term), each of which depends on the original feature values. When a feature X_i with value x_i is removed, discrete features are drawn from the distribution $p(X_i \mid X_i \neq x_i)$; when quantization is used for continuous features (LIME's default behavior[6]), continuous features are

5. `https://github.com/slundberg/shap`
6. `https://github.com/marcotcr/lime`

simulated by first generating a different quantile and then simulating from a truncated normal distribution within that bin. If we denote each feature's replacement distribution given the original value x_i as $q_{x_i}(X_i)$, then LIME for tabular data removes features as follows:

$$F(x, S) = \mathbb{E}_{\prod_{i \in D} q_{x_i}(X_i)}\big[f(x_S, X_{\bar{S}})\big]. \tag{8}$$

Although this function F agrees with f given all features, it is *not* a subset extension because it does not satisfy the invariance property necessary for subset functions.

- (Tree distribution) Dependent TreeSHAP (Lundberg et al., 2020) removes features using the distribution induced by the underlying tree model, which roughly approximates the conditional distribution. When splits for removed features are encountered in the model's trees, TreeSHAP averages predictions from the multiple paths in proportion to how often the dataset follows each path.

- (Surrogate model) Learning to Explain (L2X, Chen et al., 2018a) and REAL-X (Jethani et al., 2021a) train separate surrogate models F to match the original model's predictions when groups of features are held out. The surrogate model accommodates missing features, allowing us to represent it as a subset function $F \in \mathfrak{F}$, and it aims to provide the following approximation:

$$F(x_S) \approx \mathbb{E}\big[f(X) \mid X_S = x_S\big]. \tag{9}$$

The surrogate model approach was also proposed separately in the context of Shapley values (Frye et al., 2020).

- (Missingness during training) Instance-wise Variable Selection (INVASE, Yoon et al., 2018) uses a model that has missingness introduced at training time. Removed features are replaced with zeros, so that the model makes the following approximation:

$$F(x_S) = f(x_S, 0) \approx p(Y \mid X_S = x_S). \tag{10}$$

This approximation occurs for models trained cross entropy loss, but other loss functions may lead to different results (e.g., the conditional expectation for MSE loss). Introducing missingness during training differs from the default values approach because the model is trained to recognize zeros (or other replacement values) as missing values rather than zero-valued features.

- (Separate models) The original version of IME (Štrumbelj et al., 2009) is not based on a single model f, but rather on separate models trained for each feature subset, or $\{f_S : S \subseteq D\}$. The prediction for a subset of features is given by that subset's model:

$$F(x_S) = f_S(x_S). \tag{11}$$

Shapley Net Effects (Lipovetsky and Conklin, 2001) uses an identical approach in the context of linear models, with SPVIM generalizing the approach to black-box models

(Williamson and Feng, 2020). Similarly, feature ablation, also known as leave-one-covariate-out (LOCO, Lei et al., 2018), trains models to remove individual features, and the univariate predictors approach (used mainly for feature selection) uses models trained with individual features (Guyon and Elisseeff, 2003). Although the separate models approach is technically a subset extension of the model f_D trained with all features, its predictions given subsets of features are not based on f_D.

Most of these approaches can be viewed as subset extensions of an existing model f, so our formalisms provide useful tools for understanding how removal-based explanations remove features from models. However, there are two exceptions: the blurring technique (MP and EP) and LIME's approach with tabular data. Both provide functions of the form $F : \mathcal{X} \times \mathcal{P}(D) \mapsto \mathcal{Y}$ that agree with f given all features, but that still exhibit dependence on removed features. Based on our invariance property for held out features (Definition 2), we argue that these approaches do not fully remove features from the model.

We conclude that the first dimension of our framework amounts to choosing a subset function $F \in \mathfrak{F}$, often via a subset extension to an existing model $f \in \mathcal{F}$. We defer consideration of the trade-offs between these approaches until Section 8, where we show that one approach to removing features yields to connections with information theory.

5. Explaining Different Model Behaviors

Removal-based explanations all aim to demonstrate how a model functions, but they can do so by analyzing different model behaviors. We now consider the various choices of target quantities to observe as different features are withheld from the model.

The feature removal principle is flexible enough to explain virtually any function. For example, methods can explain a model's prediction, a model's loss function, a hidden layer in a neural network, or any node in a computation graph. In fact, removal-based explanations need not be restricted to the ML context: any function that accommodates missing inputs can be explained via feature removal by observing either its output or some function of its output as groups of inputs are removed. This perspective shows the broad potential applications for removal-based explanations.

Because our focus is the ML context, we proceed by examining how existing methods work. Each explanation method's target quantity can be understood as a function of the model output, which for simplicity is represented by a subset function $F(x_S)$. Many methods explain the model output or a simple function of the output, such as the logits or log-odds ratio. Other methods take into account a measure of the model's loss, for either an individual input or the entire dataset. Ultimately, as we show below, each method generates explanations based on a set function of the form

$$u : \mathcal{P}(D) \mapsto \mathbb{R}, \tag{12}$$

which represents a value associated with each subset of features $S \subseteq D$. This set function corresponds to the model behavior that a method is designed to explain.

We now examine the specific choices made by existing methods (see Appendix A for further details). The various model behaviors that methods analyze, and their corresponding set functions, include:

- (Prediction) Occlusion, RISE, PredDiff, MP, EP, MM, FIDO-CA, MIR, LIME, SHAP (including KernelSHAP and TreeSHAP), IME and QII all analyze a model's prediction for an individual input $x \in \mathcal{X}$:

$$u_x(S) = F(x_S). \tag{13}$$

  These methods examine how holding out different features makes an individual prediction either higher or lower. For multi-class classification models, methods often use a single output that corresponds to the class of interest, and they can optionally apply a simple function to the model's output (for example, using the log-odds ratio rather than the classification probability).

- (Prediction loss) LossSHAP and CXPlain take into account the true label y for an input x and calculate the prediction loss using a loss function ℓ:

$$v_{xy}(S) = -\ell\big(F(x_S), y\big). \tag{14}$$

  By incorporating the label, these methods quantify whether certain features make the prediction more or less correct. Note that the minus sign is necessary to give the set function a higher value when more informative features are included.

- (Prediction mean loss) INVASE considers the expected loss for a given input x according to the label's conditional distribution $p(Y \mid X = x)$:

$$v_x(S) = -\mathbb{E}_{p(Y|X=x)}\Big[\ell\big(F(x_S), Y\big)\Big]. \tag{15}$$

  By averaging the loss across the label's distribution, INVASE highlights features that correctly predict what *could* have occurred, on average.

- (Dataset loss) Shapley Net Effects, SAGE, SPVIM, feature ablation, permutation tests and univariate predictors consider the expected loss across the entire dataset:

$$v(S) = -\mathbb{E}_{XY}\Big[\ell\big(F(X_S), Y\big)\Big]. \tag{16}$$

  These methods quantify how much the model's performance degrades when different features are removed. This set function can also be viewed as the predictive power derived from sets of features (Covert et al., 2020), and recent work has proposed a SHAP value aggregation that is a special case of this approach (Frye et al., 2020).

- (Prediction loss w.r.t. output) L2X and REAL-X consider the loss between the full model output and the prediction given a subset of features:

$$w_x(S) = -\ell\big(F(x_S), F(x)\big). \tag{17}$$

  These methods highlight features that on their own lead to similar predictions as the full feature set.

- (Dataset loss w.r.t. output) Shapley Effects considers the expected loss with respect to the full model output:

$$w(S) = -\mathbb{E}_X\Big[\ell\big(F(X_S), F(X)\big)\Big]. \tag{18}$$

  Though related to the dataset loss approach (Covert et al., 2020), this approach focuses on each feature's influence on the model output rather than on the model performance.

Each set function serves a distinct purpose in exposing a model's dependence on different features. Several of the approaches listed above analyze the model's behavior for individual predictions (local explanations), while some take into account the model's behavior across the entire dataset (global explanations). Although their aims differ, these set functions are all in fact related. Each builds upon the previous ones by accounting for either the loss or data distribution, and their relationships can be summarized as follows:

$$v_{xy}(S) = -\ell\big(u_x(S), y\big) \tag{19}$$
$$w_x(S) = -\ell\big(u_x(S), u_x(D)\big) \tag{20}$$
$$v_x(S) = \mathbb{E}_{p(Y|X=x)}\big[v_{xY}(S)\big] \tag{21}$$
$$v(S) = \mathbb{E}_{XY}\big[v_{XY}(S)\big] \tag{22}$$
$$w(S) = \mathbb{E}_X\big[w_X(S)\big] \tag{23}$$

These relationships show that explanations based on one set function are in some cases related to explanations based on another. For example, Covert et al. (2020) showed that SAGE explanations are the expectation of explanations provided by LossSHAP—a relationship reflected in Eq. 22.

Understanding these connections is made easier by the fact that our framework disentangles each method's choices rather than viewing each method as a monolithic algorithm. We conclude by reiterating that removal-based explanations can explain virtually any function, and that choosing what model behavior to explain amounts to selecting a set function $u : \mathcal{P}(D) \mapsto \mathbb{R}$ to represent the model's dependence on different sets of features.

6. Summarizing Feature Influence

The third choice for removal-based explanations is how to summarize each feature's influence on the model. We examine the various summarization techniques and then discuss their computational complexity and approximation approaches.

6.1 Explaining set functions

The set functions that represent a model's dependence on different features (Section 5) are complicated mathematical objects that are difficult to communicate to users: the model's behavior can be observed for any subset of features, but there are an exponential number of feature subsets to consider. Removal-based explanations handle this challenge by providing users with a concise summary of each feature's influence.

We distinguish between two main types of summarization approaches: *feature attribution* and *feature selection*. Many methods provide explanations in the form of feature attributions, which are numerical scores $a_i \in \mathbb{R}$ given to each feature $i = 1, \ldots, d$. If we use $\mathcal{U}$ to denote the set of all functions $u : \mathcal{P}(D) \mapsto \mathbb{R}$, then we can represent feature attributions as mappings of the form $E : \mathcal{U} \mapsto \mathbb{R}^d$, which we refer to as *explanation mappings*. Other methods take the alternative approach of summarizing set functions with a set $S^* \subseteq D$ of the most influential features. We represent these feature selection summaries as explanation mappings of the form $E : \mathcal{U} \mapsto \mathcal{P}(D)$. Both approaches provide users with simple summaries of a feature's contribution to the set function.

We now consider the specific choices made by each method (see Appendix A for further details). For simplicity, we let u denote the set function each method analyzes. Surveying the various removal-based explanation methods, the techniques for summarizing each feature's influence include:

- (Remove individual) Occlusion, PredDiff, CXPlain, permutation tests and feature ablation (LOCO) calculate the impact of removing a single feature from the model, resulting in the following attribution values:

$$a_i = u(D) - u(D \setminus \{i\}). \tag{24}$$

  Occlusion, PredDiff and CXPlain can also be applied with groups of features, or superpixels, in image contexts.

- (Include individual) The univariate predictors approach calculates the impact of including individual features, resulting in the following attribution values:

$$a_i = u(\{i\}) - u(\{\}). \tag{25}$$

  This is essentially the reverse of the previous approach: rather than removing individual features from the complete set, this approach adds individual features to the empty set.

- (Additive model) LIME fits a regularized additive model to a dataset of perturbed examples. In the limit of an infinite number of samples, this process approximates the following attribution values:

$$a_1, \ldots, a_d = \underset{b_0,\ldots,b_d}{\arg\min} \sum_{S \subseteq D} \pi(S)\Big(b_0 + \sum_{i \in S} b_i - u(S)\Big)^2 + \Omega(b_1, \ldots, b_d). \tag{26}$$

  In this problem, π represents a weighting kernel and Ω is a regularization function that is often set to the ℓ_1 penalty to encourage sparse attributions (Tibshirani, 1996). Since this summary is based on an additive model, the learned coefficients $(a_1, \ldots, a_d)$ represent the incremental value associated with including each feature.

- (Mean when included) RISE determines feature attributions by sampling many subsets $S \subseteq D$ and then calculating the mean value when a feature is included. Denoting the

distribution of subsets as $p(S)$ and the conditional distribution as $p(S \mid i \in S)$, the attribution values are defined as

$$a_i = \mathbb{E}_{p(S|i \in S)}\big[u(S)\big]. \tag{27}$$

In practice, RISE samples the subsets $S \subseteq D$ by removing each feature i independently with probability p, using $p = 0.5$ in their experiments (Petsiuk et al., 2018).

- (Shapley value) Shapley Net Effects, IME, Shapley Effects, QII, SHAP (including KernelSHAP, TreeSHAP and LossSHAP), SPVIM and SAGE all calculate feature attributions using the Shapley value, which we denote as $a_i = \phi_i(u)$. Described in more detail in Section 7, Shapley values are the only attributions that satisfy several desirable properties.

- (Low-value subset) MP selects a small set of features S^* that can be removed to give the set function a low value. It does so by solving the following optimization problem:

$$S^* = \arg\min_S \; u(D \setminus S) + \lambda |S|. \tag{28}$$

In practice, MP incorporates additional regularizers and solves a relaxed version of this problem (see Section 6.2).

- (High-value subset) MIR solves an optimization problem to select a small set of features S^* that alone can give the set function a high value. For a user-defined minimum value t, the problem is given by:

$$S^* = \arg\min_S \; |S| \quad \text{s.}t. \;\; u(S) \geq t. \tag{29}$$

L2X and EP solve a similar problem but switch the terms in the constraint and optimization objective. For a user-defined subset size k, the optimization problem is given by:

$$S^* = \arg\max_S \; u(S) \quad \text{s.t.} \;\; |S| = k. \tag{30}$$

Finally, INVASE, REAL-X and FIDO-CA solve a regularized version of the problem with a parameter $\lambda > 0$ controlling the trade-off between the subset value and subset size:

$$S^* = \arg\max_S \; u(S) - \lambda |S|. \tag{31}$$

- (Partitioned subsets) MM solves an optimization problem to partition the features into S^* and $D \setminus S^*$ while maximizing the difference in the set function's values. This approach is based on the idea that removing features to find a low-value subset (as in MP) and retaining features to get a high-value subset (as in MIR, L2X, EP, INVASE,

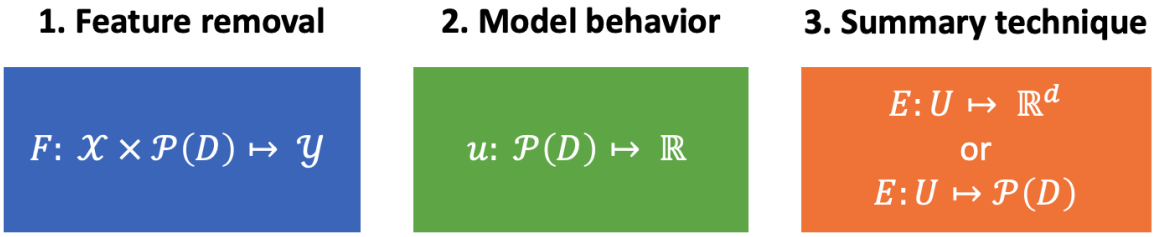

Figure 3: Removal-based explanations are specified by three precise mathematical choices: a subset function $F \in \mathfrak{F}$, a set function $u \in \mathcal{U}$, and an explanation mapping E (for feature attribution or selection).

REAL-X and FIDO-CA) are both reasonable approaches for identifying influential features. The problem is given by:

$$S^* = \arg\max_S \; u(S) - \gamma u(D \setminus S) - \lambda |S|. \tag{32}$$

In practice, MM also incorporates regularizers and monotonic link functions to enable a more flexible trade-off between $u(S)$ and $u(D \setminus S)$ (see Appendix A).

As this discussion shows, every removal-based explanation generates summaries of each feature's influence on the underlying set function. In general, a model's dependencies are too complex to communicate fully, so explanations must provide users with a concise summary instead. Feature attributions provide a granular view of each feature's influence on the model, while feature selection summaries can be understood as coarse attributions that assign binary rather than real-valued importance.

Interestingly, if the high-value subset optimization problems solved by MIR, L2X, EP, INVASE, REAL-X and FIDO-CA were applied to the set function that represents the dataset loss (Eq. 22), they would resemble conventional global feature selection problems (Guyon and Elisseeff, 2003). The problem in Eq. 30 determines the set of k features with maximum predictive power, the problem in Eq. 29 determines the smallest set of features that achieve the performance given by t, and the problem in Eq. 31 uses a parameter λ to control the trade-off. Though not generally viewed as a model explanation approach, global feature selection serves an identical purpose of identifying highly predictive features.

We conclude by reiterating that the third dimension of our framework amounts to a choice of explanation mapping, which takes the form $E : \mathcal{U} \mapsto \mathbb{R}^d$ for feature attribution or $E : \mathcal{U} \mapsto \mathcal{P}(D)$ for feature selection. Our discussion so far has shown that removal-based explanations can be specified using three precise mathematical choices, as depicted in Figure 3. These methods, which are often presented in ways that make their connections difficult to discern, are constructed in a remarkably similar fashion. The remainder of this work addresses the relationships and trade-offs among these different choices, beginning by examining the computational complexity of each summarization approach.

6.2 Complexity and approximations

Showing how certain explanation methods fit into our framework requires distinguishing between their implicit objectives and the approximations that make them practical. Our presentation of these methods deviates from the original papers, which often focus on details of a method's implementation. We now bridge the gap by describing these methods' computational complexity and the approximations that are sometimes used out of necessity.

The challenge with most of the summarization techniques described above is that they require calculating the underlying set function's value $u(S)$ for many subsets of features. In fact, without making any simplifying assumptions about the model or data distribution, several techniques must examine all 2^d subsets of features. This includes the Shapley value, RISE's summarization technique and LIME's additive model. Finding exact solutions to several of the optimization problems (MP, MIR, MM, INVASE, REAL-X, FIDO-CA) also requires examining all subsets of features, and solving the constrained optimization problem (EP, L2X) for k features requires examining $\binom{d}{k}$ subsets, or $\mathcal{O}(2^d d^{-\frac{1}{2}})$ subsets in the worst case.[7]

The only approaches with lower computational complexity are those that remove individual features (Occlusion, PredDiff, CXPlain, permutation tests, feature ablation) or include individual features (univariate predictors). These require only one subset per feature, or d total feature subsets.

Many summarization techniques have superpolynomial complexity in d, making them intractable for large numbers of features, at least with naïve implementations. These methods work in practice due to fast approximation approaches, and in some cases techniques have even been devised to generate real-time explanations. Several strategies that yield fast approximations include:

- Attribution values that are the expectation of a random variable can be estimated by Monte Carlo approximation. IME (Štrumbelj and Kononenko, 2010), Shapley Effects (Song et al., 2016) and SAGE (Covert et al., 2020) use sampling strategies to approximate Shapley values, and RISE also estimates its attributions via sampling (Petsiuk et al., 2018).

- KernelSHAP, LIME and SPVIM are based on linear regression models fitted to datasets containing an exponential number of datapoints. In practice, these techniques fit models to smaller sampled datasets, which means optimizing an approximate version of their objective function (Lundberg and Lee, 2017; Covert and Lee, 2021).

- TreeSHAP calculates Shapley values in polynomial time using a dynamic programming algorithm that exploits the structure of tree-based models. Similarly, L-Shapley and C-Shapley exploit the properties of models for structured data to provide fast Shapley value approximations (Chen et al., 2018b).

- Several of the feature selection methods (MP, L2X, REAL-X, EP, MM, FIDO-CA) solve continuous relaxations of their discrete optimization problems. While these optimization problems can be solved by representing the set of features $S \subseteq D$ as a

7. This can be seen by applying Stirling's approximation to $\binom{d}{d/2}$ as d becomes large.

mask $m \in \{0,1\}^d$, these methods instead use a continuous mask variable of the form $m \in [0,1]^d$. When these methods incorporate a penalty on the subset size $|S|$, they also sometimes use the convex relaxation $||m||_1$.

- One feature selection method (MIR) uses a greedy optimization algorithm. MIR determines a set of influential features $S \subseteq D$ by iteratively removing groups of features that do not reduce the predicted probability for the correct class.

- One feature attribution method (CXPlain) and several feature selection methods (L2X, INVASE, REAL-X, MM) generate real-time explanations by learning separate explainer models. CXPlain learns an explainer model using a dataset consisting of manually calculated explanations, which removes the need to iterate over each feature when generating new explanations. L2X learns a model that outputs a set of features (represented by a k-hot vector) and INVASE/REAL-X learn similar selector models that can output arbitrary numbers of features. Similarly, MM learns a model that outputs masks of the form $m \in [0,1]^d$ for images. These techniques can be viewed as *amortized* approaches because they learn models that perform the summarization step in a single forward pass.

In conclusion, many methods have developed approximations that enable efficient model explanation, despite sometimes using summarization techniques that are inherently intractable (e.g., Shapley values). Certain techniques are considerably faster than others (i.e., the amortized approaches), and some can trade off computational cost for approximation accuracy (Štrumbelj and Kononenko, 2010; Covert and Lee, 2021), but they are all sufficiently fast to be used in practice.

We speculate, however, that more approaches will be made to run in real-time by learning separate explainer models, as in the MM, L2X, INVASE, CXPlain and REAL-X approaches (Dabkowski and Gal, 2017; Chen et al., 2018a; Yoon et al., 2018; Schwab and Karlen, 2019; Jethani et al., 2021a). Besides these methods, others have been proposed that learn the explanation process either as a component of the original model (Fan et al., 2017; Taghanaki et al., 2019) or as a separate model after training (Schulz et al., 2020). Such approaches may be necessary to bypass the need for multiple model evaluations and make removal-based explanations as fast as gradient-based and propagation-based methods.[8]

7. Game-Theoretic Explanations

The set functions analyzed by removal-based explanations (Section 5) can be viewed as *cooperative games*—mathematical objects studied by cooperative game theory. Only a few explanation methods explicitly consider game-theoretic connections, but we show that every method described thus far can be understood through the lens of cooperative game theory.

7.1 Cooperative game theory background

Cooperative games are functions of the form $u : \mathcal{P}(D) \mapsto \mathbb{R}$ (i.e., set functions) that describe the value achieved when sets of players $S \subseteq D$ participate in a game. Intuitively, a game

8. While this work was under review, Jethani et al. (2021b) introduced an amortized approach for real-time Shapley value estimation.

might represent the profit made when a particular group of employees chooses to work together. Cooperative game theory research focuses on understanding the properties of different payoffs that can be offered to incentivize participation in the game, as well as predicting which groups of players will ultimately agree to participate.

For this discussion, we use u to denote a cooperative game. To introduce terminology from cooperative game theory, the features $i = 1, 2, \dots d$ are referred to as *players*, sets of players $S \subseteq D$ are referred to as *coalitions*, and the output $u(S)$ is referred to as the *value* of S. Player i's *marginal contribution* to the coalition $S \in D \setminus \{i\}$ is defined as the difference in value $u(S \cup \{i\}) - u(S)$. *Allocations* are vectors $z \in \mathbb{R}^d$ that represent payoffs proposed to each player in return for participating in the game.

Several fundamental concepts in cooperative game theory are related to the properties of allocations: the *core* of a game, a game's *nucleolus*, and its *bargaining sets* are all based on whether players view certain allocations as favorable (Narahari, 2014). Perhaps surprisingly, every summarization technique used by removal-based explanations (Section 6) can be viewed in terms of allocations to players in the underlying game, enabling us to connect these explanation methods to ideas from the game theory literature.

7.2 Allocation strategies

Several summarization techniques used by removal-based explanation methods are related to *solution concepts*, which in the cooperative game theory context are allocation strategies designed to be fair to the players. If we let $\mathcal{U}$ represent the set of all cooperative games with d players, then solution concepts are represented by mappings of the form $E : \mathcal{U} \mapsto \mathbb{R}^d$, similar to explanation mappings that represent feature attributions (Section 6.1).

We first discuss the Shapley value, which assumes that the *grand coalition* (the coalition containing all players) is participating and distributes the total value in proportion to each player's contributions (Shapley, 1953). The Shapley value can be derived axiomatically, and we list several of its properties below; to highlight its many desirable properties, we provide more axioms than are necessary to derive the Shapley value uniquely. The Shapley values $\phi_i(u) \in \mathbb{R}$ for a game u are the unique allocations that satisfy the following properties:

- (Efficiency) The allocations $\phi_1(u), \dots, \phi_d(u)$ add up to the difference in value between the grand coalition and the empty coalition:

$$\sum_{i \in D} \phi_i(u) = u(D) - u(\{\}).$$

- (Symmetry) Two players i, j that make equal marginal contributions to all coalitions receive the same allocation:

$$u(S \cup \{i\}) = u(S \cup \{j\}) \;\; \forall \;\; S \implies \phi_i(u) = \phi_j(u).$$

- (Dummy) A player i that makes zero marginal contribution receives zero allocation:

$$u(S \cup \{i\}) = u(S) \;\; \forall \;\; S \implies \phi_i(u) = 0.$$

- (Additivity) If we consider two games u, u' and their respective allocations $\phi_i(u)$ and $\phi_i(u')$, then the cooperative game defined as their sum $u + u'$ has allocations defined as the sum of each game's allocations:

$$\phi_i(u + u') = \phi_i(u) + \phi_i(u') \ \ \forall \ \ i.$$

- (Marginalism) For two games u, u' where all players have identical marginal contributions, the players receive equal allocations:

$$u(S \cup \{i\}) - u(S) = u'(S \cup \{i\}) - u'(S) \ \ \forall \ \ (i, S) \implies \phi_i(u) = \phi_i(u') \ \ \forall \ \ i.$$

The Shapley values are the unique allocations that satisfy these properties (Shapley, 1953; Monderer et al., 2002), and the expression for each Shapley value $\phi_i(u)$ is

$$\phi_i(u) = \frac{1}{d} \sum_{S \subseteq D \setminus \{i\}} \binom{d-1}{|S|}^{-1} \Big(u(S \cup \{i\}) - u(S) \Big). \tag{33}$$

The Shapley value has found widespread use because of its axiomatic derivation, both within game theory and in other disciplines (Aumann, 1994; Shorrocks, 1999; Petrosjan and Zaccour, 2003; Tarashev et al., 2016). In the context of model explanation, Shapley values define each feature's contribution while accounting for complex feature interactions, such as correlations, redundancy, and complementary behavior (Lipovetsky and Conklin, 2001; Štrumbelj et al., 2009; Owen, 2014; Datta et al., 2016; Lundberg and Lee, 2017; Lundberg et al., 2020; Covert et al., 2020; Williamson and Feng, 2020).

Like the Shapley value, the Banzhaf value attempts to define fair allocations for each player in a cooperative game. It generalizes the Banzhaf power index, which is a technique for measuring the impact of players in the context of voting games (Banzhaf, 1964). Links between Shapley and Banzhaf values are described by Dubey and Shapley (1979), who show that the Shapley value can be understood as an enumeration over all *permutations* of players, while the Banzhaf value can be understood as an enumeration over all *subsets* of players. The expression for each Banzhaf value $\psi_i(u)$ is:

$$\psi_i(u) = \frac{1}{2^{d-1}} \sum_{S \subseteq D \setminus \{i\}} \Big(u(S \cup \{i\}) - u(S) \Big). \tag{34}$$

The Banzhaf value fails to satisfy the Shapley value's efficiency property, but it can be derived axiomatically by introducing a variation on the efficiency axiom (Nowak, 1997):

- (2-Efficiency) If two players i, j merge into a new player $\{i, j\}$ and we re-define the game u as a game u' on the smaller set of players $\big(D \setminus \{i, j\}\big) \cup \big\{\{i, j\}\big\}$, then the Banzhaf values satisfy the following property:

$$\psi_{\{i,j\}}(u') = \psi_i(u) + \psi_j(u). \tag{35}$$

The 2-efficiency property roughly states that credit allocation is immune to the merging of players. The Banzhaf value has multiple interpretations, but the most useful for our purpose is that it represents the difference between the mean value of coalitions that do and do not include the ith player when coalitions are chosen uniformly at random. With this perspective, we observe that the RISE summarization technique is closely related to the Banzhaf value: RISE calculates the mean value of coalitions that include i, but, unlike the Banzhaf value, it disregards the value of coalitions that do not include i. While the RISE technique (Petsiuk et al., 2018) was not motivated by the Banzhaf value, it is unsurprising that such a natural idea has been explored in cooperative game theory.

Both Shapley and Banzhaf values are mathematically appealing because they can be understood as the weighted average of a player's marginal contributions (Eq. 33-34). This is a stronger version of the marginalism property introduced by Young's axiomatization of the Shapley value (Young, 1985), and solution concepts of this form are known as *probabilistic values* (Weber, 1988). Probabilistic values have their own axiomatic characterization: they have been shown to be the unique values that satisfy a specific subset of the Shapley value's properties (Monderer et al., 2002).

The notion of probabilistic values reveals links with two other solution concepts. The techniques of removing individual players (e.g., Occlusion, PredDiff, CXPlain, permutation tests and feature ablation) and including individual players (e.g., univariate predictors) can also be understood as probabilistic values, although they are simple averages that put all their weight on a single marginal contribution (Eqs. 24-25). Unlike the Shapley and Banzhaf values, these methods neglect complex interactions when quantifying each player's contribution.

The methods discussed thus far (Shapley value, Banzhaf value, removing individual players, including individual players) all satisfy the symmetry, dummy, additivity, and marginalism axioms. What makes the Shapley value unique among these approaches is the efficiency axiom, which lets us view it as a distribution of the grand coalition's value among the players (Dubey and Shapley, 1979). However, whether the efficiency property or the Banzhaf value's 2-efficiency property is preferable may depend on the use-case.

7.3 Modeling cooperative games

Unlike the methods discussed so far, LIME provides a flexible approach for summarizing each player's influence. As its summarization technique, LIME fits an additive model to the cooperative game (Eq. 26), leaving the user the option of specifying a weighting kernel π and regularization term Ω for the weighted least squares objective.

Although fitting a model to a cooperative game seems distinct from the allocation strategies discussed so far, these ideas are in fact intimately connected. Fitting models to cooperative games, including both linear and nonlinear models, has been studied by numerous works in cooperative game theory (Charnes et al., 1988; Hammer and Holzman, 1992; Grabisch et al., 2000; Ding et al., 2008, 2010; Marichal and Mathonet, 2011), and specific choices for the weighting kernel can yield recognizable attribution values.

If we fit an additive model to the underlying game with no regularization ($\Omega = 0$), then we can identify several weighting kernels that correspond to other summarization techniques:

- When we use the weighting kernel $\pi_{\text{Rem}}(S) = \mathbb{1}(|S| \geq d-1)$, where $\mathbb{1}(\cdot)$ is an indicator function, the attribution values are the marginal contributions from removing individual players from the grand coalition, or the values $a_i = u(D) - u(D \setminus \{i\})$. This is the summarization technique used by Occlusion, PredDiff, CXPlain, permutation tests, and feature ablation.
- When we use the weighting kernel $\pi_{\text{Inc}}(S) = \mathbb{1}(|S| \leq 1)$, the attribution values are the marginal contributions from adding individual players to the empty coalition, or the values $a_i = u(\{i\}) - u(\{\})$. This is the summarization technique used by the univariate predictors approach.
- Results from Hammer and Holzman (1992) show that when we use the weighting kernel $\pi_{\text{B}}(S) = 1$, the attribution values are the Banzhaf values $a_i = \psi_i(u)$.
- Results from Charnes et al. (1988) and Lundberg and Lee (2017) show that the attribution values are equal to the Shapley values $a_i = \phi_i(u)$ when we use the weighting kernel π_{Sh}, defined as follows:

$$\pi_{\text{Sh}}(S) = \frac{d-1}{\binom{d}{|S|}|S|(d-|S|)}. \tag{36}$$

Although the Shapley value connection has been noted in the model explanation context, the other results we present are new observations about LIME (proofs in Appendix B). These results show that the weighted least squares problem solved by LIME provides sufficient flexibility to yield both the Shapley and Banzhaf values, as well as simpler quantities such as the marginal contributions from removing or including individual players. The additive model approach captures every feature attribution technique discussed so far, with the caveat that RISE uses a modified version of the Banzhaf value. And, like the other allocation strategies, attributions arising from fitting an additive model can be shown to satisfy different sets of Shapley axioms (Appendix C).

We have thus demonstrated that the additive model approach for summarizing feature influence not only has precedent in cooperative game theory, but that it is intimately connected to the other allocation strategies. This suggests that the feature attributions generated by many removal-based explanations can be understood as additive decompositions of the underlying cooperative game.

7.4 Identifying coalitions using excess

To better understand removal-based explanations that perform feature selection, we look to a more basic concept in cooperative game theory: the notion of *excess*. Excess is defined for a given allocation $z \in \mathbb{R}^d$ and coalition $S \subseteq D$ as the difference between the coalition's value and its total allocation (Narahari, 2014). More formally, it is defined as follows.

Definition 4 *Given a cooperative game $u : \mathcal{P}(D) \mapsto \mathbb{R}$, a coalition $S \subseteq D$, and an allocation $z = (z_1, z_2, \ldots, z_d) \in \mathbb{R}^d$, the* **excess** *of S at z is defined as*

$$e(S, z) = u(S) - \sum_{i \in S} z_i. \tag{37}$$

The excess $e(S, z)$ represents the coalition's degree of unhappiness under the allocation z, because an allocation is unfavorable to a coalition if its value $u(S)$ exceeds its cumulative allocation $\sum_{i \in S} z_i$. For cooperative game theory concepts including the core, the nucleolus and bargaining sets, a basic assumption is that coalitions with higher excess (greater unhappiness) are more likely to break off and refuse participation in the game (Narahari, 2014). We have no analogue for features refusing participation in ML, but the notion of excess is useful for understanding several removal-based explanation methods.

Below, we show that removal-based explanations that select sets of influential features are equivalent to proposing equal allocations to all players and then determining a high-valued coalition by examining each coalition's excess. Given equal allocations, players with high contributions will be less satisfied than players with low contributions, so solving this problem leads to a set of high- and low-valued players. We show this by reformulating each method's optimization problem as follows:

- (Minimize excess) MP isolates a low-value coalition by finding the coalition with the lowest excess, or the highest satisfaction, given equal allocations $z = \mathbf{1}\lambda$:

$$S^* = \arg\min_S \ e(\bar{S}, \mathbf{1}\lambda). \tag{38}$$

  The result S^* is a coalition whose complement $D \setminus S^*$ is most satisfied with the equal allocations.

- (Maximize excess) MIR isolates the smallest possible coalition that achieves a sufficient level of excess given allocations equal to zero. For a level of excess t, the optimization problem is given by:

$$S^* = \arg\min_S \ |S| \quad \text{s.t.} \ \ e(S, \mathbf{0}) \geq t. \tag{39}$$

  L2X and EP solve a similar problem but switch the objective and constraint. The optimization problem represents the coalition of size k with the highest excess given allocations of zero to each player:

$$S^* = \arg\max_S \ e(S, \mathbf{0}) \quad \text{s.t.} \ \ |S| = k. \tag{40}$$

  Finally, FIDO-CA, INVASE and REAL-X find the coalition with the highest excess given equal allocations $z = \mathbf{1}\lambda$:

$$S^* = \arg\max_S \ e(S, \mathbf{1}\lambda). \tag{41}$$

- (Maximize difference in excess) MM combines the previous approaches. Rather than focusing on a coalition with low or high excess, MM partitions players into two coalitions while maximizing the difference in excess given equal allocations $z = \mathbf{1}\frac{\lambda}{1+\gamma}$:

$$S^* = \arg\max_S \ e\Big(S, \mathbf{1}\frac{\lambda}{1+\gamma}\Big) - \gamma e\Big(\bar{S}, \mathbf{1}\frac{\lambda}{1+\gamma}\Big). \tag{42}$$

The result S^* is a coalition of dissatisfied players whose complement $D \setminus S^*$ is comparably more satisfied.

All of the approaches listed above are different formulations of the same multi-objective optimization problem: the intuition is that there is a small set of high-valued players S and a comparably larger set of low-valued players $D \setminus S$. Most methods focus on just one of these coalitions (MP focuses on $D \setminus S$, and MIR, L2X, EP, INVASE, REAL-X and FIDO-CA focus on S), while the optimization problem solved by MM considers both coalitions.

MM's summarization technique can therefore be understood as a generalization of the other methods. The MP and INVASE/REAL-X/FIDO-CA problems (Eqs. 38, 41), for example, are special cases of the MM problem. The other problems (Eq. 39, 40) cannot necessarily be cast as special cases of the MM problem (Eq. 42), but the MM problem resembles the Lagrangians of these constrained problems; more precisely, a special case of the MM problem shares the same optimal coalition with the dual to these constrained problems (Boyd et al., 2004).

By reformulating the optimization problems solved by each method, we can see that each feature selection summarization technique can be described as minimizing or maximizing excess given equal allocations for all players. In cooperative game theory, examining each coalition's level of excess (or dissatisfaction) helps determine whether allocations will incentivize participation, and the same tool is used by removal-based explanations to find the most influential features for a model.

These feature selection approaches can be viewed as mappings of the form $E : \mathcal{U} \mapsto \mathcal{P}(D)$ because they identify coalitions $S^* \subseteq D$. Although the Shapley axioms apply only to mappings of the form $E : \mathcal{U} \mapsto \mathbb{R}^d$ (i.e., attribution methods), we find that these feature selection approaches satisfy certain analogous properties (Appendix C); however, the properties we identify are insufficient to derive an axiomatically unique method.

7.5 Summary

This discussion has shown that every removal-based explanation can be understood using ideas from cooperative game theory. Table 3 displays our findings regarding each method's game-theoretic interpretation and lists the relevant aspects of the literature for each summarization technique. Under our framework, removal-based explanations are implicitly based on an underlying cooperative game (Section 5), and these connections show that the model explanation field has in many cases either reinvented or borrowed ideas that were previously well-understood in cooperative game theory. We speculate that ongoing model explanation research may benefit from borrowing more ideas from cooperative game theory.

These connections are also important because they help use reason about the advantages of different summarization techniques via the properties that each method satisfies. Among the various techniques, we argue that the Shapley value provides the most complete explanation because it satisfies many desirable properties and gives a granular view of each player's contributions (Section 7.2). Unlike the other methods, it divides the grand coalition's value (due to the efficiency property) while capturing the nuances of each player's contributions.

In contrast, the other methods have potential shortcomings, although such issues depend on the use-case. Measuring a single marginal contribution, either by removing or including

Table 3: Each method's summarization technique can be understood in terms of concepts from cooperative game theory.

Summarization	Methods	Related To
Shapley value	Shapley Net Effects, IME, QII, SHAP, TreeSHAP, KernelSHAP, LossSHAP, Shapley Effects, SAGE, SPVIM	Shapley value, probabilistic values, modeling cooperative games
Mean value when included	RISE	Banzhaf value, probabilistic values, modeling cooperative games
Remove/include individual players	Occlusion, PredDiff, CXPlain, permutation tests, univariate predictors, feature ablation (LOCO)	Probabilistic values, modeling cooperative games
Fit additive model	LIME	Shapley value, Banzhaf value, modeling cooperative games
High/low value coalitions	MP, EP, MIR, MM, L2X, INVASE, REAL-X, FIDO-CA	Maximum/minimum excess

individual players, ignores player interactions; for example, removing individual players may lead to near-zero attributions for groups of correlated features, even if they are collectively important. The Banzhaf value provides a more nuanced view of each player's contributions, but it cannot be viewed as a division of the grand coalition's value because it violates the efficiency axiom (Dubey and Shapley, 1979). Finally, feature selection explanations provide only a coarse summary of each player's value contribution, and their results depend on user-specified hyperparameters; moreover, these methods are liable to select only one member out of a group of correlated features, which may be misleading to users.

8. Information-Theoretic Explanations

We now examine how removal-based explanations are connected to information theory. We begin by describing how features can be removed using knowledge of the underlying data distribution, and we then show that many feature removal approaches approximate marginalizing out features using their conditional distribution. Finally, we prove that this approach gives every removal-based explanation an information-theoretic interpretation.

8.1 Removing features consistently

There are many ways to remove features from a model (Section 4.2), so we consider how to do so while accounting for the underlying data distribution. Recall that removal-based explanations evaluate models while withholding groups of features using a subset function

$F \in \mathfrak{F}$, which is typically a subset extension of an existing model $f \in \mathcal{F}$ (Definition 3). We begin by introducing a specific perspective on how to interpret a subset function's predictions, which are represented by $F(x_S)$.

Supervised ML models typically approximate the response variable's conditional distribution given the input. This is clear for classification models that estimate the conditional probability $f(x) \approx p(Y \mid X = x)$, but it is also true for regression models that estimate the conditional expectation $f(x) \approx \mathbb{E}[Y \mid X = x]$. (These approximations are implicit in conventional loss functions such as cross entropy and MSE.) We propose that a subset function $F \in \mathfrak{F}$ can be viewed equivalently as a conditional probability/expectation estimate given *subsets* of features.

To illustrate this idea in the classification case, we denote the model's estimate as $q(Y \mid X = x) \equiv f(x)$, where the model's output is a discrete probability distribution. Although it is not necessarily equal to the true conditional distribution $p(Y \mid X = x)$, the estimate $q(Y \mid X = x)$ represents a valid probability distribution for all $x \in \mathcal{X}$. Similarly, we denote the subset function's estimate as $q(Y \mid X_S = x_S) \equiv F(x_S)$. A subset function F represents a set of conditional distributions $\big\{q(Y \mid X_S) : S \subseteq D\big\}$, and this interpretation is important because we must consider whether these distributions are probabilistically valid. In particular, we must verify that standard probability laws (non-negativity, unitarity, countable additivity, Bayes rule) are not violated.

As we show below, this cannot be guaranteed. The laws of probability impose a relationship between $q(Y \mid X = x)$ and $q(Y \mid X_S = x_S)$ for any $x \in \mathcal{X}$ and $S \subset D$: they are linked by the underlying distribution on $\mathcal{X}$, or, more specifically, by the conditional distribution $X_{\bar{S}} \mid X_S = x_S$. In fact, any distribution $q(X)$ implies a unique definition for $q(Y \mid X_S)$ based on $q(Y \mid X)$ due to Bayes rule and the countable additivity property (described below). The flexibility of subset functions regarding how to remove features is therefore problematic, because certain removal approaches do not yield a valid set of conditional distributions.

Constraining the feature removal approach to be probabilistically valid can ensure that the model's subset extension F is faithful both to the original model f and an underlying data distribution. Building on this perspective, we define the notion of *consistency* between a subset function and a data distribution as follows.

Definition 5 *A subset function $F \in \mathfrak{F}$ that estimates a random variable Y's conditional distribution is* **consistent** *with a data distribution $q(X)$ if its estimates satisfy the following properties:*

1. *(Countable additivity) The probability of the union of a countable number of disjoint events is the sum of their probabilities. Given events $A_1, A_2, \ldots$ such that $A_i \cap A_j = \varnothing$ for $i \neq j$, we have*

$$P\Big(\bigcup_{i=1}^{\infty} A_i\Big) = \sum_{i=1}^{\infty} P(A_i).$$

2. *(Bayes rule) The conditional probability $P(A \mid B)$ for events A and B is defined as*

$$P(A \mid B) = \frac{P(A, B)}{P(B)}.$$

This definition of consistency describes a class of subset functions that obey fundamental probability axioms (Laplace, 1781; Kolmogorov, 1950). Restricting a subset function to be consistent does not make its predictions correct, but it makes them compatible with a particular data distribution $q(X)$. Allowing for a distribution $q(X)$ that differs from the true distribution $p(X)$ reveals that certain approaches implicitly assume modified data distributions.

Based on Definition 5, the next two results show that there is a unique subset extension $F \in \mathfrak{F}$ for a model $f \in \mathcal{F}$ that is consistent with a given data distribution $q(X)$ (proofs in Appendix D). The first result relates to subset extensions of classification models that estimate conditional probabilities.

Proposition 6 *For a classification model $f \in \mathcal{F}$ that estimates a discrete Y's conditional probability, there is a unique subset extension $F \in \mathfrak{F}$ that is consistent with $q(X)$,*

$$F(x_S) = \mathbb{E}_{q(X_{\bar{S}} | X_S = x_S)}\big[f(x_S, X_{\bar{S}})\big],$$

where $q(X_{\bar{S}} \mid X_S = x_S)$ is the conditional distribution induced by $q(X)$, i.e., the distribution

$$q(x_{\bar{S}} \mid X_S = x_S) = \frac{q(x_{\bar{S}}, x_S)}{\int_{X_{\bar{S}}} q(X_{\bar{S}}, x_S)}.$$

The next result arrives at a similar conclusion, but it is specific to subset extensions of regression models that estimate the response variable's conditional expectation.

Proposition 7 *For a regression model $f \in \mathcal{F}$ that estimates a real-valued Y's conditional expectation, there is a unique subset extension $F \in \mathfrak{F}$ that is consistent with $q(X)$,*

$$F(x_S) = \mathbb{E}_{q(X_{\bar{S}} | X_S = x_S)}\big[f(x_S, X_{\bar{S}})\big],$$

where $q(X_{\bar{S}} \mid X_S = x_S)$ is the conditional distribution induced by $q(X)$.

These results differ in their focus on classification and regression models, but the conclusion in both cases is the same: the only subset extension of a model f that is consistent with $q(X)$ is one that averages the full model output $f(X)$ over the distribution of values $X_{\bar{S}}$ given by $q(X_{\bar{S}} \mid X_S)$.

When defining a model's subset extension, the natural choice is to make it consistent with the true data distribution $p(X)$. This yields precisely the approach presented by SHAP (the conditional version), SAGE, and several other methods, which is to marginalize out the removed features using their conditional distribution $p(X_{\bar{S}} \mid X_S = x_S)$ (Strobl et al., 2008; Zintgraf et al., 2017; Lundberg and Lee, 2017; Aas et al., 2019; Covert et al., 2020; Frye et al., 2020).

Besides this approach, only a few other approaches are consistent with any distribution. The QII approach (Eq. 6) is consistent with a distribution that is the product of marginals,

$q(X) = \prod_{i=1}^{d} p(X_i)$. The more recent IME approach (Eq. 7) is consistent with a distribution that is the product of uniform distributions, $q(X) = \prod_{i=1}^{d} u_i(X_i)$. And finally, any approach that sets features to default values $r \in \mathcal{X}$ (Eqs. 1-2) is consistent with a distribution that puts all of its mass on those values—which we refer to as a *constant distribution*. These approaches all achieve consistency because they are based on a simplifying assumption of feature independence.

8.2 Conditional distribution approximations

While few removal-based explanations explicitly suggest marginalizing out features using their conditional distribution, several methods can be understood as approximations of this approach. These represent practical alternatives to the exact conditional distribution, which is unavailable in practice and often difficult to estimate.

The core challenge in using the conditional distribution is modeling it accurately. Methods that use the marginal distribution sample rows from the dataset (Breiman, 2001; Lundberg and Lee, 2017), and it is possible to filter for rows that agree with the features to be conditioned on (Sundararajan and Najmi, 2019); however, this technique does not work well for high-dimensional or continuous-valued data. A relaxed alternative to this approach is using cohorts of rows with similar values (Mase et al., 2019).

While properly representing the conditional distribution for every subset of features is challenging, there are several approaches that provide either rough or high-quality approximations. These approaches include:

- **Assume feature independence.** If we assume that the features $X_1, \ldots, X_d$ are independent, then the conditional distribution $p(X_{\bar{S}} \mid X_S)$ is equivalent to the joint marginal distribution $p(X_{\bar{S}})$, and it is even equivalent to the product of marginals $\prod_{i \in \bar{S}} p(X_i)$. The removal approaches used by KernelSHAP and QII can therefore be understood as rough approximations to the conditional distribution that assume feature independence (Datta et al., 2016; Lundberg and Lee, 2017).

- **Assume model linearity (and feature independence).** As Lundberg and Lee (2017) pointed out, replacing features with their mean can be interpreted as an additional assumption of model linearity:

$$\begin{aligned}
\mathbb{E}\big[f(X) \mid X_S = x_S\big] &= \mathbb{E}_{p(X_{\bar{S}} \mid X_S = x_S)}\big[f(x_S, X_{\bar{S}})\big] && \text{(Conditional distribution)} \\
&\approx \mathbb{E}_{p(X_{\bar{S}})}\big[f(x_S, X_{\bar{S}})\big] && \text{(Assume feature independence)} \\
&\approx f\big(x_S, \mathbb{E}[X_{\bar{S}}]\big). && \text{(Assume model linearity)}
\end{aligned}$$

  While this pair of assumptions rarely holds in practice, particularly with the complex models for which explanation methods are designed, it provides some justification for methods that replace features with default values (e.g., LIME, Occlusion, MM, CXPlain, RISE).

- **Parametric assumptions.** Recent work on Shapley values has proposed parametric approximations of the conditional distribution, e.g., multivariate Gaussian and copula-based models (Aas et al., 2019, 2021). While the parametric assumptions may not hold

exactly, these approaches can provide better conditional distribution approximations than feature independence assumptions.

- **Generative model.** FIDO-CA proposes removing features by drawing samples from a conditional generative model (Chang et al., 2018). If the generative model p_G (e.g., a conditional GAN) is trained to optimality, then it produces samples from the true conditional distribution. We can then write

$$p_G(X_{\bar{S}} \mid X_S) \overset{d}{=} p(X_{\bar{S}} \mid X_S), \tag{43}$$

  where $\overset{d}{=}$ denotes equality in distribution. Given a sample $\tilde{x}_{\bar{S}} \sim p_G(X_{\bar{S}} \mid X_S = x_S)$, the prediction $f(x_S, \tilde{x}_{\bar{S}})$ can be understood as a single-sample Monte Carlo approximation of the expectation $\mathbb{E}[f(X) \mid X_S = x_S]$. Agarwal and Nguyen (2019) substituted the generative model approach into several existing methods (Occlusion, MP, LIME) and observed improvements in their performance across numerous metrics. Frye et al. (2020) demonstrated a similar approach using a variational autoencoder-like model (Ivanov et al., 2018), and future work could leverage other conditional generative models (Douglas et al., 2017; Belghazi et al., 2019).

- **Surrogate model.** Several methods require a surrogate model that is trained to match the original model's predictions given subsets of features, where missing features are represented by zeros (or other values that do not appear in the dataset). This circumvents the task of modeling an exponential number of conditional distributions and is equivalent to parameterizing a subset function $F \in \mathfrak{F}$ and training it with the following objective:

$$\min_F \ \mathbb{E}_X \mathbb{E}_S \Big[\ell\big(F(X_S), f(X)\big) \Big]. \tag{44}$$

  In Appendix E.3, we prove that for certain loss functions ℓ, the subset function F that optimizes this objective is equivalent to marginalizing out features using their conditional distribution. Frye et al. (2020) show a similar result for MSE loss, and we also show that a cross entropy loss can be used for classification models. Finally, we find that L2X (Chen et al., 2018a) may not provide a faithful approximation because the subsets S are not distributed independently from the inputs X—an issue recently addressed by REAL-X (Jethani et al., 2021a).

- **Missingness during training.** Rather than training a surrogate with missing features, we can instead learn the original model with missingness introduced during training. This is equivalent to parameterizing a subset function $F \in \mathfrak{F}$ and optimizing the following objective, which resembles the standard training loss:

$$\min_F \ \mathbb{E}_{XY} \mathbb{E}_S \Big[\ell\big(F(X_S), Y\big) \Big]. \tag{45}$$

  In Appendix E.2, we prove that if the model optimizes this objective, then it is equivalent to marginalizing out features from $f(x) \equiv F(x)$ using the conditional distribution.

An important aspect of this objective is that the subsets S must be independently distributed from the data (X, Y). INVASE (Yoon et al., 2018) does not satisfy this property, which may prevent it from providing an accurate approximation.

- **Separate models.** Finally, Shapley Net Effects (Lipovetsky and Conklin, 2001), SPVIM (Williamson and Feng, 2020) and the original IME (Štrumbelj et al., 2009) propose training separate models for each feature subset, or $\{f_S : S \subseteq D\}$. If every model is optimal, then this approach is equivalent to marginalizing out features with their conditional distribution (Appendix E.1). This is due to a relationship that arises between models that optimize the population risk for different sets of features; for example, with cross entropy loss, the optimal model (the Bayes classifier) for X_S is given by $f_S(x_S) = p(Y \mid X_S = x_S)$, which is equivalent to $\mathbb{E}[f_D(X) \mid X_S = x_S]$ because the optimal model given all features is $f_D(x) = p(Y \mid X = x)$.

This discussion shows that although few methods explicitly suggest removing features with the conditional distribution, numerous methods approximate this approach. Training separate models should provide the best approximation because each model is given a relatively simple prediction task, but this approach is unable to scale to high-dimensional datasets. The generative model approach amortizes knowledge of an exponential number of conditional distributions into a single model, which is more scalable and effective for image data (Yu et al., 2018); the supervised surrogate and missingness during training approaches also require learning up to one additional model, and these are trained with far simpler optimization objectives (Eq. 44, 45) than conditional generative models.

We conclude that, under certain assumptions of feature independence or model optimality, several feature removal strategies are consistent with the data distribution $p(X)$. Our definition of consistency provides a new lens for comparing different feature removal strategies, and Table 4 summarizes our findings.

8.3 Connections with information theory

Conventional wisdom suggests that explanation methods quantify the information contained in each feature. However, we find that precise information-theoretic connections can be identified only when held-out features are marginalized out with their conditional distribution. Our analysis expands on prior work by showing that every removal-based explanation has a probabilistic or information-theoretic interpretation when features are removed properly (Owen, 2014; Chen et al., 2018a; Covert et al., 2020).

To aide our presentation, we assume that the model f is optimal, i.e., it is the Bayes classifier $f(x) = p(Y \mid X = x)$ for classification tasks or the conditional expectation $f(x) = \mathbb{E}[Y \mid X = x]$ for regression tasks. This assumption is optimistic, but because models are typically trained to approximate one of these functions, the resulting explanations are approximately based on the information-theoretic quantities derived here.

By assuming model optimality and marginalizing out removed features with their conditional distribution, we can guarantee that the prediction given any subset of features is optimal for those features. Specifically, we have the Bayes classifier $F(x_S) = p(Y \mid X_S = x_S)$ in the classification case and the conditional expectation $F(x_S) = \mathbb{E}[Y \mid X_S = x_S]$ in the

Table 4: Consistency properties of different feature removal strategies.

Removal	Methods	Consistency
Marginalize (conditional)	Cond. permutation tests, PredDiff, SHAP, LossSHAP, SAGE, Shapley Effects	Consistent with $p(X)$
Generative model	FIDO-CA	Consistent with $p(X)$ (assuming model optimality)
Supervised surrogate	L2X, REAL-X, Frye et al. (2020)	
Missingness during training	INVASE	
Separate models	Feature ablation (LOCO), univariate predictors, Shapley Net Effects, SPVIM, IME (2009)	
Marginalize (marginal)	Permutation tests, KernelSHAP	Consistent with $p(X)$ (assuming independence)
Marginalize (marginals product)	QII	
Marginalize (marginals product)	QII	Consistent with $q(X)$ with feature independence
Marginalize (uniform)	IME (2010)	
Zeros	Occlusion, PredDiff RISE, CXPlain	Consistent with constant distributions $q(X)$
Default values	LIME (images), MM	
Tree distribution	TreeSHAP	Not consistent with any $q(X)$
Extend pixel values	MIR	
Blurring	MP, EP	Not valid $F \in \mathfrak{F}$
Marginalize (replacement dist.)	LIME (tabular)	

regression case. Using these subset functions, we can derive probabilistic and information-theoretic interpretations for each explanation method.

These connections focus on the set functions analyzed by each removal-based explanation (Section 5). We present results for classification models that use cross entropy loss here, and we show analogous results for regression models in Appendix F. Under the assumptions described above, the set functions analyzed by each method can be interpreted as follows:

- The set function $u_x(S) = F(x_S)$ represents the response variable's conditional probability for the chosen class y:

$$u_x(S) = p(y \mid X_S = x_S). \tag{46}$$

  This lets us examine each feature's true association with the response variable.

- The set function $v_{xy}(S) = -\ell\big(F(x_S), y\big)$ represents the log probability of the correct class y, which is equivalent to the *pointwise mutual information* $\mathrm{I}(y; x_S)$ (up to a constant value):

$$v_{xy}(S) = \mathrm{I}(y; x_S) + c. \tag{47}$$

  This quantifies how much information x_S contains about the outcome y, or how much less surprising y is given knowledge of x_S (Fano, 1961).

- The set function $v_x(S) = -\mathbb{E}_{p(Y|X=x)}\big[\ell(F(x_S), Y)\big]$ represents the negative Kullback-Leibler (KL) divergence between the label's conditional distribution and its partial conditional distribution (up to a constant value):

$$v_x(S) = c - D_{\mathrm{KL}}\big(p(Y \mid X = x) \;\big|\big|\; p(Y \mid X_S = x_S)\big). \tag{48}$$

  As mentioned by Yoon et al. (2018), this provides an information-theoretic measure of the deviation between the response variable's true distribution and its distribution when conditioned on a subset of features (Cover and Thomas, 2012).

- The set function $v(S) = -\mathbb{E}_{XY}\big[\ell\big(F(X_S), Y\big)\big]$ represents the *mutual information* with the response variable (up to a constant value):

$$v(S) = \mathrm{I}(Y; X_S) + c. \tag{49}$$

  As discussed by Covert et al. (2020), this quantifies the amount of information, or the predictive power, that the features X_S communicate about the response variable Y (Cover and Thomas, 2012).

- The set function $w_x(S) = -\ell\big(F(x_S), f(x)\big)$ represents the KL divergence between the full model output and its output given a subset of features. Specifically, if we define Z to be a categorical random variable $Z \sim \mathrm{Cat}\big(f(X)\big)$, then we have:

$$w_x(S) = c - D_{\mathrm{KL}}\big(p(Z \mid X = x) \;\big|\big|\; p(Z \mid X_S = x_S)\big). \tag{50}$$

  This result does not require model optimality, but under the assumption that f is the Bayes classifier, this quantity is equivalent to the KL divergence between the label's conditional and partial conditional distribution (up to a constant value):

$$w_x(S) = c - D_{\mathrm{KL}}\big(p(Y \mid X = x) \;\big|\big|\; p(Y \mid X_S = x_S)\big). \tag{51}$$

  We can therefore see that under these assumptions, L2X, INVASE and REAL-X are based on the same set function and have a similar information-theoretic interpretation (Chen et al., 2018a; Yoon et al., 2018; Jethani et al., 2021a).

Table 5: Each method's underlying set function has an information-theoretic interpretation when features are removed appropriately.

Model Behavior	Set Function	Methods	Related To
Prediction	u_x	Occlusion, MIR, MM, IME, QII, LIME, MP, EP, FIDO-CA, RISE, SHAP, KernelSHAP, TreeSHAP	Conditional probability, conditional expectation
Prediction loss	v_{xy}	LossSHAP, CXPlain	Pointwise mutual information
Prediction mean loss	v_x	INVASE	KL divergence with conditional distribution
Dataset loss	v	Permutation tests, univariate predictors, feature ablation (LOCO), Shapley Net Effects, SAGE, SPVIM	Mutual information (with label)
Prediction loss (output)	w_x	L2X, REAL-X	KL divergence with full model output
Dataset loss (output)	w	Shapley Effects	Mutual information (with output)

- The set function $w(S) = -\mathbb{E}_{XY}\big[\ell\big(F(X_S), f(X)\big)\big]$ represents the information that X_S communicates about the model output $f(X)$. Specifically, if we let $Z \sim \text{Cat}\big(f(X)\big)$, then we have:

$$w(S) = \text{I}\big(Z; X_S\big) + c. \tag{52}$$

  This is the classification version of the conditional variance decomposition provided by Shapley Effects (Owen, 2014). This result does not require model optimality, but if f is the Bayes classifier, then this is equivalent to the mutual information with the response variable (up to a constant value):

$$w(S) = \text{I}(Y; X_S) + c. \tag{53}$$

As noted, two assumptions are required to derive these results. The first is that features are marginalized out using the conditional distribution; although many methods use different removal approaches, they can be modified to use this approach or an approximation (Section 8.2). The second is that models are optimal; this assumption rarely holds in

practice, but since conventional loss functions train models to approximate either the Bayes classifier or the conditional expectation, we can view these information-theoretic quantities as the values that each set function approximates.

We conclude that when features are removed appropriately, explanation methods quantify the information communicated by each feature (see summary in Table 5). No single set function provides the "right" approach to model explanation; rather, these information-theoretic quantities span a range of perspectives that could be useful for understanding a complex ML model.

Removal-based explanations that are consistent with the observed data distribution can provide well-grounded insight into intrinsic statistical relationships in the data, and this is useful for finding hidden model influences (e.g., detecting bias from sensitive attributes) or when using ML as a tool to discover real-world relationships. However, this approach has the potentially undesirable property that features may appear important even if they are not used by the model in a functional sense (Merrick and Taly, 2019; Chen et al., 2020). When users are more interested in the model's mechanism for calculating predictions, other removal approaches may be preferable, such as interventional approaches motivated by a causal analysis of the model (Janzing et al., 2019; Heskes et al., 2020).

9. A Cognitive Perspective on Removal-Based Explanations

While the previous sections provide a mathematical perspective on removal-based explanations, we now consider this class of methods through a different lens: that of the social sciences. Analyzing this broad class of methods provides an opportunity to discuss how they all relate to cognitive psychology due to their shared reliance on feature removal.

Model explanation tools are not typically designed based on research from the social sciences (Miller et al., 2017), but, as we show, removal-based explanations have clear parallels with cognitive theories about how people understand causality. We first discuss our framework's foundation in counterfactual reasoning and then describe a trade-off between simple explanations and those that convey richer information about models.

9.1 Subtractive counterfactual reasoning

Explaining a model's predictions is fundamentally a causality question: *what makes the model behave this way?* Each input feature is a potential cause, multiple features may be causal, and explanations should quantify each feature's degree of influence on the model. We emphasize the distinction between this model-focused causality and causality between the input and response (e.g., whether a feature causes the outcome) because real-world causality is difficult to discern from observational data (Pearl, 2009).

In philosophy and psychology, *counterfactual reasoning* is a standard tool for understanding causality. A counterfactual example changes certain facts of a situation (e.g., the route a person drove to get home) to potentially achieve a different outcome (e.g., getting home safely), and counterfactuals shed light on whether each aspect of a situation caused the actual outcome (e.g., a fatal car crash). In an influential philosophical account of causality, John Stuart Mill presented five methods of induction that use counterfactual reasoning to explain cause-effect relationships (Mill, 1884; Mackie, 1974). In the psychology literature,

counterfactual thinking is the basis of multiple theories about how people explain the causes of events (Kahneman and Tversky, 1982; Hilton, 1990).

Removal-based explanations perform a specific type of counterfactual reasoning. In psychology, the process of removing an event to understand its influence on an outcome is called a *subtractive counterfactual* (Epstude and Roese, 2008), and this is precisely how removal-based explanations work. In philosophy, the same principle is called the *method of difference*, and it is one of Mill's five methods for inferring cause-effect relationships (Mill, 1884). This type of logic is also found in cognitive theories about how people understand and discuss causality (Kahneman and Tversky, 1982; Jaspars et al., 1983; Hilton, 1990).

The principle of removing something to examine its influence is pervasive in social sciences, not only as a philosophical approach but as part of descriptive psychological theories; this explains the remarkable prevalence of the feature removal principle in model explanation, even among computational researchers who were not explicitly inspired by psychology research. Perhaps surprisingly, the reliance on subtractive counterfactual reasoning (or equivalently, the method of difference) has been overlooked thus far, even by work that examined the psychological basis for SHAP (Merrick and Taly, 2019; Kumar et al., 2020).

Some prior work applies a different form of counterfactual reasoning to model explanation (Verma et al., 2020). For example, one influential approach suggests showing users counterfactuals that adjust a small number of features to change the model output (Wachter et al., 2017). This approach undoubtedly provides information about how a model works, but many such counterfactuals are required to fully illustrate each feature's influence. Removal-based explanations can provide more insight by concisely summarizing the results of many subtractive counterfactuals, e.g., via Shapley values.

9.2 Norm theory and the downhill rule

Subtractive counterfactuals are an intuitive way to understand each feature's influence, but their implementation is not straightforward. Removal-based explanations aim to remove the information that a feature communicates, or subtract the fact that it was observed, but it is not obvious how to do this: given an input $x \in \mathcal{X}$ and subset $S \subseteq D$, it is unclear how to retain x_S while removing $x_{\bar{S}}$. We consult two psychological theories to contextualize the approaches that have been considered by different methods.

Many removal-based explanations remove features by averaging the model output over a distribution of possible values for those features; this has a clear correspondence with the cognitive model described by *Norm theory* (Kahneman and Miller, 1986). According to this theory, people assess normality by gathering summary statistics from a set of recalled and simulated representations of a phenomenon (e.g., loan application outcomes for individuals with a set of characteristics). In these representations, certain features are fixed (or *immutable*) while others are allowed to vary (*mutable*); in our case these correspond to the retained and removed features, respectively.

Taking inspiration from Norm theory, we may equate a model's behavior when certain features are blocked from exerting influence (i.e., the removed features) with a "normal" outcome for the remaining features. With this perspective, we can see that Norm theory provides a cognitive justification for averaging the model output over a distribution of values

for the removed features. Merrick and Taly (2019) make a similar observation to justify how SHAP removes features.

The choice of distribution for averaging outcomes is important, but Norm theory does not prescribe a specific approach. Rather, Norm theory is a descriptive cognitive model that recognizes that individuals may have different perspectives of normality based on their experiences (Kahneman and Miller, 1986). Future model explanation research may consider how to tailor explanations to each user, as suggested by Miller (2019), but we also require systematic methods that do not solicit user input. We therefore consider whether any approach used by existing methods is justifiable from a cognitive perspective.

For guidance on the choice of distribution, we look to research on human tendencies when assigning blame. In their study of *mental undoing*, Kahneman and Tversky (1982) examined people's biases when proposing counterfactuals that change an undesirable event's outcome. Their clearest finding was the *downhill rule*, which states that people are more likely to propose changes that remove a surprising aspect of a story or otherwise increase the story's internal coherence. In other words, people are more likely to assign blame to an aspect of a situation if it has a more likely alternative that would change the outcome.

One feature removal strategy is reminiscent of the downhill rule because it considers alternative values in proportion to their plausibility: when marginalizing out removed features using their conditional distribution, alternative values and their corresponding outcomes are averaged in proportion to the coherence of the full feature set, which is represented by the data distribution $p(X_{\bar{S}} \mid X_S) \propto p(X)$. This is consistent with the downhill rule because if certain high-likelihood values change the outcome, their influence on the model will be apparent when integrating $f(X)$ over the distribution $p(X \mid X_S = x_S)$.

In summary, Norm theory provides a cognitive analogue for removing features by averaging over a distribution of alternative values, and the downhill rule suggests that the plausibility or likelihood of alternative values should be taken into account. These theories provide cognitive justification for certain approaches to removing features, but our review of relevant psychology research is far from comprehensive. However, interestingly, our findings lend support to the same approach that yields connections with information theory (Section 8), which is marginalizing out features using their conditional distribution.

9.3 Simplicity versus completeness

Building on our discussion of the human psychology aspect of removal-based explanations, we now discuss a trade-off between the amount of information conveyed by an explanation and the user's likelihood of drawing the correct conclusions. We describe this trade-off and then show that this class of methods provides the flexibility to balance these competing objectives.

Consider two explanation strategies with different levels of complexity. A counterfactual example that perturbs several features to change the model's prediction is easy to understand, but it does not convey detailed information about a model's dependence on each feature (Wachter et al., 2017). By contrast, SHAP's feature attributions provide a complex summary of each feature's influence by quantifying the impact of removing different groups of features. Perhaps due to their greater complexity, a recent user study showed that users were less likely to understand SHAP visualizations, experiencing a higher cognitive load

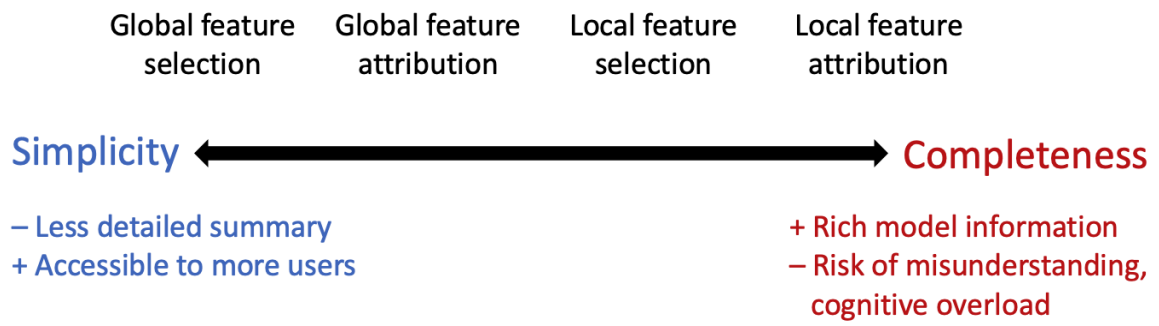

Figure 4: Each model explanation strategy represents a trade-off between an explanation's simplicity and its completeness.

than users who viewed simpler explanations (Kaur et al., 2020). Similarly, a different study found that longer explanations required more time and effort to understand, in some cases impeding a user's ability to draw appropriate conclusions (Lage et al., 2019).

We view this as a trade-off between simplicity and completeness, because explanations are typically more complex when they provide more information about a model. Providing a more complete characterization may be preferable for helping users build a mental picture of how a model works, but complicated explanations also risk overloading users, potentially leading to a false or limited sense of model comprehension.

Recognizing this trade-off and its implications, we consider simplicity as a design goal and find that removal-based explanations have the flexibility to adjust the balance between these goals. To reduce cognitive burden, one might focus on global rather than local explanations, e.g., by providing visualizations that display many local explanations (Lundberg et al., 2020), or by using global methods that summarize a model's behavior across the entire dataset (Owen, 2014; Covert et al., 2020). Alternatively, one may generate explanations that operate on feature groups rather than individual features, e.g., sets of correlated features or nearby pixels (Zeiler and Fergus, 2014; Ribeiro et al., 2016).

Certain explanation formats convey richer information than others (Figure 4). For example, feature attributions provide a granular view of a model by considering every feature as a cause and quantifying each feature's influence, and local explanations are more granular than global ones. Richer information may not always be desirable, because psychology research shows that people report higher satisfaction with explanations that cite fewer causes (Thagard, 1989; Read and Marcus-Newhall, 1993; Lombrozo, 2007; Miller, 2019). Users may therefore derive more insight from explanations that highlight fewer causes, such as the sparse feature attributions provided by LIME (Ribeiro et al., 2016).

Feature selection explanations offer the potential to go even further in the direction of simplicity. These explanations directly penalize the number of selected features (Section 6.1), guaranteeing that fewer features are labeled as important; and furthermore, they omit information about the granularity of each feature's influence. For a non-technical user, it may be simpler to understand that a model's prediction was dominated by a small number of highly informative features, whereas it may require a more sophisticated user to

interpret real-valued attributions for individual features, e.g., as coefficients in an additive decomposition of a model's behavior (Section 7.3).

Put simply, an explanation that paints an incomplete picture of a model may prove more useful if the end-users are able to understand it properly. Designers of explanation methods should be wary of overestimating people's abilities to store complex mental models (Norman, 1983), and the ML community can be mindful of this by tailoring explanations to users' degrees of sophistication.

10. Experiments

We have thus far analyzed removal-based explanations from a primarily theoretical standpoint, so we now conduct experiments to provide a complementary empirical perspective. Our experiments aim to accomplish three goals:

1. Implement and compare many new methods by filling out the space of removal-based explanations (Figure 2).

2. Demonstrate the advantages of removing features by marginalizing them out using their conditional distribution—an approach that we showed yields information-theoretic explanations (Section 8).

3. Verify the existence of relationships between various explanation methods. Specifically, explanations may be similar if they use (i) summary techniques that are probabilistic values of the same cooperative game (Section 7), or (ii) feature removal strategies that are approximately equivalent (Section 8.2).

To cover a wide range of methods, we consider many combinations of removal strategies (Section 4), as well as model behaviors (Section 5) and summary techniques (Section 6). Our implementation is available online,[9] and we tested 80 total methods (68 of which are new) that span our framework as follows:

- For *feature removal*, we considered replacing features with default values, and marginalizing out features using either (i) uniform distributions, (ii) the product of marginals, or (iii) the joint marginal distribution. Next, to approximate marginalizing out features using their conditional distribution, we trained surrogate models to match the original model's predictions with held-out features (Appendix E.3). Finally, for one dataset we also trained separate models with all feature subsets.

- Our experiments analyze three *model behaviors* using three different datasets. We explained individual classification probabilities for the census income dataset (Lichman et al., 2013), the model's loss on individual predictions for MNIST (LeCun et al., 2010), and the dataset loss for a breast cancer classification task (Berger et al., 2018).

- For *summary techniques*, we considered removing or including individual features, the mean when included strategy, and Banzhaf and Shapley values. For techniques that

9. `https://github.com/iancovert/removal-explanations`

involve an exponential number of feature subsets, we used sampling-based approximations similar to those previously used for Shapley values (Štrumbelj and Kononenko, 2010), and we detected convergence based on the width of confidence intervals.[10]

Implementing and comparing all combinations of these choices helps us better understand the relationships between methods and identify the most promising approaches. For more details about the models, datasets and hyperparameters, see Appendix G.

10.1 Census income

The census income dataset provides basic demographic information about individuals, and the task is to predict whether a person's annual income exceeds $50k. We trained a LightGBM model (Ke et al., 2017) and then generated explanations using all the combinations of removal and summary strategies described above, including training separate models for all feature subsets (as there are only 12 features). When following the default values removal strategy, we used the mean for continuous features and the mode for discrete ones.

These combinations of choices resulted in 30 distinct explanation methods, several of which are nearly equivalent to existing approaches (SHAP, QII, IME, Occlusion, RISE, PredDiff),[11] but most of which are new. Figure 5 shows a grid of explanations generated for a single person whose income did not exceed $50k. We offer two qualitative remarks about the relationships between these explanations:

1. The explanations are sometimes similar despite using different feature removal strategies (see the bottom four rows of Figure 5). This is most likely because these removal approaches all approximate the conditional distribution (product, marginal, surrogate, separate models). The first two rows (default values, uniform) deviate from the others because they offer low-quality conditional distribution approximations.

2. The explanations are sometimes similar despite using different summary techniques (see the columns for include individual, Banzhaf and Shapley values). The similarity between probabilistic values suggests that each feature's marginal contributions are largely similar, at least outside of a saturating regime; the remove individual technique deviates from the others, possibly due to saturation effects when most features are included. In contrast, the mean when included technique differs significantly from the others because it is not a probabilistic value (in the game-theoretic sense).

To verify whether these relationships hold in general, we scaled up our comparison by generating explanations for 256 instances and quantifying the similarity between methods. We calculated the mean Euclidean distance between explanations generated by each method (averaged across the instances), and, rather than displaying the pair-wise distances for all 30 methods, we only considered comparisons between methods that differ just in their summary techniques or just in their removal strategies.

Figure 6 shows the results, with the distances between explanations grouped by either summary technique (left) or by feature removal strategy (right). Regarding the summary

10. We follow the approach from Covert et al. (2020), which identifies convergence via the ratio of confidence interval width to the range in values.
11. There are some implementation differences, e.g., our conditional distribution approximation for PredDiff.

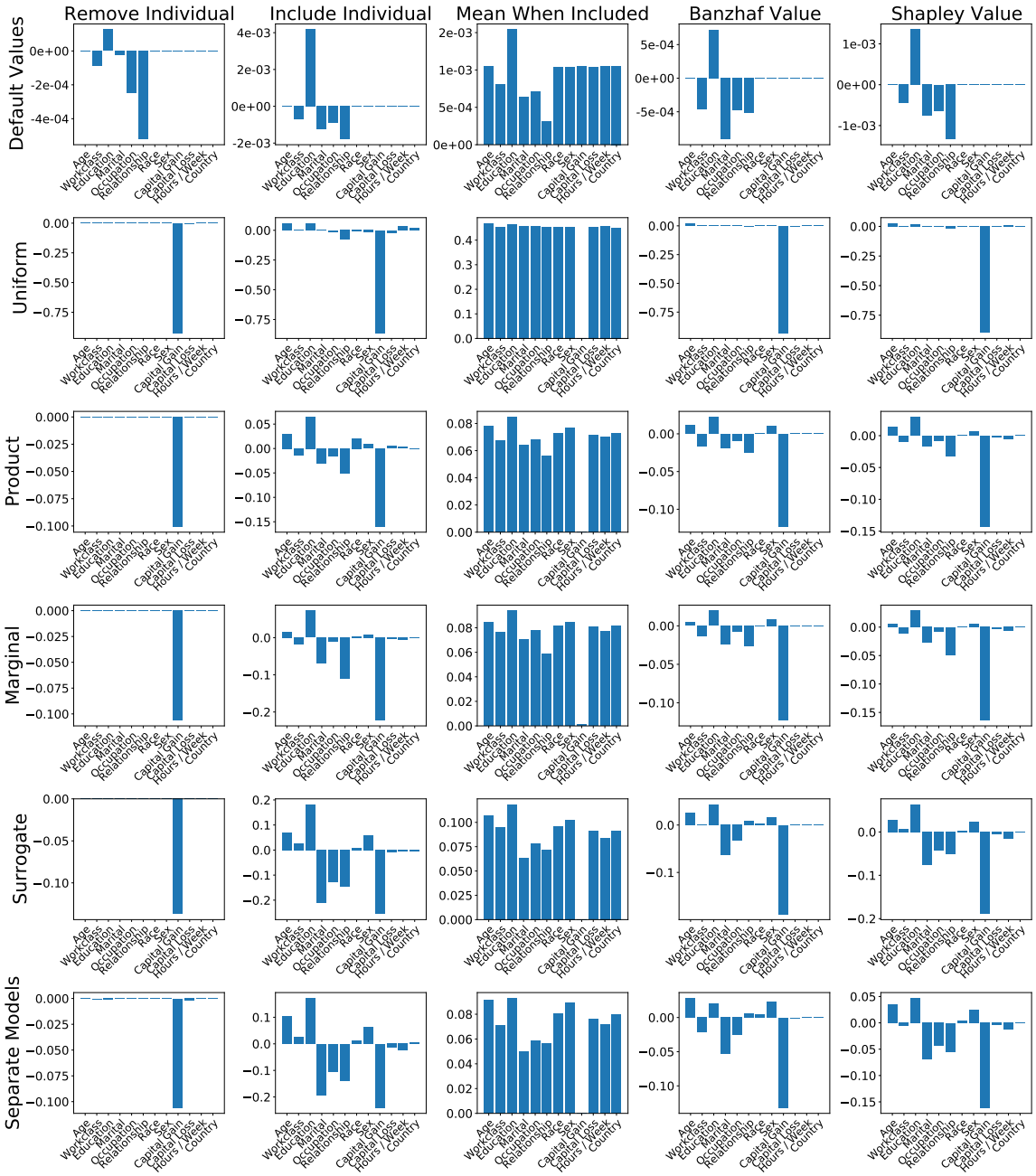

Figure 5: Explanations for a single example in the census income dataset. Each bar chart represents attribution values for a different explanation method. The vertical axis represents feature removal strategies and the horizontal axis represents summary techniques.

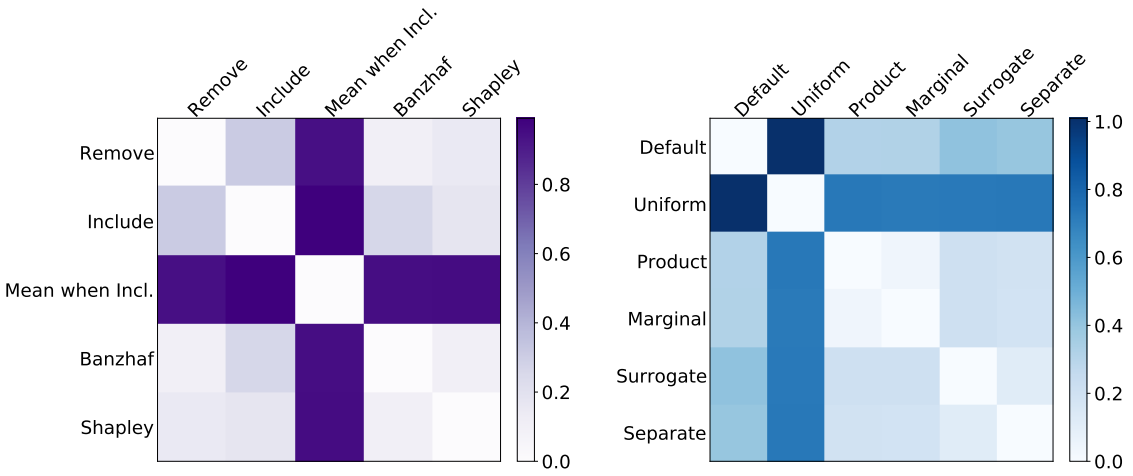

Figure 6: Census income explanation differences. The colored entries in each plot represent the mean Euclidean distance between explanations that differ in only their summary technique (left) or feature removal strategy (right), so lighter entries indicate greater similarity.

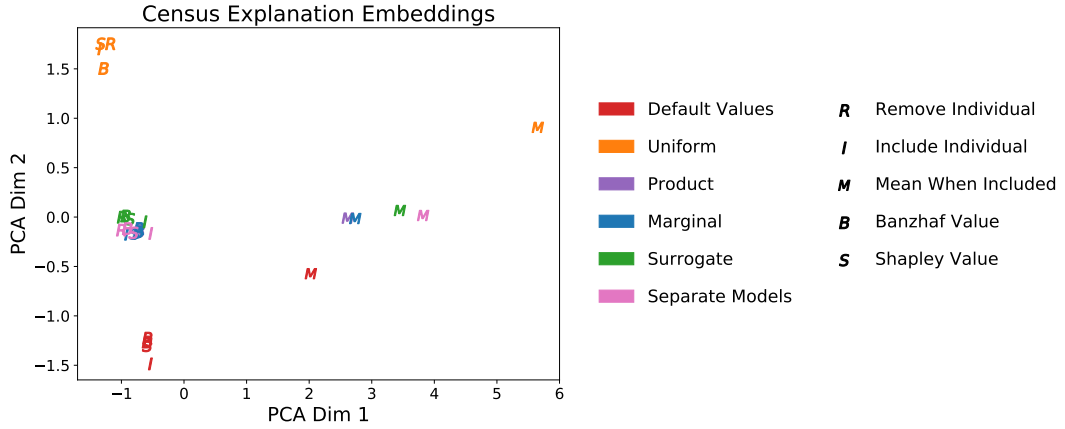

Figure 7: Census income explanation embeddings. PCA was used to generate two-dimensional embeddings for each method, allowing us to observe which methods tend to produce similar results.

techniques, the clearest finding is that the mean when included approach produces very different explanations than all the other techniques. Among the remaining ones, Shapley values are relatively close to Banzhaf values, and they are similarly close to explanations that remove or include individual features; this matches the Shapley value's formulation, because like the Banzhaf value, it is a weighted average of all marginal contributions.

Regarding the different feature removal strategies (Figure 6 right), we observe that the default values and uniform marginalization strategies produce very different explanations than the other approaches. The remaining approaches are relatively similar; for example,

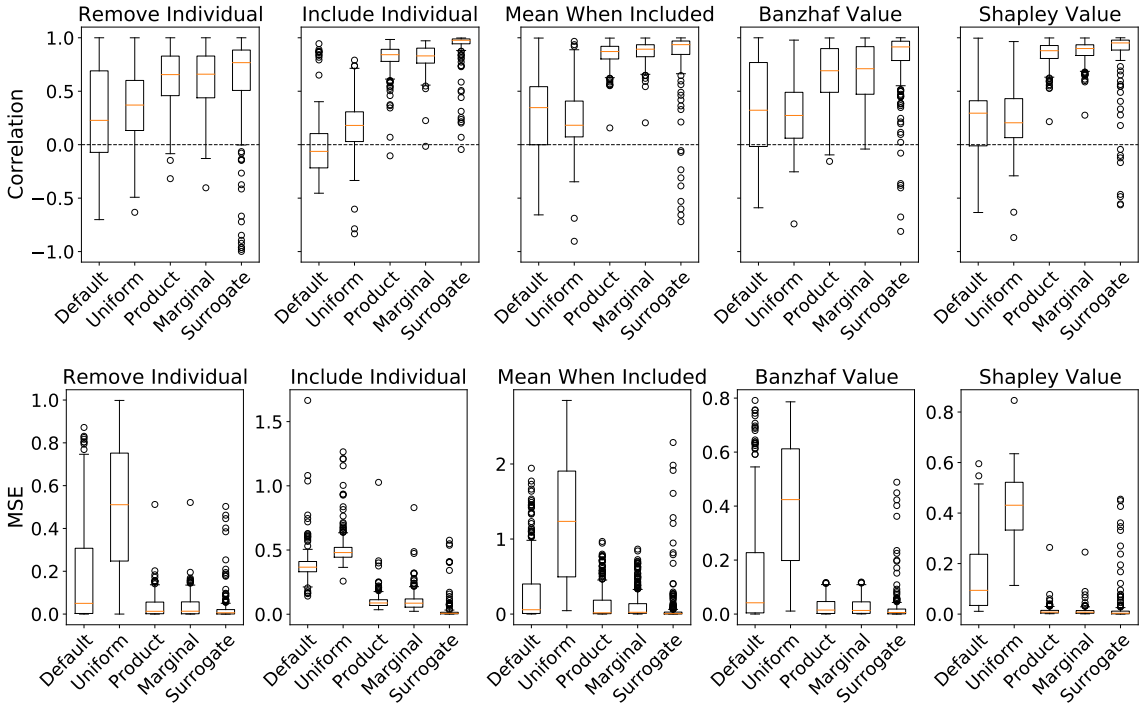

Figure 8: Boxplots quantifying the similarity of census income explanations to comparable explanations generated using separate models. The similarity metrics are correlation (top, higher is better) and MSE (bottom, lower is better).

the product of marginals and joint marginal produce nearly identical explanations, which suggests that the feature dependencies either are not strong in this dataset or are not captured by our model. We also observe that the surrogate approach is closest to training separate models, as we would expect, but that they are not identical.

Next, we visualized the different explanations using low-dimensional embeddings (Figure 7). We generated embeddings by applying principal components analysis (PCA, Jolliffe, 1986) to vectors containing the concatenated explanations for all 256 explanations (i.e., 30 vectors of size 3,072). The results further support our observation that the mean when included technique differs most from the others. The explanations whose feature removal strategy approximates the conditional distribution are strongly clustered, with the product of marginals and joint marginal approaches overlapping almost entirely.

Finally, since many feature removal strategies can be understood as approximating the conditional distribution approach (Section 8.2), we attempted to quantify the approximation quality. We cannot measure this exactly because we lack access to the conditional distributions, so we considered the explanations generated with separate models to be our ground truth, because this is our most reliable proxy. To measure each method's faithfulness to the underlying data distribution, we grouped explanations by their summary technique (e.g., Banzhaf value, Shapley value) and quantified their similarity to explanations generated with separate models (Figure 8).

We calculated two similarity metrics, MSE and correlation, and both show that the surrogate approach is closest to training separate models. This reflects the fact that both approaches provide flexible approximations to marginalizing out features using their conditional distribution (Section 8.2). The default and uniform approaches tend to produce very different explanations. The product and marginal approaches are sometimes competitive with the surrogate approach, but they are noticeably less similar in most cases. We remark, however, that the surrogate approach has more outliers; this suggests that although it is closest to using separate models on average, the surrogate approach is prone to occasionally making large errors.

In sum, these results show that the surrogate approach provides a reliable proxy for the conditional distribution, producing explanations that are faithful to the underlying data distribution. Unlike the separate models approach, the surrogate approach scales to high-dimensional datasets because it requires training just one additional model. And in comparison to other approaches that marginalize out features (uniform, product, marginal), the surrogate approach produces relatively fast explanations because it does not require a sampling-based approximation (i.e., considering multiple values for held-out features) when evaluating the prediction for each feature subset.

10.2 MNIST

For the MNIST digit recognition dataset, we trained a 14-layer CNN and generated explanations for the model's loss rather than its classification probabilities. We used zeros as default values, and, as in the previous experiment, we trained a surrogate model to approximate the conditional distribution.

Combining different removal and summary strategies resulted in 25 explanation methods, only two of which corresponded to existing approaches (LossSHAP, and CXPlain without the amortized explainer model). Using these methods, we first generated a grid of explanations for a single example in the dataset (Figure 9). More so than in the previous experiment, we now observe significant differences between explanations generated by each method. We make the following qualitative observations about these explanations:

1. The empty region at the top of the digit should be relevant because it clearly distinguishes fours from nines. This is successfully highlighted by most, but not all methods.

2. Two removal strategies (uniform, product) frequently produce low-quality, noisy explanations. In contrast, the default value explanations are noiseless, but zero pixels always receive zero attribution because removing them does not impact the model; this is not ideal, because zero pixels can be highly informative.

3. The mean when included technique is difficult to visualize because its attributions are not marginal contributions in the game-theoretic sense, where positive (negative) values improve (hurt) the model's loss.

4. When using the surrogate approach, the Shapley value explanation (Figure 9 bottom right) is most similar to the one that includes individual features. In contrast,

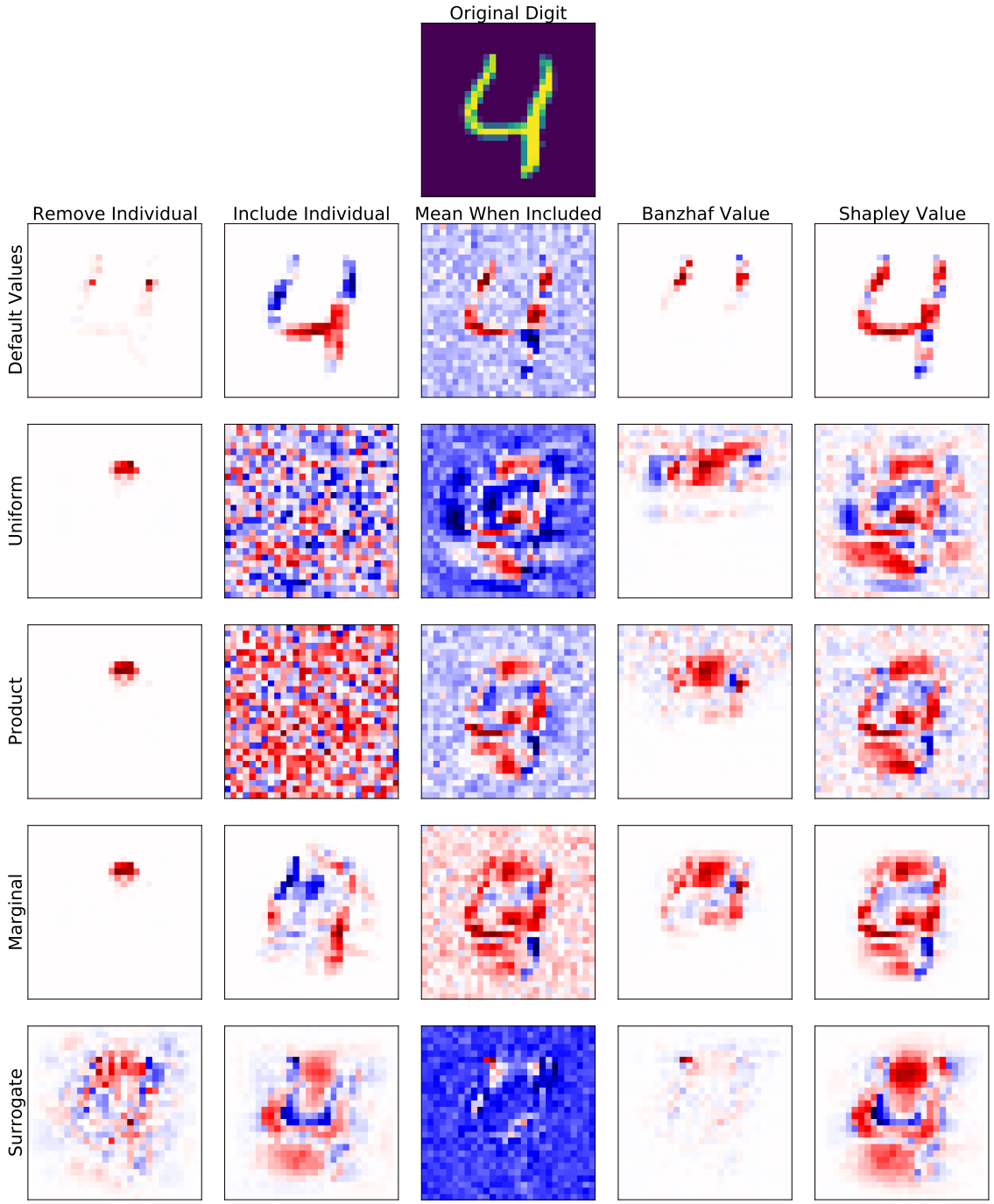

Figure 9: Loss explanations for a single MNIST digit. Each heatmap represents attribution values for a different explanation method, with red (blue) pixels improving (hurting) the loss, except for explanations that use the mean when included technique. The vertical axis represents feature removal strategies and the horizontal axis represents summary techniques.

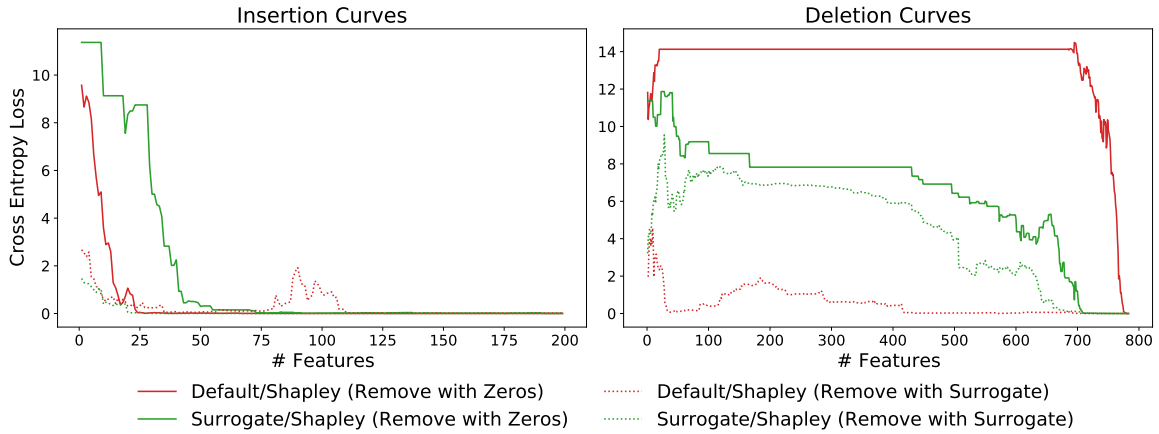

Figure 10: Insertion and deletion curves for a single MNIST digit. Curves are displayed for two explanations (Default/Shapley and Surrogate/Shapley) and two feature removal approaches (Zeros, Surrogate). The insertion curves are only shown up to 200 features due to near-zero loss.

removing individual features has a negligible impact on the model, leading to significant noise artifacts (Figure 9 bottom left). This suggests that removing individual features, which is far more common than including individual features (see Table 1), can be incompatible with close approximations of the conditional distribution, likely due to strong feature correlations.

Overall, the most visually appealing explanation is the one produced by the surrogate and Shapley value combination (Figure 9 bottom right); this method roughly corresponds to LossSHAP (Lundberg et al., 2020), although it uses the surrogate approach as a conditional distribution approximation. For the digit displayed in Figure 9, the explanation highlights two regions at the top and bottom that distinguish the four from an eight or a nine; it has minimal noise artifacts; and it indicates that the unusual curvature on the left side of the four may hurt the model's loss, which is highlighted by only a few other explanations. Appendix G shows more LossSHAP explanations on MNIST digits.

To avoid the pitfalls of a purely qualitative analysis, we also evaluate these explanations using quantitative metrics. Many works have proposed techniques for evaluating explanation methods (Cao et al., 2015; Ancona et al., 2017; Petsiuk et al., 2018; Zhang et al., 2018; Adebayo et al., 2018; Hooker et al., 2018; Lundberg et al., 2020; Jethani et al., 2021a), including sanity checks and measures of correctness. Among these, several consider human-assigned definitions of importance (e.g., the pointing game), while others focus on the model's prediction mechanism. Most of the model-focused metrics involve removing features and measuring their impact on the model's predictions (Ancona et al., 2017; Petsiuk et al., 2018; Hooker et al., 2018; Lundberg et al., 2020; Jethani et al., 2021a), an approach that we explore here.

Evaluation metrics that rely on removing, masking or deleting features have a clear connection with our framework: they mirror the process that removal-based methods follow when generating explanations. With this perspective, we can see that seemingly neutral

Table 6: Insertion and deletion scores for MNIST explanations when masking removed features with zeros. Results for the favored removal strategy are italicized, and the best scores are bolded (accounting for 95% confidence intervals).

	Insertion (lower is better)					Deletion (higher is better)				
	RI	II	MWI	BV	SV	RI	II	MWI	BV	SV
Default	*0.683*	*0.478*	***0.131***	***0.176***	***0.134***	*4.797*	***8.356***	***9.508***	*5.636*	***10.286***
Uniform	1.523	1.188	0.427	0.791	0.424	4.121	2.325	4.708	4.105	5.003
Product	1.204	1.017	0.204	0.588	0.224	4.316	2.308	5.428	4.415	5.611
Marginal	1.208	1.038	**0.126**	0.335	**0.125**	4.289	3.726	6.213	4.895	6.797
Surrogate	1.745	0.479	0.516	0.803	0.347	1.486	3.239	2.913	1.943	4.443

quantitative metrics may implicitly favor explanations that share their choices of how to remove features from the model. For example, the sensitivity-n metric (Ancona et al., 2017) evaluates the model's predictions when features are replaced with zeros, which favors methods that use this approach when generating explanations (e.g., Occlusion, RISE, CXPlain). Similarly, ROAR (Hooker et al., 2018) measures the decrease in model accuracy after training a new model, which favors methods that account for model retraining.

Rather than relying on a single evaluation metric, we demonstrate how we can create metrics that implicitly favor certain explanations. Following the *insertion* and *deletion* metrics from Petsiuk et al. (2018), we used explanations from each method to produce feature rankings, and we iteratively removed either the most important (deletion) or least important (insertion) features to observe the impact on the model. We evaluated the model's loss for each feature subset, tracing a curve whose area we then measured; we focus on the loss rather than the predictions because our MNIST explanations are based on the model's loss. Critically, we handled removed features using three different strategies: 1) replacing them with zeros, 2) using a surrogate model (Section 8.2), and 3) using a new model trained with missing features. (Note that this choice is made separately from the explanation method.)

Based on this approach, Figure 10 shows examples of insertion and deletion curves for a single MNIST digit. We show curves for only two explanation methods—default values and the surrogate approach, both with Shapley values—and we use two masking strategies to illustrate how each one is advantageous for a particular method. When our evaluation replaces missing features with zeros, the explanation that uses default values produces a better (lower) insertion curve and a better (higher) deletion curve. When we switch to removing features using the surrogate model, the results are reversed, with the explanation that uses the surrogate model performing significantly better. This experiment demonstrates how evaluation metrics can be aligned with the explanation method.

Next, we performed a similar analysis with 100 MNIST digits: we generated insertion and deletion curves for each digit and measured the mean area under the curve (normalized by the number of features). We considered all combinations of removal strategies and summary techniques, and we generated results separately for the three versions of our in-

Table 7: Insertion and deletion scores for MNIST explanations when masking removed features with a surrogate model. Results for the favored removal strategy are italicized, and the best scores are bolded (accounting for 95% confidence intervals).

	Insertion (lower is better)					Deletion (higher is better)				
	RI	II	MWI	BV	SV	RI	II	MWI	BV	SV
Default	0.085	0.152	0.053	0.096	0.100	0.403	0.638	0.325	0.296	0.466
Uniform	0.086	0.080	0.036	0.057	0.038	0.807	0.200	0.836	0.682	0.720
Product	0.068	0.067	0.027	0.046	0.027	0.873	0.179	0.934	0.777	0.888
Marginal	0.067	0.071	0.025	0.033	0.025	0.875	0.189	0.966	1.029	1.451
Surrogate	*0.044*	***0.016***	*0.032*	*0.033*	***0.014***	*0.426*	*1.059*	*1.716*	*0.864*	***3.151***

Table 8: Insertion and deletion scores for MNIST explanations when masking removed features using a model trained with missingness. The best scores for each metric are bolded (accounting for 95% confidence intervals).

	Insertion (lower is better)					Deletion (higher is better)				
	RI	II	MWI	BV	SV	RI	II	MWI	BV	SV
Default	0.060	0.105	0.043	0.054	0.054	0.449	0.757	0.477	0.353	0.690
Uniform	0.112	0.074	0.034	0.052	0.034	0.971	0.220	0.893	0.871	0.833
Product	0.079	0.068	0.024	0.044	0.024	1.080	0.191	1.189	0.933	1.060
Marginal	0.079	0.063	0.023	0.035	0.023	1.054	0.181	1.262	1.327	1.781
Surrogate	0.051	**0.011**	0.035	0.037	**0.015**	0.300	1.189	1.418	0.651	**3.728**

sertion/deletion metrics (Tables 6-8). The results confirm our previous analysis: replacing missing features with zeros favors the default values approach (Table 6), with the corresponding explanations achieving the best insertion and deletion scores; similarly, handling them using a surrogate model favors the surrogate model approach (Table 7). Interestingly, explanations that use the Shapley value produce the best results in both cases; we understand this as a consequence of the metrics checking subsets of all cardinalities, which matches the Shapley value's formulation (Section 7.2).

Evaluating held out features using a separate model trained with missingness does not explicitly favor any explanation method, because it does not mirror the process by which any explanations are generated. Rather, it can be understood as measuring the information content of the features identified by each method, independently of the original model's prediction mechanism. However, we find that the results in Table 8 are very close to those in Table 7, with the surrogate approach outperforming the other removal strategies by a significant margin. We understand this as a consequence of the relationship between the surrogate and the model trained with missingness: although the explanation and evaluation metric are based on different models, both models approximate removing features from the

Bayes classifier using the conditional distribution (Section 8.2). According to this metric, the method that performs best is also the surrogate model and Shapley value combination, or LossSHAP (Lundberg et al., 2020)—the same method that provided the most visually appealing explanations above.

Measuring an explanation's faithfulness to a model is challenging because it requires making a choice for how to accommodate missing features. As our experiments show, the metrics proposed by recent work implicitly favor methods that align with how the metric is calculated. While using a separate model trained with missingness is less obviously aligned with any explanation method, it still favors methods that approximate the conditional distribution; and furthermore, it measures information content rather than faithfulness to the original model. Users must be wary of the hidden biases in available metrics, and they should rely on evaluations that are most relevant to their intended use-case.

10.3 Breast cancer subtype classification

In our final experiment, we analyzed gene microarray data from The Cancer Genome Atlas (TCGA)[12] for breast cancer (BRCA) patients whose tumors were categorized into different molecular subtypes (Berger et al., 2018). Due to the small dataset size (only 510 patients), we prevented overfitting by analyzing a random subset of 100 genes (details in Appendix G) and training a regularized logistic regression model. Rather than explaining individual predictions, we explained the dataset loss to determine which genes contain the most information about BRCA subtypes.

We used the same removal and summary strategies as in the previous experiments, including using the mean expression as default values and training a surrogate model to approximate the conditional distribution. Figure 11 shows a grid of explanations generated by each method. Three of these explanations correspond to existing approaches (permutation tests, conditional permutation tests, SAGE), but many are new methods. We observe that most explanations identify the same gene as being most important (ESR1), but, besides this, the explanations are difficult to compare qualitatively. In Appendix G, we measure the similarity between each of these explanations, finding that there are often similarities across removal strategies and sometimes across summary techniques.

For a quantitative evaluation, we assessed each method by training new models with the top-ranked genes. The best performing method for the top n genes may differ for different values of n, so we trained separate models with the top $n = 1, 2, \ldots, 20$ genes and averaged their performance (Figure 12). This is similar to the insertion metric (Petsiuk et al., 2018) used above, except that we measure the dataset loss and focusing on subsets with relatively small cardinalities. The results represent each method's usefulness for performing global feature selection.

According to Figure 12, the surrogate and Shapley value combination performs best overall; this method roughly corresponds to SAGE (Covert et al., 2020).[13] The mean when included summary performs comparably well, and the results are generally better when using the surrogate rather than any other feature removal approach. Including individual

12. `https://www.cancer.gov/tcga`
13. The SAGE authors suggested marginalizing out features with their conditional distribution, but the surrogate model approximation had not been developed at the time of publication (Covert et al., 2020).

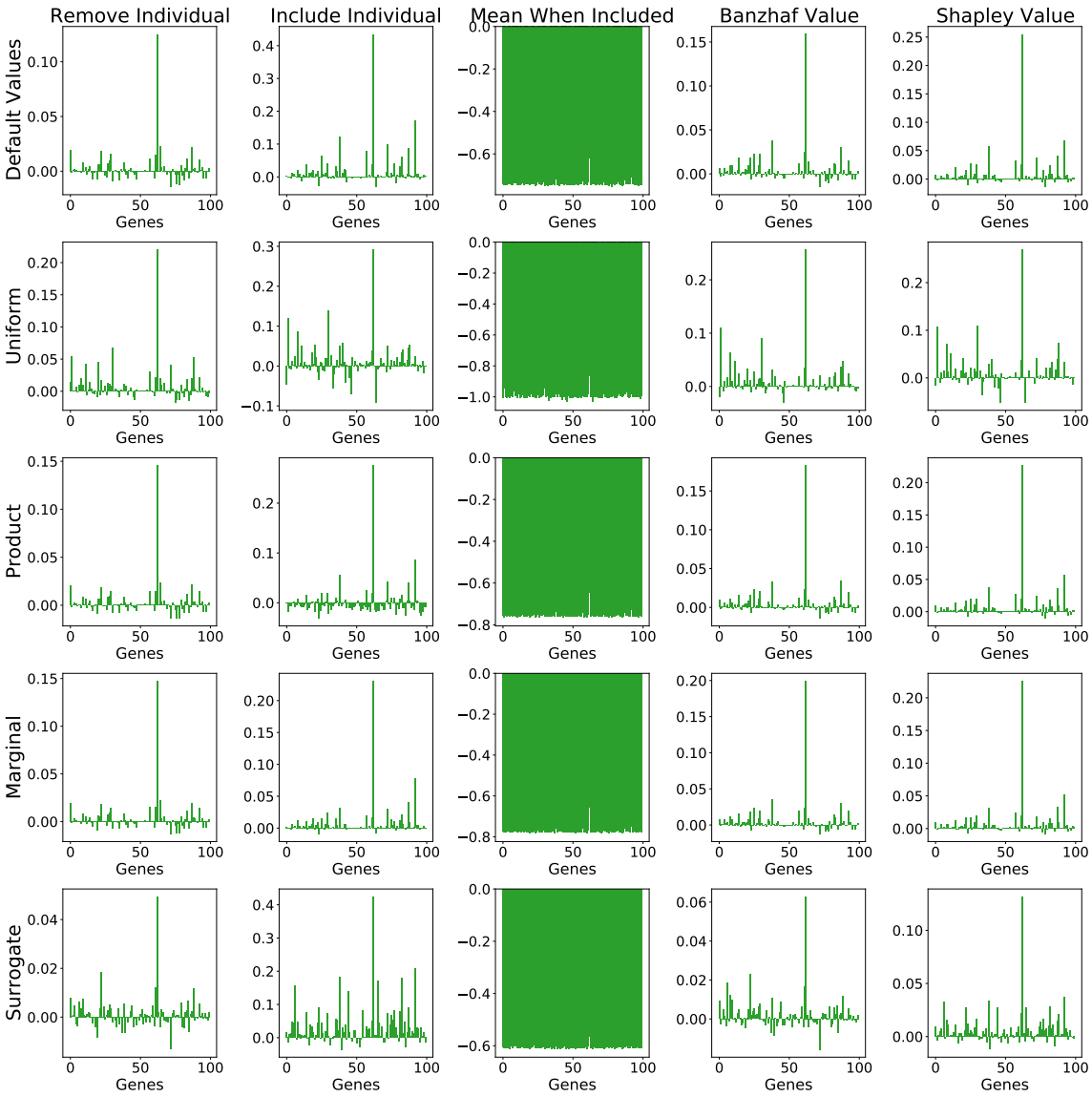

Figure 11: Dataset loss explanations for BRCA subtype classification. Each bar chart represents gene attribution values for a different explanation method, with positive (negative) values improving (hurting) the dataset loss, except for the mean when included technique. The vertical axis represents feature removal strategies and the horizontal axis represents summary techniques.

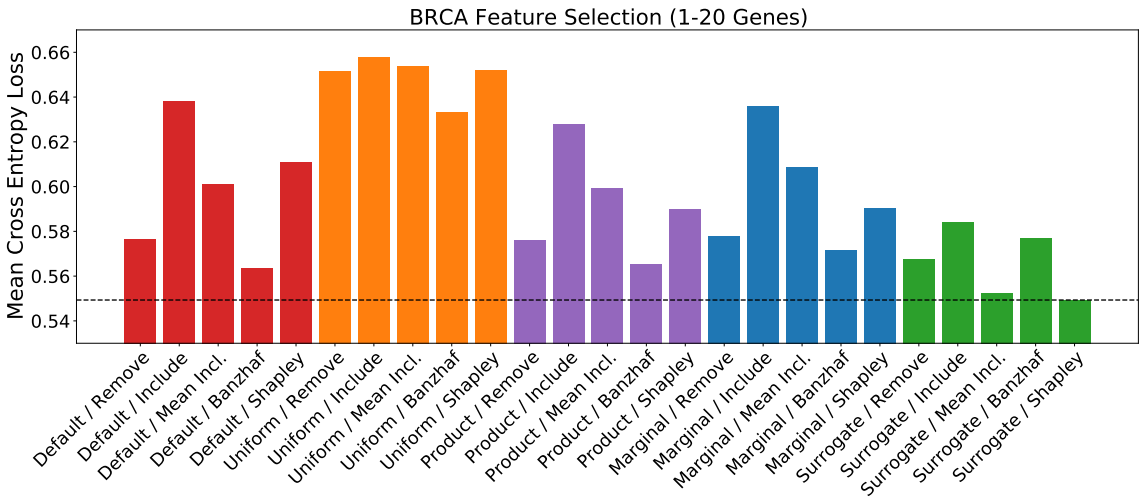

Figure 12: Feature selection results for BRCA subtype classification when using top genes identified by each global explanation. Each bar represents the average loss for models trained using 1-20 top genes (lower is better).

features performs poorly, possibly because it neglects feature interactions; and removing features with uniform distributions consistently yields poor results.

We performed a literature review for the most important genes identified by the surrogate and Shapley value combination (SAGE), and we found that many had documented BRCA associations, including ESR1 (Robinson et al., 2013), CCNB2 (Shubbar et al., 2013) and TXNL4B (Nordgard et al., 2008). Other highly ranked genes had known associations with other cancers, including DDC (Koutalellis et al., 2012) and GSS (Kim et al., 2015). We did not evaluate explanation methods based their ability to identify true associations, however, due to the ambiguity in verifying an association from the literature. Beyond confirming known relationships, we remark that these global explanations can also be used to generate new scientific hypotheses to be tested in a lab setting.

11. Discussion

In this work, we developed a unified framework that encompasses a significant portion of the model explanation field, including 26 existing methods. These methods vary in numerous ways and represent disparate parts of the explainability literature, but our framework systematizes them by showing that each method is specified by three precise mathematical choices:

1. **How the method removes features.** Each method specifies a subset function $F \in \mathfrak{F}$ to make predictions with subsets of features, often based on an existing model $f \in \mathcal{F}$ trained using all the features.

2. **What model behavior the method analyzes.** Each method relies on a set function $u : \mathcal{P}(D) \mapsto \mathbb{R}$ to represent the model's dependence on different groups of features.

The set function describes the model's behavior either for an individual prediction (local explanations) or across the entire dataset (global explanations).

3. **How the method summarizes each feature's influence.** Each method generates explanations that provide a concise summary of the features' contributions to the set function $u \in \mathcal{U}$. Mappings of the form $E : \mathcal{U} \mapsto \mathbb{R}^d$ generate feature attribution explanations, and mappings of the form $E : \mathcal{U} \mapsto \mathcal{P}(D)$ generate feature selection explanations.

Our framework reveals that the field is highly interconnected: we find that many state-of-the-art methods are built on the same foundation, are based on different combinations of interchangeable choices, and in many cases differ along only one or two dimensions (recall Table 2). This perspective provides a systematic characterization of the literature and makes it easier to reason about the conceptual and computational advantages of different approaches.

To shed light on the relationships and trade-offs between different methods, we explored perspectives from three related fields that have been overlooked by most explainability research: cooperative game theory (Section 7), information theory (Section 8) and cognitive psychology (Section 9). We found that removal-based explanations are a simple application of *subtractive counterfactual reasoning*, or, equivalently, of Mill's *method of difference*; and we showed that most explanation methods can be interpreted using existing ideas in cooperative game theory and information theory, in many cases leading to a richer understanding of how each method should be interpreted.

Consulting the game theory literature helped us draw connections between several approaches that can be viewed as *probabilistic values* (in the game-theoretic sense), and which are equivalent to fitting a weighted additive model to a cooperative game (Ribeiro et al., 2016). We also found that many feature selection methods can be understood in terms of coalitional excess, and that these approaches are generalized by the optimization problem solved in the Masking Model approach (Dabkowski and Gal, 2017). These game-theoretic connections allow us to compare the various summarization techniques through the properties they satisfy (e.g., the Shapley axioms), as well as through their ease of interpretation for end-users (Section 9).

Because of its axiomatic derivation and its many desirable properties, the Shapley value provides, in our view, the most complete summary of how a model works. However, while several approaches provide fast Shapley value approximations (Štrumbelj and Kononenko, 2010; Lundberg and Lee, 2017; Lundberg et al., 2020; Covert and Lee, 2021), these techniques are considerably slower than the fastest removal-based explanations (Section 6.2), particularly in the model-agnostic setting. Furthermore, Shapley value-based explanations may not always be the best choice: they can be difficult for users to interpret due to their complexity, and variations of the Shapley value may be required to reflect causal relationships in the data (Frye et al., 2019; Heskes et al., 2020; Wang et al., 2021).

Building on work that discusses probabilistic and information-theoretic explanations (Owen, 2014; Chen et al., 2018a; Covert et al., 2020), we found that multiple model behaviors can be understood as information-theoretic quantities. These connections require that removed features are marginalized out using their conditional distribution, and although this approach is challenging to implement, we showed that several removal strategies provide

high-quality conditional distribution approximations (e.g., generative modeling approaches and models trained with missing features). Our work is among a growing number of papers lending support to this approach (Strobl et al., 2008; Zintgraf et al., 2017; Agarwal and Nguyen, 2019; Aas et al., 2019; Slack et al., 2020; Covert et al., 2020; Frye et al., 2020), but we present a novel perspective on why this choice is justified: besides providing information-theoretic explanations, we find that marginalizing out features with their conditional distribution is the only approach that yields predictions that satisfy standard probability axioms (Section 8.1).

We recognize, however, that users do not always require information-theoretic explanations. These explanations have the potentially undesirable property that features can be deemed important even if they are not used by the model in a functional sense; this property is useful in certain applications (e.g., bias detection), but in other cases it may be confusing to users. Another concern with this approach is spreading credit among correlated features (Merrick and Taly, 2019; Kumar et al., 2020), because marginalizing out features with their conditional distribution guarantees that perfectly correlated features receive equal attributions (Covert et al., 2020). This property can be avoided by using a different feature removal approach (Janzing et al., 2019), but recent work also argues for sparsifying attributions using a variant of the Shapley value (Frye et al., 2019); interestingly, this is a summary technique (third dimension of our framework) that helps resolve an issue that arises from the feature removal strategy (first dimension of our framework).

The growing interest in black-box ML models has spurred a remarkable amount of model explanation research, and the past decade has yielded numerous publications proposing innovative new methods. However, as the field has matured we have also seen a growing number of unification theories that reveal underlying similarities and implicit relationships among state-of-the-art methods (Lundberg and Lee, 2017; Ancona et al., 2017; Covert et al., 2020). Our framework for removal-based explanations is the broadest unifying theory yet, and it bridges the gap between parts of the explainability literature that may have otherwise been viewed as disjoint. We believe that this work represents an important step towards making the field more organized, rigorous, and easier to navigate.

An improved understanding of the field presents new opportunities for both explainability practitioners and researchers. For practitioners, our framework enables more explicit reasoning about the trade-offs between available explanation tools. The unique advantages of different methods are difficult to understand when each approach is viewed as a monolithic algorithm, but disentangling their choices makes it simpler to reason about their strengths and weaknesses, and potentially develop hybrid methods (as in Section 10).

For researchers, our framework offers a new theoretical perspective that can guide and strengthen ongoing research. Using the tools we developed, future work will be better equipped to (i) specify the dimensions along which new methods differ from existing ones, (ii) distinguish the implicit objectives of new approaches from their approximations, (iii) resolve shortcomings in existing explanation methods using solutions along different dimensions of the framework, and (iv) rigorously justify new approaches in light of their connections with cooperative game theory, information theory and psychology. As the number of removal-based explanations continues to grow, we hope that our framework will serve as a strong foundation upon which future research can build.

Appendix A. Method Details

Here, we provide additional details about the methods discussed in the main text. In several cases, we presented generalized versions of methods that deviated from their descriptions in the original papers, so we clarify those reformulations here.

A.1 Meaningful Perturbations (MP)

Meaningful Perturbations (Fong and Vedaldi, 2017) considers multiple ways of deleting information from an input image. Given a mask $m \in [0,1]^d$, MP uses a function $\Phi(x,m)$ to denote the modified input and suggests that the mask may be used to 1) set pixels to a constant value, 2) replace them with Gaussian noise, or 3) blur the image. In the blurring approach, which the authors recommend, each pixel x_i is blurred separately using a Gaussian kernel with standard deviation given by $\sigma \cdot m_i$ (for a user-specified $\sigma > 0$).

To prevent adversarial solutions, MP incorporates a total variation norm on the mask, upsamples the mask from a low-resolution version, and uses a random jitter on the image during optimization. Additionally, MP uses a continuous mask $m \in [0,1]^d$ in place of a binary mask $\{0,1\}^d$ and a continuous ℓ_1 penalty on the mask in place of the ℓ_0 penalty. Although MP's optimization tricks are key to providing visually compelling explanations, our presentation focuses on the most important part of the optimization objective, which is reducing the classification probability while blurring only a small part of the image (Eq. 28).

A.2 Extremal Perturbations (EP)

Extremal Perturbations (Fong et al., 2019) is an extension of MP with several modifications. The first is switching the objective from a "removal game" to a "preservation game," which means learning a mask that preserves rather than removes the most salient information. The second is replacing the penalty on the subset size (or the mask norm) with a constraint. In practice, the constraint is enforced using a penalty, but the authors argue that it should be viewed as a constraint due to the use of a large regularization parameter.

EP uses the same blurring operation as MP and introduces new tricks to ensure a smooth mask, but our presentation focuses on the most important part of the optimization problem, which is maximizing the classification probability while blurring a fixed portion of the image (Eq. 30).

A.3 FIDO-CA

FIDO-CA (Chang et al., 2018) is similar to EP, but it replaces the blurring operation with features drawn from a generative model. The generative model p_G can condition on arbitrary subsets of features, and although its samples are non-deterministic, FIDO-CA achieves strong results using a single sample. The authors consider multiple generative models but recommend a generative adversarial network (GAN) that uses contextual attention (Yu et al., 2018). The optimization objective is based on the same "preservation game" as EP, and the authors use the Concrete reparameterization trick (Maddison et al., 2016) for optimization.

A.4 Minimal Image Representation (MIR)

In the Minimal Image Representation approach, Zhou et al. (2014) remove information from an image to determine which regions are salient for the desired class. MIR creates a segmentation of edges and regions and iteratively removes segments from the image (selecting those that least decrease the classification probability) until the remaining image is incorrectly classified. We view this as a greedy approach for solving the constrained optimization problem

$$\min_{S} \ |S| \quad \text{s.t.} \ \ u(S) \geq t,$$

where $u(S)$ represents the prediction with the specified subset of features, and t represents the minimum allowable classification probability. Our presentation of MIR in the main text focuses on this view of the optimization objective rather than the specific choice of greedy algorithm (Eq. 29).

A.5 Masking Model (MM)

The Masking Model approach (Dabkowski and Gal, 2017) observes that removing salient information (while preserving irrelevant information) and removing irrelevant information (while preserving salient information) are both reasonable approaches for understanding image classifiers. The authors refer to these tasks as discovering the smallest destroying region (SDR) and smallest sufficient region (SSR).

The authors adopt notation similar to Fong and Vedaldi (2017), using $\Phi(x, m)$ to denote the transformation to the input given a mask $m \in [0, 1]^d$. For an input $x \in \mathcal{X}$, MM aims to solve the following optimization problem, which requires four hyperparameters:

$$\min_{m} \ \lambda_1 \text{TV}(m) + \lambda_2 ||m||_1 - \log f\big(\Phi(x, m)\big) + \lambda_3 f\big(\Phi(x, 1-m)\big)^{\lambda_4}.$$

The TV (total variation) and ℓ_1 penalty terms are both similar to MP and respectively encourage smoothness and sparsity in the mask. Unlike MP, MM learns a global explainer model that outputs approximate solutions to this problem in a single forward pass. In the main text, we present a simplified version of the problem that does not include the logarithm in the third term or the exponent in the fourth term (Eq. 32). We view these as monotonic link functions that provide a more complex trade-off between the objectives, but that are not necessary for finding informative solutions.

A.6 Learning to Explain (L2X)

The L2X method performs instance-wise feature selection by learning an auxiliary model g_α and a selector model V_θ (see Eq. 6 of Chen et al., 2018a). These models are learned jointly and are optimized via the similarity between predictions from g_α and from the original model, denoted as $\mathbb{P}_m$ by Chen et al. (2018a). With slightly modified notation that highlights the selector model's dependence on X, the L2X objective can be written as:

$$\max_{\alpha, \theta} \ \mathbb{E}_{X,\zeta} \mathbb{E}_{Y \sim \mathbb{P}_m(X)} \Big[\log g_\alpha\big(V_\theta(X, \zeta) \odot X, Y\big) \Big]. \tag{54}$$

In Eq. 54, the random variables X and ζ are sampled independently, Y is sampled from the model's distribution $\mathbb{P}_m(X)$, $V_\theta(X, \zeta) \odot X$ represents an element-wise multiplication with (approximately) binary indicator variables $V_\theta(X, \zeta)$ sampled from the Concrete distribution (Maddison et al., 2016), and $\log g_\alpha(\cdot, Y)$ represents the model's estimate of Y's log-likelihood.

We can gain more insight into this objective function by reformulating it. If we let $V_\theta(X, \zeta)$ be a deterministic function $\epsilon(X)$, interpret the log-likelihood as a loss function ℓ for the prediction from g_α (e.g., cross entropy loss) and represent g_α as a subset function F, then we can rewrite the L2X objective as follows:

$$\max_{F,\epsilon} \mathbb{E}_X \mathbb{E}_{Y \sim \mathbb{P}_m(X)} \Big[\ell\big(F\big(X, \epsilon(X)\big), Y\big) \Big].$$

Next, rather than considering the expected loss for labels Y distributed according to $\mathbb{P}_m(X)$, we can rewrite this as a loss between the subset function's prediction $F\big(X, \epsilon(X)\big)$ and the full model prediction $f(X) \equiv \mathbb{P}_m(X)$:

$$\max_{F,\epsilon} \mathbb{E}_X \Big[\ell\big(F\big(X, \epsilon(X)\big), f(X)\big) \Big].$$

Finally, we can see that L2X implicitly trains a surrogate model F to match the original model's predictions, and that the optimization objective for each input $x \in \mathcal{X}$ is given by

$$S^* = \arg\max_{|S|=k} \ell\big(F(x_S), f(x)\big).$$

This matches the description of L2X provided in the main text (Eqs. 9, 17, 30). It is only when we have $f(x) = \mathbb{P}_m(x) = p(Y \mid X = x)$ and $F(x_S) = p(Y \mid X_S = x_S)$ that L2X's information-theoretic interpretation holds, at least in the classification case. Or, in the regression case, where we can replace the log-likelihood with a simpler MSE loss, L2X can be interpreted in terms of conditional variance minimization (rather than mutual information maximization) when we have $f(x) = \mathbb{E}[Y \mid X = x]$ and $F(x_S) = \mathbb{E}[Y \mid X_S = x_S]$.

A.7 Instance-wise Variable Selection (INVASE)

INVASE (Yoon et al., 2018) is similar to L2X in that it performs instance-wise feature selection using a learned selector model. However, INVASE has several differences in its implementation and objective function. INVASE relies on three separate models: a prediction model, a baseline model and a selector model. The baseline model is trained to predict the true label Y given the full feature vector X, and it can be trained independently of the remaining models; the predictor model makes predictions given subsets of features X_S (with S sampled according to the selector model), and it is trained to predict the true labels Y; and finally, the selector model takes a feature vector X and outputs a probability distribution for a subset S.

The selector model, which ultimately outputs explanations, relies on the baseline model primarily for variance reduction purposes (Yoon et al., 2018). Because the sampled subsets are used only for the predictor model, which is trained to predict the true label Y (rather than the baseline model's predictions), we view the prediction model as the model

being explained, and we understand it as removing features via a strategy of introducing missingness during training (Eq. 10).

For the optimization objective, Yoon et al. (2018) explain that their aim is to minimize the following KL divergence for each input $x \in \mathcal{X}$:

$$S^* = \arg\min_S \; D_{\mathrm{KL}}\big(p(Y \mid X = x) \;\big|\big|\; p(Y \mid X_S = x_S)\big) + \lambda|S|.$$

This is consistent with their learning algorithm if we assume that the predictor model outputs the Bayes optimal prediction $p(Y \mid X_S = x_S)$. If we denote their predictor model as a subset function F and interpret the KL divergence as a loss function with the true label Y (i.e., cross entropy loss), then we can rewrite this objective as follows:

$$S^* = \arg\min_S \; \mathbb{E}_{Y|X=x}\Big[\ell\big(F(x_S), Y\big)\Big] + \lambda|S|.$$

This is the description of INVASE provided in the main text.

A.8 REAL-X

REAL-X (Jethani et al., 2021a) is similar to L2X and INVASE in that it uses a learned selector model to perform instance-wise feature selection. REAL-X is designed to resolve a flaw in L2X and INVASE, which is that both methods learn the selector model jointly with their subset functions, enabling label information to be leaked via the selected subset S.

To avoid this issue, REAL-X learns a subset function F independently from the selector model using the following objective function (with modified notation):

$$\min_F \; \mathbb{E}_X \mathbb{E}_{Y \sim f(X)} \mathbb{E}_S \Big[\ell\big(F(X_S), Y\big)\Big].$$

The authors point out that Y may be sampled from its true conditional distribution $p(Y \mid X)$ or from a model's distribution $Y \sim f(X)$; we remark that the former is analogous to INVASE (missingness introduced during training) and that the latter is analogous to L2X (training a surrogate model). Notably, unlike L2X or INVASE, REAL-X optimizes its subset function with the subsets S sampled independently from the input X, enabling it to approximate the Bayes optimal predictions $F(x_S) \approx p(Y \mid X_S = x_S)$ (Appendix E).

We focus on the case with the label sampled according to $Y \sim f(X)$, which can be understood as fitting a surrogate model F to the original model f. With the learned subset function F fixed, REAL-X then learns a selector model that optimizes the following objective for each input $x \in \mathcal{X}$:

$$S^* = \arg\min_S \; \mathbb{E}_{Y \sim f(x)}\Big[\ell\big(F(x_S), Y\big)\Big] + \lambda|S|.$$

Rather than viewing this as the mean loss for labels sampled according to $f(x)$, we interpret this as a loss function between $F(x_S)$ and $f(x)$, as we did with L2X:

$$S^* = \arg\min_S \; \ell\big(F(x_S), f(x)\big) + \lambda|S|.$$

This is our description of REAL-X provided in the main text.

A.9 Prediction Difference Analysis (PredDiff)

Prediction Difference Analysis (Zintgraf et al., 2017), which is based on a precursor method to IME (Robnik-Šikonja and Kononenko, 2008; Štrumbelj et al., 2009), removes individual features (or groups of features) and analyzes the difference in the model's prediction. Removed pixels are imputed by conditioning on their bordering pixels, which approximates sampling from the full conditional distribution. The authors then calculate attribution scores based on the difference in log-odds ratio:

$$a_i = \log \frac{F(x)}{1 - F(x)} - \log \frac{F(x_{D\setminus\{i\}})}{1 - F(x_{D\setminus\{i\}})}.$$

We view this as another way of analyzing the difference in the model output for an individual prediction, simply substituting the log-odds ratio for the classification probabilities.

A.10 Causal Explanations (CXPlain)

CXPlain removes single features (or groups of features) for individual inputs and measures the change in the loss function (Schwab and Karlen, 2019). The authors propose calculating the attribution values

$$a_i(x) = \ell\big(F(x_{D\setminus\{i\}}), y\big) - \ell\big(F(x, y)\big)$$

and then computing the normalized values

$$w_i(x) = \frac{a_i(x)}{\sum_{j=1}^{d} a_j(x)}.$$

The normalization step enables the use of a KL divergence-based learning objective for the explainer model (although it is not obvious how to handle negative values $a_i(x) < 0$), which is ultimately used to calculate attribution values in a single forward pass. The authors explain that this approach is based on a "causal objective," but CXPlain is only causal in the same sense as every other method described in our work (Section 9), i.e., it measures how each feature causes the model output to change. This is not to be confused with methods that integrate ideas from causal inference (Heskes et al., 2020; Wang et al., 2021).

A.11 Randomized Input Sampling for Explanation (RISE)

RISE (Petsiuk et al., 2018) begins by generating a large number of randomly sampled binary masks. In practice, the masks are sampled by dropping features from a low-resolution mask independently with probability p, upsampling to get an image-sized mask, and then applying a random jitter. Due to the upsampling, the masks have values $m \in [0, 1]^d$ rather than $m \in \{0, 1\}^d$.

The mask generation process induces a distribution over the masks, which we denote as $p(m)$. The method then uses the randomly generated masks to obtain a Monte Carlo estimate of the following attribution values:

$$a_i = \frac{1}{\mathbb{E}[M_i]} \mathbb{E}_{p(M)}\big[f(x \odot M) \cdot M_i\big].$$

If we ignore the upsampling step that creates continuous mask values, we see that these attribution values are the mean prediction when a given pixel is included:

$$\begin{aligned} a_i &= \frac{1}{\mathbb{E}[M_i]} \mathbb{E}_{p(M)}\big[f(x \odot M) \cdot M_i\big] \\ &= \sum_{m \in \{0,1\}^d} f(x \odot m) \cdot m_i \cdot \frac{p(m)}{\mathbb{E}[M_i]} \\ &= \mathbb{E}_{p(M|M_i=1)}\big[f(x \odot M)\big]. \end{aligned}$$

This is our description of RISE provided in the main text.

A.12 Interactions Methods for Explanations (IME)

IME was presented in two separate papers (Štrumbelj et al., 2009; Štrumbelj and Kononenko, 2010). In the original version, the authors recommended training a separate model for each subset of features; in the second version, the authors proposed the more efficient approach of marginalizing out the removed features from a single model f.

The latter paper is somewhat ambiguous about the specific distribution used when marginalizing out held out features. Lundberg and Lee (2017) view that features are marginalized out using their distribution from the training dataset (i.e., the marginal distribution), whereas Merrick and Taly (2019) view IME as marginalizing out features using a uniform distribution. We opt for the uniform interpretation, but IME's specific choice of distribution does not impact any of our conclusions.

A.13 SHAP

SHAP (Lundberg and Lee, 2017) explains individual predictions by decomposing them with the game-theoretic Shapley value (Shapley, 1953), similar to IME (Štrumbelj et al., 2009; Štrumbelj and Kononenko, 2010) and QII (Datta et al., 2016). The original work proposed marginalizing out removed features with their conditional distribution but remarked that the joint marginal provided a practical approximation (see Section 8.2 for a similar argument). Marginalizing using the joint marginal distribution is now the default behavior in SHAP's online implementation. KernelSHAP is an approximation approach based on solving a weighted least squares problem (Lundberg and Lee, 2017; Covert and Lee, 2021).

A.14 TreeSHAP

TreeSHAP uses a unique approach to handle held out features in tree-based models (Lundberg et al., 2020): it accounts for missing features using the distribution induced by the underlying trees, which, since it exhibits no dependence on the held out features, is a valid subset extension of the original model. However, it cannot be viewed as marginalizing out features using a simple distribution (i.e., one whose density function we can write down).

Given a subset of features, TreeSHAP makes a prediction separately for each tree and then combines each tree's prediction in the standard fashion. But when a split for an unknown feature is encountered, TreeSHAP averages predictions over the multiple paths in proportion to how often the dataset follows each path. This is similar but not identical to

the conditional distribution because each time this averaging step is performed, TreeSHAP conditions only on coarse information about the features that preceded the split.

A.15 LossSHAP

LossSHAP is a version of SHAP that decomposes the model's loss for an individual prediction rather than the prediction itself. The approach was first considered in the context of TreeSHAP (Lundberg et al., 2020), and it has been discussed in more detail as a local analogue to SAGE (Covert et al., 2020).

A.16 Shapley Net Effects

Shapley Net Effects (Lipovetsky and Conklin, 2001) was proposed as a variable importance measure for linear models. The method becomes impractical with large numbers of features or non-linear models, but Williamson and Feng (2020) generalize the approach by using an efficient linear regression-based Shapley value estimator: SPVIM can be run with larger datasets and non-linear models because it requires checking a smaller number of feature subsets. Both Shapley Net Effects and SPVIM can also be used with other model performance measures, such as area under the ROC curve or the R^2 value.

A.17 Shapley Effects

Shapley Effects analyzes a variance-based measure of a function's sensitivity to its inputs, with the goal of discovering which features are responsible for the greatest variance reduction in the model output (Owen, 2014). The cooperative game described in the original paper is the following:

$$u(S) = \mathrm{Var}\Big(\mathbb{E}\big[f(X) \mid X_S\big]\Big).$$

We present a generalized version to cast this method in our framework. In the appendix of Covert et al. (2020), it was shown that this game can be reformulated as follows:

$$\begin{aligned}
u(S) &= \mathrm{Var}\Big(\mathbb{E}\big[f(X) \mid X_S\big]\Big) \\
&= \mathrm{Var}\big(f(X)\big) - \mathbb{E}\Big[\mathrm{Var}\big(f(X) \mid X_S\big)\Big] \\
&= c - \mathbb{E}\Big[\ell\big(\mathbb{E}\big[f(X) \mid X_S\big], f(X)\big)\Big] \\
&= c - \underbrace{\mathbb{E}\Big[\ell\big(F(X_S), f(X)\big)\Big]}_{\text{Dataset loss w.r.t. output}}.
\end{aligned}$$

This derivation assumes that the loss function ℓ is MSE and that the subset function F is $F(x_S) = \mathbb{E}[f(X) \mid X_S = x_S]$. Rather than the original formulation, we present a cooperative game that is equivalent up to a constant value and that provides flexibility in the choice of loss function:

$$w(S) = -\mathbb{E}\Big[\ell\big(F(X_S), f(X)\big)\Big].$$

A.18 LIME

For an overview of LIME (Ribeiro et al., 2016), we direct readers to Appendix B.

Appendix B. Additive Model Proofs

LIME calculates feature attribution values by fitting a weighted regularized linear model to an *interpretable representation* of the input (Ribeiro et al., 2016). If we consider that the interpretable representation is binary (the default behavior in LIME's implementation), then the model is represented by a set function $u : \mathcal{P}(D) \mapsto \mathbb{R}$ when we take an expectation over the distribution of possible feature imputations. LIME is therefore equivalent to fitting an additive model to a set function, which means solving the optimization problem

$$\min_{b_0,\ldots,b_d} L(b_0,\ldots,b_d) + \Omega(b_1,\ldots,b_d),$$

where we define L (the weighted least squares component) as

$$L(b_0,\ldots,b_d) = \sum_{S\subseteq D} \pi(S)\Big(b_0 + \sum_{i\in S} b_i - u(S)\Big)^2. \tag{55}$$

For convenience, we refer to this as the weighted least squares (WLS) approach to summarizing set functions, and we show that several familiar attribution values can be derived by choosing different weighting kernels π and omitting the regularization term (i.e., setting $\Omega = 0$).

B.1 Include individual

Consider the weighting kernel $\pi_{\text{Inc}}(S) = \mathbb{1}(|S| \leq 1)$, which puts weight only on coalitions that have no more than one player. With this kernel, the WLS problem reduces to:

$$a_1,\ldots a_d = \underset{b_0,\ldots,b_d}{\arg\min} \Big(b_0 - u(\{\})\Big)^2 + \sum_{i=1}^{d} \Big(b_0 + b_i - u(\{i\})\Big)^2.$$

It is clear that the unique global minimizer of this problem is given by the following solution:

$$\begin{aligned} a_0 &= u(\{\}) \\ a_i &= u(\{i\}) - u(\{\}). \end{aligned}$$

The WLS approach will therefore calculate the attribution values $a_i = u(\{i\}) - u(\{\})$, which is equivalent to how Occlusion, PredDiff and CXPlain calculate local feature importance and how permutation tests and feature ablation (LOCO) summarize global feature importance (Breiman, 2001; Strobl et al., 2008; Zeiler and Fergus, 2014; Zintgraf et al., 2017; Lei et al., 2018; Schwab and Karlen, 2019).

B.2 Remove individual

Consider the weighting kernel $\pi_{\text{Ex}}(S) = \mathbb{1}(|S| \geq d-1)$, which puts weight only on coalitions that are missing no more than one player. With this kernel, the WLS problem reduces to:

$$a_1, \ldots, a_d = \underset{b_0,\ldots,b_d}{\arg\min} \Big(b_0 + \sum_{j \in D} b_j - u(D)\Big)^2 + \sum_{i=1}^{d} \Big(b_0 + \sum_{j \in D \setminus \{i\}} b_j - u(D \setminus \{i\})\Big)^2.$$

It is clear that the unique global minimizer of this problem is given by the following solution:

$$\begin{aligned} a_0 &= u(D) - \sum_{i \in D} a_i \\ a_i &= u(D) - u(D \setminus \{i\}). \end{aligned}$$

The WLS approach will therefore calculate the attribution values $a_i = u(D) - u(D \setminus \{i\})$, which is equivalent to how the univariate predictors approach summarizes global feature importance (Guyon and Elisseeff, 2003).

B.3 Banzhaf value

Consider the weighting kernel $\pi_{\text{B}}(S) = 1$, which yields an unweighted least squares problem. This version of the WLS problem has been analyzed in prior work, which showed that the optimal coefficients are the Banzhaf values (Hammer and Holzman, 1992).

As an alternative proof, we demonstrate that a solution that uses the Banzhaf values is optimal by proving that its partial derivatives are zero. To begin, consider the following candidate solution, which uses the Banzhaf values $a_i = \psi_i(u)$ for $i = 1, \ldots, d$ and a carefully chosen intercept term a_0:

$$\begin{aligned} a_0 &= \frac{1}{2^d} \sum_{S \subseteq D} u(S) - \frac{1}{2} \sum_{j=1}^{d} a_j \\ a_i &= \frac{1}{2^{d-1}} \sum_{S \subseteq D \setminus \{i\}} \Big(u(S \cup \{i\}) - u(S)\Big) = \psi_i(u). \end{aligned}$$

We can verify whether this is a solution to the unweighted least squares problem by checking if the partial derivatives are zero. We begin with the derivative for the intercept:

$$\begin{aligned} \frac{\partial}{\partial b_0} L(a_0, \ldots, a_d) &= 2 \sum_{S \subseteq D} \Big(a_0 + \sum_{j \in S} a_j - u(S)\Big) \\ &= 2\Big(2^d a_0 + 2^{d-1} \sum_{j=1}^{d} a_j - \sum_{S \subseteq D} u(S)\Big) \\ &= 0. \end{aligned}$$

Next, we verify the derivatives for the other parameters a_i for $i = 1, \ldots, d$:

$$\begin{aligned}
\frac{\partial}{\partial b_i} L(a_0, \ldots, a_d) &= 2 \sum_{T \supseteq \{i\}} \Big(a_0 + \sum_{j \in T} a_j - u(T) \Big) \\
&= 2 \Big(2^{d-1} a_0 + 2^{d-2} \sum_{j=1}^{d} a_j + 2^{d-2} a_i - \sum_{T \supseteq \{i\}} u(T) \Big) \\
&= 2^{d-1} a_i + \sum_{S \subseteq D \setminus \{i\}} \Big(u(S) - u(S \cup \{i\}) \Big) \\
&= 0.
\end{aligned}$$

Because the gradient is zero and the problem is jointly convex in $(b_0, \ldots, b_d)$, we conclude that the solution given above is optimal and unique. The optimal coefficients $(a_1, \ldots, a_d)$ are precisely the Banzhaf values $\psi_1(u), \ldots, \psi_d(u)$ of the cooperative game u.

B.4 Shapley value

The weighted least squares problem is optimized by the Shapley value when we use the following weighting kernel:

$$\pi_{\mathrm{Sh}}(S) = \frac{d - 1}{\binom{d}{|S|} |S| (d - |S|)}.$$

Since this connection has been noted in other model explanation works, we direct readers to existing proofs (Charnes et al., 1988; Lundberg and Lee, 2017).

Appendix C. Axioms for Other Approaches

Here, we describe which Shapley axioms or Shapley-like axioms apply to other summarization approaches.

C.1 Additive model axioms

By fitting a regularized weighted least squares model to a cooperative game, as in LIME (Ribeiro et al., 2016), we effectively create an explanation mapping of the form $E : \mathcal{U} \mapsto \mathbb{R}^d$. We can show that this mapping satisfies a subset of the Shapley value axioms (Section 7.2). To do so, we make the following assumptions about the weighting kernel π and regularization function Ω (introduced in Eq. 26):

1. The weighting kernel π is non-negative and finite for all $S \subseteq D$ except for possibly the sets $\{\}$ and D.

2. The weighting kernel π satisfies the inequality

$$\begin{pmatrix} \mathbf{1} & X \end{pmatrix}^T W \begin{pmatrix} \mathbf{1} & X \end{pmatrix} \succeq 0,$$

where $X \in \mathbb{R}^{2^d \times d}$ contains an enumeration of binary representations for all subsets $S \subseteq D$, and $W \in \mathbb{R}^{2^d \times 2^d}$ is a diagonal matrix containing an aligned enumeration of $\pi(S)$ for $S \subseteq D$. This ensures that the weighted least squares component of the objective is strictly convex.

3. The regularizer Ω is convex (e.g., the Lasso or ridge penalty).

We now address each property in turn for the weighted least squares (WLS) approach:

- (Efficiency) The WLS approach satisfies the efficiency property only when the weighting kernel is chosen such that $\pi(\{\}) = \pi(D) = \infty$. These weights are equivalent to the constraints $b_0 = u(\{\})$ and $\sum_{i \in D} b_i = u(D) - u(\{\})$. In cooperative game theory, additive models with these constraints are referred to as *faithful linear approximations* (Hammer and Holzman, 1992).

- (Symmetry) The WLS approach satisfies the symmetry axiom as long as the weighting kernel π and the regularizer Ω are permutation-invariant (i.e., π is a function of the subset size, and Ω is invariant to the ordering of parameters). To see this, consider an optimal solution with parameters $(b_0^*, \dots b_d^*)$. Swapping the coefficients b_i^* and b_j^* for features with identical marginal contributions gives the same objective value, so this is still optimal. The strict convexity of the objective function implies that there is a unique global optimum, so we conclude that $b_i^* = b_j^*$.

- (Dummy) The dummy property holds for the Shapley and Banzhaf weighting kernels π_{B} and π_{Sh}, but it does not hold for arbitrary π, Ω.

- (Additivity) The additivity property holds when the regularizer is set to $\Omega = 0$. This can be seen by viewing the solution to the WLS problem as a linear function of the response variables (Kutner et al., 2005).

- (Marginalism) Given two games u, u' where players have identical marginal contributions, we can see that $u' = u + c$ for some $c \in \mathbb{R}$. The WLS approach satisfies the marginalism property because it learns identical coefficients $b_1^*, \dots, b_d^*$ but different intercepts connected by the equation $b_0^*(u') = b_0^*(u) + c$.

C.2 Feature selection axioms

The feature selection summarization techniques (MP, MIR, L2X, INVASE, REAL-X, EP, MM, FIDO-CA) satisfy properties that are similar to the Shapley value axioms. Each method outputs an optimal coalition $S^* \subseteq D$ rather than an allocation $a \in \mathbb{R}^d$, so the Shapley value axioms do not apply directly. However, we identify the following analogous properties:

- (Symmetry) If there are two players i, j with identical marginal contributions and there exists an optimal coalition S^* that satisfies $i \in S$ and $j \notin S$, then the coalition $\big(S^* \cup \{j\}\big) \setminus \{i\}$ is also optimal.

- (Dummy) For a player i that makes zero marginal contribution, there must be an optimal solution S^* such that $i \notin S^*$.

- (Marginalism) For two games u, u' where all players have identical marginal contributions, the coalition S^* is optimal for u if and only if it is optimal for u'.

The feature selection explanations do not seem to satisfy properties that are analogous to the efficiency or additivity axioms. And unlike the attribution case, these properties are insufficient to derive a unique, axiomatic approach.

Appendix D. Consistency Proofs

Here, we restate and prove the results from Section 8.1, which relate to subset extensions of a model $f \in \mathcal{F}$ that are consistent (Definition 5) with a probability distribution $q(X)$. Both results can be shown using simple applications of basic probability laws.

Proposition 6 *For a classification model $f \in \mathcal{F}$ that estimates a discrete Y's conditional probability, there is a unique subset extension $F \in \mathfrak{F}$ that is consistent with $q(X)$,*

$$F(x_S) = \mathbb{E}_{q(X_{\bar{S}}|X_S=x_S)}[f(x_S, X_{\bar{S}})],$$

where $q(X_{\bar{S}} \mid X_S = x_S)$ is the conditional distribution induced by $q(X)$.

Proof We begin by assuming the existence of a subset function $F \in \mathfrak{F}$ that satisfies $q(Y \mid X = x) = F(x) = f(x)$ for all $x \in \mathcal{X}$ and consider how the probability axioms can be used to compute the conditional probability $q(Y \mid X_S = x_S)$ given only $q(X)$ and $q(Y \mid X = x)$.

For this proof we consider the case of discrete X, but a similar argument can be used to prove the same result with continuous X. Below, we provide a step-by-step derivation of $q(Y \mid X_S = x_S)$ that indicates which axioms or definitions are used in each step. (Axiom 1 refers to the countable additivity property and Axiom 2 refers to Bayes rule.)

$$\begin{aligned}
q(y \mid x_S) &= \sum_{x_{\bar{S}} \in \mathcal{X}_{\bar{S}}} q(y, x_{\bar{S}} \mid x_S) && \text{(Axiom 1)} \\
&= \sum_{x_{\bar{S}} \in \mathcal{X}_{\bar{S}}} \frac{q(y, x_S, x_{\bar{S}})}{q(x_S)} && \text{(Axiom 2)} \\
&= \sum_{x_{\bar{S}} \in \mathcal{X}_{\bar{S}}} q(y \mid x_S, x_{\bar{S}}) \frac{q(x_S, x_{\bar{S}})}{q(x_S)} && \text{(Axiom 2)} \\
&= \sum_{x_{\bar{S}} \in \mathcal{X}_{\bar{S}}} f(x_S, x_{\bar{S}}) \cdot q(x_{\bar{S}} \mid x_S) && \text{(Definition of } f\text{, Axiom 2)} \\
&= \mathbb{E}_{q(X_{\bar{S}}|X_S=x_S)}\big[f(x_S, X_{\bar{S}})\big] && \text{(Definition of expectation)}
\end{aligned}$$

This derivation shows that in order to be consistent with $q(X)$ according to the probability laws, F must be defined as follows:

$$F(x_S) \equiv \mathbb{E}_{q(X_{\bar{S}}|X_S=x_S)}\big[f(x_S, X_{\bar{S}})\big]. \tag{56}$$

To complete the proof, we consider whether there are other ways of deriving F's behavior that may demonstrate inconsistency. The only other case to consider is whether we can derive a unique definition for $F(x_{S\setminus T})$ when beginning from $F(x)$ and when beginning from $F(x_S)$ for $T \subset S \subset D$. The first result is given by Eq. 56, and we derive the second result as follows:

$$
\begin{aligned}
F(x_{S\setminus T}) &= q(y \mid x_{S\setminus T}) \\
&= \sum_{x_T \in \mathcal{X}_T} q(y, x_T \mid x_{S\setminus T}) && \text{(Axiom 1)} \\
&= \sum_{x_T \in \mathcal{X}_T} \frac{q(y, x_T, x_{S\setminus T})}{q(x_{S\setminus T})} && \text{(Axiom 2)} \\
&= \sum_{x_T \in \mathcal{X}_T} q(y \mid x_T, x_{S\setminus T}) \frac{q(x_T, x_{S\setminus T})}{q(x_{S\setminus T})} && \text{(Axiom 2)} \\
&= \sum_{x_T \in \mathcal{X}_T} q(y \mid x_T, x_{S\setminus T}) \cdot q(x_T \mid x_{S\setminus T}) && \text{(Axiom 2)} \\
&= \sum_{x_T \in \mathcal{X}_T} F(x_T, x_{S\setminus T}) \cdot q(x_T \mid x_{S\setminus T}) && \text{(Definition of } F\text{)} \\
&= \sum_{x_T \in \mathcal{X}_T} \mathbb{E}_{q(X_{\bar{S}} \mid X_S = x_S)}\big[f(x_T, x_{S\setminus T}, X_{\bar{S}})\big] \cdot q(x_T \mid x_{S\setminus T}) && \text{(Definition of } F\text{)} \\
&= \sum_{x_T \in \mathcal{X}_T} \sum_{x_{\bar{S}} \in \mathcal{X}_{\bar{S}}} f(x_T, x_{S\setminus T}, x_{\bar{S}}) \cdot q(x_{\bar{S}} \mid x_T, x_{S\setminus T}) \cdot q(x_T \mid x_{S\setminus T}) && \text{(Expectation)} \\
&= \sum_{x_T \in \mathcal{X}_T} \sum_{x_{\bar{S}} \in \mathcal{X}_{\bar{S}}} f(x_T, x_{S\setminus T}, x_{\bar{S}}) \cdot q(x_T, x_{\bar{S}} \mid x_{S\setminus T}) && \text{(Axiom 2)} \\
&= \mathbb{E}_{q(X_T, X_{\bar{S}} \mid X_{S\setminus T} = x_{S\setminus T})}\big[f(X_T, x_{S\setminus T}, X_{\bar{S}})\big] && \text{(Definition of expectation)}
\end{aligned}
$$

This result shows that deriving $F(x_{S\setminus T})$ from $F(x)$ or from $F(x_S)$ yields a consistent result. We conclude that our definition of F, which marginalizes out missing features with the conditional distribution induced by $q(X)$, provides the unique subset extension of f that is consistent with $q(X)$. ■

Proposition 7 *For a regression model $f \in \mathcal{F}$ that estimates Y's conditional expectation, there is a unique subset extension $F \in \mathfrak{F}$ that is consistent with $q(X)$,*

$$F(x_S) = \mathbb{E}_{q(X_{\bar{S}} \mid X_S = x_S)}[f(x_S, X_{\bar{S}})],$$

where $q(X_{\bar{S}} \mid X_S = x_S)$ is the conditional distribution induced by $q(X)$.

Proof Unlike the previous proof, F does not directly define some $q(Y \mid X_S = x_S)$. However, F represents an estimate of the conditional expectation $\mathbb{E}[Y \mid X_S = x_S]$ for each $S \subseteq D$, so we can assume the existence of conditional distributions $q(Y \mid X_S = x_S)$ that satisfy

$$F(x_S) = \mathbb{E}_{q(Y|X_S=x_S)}[Y].$$

We show that our probability laws are sufficient to constrain the conditional expectation represented by F to have a unique definition for each $S \subset D$.

Consider how the probability laws can be used to compute $q(Y \mid X_S = x_S)$ given only $q(X)$ and the assumed $q(Y \mid X = x)$. For this proof, we consider the case of discrete X and Y, but a similar argument can be used to prove the same result for continuous X and Y. Below, we provide a step-by-step derivation of $q(Y \mid X_S = x_S)$ that indicates which axioms or definitions are used in each step. (Axiom 1 refers to the countable additivity property and Axiom 2 refers to Bayes rule.)

$$\begin{aligned}
q(y \mid x_S) &= \sum_{x_{\bar{S}} \in \mathcal{X}_{\bar{S}}} q(y, x_{\bar{S}} \mid x_S) && \text{(Axiom 1)} \\
&= \sum_{x_{\bar{S}} \in \mathcal{X}_{\bar{S}}} \frac{q(y, x_S, x_{\bar{S}})}{q(x_S)} && \text{(Axiom 2)} \\
&= \sum_{x_{\bar{S}} \in \mathcal{X}_{\bar{S}}} q(y \mid x_S, x_{\bar{S}}) \frac{q(x_S, x_{\bar{S}})}{q(x_S)} && \text{(Axiom 2)} \\
&= \sum_{x_{\bar{S}} \in \mathcal{X}_{\bar{S}}} q(y \mid x_S, x_{\bar{S}}) \cdot q(x_{\bar{S}} \mid x_S) && \text{(Axiom 2)} \\
&= \mathbb{E}_{q(X_{\bar{S}}|X_S=x_S)}\big[q(y \mid x_S, X_{\bar{S}})\big] && \text{(Definition of expectation)}
\end{aligned}$$

This derivation shows that in order to be consistent with $q(X)$, the conditional distribution $q(Y \mid X_S = x_S)$ must be defined as follows:

$$q(Y \mid X_S = x_S) = \mathbb{E}_{q(X_{\bar{S}}|X_S=x_S)}[q(x_S, X_{\bar{S}})].$$

Since F represents the expectation of these distributions, it can be derived as follows:

$$\begin{aligned}
F(x_S) &= \mathbb{E}_{q(Y|X_S=x_S)}[Y] && \text{(Definition of } F\text{)} \\
&= \sum_{y \in \mathcal{Y}} y \cdot q(y \mid x_S) && \text{(Definition of expectation)} \\
&= \sum_{y \in \mathcal{Y}} \sum_{x_{\bar{S}} \in \mathcal{X}_{\bar{S}}} y \cdot q(y \mid x_S, x_{\bar{S}}) \cdot q(x_{\bar{S}} \mid x_S) && \text{(Previous derivation)} \\
&= \sum_{x_{\bar{S}} \in \mathcal{X}_{\bar{S}}} \mathbb{E}[Y \mid x_S, x_{\bar{S}}] \cdot q(x_{\bar{S}} \mid x_S) && \text{(Interchanging order of sums)} \\
&= \sum_{x_{\bar{S}}} f(x_S, x_{\bar{S}}) \cdot q(x_{\bar{S}} \mid x_S) && \text{(Definition of } f\text{)} \\
&= \mathbb{E}_{q(X_{\bar{S}}|X_S=x_S)}\big[f(x_S, X_{\bar{S}})\big] && \text{(Definition of expectation)}
\end{aligned}$$

According to this result, the probability laws imply that F must be defined as follows:

$$F(x_S) \equiv \mathbb{E}_{q(X_{\bar{S}}|X_S=x_S)}\big[f(x_S, X_{\bar{S}})\big]. \tag{57}$$

To complete the proof, we consider whether there are other ways of deriving F's behavior that may demonstrate inconsistency. The only other case to consider is whether we can derive a unique definition for $F(x_{S\setminus T})$ when beginning from $F(x)$ and when beginning from $F(x_S)$ for $T \subset S \subset D$. The first result is given by Eq. 57, and we now derive the second result. To begin, we derive $q(Y \mid X_{S\setminus T})$ from $q(Y \mid X_S)$:

$$\begin{aligned}
q(y \mid x_{S\setminus T}) &= \sum_{x_T \in \mathcal{X}_T} q(y, x_T \mid x_{S\setminus T}) && \text{(Axiom 1)} \\
&= \sum_{x_T \in \mathcal{X}_T} \frac{q(y, x_T, x_{S\setminus T})}{q(x_{S\setminus T})} && \text{(Axiom 2)} \\
&= \sum_{x_T \in \mathcal{X}_T} q(y \mid x_T, x_{S\setminus T}) \frac{q(x_T, x_{S\setminus T})}{q(x_{S\setminus T})} && \text{(Axiom 2)} \\
&= \sum_{x_T \in \mathcal{X}_T} q(y \mid x_T, x_{S\setminus T}) \cdot q(x_T \mid x_{S\setminus T}) && \text{(Axiom 2)} \\
&= \mathbb{E}_{q(X_T|X_{S\setminus T}=x_{S\setminus T})}\big[q(y \mid x_T, x_{S\setminus T})\big] && \text{(Definition of expectation)}
\end{aligned}$$

We can now derive $F(x_{S\setminus T})$ by taking the expectation of this distribution:

$$\begin{aligned}
F(x_{S\setminus T}) &= \mathbb{E}_{q(Y|X_{S\setminus T}=x_{S\setminus T})}[Y] && \text{(Definition of } F\text{)} \\
&= \sum_{y \in \mathcal{Y}} y \cdot q(y \mid x_{S\setminus T}) && \text{(Definition of expectation)} \\
&= \sum_{y \in \mathcal{Y}} \sum_{x_T \in \mathcal{X}_T} y \cdot q(y \mid x_T, x_{S\setminus T}) \cdot q(x_T \mid x_{S\setminus T}) && \text{(Previous derivation)} \\
&= \sum_{x_T \in \mathcal{X}_T} \mathbb{E}[Y \mid x_T, x_{S\setminus T}] \cdot q(x_T \mid x_{S\setminus T}) && \text{(Interchanging order of sums)} \\
&= \sum_{x_T \in \mathcal{X}_T} F(x_T, x_{S\setminus T}) \cdot q(x_T \mid x_{S\setminus T}) && \text{(Definition of } F\text{)} \\
&= \sum_{x_T \in \mathcal{X}_T} \sum_{x_{\bar{S}} \in \mathcal{X}_{\bar{S}}} f(x_T, x_{S\setminus T}, x_{\bar{S}}) \cdot q(x_{\bar{S}} \mid x_T, x_{S\setminus T}) \cdot q(x_T \mid x_{S\setminus T}) && \text{(Definition of } F\text{)} \\
&= \mathbb{E}_{q(X_T, X_{\bar{S}}|X_{S\setminus T}=x_{S\setminus T})}\big[f(X_T, x_{S\setminus T}, X_{\bar{S}})\big]\big] && \text{(Definition of expectation)}
\end{aligned}$$

This result shows that deriving $F(x_{S\setminus T})$ from $F(x)$ or from $F(x_S)$ yields a consistent result. We conclude that our definition of F, which marginalizes out missing features with the conditional distribution induced by $q(X)$, provides the unique subset extension of f that is consistent with $q(X)$. ■

Appendix E. Conditional Distribution Approximations

We now describe how several approaches to removing features can be understood as approximations of marginalizing out missing features using their conditional distribution.

E.1 Separate models

Shapley Net Effects (Lipovetsky and Conklin, 2001), SPVIM (Williamson and Feng, 2020) and the original IME (Štrumbelj et al., 2009) require training separate models for each subset of features. As in the main text, we denote these models as $\{f_S : S \subseteq D\}$. Similarly, the univariate predictors approach requires training models with individual features, and feature ablation requires training models with individual features held out. These models are used to make predictions in the presence of missing features, and they can be represented by the subset function $F(x_S) = f_S(x_S)$.

Note that this F satisfies the necessary properties to be a subset extension of f_D (invariance to missing features and agreement with f_D in the presence of all features) despite the fact that its predictions with held out features do not explicitly reference f_D. However, under the assumption that each f_S optimizes the population risk, we show that each f_S can be understood in relation to f_D.

For a regression task where each model f_S is trained with MSE loss, the model that optimizes the population risk is the conditional expectation $f_S(x_S) = \mathbb{E}[Y \mid X_S = x_S]$. Similarly, if the prediction task is classification and the loss function is cross entropy (or another strictly proper scoring function, see Gneiting and Raftery, 2007) then the model that optimizes the population risk is the conditional probability function or Bayes classifier $f_S(x_S) = p(Y \mid X_S = x_S)$. In both cases, if each f_S for $S \subseteq D$ optimizes the population risk, then we observe the following relationship between F and f_D:

$$F(x_S) = f_S(x_S) = \mathbb{E}\big[f_D(X) \mid X_S = x_S\big].$$

This is precisely the approach of marginalizing out missing features from f_D using their conditional distribution.

E.2 Missingness during training

INVASE (Yoon et al., 2018) is based on a strategy of introducing missing features during model training (Appendix A.7). It trains a model where zeros (or potentially other values) take the place of removed features, so that the model can recognize these as missing values and make the best possible prediction given the available information.

We show here how this approach can be understood as an approximation of marginalizing out features with their conditional distribution. First, we note that replacing features with a default value is problematic if that value is observed in the dataset, because the model then faces ambiguity about whether the value is real or represents missingness. This issue can be resolved either by ensuring that the replacement value does not occur in the dataset, or by providing a mask vector $m \in \{0, 1\}^d$ indicating missingness as an additional input to the model.

We assume for simplicity that this binary vector is provided as an additional input, and we let $x \odot m$ (the element-wise product) represent feature masking and $f(x \odot m, m)$

denote the model's prediction. This can be viewed as a technique for parameterizing a subset function $F \in \mathfrak{F}$ because it ensures invariance to the features that are not selected. Specifically, we can write

$$F(x_S) = f\big(x \odot m(S), m(S)\big),$$

where $m(S)$ is a binary vector with ones corresponding to the members in S. If we let M denote a random binary mask variable representing feature subsets, then the loss for training this model is:

$$\min_f \mathbb{E}_{MXY}\Big[\ell\big(f(X \odot M, M), Y\big)\Big].$$

Then, if M is independent from (X, Y), we can decompose the loss as follows:

$$\begin{aligned}\mathbb{E}_{MXY}\Big[\ell\big(f(X \odot M, M), Y\big)\Big] &= \mathbb{E}_M \mathbb{E}_{XY}\Big[\ell\big(f(X \odot M, M), Y\big)\Big] \\ &= \sum_m p(m) \cdot \mathbb{E}_{XY}\Big[\ell\big(f(X \odot m, m), Y\big)\Big].\end{aligned}$$

For each value of m, we can regard $f(x \odot m, m)$ as a separate function on the specified subset of features $\{x_i : m_i = 1\}$. Then, for classification tasks using cross entropy loss, the objective is optimized by the function f^* such that

$$f^*(x \odot m, m) = p(Y \mid X_S = x_S),$$

where $S = \{i : m_i = 1\}$. A similar result holds other strictly proper scoring functions (Gneiting and Raftery, 2007) and for regression tasks trained with MSE loss (with the conditional probability replaced by the conditional expectation). In these cases, the result is equivalent to marginalizing out missing features according to their conditional distribution.

One issue with INVASE, noted by Jethani et al. (2021a), is that the mask variable M is not independent from the input vector X. Intuitively, this means that the selected features can communicate information about the held out features, which then inform the model's prediction about the label Y. The INVASE model therefore may not approximate $p(Y \mid X_S = x_S)$, potentially invalidating its information-theoretic interpretation in terms of KL divergence minimization (Section 8.3).

Nonetheless, it is possible to learn a model with missingness introduced at training time that approximates marginalizing out features using their conditional distribution. It suffices to sample masks during training independently from the model input, e.g., by sampling masks uniformly at random (Jethani et al., 2021a) or according to the distribution described in Appendix E.3.

E.3 Surrogate models

To remove features from an existing model f, we can train a surrogate model to match its predictions when features are held out. This technique, described by Frye et al. (2020) and implemented by L2X (Chen et al., 2018a) and REAL-X (Jethani et al., 2021a), can also approximate marginalizing out features using their conditional distribution.

To train such a model, we require a mechanism for removing features. We let f denote the original model and g the surrogate, and similar to the model trained with missingness (Appendix E.2), we can remove features using a mask variable $m \in \{0,1\}^d$. Frye et al. (2020) suggest replacing held out features using a value that does not occur in the training set, but we can also provide the mask variable as an input to the surrogate. We therefore represent the surrogate's predictions as $g(x \odot m, m)$, where g can be understood as a subset function $F \in \mathfrak{F}$ (with m defined as in Appendix E.2):

$$F(x_S) = g\big(x \odot m(S), m(S)\big).$$

The surrogate is then trained using the following objective function:

$$\min_g \; \mathbb{E}_{MX}\Big[\ell\big(g(X \odot M, M), f(X)\big)\Big]. \tag{58}$$

If M is sampled independently from X, then we can decompose the loss as follows:

$$\begin{aligned}\mathbb{E}_{MX}\Big[\ell\big(g(X \odot M, M), f(X)\big)\Big] &= \mathbb{E}_M \mathbb{E}_X\Big[\ell\big(g(X \odot M, M), f(X)\big)\Big] \\ &= \sum_m p(m) \mathbb{E}_X\Big[\ell\big(g(X \odot m, m), f(X)\big)\Big].\end{aligned}$$

For each value of m, we can regard $g(x \odot m, m)$ as a separate function on the specified subset of features $\{x_i : m_i = 1\}$. Then, for specific loss functions, we can reason about the optimal surrogate that matches the original model f most closely. For models that are compared using MSE loss, we point to a result from Covert et al. (2020):

$$\min_h \; \mathbb{E}_X\Big[\big(f(X) - h(X_S)\big)^2\Big] = \mathbb{E}_X\Big[\big(f(X) - \mathbb{E}[f(X) \mid X_S]\big)^2\Big].$$

This shows that to match f's predictions in the sense of MSE loss, the optimal surrogate function g^* is given by

$$g^*(x \odot m, m) = \mathbb{E}[f(X) \mid X_S = x_S], \tag{59}$$

where $S = \{i : m_i = 1\}$. This result justifies the approach used by Frye et al. (2020); however, while Frye et al. (2020) focus on MSE loss, we also find that a cross entropy loss can be used for classification tasks. When the original model f and g both output discrete probability distributions, their predictions can be compared through a soft cross entropy loss, which we define as follows:

$$H(a, b) = -\sum_j a_j \log b_j.$$

Now, for classifications models that are compared via cross entropy loss, we point to the following result from Covert et al. (2020):

$$\min_h \; \mathbb{E}_X\Big[H\big(f(X), h(X_S)\big)\Big] = \mathbb{E}_X\Big[H\big(f(X), \mathbb{E}[f(X) \mid X_S]\big)\Big].$$

This shows that the optimal surrogate model is once again given by marginalizing out the missing features using their conditional distribution, as in Eq. 59. Notably, these results require that X and M are sampled independently during training, which is a property that is satisfied by Frye et al. (2020) and Jethani et al. (2021a) (REAL-X), but not Chen et al. (2018a) (L2X).

The only detail left to specify is the distribution for M in Eq. 58. Any distribution that places mass on all $m \in \{0,1\}^d$ should suffice, at least in principle. Masks could be sampled uniformly at random, but for models with large numbers of features, this places nearly all the probability mass on subsets with approximately half of the features included. We therefore opt to use a distribution that samples the *subset size* uniformly at random. In our experiments, we sample masks m as follows:

1. Sample $k \in \{0, 1, \ldots, d\}$ uniformly at random.
2. Sample k indices $(i_1, \ldots, i_k)$ from $\{1, 2, \ldots, d\}$ at random and without replacement.
3. Set m with $m_i = \mathbb{1}\big(i \in \{i_1, \ldots, i_k\}\big)$.

Appendix F. Information-Theoretic Connections in Regression

Here, we describe probabilistic interpretations of each explanation method's underlying set function (Section 5) in the context of regression models rather than classification models (Section 8.3). We assume that the models are evaluated using MSE loss, and, as in the main text, we assume model optimality. The different set functions have the following interpretations.

- The set function $u_x(S) = F(x_S)$ quantifies the response variable's conditional expectation:

$$u_x(S) = \mathbb{E}[Y \mid X_S = x_S]. \tag{60}$$

  This set function lets us examine each feature's true relationship with the response variable.

- The set function $v_{xy}(S) = -\ell\big(F(x_S), y\big)$ quantifies the squared distance between the model output and the correct label:

$$v_{xy}(S) = -\Big(\mathbb{E}[Y \mid X_S = x_S] - y\Big)^2. \tag{61}$$

  Under the assumption that the response variable's conditional distribution is Gaussian, this represents the pointwise mutual information between x_S and y (up to factors that depend on S):

$$\begin{aligned}\mathrm{I}(y; x_S) = & -\log p(y) - \frac{1}{2}\log 2\pi - \frac{1}{2}\log \mathrm{Var}(Y \mid X_S = x_S) \\ & - \frac{1}{2}\underbrace{\Big(\mathbb{E}[Y \mid X_S = x_S] - y\Big)^2}_{v_{xy}(S)} / \mathrm{Var}(Y \mid X_S = x_S).\end{aligned} \tag{62}$$

- The set function $v_x(S) = -\mathbb{E}_{p(Y|X=x)}\big[\ell\big(F(x_S), Y\big)\big]$ quantifies the squared difference of the conditional expectation from the model output, up to a constant value:

$$v_x(S) = -\Big(\mathbb{E}[Y \mid X = x] - \mathbb{E}[Y \mid X_S = x_S]\Big)^2 + c. \tag{63}$$

Under the assumption that the response variable's distribution conditioned on $X = x$ and $X_S = x_S$ are both Gaussian, this quantity has a relationship with the negative KL divergence between $p(Y \mid X = x)$ and $p(Y \mid X_S = x_S)$:

$$\begin{aligned}& D_{\mathrm{KL}}\big(p(Y \mid X = x) \,\big|\big|\, p(Y \mid X_S = x_S)\big) \\ = & \frac{1}{2}\log\frac{\mathrm{Var}(Y \mid X_S = x_S)}{\mathrm{Var}(Y \mid X = x)} - \frac{1}{2} \\ & + \Big(\mathrm{Var}(Y \mid X = x) + \underbrace{\big(\mathbb{E}[Y \mid X = x] - \mathbb{E}[Y \mid X_S = x_S]\big)^2}_{v_x(S)}\Big) / \Big(2 \cdot \mathrm{Var}(Y \mid X_S = x_S)\Big).\end{aligned} \tag{64}$$

- The set function $v(S) = -\mathbb{E}_{XY}\Big[\ell\big(F(X_S), Y\big)\Big]$ quantifies the explained variance in the response variable Y, up to a constant value:

$$v(S) = \mathrm{Var}(Y) - \mathbb{E}[\mathrm{Var}(Y \mid X_S)] + c. \tag{65}$$

Using the entropy maximizing property of the Gaussian distribution (Cover and Thomas, 2012), we see that the explained variance has the following relationship with the mutual information between X_S and the response variable Y:

$$\begin{aligned}\mathrm{I}(Y; X_S) &= H(Y) - \mathbb{E}\big[H(Y \mid X_S)\big] \\ &\geq H(Y) - \frac{1}{2}\mathbb{E}\Big[\log\big(2\pi e \cdot \mathrm{Var}(Y \mid X_S)\big)\Big] \\ &\geq H(Y) - \frac{1}{2}\log 2\pi e - \frac{1}{2}\log \underbrace{\mathbb{E}\big[\mathrm{Var}(Y \mid X_S)\big]}_{v(S)}.\end{aligned} \tag{66}$$

Equality is achieved in the first bound if the distribution $p(Y \mid X_S = x_S)$ is Gaussian. The second bound is due to Jensen's inequality.

- The set function $w_x(S) = -\ell\big(F(x_S), f(x)\big)$ quantifies the squared difference between the model output and the expected model output when conditioned on a subset of features:

$$w_x(S) = -\Big(f(x) - \mathbb{E}[f(X) \mid X_S = x_S]\Big)^2. \tag{67}$$

If we assume that $p(f(X) \mid X_S)$ is a Gaussian distribution, then we can view this as a component in the KL divergence between $p(f(X) \mid X)$ (a deterministic distribution) and $p(f(X) \mid X_S = x_S)$:

$$\begin{aligned}
&D_{\mathrm{KL}}\big(p(f(X) \mid X = x) \,\big|\big|\, p(f(X) \mid X_S = x_S)\big) \\
&= \lim_{\sigma \to 0} \frac{1}{2} \log \frac{\mathrm{Var}(f(X) \mid X_S = x_S)}{\sigma^2} - \frac{1}{2} \\
&\qquad + \Big(\sigma^2 + \underbrace{\big(f(x) - \mathbb{E}[f(X) \mid X_S = x_S]\big)^2}_{w_X(S)}\Big) \Big/ \Big(2\mathrm{Var}(f(X) \mid X_S = x_S)\Big) \\
&= \infty.
\end{aligned} \tag{68}$$

This result is somewhat contrived, however, because the KL divergence is ultimately infinite.

- The set function $w(S) = -\mathbb{E}_X\Big[\ell\big(F(X_S), f(X)\big)\Big]$ quantifies the explained variance in the model output, up to a constant value:

$$w(S) = \mathrm{Var}\Big(f(X)\Big) - \mathbb{E}\Big[\mathrm{Var}\big(f(X) \mid X_S\big)\Big] + c. \tag{69}$$

This is the variance decomposition result presented for Shapley Effects (Owen, 2014). Using the same entropy maximizing property of the Gaussian distribution, we can also see that the explained variance is related to the mutual information with the model output, which can be viewed as a random variable $f(X)$:

$$\begin{aligned}
\mathrm{I}\big(f(X); X_S\big) &= H\big(f(X)\big) - \mathbb{E}\big[H(f(X) \mid X_S)\big] \\
&\geq H\big(f(X)\big) - \frac{1}{2}\mathbb{E}\Big[\log\big(2\pi e \cdot \mathrm{Var}(f(X) \mid X_S)\big)\Big] \\
&\geq H\big(f(X)\big) - \frac{1}{2}\log 2\pi e - \frac{1}{2}\log \underbrace{\mathbb{E}\Big[\mathrm{Var}\big(f(X) \mid X_S\big)\Big]}_{w(S)}
\end{aligned} \tag{70}$$

Equality is achieved in the first bound if $f(X)$ has a Gaussian distribution when conditioned on $X_S = x_S$. The second bound is due to Jensen's inequality.

These results are analogous to those from the classification case, and they show that each explanation method's set function has an information-theoretic interpretation even in the context of regression tasks. However, the regression case requires stronger assumptions about the data distribution to yield these information-theoretic links (i.e., Gaussian distributions). To avoid strong distributional assumptions, it is more conservative to interpret these quantities in terms of Euclidean distances (Eqs. 61, 63, 67) and conditional variances (Eqs. 65, 69).

Appendix G. Experiment Details

This section provides additional details about models, datasets and hyperparameters, as well as some additional results.

G.1 Hyperparameters

The original models used for each dataset are:

- For the census income dataset, we trained a LightGBM model with a maximum of 10 leaves per tree and a learning rate of 0.05 (Ke et al., 2017).
- For MNIST, we trained a 14-layer CNN consisting of convolutional layers with kernel size 3, max pooling layers, and ELU activations (Clevert et al., 2015). The output was produced by flattening the convolutional features and applying two fully connected layers, similar to the VGG architecture (Simonyan and Zisserman, 2014). We trained the model with Adam using a learning rate of 10^{-3} (Kingma and Ba, 2014).
- For the BRCA dataset, we trained a ℓ_1 regularized logistic regression model and selected the regularization parameter using a validation set.

For the BRCA dataset, to avoid overfitting, we also randomly selected a subset of 100 genes to analyze out of 17,814 total. To ensure that a sufficient number BRCA-associated genes were selected, we tried 10 random seeds for the gene selection step and selected the seed whose 100 genes achieved the best performance (displayed in Table 9). A small portion of missing expression values were imputed with their mean, and the data was centered and normalized prior to fitting the regularized logistic regression model.

When generating explanations using feature removal approaches that required sampling multiple values for the missing features (marginalizing with uniform, product of marginals, or joint marginal), we used 512 samples for the census income and MNIST datasets and 372 for the BRCA dataset (the size of the train split).

As described in the main text (Section 10) and in Appendix E.3, we trained surrogate models to represent marginalizing out features according to the conditional distribution. Our surrogate models were trained as follows:

- For the census income data, the surrogate was a MLP with a masking layer and four hidden layers of size 128 followed by ELU activations. During training, the mask variable was sampled according to the procedure described in Appendix E.3. Our masking layer replaced missing values with -1 (a value that did not occur in the

Table 9: List of genes analyzed.

Genes 1-17	Genes 18-34	Genes 35-51	Genes 52-68	Genes 69-85	Genes 86-100
OSTbeta	NBR2	TSHR	HPS4	GRINA	C20orf111
STATH	CCDC64	C7	ZFPM1	YTHDF3	OMA1
MAPK10	NUP210	CRYBB2	OAS2	TMCC1	NCAPH2
PLEKHG5	HEMGN	PPAPDC3	TUBA1C	UBE1DC1	GPX2
ERO1L	SLC25A3	TXNL4B	OR8K5	C6orf15	BPY2C
ZNF711	LEF1	CHST9	THSD3	PDE6A	ZNF324
ZNF385	MVD	HACE1	ATP6V0C	PEO1	CDC27
OR52E8	OTUD3	AYTL1	RAB22A	TMEM52	CCNB2
SLC5A11	KIAA1949	PRSS35	AP1B1	PARP1	CNOT7
P4HA3	SLC44A3	ZNF408	CTAGE6	GSS	BIRC3
LHFPL4	ZNF775	DDC	C6orf26	RDH11	GAL3ST3
MGC33657	THY1	CSTL1	ESR1	STXBP1	PLEKHM1
CAPZB	DYNC1I2	OR2F1	UPK3B	ACLY	SPOCD1
RBM15B	CYP1A1	C12orf50	ROBO4	TMSB10	PENK
C1orf176	SPTA1	SH3YL1	TMEFF1	TUBB	TAS2R9
KLF3	CLEC4M	SNUPN	KIAA1279	LIPK	
OLFM4	RXFP3	COL25A1	ZFP36L1	HRC	

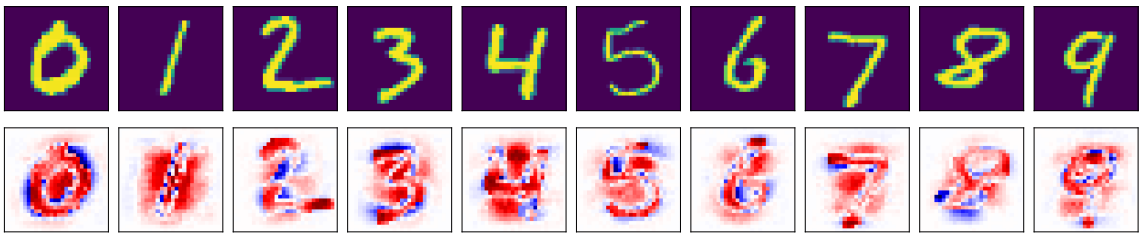

Figure 13: MNIST prediction loss explanations using LossSHAP (feature removal with the conditional distribution and summary with the Shapley value).

dataset) and also appended the mask as an additional set of features, which improved its ability to match the original model's predictions.

- For MNIST, the surrogate was a CNN with an identical architecture to the original model (see above) except for a masking layer at the input. The masking layer replaced missing values with zeros and appended the mask along the channels dimension.

- For the BRCA data, the surrogate was an MLP with two hidden layers of size 64 followed by ELU activations. The masking layer replaced missing values with their mean and appended the mask as an additional set of features.

G.2 Additional results

We present two supplementary results for the experiments described in Section 10. Figure 13 shows more examples of MNIST explanations using the combination of the conditional distribution and Shapley value (i.e., LossSHAP). These explanations consistently highlight important pixels within the digit as well as empty regions that distinguish the digit from other possible classes (e.g., see 3, 4, 9).

Figure 14 quantifies the similarity between dataset loss explanations for the BRCA dataset using their correlation (top) and Spearman rank correlation (bottom). We see patterns in these explanations that are similar to those seen with the census dataset: explanations generated using Shapley values are relatively similar to those that remove individual features or use Banzhaf values, while the include individual technique tends to differ most from the others. Because these plots visualize correlation rather than Euclidean distance, we can also see that Banzhaf value explanations are correlated with those that use the mean when included technique, which was predicted by our theory (Section 7.2). Interestingly, the Shapley value explanations are more strongly correlated across different removal strategies than they are to explanations that use other summarization strategies but that remove features in the same way (see bottom row of Figure 14).

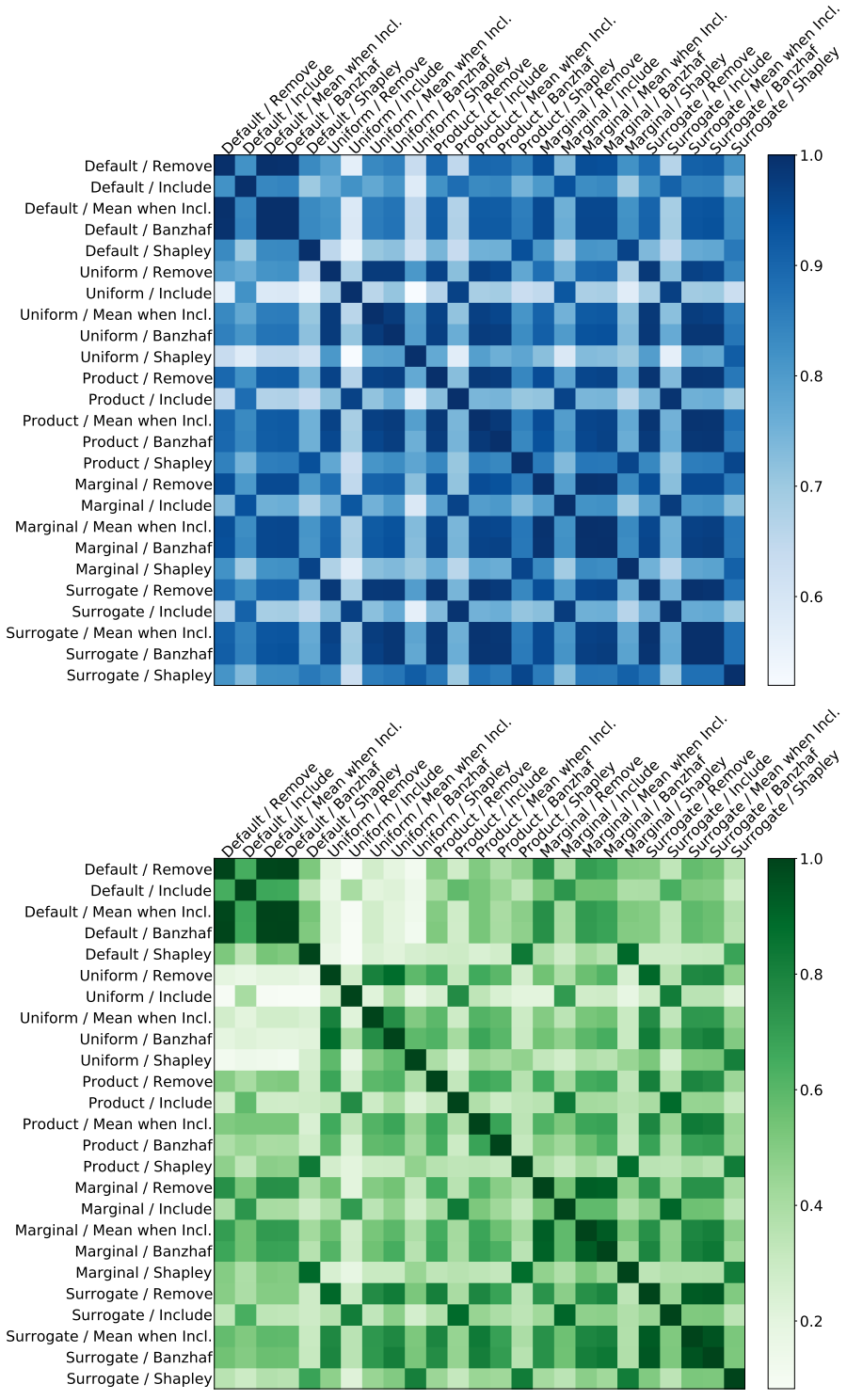

Figure 14: Mean correlation (top) and Spearman rank correlation (bottom) between different explanation methods on the BRCA dataset.

References

Kjersti Aas, Martin Jullum, and Anders Løland. Explaining individual predictions when features are dependent: More accurate approximations to Shapley values. *arXiv preprint arXiv:1903.10464*, 2019.

Kjersti Aas, Thomas Nagler, Martin Jullum, and Anders Løland. Explaining predictive models using Shapley values and non-parametric vine copulas. *arXiv preprint arXiv:2102.06416*, 2021.

Amina Adadi and Mohammed Berrada. Peeking inside the black-box: A survey on explainable artificial intelligence (XAI). *IEEE Access*, 6:52138–52160, 2018.

Julius Adebayo, Justin Gilmer, Michael Muelly, Ian Goodfellow, Moritz Hardt, and Been Kim. Sanity checks for saliency maps. *arXiv preprint arXiv:1810.03292*, 2018.

Chirag Agarwal and Anh Nguyen. Explaining an image classifier's decisions using generative models. *arXiv preprint arXiv:1910.04256*, 2019.

Marco Ancona, Enea Ceolini, Cengiz Öztireli, and Markus Gross. Towards better understanding of gradient-based attribution methods for deep neural networks. *arXiv preprint arXiv:1711.06104*, 2017.

Robert JJ Aumann. Economic applications of the Shapley value. In *Game-theoretic Methods in General Equilibrium Analysis*, pages 121–133. Springer, 1994.

Sebastian Bach, Alexander Binder, Grégoire Montavon, Frederick Klauschen, Klaus-Robert Müller, and Wojciech Samek. On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PloS One*, 10(7):e0130140, 2015.

John F Banzhaf. Weighted voting doesn't work: a mathematical analysis. *Rutgers Law Review*, 19:317, 1964.

Mohamed Belghazi, Maxime Oquab, and David Lopez-Paz. Learning about an exponential amount of conditional distributions. In *Advances in Neural Information Processing Systems*, pages 13703–13714, 2019.

Ashton C Berger, Anil Korkut, Rupa S Kanchi, Apurva M Hegde, Walter Lenoir, Wenbin Liu, Yuexin Liu, Huihui Fan, Hui Shen, Visweswaran Ravikumar, et al. A comprehensive pan-cancer molecular study of gynecologic and breast cancers. *Cancer Cell*, 33(4):690–705, 2018.

Umang Bhatt, Alice Xiang, Shubham Sharma, Adrian Weller, Ankur Taly, Yunhan Jia, Joydeep Ghosh, Ruchir Puri, José MF Moura, and Peter Eckersley. Explainable machine learning in deployment. In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, pages 648–657, 2020.

Stephen Boyd, Stephen P Boyd, and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

Leo Breiman. Random forests. *Machine Learning*, 45(1):5–32, 2001.

Chunshui Cao, Xianming Liu, Yi Yang, Yinan Yu, Jiang Wang, Zilei Wang, Yongzhen Huang, Liang Wang, Chang Huang, Wei Xu, et al. Look and think twice: Capturing top-down visual attention with feedback convolutional neural networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2956–2964, 2015.

Chun-Hao Chang, Elliot Creager, Anna Goldenberg, and David Duvenaud. Explaining image classifiers by counterfactual generation. *arXiv preprint arXiv:1807.08024*, 2018.

A Charnes, B Golany, M Keane, and J Rousseau. Extremal principle solutions of games in characteristic function form: core, Chebychev and Shapley value generalizations. In *Econometrics of Planning and Efficiency*, pages 123–133. Springer, 1988.

Hugh Chen, Joseph D Janizek, Scott Lundberg, and Su-In Lee. True to the model or true to the data? *arXiv preprint arXiv:2006.16234*, 2020.

Jianbo Chen, Le Song, Martin J Wainwright, and Michael I Jordan. Learning to explain: An information-theoretic perspective on model interpretation. *arXiv preprint arXiv:1802.07814*, 2018a.

Jianbo Chen, Le Song, Martin J Wainwright, and Michael I Jordan. L-Shapley and C-Shapley: Efficient model interpretation for structured data. *arXiv preprint arXiv:1808.02610*, 2018b.

Djork-Arné Clevert, Thomas Unterthiner, and Sepp Hochreiter. Fast and accurate deep network learning by exponential linear units (ELUs). *arXiv preprint arXiv:1511.07289*, 2015.

Thomas M Cover and Joy A Thomas. *Elements of Information Theory*. John Wiley & Sons, 2012.

Ian Covert and Su-In Lee. Improving KernelSHAP: Practical Shapley value estimation using linear regression. In *International Conference on Artificial Intelligence and Statistics*, pages 3457–3465. PMLR, 2021.

Ian Covert, Scott Lundberg, and Su-In Lee. Understanding global feature contributions with additive importance measures. *arXiv preprint arXiv:2004.00668*, 2020.

Piotr Dabkowski and Yarin Gal. Real time image saliency for black box classifiers. In *Advances in Neural Information Processing Systems*, pages 6967–6976, 2017.

Anupam Datta, Shayak Sen, and Yair Zick. Algorithmic transparency via quantitative input influence: Theory and experiments with learning systems. In *2016 IEEE Symposium on Security and Privacy (SP)*, pages 598–617. IEEE, 2016.

Guoli Ding, Robert F Lax, Jianhua Chen, and Peter P Chen. Formulas for approximating pseudo-boolean random variables. *Discrete Applied Mathematics*, 156(10):1581–1597, 2008.

Guoli Ding, Robert F Lax, Jianhua Chen, Peter P Chen, and Brian D Marx. Transforms of pseudo-boolean random variables. *Discrete Applied Mathematics*, 158(1):13–24, 2010.

Laura Douglas, Iliyan Zarov, Konstantinos Gourgoulias, Chris Lucas, Chris Hart, Adam Baker, Maneesh Sahani, Yura Perov, and Saurabh Johri. A universal marginalizer for amortized inference in generative models. *arXiv preprint arXiv:1711.00695*, 2017.

Pradeep Dubey and Lloyd S Shapley. Mathematical properties of the Banzhaf power index. *Mathematics of Operations Research*, 4(2):99–131, 1979.

Kai Epstude and Neal J Roese. The functional theory of counterfactual thinking. *Personality and social psychology review*, 12(2):168–192, 2008.

Gabriel Erion, Joseph D Janizek, Pascal Sturmfels, Scott Lundberg, and Su-In Lee. Learning explainable models using attribution priors. *arXiv preprint arXiv:1906.10670*, 2019.

Lijie Fan, Shengjia Zhao, and Stefano Ermon. Adversarial localization network. In *Learning with limited labeled data: weak supervision and beyond, NIPS Workshop*, 2017.

Robert M Fano. Transmission of information: A statistical theory of communications. *American Journal of Physics*, 29(11):793–794, 1961.

Ruth Fong, Mandela Patrick, and Andrea Vedaldi. Understanding deep networks via extremal perturbations and smooth masks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2950–2958, 2019.

Ruth C Fong and Andrea Vedaldi. Interpretable explanations of black boxes by meaningful perturbation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3429–3437, 2017.

Christopher Frye, Ilya Feige, and Colin Rowat. Asymmetric Shapley values: incorporating causal knowledge into model-agnostic explainability. *arXiv preprint arXiv:1910.06358*, 2019.

Christopher Frye, Damien de Mijolla, Laurence Cowton, Megan Stanley, and Ilya Feige. Shapley-based explainability on the data manifold. *arXiv preprint arXiv:2006.01272*, 2020.

Tilmann Gneiting and Adrian E Raftery. Strictly proper scoring rules, prediction, and estimation. *Journal of the American Statistical Association*, 102(477):359–378, 2007.

Michel Grabisch, Jean-Luc Marichal, and Marc Roubens. Equivalent representations of set functions. *Mathematics of Operations Research*, 25(2):157–178, 2000.

Riccardo Guidotti, Anna Monreale, Salvatore Ruggieri, Franco Turini, Fosca Giannotti, and Dino Pedreschi. A survey of methods for explaining black box models. *ACM Computing Surveys (CSUR)*, 51(5):1–42, 2018.

Isabelle Guyon and André Elisseeff. An introduction to variable and feature selection. *Journal of Machine Learning Research*, 3(Mar):1157–1182, 2003.

Peter L Hammer and Ron Holzman. Approximations of pseudo-boolean functions; applications to game theory. *Zeitschrift für Operations Research*, 36(1):3–21, 1992.

Tom Heskes, Evi Sijben, Ioan Gabriel Bucur, and Tom Claassen. Causal Shapley values: Exploiting causal knowledge to explain individual predictions of complex models. *arXiv preprint arXiv:2011.01625*, 2020.

Denis J Hilton. Conversational processes and causal explanation. *Psychological Bulletin*, 107(1):65, 1990.

Giles Hooker and Lucas Mentch. Please stop permuting features: An explanation and alternatives. *arXiv preprint arXiv:1905.03151*, 2019.

Sara Hooker, Dumitru Erhan, Pieter-Jan Kindermans, and Been Kim. A benchmark for interpretability methods in deep neural networks. *arXiv preprint arXiv:1806.10758*, 2018.

Oleg Ivanov, Michael Figurnov, and Dmitry Vetrov. Variational autoencoder with arbitrary conditioning. *arXiv preprint arXiv:1806.02382*, 2018.

Dominik Janzing, Lenon Minorics, and Patrick Blöbaum. Feature relevance quantification in explainable AI: A causality problem. *arXiv preprint arXiv:1910.13413*, 2019.

Jos Jaspars, Miles Hewstone, and Frank D Fincham. Attribution theory and research: The state of the art. *Attribution theory and research: Conceptual, developmental and social dimensions*, pages 3–36, 1983.

Neil Jethani, Mukund Sudarshan, Yindalon Aphinyanaphongs, and Rajesh Ranganath. Have we learned to explain?: How interpretability methods can learn to encode predictions in their interpretations. In *International Conference on Artificial Intelligence and Statistics*, pages 1459–1467. PMLR, 2021a.

Neil Jethani, Mukund Sudarshan, Ian Covert, Su-In Lee, and Rajesh Ranganath. FastSHAP: Real-time Shapley value estimation. *arXiv e-prints*, pages arXiv–2107, 2021b.

Ian T Jolliffe. Principal components in regression analysis. In *Principal Component Analysis*, pages 129–155. Springer, 1986.

Daniel Kahneman and Dale T Miller. Norm theory: Comparing reality to its alternatives. *Psychological Review*, 93(2):136, 1986.

Daniel Kahneman and Amos Tversky. The simulation heuristic. *Judgment Under Uncertainty: Heuristics and Biases*, pages 201–208, 1982.

Harmanpreet Kaur, Harsha Nori, Samuel Jenkins, Rich Caruana, Hanna Wallach, and Jennifer Wortman Vaughan. Interpreting interpretability: Understanding data scientists' use of interpretability tools for machine learning. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, pages 1–14, 2020.

Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. LightGBM: A highly efficient gradient boosting decision tree. In *Advances in Neural Information Processing Systems*, pages 3146–3154, 2017.

Areum Daseul Kim, Rui Zhang, Xia Han, Kyoung Ah Kang, Mei Jing Piao, Young Hee Maeng, Weon Young Chang, and Jin Won Hyun. Involvement of glutathione and glutathione metabolizing enzymes in human colorectal cancer cell lines and tissues. *Molecular Medicine Reports*, 12(3):4314–4319, 2015.

Pieter-Jan Kindermans, Kristof T Schütt, Maximilian Alber, Klaus-Robert Müller, Dumitru Erhan, Been Kim, and Sven Dähne. Learning how to explain neural networks: PatternNet and PatternAttribution. *arXiv preprint arXiv:1705.05598*, 2017.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Andrey N Kolmogorov. Foundations of the theory of probability, 1950.

Georgios Koutalellis, Konstantinos Stravodimos, Margaritis Avgeris, Konstantinos Mavridis, Andreas Scorilas, Andreas Lazaris, and Constantinos Constantinides. L-dopa decarboxylase (DDC) gene expression is related to outcome in patients with prostate cancer. *BJU International*, 110(6b):E267–E273, 2012.

I Elizabeth Kumar, Suresh Venkatasubramanian, Carlos Scheidegger, and Sorelle Friedler. Problems with Shapley-value-based explanations as feature importance measures. *arXiv preprint arXiv:2002.11097*, 2020.

Michael H Kutner, Christopher J Nachtsheim, John Neter, William Li, et al. *Applied linear statistical models*, volume 5. McGraw-Hill Irwin New York, 2005.

Isaac Lage, Emily Chen, Jeffrey He, Menaka Narayanan, Been Kim, Samuel J Gershman, and Finale Doshi-Velez. Human evaluation of models built for interpretability. In *Proceedings of the AAAI Conference on Human Computation and Crowdsourcing*, volume 7, pages 59–67, 2019.

Pierre-Simon Laplace. Mémoire sur les probabilités. *Mémoires de l'Académie Royale des sciences de Paris*, 1778:227–332, 1781.

Yann LeCun, Corinna Cortes, and CJ Burges. MNIST handwritten digit database. AT&T labs, 2010.

Jing Lei, Max G'Sell, Alessandro Rinaldo, Ryan J Tibshirani, and Larry Wasserman. Distribution-free predictive inference for regression. *Journal of the American Statistical Association*, 113(523):1094–1111, 2018.

Moshe Lichman et al. UCI machine learning repository, 2013.

Stan Lipovetsky and Michael Conklin. Analysis of regression in game theory approach. *Applied Stochastic Models in Business and Industry*, 17(4):319–330, 2001.

Zachary C Lipton. The mythos of model interpretability. *Queue*, 16(3):31–57, 2018.

Tania Lombrozo. Simplicity and probability in causal explanation. *Cognitive Psychology*, 55(3):232–257, 2007.

Scott M Lundberg and Su-In Lee. A unified approach to interpreting model predictions. In *Advances in Neural Information Processing Systems*, pages 4765–4774, 2017.

Scott M. Lundberg, Gabriel Erion, Hugh Chen, Alex DeGrave, Jordan M. Prutkin, Bala Nair, Ronit Katz, Jonathan Himmelfarb, Nisha Bansal, and Su-In Lee. From local explanations to global understanding with explainable AI for trees. *Nature Machine Intelligence*, 2(1):2522–5839, 2020.

John Leslie Mackie. *The cement of the universe: A study of causation*. Oxford: Clarendon Press, 1974.

Chris J Maddison, Andriy Mnih, and Yee Whye Teh. The concrete distribution: A continuous relaxation of discrete random variables. *arXiv preprint arXiv:1611.00712*, 2016.

Jean-Luc Marichal and Pierre Mathonet. Weighted Banzhaf power and interaction indexes through weighted approximations of games. *European Journal of Operational Research*, 211(2):352–358, 2011.

Masayoshi Mase, Art B Owen, and Benjamin Seiler. Explaining black box decisions by Shapley cohort refinement. *arXiv preprint arXiv:1911.00467*, 2019.

Luke Merrick and Ankur Taly. The explanation game: Explaining machine learning models with cooperative game theory. *arXiv preprint arXiv:1909.08128*, 2019.

John Stuart Mill. *A system of logic, ratiocinative and inductive: Being a connected view of the principles of evidence and the methods of scientific investigation*. Harper, 1884.

Tim Miller. Explanation in artificial intelligence: Insights from the social sciences. *Artificial Intelligence*, 267:1–38, 2019.

Tim Miller, Piers Howe, and Liz Sonenberg. Explainable AI: Beware of inmates running the asylum or: How I learnt to stop worrying and love the social and behavioural sciences. *arXiv preprint arXiv:1712.00547*, 2017.

Dov Monderer, Dov Samet, et al. Variations on the Shapley value. *Handbook of Game Theory*, 3:2055–2076, 2002.

Grégoire Montavon, Sebastian Lapuschkin, Alexander Binder, Wojciech Samek, and Klaus-Robert Müller. Explaining nonlinear classification decisions with deep taylor decomposition. *Pattern Recognition*, 65:211–222, 2017.

Yadati Narahari. *Game Theory and Mechanism Design*, volume 4. World Scientific, 2014.

Silje H Nordgard, Fredrik E Johansen, Grethe IG Alnæs, Elmar Bucher, Ann-Christine Syvänen, Bjørn Naume, Anne-Lise Børresen-Dale, and Vessela N Kristensen. Genome-wide analysis identifies 16q deletion associated with survival, molecular subtypes, mRNA expression, and germline haplotypes in breast cancer patients. *Genes, Chromosomes and Cancer*, 47(8):680–696, 2008.

Donald A Norman. Some observations on mental models. *Mental Models*, 7(112):7–14, 1983.

Andrzej S Nowak. On an axiomatization of the Banzhaf value without the additivity axiom. *International Journal of Game Theory*, 26(1):137–141, 1997.

Art B Owen. Sobol' indices and Shapley value. *SIAM/ASA Journal on Uncertainty Quantification*, 2(1):245–251, 2014.

Judea Pearl. *Causality*. Cambridge University Press, 2009.

Leon Petrosjan and Georges Zaccour. Time-consistent Shapley value allocation of pollution cost reduction. *Journal of Economic Dynamics and Control*, 27(3):381–398, 2003.

Vitali Petsiuk, Abir Das, and Kate Saenko. RISE: Randomized input sampling for explanation of black-box models. *arXiv preprint arXiv:1806.07421*, 2018.

Stephen J Read and Amy Marcus-Newhall. Explanatory coherence in social explanations: A parallel distributed processing account. *Journal of Personality and Social Psychology*, 65(3):429, 1993.

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. "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*, pages 1135–1144, 2016.

Dan R Robinson, Yi-Mi Wu, Pankaj Vats, Fengyun Su, Robert J Lonigro, Xuhong Cao, Shanker Kalyana-Sundaram, Rui Wang, Yu Ning, Lynda Hodges, et al. Activating ESR1 mutations in hormone-resistant metastatic breast cancer. *Nature Genetics*, 45(12):1446, 2013.

Marko Robnik-Šikonja and Igor Kononenko. Explaining classifications for individual instances. *IEEE Transactions on Knowledge and Data Engineering*, 20(5):589–600, 2008.

Cynthia Rudin. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature Machine Intelligence*, 1(5):206–215, 2019.

Karl Schulz, Leon Sixt, Federico Tombari, and Tim Landgraf. Restricting the flow: Information bottlenecks for attribution. *arXiv preprint arXiv:2001.00396*, 2020.

Patrick Schwab and Walter Karlen. CXPlain: Causal explanations for model interpretation under uncertainty. In *Advances in Neural Information Processing Systems*, pages 10220–10230, 2019.

Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-CAM: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 618–626, 2017.

Lloyd S Shapley. A value for n-person games. *Contributions to the Theory of Games*, 2(28): 307–317, 1953.

Anthony F Shorrocks. Decomposition procedures for distributional analysis: a unified framework based on the Shapley value. Technical report, Mimeo, University of Essex, 1999.

Avanti Shrikumar, Peyton Greenside, Anna Shcherbina, and Anshul Kundaje. Not just a black box: Learning important features through propagating activation differences. *arXiv preprint arXiv:1605.01713*, 2016.

Emman Shubbar, Anikó Kovács, Shahin Hajizadeh, Toshima Z Parris, Szilárd Nemes, Katrin Gunnarsdóttir, Zakaria Einbeigi, Per Karlsson, and Khalil Helou. Elevated cyclin b2 expression in invasive breast carcinoma is associated with unfavorable clinical outcome. *BMC Cancer*, 13(1):1, 2013.

Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. *arXiv preprint arXiv:1312.6034*, 2013.

Dylan Slack, Sophie Hilgard, Emily Jia, Sameer Singh, and Himabindu Lakkaraju. Fooling LIME and SHAP: Adversarial attacks on post hoc explanation methods. In *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*, pages 180–186, 2020.

Daniel Smilkov, Nikhil Thorat, Been Kim, Fernanda Viégas, and Martin Wattenberg. Smoothgrad: removing noise by adding noise. *arXiv preprint arXiv:1706.03825*, 2017.

Eunhye Song, Barry L Nelson, and Jeremy Staum. Shapley effects for global sensitivity analysis: Theory and computation. *SIAM/ASA Journal on Uncertainty Quantification*, 4(1):1060–1083, 2016.

Jost Tobias Springenberg, Alexey Dosovitskiy, Thomas Brox, and Martin Riedmiller. Striving for simplicity: The all convolutional net. *arXiv preprint arXiv:1412.6806*, 2014.

Carolin Strobl, Anne-Laure Boulesteix, Thomas Kneib, Thomas Augustin, and Achim Zeileis. Conditional variable importance for random forests. *BMC Bioinformatics*, 9 (1):307, 2008.

Erik Štrumbelj and Igor Kononenko. An efficient explanation of individual classifications using game theory. *Journal of Machine Learning Research*, 11:1–18, 2010.

Erik Štrumbelj, Igor Kononenko, and M Robnik Šikonja. Explaining instance classifications with interactions of subsets of feature values. *Data & Knowledge Engineering*, 68(10): 886–904, 2009.

Mukund Sundararajan and Amir Najmi. The many Shapley values for model explanation. *arXiv preprint arXiv:1908.08474*, 2019.

Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 3319–3328. JMLR. org, 2017.

Saeid Asgari Taghanaki, Mohammad Havaei, Tess Berthier, Francis Dutil, Lisa Di Jorio, Ghassan Hamarneh, and Yoshua Bengio. Infomask: Masked variational latent representation to localize chest disease. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 739–747. Springer, 2019.

Nikola Tarashev, Kostas Tsatsaronis, and Claudio Borio. Risk attribution using the Shapley value: Methodology and policy applications. *Review of Finance*, 20(3):1189–1213, 2016.

Paul Thagard. Explanatory coherence. *Behavioral and Brain Sciences*, 12(3):435–502, 1989.

Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, 58(1):267–288, 1996.

Sahil Verma, John Dickerson, and Keegan Hines. Counterfactual explanations for machine learning: A review. *arXiv preprint arXiv:2010.10596*, 2020.

Sandra Wachter, Brent Mittelstadt, and Chris Russell. Counterfactual explanations without opening the black box: Automated decisions and the GDPR. *Harvard Journal of Law & Technology*, 31:841, 2017.

Jiaxuan Wang, Jenna Wiens, and Scott Lundberg. Shapley flow: A graph-based approach to interpreting model predictions. In *International Conference on Artificial Intelligence and Statistics*, pages 721–729. PMLR, 2021.

Robert J Weber. Probabilistic values for games. *The Shapley Value. Essays in Honor of Lloyd S. Shapley*, pages 101–119, 1988.

Brian Williamson and Jean Feng. Efficient nonparametric statistical inference on population feature importance using Shapley values. In *International Conference on Machine Learning*, pages 10282–10291. PMLR, 2020.

Shawn Xu, Subhashini Venugopalan, and Mukund Sundararajan. Attribution in scale and space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9680–9689, 2020.

Jinsung Yoon, James Jordon, and Mihaela van der Schaar. INVASE: Instance-wise variable selection using neural networks. In *International Conference on Learning Representations*, 2018.

H Peyton Young. Monotonic solutions of cooperative games. *International Journal of Game Theory*, 14(2):65–72, 1985.

Jiahui Yu, Zhe Lin, Jimei Yang, Xiaohui Shen, Xin Lu, and Thomas S Huang. Generative image inpainting with contextual attention. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5505–5514, 2018.

Matthew D Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *European Conference on Computer Vision*, pages 818–833. Springer, 2014.

Jianming Zhang, Sarah Adel Bargal, Zhe Lin, Jonathan Brandt, Xiaohui Shen, and Stan Sclaroff. Top-down neural attention by excitation backprop. *International Journal of Computer Vision*, 126(10):1084–1102, 2018.

Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Object detectors emerge in deep scene CNNs. *arXiv preprint arXiv:1412.6856*, 2014.

Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2921–2929, 2016.

Luisa M Zintgraf, Taco S Cohen, Tameem Adel, and Max Welling. Visualizing deep neural network decisions: Prediction difference analysis. *arXiv preprint arXiv:1702.04595*, 2017.