Algorithms to estimate Shapley value feature attributions

Hugh Chen, Ian C. Covert, Scott M. Lundberg & Su-In Lee

Feature attributions based on the Shapley value are popular for explaining machine learning models. However, their estimation is complex from both theoretical and computational standpoints. We disentangle this complexity into two main factors: the approach to removing feature information and the tractable estimation strategy. These two factors provide a natural lens through which we can better understand and compare 24 distinct algorithms. Based on the various feature-removal approaches, we describe the multiple types of Shapley value feature attributions and the methods to calculate each one. Then, based on the tractable estimation strategies, we characterize two distinct families of approaches: model-agnostic and model-specific approximations. For the model-agnostic approximations, we benchmark a wide class of estimation approaches and tie them to alternative yet equivalent characterizations of the Shapley value. For the model-specific approximations, we clarify the assumptions crucial to each method’s tractability for linear, tree and deep models. Finally, we identify gaps in the literature and promising future research directions.

Machine learning models are increasingly prevalent because they have matched or surpassed human performance in many applications, including Go, poker, StarCraft, protein folding, language translation and so on. One critical component in their success is flexibility, or expressive power, which has been facilitated by more complex models and improved hardware. Unfortunately, their flexibility also makes models opaque, or challenging for humans to understand. Combined with the tendency of machine learning to rely on shortcuts (that is, unintended learning strategies that fail to generalize to unseen data), there is a growing demand for model interpretability. This demand is reflected in increasing calls for explanations by diverse regulatory bodies, such as the General Data Protection Regulation’s ‘right to explanation’ and the Equal Credit Opportunity Act’s adverse action notices.

There are many possible ways to explain machine learning models (for example, counterfactuals, exemplars, surrogate models and so on), but one extremely popular approach is ‘local feature attribution’. In this approach, individual predictions are explained by an attribution vector, \( \phi \in \mathbb{R}^d \), where \( d \) is the number of features used by the model. One prominent example is LIME (local interpretable model-agnostic explanations), which fits a simple model to capture the model’s behaviour in the neighbourhood of a single sample; when a linear model is used, the coefficients serve as attribution scores for each feature. In addition to LIME, many other methods exist to compute local feature attributions. One popular class of approaches comprises the additive feature attribution methods, which are those whose attributions sum to a specific value, such as the model’s prediction. To unify the class of additive feature attribution methods, Lundberg and Lee introduced SHAP (Shapley additive explanations) as a unique solution determined by additional desirable properties (see the section Shapley values). Its uniqueness depends on defining a coalitional game, or set function, based on the model being explained—a concept first introduced by Strumbelj and Kononenko. Lundberg and Lee initially defined the game as the expectation of the model’s output when conditioned on a set of observed features; however, given the difficulty of computing conditional expectations in practice, the authors suggested using the marginal expectation, which ignores dependencies between the observed and unobserved features. This point of complexity has led to distinct Shapley value approaches that differ in how they remove features, as well as subsequent interpretations of how these two approaches relate to causal interventions.

1Paul G. Allen School of Computer Science & Engineering, University of Washington, Seattle, WA, USA. 2Microsoft Research, New York, USA. 3These authors contributed equally: Hugh Chen, Ian C. Covert. e-mail: suinlee@cs.washington.edu
information theory\textsuperscript{25,26}. Throughout this work, we refer to all feature attributions based on the Shapley value as Shapley value explanations.

Alongside the definition of the coalitional game, another challenge for Shapley value explanations is that their computational complexity is exponential in the number of features. The original SHAP paper\textsuperscript{15} therefore discussed several strategies for approximating Shapley values, including weighted linear regression (KernelSHAP\textsuperscript{16}), sampling feature combinations (IME\textsuperscript{21}), and several model-specific approximations (LinearSHAP\textsuperscript{17,22}, MaxSHAP\textsuperscript{18} and DeepSHAP\textsuperscript{19,23}). Since the original work, other methods have been developed to estimate Shapley value explanations more efficiently, using both model-agnostic (permutation\textsuperscript{24}, multilinear extension\textsuperscript{25} and FastSHAP\textsuperscript{26}) and model-specific (linear models\textsuperscript{27,28}, tree models\textsuperscript{29} and deep models\textsuperscript{30,31}) strategies. Of these two categories, model-agnostic approaches are more flexible but potentially biased. To better understand the model-agnostic approaches, we present a categorization of the approximation algorithms based on different mathematical definitions of the Shapley value, and we empirically compare their convergence properties (section Shapley value explanations). Then, to better understand the model-specific approaches, we highlight the key assumptions underlying each approach (section Algorithms to estimate Shapley value explanations).

These two challenges, properly removing features and accurately approximating Shapley values, are sources of complexity that have led to a wide variety of papers and algorithms on the subject. Unfortunately, the abundance of algorithms, coupled with the inherent complexity of the topic, have made the literature difficult to navigate. This can lead to misuse and misunderstanding, which is especially concerning because Shapley value explanations are now widely used, as shown in Fig. 1a. To address this, we aim to provide an approachable overview of the sources of complexity underlying the computation of Shapley value explanations.

We discuss these difficulties in detail, beginning by introducing the preliminary concepts of feature attribution (section Feature attributions) and the Shapley value (section Shapley values). Based on the various feature-removal approaches, we then describe popular variants of Shapley value explanations, as well as approaches to estimate the corresponding coalitional games (section Shapley value explanations). Next, based on the estimation strategies, we describe both model-agnostic and model-specific algorithms that rely on approximations and/or assumptions to tractably estimate Shapley value explanations (section Algorithms to estimate Shapley value explanations). These two sources of complexity provide a natural lens through which we present what is, to our knowledge, the first comprehensive survey of 24 distinct algorithms (excluding minor variations) that combine different feature-removal and tractable estimation strategies. Finally, we identify gaps in the field’s current knowledge and important future research directions.

**Feature attributions**

Given a model $f$ and input features $x_1, \ldots, x_d$, feature attributions explain predictions by assigning scalar values that represent each feature’s importance. For an intuitive description of feature attributions, consider the case of linear models. Linear models of the form $f(x) = \beta_0 + \beta_1 x_1 + \cdots + \beta_d x_d$ are often considered interpretable because each feature is linearly related to the prediction via a single parameter. In this case, a common global feature attribution that describes the model’s overall dependence on feature $i$ is the corresponding coefficient, or $\beta_i$. For linear models, each coefficient $\beta_i$ describes the influence that variations in feature $x_i$ have on the model output.

Alternatively, it may be preferable to give an individualized explanation that does not describe the model as a whole, but rather the prediction $f(x^*)$ given a specific sample $x^*$. These types of explanation are known as local feature attributions, and the sample being explained ($x^*$) is called the explicand. For linear models, one reasonable local feature attribution is $\phi_i(f, x^*) = \beta_i x_i^*$, because this is exactly the contribution that feature $i$ makes to the model’s prediction. However, note that this attribution implicitly compares against a value of $x_i = 0$. We may instead wish to consider alternative values based on the feature’s probability distribution, or to account for the feature’s statistical relationship with other features (section Shapley value explanations).

**Fig. 1** Shapley value explanations are widely used in practice. a, The large number of GitHub stars on SHAP (https://github.com/slundberg/shap), the most well-known package for estimating Shapley value explanations, indicates their popularity. b–d, A real-world example of Shapley value explanations for a tree ensemble model trained to predict whether individuals have income greater than US$50,000 based on census data: local feature attributions help explain a global understanding of important features (b); local feature attributions help explain nonlinear and interaction effects (c); local feature attributions explain how an individual’s features influence their outcome (d).
Linear models offer a simple case where we can understand each feature’s role via the model parameters, but this approach does not extend naturally to more complex models. For the model types that are most widely used today, including tree ensembles and deep learning models, the large number of operations prevents us from understanding each feature’s role by examining model parameters. These flexible, nonlinear models can capture more patterns in data, but they require us to develop more sophisticated and generalizable notions of feature importance. Thus, many researchers have begun turning to Shapley value explanations to identify important features (using summary plots as in Fig. 1b), to surface nonlinear effects (using dependence plots as in Fig. 1c), and to provide individualized explanations (using force plots as in Fig. 1d).

**Shapley values**

Shapley values are a tool from game theory that are designed to allocate credit among players in coalitional games. The players are represented by a set $D = \{1, \ldots, d\}$, and the ‘coalitional game’ is a function that maps from subsets of the players to a scalar value. A game is represented by a set function $v(S): 2^D \rightarrow \mathbb{R}$, where $2^D$ is the power set of $D$ and represents all possible subsets of players. See Fig. 2a for two example coalitional games.

To make these concepts more concrete, consider a company that makes a profit $v(S)$ determined by the set of employees $S \subseteq D$ that choose to work that day. A natural question is how to compensate the employees for their contribution to the total profit. Assuming we know the profit for all subsets of employees, Shapley values assign credit to an individual $i$ by calculating a weighted average of the profit increase $i$ makes for all subsets of employees.

Shapley values offer a compelling way to spread credit in coalitional games, and they have been widely adopted in fields such as computational biology and finance. Furthermore, they are a unique solution to the credit allocation problem as defined by several desirable properties. Figure 2b shows several key properties satisfied by the Shapley value, but there are additional implied properties distinguishing it from other credit allocation schemes.

**Shapley value explanations**

In this section, we present common strategies to define local feature attributions based on the Shapley value. A key choice is how to formulate a coalitional game representing the model’s prediction, and we discuss tradeoffs between the most common approaches in the literature.

**Machine learning models are not coalitional games**

Shapley values are an attractive solution for allocating credit in coalitional games, but our goal is to allocate credit among features $x_1, \ldots, x_d$ in a machine learning model $f(x) \in \mathbb{R}$. As shown in Fig. 3a, machine learning models are not coalitional games by default, so to use Shapley values we must first define a coalitional game $v(S)$ based on the model $f(x)$. The coalitional game can be chosen to represent various model behaviours, including the model’s loss for a single sample or for the entire dataset. We focus on the most common choice, which is explaining the prediction $f(x^*)$ for a single sample $x^*$.

When explaining a machine learning model, it is natural to view each feature $x_i$ as a player in the coalitional game. However, we must then define what is meant by the presence or absence of each feature. Given our focus on a single explanation $x^*$, the presence of feature $i$ will mean that the model is evaluated with the observed value $x_i$ (Fig. 3b). As for the absent features, we next discuss how to remove them to properly assess the influence of the present features.

**Removing features with baseline values**

One straightforward way to remove a feature is to replace its value using a baseline sample $x^*$. That is, if a feature $i$ is absent, we simply set...
that feature’s value to be $x_b^i$. The coalitional game is thus defined as $\nu(S) = \mathbb{E}(f(x^e, x_b^i, S))$, where we define $f(x^e, x_b^i, S) = x_e^i$ if $e \in S$ or $x_b^i$ otherwise (Fig. 3c). In other words, we evaluate the model on a hybrid sample where present features are the explicand values and absent features are the baseline values. As shorthand notation, we will refer to $f(t(x^e, x_b^i, S))$ as $f(x^e_S, x_b^i_S)$ in the remainder of this Review article.

The Shapley values for this coalitional game are referred to as ‘baseline Shapley values’23. This approach is simple to implement, but the choice of the baseline is not straightforward and can be somewhat arbitrary. Many different baselines have been considered, including an all-zeros baseline, an average across features, a baseline drawn from a uniform distribution, and more29,30,43,44,46,47. Unfortunately, the choice of baseline heavily influences the feature attributions, and the criteria for choosing a baseline can be unclear. One possible motivation could be to find a neutral or uninformative baseline, but such a baseline value may not exist for a given dataset. For these reasons, it is common to use a distribution of baselines instead of relying on a single value, as we discuss next.

Removing features with distributional values

Rather than setting the removed features to fixed baseline values, another option is to average the model’s prediction across randomly sampled replacement values. A first approach is to sample from the conditional distribution for the removed features. That is, given an explicand $x^e$ and subset $S \subseteq D$, we can consider the set of present features $x^e_S$ and sample replacement values for the absent features according to $x_b^i \sim p(x_b^i | x^e_S)$. In this case, the coalitional game is commonly defined as the expectation of the prediction $f(x^e_S, x_b^i)$ across this distribution, or

$$\nu(S) = \mathbb{E}_{p(x_b^i | x^e_S)}[f(x^e_S, x_b^i)] \tag{2}$$

where $x_b^i$ is considered a random variable following the conditional distribution $p(x_b^i | x^e_S)$. There are several names for Shapley values with this coalitional game: observational Shapley values23, conditional expectation Shapley21, and finally conditional Shapley values23, which is how we will refer to them. Two potential challenges with this approach are that estimating the conditional expectation is challenging (section Conditional Shapley values), and that the resulting explanations will spread credit among correlated features, which may or may not be desirable (section Tradeoffs between removal approaches).

An alternative approach is to use the marginal distribution when sampling replacement values. That is, we can ignore the values for the observed features $x^e_S$ and sample replacement values according to $x_b^i \sim p(x_b^i)$. As in the previous case, the coalitional game is defined as the expectation of the prediction across this distribution:

$$\nu(S) = \mathbb{E}_{p(x_b^i)}[f(x^e_S, x_b^i)] \tag{3}$$

where $x_b^i$ is treated as a random variable following the marginal distribution $p(x_b^i)$. This approach is equivalent to averaging over baseline Shapley values with baselines drawn from the data distribution $p(x)$ (ref. 23). It also has an interpretation as a causal explanation of the model, where the inputs of the model are distinct to the features in the real world; this is equivalent to assuming a flat causal graph (that is, a causal graph with no causal links among features)23,24. The latter interpretation has led to the name ‘interventional Shapley values’, but to avoid ambiguity we opt for the term marginal Shapley values25.

The conditional and marginal approaches are the most common feature-removal approaches in practice. Two other formulations based on random sampling are (1) the uniform approach, where absent features are drawn from a uniform distribution covering the feature range24,40,46 and (2) the product of marginals approach, where absent features are drawn from their individual marginal distributions (which assumes independence between all absent features)49,48. These formulations make strong feature independence assumptions, which may be why marginal Shapley values, which only assume independence between the observed and unobserved features, are more commonly

| Methods of handling absent features | Baseline (uniform) | Marginal | Conditional |
|-----------------------------------|--------------------|----------|-------------|
| Absent features                  | $x_b^i \sim p(x_b^i)$ | $x_b^i \sim p(x_b^i)$ | $x_b^i \sim p(x_b^i)$ |
| Present features                  | $x^e_S$            | $x^e_S$            | $x^e_S$            |
| Coalitional Game                  | $\nu(S) = \mathbb{E}[f(x^e_S, x_b^i)]$ | $\nu(S) = \mathbb{E}[f(x^e_S, x_b^i)]$ | $\nu(S) = \mathbb{E}[f(x^e_S, x_b^i)]$ |

Fig. 3 | Empirical strategies for handling absent features. a. Machine learning models have vector inputs, whereas coalitional games have set inputs. For simplicity of notation, we assume real-valued features, but Shapley value explanations can accommodate discrete features (unlike gradient-based methods). b. Present features are replaced according to the explicand. c. Absent features can be replaced according to a set of baselines with different distributional assumptions. In particular, the uniform approach uses the range of the baselines’ values to define independent uniform distributions from which to draw replacement values. The product of marginals approach draws each absent feature independently according to the values seen in the baselines. The marginal approach draws groups of absent feature values that appeared in the baselines. Finally, the conditional approach only considers samples that exactly match on the present features. Note that this figure depicts empirically estimating each distribution, and the explanations will spread credit among correlated features, which may or may not be desirable (section Tradeoffs between removal approaches).
used. In addition, there are several other approaches for handling absent features in Shapley value-like explanations, but these can often be interpreted as approximations of the aforementioned approaches. The three main removal approaches are visualized in Fig. 3d, where, for simplicity, we show empirical versions that use a finite set of baselines (for example, a training dataset) to compute each expectation.

Tradeoffs between removal approaches

Given the many ways to formulate the coalitional game, or to handle absent features, a natural question is ‘Which Shapley value explanation is preferred?’ This question is often debated in the Shapley value literature, with some papers defending marginal Shapley values20–24, others advocating for conditional Shapley values24–28, and still others arguing for causal solutions25. Before discussing their differences, one way in which these approaches are alike is that each version always satisfies the Shapley value axioms for its corresponding coalitional game, although the interpretation of the axioms can differ; this point has been discussed in earlier work29, but it is important to avoid conflating axioms defined relative to the coalitional game and to the model. In the following, we discuss tradeoffs between marginal and conditional Shapley values, because these are most commonly implemented in public repositories and discussed in the literature.

Conditional Shapley values tend to spread credit between correlated features, which can surface hidden dependencies26, whereas marginal Shapley values offer a closer description of the model’s functional form27 (Supplementary Information section Shapley value explanations for linear models). This discrepancy arises from the distributional assumptions, because conditioning on a feature implicitly introduces information about all correlated features, thereby leading to groups of features sharing credit as in Fig. 3c (conditional). For example, if we introduce the feature weight when body mass index (BMI) is absent, then conditional Shapley values will only consider values of BMI that make sense given the known weight value (that is, ‘on-manifold’ values): as a consequence, if the model depends on BMI but not weight, we would still observe that introducing weight affects the conditional expectation of the model output. By contrast, although marginal Shapley values perturb the data in less realistic ways (‘off-manifold’), they are able to distinguish between correlated features and identify whether the model functionally depends on BMI or weight, which is useful for model debugging28. The differences between these approaches imply that marginal Shapley values may be more appropriate when the user aims to understand the functional form of model, whereas conditional Shapley values may be more suitable when the user aims to understand an underlying mechanism in the data or the world.

Having two popular types of Shapley value explanation has been cited as a weakness29, but this issue is not unique to Shapley values: it is encountered by a large number of model explanation methods26 and is fundamental to understanding feature importance with correlated features sharing credit as in Fig. 3d: for linear models,30 marginal Shapley values correspond to a complete graph with cyclic dependencies between all features. Thus, the underlying causal graph plays a crucial role, and is perhaps more important than whether an approach can be said to be causal.

In this Review, we advocate for marginal and conditional Shapley values because they are more practical than causal approaches, and they avoid the problematic choice of a fixed baseline as in baseline Shapley values. In addition, they cover two of the most common use cases for Shapley value explanations and model interpretation in general: understanding a model’s informational dependencies and understanding a model’s functional form. An important final distinction between marginal and conditional Shapley values is the ease of estimation: as we discuss in the next section, marginal Shapley values turn out to be simpler to estimate than conditional Shapley values.

Algorithms to estimate Shapley value explanations

In this section we describe algorithmic approaches to address the two main challenges for generating Shapley value explanations: removing features to estimate the coalitional game and tractably calculating Shapley values despite their exponential complexity. A summary of the approaches used by specific implementations is provided in Table 1, and additional related work is discussed in the Supplementary Information section Related work.

Feature removal approaches

We previously introduced three main feature-removal approaches and discussed the tradeoffs between them. In this section, we reiterate their definitions and discuss how to calculate the corresponding coalitional games, which are required to calculate baseline, marginal and conditional Shapley values.

Baseline Shapley values. The coalitional game for baseline Shapley values is defined as \( v(S) = f(x^S, x^b) \), where \( f(x^S, x^b) \) denotes evaluating \( f \) on a hybrid sample with features from the explicand \( x^S \) and baseline \( x^b \). To compute the value of this coalitional game, we can simply create a hybrid sample and return the model’s prediction. It is possible to exactly compute this coalitional game, unlike the remaining approaches. The only parameter is the choice of baseline, which can be a somewhat arbitrary decision (section Baseline Shapley values).

Marginal Shapley values. For marginal Shapley values, the coalitional game is the marginal expectation of the model output, or \( v(S) = E_{p(x)}[f(x^S, x_b)] \), where \( x_b \) is treated as a random variable with distribution \( p(x_b) \). A natural approach to compute the marginal expectation is to obtain an empirical estimate using the training or test data. Data in machine learning are commonly assumed to be independent draws from the data distribution \( p(x) \), so we can designate a set of observed samples \( E \) as an empirical distribution and use their values for the absent features (as shown in Fig. 3d):

\[
\nu(S) = \frac{1}{|E|} \sum_{x^b \in E} f(x^S, x^b).
\]

From equation (4), it is clear that the empirical marginal expectation is the average over the coalitional games for baseline Shapley values with many different baselines. As a consequence, marginal Shapley values are also the average over baseline Shapley values with many different baselines28. For this reason, some algorithms estimate...
marginal Shapley values by first estimating baseline Shapley values with different baselines, and then averaging the results\(^{30,35}\). Note that marginal Shapley values based on empirical estimates are unbiased if the baselines are drawn i.i.d. from the baseline distribution (for example, a random subset of rows from the dataset). As such, this empirical estimate is considered a reliable way to approximate the true marginal expectation.

The empirical distribution can be the entire training dataset, but in practice it is often set to a moderate number of samples to reduce computation\(^{30,35}\). The primary parameter is the number of baseline samples and how to choose them; if a sufficient number of baselines is chosen, they can safely be chosen uniformly at random. This empirical approach can be applied to other coalitional games such as the uniform and product of marginals, which are similarly easy to estimate\(^{31,35}\), but we do not discuss these in detail.

**Conditional Shapley values.** For conditional Shapley values, the coalitional game is the conditional expectation of the model output, or \(\phi(S) = \mathbb{E}_{x_S | x_{\neg S}} f(x_S, x_{\neg S})\), where \(x_{\neg S}\) is considered a random variable with distribution \(p(x_S | x_{\neg S})\). Computing conditional Shapley values is more difficult because the required conditional distributions are not directly available from the training data. As shown in Fig. 3d (conditional), we can empirically estimate conditional expectations by averaging model predictions across samples matching the explicand’s present features, and this reliably estimates the conditional expectations as the dataset size goes to infinity. However, this empirical estimate does not work well in practice: the number of matching rows may be too low in the presence of continuous features or a large number of features, leading to inaccurate and unreliable estimates\(^{31,32}\). For example, if we condition on a height of 5.879 ft, there may be very few individuals with this exact height in our data, so the empirical conditional expectation will average over very few samples’ predictions, or potentially just the single prediction from the explicand itself.

One natural solution is to approximate the conditional expectation based on similar feature values rather than exact matches\(^{31,32}\). For instance, rather than filtering for individuals that are 5.879 ft tall, we can use those who are 5.879 ± 0.025 ft tall. We refer to methods of this type as empirical conditional distribution approximations (Table 1), because they are essentially empirical distributions coarsened by some notion of similarity\(^{33}\). Such approaches require defining a close-enough match in the observed features, which is not obvious and may be an undesirable prerequisite for an explanation method. Furthermore, conditioning on many features can still lead to inaccurate estimates, because we are less likely to find similar samples in high-dimensional spaces.

### Table 1 | Methods to estimate Shapley value explanations

| Method                          | Factors of complexity | Properties     |
|---------------------------------|-----------------------|----------------|
|                                 | Estimation strategy   | Removal approach | Removal variant | Model-agnostic | Bias-free | Variance-free |
| ApproSemivalue\(^{15}\)         | SV                    | None            | Exact           | Yes           | Yes       | No           |
| L-Shapley\(^{27}\)              | SV                    | Marginal        | Empirical       | Yes           | No        | No           |
| C-Shapley\(^{27}\)              | SV                    | Marginal        | Empirical       | Yes           | No        | No           |
| ApproShapley\(^{30}\)           | RO                    | None            | Exact           | Yes           | Yes       | No           |
| IME\(^{11}\)                    | RO                    | Uniform         | Empirical       | Yes           | No        | Yes          |
| CES\(^{33}\)                    | RO                    | Conditional     | Empirical       | Yes           | Yes       | No           |
| Shapley cohort refinement\(^{47}\) | RO                    | Conditional     | Empirical       | Yes           | Yes       | No           |
| Generative model\(^{16,23}\)   | RO                    | Conditional     | Generative      | Yes           | No        | No           |
| Surrogate model\(^{16}\)        | RO                    | Conditional     | Surrogate       | Yes           | No        | No           |
| Multilinear extension sampling\(^{45}\) | ME              | Baseline        | Empirical       | Yes           | Yes       | No           |
| SGD-Shapley\(^{30}\)            | LS                    | Baseline        | Exact           | Yes           | No        | No           |
| KernelSHAP\(^{15,72}\)          | LS                    | Marginal        | Empirical       | Yes           | Yes       | No           |
| Parametric KernelSHAP\(^{12,44}\) | LS              | Conditional     | Parametric      | Yes           | No        | No           |
| Empirical KernelSHAP\(^{10,23,53}\) | LS              | Conditional     | Empirical       | Yes           | No        | No           |
| FastSHAP\(^{32,38}\)            | LS                    | Conditional     | Surrogate       | Yes           | No        | No           |
| LinearSHAP\(^{26}\)             | Linear                | Marginal        | Empirical       | Yes           | No        | Yes          |
| Correlated LinearSHAP\(^{26}\)  | Linear                | Conditional     | Parametric      | No            | No        | No           |
| Interventional TreeSHAP\(^{48}\) | Tree                 | Marginal        | Empirical       | No            | Yes       | Yes          |
| Path-dependent TreeSHAP\(^{47,71}\) | Tree             | Conditional     | Empirical       | No            | No        | Yes          |
| DeepLIFT\(^{37}\)               | Deep                  | Baseline        | Exact           | No            | No        | Yes          |
| DeepSHAP\(^{37}\)               | Deep                  | Marginal        | Empirical       | No            | No        | Yes          |
| DASP\(^{35}\)                   | Deep                  | Base line       | Exact           | No            | No        | No           |
| Shallow ShapNet\(^{14}\)        | Deep                  | Baseline        | Exact           | No            | Yes       | Yes          |
| Deep ShapNet\(^{41}\)           | Deep                  | Baseline        | Exact           | No            | No        | Yes          |

We order approaches based on whether or not they are model-agnostic. Then, there are two factors of complexity. The first is the estimation strategy to handle the exponential complexity of Shapley values. For the model-agnostic approaches, the strategies include semivalue (SV), random order value (RO), multilinear extension (ME) and least-squares value (LS). Note that the model-agnostic estimation strategies can generally be adapted to apply to any removal approach. For model-specific approaches, the strategies differ for linear, tree and deep models. Then, the second factor of complexity is the feature-removal approach which determines the type of Shapley value explanation (section Feature removal approaches). ‘None’ denotes that it we do not discuss these in detail.

Correlated LinearSHAP\(^{26}\) | Linear | Conditional | Parametric | No | No | No
Interventional TreeSHAP\(^{48}\) | Tree | Marginal | Empirical | No | Yes | Yes
Path-dependent TreeSHAP\(^{47,71}\) | Tree | Conditional | Empirical | No | No | Yes
DeepLIFT\(^{37}\) | Deep | Baseline | Exact | No | No | Yes
DeepSHAP\(^{37}\) | Deep | Marginal | Empirical | No | No | Yes
DASP\(^{35}\) | Deep | Base line | Exact | No | No | No
Shallow ShapNet\(^{14}\) | Deep | Baseline | Exact | No | Yes | Yes
Deep ShapNet\(^{41}\) | Deep | Baseline | Exact | No | No | Yes
Instead of estimating conditional expectations based on matching data examples, a number of approaches based on fitting models have been proposed. Several were identified in the broader context of removal-based explanations, but we reiterate them here and summarize their practical strengths and weaknesses:

- **Parametric assumptions.** Chen et al.\(^{28}\) and Aas et al.\(^{47}\) assume Gaussian or Gaussian-copula distributions. Conditional expectations for Gaussian random variables have closed-form solutions and are computationally efficient once the joint distribution’s parameters are estimated, but these approaches can have large bias if the parametric assumptions are incorrect. Aas et al.\(^{47}\) extended this approach to vine copulas, which offer more flexibility when modelling joint data distributions.

- **Generative model.** Frye et al.\(^{48}\) and Olsen et al.\(^{55}\) use a conditional generative model to learn the conditional distributions given every subset of features. The generative model provides samples from approximate conditional distributions, and with these we can average model predictions to estimate the conditional expectation. In general, this approach is more flexible than simple parametric assumptions, but it has high variance due to the stochastic nature of training deep generative models, and it is difficult to assess whether the generative model accurately approximates the exponential number of conditional distributions.

- **Surrogate model.** Frye et al.\(^{48}\) and Olsen et al.\(^{55}\) use a surrogate model to directly learn the conditional expectation of the original model given every subset of features. The surrogate model is trained to match the original model’s predictions with arbitrarily held-out features, and doing so has been shown to directly approximate the conditional expectation, both for regression and classification models.\(^{60}\) This approach is as flexible as the generative model, but it has several practical advantages: it is simpler to train, it requires only one model evaluation to estimate the conditional expectation, and it has been shown to be more accurate in practice.\(^{69}\)

- **Missingness during training.** Covert et al.\(^{30}\) describe an approach for directly estimating the conditional expectation by training the original model to accommodate missing features. Unlike the previous approaches, this approach cannot be applied post hoc with arbitrary models because it requires modifying the training process.

- **Separate models.** Lipovetsky and Conklin\(^{55}\), Štrumbelj et al.\(^{57}\) and Williamson and Feng\(^{46}\) directly estimate the conditional expectation given a subset of features as the output of a model trained with that feature subset. If every model is optimal (for example, the Bayes classifier), then the conditional expectation estimate is exact.\(^{60}\) In practice, however, the various models will be sub-optimal and only loosely related to the original one, making it unsatisfying to view it as an explanation for the original model trained on all features. Furthermore, the computational cost of training models with many feature subsets is substantial, particularly for models such as tree ensembles and neural networks.

As this discussion shows, there exists a wide variety of approaches to model conditional distributions or directly estimate the conditional expectations. These approaches will generally be biased, or inexact, because the coalitional game we require is based on the true underlying conditional expectation. Compounding this, it is difficult to quantify the approximation quality because the conditional expectations are generally unknown, except in very simple cases (for example, synthetic multivariate Gaussian data).

Of these approaches, the empirical approach produces poor estimates; parametric approaches require strong assumptions, missingness during training is not model-agnostic, and fitting separate models for each feature set is not exactly an explanation of the original model. Instead, we believe approaches based on a generative model or a surrogate model are more promising. These approaches are more flexible, but both require fitting an additional deep model. To assess these approaches, Frye et al.\(^{48}\) propose two reasonable metrics based on mean squared error relative to the full model output. Future work may include identifying robust architectures and hyperparameters for surrogate and generative models, analysing how conditional Shapley value estimates change for non-optimal surrogate and generative models, and evaluating the approximation quality of conditional Shapley value estimates for data with known conditional distributions.

Some of the approaches we have discussed approximate the intermediate conditional distributions (empirical, parametric assumptions, generative model), whereas others directly approximate conditional expectations (surrogate model, missingness during training, separate models). It is worth noting that approaches based on modelling conditional distributions are independent of the particular model. This means that if a researcher fits a high-quality generative model to a popular dataset, then subsequent researchers can re-use this generative model to estimate conditional Shapley values for their own predictive models. However, even if fit properly, approaches based on modelling conditional distributions may be more computationally demanding because they require evaluating the model with many generated samples to estimate the conditional expectation. As such, the surrogate model approach may be more effective than the generative model approach in practice, and it has been used successfully in recent work.\(^{69,46}\)

In summary, when computing conditional Shapley values, there are two primary parameters: (1) the approach to model the conditional expectation, for which there are several choices, some of which require additional choices of model architecture, training routines and hyperparameters, and (2) the data used to estimate the conditional distribution or model the conditional expectation, because each approach requires a dataset to learn dependencies between features (for example, the entire training dataset). Different training data can lead to different scientific questions.\(^{46}\) For example, by using data drawn from older male subpopulations, we can ask why does an older male individual have a mortality risk of X% relative to the subpopulation of older males?\(^{49}\)

**Tractable estimation strategies**

Calculating Shapley values is, in the general case, an NP-hard problem.\(^{60,61}\) Intuitively, a brute-force calculation based on equation (1) has exponential complexity in the number of features, because it requires evaluating the model with all feature subsets. Given the long history of Shapley values, there is naturally considerable research into their calculation. Within the game theory literature, two types of estimation strategy have emerged: (1) approximation-based strategies that produce unbiased Shapley value estimates for any game and (2) assumption-based strategies that produce exact results in polynomial time for specific types of game.\(^{60,64}\)

These two strategies have also prevailed in Shapley value explanation literature. However, because some approaches combine both assumptions and approximations, we instead categorize approaches as either model-agnostic or model-specific. Model-agnostic approaches make no assumptions about the model class and are often stochastic, sampling-based estimators.\(^{46,62,63}\) In contrast, model-specific approaches rely on assumptions about the model’s structure to improve the speed of calculation, although sometimes at the expense of exactness.\(^{46,62,63,31,32}\) We next discuss many of these approaches in detail.

**Model-agnostic approaches.** There are several model-agnostic methods that estimate Shapley value explanations with random sampling. These approaches are generally unbiased but non-deterministic, producing correct results in expectation. Intriguingly, each model-agnostic approach can be tied to a distinct mathematical formulation of the Shapley value, of which there are many. For simplicity, we discuss these
approaches in the context of an arbitrary game and ignore the choice of feature-removal technique. We provide a brief overview of each method here, with more detailed information in the Supplementary Information section. Detailed description of model-agnostic algorithms.

The classic Shapley value is defined as a semivalue\(^{67}\), where each player’s credit is a weighted average of the player’s marginal contributions for all possible subsets with a weighting function \(P(S)\) that depends only on the subset’s cardinality. Castro et al.\(^{30}\) proposed ApproSemivalue, an unbiased, stochastic estimator for any semivalue, which samples subsets with probability given by \(P(S)\). ApproSemivalue estimates Shapley values when \(P(S) = \frac{1}{|S|}\) and does so independently for each player.

Another definition of the Shapley value is a random order value\(^{35,41}\), where a player’s credit is their average marginal contribution across many possible orderings. There are two approaches based on this definition: (1) IME (interactions-based method for explanation)\(^{21}\), which independently estimates the Shapley value for each player using random sets of sampled permutations and (2) ApproShapley\(^{47}\), which simultaneously estimates all players’ Shapley values using the same set of sampled permutations.

The least-squares value\(^{48,69}\) perspective defines the Shapley value as the solution to a weighted least-squares (WLS) problem. However, this method requires exponentially many ‘datapoints’ (one for each subset), meaning that the key challenge is not in solving the WLS problem, but in handling the large number of datapoints. A natural approximation is KernelSHAP\(^{38}\), which samples a moderate number of subsets and then solves the approximate WLS problem. Another approach that we refer to as SGD-Shapley similarly samples a moderate number of subsets, but instead iteratively estimates the WLS solution using projected stochastic gradient descent (SGD)\(^{70}\). Finally, FastSHAP\(^{32,43}\) uses a separate explainer model to amortize the WLS problem across many samples, enabling developers to forefront the cost of training the explainer, and provide subsequent users with fast Shapley value explanations.

The fourth definition is based on a multilinear extension of the game\(^{43,121}\), which extends a coalitional game to be a function on the \(d\)-cube \([0,1]^d\) that is separately linear in each variable. The Shapley value is then defined based on an integral of the multilinear extension’s partial derivatives. Based on this definition, ref. 31 introduced an unbiased sampling-based estimator to approximate this integral that we refer to as multilinear extension sampling.

Within these approaches there are several key variants. First, adaptive sampling speeds up convergence by allocating more samples to players with high variance in their marginal contributions. It has thus far only been applied to IME, but it can be extended to multilinear extension sampling. Second, efficient sampling improves convergence by generating samples intelligently. One version of efficient sampling, antithetic sampling, is used in ApproShapley, KernelSHAP and multilinear extension sampling. Antithetic sampling uses negatively correlated samples to improve convergence and is also known as ‘paired’ or ‘halved’ sampling\(^{43,124}\) in some contexts. Third, amortized explanation learns an explainer model to speed up explanation computation. It has only been applied to the least-squares definition of Shapley values\(^{31,39}\), but it can be extended to other definitions where the Shapley value is viewed as the solution to an optimization problem.

To better understand these approaches and their variants, in Fig. 4 we empirically compare four stochastic estimators (IME, ApproShapley, KernelSHAP and multilinear extension sampling) and two popular variants of these approaches (adaptive and antithetic sampling) in terms of their convergence to true Shapley values. We calculate the mean squared error of estimates of baseline Shapley values to examine convergence speed. We compute baseline Shapley values, as we aim to evaluate the estimation strategy rather than the feature-removal approach, as in ref. 73. We obtain the true baseline Shapley values with Interventional TreeSHAP.

First, adaptive and antithetic sampling variants consistently improve convergence at no additional cost. Next, we find that the best-performing method differs depending on the dataset. For the diabetes and NHANES datasets (10 and 79 features, respectively), the antithetic version of KernelSHAP converges fastest. However, for the blog dataset (280 features), IME and multilinear (feature) converge fastest, probably due to their use of adaptive sampling. We hypothesize that adaptive sampling is important for datasets with many features, because there are more likely to be features with little to no variance in their marginal contributions. Finally, our evaluation compared the convergence of baseline Shapley value estimates for multiple datasets, but we acknowledge that the choice of feature-removal approach and machine learning model may impact convergence. Therefore, future research should investigate the influence of these factors on convergence speed.

Model-agnostic strategies are appealing because they are applicable to any coalitional game, and therefore any model and feature-removal approach. The downside is that they are inherently stochastic and may have non-trivial variance. In response, some techniques have been developed to forecast and detect convergence\(^{72,74}\). However, model-agnostic explanations can still be computationally costly, so we next discuss methods that estimate Shapley value explanations more efficiently for specific model types.

Model-specific approaches. In terms of model-specific approaches, specialized algorithms have been developed for several popular model types: linear models, tree models and deep models. These are less flexible than model-agnostic approaches, in that they assume a specific model type and often a specific feature-removal approach, but they are generally substantially faster to calculate. We describe each method at a high level here, but Supplementary Information section Detailed description of model-specific algorithms provides more details.

First, there are two model-specific approaches for linear models: LinearSHAP and correlated LinearSHAP. LinearSHAP computes baseline and marginal Shapley values exactly, with a time complexity that scales linearly with the number of features. Correlated LinearSHAP is a stochastic estimator for conditional Shapley values that assumes the data follow a multivariate Gaussian distribution; the estimates are biased when the data do not follow a multivariate Gaussian distribution, and the estimates are not variance-free because they require sampling to estimate summations over an exponential number of coalitions. However, correlated LinearSHAP may still provide a good cost–accuracy tradeoff compared to model-agnostic methods, because it incurs most of its computational cost up-front and then computes explanations very quickly (similar to FastSHAP\(^{35}\)).

Next, there are two model-specific approaches for tree models: path-dependent TreeSHAP and Interventional TreeSHAP. Interventional TreeSHAP computes baseline and marginal Shapley values exactly, and in time linear in the size of the tree and the number of baseline values. Path-dependent TreeSHAP estimates conditional Shapley values deterministically, but it is biased because it assumes the tree model itself can approximate conditional expectations. This assumption is analogous to the one made by Shapley cohort refinement\(^{45}\), but the cohort here is defined by the splits within the tree model. To estimate Shapley value explanations for tree ensemble models, both approaches calculate explanations separately for each tree in the ensemble and then take their linear combination across the trees. This provides an exact calculation for baseline and marginal Shapley values due to the Shapley value’s linearity property\(^{45}\).

Finally, there are several model-specific approaches for deep models: DeepLIFT\(^{77}\), DeepSHAP\(^{39}\), Deep Approximate Shapely Propagation (DASP)\(^{29}\) and Shapley Explanation Networks (ShapNets)\(^{29}\). DeepLIFT propagates attributions through a deep network based on a set of rules, which can be viewed as a layer-wise approximation.
of the Shapley value for a single baseline. Due to these approximations, DeepLIFT produces biased estimates of baseline Shapley values. DeepSHAP extends DeepLIFT to produce biased estimates of marginal Shapley values. DASP utilizes uncertainty propagation and a characterization of the Shapley value that averages the expected marginal contribution for each coalition size to estimate baseline Shapley values. DASP is deterministic with $O(d^2)$ model evaluations, where $d$ is the number of features, but it can be used in a stochastic manner using fewer evaluations. ShapNets are specific deep architectures for which Shapley values are easier to estimate: they require layers with specific connectivity constraints that enable fast estimation of baseline Shapley values. In terms of their assumptions, ShapNets are the most restrictive because they cannot explain other deep models, and DASP is also restrictive because it requires first- and second-order central moment matching for each layer in the deep model, which is only described for certain layers. DeepLIFT and DeepSHAP are more flexible, because their rules are generally applicable to many layers, but as a consequence of DeepLIFT’s flexibility, its baseline Shapley value estimates have higher bias compared to DASP or ShapNets. In general, these model-specific approaches can be much faster than model-agnostic alternatives, but their high bias can be problematic depending on the specific model and dataset.

**Discussion**

In this Review, we describe numerous algorithms to generate Shapley value explanations. We have delved into the two main factors underlying their complexity: the feature-removal approach and the tractable estimation strategy. Disentangling their complexity allows us to more easily understand key innovations in recent approaches and provide practical recommendations (Supplementary Information section Practical recommendations).

In terms of the different Shapley value explanations, algorithms that estimate baseline Shapley values are generally unbiased, but choosing a single baseline to represent absent information is often arbitrary. Similarly, algorithms that estimate marginal Shapley values are often unbiased. In contrast, algorithms that estimate conditional Shapley values are more error-prone because the conditional expectation is fundamentally more challenging to estimate; as a result, conditional Shapley values are currently difficult to approximate except in the special case of linear models. However, depending on the intended use case, it may be preferable to use an imperfect approximation than to switch to baseline or marginal Shapley values.

In terms of the exponential complexity of Shapley values, model-agnostic approaches are more flexible and often bias-free, but they can have large variance if they are not run with a sufficient number of

---

**Fig. 4** | Benchmarking unbiased, model-agnostic algorithms to estimate baseline Shapley values for a single explicand and baseline on XGB models with 100 trees. The algorithms include multilinear, random order, random order (feature) and least-squares, which correspond to multilinear extension sampling, ApproShapley, IME and KernelSHAP, respectively. Multilinear (feature) is a new approach based on multilinear characterization, which explains one feature at a time. In addition, some methods are variants—either antithetic or adaptive sampling. On the $x$ axis we report the number of samples (subsets) used for each estimate, and on the $y$ axis we report the mean squared error relative to the true baseline Shapley value for 100 estimates with that many samples. We use three real-world datasets: diabetes (10 features, regression), NHANES (79 features, classification) and blog (280 features, regression). For some variants, no error is shown for small numbers of samples, because each approach requires a different minimum number of samples. A, Variants of the random-order, feature-wise strategy. B, Benchmarking the most competitive variants chosen according to the lowest error for 10^5 samples. The full blog error plot (Supplementary Fig. 5) is truncated. Additional results are provided in Supplementary Figs. 2–4 and Supplementary Tables 1–6.
samples. In contrast, model-specific approaches are typically deterministic and fast, but they are biased except in special cases. Of the model-specific methods, only LinearSHAP and Interventional TreeSHAP have no bias for baseline and marginal Shapley values. We find that Interventional TreeSHAP in particular is remarkable for being non-trivial, bias-free and variance-free. As such, tree models including decision trees, random forests and gradient boosted trees are particularly well-suited to Shapley value explanations.

As a result of disentangling these factors of complexity, we can understand the sources of bias and variance within many existing algorithms (Table 1). For example, the method proposed by Okhrati and Lipani13 is bias-free, because the coalitional game for baseline Shapley values is exact, and multilinear extension sampling is an unbiased estimation algorithm; however, their estimates have non-zero variance because their estimation strategy is stochastic. By contrast, the Shapley cohort refinement estimates approach13 have both non-zero bias and non-zero variance; here, the bias comes from modeling the conditional expectation using an empirical similarity-based approach, and the variance comes from the random order-based sampling strategy.

Shapley value explanations are now widely used in both industry and academia. They are powerful tools, but it is important for users to be aware of important parameters of the algorithms used to estimate them. In particular, we recommend that any analysis based on Shapley values should report parameters, including the type of Shapley value explanation (the feature-removal approach), the data or baselines used to estimate the coalitional game, and the specific estimation strategy. For sampling-based strategies, it is important for users to include a discussion of convergence so as to validate their feature-attribution estimates. Finally, developers of Shapley value explanation tools should strive to be transparent about convergence by explicitly performing automatic convergence detection. Convergence results based on the central limit theorem or concentration inequalities are straightforward for the majority of model-agnostic estimators discussed here, but they are not always implemented in public packages. Note that convergence analysis is more difficult for the least-squares estimators, but Covert and Lee2 discuss this issue and present a convergence detection approach for KernelSHAP.

Our overview highlights several important future research directions. One direction is investigating new convergence criteria for model-agnostic methods. Existing work detects when the largest estimated standard deviation is smaller than a prescribed threshold72. There are many possible alternatives, such as when additional marginal contributions are unlikely to change the ranking of important features. Another important future direction is Shapley value explanations for deep models, which are now predominant in many applications. Current model-specific algorithms for deep models are biased, and no model-specific algorithms can estimate conditional Shapley values. One promising approach is FastSHAP14,15, which speeds up explanations using an explainer model, but it requires a large up-front cost to train the explainer model. Finally, because approximating the conditional expectation for conditional Shapley values is difficult, this constitutes a research direction that would benefit from new methods, as well as systematic evaluations of existing approaches.

**Data availability**
The diabetes dataset is publicly available (https://www4.stat.ncsu.edu/~boos/var.select/diabetes.html), and we use the version from the sklearn package. The NHANES dataset is publicly available (https://www.cdc.gov/nchs/nhanes/nhes/), and we use the version from the SHAP package. The blog dataset is publicly available (https://archive.ics.uci.edu/ml/datasets/BlogFeedback).

**Code availability**
Code for the experiments is available at https://github.com/suinleelab/shapley_algorithms.

**References**
1. Silver, D. et al. Mastering the game of Go without human knowledge. *Nature* **550**, 354–359 (2017).
2. Moravcik, M. et al. Deepstack: expert-level artificial intelligence in heads-up no-limit poker. *Science* **356**, 508–513 (2017).
3. Vinyals, O. et al. Grandmaster level in StarCraft II using multi-agent reinforcement learning. *Nature* **575**, 350–354 (2019).
4. Jumper, J. et al. Highly accurate protein structure prediction with AlphaFold. *Nature* **596**, 583–589 (2021).
5. Jean, S., Cho, K., Memisevic, R. & Bengio, Y. On using very large target vocabulary for neural machine translation. In Proc. 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1, Long Papers) P15-1001 (2015).
6. Breiman, L. Random forests. *Mach. Learn.* **45**, 5–32 (2001).
7. LeCun, Y., Bengio, Y. & Hinton, G. Deep learning. *Nature* **521**, 436–444 (2015).
8. Chen, T. & Guestrin, C. XGBoost: a scalable tree boosting system. In Proc. 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining 785–794 (ACM, 2016).
9. Steinkraus, D., Buck, I. & Simard, P. Using GPUs for machine learning algorithms. In Proc. Eighth International Conference on Document Analysis and Recognition (ICDAR’05) 1115–1120 (IEEE, 2005).
10. Geirhos, R. et al. Shortcut learning in deep neural networks. *Nat. Mach. Intell.* **2**, 665–673 (2020).
11. Doshi-Velez, F. & Kim, B. Towards a rigorous science of explainable AI for black boxes. In Proc. Conference on Fairness, Accountability and Transparency 48–48 (PMLR, 2018).
12. Selbst, A. & Powles, J. ‘Meaningful information’ and the right to explanation. In Proc. Conference on Fairness, Accountability and Transparency 48–48 (PMLR, 2018).
13. Knight, E. AI and machine learning-based credit underwriting and adverse action under the ECOA. *Bus. Fin. L. Rev.* **2005**.
14. Ribeiro, M. T., Singh, S. & Guestrin, C. ‘Why should I trust you?’ Explaining the predictions of any classifier. In Proc. 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining 1135–1144 (ACM, 2016).
15. Lundberg, S. M. & Lee, S.-I. A unified approach to interpreting model predictions. In *Advances in Neural Information Processing Systems* 4765–4774 (ACM, 2017).
16. Lundberg, S. M. et al. From local explanations to global understanding with explainable AI for trees. *Nat. Mach. Intell.* **2**, 56–67 (2020).
17. Shrikumar, A., Greenside, P. & Kundaje, A. Learning important features through propagating activation differences. In Proc. 34th International Conference on Machine Learning Vol. 70, 3145–3153 (JMLR, 2017).
18. Binder, A., Montavon, G., Lapuschkin, S., Muller, K.-R. & Samek, W. Layer-wise relevance propagation for neural networks with local renormalization layers. In Proc. International Conference on Artificial Neural Networks 63–71 (Springer, 2016).
19. Datta, A., Sen, S. & Zick, Y. Algorithmic transparency via quantitative input influence: theory and experiments with learning systems. In *Proc. 2016 IEEE Symposium on Security and Privacy (SP)* 598–617 (IEEE, 2016).
20. Sundararajan, M., Taly, A. & Yan, Q. Axiomatic attribution for deep networks. In *Proc. International Conference on Machine Learning* 3319–3328 (PMLR, 2017).
21. Strumbelj, E. & Kononenko, I. An efficient explanation of individual classifications using game theory. *J. Mach. Learn. Res.* **11**, 1–18 (2010).
Review article https://doi.org/10.1038/s42256-023-00657-x

64. Illés, F. & Kerényi, P. Estimation of the Shapley value by ergodic sampling. Preprint at https://arxiv.org/abs/1906.05224 (2019).
65. Megiddo, N. Computational complexity of the game theory approach to cost allocation for a tree. Math. Op. Res. 3, 189–196 (1978).
66. Granot, D., Kuipers, J. & Chopra, S. Cost allocation for a tree network with heterogeneous customers. Math. Op. Res. 27, 647–661 (2002).
67. Dubey, P., Neyman, A. & Weber, R. J. Value theory without efficiency. Math. Op. Res. 6, 122–128 (1981).
68. Charnes, A., Golany, B., Keane, M. & Rousseau, J. in Econometrics of Planning and Efficiency 123–133 (Springer, 1988).
69. Ruiz, L. M., Valenciano, F. & Zarzuelo, J. M. The family of least square values for transferable utility games. Games Econ. Behav. 24, 109–130 (1998).
70. Simon, G. & Vincent, T. A projected stochastic gradient algorithm for estimating Shapley value applied in attribute importance. In Proc. International Cross-Domain Conference for Machine Learning and Knowledge Extraction 97–115 (Springer, 2020).
71. Owen, G. Multilinear extensions of games. Manag. Sci. 18, 64–79 (1972).
72. Covert, I. & Lee, S.-I. Improving KernelSHAP: practical Shapley value estimation using linear regression. In Proc. International Conference on Artificial Intelligence and Statistics 3457–3465 (PMLR, 2021).
73. Mitchell, R., Cooper, J., Frank, E. & Holmes, G. Sampling permutations for Shapley value estimation. J. Mach. Learn. Res. 23, 1–46 (2022).
74. Covert, I., Lundberg, S. M. & Lee, S.-I. Understanding global feature contributions with additive importance measures. Advances Neural Information Processing Systems 33, 17212–17223 (2020).
75. Yang, J. Fast TreeSHAP: accelerating SHAP value computation for trees. Preprint at https://arxiv.org/abs/2109.09847 (2021).

Acknowledgements
We thank P. Sturmfels, J. Janizek, G. Erion and A. DeGrave for discussions. This work was funded by the National Science Foundation (DBI-1759487, DBI-1552309, DGE-1762114 and DGE-1256082) and the National Institutes of Health (R35 GM 128638 and R01 NIA AG 061321).

Competing interests
The authors declare no competing interests.

Additional information

Supplementary information The online version contains supplementary material available at https://doi.org/10.1038/s42256-023-00657-x.

Correspondence and requests for materials should be addressed to Su-In Lee.

Peer review information Nature Machine Intelligence thanks Martin Jullum and Benedek Rozemberczki for their contribution to the peer review of this work.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher’s note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.

© Springer Nature Limited 2023