Shapley-based explainability on the data manifold

Christopher Frye, Damien de Mijolla, Laurence Cowton, Megan Stanley, Ilya Feige

Faculty, 54 Welbeck Street, London, UK

Abstract

Explainability in machine learning is crucial for iterative model development, compliance with regulation, and providing operational nuance to model predictions. Shapley values provide a general framework for explainability by attributing a model's output prediction to its input features in a mathematically principled and model-agnostic way. However, practical implementations of the Shapley framework make an untenable assumption: that the model's input features are uncorrelated. In this work, we articulate the dangers of this assumption and introduce two solutions for computing Shapley explanations that respect the data manifold. One solution, based on generative modelling, provides flexible access to on-manifold data imputations, while the other directly learns the Shapley value function in a supervised way, providing performance and stability at the cost of flexibility. While the commonly used "off-manifold" Shapley values can (i) break symmetries in the data, (ii) give rise to misleading wrong-sign explanations, and (iii) lead to uninterpretable explanations in high-dimensional data, our approach to on-manifold explainability demonstrably overcomes each of these problems.

1 Introduction

AI's potential to improve economic productivity is driven by its ability to significantly reduce the cost of predictions [3]. For these predictions to be beneficial, they should be mostly correct, operationally consumable, and cannot lead to unexpected systemic harm. The ability to explain how AI models make their predictions is a critical step towards this goal. The discipline of AI explainability is thus central to the practical impact of AI on society.

One could conservatively demand that only simple, by-construction-interpretable models are used for predictions that meaningfully impact people's lives [26]. Such an approach, however, sacrifices the performance upside of complex, non-interpretable models. This motivates the study of *post-hoc* AI explainability, where the goal is to explain arbitrarily complex models.

Further distinction exists between *model-specific* and *model-agnostic* explainability. Model-specific methods explain a model's predictions by referencing its internal structure; see e.g. [7] or [28]. Model-agnostic methods explain predictions through input-output attribution, treating the model as a black box. Not only do model-agnostic methods offer general applicability, but they also provide a common language for explainability that does not require expert knowledge of the model.

Within the paradigm of post-hoc, model-agnostic explainability, a number of methods are used in practice. Many measure the effect of varying features on model performance [5, 29] or an individual prediction [4]. Another method fits an interpretable model to the original around the point of prediction to garner local understanding [25]. However, these methods are widely ad-hoc and founded on prohibitively stringent assumptions, e.g. independence or linearity.

Fortunately, the general problem of attribution, of which model-agnostic explainability is an example, has been extensively developed in cooperative game theory. Shapley values [27] provide the unique attribution method satisfying 4 intuitive axioms: they capture all interactions between features, they sum to the model prediction, and their linearity enables aggregation without loss of theoretical control. Shapley-based AI explainability has matured over the last two decades [17, 15, 30, 8, 20].

However, Shapley values suffer from a problematic assumption: they involve marginalisation over subsets of features, generally achieved by splicing data points together and thus evaluating the model on highly unrealistic data (e.g. Fig. 1). While such splicing is common in model-agnostic methods, it is only justified if all the data's features are independent, an assumption almost never satisfied; otherwise, such spliced data lies *off the data manifold*. While work has been done towards remedying this flaw [1, 23, 18], a satisfactorily general and performant solution has yet to appear.

In this paper, we provide a detailed study of the off-manifold problem in explainability, and provide solutions to computing Shapley values *on the data manifold*. Our main contribution is the introduction of two new methods to compute on-manifold Shapley values for high-dimensional, multi-type data:

1. a flexible generative-modelling technique to learn on-manifold conditional distributions;
2. a robust supervised-learning technique that learns the on-manifold value function directly.

In Sec. 2, we provide precise definitions of key quantities in Shapley explainability, including *global Shapley values*, which to our knowledge have not been introduced elsewhere. In Sec. 3, we elucidate the conceptual difference between off- and on-manifold explanations and the marked drawbacks of off-manifold approaches. After presenting our solutions in Sec. 4, we demonstrate the practical effectiveness of on-manifold explainability with varied experiments in Sec. 5.

2 Shapley values on the data manifold

Here we review the Shapley framework for model explainability, define on-manifold Shapley values precisely, and introduce global explanations that obey the Shapley axioms.

2.1 Shapley values for model explainability

In cooperative game theory, a team $N = \{1, 2, \ldots, n\}$ of players work together to earn value $v(N)$ [33]. Given a value function $v : 2^N \to \mathbb{R}$ indicating the value $v(S)$ that a coalition $S \subseteq N$ of players would earn on their own, the *Shapley value* $\phi_v(i)$ provides a principled approach to distributing credit for the total earnings $v(N)$ among the players $i \in N$ [27]:

$$\phi_v(i) = \sum_{S \subseteq N \setminus \{i\}} \frac{s!\,(n-s-1)!}{n!} \big[v(S \cup \{i\}) - v(S)\big] \tag{1}$$

The Shapley value $\phi_v(i)$ computes player i's marginal value-added upon joining the team, averaged over all orderings in which the team can be constructed.

In the context of supervised learning, let $f_y(x)$ represent a model's predicted probability that data point x belongs to class y, so that $\sum_y f_y(x) = 1$. To apply Shapley attribution to model explainability, one interprets the input features $\{x_1, \ldots, x_n\}$ as players in a game and the output $f_y(x)$ as their earned value. To compute the Shapley value of each feature x_i, one must define a value function to represent the outcome of the model on a restricted coalition of inputs $x_S \subseteq \{x_1, \ldots, x_n\}$.

While the value function should act as a proxy for "$f_y(x_S)$", the model is undefined given only partial input x_S, so one cannot leave out-of-coalition slots empty. In the standard treatment [20] one averages over out-of-coalition features, $x'_{\bar{S}}$ where $\bar{S} = N \setminus S$, drawn unconditionally from the data:

$$v^{(\text{off})}_{f_y(x)}(S) = \mathbb{E}_{p(x')}\big[f_y(x_S \sqcup x'_{\bar{S}})\big] \tag{2}$$

We refer to this value function, and the corresponding Shapley values $\phi_{f_y(x)}(i)$, as lying *off-manifold* since the splices $x_S \sqcup x'_{\bar{S}}$ generically lie far from the data manifold (e.g. Fig. 1). Even so, the Shapley framework guarantees that model explanations satisfy an intuitive set of properties [27]:

- *Efficiency.* Shapley values distribute the model prediction $f_y(x)$ fully among the features, up to an offset term (not attributed to any feature) representing the average probability f assigns to class y:

$$\sum_{i \in N} \phi_{f_y(x)}(i) = f_y(x) - \mathbb{E}_{p(x')}\big[f_y(x')\big] \tag{3}$$

- *Linearity.* Shapley values aggregate linearly in a linear-ensemble model.
- *Nullity.* Features that do not influence the value function $v_{f_y(x)}(S)$ receive zero Shapley value.
- *Symmetry.* Features that influence the value function identically receive equal Shapley values.

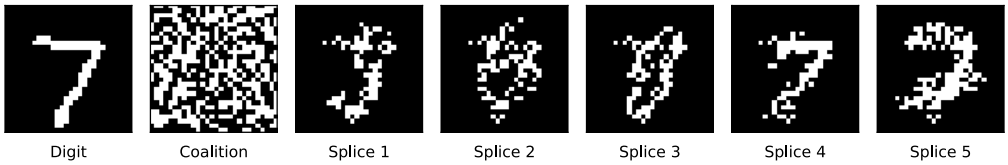


Figure 1: An MNIST digit, a coalition of pixels in a Shapley calculation, and 5 off-manifold splices.

2.2 On-manifold Shapley values

In practice, Shapley explanations are widely based on the off-manifold value function, Eq. (2), which evaluates the model on splices, $f(x_S \sqcup x'_{\bar{S}})$, with x' drawn independently of x. Splicing features from unrelated data points generically leads to unrealistic model inputs. Such unrealistic splices lies outside the model's regime of validity, where there is no reason to expect controlled model behaviour. Off-manifold explanations thus obfuscate insights into the model's behaviour on real data.

To fix the off-manifold problem, one should condition out-of-coalition features $x'_{\bar{S}}$ on in-coalition features x_S, thus basing Shapley explanations on an *on-manifold* value function:

$$v^{(\text{on})}_{f_y(x)}(S) = \mathbb{E}_{p(x'|x_S)}\big[f_y(x_S \sqcup x'_{\bar{S}})\big] \tag{4}$$

Note that, since the *Nullity* and *Symmetry* axioms reference the value function directly, these properties will manifest differently off- and on-manifold; see e.g. Secs. 3.1 and 5.2.

Preference for an on-manifold value function is widely acknowledged [20, 12, 21]. However, the requisite conditional distribution $p(x'|x_S)$ is not empirically accessible in practical scenarios with high-dimensional data or features that take many (e.g. continuous) values. A performant method to estimate the on-manifold value function is until-now lacking and the focus of this work.

2.3 Global Shapley values

As presented above, Shapley values provide a method for local explainability, explaining prediction $f_y(x)$ on individual data point x. For a global understanding of model behaviour, one might average $\phi_{f_y(x)}(i)$ over the data, with the class-of-interest y fixed. However, for an important feature i, its local Shapley value can vary between large-positive and large-negative values, as x_i may correlate with y in some regions and anti-correlate in others. As this would lead to large cancellations, it is common to average the absolute value $|\phi_{f_y(x)}(i)|$ instead [18]. However, such a nonlinear aggregation leads to a global explanation that breaks the axioms underlying the Shapley framework.

To both preserve the Shapley axioms and avoid large cancellations, we define *global Shapley values*:

$$\Phi_f(i) \;=\; \mathbb{E}_{p(x,y)}\big[\phi_{f_y(x)}(i)\big] \tag{5}$$

where $p(x, y)$ is the distribution from which the labelled data is drawn, and – crucially – class y varies with data point x in the average. Global Shapley values obey a sum rule that follows from Eq. (3):

$$\sum_{i\in N} \Phi_f(i) \;=\; \mathbb{E}_{p(x,y)}\big[f_y(x)\big] - \mathbb{E}_{p(x')}\mathbb{E}_{p(y)}\big[f_y(x')\big] \tag{6}$$

On can thus interpret the global Shapley value $\Phi_f(i)$ as the portion of the model's accuracy attributable to the i^{th} feature. Indeed, the first term in Eq. (6) is the accuracy one achieves by drawing labels from f's predicted probability distribution over classes. The offset term is the accuracy one achieves using *none* of the features: drawing the label of x from the model's output $f_y(x')$ on a random input x'.

3 Off- versus on-manifold Shapley values

This section articulates key differences between model explanations *off* versus *on* the data manifold.

3.1 Functional versus informational dependence

The only effect in-coalition features x_S have in the off-manifold value function, Eq. (2), is through their role as direct model inputs, $f_y(x_S \sqcup x'_{\bar{S}})$. It follows that if f does not have explicit functional dependence on feature x_i, then the off-manifold Shapley value $\phi_{\text{off}}(i)$ vanishes.

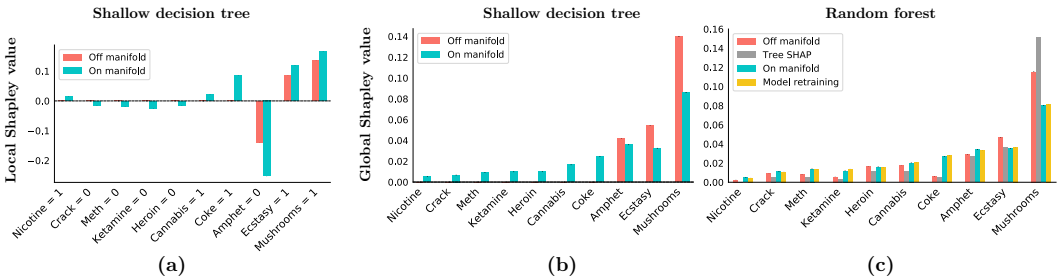


Figure 2: Explaining shallow decision tree (a & b) and random forest (c) on Drug Consumption data.

By contrast, in-coalition features x_S affect the on-manifold value function, Eq. (4), through a second channel: implicitly through f's dependence on $x'_{\bar{S}}$ when x_S and $x'_{\bar{S}}$ correlate. The on-manifold Shapley value $\phi_{\text{on}}(i)$ can thus be nonzero even for a feature x_i that f does not act upon directly. In such a case, the model *does* use information in x_i, but extracts it via other features.

To demonstrate this on the Drug Consumption data from the UCI repository [9], we used the 10 binary features listed in Fig. 2 (Mushrooms, Ecstacy, etc.) to predict whether individuals had consumed an 11th drug: LSD. We modelled this data using a shallow decision tree f with only 3 nodes.

Fig. 2(a) shows local explanations of the decision tree's prediction "LSD = True" for a test-set individual with features listed on the horizontal axis. The explanations are computed using Monte Carlo approximations to Eq. (1), with value functions approximated using the empirical distribution (accessible in this case, with just 10 binary features). Note that the off-manifold Shapley values are nonzero only for the 3 features that f depends on explicitly, while the on-manifold explanation indicates f's implicit dependence on information contained in all features.

Fig. 2(b) shows global Shapley values for this shallow decision tree. These global explanations are the expectation values of local explanations, as in Eq. (5). Note that all on-manifold global Shapley values are non-negative, consistent with their interpretation as the portion of model accuracy attributable to the information contained in each feature.

3.2 The garbage-in, garbage-out problem

While Sec. 3.1 might lead one to believe that off-manifold explainability provides useful insight into the functional dependence of a model, it serves as a perilously uncontrolled approach, especially in complex nonlinear models such as neural networks. Indeed, it is widely known that machine learning models are not robust to distributional shift [22, 10]. Still, the off-manifold value function of Eq. (2) evaluates the model outside its domain of validity, where it is untrained and potentially wildly misbehaved, in hopes that an aggregation of such evaluations will be meaningful. This garbage-in, garbage-out problem is the clearest reason to avoid off-manifold Shapley values.

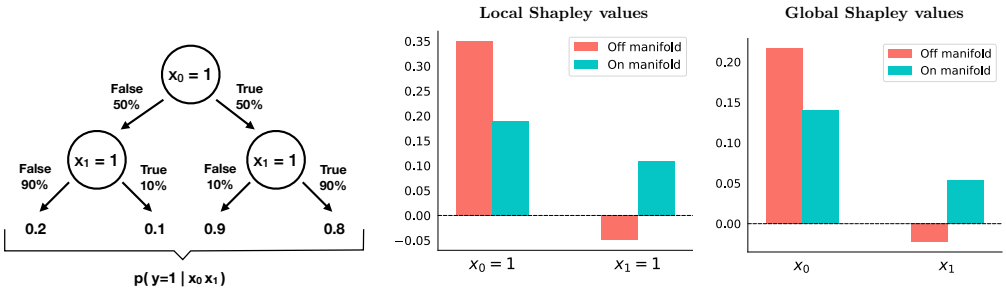
Since this point is understood in the literature [12, 32], we simply provide an example of this problem in Fig. 1, which shows an example binary MNIST digit x [16], a coalition S of pixels, and 5 random splices $x_S \sqcup x'_{\bar{S}}$ that would be used in an off-manifold explanation.

3.3 Misleading explanations off manifold

To demonstrate that off-manifold Shapley values can be misleading in practice, we generated synthetic data according to the process in Fig. 3(a). The data has two binary features and a binary label, all class-balanced. We fit a decision tree to this data: a precise match to Fig. 3(a).

Note that the features x_0 and x_1 are positively correlated, both with each other and with label y. However, with x_0 fixed, the likelihood of $y = 1$ decreases slightly from $x_1 = 0$ to $x_1 = 1$. One might think of x_0 as disease severity, x_1 as treatment intensity, and y as mortality rate.

The local Shapley values for the frequent scenario $(x_0, x_1, y) = (1, 1, 1)$ are plotted in Fig. 3(b). We find the negative off-manifold Shapley value shown for x_1 to be misleading, as it would suggest that the observation $x_1 = 1$ is more commonly associated with a prediction of $y = 0$ (a 20% occurrence), rather than the true label $y = 1$ (80%). The negative value of $\phi_{\text{off}}(1)$ is due to the model's decreased



(a) (b) (c)

Figure 3: Local and global explanations of decision tree fit to simple synthetic data set.

confidence in $y = 1$ when one goes from $(x_0 = 1, x_1 = 0)$ to $(x_0 = 1, x_1 = 1)$, as well as from $(x_0 = 0, x_1 = 0)$ to $(x_0 = 0, x_1 = 1)$. This misleading sign is therefore due to $\phi_{\text{off}}(1)$'s heavy sensitivity to the model's behaviour when $x_0 \neq x_1$, despite this being exceedingly rare in the data.

Fig. 3(c) displays global Shapley values for this model. Note that the on-manifold global values are positive, consistent with their interpretation as the portion of model accuracy attributable to each feature. However, there is a negative off-manifold global value that results from aggregating wrong-sign local explanations. Such a negative value would indicate that input x_1 is actually detrimental to the model's overall performance, which of course is not the case.

3.4 On-manifold Shapley in the non-parametric limit

Here we present a result that strengthens the connection between on-manifold Shapley values and the data distribution: in the limit of a perfect model of the data, on-manifold Shapley values converge to an explanation of how the information in the data associates with the labelled outcomes.

To show why this holds, suppose the predicted probability $f_y(x)$ converges to the true underlying distribution $p(y|x)$. In this non-parametric limit, the on-manifold value function of Eq. (4) becomes

$$v^{(\text{on})}_{f_y(x)}(S) \to \int dx'_{\bar{S}}\; p(x'_{\bar{S}}|x_S)\; p(y \,|\, x_S \sqcup x'_{\bar{S}}) = p(y \,|\, x_S) \tag{7}$$

in which case on-manifold value is attributed to x_i based on x_i's predictivity of the label y.

To demonstrate this empirically, we fit a random forest f to the Drug Consumption data and plotted its off- and on-manifold global Shapley values in Fig. 2(c). Next we fit a separate random forest g_S to each coalition S of features, 2^{10} models in total, as in [31]. We used the accuracy $A(g_S)$ of each model – in the sense of Sec. 2.3 – as the value function for an additional Shapley computation:

$$\Phi_g(i) = \sum_{S \subseteq N \setminus i} \frac{s!\,(n-s-1)!}{n!}\,[A(g_{S \cup i}) - A(g_S)] \tag{8}$$

where $\Phi_g(i)$ is directly the average gain in accuracy that results from adding feature i to the set of inputs. These values are labelled "Retrained models" in Fig. 2(c). Note their agreement with the on-manifold explanation of fixed model f. On-manifold Shapley values thus indicate which features in the data are most predictive of the label.

This consistency check allows us to show that Tree SHAP [19, 18] does not provide a method for on-manifold explainability. Observe in Fig. 2(c) that Tree SHAP values roughly track the off-manifold explanation: somewhat larger on the most predictive feature and somewhat smaller on the others. This occurs because trees tend to split on high-predictivity features first, and Tree SHAP privileges early-splitting features in an otherwise off-manifold calculation.

4 Scalable approaches to computing on-manifold Shapley values

For the results of Sec. 3, the on-manifold value function, Eq. (4), was estimated from the empirical data distribution, an approach which is not practical for complex realistic data. In this section, we develop two methods of learning the on-manifold value function: (i) unsupervised learning of the conditional distribution $p(x'|x_S)$, and (ii) a supervised technique to learn the value function directly.

4.1 Unsupervised approach

To take an unsupervised approach to the data manifold, one can learn the conditional distribution $p(x'|x_S)$ that appears in the on-manifold value function. We do this using variational inference and two model components. The first component is a variational autoencoder [14, 24], with encoder $q_\phi(z|x)$ and decoder $p_\theta(x|z)$. The second is a masked encoder, $r_\psi(z|x_S)$, for which the goal is to map the coalition x_S to a distribution in latent space that agrees with the encoder $q_\phi(z|x)$ as well as possible. A model of $p(x'|x_S)$ is then provided by the composition:

$$\hat{p}(x'|x_S) = \int dz\; p_\theta(x'|z)\; r_\psi(z|x_S) \tag{9}$$

and a good fit to the data should maximise $\hat{p}(x'|x_S)$. A lower bound to its log-likelihood is given by

$$\mathcal{L}_0 = \mathbb{E}_{q_\phi(z|x')}\big[\log p_\theta(x'|z)\big] - \mathcal{D}_{\text{KL}}\big(q_\phi(z|x')\,||\,r_\psi(z|x_S)\big)\big] \tag{10}$$

While $\mathcal{L}_0$ could be used on its own as the objective function to learn $\hat{p}(x'|x_S)$, this would leave the variational distribution $q_\phi(z|x)$ unconstrained, at odds with our goal of learning a smooth-manifold structure in latent space. This concern can be mitigated by introducing

$$\mathcal{L}_{\text{reg}} = -\mathcal{D}_{\text{KL}}\big(q_\phi(z|x)\,||\,p(z)\big) \tag{11}$$

which regularises $q_\phi(z|x)$ by penalising differences from a smooth (e.g. unit normal) prior distribution $p(z)$. We thus include $\mathcal{L}_{\text{reg}}$ as a regularisation term in our unsupervised objective: $\mathcal{L} = \mathcal{L}_0 + \beta\,\mathcal{L}_{\text{reg}}$. This objective contains a hyperparameter β that prevents a fair comparison between models trained with different values. A separate metric to judge performance is discussed next.

4.2 Metric for the learnt value function

The unsupervised method of Sec. 4.1 leads to a learnt estimate $\hat{p}(x'|x_S)$ of the conditional distribution, and thus to an estimate of the on-manifold value function: $\hat{v}_{f_y(x)}(S) = \mathbb{E}_{\hat{p}(x'|x_S)}[f_y(x_S \sqcup x'_{\bar{S}})]$. With the goal of judging this estimate, consider the following formal quantity:

$$\text{MSE}(x_S, y) = \mathbb{E}_{p(x'|x_S)}\,\big|f_y(x') - \hat{v}_{f_y(x)}(S)\big|^2 \tag{12}$$

This quantity is minimal with respect to $\hat{v}_{f_y(x)}(S)$ when $\hat{v}_{f_y(x)}(S) = \mathbb{E}_{p(x'|x_S)}[f_y(x')]$, in agreement with the definition, Eq. (4), of the on-manifold value function. We can then quantitatively judge the performance of the unsupervised model $\hat{p}(x'|x_S)$ by computing

$$\text{MSE} = \mathbb{E}_{p(x)}\;\mathbb{E}_{S\sim\text{Shapley}}\;\mathbb{E}_{y\sim\text{Unif}}\,\big|f_y(x) - \hat{v}_{f_y(x)}(S)\big|^2 \tag{13}$$

Note that Eq. (13) is precisely Eq. (12) averaged over coalitions S drawn from the Shapley sum, features x_S drawn from the data, and labels y drawn uniformly over classes. Moreover, the mean-square-error in Eq. (13) is easy to estimate using the empirical distribution $p(x)$ and the learnt model $\hat{p}(x'|x_S)$, thus providing an unambiguous metric to judge the outcome of the unsupervised approach.

4.3 Supervised approach

The MSE metric of Eq. (13) supports a supervised approach to learning the on-manifold value function directly. We do this by defining a surrogate model $g_y(x_S)$ that can operate on coalitions of features x_S (e.g. by masking out-of-coalition features) and that is trained to maximise the objective:

$$\mathcal{L} = \mathbb{E}_{p(x)}\;\mathbb{E}_{S\sim\text{Shapley}}\;\mathbb{E}_{y\sim\text{Unif}}\,\big|f_y(x) - g_y(x_S)\big|^2 \tag{14}$$

As discussed in Sec. 4.2, this objective is maximised as the surrogate model $g_y(x_S)$ approaches the on-manifold value function $\mathbb{E}_{p(x'|x_S)}[f_y(x')]$ of the model-to-be-explained.

4.4 Comparison of approaches

Our implementations of the unsupervised and supervised approaches to on-manifold explainability are summarised in App. A. Both approaches lead to broadly similar results. Fig. 4(a) compares the two techniques on the Drug Consumption data, where explanations are given for the random forest of Sec. 3.4 and compared against the computation using the empirical distribution.

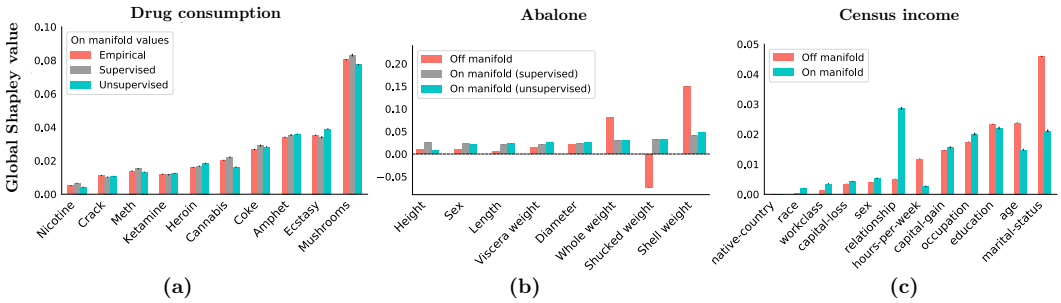


Figure 4: (a) Unsupervised and supervised techniques for on-manifold explainability compared on Drug Consumption data. Global Shapley values for (b) Abalone data and (c) Census Income data.

The unsupervised approach to on-manifold explainability is flexible but untargeted: $p(x'|x_S)$ is data-set-specific but model-agnostic, accommodating explanations for many models trained on the same data. The supervised approach trades flexibility for increased performance: while the technique must be retrained to explain each model, it entails direct minimisation of the MSE.

The supervised method is thus expected to achieve higher accuracy. We confirmed this on all data sets studied in this paper; see Table 1 in App. A for a numerical comparison. The supervised approach also offers increased stability, leading to a smaller variance in MSE in repeated experiments (cf. Table 1). The supervised method is more efficient as well: while the unsupervised technique estimates the value function by sampling from $\hat{p}(x'|x_S)$, the supervised approach learns the value function directly. As a result, to compute Shapley values for the experiments of Sec. 5, the supervised method required sampling roughly 10 times fewer coalitions to match the standard-error of the unsupervised method.

5 Experiments and results

Here we demonstrate the practical utility of on-manifold explainability through experiments. All numerical details, including a description of uncertainties, are given in App. B.

5.1 Abalone data

Global Shapley values represent the portion of a model's accuracy attributable to each feature. To show that staying on manifold is required for this interpretation to be robust, we experimented on Abalone data from the UCI repository [9]. We trained a neural network on the physical characteristics contained in the data to classify abalone as younger than or older than the median age.

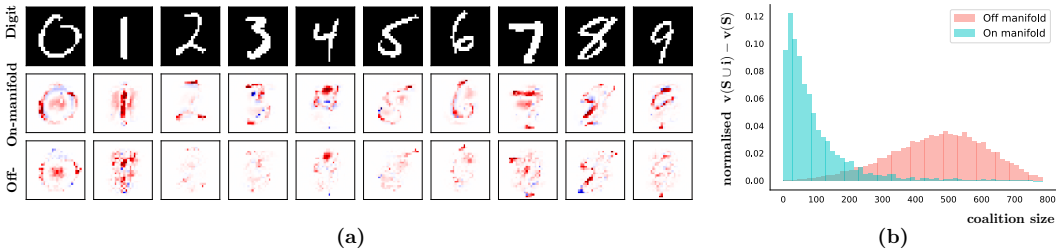
Fig. 4(b) displays global Shapley values for this model. While the supervised and unsupervised techniques lead to broadly similar on-manifold explanations, observe the drastic difference that arises off manifold. This is due to the tight correlations between features in the data (4 different weights and 3 lengths) making the data manifold low-dimensional and important.

Notice further that Fig. 4(b) displays negative global Shapley values off manifold, negating their interpretation as portions of the model accuracy attributable to each feature.

5.2 Census Income data

To demonstrate that on-manifold explanations are consistent with correlations that appear in the data, we experimented on UCI Census Income data [9]. We trained an xgboost classifier [7] to predict whether an individual's income exceeds \$50k based on demographic features in the data.

Fig. 4(c) displays global Shapley values for this model, using the supervised method for the on-manifold explanation. Note the large discrepancy between the off-manifold Shapley values for marital-status and relationship. These features are strongly correlated (married individuals most often have relationship = husband or wife) and their roughly-equal on-manifold values indicate that these features are nearly identically predictive of the model's output.



(a) (b)

Figure 5: (a) Randomly drawn MNIST digits explained on / off manifold. Red / blue pixels indicate positive / negative Shapley values, and the colour scale in each column is fixed. (b) Shapley summand as a function of coalition size – averaged over coalitions, pixels, and the MNIST test set.

Notice further that the Shapley value for age is significantly larger off-manifold than on. This means that the model heavily relies on age to determine its output, but that age correlates with other features, e.g. marital-status and education, that are also predictive of the model's output.

5.3 MNIST

To demonstrate on-manifold explainability on higher-dimensional data, we trained a simple feed-forward network on binary MNIST [16] and explained randomly drawn digits in Fig. 5(a).

Despite having the same sum over pixels – as controlled by Eq. (3) – and explaining the same model prediction, each on-manifold explanation is more concentrated, with more interpretable structure, than its off-manifold counterpart. The handwritten strokes are clearly visible on-manifold, with key off-stroke regions highlighted as well. Off-manifold explanations generally display lower intensities spread less informatively across the digit-region.

These off-manifold explanations are a direct result of splices as in Fig. 1. With such unrealistic input, the model's output is uncontrolled and uninformative. In fact, it is only on very large coalitions of pixels, subject to minimal splicing, that the model can make intelligent predictions off-manifold. This is confirmed in Fig. 5(b), which shows the average Shapley summand as a function of coalition size on MNIST. Note that primarily large coalitions underpin off-manifold explanations, whereas far fewer pixels are required on-manifold, consistent with the low-dimensional manifold underlying the data.

6 Related work

Within the Shapley paradigm, initial work has been done to produce on-manifold explanations: [1] (similar to [34, 11]) explores empirical and distribution-fitting techniques, while [19] takes a tree-specific approach, conditioning out-of-coalition features on in-coalition features appearing earlier in the tree. In contrast to these methods, we compute on-manifold Shapley values with more-scalable methods of learning the data manifold, either through variational inference or supervised learning. Moreover, we show in Fig. 2(c) that Tree SHAP does not remedy the off-manifold problem.

Other on-manifold explainability methods exist as well; see e.g. [6] and [2]. Complementary to our work, these methods apply to images, lie outside the Shapley paradigm, and require generative methods. We focus on general data types, operate within the Shapley framework, and offer a simpler alternative (Sec. 4.3) to generative methods.

7 Conclusion

In this work, we took a careful study of the off-manifold problem in AI explainability. We presented the distinction between on- and off-manifold Shapley values in the conceptually clear setting of tree-based models and low-dimensional data. We then introduced two novel techniques to compute on-manifold Shapley values for any model on any data: one technique learns to impute features on the data manifold, while the other learns the Shapley value function directly. In-so-doing, we provided compelling evidence against the use of off-manifold explainability, and demonstrated that on-manifold Shapley values offer a viable approach to AI explainability in real-world contexts.

Acknowledgements

This work was developed and experiments were run on the Faculty Platform for machine learning. The authors benefited from discussions with Tom Begley, Markus Kunesch, and John Mansir. DDM was partially supported by UCL's Centre for Doctoral Training in Data Intensive Science.

References

[1] J. Aas, M. Jullum, and A. Løland. Explaining individual predictions when features are dependent: More accurate approximations to Shapley values, 2019. [arXiv:1903.10464].

[2] C. Agarwal, D. Schonfeld, and A. Nguyen. Removing input features via a generative model to explain their attributions to classifier's decisions, 2019. [arXiv:1910.04256].

[3] A. Agrawal, J. Gans, and A. Goldfarb. *Prediction Machines: The simple economics of artificial intelligence*. Harvard Business Press, 2018.

[4] D. Baehrens, T. Schroeter, S. Harmeling, M. Kawanabe, K. Hansen, and K.-R. MÃžller. How to explain individual classification decisions. *Journal of Machine Learning Research*, 2010.

[5] L. Breiman. Random forests. *Machine learning*, 2001.

[6] C. Chang, E. Creager, A. Goldenberg, and D. Duvenaud. Explaining image classifiers by counterfactual generation. In *International Conference on Learning Representations*, 2019.

[7] T. Chen and C. Guestrin. Xgboost: A scalable tree boosting system. In *International Conference on Knowledge Discovery and Data Mining*, 2016.

[8] A. Datta, S. Sen, and Y. Zick. Algorithmic transparency via quantitative input influence: Theory and experiments with learning systems. In *IEEE Symposium on Security and Privacy*, 2016.

[9] D. Dua and C. Graff. UCI machine learning repository, 2017. [archive.ics.uci.edu/ml].

[10] I. J. Goodfellow, J. Shlens, and C. Szegedy. Explaining and harnessing adversarial examples. In *International Conference on Learning Representations*, 2015.

[11] J. Gu and V. Tresp. Contextual prediction difference analysis, 2019. [arXiv:1910.09086].

[12] G. Hooker and L. Mentch. Please stop permuting features: An explanation and alternatives, 2019. [arXiv:1905.03151].

[13] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations*, 2015.

[14] D. P. Kingma and M. Welling. Auto-encoding variational bayes. In *International Conference on Learning Representations*, 2014.

[15] I. Kononenko et al. An efficient explanation of individual classifications using game theory. *Journal of Machine Learning Research*, 2010.

[16] Y. LeCun and C. Cortes. MNIST database, 2010. [yann.lecun.com/exdb/mnist].

[17] S. Lipovetsky and M. Conklin. Analysis of regression in game theory approach. *Applied Stochastic Models in Business and Industry*, 2001.

[18] S. M. Lundberg, G. Erion, H. Chen, A. DeGrave, J. Prutkin, B. Nair, R. Katz, J. Himmelfarb, N. Bansal, and S. Lee. From local explanations to global understanding with explainable AI for trees. *Nature Machine Intelligence*, 2020.

[19] S. M. Lundberg, G. G. Erion, and S.-I. Lee. Consistent individualized feature attribution for tree ensembles, 2018. [arXiv:1802.03888].

[20] S. M. Lundberg and S.-I. Lee. A unified approach to interpreting model predictions. In *Advances in Neural Information Processing Systems*, 2017.

[21] M. Mase, A. B. Owen, and B. Seiler. Explaining black box decisions by Shapley cohort refinement, 2019. [arXiv:1911.00467].

[22] A. Nguyen, J. Yosinski, and J. Clune. Deep neural networks are easily fooled: High confidence predictions for unrecognizable images. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2015.

[23] P. Rasouli and I. C. Yu. Meaningful data sampling for a faithful local explanation method. In *Intelligent Data Engineering and Automated Learning*, 2019.

[24] D. J. Rezende, S. Mohamed, and D. Wierstra. Stochastic backpropagation and approximate inference in deep generative models. In *International Conference on Machine Learning*, 2014.

[25] M. T. Ribeiro, S. Singh, and C. Guestrin. Why should I trust you: Explaining the predictions of any classifier. In *International Conference on Knowledge Discovery and Data Mining*, 2016.

[26] C. Rudin. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature Machine Intelligence*, 2019.

[27] L. S. Shapley. A value for n-person games. In *Contribution to the theory of games*, 1953.

[28] A. Shrikumar, P. Greenside, and A. Kundaje. Learning important features through propagating activation differences. In *International Conference on Machine Learning*, 2017.

[29] C. Strobl, A.-L. Boulesteix, T. Kneib, T. Augustin, and A. Zeileis. Conditional variable importance for random forests. *BMC Bioinformatics*, 2008.

[30] E. Štrumbelj and I. Kononenko. Explaining prediction models and individual predictions with feature contributions. *Knowledge and information systems*, 2014.

[31] E. Štrumbelj, I. Kononenko, and M. Robnik-Sikonja. Explaining instance classifications with interactions of subsets of feature values. *Data Knowl. Eng.*, 2009.

[32] M. Sundararajan and A. Najmi. The many Shapley values for model explanation, 2019. `[arXiv:1908.08474]`.

[33] J. von Neumann and O. Morgenstern. *Theory of Games and Economic Behavior*. Princeton University Press, 1944.

[34] L. M. Zintgraf, T. S. Cohen, T. Adel, and M. Welling. Visualizing deep neural network decisions: Prediction difference analysis. In *International Conference on Learning Representations*, 2017.

Table 1: Performance and stability, with respect to MSE, of supervised and unsupervised approaches to on-manifold explainability.

DATA SET	SUPERVISED	UNSUPERVISED
DRUG	0.0441 ± 0.0002	0.0536 ± 0.0007
ABALONE	0.0200 ± 0.0001	0.0293 ± 0.0009
CENSUS	0.02496 ± 0.00008	0.0300 ± 0.0006
MNIST	0.0121 ± 0.0001	0.0257 ± 0.0005

A Implementation details

For the unsupervised approach, we modelled the encoder $q_\phi(z|x)$ as a diagonal normal distribution with mean and variance determined by a neural network:

$$q_\phi(z|x) = \mathcal{N}\big(\mu_\phi(x), \sigma_\phi(x)\big) \tag{15}$$

We modelled the decoder $p_\theta(x|z)$ as a product distribution:

$$p_\theta(x|z) = \prod_i p_\theta(x_i|z) \tag{16}$$

where the distribution type (e.g. normal, categorical) of each x_i is chosen per-data-set and each distribution's parameters are determined by a shared neural network. We modelled the masked encoder $r_\psi(z|x_S)$ as a Gaussian mixture:

$$r_\psi(z|x_S) = \sum_j w_\phi^{(j)}(x)\, \mathcal{N}\Big(\mu_\phi^{(j)}(x), \sigma_\phi^{(j)}(x)\Big) \tag{17}$$

To allow $r_\psi(z|x_S)$ to accept variable-size coalitions x_S as input, we simply masked out-of-coalition features with a special value (-1) that never appears in the data.

The unsupervised method has several hyperparameters: β which multiplies the regularisation term in Eq. (11), the number of components in Eq. (17), as well the architecture and optimisation of the networks involved. For each experiment in this paper, we tuned hyperparameters to minimise the MSE of Eq. (13) on a held-out validation set; see App. B for numerical details.

For the supervised approach, we modelled $g_y(x_S)$ using a neural network, again masking out-of-coalition features with -1 to accommodate variable-size coalitions x_S. This method's hyperparameters, relating to architecture and optimisation, were similarly tuned to minimise validation-set MSE; see App. B for details.

As discussed in Sec. 4.4, the supervised method is expected to achieve a smaller MSE than the supervised approach. We confirmed this on all data sets studied in this paper; see Table 1 for a numerical comparison. In the table, central values indicate the test-set MSE achieved by each method. The uncertainties represent the standard deviation in test-set MSE upon re-training each method with fixed hyperparameters 10 times. This indicates that the supervised method also offers increased stability over the unsupervised approach.

B Details of experiments

Here we provide numerical details for all experiments presented in the paper.

B.1 Drug Consumption experiment

Several experiments were performed on the Drug Consumption data from the UCI repository [9]. We used 10 binary features from the data set – Mushrooms, Ecstasy, etc., as displayed in Fig. 2 – to predict whether individuals had ever consumed an 11th drug: LSD.

The Shapley values in Fig. 2(a) and Fig. 2(b) describe a single decision tree fit with default sklearn parameters as well as max_depth = 1 and max_features = None. While the data exhibits a $57:43$ class balance, the decision tree achieves 82.0% accuracy on a held-out test set.

Local off-manifold Shapley values in Fig. 2(a) were computed by Monte Carlo sampling permutations to estimate Eq. (1). For each sampled permutation, a random data point x' was drawn from the test set to estimate the off-manifold value function of Eq. (2). Bar heights in Fig. 2(a) are the means that resulted from 10^6 Monte Carlo samples per feature. Throughout the paper, error bars represent standard errors of the means.

Local on-manifold Shapley values in Fig. 2(a) were again computed using 10^6 Monte Carlo samples of Eq. (1), but this time using the on-manifold value function of Eq. (4). For each sampled coalition x_S, a random data point x' was drawn from the test set, with the crucial requirement that $x'_S = x_S$. In the text, we refer to this as empirically estimating the conditional distribution $p(x'|x_S)$. Such empirical estimation is only possible because this data set has a small number of all-binary features.

Global Shapley values in Fig. 2(b) were similarly computed using 10^6 Monte Carlo samples of Eq. (5). For each labelled data point (x, y) sampled from the test set, a single permutation was drawn to estimate Eq. (1) and a single data point x' was drawn to estimate the value function.

The Shapley values of Fig. 2(c) describe a random forest fit with default sklearn parameters and max_features = None, which achieves 82.2% test-set accuracy. Global off- and on-manifold Shapley values were computed just as in Fig. 2(b). Tree SHAP values were computed with the SHAP package [20] with model_output = margin and feature_perturbation = tree_path_dependent.

The values labelled "Model retraining" in Fig. 2(c) were computed by fitting a separate random forest g_S for each coalition S of features in the data set: 2^{10} models in all. We used these models to compute the sum of Eq. (8), where $A(g_S)$ represents a variant of model g_S's accuracy: it is the accuracy achieved if one predicts labels by drawing stochastically from g_S's predicted probability distribution (as opposed to deterministically drawing the maximum-probability class).

The global on-manifold Shapley values in Fig. 2(c) appear in Fig. 4(a) as well, labelled "Empirical". Fig. 4(a) also displays on-manifold Shapley values computed using the supervised and unsupervised methods introduced in this paper. As above, these are Monte Carlo estimates of Eq. (5). The supervised method involved training a fully-connected network on the MSE loss of Eq. (14). All neural networks in this paper used 2 flat hidden layers, Adam [13] for optimisation, and a batch size of 256. We scanned over a grid with

$$\begin{aligned} \text{hidden layer size} &= \{128,\ 256,\ 512\} \\ \text{learning rate} &= \{10^{-3}, 10^{-4}\} \end{aligned} \tag{18}$$

choosing the point with minimal MSE on a held-out validation set after 10k epochs of training; see Table 2. Each supervised value in Fig. 4(a) corresponds to 10^4 Monte Carlo samples.

The unsupervised method involved training a variational autoencoder to minimise the loss of Sec. 4.1, as described in App. A. The encoder, decoder, and masked encoder were each modelled using fully-connected networks, trained using early stopping, with patience 100. We scanned over a grid of hidden layer sizes and learning rates as in Eq. (18) as well as

$$\begin{aligned} \text{latent dimension} &= \{2,\ 4,\ 8,\ 16\} \\ \text{latent modes} &= \{1, 2\} \\ \text{regularisation } \beta &= \{0.05,\ 0.1,\ 0.5,\ 1\} \end{aligned} \tag{19}$$

choosing the point with minimal validation-set MSE; see Table 2. Unsupervised values in Fig. 4(a) correspond to 10^6 Monte Carlo samples.

B.2 Abalone experiment

For the experiment of Sec. 5.1, we used the Abalone data set from the UCI repository [9]. The data contains 8 features corresponding to physical measurements (see Fig. 4b) which we used to classify abalone as younger than or older than the median age. We trained a neural network to perform this task – with hidden layer size 100, default sklearn parameters, and early stopping – obtaining a test-set accuracy of 78%.

Shapley values in Fig. 4(b) were computed exactly as described in Sec. B.1, except that the supervised method involved training for 5k epochs. Optimised hyperparameters are given in Table 2.

Table 2: Optimal hyperparameters found for computing on-manifold Shapley values.

DATA SET	METHOD	HIDDEN DIM.	LEARN. RATE	LATENT DIM.	MODES	β
DRUG	SUPERVISED	512	10^{-3}			
	UNSUPERVISED	128	10^{-3}	4	1	0.5
ABALONE	SUPERVISED	512	10^{-3}			
	UNSUPERVISED	256	10^{-3}	2	1	0.05
CENSUS	SUPERVISED	512	10^{-3}			
	UNSUPERVISED	128	10^{-3}	8	1	1
MNIST	SUPERVISED	512	10^{-4}			
	UNSUPERVISED	512	10^{-4}	16	1	1

B.3 Census Income experiment

For the experiment of Sec. 5.2, we used the Census Income data set from the UCI repository [9]. The data contains 49k individuals from the 1994 US Census, as well as 13 features (see Fig. 4c) which we used to predict whether annual income exceeded \$50k. We trained an xgboost classifier [7] with default parameters, achieving a test-set accuracy of 85% amidst a $76 : 24$ class balance.

Shapley values in Fig. 4(c) were computed exactly as described in Sec. B.1, except that the supervised method used 5k epochs, and the unsupervised method used patience 50. Optimised hyperparameters are given in Table 2. The on-manifold values in Fig. 4(c) were computed using the supervised method. While the unsupervised method does not appear in the figure, it was performed to complete Table 1.

B.4 MNIST experiment

In Sec. 5.3, we used binary MNIST [16]. We trained a fully-connected network – with hidden layer size 512, default parameters, and early stopping – achieving 98% test-set accuracy.

The digits in Fig. 5(a) were randomly drawn from the test set. Shapley values in Fig. 5(a) were computed exactly as described in Sec. B.1, except that the supervised method involved training for 2k epochs, and the on-manifold explanations are based on 16k Monte Carlo samples per pixel. Optimised hyperparameters are given in Table 2. The on-manifold explanations in Fig. 5(a) were computed using the supervised method. While the unsupervised method does not appear in the figure, it was performed to complete Table 1.

The average uncertainty, which is not shown in Fig. 5(a), is roughly 0.002 – stated as a fraction of the maximum Shapley value in each image.