CAN WE FAITHFULLY REPRESENT ABSENCE STATES TO COMPUTE SHAPLEY VALUES ON A DNN?

Anonymous authors
Paper under double-blind review

ABSTRACT

Although many methods have been proposed to estimate attributions of input variables, there still exists a significant theoretical flaw in masking-based attribution methods, *i.e.* it is hard to examine whether the masking method faithfully represents the absence of input variables. Specifically, for the masking-based attributions, setting an input variable to the baseline value is a typical way of representing the absence of the variable. However, there are no studies investigating how to represent the absence of input variables and verify the faithfulness of baseline values. Therefore, we revisit the feature representation of a DNN in terms of causality, and propose to use causal patterns to examine whether the masking method faithfully removes information encoded in input variables. More crucially, it is proven that the causality can be explained as elementary rationale of the Shapley value. Furthermore, we define the optimal baseline value from the perspective of causality, and we propose a method to learn the optimal baseline value. Experimental results have demonstrated the effectiveness of our method.

1 INTRODUCTION

Many attribution methods (Zhou et al., 2016; Selvaraju et al., 2017; Lundberg and Lee, 2017) have been proposed to estimatie the attribution of input variables to the model output, which represents an important direction in explainable AI. In this direction, many methods estimated attributions of input variables based on masking some input variables. For example, the Shapley value (Shapley, 1953) of an input variable can be considered as the average marginal utility of masking the variable in different contexts. To this end, Dabkowski and Gal (2017); Ancona et al. (2019); Covert et al. (2020b) proposed different ways to compute baseline values to represent the masked state of input variables, as shown in Fig. 1.

Theoretically, the trustworthiness of attribution methods highly depends on whether the current baseline value can really remove the information of the input variable without bring in new out-of-distribution (OOD) features. However, there is no criterion to evaluate the information removal of masking methods. To this end, we need to first break the ***blind faith*** that seemingly reasonable baseline values can faithfully represent the absence of input variables, and seemingly OOD baseline values definitely cause abnormal features. In fact, because different DNNs have different inference logic, seemingly OOD baseline values are not necessarily equivalent to generating OOD features.

Fortunately, Ren et al. (2021a) have provided a new perspective to analyze feature representation of a DNN. In fact, it has been widely believed that the learning of DNN is a fitting problem, instead of explicitly formulating causality like how graphical models do. However, Ren et al. (2021a); Deng et al. (2022) have surprisingly discovered that when the DNN was sufficiently trained, the sparse and symbolic AND relationships between input variables emerge. Different AND relationships act like different causal patterns and construct a sparse causal graph to explain the DNN.

As Fig. 1 shows, the first layer of the causal graph represents input variables. The second layer contains many causal patterns, each representing a specific AND interaction between input variables. For example, the pattern of a bird head consists of the *forehead*, *eyes*, and *beak*, *i.e.,* S={*forehead, eyes, beak*}=$\{f, e, b\}$ for short. Only if input variables f, e, and b co-appear, the causal pattern S is triggered; otherwise not. The third layer is the output of the causal graph. Each triggered causal pattern S has a numerical contribution I_S to the output, which is termed its *causal effect*.

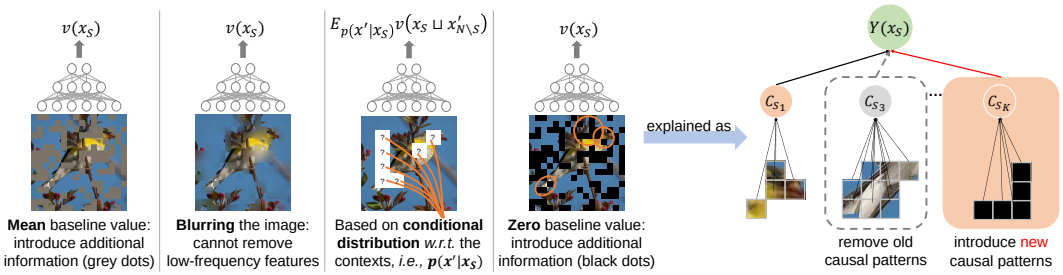


Figure 1: (Left) Previous masking methods may either introduce additional information, or cannot remove all the old information. (Right) The inference of the DNN on images masked by these baseline values can be explained as a three-layer causal graph.

Therefore, we can consider causal patterns in the second layer as a set of interactive concepts memorized by the DNN, and the causal graph can use the triggering of causal patterns to explain the DNN. It is because given each input sample with n variables, we can generate 2^n different masked samples. *It is proven that we can use a sparse causal graph with a relatively small number of causal patterns to accurately mimic the network output on all 2^n masked samples.*

From this perspective, whether baseline values look reasonable is no longer the key factor to determine the trustworthiness of baseline values. Instead, we check whether setting an input variable to its baseline value can remove all potential AND relationships related to this variable without creating new causal patterns. It is because setting incorrect baseline values let partial effects of the AND relationship be mistakenly explained as new causal patterns without being removed.

• For example, let the AND pattern $S=\{f,e,b\}$ contribute a score E to the output. If the baseline value b_f of the variable f just blur the f (*forehead*), rather than fully remove its appearance, then partial effects of E is remained not explained by the causal effect $I_{\{f,e,b\}}$, *i.e.*, masking the *forehead* cannot remove all score E. The remaining score $E-I_{\{f,e,b\}}$ will be explained as $I_{\{e,b\}}$. Thus, masking *forehead* and deactivating the effect $I_{\{f,e,b\}}$ cannot remove all effects E of the AND relationship.

• If baseline values of $\{f,e,b\}$ are set all black regions, then masking all the three regions may generate a new pattern of a black line, as a new causal pattern that influences the network output.

Therefore, we consider the optimal baseline values, which faithfully reflect the true inference logic, usually most reduce the overall strength of existing causal effects, without introducing new causal effects. **However, we find that most existing masking methods are not satisfactory from this perspective (see Section 2.1 and Table 1). The masking method based on conditional distribution (Covert et al., 2020b; Frye et al., 2021) performs a bit better but still has problems.**

In particular, we notice that Shapley values can be derived from effects of causal patterns in theory, *i.e.,* the causal patterns are proven to be elementary effects of Shapley values. Therefore, we propose a new method to learn optimal baseline values for Shapley values, which is verified to be effective in removing original causal effects of absent input variables while not introducing new causal effects.

Contributions of this paper can be summarized as follows. (1) We propose a metric to examine whether the masking approach in attribution methods could faithfully represent the absence state of input variables. Based on this metric, we find that most previous masking methods are not reliable. (2) We define and develop an approach to estimating optimal baseline values for Shapley values, which ensures the trustworthiness of the attribution.

2 Problems with the representation of the masked states

The Shapley value (Shapley, 1953) was first introduced in game theory, which measures the contribution of each player in a game. Actually, given an input $\boldsymbol{x}$ with n input variables, *i.e.,* $\boldsymbol{x}=[x_1,\ldots,x_n]$, we can consider a deep model as a game with n players $N=\{1,2,\cdots,n\}$. Each player i is an input variable x_i (*e.g.* a input dimension, a pixel, or a word). In this way, the problem of *fairly estimating attributions of input variables in the DNN* is equivalent to the problem of *fairly assigning the total reward in the game to each player*. To this end, the Shapley value is considered as the unique approach that satisfies the linearity, dummy, symmetry, and efficiency axioms (please refer to Appendix B), and it fairly allocates attributions to input variables (Weber, 1988). The Shapley value of the i-th input variable ϕ_i is defined as follows.

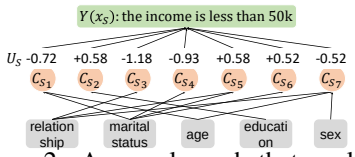


Figure 2: A causal graph that explains the inference on a sample in the income dataset.

Table 1: The ratio R of the remained and newly introduced causal effects in the masked inputs. A small value of R meant that baseline values removed most original causal effects and did not introduce many new effects.

	$R^{(\text{zero})}$	$R^{(\text{mean})}$	$R^{(\text{blur})}$	$R^{(\text{conditional})}$	$R^{(\text{ours})}$
MNIST	1.1736	0.3043	0.4159	0.3780	**0.2185**
CIFAR-10	0.6630	0.8042	0.7288	0.4027	**0.1211**

$$\phi_i = \sum\nolimits_{S\subseteq N\setminus\{i\}} |S|!(n-|S|-1)!/n! \cdot \big[v(\boldsymbol{x}_{S\cup\{i\}}) - v(\boldsymbol{x}_S)\big] \tag{1}$$

where $v(\boldsymbol{x}_S)\in\mathbb{R}$ denotes the model output when variables in S are present and variables in $N\setminus S$ are absent. Specifically, $v(\boldsymbol{x}_\emptyset)$ represents the model output when an empty image is given. The Shapley value of the variable i is computed as the weighted marginal contribution of i when the variable i is present *w.r.t.* the case when the variable i is absent, *i.e.* $v(\boldsymbol{x}_{S\cup\{i\}}) - v(\boldsymbol{x}_S)$.

Originally, the Shapley value is defined in game theory, in which the presence and absence of a player can be defined without ambiguity. However, when we explain a DNN, a core challenge is how to faithfully define the absence of an input variable. Although we admit some alternative methods (Covert et al., 2020b; Frye et al., 2021) compute $v(\boldsymbol{x}_S)$ based on the conditional distribution, the most classical way is to use ***baseline values*** (or called *reference values*) $\boldsymbol{b} = [b_1, b_2, \ldots, b_n]$ to mask variables to represent their absence. Specifically, we construct the following masked sample $\boldsymbol{x}_S$ based on $\boldsymbol{x}$, where only variables in S are present, and variables not in S are masked.

$$\text{If } i\in S,\ (\boldsymbol{x}_S)_i = x_i;\ \text{ otherwise, } (\boldsymbol{x}_S)_i = b_i \tag{2}$$

We aim to learn optimal baseline values $\boldsymbol{b}$ to faithfully represent absent states of input variables.[1]

Representing a DNN using the causal graph. Given a trained DNN v and an input sample $\boldsymbol{x}$, it has been proven by Ren et al. (2021a) that the three-layer causal graph in Fig. 1 and Fig. 2 can faithfully explain the inference logic of the trained DNN. The first layer contains n source nodes, and each node is an indicator variable $A_i\in\{0,1\}$ to present the presence of the variable x_i ($A_i=1$) and the absence of x_i ($A_i=0$). Nodes in the second layer are indicator variables C_S that represent the AND relationship between input variables in the causal pattern S. For example, the head pattern of a bird consists of $\{$*forehead*, *eyes*, *beak*$\}$. When the three parts co-appear, the head pattern $S=\{$*forehead, eyes, beak*$\}$ is triggered and $C_S=1$; otherwise, $C_S=0$. In the third layer, the single sink node Y denotes the output of the causal graph. Thus, the transition probability of the causal graph is given as follows.

$$P(C_S=1|A_1,A_2,\ldots,A_n) = \prod\nolimits_{k\in S} A_k,\ \ P(Y|\{C_S|S\in\Omega\}) = \mathbb{1}(Y=\sum\nolimits_{S\in\Omega} C_S\cdot U_S) \tag{3}$$

where $P(C_S=0|A_1,A_2,\ldots,A_n) = 1-P(C_S=1|A_1,A_2,\ldots,A_n)$. Ω denotes the set of all causal patterns. $\mathbb{1}(\cdot)$ denotes the indicator function. U_S denotes the effects of the causal pattern S on the output Y. According to Eq. (3), the causal relationship between C_S and the output Y can be rewritten as the following structural causal model (SCM) (Neuberg, 2003), *i.e.*, the output Y of the causal graph sums up causal effects of all the triggered causal patterns.

$$Y(\boldsymbol{x}) = \sum\nolimits_{S\in\Omega} I_S(\boldsymbol{x}),\ \ I_S(\boldsymbol{x}) = U_S\cdot C_S(\boldsymbol{x}) = U_S\cdot\prod\nolimits_{i\in S} A_i(\boldsymbol{x}) \tag{4}$$

where $I_S(\boldsymbol{x})$ denotes the actual causal effect of the causal pattern S. In implementations, the causal effect U_S is computed as Harsanyi dividend (Harsanyi, 1982), *i.e.*, $U_S = \sum_{S'\subseteq S}(-1)^{|S|-|S'|}\cdot v(\boldsymbol{x}_{S'})$. According to the definition of the Harsanyi dividend, U_S measures the pure numerical contribution (causal effect) of the interaction between all variables in S to the DNN output. But for each smaller subset $L\subsetneq S$, the interaction between variables in L is measured by U_L, rather than U_S.

Therefore, each causal pattern represents an AND relationship between input variables, and can be considered as an interactive concept memorized by the DNN. In the aforementioned example, if the *forehead*, *eyes*, and *beak* of the bird co-appear, then the head pattern $S=\{$*forehead*, *eyes*, *beak*$\}$ is triggered and makes a causal effect $I_S=U_S$ on the output Y. Otherwise, if any part is masked, the causal pattern S will not be triggered, *i.e.*, $C_S=0$, according to Eq. (4). Then, the DNN's inference score $Y(\boldsymbol{x})$ will not receive the causal effect U_S, and $I_S=C_S=0$. Note that the causal pattern $\{$*eyes*, *beak*$\}$ is not deactivated by the absence of *forehead*, because this pattern represents the AND relationship between *eyes* and *beak*, and it does not contain *forehead*.

Because different input variables interact with each other to form AND relationships, when we mask an input variable i, it is supposed to remove not only independent effect of the variable i, but also all

[1] In Section 4, we tested the proposed method under different initializations of baseline values.

AND relationship related to the variable i. To this end, Appendix E proves that masking the variable i will remove all causal effects of causal patterns containing this variable i.

As both proved in Theorem 3 and discussed in the introduction, an incorrect baseline value b_i will make partial effects of the AND relationship of the variable i be mistakenly explained as effects of redundant patterns without being removed, thereby complicating the explanation. Therefore, *the optimal baseline value is supposed to generate the simplest causal graph explanation. Compared to the causal graph generated by sub-optimal baseline values, the simplest causal graph removes as much as existing causal effects without introducing additional causal effects.*

Theorem 1 *(**Faithfulness**, proven by Ren et al. (2021a) and Appendix D.1) Let us consider a DNN v and a specific sample $\boldsymbol{x}$ with n input variables. Given any of the 2^n arbitrarily masked input $\boldsymbol{x}_S$, the causal graph defined in Eq. (4) can always well mimic the output of the DNN,* i.e., $\forall S \subseteq N, Y(\boldsymbol{x}_S) = v(\boldsymbol{x}_S)$, *when we set the causal patterns in the causal graph are* $\Omega = 2^N = \{S | S \subseteq N\}$.

Remark 1 *(**Sparsity**) Empirically, the causal graph to explain the DNN is actually very sparse. Most nodes $S \in 2^N$ have almost zero causal effects $U_S \approx 0$. A subgraph of the causal graph with a few nodes (usually less than 100 nodes with salient causal effects in most cases) are already enough to approximate the DNN,* i.e., $\forall S \subseteq N, Y(\boldsymbol{x}_S) \approx v(\boldsymbol{x}_S)$.

Remark 1 has been verified on different DNNs learned for various tasks by experiments in both Appendix F.2 and (Ren et al., 2021a). Theorem 1 and Remark 1 prove that a sparse causal graph can faithfully explain the DNN. Theoretically, given an input $\boldsymbol{x}$ with n variables, we can construct 2^n different masked samples $\boldsymbol{x}_S$ *w.r.t.* all subsets $S \subseteq N$. The causal graph can accurately mimic diverse outputs of the DNN on all 2^n masked samples. Therefore, the causal graph is considered to faithfully explain the DNN's inference.

2.1 Problems with previous masking methods

We consider that nodes with salient effects as interactive concepts memorized by the DNN, and use such salient concepts to evaluate the quality of masking methods. Specifically, we examine *whether baseline values remove most causal effects depending on x_i in the causal graph, and whether baseline values generate new causal effects to the causal graph.*

Theoretical support from three perspectives. The evaluation of the masking methods based on salient causal effects is theoretically supported from the following three perspectives. First, according to Theorem 1 and Remark 1, the diverse inference on all 2^n masked samples can be well mimicked by a few salient nodes (usually less than 100 nodes with large $|I_S|$ values) in the causal graph. Such a phenomenon has been illustrated in Appendix F.2 and (Ren et al., 2021a). Masking input variables variables will remove causal effects of the related causal nodes in the causal graph.

Theorem 2 *(proven by Harsanyi (1982) and Appendix D.2) We can directly derive Shapley values from the effects I_S of causal patterns. The Shapley value can be considered as uniformly allocating each causal pattern S's effect I_S to all its variables,* i.e. $\phi_i = \sum_{S \subseteq N \setminus \{i\}} \frac{1}{|S|+1} I_{S \cup \{i\}}$.

Second, Theorem 2 shows that Shapley values are a re-allocation of causal effects to input variables. **Reducing effects of salient patterns means removing elementary factors of Shapley values.**

Besides, in order to verify that the reduction of causal patterns can really represent the absence of input variables, we have conducted experiments to find that *salient patterns triggered by white noise inputs were much less than those triggered by normal images.* Please see Appendix F.3 for details.

Remark 2 *(proof in Appendix D.3) Let us consider a function with a single causal pattern $f(\boldsymbol{x}_S) = w_S \prod_{j \in S}(x_j - \delta_j)$. Accordingly, ground-truth baseline values of variables are obviously $\{\delta_j\}$, because setting any variable $\forall j \in S$, $x_j = \delta_j$ will deactivate this pattern. Given the correct baseline values $b_j^* = \delta_j$, we can use a single causal pattern to regress $f(\boldsymbol{x}_S)$,* i.e., $I_S = f(\boldsymbol{x}_S)$, $\forall\, S' \neq S, I_{S'} = 0$.

Theorem 3 *(proof in Appendix D.3) For the function $f(\boldsymbol{x}_S) = w_S \prod_{j \in S}(x_j - \delta_j)$, if we use m' incorrect baseline values $\{b'_j | b'_j \neq \delta_j\}$ to replace correct ones to compute causal effects, then the function will be explained to contain at most $2^{m'}$ causal patterns.*

Theorem 4 *(proof in Appendix D.3) If we use m' incorrect baseline values to compute causal effects in the function $f(\boldsymbol{x}_S) = w_S \prod_{j \in S}(x_j - \delta_j)$, a total of $\binom{m'}{k-|S|+m'}$ causal patterns of the k-th order emerge, $k \geq |S| - m'$. A causal pattern of the k-th order means that this causal pattern represents the AND relationship between k variables.*

Table 2: Analysis about previous masking methods.

Setting of baseline values	Baseline values are constant or not	Different samples share the same baseline values or not	Shortcomings
Mean baseline value	✓	✓	introduce additional information
Zero baseline value	✓	✓	introduce additional information
Blurring baseline value	✓	✗	not remove low-frequency components
Marginal distribution	✗	✓	assume feature independence
Conditional distribution	✗	✗	balance between the computational cost and the accuracy
Ours	✓	both are ok	high computational cost

Third, Remark 2, Theorems 3 and 4 provide a new perspective to understand how incorrect baseline values generate new causal patterns. Remark 2 proves that correct baseline values faithfully explain the simplest causality in the model. **Theorems 3 and 4 show that incorrect baseline values will use an exponential number of redundant low-order patterns to explain a single high-order pattern.** For example, we are given the function $f(\boldsymbol{x}) = w(x_p - \delta_p)(x_q - \delta_q)$ *s.t.* $x_p = 3, x_q = 4, \delta_p = 2, \delta_q = 3$. If we use ground-truth baseline values $\{\delta_p, \delta_q\}$, then the function is explained as simple as a single causal pattern $\Omega = \{\{p, q\}\}$, which yields correct Shapley values $\phi_p = \phi_q = 0.5 \cdot w$, according to Theorem 2. Otherwise, if we use incorrect baseline values $\{b'_p = 1, b'_q = 1\}$, then this function will be explained as four causal patterns $\Omega = \{\emptyset, \{p\}, \{q\}, \{p, q\}\}$, *i.e.,* $f(\boldsymbol{x}) = U_\emptyset C_\emptyset + U_{\{p\}} C_{\{p\}} + U_{\{q\}} C_{\{q\}} + U_{\{p,q\}} C_{\{p,q\}}$, where $U_\emptyset = 2w$, $U_{\{p\}} = -4w$, $U_{\{q\}} = -3w$, and $U_{\{p,q\}} = 6w$ are computed using incorrect baseline values. Incorrect baseline values increase complicate causal patterns and lead to incorrect Shapley values $\phi_p = -w, \phi_q = 0$. In fact, the existence of most newly introduced causal patterns is due to that the effects of a high-order causal pattern are not fully removed, besides OOD causal patterns (new OOD edges or shapes) caused by baseline values.

Evaluating previous baseline values. Therefore, we can compare causal patterns in the masked sample with causal patterns in the original sample to evaluate the following baseline values.

(1) *Mean baseline values.* As Fig. 1 shows, the baseline value of each input variable is set to the mean value of this variable over all samples (Dabkowski and Gal, 2017), *i.e.* $b_i = \mathbb{E}_x[x_i]$. However, empirically, this method actually introduces additional information to the input. For example, mean values introduce massive grey dots to images and may form new edges as abnormal causal patterns, which has been verified by experiments in Table 1. Experimental details are introduced later.

(2) *Zero baseline values.* Baseline values of all input variables are set to zero (Ancona et al., 2019; Sundararajan et al., 2017), *i.e.* $\forall i \in N,\ b_i = 0$. As Fig. 1 shows, just like mean baseline values, zero baseline values also introduce additional information (black dots) to the input (verified in Table 1).

(3) *Blurring input samples.* Fong and Vedaldi (2017) and Fong et al. (2019) blur image pixels x_i using a Gaussian kernel as its masked state. Covert et al. (2020a); Sturmfels et al. (2020) mentioned that this approach only removed high-frequency signals, but failed to remove low-frequency signals.

(4) *For each input variable, determining a different baseline value for each specific context S.* Instead of fixing baseline values as constants, some studies use varying baseline values to compute $v(\boldsymbol{x}_S)$ given $\boldsymbol{x}$, which are determined temporarily by the context S in $\boldsymbol{x}$. Some methods (Frye et al., 2021; Covert et al., 2020b) define $v(\boldsymbol{x}_S)$ by modeling the conditional distribution of variable values in $N \setminus S$ given the context S, *i.e.* $v(\boldsymbol{x}_S) = \mathbb{E}_{p(\boldsymbol{x}'|\boldsymbol{x}_S)}[\mathit{model}(\boldsymbol{x}_S \sqcup \boldsymbol{x}'_{N\setminus S})]$. The operation $\sqcup$ means the concatenation of $\boldsymbol{x}$'s dimensions in S and $\boldsymbol{x}'$'s dimensions in $N \setminus S$. By assuming the independence between input variables, the above conditional baseline values can be simplified to marginal baseline values (Lundberg and Lee, 2017), *i.e.* $v(\boldsymbol{x}_S) = \mathbb{E}_{p(\boldsymbol{x}')}[\mathit{model}(\boldsymbol{x}_S \sqcup \boldsymbol{x}'_{N\setminus S})]$.

We conducted experiments to examine whether the above baseline values remove all causal patterns in the original input and whether baseline values introduce new causal patterns. We used the metric $R = \mathbb{E}_{\boldsymbol{x}}\left[(\sum_{S\subseteq N} |I'_S|) - \sum_{S\subseteq N} |I_S^{(\text{noise})}|)/(\sum_{S\subseteq N} |I_S|)\right]$ to evaluate the quality of masking. We generated a set of samples based on $\boldsymbol{x}$, where a random set of input variables were masked, and I'_S denote the causal effect in such masked samples. I_S denote the causal effect in the original sample $\boldsymbol{x}$, which was used for normalization. $I_S^{(\text{noise})}$ denote the causal effect in a white noise input, and it represents unavoidable effects of huge amounts of noise patterns. Thus, we considered the $I_S^{(\text{noise})}$ term as an inevitable anchor value and removed it from R for more convincing evaluation. The masking method would have two kinds of effects on causal patterns. (1) We hoped to remove all existing salient patterns in the original sample. (2) We did not expect the masking method to introduce new salient patterns. Interestingly, the removal of existing salient patterns decreased the R value, while the triggering of new patterns increased the R value. Thus, the R metric reflected the both effects. A small value of R indicated a good setting of baseline values.

We used 20 images in the MNIST dataset (LeCun et al., 1998) and 20 images in the CIFAR-10 dataset (Krizhevsky et al., 2009) to compute R, respectively. We split each image into 8×8 grids. We randomly masked some grids using the zero baseline, mean baseline, blur baseline, and the baseline based on conditional distribution, and computed the metric of $R^{(\text{zero})}$, $R^{(\text{mean})}$, $R^{(\text{blur})}$, and $R^{(\text{conditional})}$, respectively. Table 1 shows that the ratio R by using previous baseline values were all large. Although the masking method based on conditional distribution (Frye et al., 2021) performed better than some other baseline values, our method exhibited the best performance. It indicates that previous masking methods did not remove most existing patterns and/or trigger new patterns.

2.2 Absence states and optimal baseline values

In the original scenario of game theory, the Shapley value was proposed without the need to define the absence of players. When people explain a DNN, we consider the **true absence state** of variables as triggering the least causal patterns. This is because the absence of input variables is supposed to remove all causal effects containing these masked variables. To this end, Remark 2 and Theorem 3 show that correct baseline values usually generate the simplest causality-based explanation, *i.e.,* using the least number of causal patterns to explain the DNN. If an incorrect baseline value b_i does not fully remove all effects of AND relationships of the variable i, then the remained effects will be mistakenly explained as a large number of other redundant patterns.

The above proof well fits Occam's Razor, *i.e.*, the simplest causality with the minimum causal patterns may potentially represent the essence of the DNN's inference logic. This perspective also supports us to consider the baseline values that minimize the number of salient causal patterns (*i.e.*, achieving the simplest causality) as the optimal baseline values.

Therefore, the learning of the baseline value b_i^* of the i-th variable can be formulated to sparsify causal patterns in the deep model. Particularly, such baseline values are supposed to remove existing causal effects without introducing many new effects.

$$\boldsymbol{b}^* = \arg\min_{\boldsymbol{b}} \sum\nolimits_{\boldsymbol{x}} |\Omega(\boldsymbol{x})|, \quad \text{subject to } \Omega(\boldsymbol{x}) = \{S' \subseteq N \,||I_S(\boldsymbol{x}|\boldsymbol{b})| > \tau\} \tag{5}$$

where $I_S(\boldsymbol{x}|\boldsymbol{b})$ denotes the causal effect computed on the sample $\boldsymbol{x}$ by setting baseline values to $\boldsymbol{b}$.

3 Estimating baseline values

Based on Theorem 3, we derive Eq. (5) to learn optimal baseline values, but the computational cost of enumerating all causal patterns is exponential. Thus, we explore an approximate solution to learning baseline values. According to Theorem 4, *incorrect baseline values usually mistakenly explain high-order causal patterns as an unnecessarily large number of low-order causal patterns,* where the order m of the causal effect I_S is defined as the cardinality of S, $m = |S|$.

Thus, we aim to learn baseline values that correctly reflect high-order causal patterns in the model without introducing redundant low-order patterns. *The objective of learning baseline values* is roughly transformed to a loss function that penalizes effects of low-order causal patterns, in order to prevent learning incorrect baseline values that mistakenly represent the high-order pattern as an exponential number of low-order patterns.

$$\min_{\boldsymbol{b}} L(\boldsymbol{b}), \quad \text{subject to } L(\boldsymbol{b}) = \sum\nolimits_{\boldsymbol{x}} \sum\nolimits_{S \subseteq N, |S| \leq k} |I_S(\boldsymbol{x}|\boldsymbol{b})| \tag{6}$$

An approximate-yet-efficient solution. When each input sample contains a huge number of variables, *e.g.*, an image sample, directly optimizing Eq. (6) is NP-hard. Fortunately, we find the multi-order Shapley value and the multi-order marginal benefit in the following equation have strong connections with multi-order causal patterns (proven in Appendix G), as follows.

$$\begin{aligned} \phi_i^{(m)}(\boldsymbol{x}|\boldsymbol{b}) &\overset{\text{def}}{=} \mathbb{E}_{\substack{S \subseteq N \setminus \{i\} \\ |S|=m}} \left[v(\boldsymbol{x}_{S \cup \{i\}}, \boldsymbol{b}) - v(\boldsymbol{x}_S, \boldsymbol{b}) \right] = \mathbb{E}_{\substack{S \subseteq N \setminus \{i\} \\ |S|=m}} \left[\sum\nolimits_{L \subseteq S} I_{L \cup \{i\}}(\boldsymbol{x}|\boldsymbol{b}) \right] \\ \Delta v_i(S|\boldsymbol{x}, \boldsymbol{b}) &\overset{\text{def}}{=} v(\boldsymbol{x}_{S \cup \{i\}}, \boldsymbol{b}) - v(\boldsymbol{x}_S, \boldsymbol{b}) = \sum\nolimits_{L \subseteq S} I_{L \cup \{i\}}(\boldsymbol{x}|\boldsymbol{b}) \end{aligned} \tag{7}$$

where $\phi_i^{(m)}(\boldsymbol{x}|\boldsymbol{b})$ and $\Delta v_i(S|\boldsymbol{x}, \boldsymbol{b})$ denote the m-order Shapley value and the m-order marginal benefit computed using baseline values $\boldsymbol{b}$, respectively, where the order m is given as $m = |S|$.

According to the above equation, high-order casual patterns I_S are only contained by high-order Shapley values $\phi_i^{(m)}$ and high-order marginal benefits Δv_i. Therefore, in order to penalize the effects of low-order causal patterns, we penalize the strength of low-order Shapley values and low-order marginal benefits, respectively, as an engineering solution. We design to minimize the following

Table 3: Examples of generated functions and their ground-truth baseline values.

Functions ($\forall i \in N, x_i \in \{0,1\}$)	The ground truth of baseline values
$-0.185x_1(x_2+x_3)^{2.432} - x_4x_5x_6x_7$	$b_i^* = 0$ for $i \in \{1,2,3,4,5,6,7\}$
$-x_1x_2x_3 + \textit{sigmoid}(-5x_4x_5x_6x_7 + 2.50) - x_8x_9$	$b_i^* = 1$ for $i \in \{4,5,6,7\}$, $b_i^* = 0$ for $i \in \{1,2,3,8,9\}$
$-\textit{sigmoid}(+4x_1 - 4x_2 + 4x_3 - 6.00) - x_4x_5x_6x_7 - x_8x_9x_{10}$	$b_i^* = 1$ for $i = 2$, $b_i^* = 0$ for $i \in \{1,3,4,5,6,7,8,9,10\}$

Table 4: Accuracy of the learned baseline values.

	L_{Shapley}			L_{marginal}		
	initialize as 0	initialize as 0.5	initialize as 1	initialize as 0	initialize as 0.5	initialize as 1
Synthetic functions	98.06%	98.70%	98.70%	98.06%	98.14%	98.14%
Functions in (Tsang et al., 2018)	88.52%	91.80%	90.16%	86.89%	91.80%	90.16%

two loss functions to penalize low-order causal patterns, which boost the computational efficiency. In experiments, these loss functions were optimized via SGD.

$$L_{\text{Shapley}}(\boldsymbol{b}) = \sum_{m\sim\text{Unif}(0,\lambda)} \sum_{\boldsymbol{x}\in X} \sum_{i\in N} |\phi_i^{(m)}(\boldsymbol{x}|\boldsymbol{b})|, \quad L_{\text{marginal}}(\boldsymbol{b}) = \sum_{m\sim\text{Unif}(0,\lambda)} \sum_{\boldsymbol{x}\in X} \sum_{i\in N} \mathbb{E}_{\substack{S\subseteq N \\ |S|=m}} |\Delta v_i(S|\boldsymbol{x},\boldsymbol{b})| \tag{8}$$

where $\lambda \geq m$ denotes the maximum order to be penalized. We have conducted experiments to verify that baseline values $\boldsymbol{b}$ learned by loss functions in Eq. (8) could effectively reduced causal effects of low-order causal patterns in Eq. (6). Please see Appendix F.4 for results.

Most importantly, we still used the metric R in Section 2 to check whether the learned baseline values removed original causal patterns in the input while not introducing new patterns. The low value of R^{ours} in Table 1 shows that baseline values learned by our method **successfully removed existing salient causal patterns without introducing many new salient patterns.**

4 Experiments

4.1 Verification of correctness of baseline values and Shapley values

Correctness of baseline values on synthetic functions. People usually cannot determine the ground truth of baseline values for real images, such as the MNIST dataset. Therefore, we conducted experiments on synthetic functions with ground-truth baseline values, in order to verify the correctness of the learned baseline values. We randomly generated 100 functions, whose causal patterns and ground truth of baseline values could be easily determined. This dataset will be released after the paper acceptance. The generated functions were composed of addition, subtraction, multiplication, exponentiation, and the *sigmoid* operations (see Table 3). For example, for the function $y = \textit{sigmoid}(3x_1x_2 - 3x_3 - 1.5) - x_4x_5 + 0.25(x_6+x_7)^2$, $x_i \in \{0,1\}$, there were three causal patterns (*i.e.* $\{x_1,x_2,x_3\}, \{x_4,x_5\}, \{x_6,x_7\}$), which were activated only if $x_i = 1$ for $i \in \{1,2,4,5,6,7\}$ and $x_3 = 0$. In this case, the ground truth of baseline values was $b_i^* = 0$ for $i \in \{1,2,4,5,6,7\}$ and $b_3^* = 1$. Please see Appendix F.5 for more discussions about the setting of ground-truth baseline values.

We used our method to learn baseline values on these functions and tested the accuracy. Note that $|b_i - b_i^*| \in [0,1]$ and $b_i^* \in \{0,1\}$. If $|b_i - b_i^*| < 0.5$, we considered the learned baseline value correct. We set $\lambda = 0.5n$ in both L_{Shapley} and L_{marginal}. The results are reported in Table 4 and are discussed later.

Correctness of baseline values on functions in (Tsang et al., 2018). Besides, we also evaluated the correctness of the learned baseline values using functions in Tsang et al. (2018). Among all the 92 input variables in these functions, the ground truth of 61 variables could be determined (see Appendix F.5). Thus, we used these annotated baseline values to test the accuracy. Table 4 reports the accuracy of the learned baseline values on the above functions. In most cases, the accuracy was above 90%, showing that our method could effectively learn correct baseline values. A few functions in (Tsang et al., 2018) did not have salient causal patterns, which caused errors in the learning. Besides, in experiments, we tested our method under three different intializations of baseline values (*i.e.,* 0, 0.5 and 1). Table 4 show that baseline values learned with different initialization settings all converged to similar and high accuracy.

Correctness of the computed Shapley values. Incorrect baseline values lead to incorrect Shapley values. We verified the correctness of the computed Shapley values on the extended Addition-Multiplication dataset (Zhang et al., 2021). We added the subtraction operation to avoid all baseline values being zero. Theorem 2 considers the Shapley value as a uniform assignment of effects of each causal pattern to its compositional variables. This enabled us to determine the ground-truth Shapley value of variables without baseline values based on causal patterns. For example, the function $f(\boldsymbol{x}) = 3x_1x_2 + 5x_3x_4 + x_5$ *s.t.* $\boldsymbol{x} = [1,1,1,1,1]$ contained three causal patterns, according to

Table 5: Accuracy of Shapley values on the extended Addition-Multiplication dataset using different settings of baseline values.

	Zero	Mean	Baseline values in SHAP	Kernel SHAP	Frye et al. (2021)	Ours
Accuracy	82.88%	72.63%	81.25%	33.88%	66.00%	100%

Table 6: An example of Shapley values computed on different baseline values. The function is $model(\boldsymbol{x}) = -2.62x_1 - 5x_3 - 1.98x_6(x_4 - 0.94) + 1.15(x_5 - 0.91) - 4.23x_7$, and the input is $\boldsymbol{x} = [0, 1, 1, 1, 1, 1, 1]$.

Baseline values	The computed Shapley values $\{\phi_i\}$
Truth/Ours	$\{0, 0, -5, -0.06, 0.10, -0.06, -4.23\}$
Zero baseline	$\{0, 0, -5, -0.99, 1.15, 0.87, -4.23\}$
Mean baseline	$\{1.31, 0, -2.50, -0.74, 0.58, 0.19, -2.11\}$
Setting in SHAP	$\{0.014, 0, -0.011, -0.003, 0.003, 0.001, -0.010\}$

the principle of the most simplified causality. Accordingly, the ground-truth Shapley values were $\hat{\phi}_1 = \hat{\phi}_2 = 3/2$, $\hat{\phi}_3 = \hat{\phi}_4 = 5/2$, and $\hat{\phi}_5 = 1$. See Appendix F.6 for more details. The estimated Shapley value ϕ_i was considered correct if $|\phi_i - \hat{\phi}_i| \leq 0.01$; otherwise, incorrect. Then, we computed the accuracy of the estimated Shapley values as the ratio of input variables with correct Shapley values.

We compared our methods with six previous choices of baseline values in Section 2, *i.e. zero baseline values* (Ancona et al., 2019), *mean baseline values* (Dabkowski and Gal, 2017), *blurring images* (Fong et al., 2019), and three implementations of varying baseline values in SHAP (Lundberg and Lee, 2017), SAGE (Covert et al., 2020b), and (Frye et al., 2021).

Discussion on why the learned baseline values generated correct Shapley values. We computed Shapley values of variables in the extended Addition-Multiplication dataset using different baseline values, and compared their accuracy in Table 5. The result shows that our method exhibited the highest accuracy. Table 6 shows an example of incorrect Shapley values computed by using other baseline values. Our method generated correct Shapley values on this example. For the variable x_6, due to its negative coefficient -1.98, its contribution should be negative. However, all other baseline values generated positive Shapley values for x_6. The term $-4.23x_7$ showed the significant effect of the variable x_7 on the output, but its Shapley value computed using baseline values in SHAP was just -0.010, which was obviously incorrect. Also, the absolute value of the Shapley value of x_1 computed using baseline values in SHAP was even larger than that of x_7, demonstrating the unreliability of SHAP. Please see Appendix F.7 for more examples.

4.2 Results and Evaluation on Realistic Datasets and Models

Learning baseline values. We used our method to learn baseline values for MLPs, LeNet (LeCun et al., 1998), and ResNet-20 (He et al., 2016) trained on the UCI South German Credit dataset (namely *credit dataset*) (Dua and Graff, 2017), the UCI Census Income dataset (namely *income dataset*) (Dua and Graff, 2017), the MNIST dataset (LeCun et al., 1998), and the CIFAR-10 dataset (Krizhevsky et al., 2009), respectively. We learned baseline values by using either L_{Shapley} or L_{marginal} as the loss function. In the computation of L_{Shapley}, we set $v(\boldsymbol{x}_S) = \log \frac{p(y^{\text{truth}}|\boldsymbol{x}_S)}{1-p(y^{\text{truth}}|\boldsymbol{x}_S)}$. In the computation of L_{marginal}, $|\Delta v_i(S)|$ was set to $|\Delta v_i(S)| = \|h(\boldsymbol{x}_{S\cup\{i\}}) - h(\boldsymbol{x}_S)\|_1$, where $h(\boldsymbol{x}_S)$ denotes the output feature of the penultimate layer given the masked input $\boldsymbol{x}_S$, in order to boost the efficiency of learning. Please see Appendix F.8 for other potential settings of $v(\boldsymbol{x}_S)$. We set $\lambda = 0.2n$ for the MNIST and the CIFAR-10 datasets, and set $\lambda = 0.5n$ for the simpler data in two UCI datasets. Given baseline values, we used the sampling-based approximation (Castro et al., 2009) to estimate Shapley values, and we have verified the stability of the estimated Shapley values in Appendix F.10.

We used two ways to initialize baseline values before learning, *i.e.* setting to zero or mean values over different samples, namely *zero-init* and *mean-init*, respectively. Fig. 3 (left) and Fig. 12 show that baseline values learned with different initialization settings all converged to similar baseline values, except for very few dimensions having multiple local-minimum solutions (discussed in Appendix F.12), which proved the stability of our method.

Comparison of attributions computed using different baseline values. Fig. 3 shows the learned baseline values and the computed Shapley values on the income dataset. We found that attributions generated by zero/mean baseline values conflicted with the results of all other methods. Our method obtained that the *occupation* had more influence than the *marital status* on the income, which was somewhat consistent with our life experience. However, baseline values in SHAP and SAGE sometimes generated abnormal explanations. In this top-right example, the attribute *capital gain* was zero, which was not supposed to support the prediction of "the person made over 50K a year." However, the SAGE's baseline values generated a large positive Shapley value for *capital gain*. In the

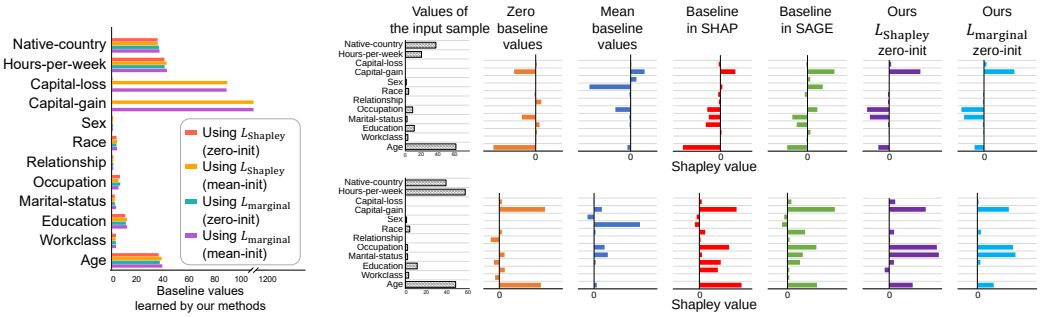


Figure 3: The learned baseline values (left) and Shapley values computed with different baseline values (right) on the income dataset. **Results on the MNIST, the CIFAR-10, and the credit datasets are shown in Appendix F.11 and F.12.**

Table 7: Accuracy of Shapley values computed on the pre-defined decision tree, which was based on the MNIST dataset.

Baseline value	zero	mean	baseline in SHAP	ours
Accuracy	92.99%	92.93%	93.37%	**96.84%**

Table 8: The learned baseline values could recover original samples from adversarial examples.

	$\|\boldsymbol{x}^{\text{adv}}-\boldsymbol{x}\|_2$	$\|\boldsymbol{b}-\boldsymbol{x}\|_2$
MNIST on LeNet	2.33	**0.43**
MNIST on AlexNet	2.53	**1.15**
CIFAR-10 on ResNet-20	1.19	**1.11**

bottom-right example, both SHAP and SAGE considered the *marital status* important for the prediction. SHAP did not considered the *occupation* as an important variable. Therefore, we considered these explanations not reliable. Attribution maps and baseline values generated on the CIFAR-10 and the MNIST datasets are provided in Appendix F.11. Compared to zero/mean/blurring baseline values, our baseline values were more likely to remove noisy variables on the background, which were far from the foreground in images. Compared to SHAP, our method yielded more informative attributions. Besides, our method generated smoother attributions than SAGE.

Quantitative evaluation of attributions for image classification. In order to quantitatively evaluate Shapley values computed by different baseline values on the MNIST dataset, we constructed an And-Or decision tree following (Harradon et al., 2018), whose structure directly provided the ground-truth Shapley value for each input variable. We used different attribution methods to explain the decision tree. Please see Appendix F.13 for more about the And-Or tree. Table 7 shows that our method generated more accurate Shapley values than other baseline values.

4.3 Removing adversarial perturbations from the input

Let $\boldsymbol{x}$ denote the normal sample, and let $\boldsymbol{x}^{\text{adv}} = \boldsymbol{x} + \delta$ denote the adversarial example generated by (Madry et al., 2018). According to (Ren et al., 2021b), the adversarial example $\boldsymbol{x}^{\text{adv}}$ mainly created out-of-distribution bivariate interactions with high-order contexts, which were actually related to the high-order interactions (causal patterns) in this paper. Thus, in the scenario of this study, the adversarial utility was owing to out-of-distribution high-order interactions (causal patterns). The removal of input variables was supposed to remove most high-order causal patterns.

Therefore, the baseline value can be considered as the recovery of the original sample. In this way, we used the adversarial example $\boldsymbol{x}^{\text{adv}}$ to initialize baseline values before learning, and used L_{marginal} to learn baseline values. If the learned baseline values $\boldsymbol{b}$ satisfy $\|\boldsymbol{b}-\boldsymbol{x}\|_1 \leq \|\boldsymbol{x}^{\text{adv}}-\boldsymbol{x}\|_1$, we considered that our method successfully recovered the original sample to some extent. We conducted experiments using LeNet, AlexNet (Krizhevsky et al., 2012), and ResNet-20 on the MNIST dataset ($\|\delta\|_\infty \leq 32/255$) and the CIFAR-10 dataset ($\|\delta\|_\infty \leq 8/255$). Table 8 shows that our method recovered original samples from adversarial examples, which demonstrated the effectiveness of our method. Please see Appendix F.14 for more discussions.

5 Conclusions

In this paper, we have defined the absence state of input variables in terms of causality. Then, we have found that most existing masking methods cannot faithfully remove existing causal patterns without triggering new patterns. In this way, we have formulated optimal baseline values for the computation of Shapley values as those that remove most causal patterns. Then, we have proposed an approximate-yet-efficient method to learn optimal baseline values that represent the absence states of input variables. Experimental results have demonstrated the effectiveness of our method.

ETHIC STATEMENT

This paper aims to examine the masking approach in previous explaining methods. We find that previous settings of the masking approach cannot faithfully represent the absence of input variables, thereby hurt the trustworthiness of the obtained explanations. Therefore, we propose a new method to learn optimal baseline values to represent the absence of input variables. In this way, the trustworthiness of explanations of the DNN is further boosted. There is no ethic issues with this paper.

REPRODUCIBILITY STATEMENT

We have provided proofs for all theoretical results in Appendix D and Appendix G. We have also provided experimental details in Section 4 and Appendix F. Furthermore, we will release the code when the paper is accepted.

REFERENCES

Kjersti Aas, Martin Jullum, and Anders Løland. Explaining individual predictions when features are dependent: More accurate approximations to shapley values. *Artif. Intell.*, 298(C), sep 2021. ISSN 0004-3702. doi: 10.1016/j.artint.2021.103502.

Chirag Agarwal and Anh Nguyen. Explaining image classifiers by removing input features using generative models. In Hiroshi Ishikawa, Cheng-Lin Liu, Tomas Pajdla, and Jianbo Shi, editors, *Computer Vision – ACCV 2020*, pages 101–118, Cham, 2021. Springer International Publishing. ISBN 978-3-030-69544-6.

David Alvarez-Melis and Tommi Jaakkola. A causal framework for explaining the predictions of black-box sequence-to-sequence models. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 412–421, Copenhagen, Denmark, September 2017. Association for Computational Linguistics. doi: 10.18653/v1/D17-1042.

Marco Ancona, Cengiz Oztireli, and Markus Gross. Explaining deep neural networks with a polynomial time algorithm for shapley value approximation. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 272–281. PMLR, 09–15 Jun 2019.

Robert J. Aumann and Lloyd S. Shapley. *Values of Non-Atomic Games*. Princeton University Press, Princeton, 2015. ISBN 9781400867080. doi: doi:10.1515/9781400867080.

Peer Bork, Lars J Jensen, Christian von Mering, Arun K Ramani, Insuk Lee, and Edward M Marcotte. Protein interaction networks from yeast to human. *Current Opinion in Structural Biology*, 14(3):292–299, 2004. ISSN 0959-440X. doi: https://doi.org/10.1016/j.sbi.2004.05.003.

Javier Castro, Daniel Gómez, and Juan Tejada. Polynomial calculation of the shapley value based on sampling. *Computers & Operations Research*, 36(5):1726–1730, 2009. ISSN 0305-0548. doi: https://doi.org/10.1016/j.cor.2008.04.004. Selected papers presented at the Tenth International Symposium on Locational Decisions (ISOLDE X).

Aditya Chattopadhyay, Piyushi Manupriya, Anirban Sarkar, and Vineeth N Balasubramanian. Neural network attributions: A causal perspective. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 981–990. PMLR, 09–15 Jun 2019.

Ian Covert, Scott Lundberg, and Su-In Lee. Feature removal is a unifying principle for model explanation methods. *arXiv preprint arXiv:2011.03623*, 2020a.

Ian C. Covert, Scott Lundberg, and Su-In Lee. Understanding global feature contributions with additive importance measures. In *NeurIPS 2020*. ACM, December 2020b.

Tianyu Cui, Pekka Marttinen, and Samuel Kaski. Learning global pairwise interactions with bayesian neural networks. In Giuseppe De Giacomo, Alejandro Catalá, Bistra Dilkina, Michela Milano, Senén Barro, Alberto Bugarín, and Jérôme Lang, editors, *ECAI 2020 - 24th European Conference on Artificial Intelligence, 29 August-8 September 2020, Santiago de Compostela, Spain, August 29 - September 8, 2020 - Including 10th Conference on Prestigious Applications of Artificial Intelligence (PAIS 2020)*, volume 325 of *Frontiers in Artificial Intelligence and Applications*, pages 1087–1094. IOS Press, 2020. doi: 10.3233/FAIA200205.

Piotr Dabkowski and Yarin Gal. Real time image saliency for black box classifiers. In Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 6967–6976, 2017.

Huiqi Deng, Qihan Ren, Hao Zhang, and Quanshi Zhang. Discovering and explaining the representation bottleneck of DNNS. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net, 2022.

Alexey Dosovitskiy and Thomas Brox. Inverting visual representations with convolutional networks. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pages 4829–4837. IEEE Computer Society, 2016. doi: 10.1109/CVPR.2016.522.

Dheeru Dua and Casey Graff. UCI machine learning repository, 2017.

Ruth Fong, Mandela Patrick, and Andrea Vedaldi. Understanding deep networks via extremal perturbations and smooth masks. In *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*, pages 2950–2958. IEEE, 2019. doi: 10.1109/ICCV.2019.00304.

Ruth C. Fong and Andrea Vedaldi. Interpretable explanations of black boxes by meaningful perturbation. In *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, pages 3449–3457. IEEE Computer Society, 2017. doi: 10.1109/ICCV.2017.371.

Christopher Frye, Damien de Mijolla, Tom Begley, Laurence Cowton, Megan Stanley, and Ilya Feige. Shapley explainability on the data manifold. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021.

Michel Grabisch and Marc Roubens. An axiomatic approach to the concept of interaction among players in cooperative games. *International Journal of game theory*, 28(4):547–565, 1999. doi: 10.1007/s001820050125.

Ulrike Grömping. Estimators of relative importance in linear regression based on variance decomposition. *The American Statistician*, 61(2):139–147, 2007.

Riccardo Guidotti, Anna Monreale, Salvatore Ruggieri, Dino Pedreschi, Franco Turini, and Fosca Giannotti. Local rule-based explanations of black box decision systems. *arXiv preprint arXiv:1805.10820*, 2018.

Michael Harradon, Jeff Druce, and Brian Ruttenberg. Causal learning and explanation of deep neural networks via autoencoded activations. *arXiv preprint arXiv:1802.00541*, 2018.

John C. Harsanyi. *A Simplified Bargaining Model for the n-Person Cooperative Game*, pages 44–70. Springer Netherlands, Dordrecht, 1982. ISBN 978-94-017-2527-9. doi: 10.1007/978-94-017-2527-9_3.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016. doi: 10.1109/CVPR.2016.90.

Patrik O. Hoyer, Dominik Janzing, Joris M. Mooij, Jonas Peters, and Bernhard Schölkopf. Nonlinear causal discovery with additive noise models. In Daphne Koller, Dale Schuurmans, Yoshua Bengio, and Léon Bottou, editors, *Advances in Neural Information Processing Systems 21, Proceedings of the Twenty-Second Annual Conference on Neural Information Processing Systems, Vancouver, British Columbia, Canada, December 8-11, 2008*, pages 689–696. Curran Associates, Inc., 2008.

Joseph D. Janizek, Pascal Sturmfels, and Su-In Lee. Explaining explanations: Axiomatic feature interactions for deep networks. *J. Mach. Learn. Res.*, 22:104:1–104:54, 2021.

Xisen Jin, Zhongyu Wei, Junyi Du, Xiangyang Xue, and Xiang Ren. Towards hierarchical importance attribution: Explaining compositional semantics for neural sequence models. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020.

Alon Keinan, Ben Sandbank, Claus C. Hilgetag, Isaac Meilijson, and Eytan Ruppin. Fair attribution of functional contribution in artificial and biological networks. *Neural Comput.*, 16(9):1887–1915, 2004. doi: 10.1162/0899766041336387.

Murat Kocaoglu, Christopher Snyder, Alexandros G. Dimakis, and Sriram Vishwanath. Causalgan: Learning causal implicit generative models with adversarial training. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 2018.

Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. In Peter L. Bartlett, Fernando C. N. Pereira, Christopher J. C. Burges, Léon Bottou, and Kilian Q. Weinberger, editors, *Advances in Neural Information Processing Systems 25: 26th Annual Conference on Neural Information Processing Systems 2012. Proceedings of a meeting held December 3-6, 2012, Lake Tahoe, Nevada, United States*, pages 1106–1114, 2012.

I. Elizabeth Kumar, Suresh Venkatasubramanian, Carlos Scheidegger, and Sorelle A. Friedler. Problems with shapley-value-based explanations as feature importance measures. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pages 5491–5500. PMLR, 2020.

Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proc. IEEE*, 86(11):2278–2324, 1998. doi: 10.1109/5.726791.

Scott M. Lundberg and Su-In Lee. A unified approach to interpreting model predictions. In Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 4765–4774, 2017.

Scott M Lundberg, Gabriel G Erion, and Su-In Lee. Consistent individualized feature attribution for tree ensembles. *arXiv preprint arXiv:1802.03888*, 2018.

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 2018.

Luke Merrick and Ankur Taly. The explanation game: Explaining machine learning models using shapley values. In Andreas Holzinger, Peter Kieseberg, A Min Tjoa, and Edgar R. Weippl, editors, *Machine Learning and Knowledge Extraction - 4th IFIP TC 5, TC 12, WG 8.4, WG 8.9, WG 12.9 International Cross-Domain Conference, CD-MAKE 2020, Dublin, Ireland, August 25-28, 2020, Proceedings*, volume 12279 of *Lecture Notes in Computer Science*, pages 17–38. Springer, 2020. doi: 10.1007/978-3-030-57321-8_2.

Alexander Mordvintsev, Christopher Olah, and Mike Tyka. Inceptionism: Going deeper into neural networks. In *Google Research Blog*, 2015.

W. James Murdoch, Peter J. Liu, and Bin Yu. Beyond word importance: Contextual decomposition to extract interactions from lstms. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 2018.

Leland Gerson Neuberg. Causality: Models, reasoning, and inference, by judea pearl, cambridge university press, 2000. *Econometric Theory*, 19(4):675–685, 2003. doi: 10.1017/S0266466603004109.

Judea Pearl. Causal inference in statistics: An overview. *Statistics Surveys*, 3:96–146, 2009.

Jie Ren, Mingjie Li, Qirui Chen, Huiqi Deng, and Quanshi Zhang. Towards axiomatic, hierarchical, and symbolic explanation for deep models. *arXiv preprint arXiv:2111.06206*, 2021a.

Jie Ren, Die Zhang, Yisen Wang, Lu Chen, Zhanpeng Zhou, Yiting Chen, Xu Cheng, Xin Wang, Meng Zhou, Jie Shi, and Quanshi Zhang. Towards a unified game-theoretic view of adversarial perturbations and robustness. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 3797–3810. Curran Associates, Inc., 2021b.

Ramprasaath R. Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, pages 618–626. IEEE Computer Society, 2017. doi: 10.1109/ICCV.2017.74.

Pranab Kumar Sen, Richard H. Lindeman, Peter F. Merenda, and Ruth Z. Gold. Introduction to bivariate and multivariate analysis. *Journal of the American Statistical Association*, 76:752, 1981.

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

Avanti Shrikumar, Peyton Greenside, and Anshul Kundaje. Learning important features through propagating activation differences. In Doina Precup and Yee Whye Teh, editors, *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, volume 70 of *Proceedings of Machine Learning Research*, pages 3145–3153. PMLR, 2017.

Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. In Yoshua Bengio and Yann LeCun, editors, *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Workshop Track Proceedings*, 2014.

Chandan Singh, W James Murdoch, and Bin Yu. Hierarchical interpretations for neural network predictions. In *International Conference on Learning Representations*, 2018.

Xi Song, Tianfu Wu, Yunde Jia, and Song-Chun Zhu. Discriminatively trained and-or tree models for object detection. In *2013 IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA, June 23-28, 2013*, pages 3278–3285. IEEE Computer Society, 2013. doi: 10.1109/CVPR.2013.421. URL `https://doi.org/10.1109/CVPR.2013.421`.

Daria Sorokina, Rich Caruana, Mirek Riedewald, and Daniel Fink. Detecting statistical interactions with additive groves of trees. In William W. Cohen, Andrew McCallum, and Sam T. Roweis, editors, *Machine Learning, Proceedings of the Twenty-Fifth International Conference (ICML 2008), Helsinki, Finland, June 5-9, 2008*, volume 307 of *ACM International Conference Proceeding Series*, pages 1000–1007. ACM, 2008. doi: 10.1145/1390156.1390282.

Erik Štrumbelj and Igo Kononenko. Explaining prediction models and individual predictions with feature contributions. *Knowledge and information systems*, 41(3):647–665, 2014. doi: 10.1007/s10115-013-0679-x.

Erik Štrumbelj, Igor Kononenko, and M Robnik Šikonja. Explaining instance classifications with interactions of subsets of feature values. *Data & Knowledge Engineering*, 68(10):886–904, 2009. doi: 10.1016/j.datak.2009.01.004.

Pascal Sturmfels, Scott Lundberg, and Su-In Lee. Visualizing the impact of feature attribution baselines. *Distill*, 5(1):e22, 2020.

Mukund Sundararajan and Amir Najmi. The many shapley values for model explanation. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pages 9269–9278. PMLR, 2020.

Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In Doina Precup and Yee Whye Teh, editors, *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, volume 70 of *Proceedings of Machine Learning Research*, pages 3319–3328. PMLR, 2017.

Mukund Sundararajan, Kedar Dhamdhere, and Ashish Agarwal. The shapley taylor interaction index. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pages 9259–9268. PMLR, 2020.

Michael Tsang, Dehua Cheng, and Yan Liu. Detecting statistical interactions from neural network weights. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 2018.

Robert James Weber. *Probabilistic values for games*, page 101–120. Cambridge University Press, 1988. doi: 10.1017/CBO9780511528446.008.

Jason Yosinski, Jeff Clune, Anh Nguyen, Thomas Fuchs, and Hod Lipson. Understanding neural networks through deep visualization. *arXiv preprint arXiv:1506.06579*, 2015.

Hao Zhang, Yichen Xie, Longjie Zheng, Die Zhang, and Quanshi Zhang. Interpreting multivariate shapley interactions in dnns. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*, pages 10877–10886. AAAI Press, 2021.

Bolei Zhou, Aditya Khosla, Àgata Lapedriza, Aude Oliva, and Antonio Torralba. Object detectors emerge in deep scene cnns. In Yoshua Bengio and Yann LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015.

Bolei Zhou, Aditya Khosla, Àgata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pages 2921–2929. IEEE Computer Society, 2016. doi: 10.1109/CVPR.2016.319.

A RELATED WORKS

No previous methods directly examined the faithfulness of the masking methods. Instead, we made a survey in a larger scope of attribution methods and other explainable AI studies, and put them in the appendix. Nevertheless, we will put this section back to the main paper if the paper is accepted.

In the scope of explainable AI, many methods (Simonyan et al., 2014; Yosinski et al., 2015; Mordvintsev et al., 2015; Dosovitskiy and Brox, 2016; Zhou et al., 2015) have been proposed to explain the DNN. Among all methods, the estimation of attributions for each input variable represents a classical direction (Zhou et al., 2016; Selvaraju et al., 2017; Lundberg and Lee, 2017; Shrikumar et al., 2017). In this paper, we mainly focus on attributions based on Shapley values.

Shapley values. The Shapley value (Shapley, 1953) in game theory was widely considered as a fair distribution of the overall reward in a game to each player (Weber, 1988). (Sen et al., 1981) and (Grömping, 2007) used the Shapley value to attribute the correlation coefficient of a linear regression to input features. (Štrumbelj et al., 2009; Štrumbelj and Kononenko, 2014) used the Shapley value to attribute the prediction of a model to input features. (Bork et al., 2004) used the Shapley value to measure importances of protein interactions in large, complex biological interaction networks. (Keinan et al., 2004) employed the Shapley value to measure causal effects in neurophysical models. (Sundararajan et al., 2017) proposed Integrated Gradients based on the AumannShapley(Aumann and Shapley, 2015) cost-sharing technique. Besides above local explanations, (Covert et al., 2020b) focused on the global interpretability.

In order to compute the Shapley value in deep models efficiently, (Lundberg and Lee, 2017) proposed various approximations for Shapley valus in DNNs. (Lundberg et al., 2018) further computed the Shapley value on tree emsembles. (Aas et al., 2021) generalized the approximation method in (Lundberg and Lee, 2017) to the case when features were related to each other. (Ancona et al., 2019) further formulated a polynomial-time approximation of Shapley values for DNNs.

Baseline values. In terms of baseline values of Shapley values, most studies (Covert et al., 2020a; Merrick and Taly, 2020; Sundararajan and Najmi, 2020; Kumar et al., 2020) compared influences of baseline values on explanations, without providing any principles for setting baseline values. Shrikumar et al. (2017) proposed DeepLIFT to estimate attributions of input variables, and also mentioned the choice of baseline values. Besides, Agarwal and Nguyen (2021) and Frye et al. (2021) used generative models to alleviate the out-of-distribution problem caused by baseline values. Unlike previous studies, we rethink and formulate baseline values from the perspective of game-theoretic causality. We define the absent state of input variables, and propose a method to learn optimal baseline values based on the number of causal patterns.

Using causality to explain neural networks. Causality and causal learning have been used in deep models. Hoyer et al. (2008) and Pearl (2009) studied the causal structure between some observed variables in the data. Kocaoglu et al. (2018) built a generative model based on causality. The causality has also been used to explain the relationship between inputs and intermediate features/outputs of DNNs (Alvarez-Melis and Jaakkola, 2017; Chattopadhyay et al., 2019; Harradon et al., 2018). In comparison, in this study, the causal graph is constructed to represent the inference logic in the DNN.

Interactions. Many people defined interactions between input variables or weights to explain different models from different perspectives. (Sorokina et al., 2008) proposed an approach to detect interactions of input variables in an additive model. (Tsang et al., 2018) measured interactions of weights in a DNN. (Murdoch et al., 2018; Singh et al., 2018; Jin et al., 2020) used the contextual decomposition (CD) technique to extract variable interactions. (Cui et al., 2020) proposed a non-parametric probabilistic method to measure interactions using a Bayesian neural network. In game theory, (Grabisch and Roubens, 1999; Lundberg et al., 2018) proposed and used the Shapley interaction index based on Shapley values. (Janizek et al., 2021) extended the Integrated Gradients method (Sundararajan et al., 2017) to explain pairwise feature interactions in DNNs. (Sundararajan et al., 2020) defined the Shapley-Taylor index to measure interactions over binary features. In this paper, we use the multi-variate interaction of input variables based on the Harsanyi dividend (Harsanyi, 1982). Our interaction metric has strong connections to (Grabisch and Roubens, 1999), but represents elementary causal patterns in a more detailed manner, and satisfies the efficiency axiom.

B AXIOMS OF THE SHAPLEY VALUE

The Shapley value satisfies the following four axioms (Weber, 1988).
(1) *Linearity axiom*: If two games can be merged into a new game $u(\boldsymbol{x}_S) = v(\boldsymbol{x}_S) + w(\boldsymbol{x}_S)$, then Shapley values in the two old games also can be merged, *i.e.* $\forall i \in N$, $\phi_{i,u} = \phi_{i,v} + \phi_{i,w}$.
(2) *Dummy axiom and nullity axiom*: The dummy player i is defined as a player without any interactions with other players, *i.e.* satisfying $\forall S \subseteq N \setminus \{i\}$, $v(\boldsymbol{x}_{S\cup\{i\}}) = v(\boldsymbol{x}_S) + v(\boldsymbol{x}_{\{i\}})$. Then, the dummy player's Shapley value is computed as $\phi_i = v(\boldsymbol{x}_{\{i\}})$. The null player i is defined as a player that satisfies $\forall S \subseteq N \setminus \{i\}$, $v(\boldsymbol{x}_{S\cup\{i\}}) = v(\boldsymbol{x}_S)$. Then, the null player's Shapley value is $\phi_i = 0$.
(3) *Symmetry axiom*: If $\forall S \subseteq N \setminus \{i, j\}$, $v(\boldsymbol{x}_{S\cup\{i\}}) = v(\boldsymbol{x}_{S\cup\{j\}})$, then $\phi_i = \phi_j$.
(4) *Efficiency axiom*: The overall reward of the game is equal to the sum of Shapley values of all players, *i.e.* $v(\boldsymbol{x}_N) - v(\boldsymbol{x}_\emptyset) = \sum_{i\in N} \phi_i$.

C EXTENDED DISCUSSIONS ABOUT THE HARSANYI DIVIDEND

This section provides extended discussions about the causal effect based on the Harsanyi dividend (Harsanyi, 1982), and provides proofs of axioms and properties that the Harsanyi dividend satisfies.

Given a trained model and the input sample $\boldsymbol{x}$ with n variables $N = \{1, 2, \cdots, n\}$, let $v(\boldsymbol{x}_S)$ denote the model output when only variables in S are given. Then, $v(\boldsymbol{x}_N) - v(\boldsymbol{x}_\emptyset)$ represents the overall benefit owing to all input variables in $\boldsymbol{x}$, *w.r.t.* the model output without given any variables. In an input sample, different input variables interact with each other for inference, instead of working individually. The Harsanyi dividend I_S can be considered to measure the benefit to the model output from the interaction between variables in the subset $S \subseteq N$. Furthermore, the Harsanyi dividend ensures that the overall benefit $v(\boldsymbol{x}_N) - v(\boldsymbol{x}_\emptyset)$ can be decomposed into the sum of I_S of different subsets $\{S|S \subseteq N, S \neq \emptyset\}$.

$$v(\boldsymbol{x}_N) - v(\boldsymbol{x}_\emptyset) = \sum\nolimits_{S\subseteq N, S\neq\emptyset} I_S \tag{9}$$

For example, let us consider the $S_1 = \{head\}$ denote the bird head pattern in Figure 2 of the main paper. Patches inside S_1 collaborate with each other to form the head pattern, and to contribute an benefit I_{S_1} for the model output. Then, the overall benefit of all patches in the image $v(\boldsymbol{x}_N) - v(\boldsymbol{x}_\emptyset)$ can be represented as the sum of various patterns, such as the head pattern S_1 with its benefit I_{S_1}, the tail pattern S_2 with the benefit I_{S_2}, the torso pattern S_3 with the benefit I_{S_3}, etc.

The Harsanyi dividend I_S can be understood as measuring the additional benefit from the collaboration of input variables in S, in comparison with the benefit when they work individually or form smaller patterns. Specifically, let $v(\boldsymbol{x}_S) - v(\boldsymbol{x}_\emptyset)$ denote the overall benefit from all variables in S, then we remove the marginal benefits owing to collaborations of all smaller subsets L of variables in S, *i.e.* $\{I_L|L \subsetneq S, L \neq \emptyset\}$.

$$I_S \overset{\text{def}}{=} \underbrace{v(\boldsymbol{x}_S) - v(\boldsymbol{x}_\emptyset)}_{\text{the benefit from all variables in } S} - \sum_{L\subsetneq S, L\neq\emptyset} I_L \tag{10}$$

Axioms and properties of the Harsanyi dividend. The Harsanyi dividend is a trustworthy metric for the interaction utility, because it has been proven to satisfy *efficiency, linearity, dummy, symmetry, anonymity, recursive* and *interaction distribution* axioms. The Harsanyi dividend can also serve as an elementary explanation for existing game-theoretic interaction metrics (Grabisch and Roubens, 1999; Sundararajan et al., 2017).

C.1 PROOF OF AXIOMS OF THE HARSANYI DIVIDEND

Here, we provide details and proofs for the seven axioms that the Harsanyi dividend satisfies. These axioms are also proved in (Ren et al., 2021a). In the following part, we use $v(S) = v(\boldsymbol{x}_S)$ to denote the model output given only variables in S for simplicity. Before proving the axioms and properties of the Harsanyi dividend, we first prove the following lemma.

Lemma 1 *$\forall S \subseteq N$, if $S \neq \emptyset$, $\sum_{L\subseteq S}(-1)^{|S|-|L|} = 0$; otherwise, $\sum_{L\subseteq S}(-1)^{|S|-|L|} = 1$.*

• *Proof*:

$$\begin{aligned}\sum_{L\subseteq S}(-1)^{|S|-|L|} &= \sum_{l=0}^{s}\binom{s}{l}(-1)^{s-l} \quad \text{\% Let } s=|S|, l=|L| \\ &= (1+(-1))^s \\ &= 0^s = \begin{cases}1, & s=0(S=\emptyset)\\ 0, & \text{otherwise}\end{cases}\end{aligned}$$

Axiom 1 (Linearity) *If we merge outputs of two models, $u(S)=w(S)+v(S)$, then, $\forall S\subseteq N$, the interaction I_S^u* w.r.t. *the new output u can be decomposed into $I_S^u = I_S^w + I_S^v$.*

• *Proof*:

$$\begin{aligned}I_S^u &= \sum_{L\subseteq S}(-1)^{|S|-|L|}u(L) \\ &= \sum_{L\subseteq S}(-1)^{|S|-|L|}\left[w(S)+v(S)\right] \\ &= \sum_{L\subseteq S}(-1)^{|S|-|L|}w(L) + \sum_{L\subseteq S}(-1)^{|S|-|L|}v(L) \\ &= I_S^w + I_S^v\end{aligned}$$

Axiom 2 (Dummy) *The dummy variable $i\in N$ satisfies $\forall S\subseteq N\backslash\{i\}$, $v(S\cup\{i\})=v(S)+v(\{i\})$. It means that the variable i has no interactions with other variables,* i.e. *$\forall S\subseteq N\setminus\{i\}, S\neq\emptyset$, $I_{S\cup\{i\}}=0$.*

• *Proof*:

$$\begin{aligned}I_{S\cup\{i\}} &= \sum_{L\subseteq S\cup\{i\}}(-1)^{|S|+1-|L|}v(L) \\ &= \sum_{L\subseteq S}(-1)^{|S|+1-|L|}v(L) + \sum_{L\subseteq S}(-1)^{|S|-|L|}v(L\cup\{i\}) \\ &= \sum_{L\subseteq S}(-1)^{|S|+1-|L|}v(L) + \sum_{L\subseteq S}(-1)^{|S|-|L|}\left[v(L)+v(\{i\})\right] \\ &= \sum_{L\subseteq S}(-1)^{|S|+1-|L|}\left[-v(L)+v(L)\right] + \sum_{L\subseteq S}(-1)^{|S|-|L|}v(\{i\}) \\ &= \left(\sum_{L\subseteq S}(-1)^{|S|-|L|}\right)v(\{i\}) \\ &= 0 \quad \text{\% according to Lemma 1 and } S\neq\emptyset\end{aligned}$$

Axiom 3 (Symmetry) *If input variables $i,j\in N$ have same cooperations with other variables $\forall S\subseteq N\backslash\{i,j\}$, $v(S\cup\{i\})=v(S\cup\{j\})$, then they have same interactions with other variables, $\forall S\subseteq N\setminus\{i,j\}$, $I_{S\cup\{i\}}=I_{S\cup\{j\}}$.*

• *Proof*:

$$\begin{aligned}I_{S\cup\{i\}} &= \sum_{L\subseteq S\cup\{i\}}(-1)^{|S|+1-|L|}v(L) \\ &= \sum_{L\subseteq S}(-1)^{|S|+1-|L|}v(L) + \sum_{L\subseteq S}(-1)^{|S|-|L|}v(L\cup\{i\}) \\ &= \sum_{L\subseteq S}(-1)^{|S|+1-|L|}v(L) + \sum_{L\subseteq S}(-1)^{|S|-|L|}v(L\cup\{j\}) \\ &= \sum_{L\subseteq S\cup\{j\}}(-1)^{|S|+1-|L|}v(L) \\ &= I_{S\cup\{j\}}\end{aligned}$$

Axiom 4 (Efficiency) *The output of a model can be decomposed into interactions of different subsets of variables,* $v(N) = v(\emptyset) + \sum_{S \subseteq N, S \neq \emptyset} I_S$.

• *Proof*:

$$
\begin{aligned}
\text{right} &= v(\emptyset) + \sum_{S \subseteq N, S \neq \emptyset} I_S \\
&= v(\emptyset) + \sum_{S \subseteq N, S \neq \emptyset} \sum_{L \subseteq S} (-1)^{|S|-|L|} v(L) \\
&= \sum_{S \subseteq N} \sum_{L \subseteq S} (-1)^{|S|-|L|} v(L) \\
&= \sum_{L \subseteq N} \sum_{K \subseteq N \setminus L} (-1)^{|K|} v(L) \qquad \text{\% Let } K = S \setminus L \\
&= \sum_{L \subseteq N} \left[\sum_{|K|=0}^{n-|L|} \binom{n-|L|}{|K|} (-1)^{|K|} \right] v(L) \\
&= \sum_{L \subseteq N} \left[(1 + (-1))^{n-|L|} \right] v(L) \\
&= v(N) = \text{left} \quad \text{\% according to Lemma 1}
\end{aligned}
$$

Axiom 5 (Anonymity) *For any permutations* π *on* N*, we have* $\forall S \subseteq N, I_S^v = I_{\pi S}^{\pi v}$*, where* $\pi S \triangleq \{\pi(i) | i \in S\}$*, and the new model* πv *is defined by* $(\pi v)(\pi S) = v(S)$.

• *Proof:*

$$
\begin{aligned}
I_{\pi S}^{\pi v} &= \sum_{L \subseteq S} (-1)^{|S|-|L|} (\pi v)(\pi L) \\
&= \sum_{L \subseteq S} (-1)^{|S|-|L|} v(L) \\
&= I_S^v.
\end{aligned}
$$

Axiom 6 (Recursive) *The interaction benefit can be computed recursively. For* $i \in N$ *and* $S \subseteq N \setminus \{i\}$*, the interaction benefit of the pattern* $S \cup \{i\}$ *is equal to the interaction benefit of* S *with the presence of* i *minus the the interaction benefit of* S *with the absence of* i*,* i.e. $I_{S \cup \{i\}} = I_S|_{i \text{ is always present}} - I_S$ *(s.t.* $S \subseteq N \setminus \{i\}$*).* $I_S|_{i \text{ is always present}}$ *denotes the interaction benefit when the variable* i *is always present as a constant context,* i.e. $I_S|_{i \text{ is always present}} = \sum_{L \subseteq S} (-1)^{|S|-|L|} \cdot v(L \cup \{i\})$.

• *Proof:*

$$
\begin{aligned}
I_{S \cup \{i\}} &= \sum_{L \subseteq S \cup \{i\}} (-1)^{|S|+1-|L|} v(L) \\
&= \sum_{L \subseteq S} (-1)^{|S|+1-|L|} v(L) + \sum_{L \subseteq S} (-1)^{|S|-|L|} v(L \cup \{i\}) \\
&= \sum_{L \subseteq S} (-1)^{|S|-|L|} v(L \cup \{i\}) - \sum_{L \subseteq S} (-1)^{|S|-|L|} v(L) \\
&= I_S|_{i \text{ is always present}} - I_S.
\end{aligned}
$$

Axiom 7 (Interaction distribution) *This axiom characterizes how interactions are distributed for "interaction functions" (Sundararajan et al., 2020). An interaction function* v_T *parameterized by a context* T *satisfies* $\forall S : T \subseteq S \subseteq N, v_T(S) = c$*, and* $v_T(S) = 0$ *if* $T \subsetneq S$*. Then, we have* $I_T = c$*, and* $\forall S \neq T$*,* $I_S = 0$.

• *Proof:* If $S \subsetneq T$, we have

$$
I_S = \sum_{L \subseteq S} (-1)^{|S|-|L|} \cdot \underbrace{v(L)}_{\forall L \subsetneq S \subsetneq T, v(L)=0} = 0.
$$

If $S = T$, we have

$$
\begin{aligned}
I_S =& I_T = \sum_{L \subseteq T} (-1)^{|T|-|L|} v(L) \\
=& v(T) + \sum_{L \subsetneq T} (-1)^{|T|-|L|} \underbrace{v(L)}_{=0} = c.
\end{aligned}
$$

If $S \subsetneq T$, we have

$$
\begin{aligned}
I_S =& \sum_{L \subseteq S} (-1)^{|S|-|L|} v(L) \\
=& c \cdot \sum_{\substack{L \subseteq S \\ L \supseteq T}} (-1)^{|S|-|L|} \\
=& c \cdot \sum_{m=0} \binom{|S| - |T|}{m} (-1)^m = 0.
\end{aligned}
$$

C.2 Proof of connections between the Harsanyi dividend and other game-theoretic interaction metrics

In this section, we prove the relationship between the Harsanyi dividend and other game-theoretic interaction metrics, including the Shapley interaction index (Grabisch and Roubens, 1999) and the Shapley Taylor interaction index (Sundararajan et al., 2020). Such relationship has been proven in (Ren et al., 2021a).

Lemma 2 (Connection to the marginal benefit) $\Delta v_T(S) = \sum_{L \subseteq T} (-1)^{|T|-|L|} v(L \cup S)$ *denotes the marginal benefit (Grabisch and Roubens, 1999) of variables in* $T \subseteq N \setminus S$ *given the environment* S. $\Delta v_T(S)$ *can be decomposed as* $\Delta v_T(S) = \sum_{S' \subseteq S} I_{T \cup S'}$.

• *Proof:*

$$
\begin{aligned}
\Delta v_T(S) &= \sum_{L \subseteq T} (-1)^{|T|-|L|} v(L \cup S) \\
&= \sum_{L \subseteq T} (-1)^{|T|-|L|} \sum_{K \subseteq L \cup S} I_K \\
&= \sum_{L \subseteq T} (-1)^{|T|-|L|} \sum_{L' \subseteq L} \sum_{S' \subseteq S} I_{L' \cup S'} \quad \text{\% since } L \cap S = \emptyset \\
&= \sum_{S' \subseteq S} \left[\sum_{L \subseteq T} (-1)^{|T|-|L|} \sum_{L' \subseteq L} I_{L' \cup S'} \right] \\
&= \sum_{S' \subseteq S} \left[\sum_{L' \subseteq T} \sum_{\substack{L \subseteq T \\ L \supseteq L'}} (-1)^{|T|-|L|} I_{L' \cup S'} \right] \\
&= \sum_{S' \subseteq S} \left[\underbrace{I_{S' \cup T}}_{\text{when } L'=T} + \underbrace{\sum_{L' \subsetneq T} \left(\sum_{l=|L'|}^{|T|} \binom{|T|-|L'|}{l-|L'|} (-1)^{|T|-|L|} I_{L' \cup S'} \right)}_{\text{when } L' \subsetneq T} \right] \\
&= \sum_{S' \subseteq S} \left[I_{S' \cup T} + \sum_{L' \subsetneq T} \left(I_{L' \cup S'} \cdot \underbrace{\sum_{l=|L'|}^{|T|} \binom{|T|-|L'|}{l-|L'|} (-1)^{|T|-|L|}}_{=0} \right) \right] \quad \text{\% according to Lemma 1} \\
&= \sum_{S' \subseteq S} I_{S' \cup T}
\end{aligned}
$$

Property 1 (Connection to the Shapley interaction index) *Given a subset of input variables $T \subseteq N$, $I_T^{Shapley} = \sum_{S \subseteq N \setminus T} \frac{|S|!(|N|-|S|-|T|)!}{(|N|-|T|+1)!} \Delta v_T(S)$ denotes the Shapley interaction index (Grabisch and Roubens, 1999) of T. The Shapley interaction index can be represented as $I_T^{Shapley} = \sum_{S \subseteq N \setminus T} \frac{1}{|S|+1} I_{S \cup T}$.*

• *Proof:*

$$
\begin{aligned}
I_T^{\text{Shapley}} &= \sum_{S \subseteq N \setminus T} \frac{|S|!(|N|-|S|-|T|)!}{(|N|-|T|+1)!} \Delta v_T(S) \\
&= \frac{1}{|N|-|T|+1} \sum_{m=0}^{|N|-|T|} \frac{1}{\binom{|N|-|T|}{m}} \sum_{\substack{S \subseteq N \setminus T \\ |S|=m}} \Delta v_T(S) \\
&= \frac{1}{|N|-|T|+1} \sum_{m=0}^{|N|-|T|} \frac{1}{\binom{|N|-|T|}{m}} \sum_{\substack{S \subseteq N \setminus T \\ |S|=m}} \left[\sum_{L \subseteq S} I_{L \cup T} \right] \\
&= \frac{1}{|N|-|T|+1} \sum_{L \subseteq N \setminus T} \sum_{m=|L|}^{|N|-|T|} \frac{1}{\binom{|N|-|T|}{m}} \sum_{\substack{S \subseteq N \setminus T \\ |S|=m \\ S \supseteq L}} I_{L \cup T} \\
&= \frac{1}{|N|-|T|+1} \sum_{L \subseteq N \setminus T} \sum_{m=|L|}^{|N|-|T|} \frac{1}{\binom{|N|-|T|}{m}} \binom{|N|-|L|-|T|}{m-|L|} I_{L \cup T} \\
&= \frac{1}{|N|-|T|+1} \sum_{L \subseteq N \setminus T} I_{L \cup T} \underbrace{\sum_{k=0}^{|N|-|L|-|T|} \frac{1}{\binom{|N|-|T|}{|L|+k}} \binom{|N|-|L|-|T|}{k}}_{w_L}
\end{aligned}
$$

Then, we leverage the following properties of combinatorial numbers and the Beta function to simplify the term $w_L = \sum_{k=0}^{|N|-|L|-1} \frac{1}{\binom{|N|-1}{|L|+k}} \cdot \binom{|N|-|L|-1}{k}$.

(i) A property of combinitorial numbers. $m \cdot \binom{n}{m} = n \cdot \binom{n-1}{m-1}$.

(ii) The definition of the Beta function. For $p, q > 0$, the Beta function is defined as $B(p,q) = \int_0^1 x^{p-1}(1-x)^{1-q}dx$.

(iii) Connections between combinitorial numbers and the Beta function.

- When $p, q \in \mathbb{Z}^+$, we have $B(p,q) = \frac{1}{q \cdot \binom{p+q-1}{p-1}}$.
- For $m, n \in \mathbb{Z}^+$ and $n > m$, we have $\binom{n}{m} = \frac{1}{m \cdot B(n-m+1, m)}$.

Hence, we leverage the properties of combinitorial numbers and the Beta function to simplify w_L.

$$
\begin{aligned}
w_L &= \sum_{k=0}^{|N|-|L|-|T|} \frac{1}{\binom{|N|-|T|}{|L|+k}} \binom{|N|-|L|-|T|}{k} \\
&= \sum_{k=0}^{|N|-|L|-|T|} \binom{|N|-|L|-|T|}{k} \cdot \Big(|L|+k\Big) \cdot B\Big(|N|-|L|-|T|-k+1, |L|+k\Big) \\
&= \sum_{k=0}^{|N|-|L|-|T|} |L| \cdot \binom{|N|-|L|-|T|}{k} \cdot B\Big(|N|-|L|-|T|-k+1, |L|+k\Big) \qquad \cdots ① \\
&\quad + \sum_{k=0}^{|N|-|L|-|T|} k \cdot \binom{|N|-|L|-|T|}{k} \cdot B\Big(|N|-|L|-|T|-k+1, |L|+k\Big) \qquad \cdots ②
\end{aligned}
$$

Then, we solve ① and ② respectively. For ①, we have

$$
\begin{aligned}
① &= \int_0^1 |L| \sum_{k=0}^{|N|-|L|-|T|} \binom{|N|-|L|-|T|}{k} \cdot x^{|N|-|L|-|T|-k} \cdot (1-x)^{|L|+k-1}\, dx \\
&= \int_0^1 |L| \cdot \underbrace{\left[\sum_{k=0}^{|N|-|L|-|T|} \binom{|N|-|L|-|T|}{k} \cdot x^{|N|-|L|-|T|-k} \cdot (1-x)^{k} \right]}_{=1} \cdot (1-x)^{|L|-1}\, dx \\
&= \int_0^1 |L| \cdot (1-x)^{|L|-1}\, dx = 1
\end{aligned}
$$

For ②, we have

$$
\begin{aligned}
② &= \sum_{k=1}^{|N|-|L|-|T|} (|N|-|L|-|T|) \binom{|N|-|L|-|T|-1}{k-1} \cdot B\Big(|N|-|L|-|T|-k+1, |L|+k\Big) \\
&= (|N|-|L|-|T|) \sum_{k'=0}^{|N|-|L|-|T|-1} \binom{|N|-|L|-|T|-1}{k'} \cdot B\Big(|N|-|L|-|T|-k', |L|+k'+1\Big) \\
&= (|N|-|L|-|T|) \int_0^1 \sum_{k'=0}^{|N|-|L|-|T|-1} \binom{|N|-|L|-|T|-1}{k'} \cdot x^{|N|-|L|-|T|-k'-1} \cdot (1-x)^{|L|+k'}\, dx \\
&= (|N|-|L|-|T|) \int_0^1 \underbrace{\left[\sum_{k'=0}^{|N|-|L|-|T|-1} \binom{|N|-|L|-|T|-1}{k'} \cdot x^{|N|-|L|-|T|-k'-1} \cdot (1-x)^{k'} \right]}_{=1} \cdot (1-x)^{|L|}\, dx \\
&= (|N|-|L|-|T|) \int_0^1 (1-x)^{|L|}\, dx = \frac{|N|-|L|-|T|}{|L|+1}
\end{aligned}
$$

Hence, we have

$$w_L = ① + ② = 1 + \frac{|N| - |L| - |T|}{|L| + 1} = \frac{|N| - |T| + 1}{|L| + 1}$$

Therefore, we proved that $I_T^{\text{Shapley}} = \frac{1}{|N|-|T|+1} \sum_{L \subseteq N \setminus T} w_L \cdot I_{L \cup T} = \sum_{L \subseteq N \setminus T} \frac{1}{|L|+1} I_{L \cup T}$.

Property 2 (Connection to the Shapley Taylor interaction index) *Given a subset of input variables $T \subseteq N$, let $I_T^{\text{Shapley-Taylor}}$ denote the Shapley Taylor interaction index (Sundararajan et al., 2020) of order k for T. The Shapley taylor interaction index can be represented as the weighted sum of interaction utilities,* i.e. $I_T^{\text{Shapley-Taylor}} = I_T$ *if* $|T| < k$*;* $I_T^{\text{Shapley-Taylor}} = \sum_{S \subseteq N \setminus T} \binom{|S|+k}{k}^{-1} I_{S \cup T}$ *if* $|T| = k$*; and* $I_T^{\text{Shapley-Taylor}} = 0$ *if* $|T| > k$*.*

• *Proof:* By the definition of the Shapley Taylor interaction index,

$$I_T^{\text{Shapley-Taylor}(k)} = \begin{cases} \Delta v_T(\emptyset) & \text{if } |T| < k \\ \frac{k}{|N|} \sum_{S \subseteq N \setminus T} \frac{1}{\binom{|N|-1}{|S|}} \Delta v_T(S) & \text{if } |T| = k \\ 0 & \text{if } |T| > k \end{cases}$$

When $|T| < k$, by the definition of the interaction benefit in Equation (4) in the main paper, we have

$$I_T^{\text{Shapley-Taylor}(k)} = \Delta v_T(\emptyset) = \sum_{L \subseteq T} (-1)^{|T|-|L|} \cdot v(L) = I_T.$$

When $|T| = k$, we have

$$\begin{aligned}
I_T^{\text{Shapley-Taylor}(k)} &= \frac{k}{|N|} \sum_{S \subseteq N \setminus T} \frac{1}{\binom{|N|-1}{|S|}} \cdot \Delta v_T(S) \\
&= \frac{k}{|N|} \sum_{m=0}^{|N|-k} \sum_{\substack{S \subseteq N \setminus T \\ |S|=m}} \frac{1}{\binom{|N|-1}{|S|}} \cdot \Delta v_T(S) \\
&= \frac{k}{|N|} \sum_{m=0}^{|N|-k} \sum_{\substack{S \subseteq N \setminus T \\ |S|=m}} \frac{1}{\binom{|N|-1}{|S|}} \left[\sum_{L \subseteq S} I_{L \cup T} \right] \\
&= \frac{k}{|N|} \sum_{L \subseteq N \setminus T} \sum_{m=|L|}^{|N|-k} \frac{1}{\binom{|N|-1}{|S|}} \sum_{\substack{S \subseteq N \setminus T \\ |S|=m \\ S \supseteq L}} I_{L \cup T} \\
&= \frac{k}{|N|} \sum_{L \subseteq N \setminus T} \sum_{m=|L|}^{|N|-k} \frac{1}{\binom{|N|-1}{|S|}} \binom{|N| - |L| - k}{m - |L|} I_{L \cup T} \\
&= \frac{k}{|N|} \sum_{L \subseteq N \setminus T} I_{L \cup T} \underbrace{\sum_{m=0}^{|N|-|L|-k} \frac{1}{\binom{|N|-1}{|L|+m}} \binom{|N| - |L| - k}{m}}_{w_L}
\end{aligned}$$

Just like the proof of Property 1, we leverage the properties of combinitorial numbers and the Beta function to simplify w_L.

$$
\begin{aligned}
w_L &= \sum_{m=0}^{|N|-|L|-k} \frac{1}{\binom{|N|-1}{|L|+m}} \binom{|N|-|L|-k}{m} \\
&= \sum_{m=0}^{|N|-|L|-k} \binom{|N|-|L|-k}{m} \cdot \Big(|L|+m\Big) \cdot B\Big(|N|-|L|-m, |L|+m\Big) \\
&= \sum_{m=0}^{|N|-|L|-k} |L| \cdot \binom{|N|-|L|-k}{m} \cdot B\Big(|N|-|L|-m, |L|+m\Big) \qquad \cdots ① \\
&\quad + \sum_{m=0}^{|N|-|L|-k} m \cdot \binom{|N|-|L|-k}{m} \cdot B\Big(|N|-|L|-m, |L|+m\Big) \qquad \cdots ②
\end{aligned}
$$

Then, we solve ① and ② respectively. For ①, we have

$$
\begin{aligned}
① &= \int_0^1 |L| \cdot \sum_{m=0}^{|N|-|L|-k} \binom{|N|-|L|-k}{m} \cdot x^{|N|-|L|-m-1} \cdot (1-x)^{|L|+m-1} \, dx \\
&= \int_0^1 |L| \cdot \underbrace{\left[\sum_{m=0}^{|N|-|L|-k} \binom{|N|-|L|-k}{m} \cdot x^{|N|-|L|-m-k} \cdot (1-x)^m \right]}_{=1} \cdot x^{k-1} \cdot (1-x)^{|L|-1} \, dx \\
&= \int_0^1 |L| \cdot x^{k-1} \cdot (1-x)^{|L|-1} \, dx = |L| \cdot B(k, |L|) = \frac{1}{\binom{|L|+k-1}{k-1}}
\end{aligned}
$$

For ②, we have

$$
\begin{aligned}
② &= \sum_{m=1}^{|N|-|L|-k} (|N|-|L|-k) \cdot \binom{|N|-|L|-k-1}{m-1} \cdot B\Big(|N|-|L|-m, |L|+m\Big) \\
&= \sum_{m'=0}^{|N|-|L|-k-1} (|N|-|L|-k) \cdot \binom{|N|-|L|-k-1}{m'} \cdot B\Big(|N|-|L|-m'-1, |L|+m'+1\Big) \\
&= \int_0^1 (|N|-|L|-k) \sum_{m'=0}^{|N|-|L|-k-1} \binom{|N|-|L|-k-1}{m'} \cdot x^{|N|-|L|-m'-2} \cdot (1-x)^{|L|+m'} \, dx \\
&= \int_0^1 (|N|-|L|-k) \underbrace{\left[\sum_{m'=0}^{|N|-|L|-k-1} \binom{|N|-|L|-k-1}{m'} \cdot x^{|N|-|L|-m'-k-1} \cdot (1-x)^{m'} \right]}_{=1} \cdot x^{k-1} \cdot (1-x)^{|L|} \, dx \\
&= \int_0^1 (|N|-|L|-k) \cdot x^{k-1} \cdot (1-x)^{|L|} \, dx = (|N|-|L|-k) \cdot B(k, |L|+1) \\
&= \frac{|N|-|L|-k}{(|L|+1)\binom{|L|+k}{k-1}}
\end{aligned}
$$

Hence, we have

$$
\begin{aligned}
w_L =& ① + ② = \frac{1}{\binom{|L|+k-1}{k-1}} + \frac{|N|-|L|-k}{(|L|+1)\binom{|L|+k}{k-1}} \\
=& \frac{|L|!\cdot(k-1)!}{(|L|+k-1)!} + \frac{|N|-|L|-k}{|L|+1}\cdot\frac{(|L|+1)!\cdot(k-1)!}{(|L|+k)!} \\
=& \frac{|L|!\cdot(k-1)!}{(|L|+k-1)!} + \frac{|N|-|L|-k}{|L|+k}\cdot\frac{|L|!\cdot(k-1)!}{(|L|+k-1)!} \\
=& \left[1+\frac{|N|-|L|-k}{|L|+k}\right]\cdot\frac{|L|!\cdot(k-1)!}{(|L|+k-1)!} \\
=& \frac{|N|}{|L|+k}\cdot\frac{|L|!\cdot(k-1)!}{(|L|+k-1)!} \\
=& \frac{|N|}{k}\cdot\frac{|L|!\cdot k!}{(|L|+k)!} \\
=& \frac{|N|}{k}\cdot\frac{1}{\binom{|L|+k}{k}}
\end{aligned}
$$

Therefore, we proved that when $|T| = k$, $I_T^{\text{Shapley-Taylor}} = \frac{k}{|N|}\sum_{L\subseteq N\setminus T} w_L \cdot I_{L\cup T} = \frac{k}{|N|}\sum_{L\subseteq N\setminus T}\frac{|N|}{k}\cdot\frac{1}{\binom{|L|+k}{k}}\cdot I_{L\cup T} = \sum_{L\subseteq N\setminus T}\binom{|L|+k}{k}^{-1} I_{L\cup T}$.

D PROOFS OF THEOREMS

This section provides proofs of theorems in the main paper.

D.1 PROOF OF THEOREM 1

Theorem 1 ***(Faithfulness)*** *Let us consider a DNN v and a specific sample $\boldsymbol{x}$ with n input variables. Given any of the 2^n arbitrarily masked input $\boldsymbol{x}_S$, the causal graph defined in Eq. (4) can always well mimic the output of the DNN,* i.e., $\forall S \subseteq N, Y(\boldsymbol{x}_S) = v(\boldsymbol{x}_S)$*, when we set the causal patterns in the causal graph are $\Omega = 2^N = \{S|S\subseteq N\}$.*

Proof: According to the SCM in Eq. (4), we have $Y(\boldsymbol{x}_S) = \sum_{S'\in\Omega} U_{S'}\cdot C_{S'}(\boldsymbol{x}_S) = \sum_{S'\subseteq S} U_{S'}$. Therefore, the faithfulness requirement can be equivalently re-written as $\forall S \subseteq \mathcal{N}, v(\boldsymbol{x}_S) = \sum_{S'\subseteq S} U_{S'}$.

According to the definition of the Harsanyi dividend, we have $\forall S\subseteq\mathcal{N}$,

$$
\begin{aligned}
\sum_{S'\subseteq S} U_{S'} &= \sum_{S'\subseteq S}\sum_{L\subseteq S'}(-1)^{|S'|-|L|}v(\boldsymbol{x}_L) \\
&= \sum_{L\subseteq S}\sum_{S'\subseteq S:S'\supseteq L}(-1)^{|S'|-|L|}v(\boldsymbol{x}_L) \\
&= \sum_{L\subseteq S}\sum_{s'=|L|}^{|S|}\sum_{\substack{S'\subseteq S:S\supseteq L\\ |S'|=s'}}(-1)^{s'-|L|}v(\boldsymbol{x}_L) \\
&= \sum_{L\subseteq S} v(\boldsymbol{x}_L)\sum_{m=0}^{|S|-|L|}\binom{|S|-|L|}{m}(-1)^m = v(\boldsymbol{x}_S)
\end{aligned}
$$

D.2 PROOF OF THEOREM 2

Theorem 2 *Harsanyi dividends can be considered as causal patterns of the Shapley value.*

$$
\phi_i = \sum_{S\subseteq N\setminus\{i\}}\frac{1}{|S|+1}I_{S\cup\{i\}} \tag{11}
$$

In this way, the effect of an causal pattern consisting of m variables can be fairly assigned to the m variables. This connection has been proved in (Harsanyi, 1982).

• ***Proof:***

$$
\begin{aligned}
\text{right} &= \sum_{S\subseteq N\setminus\{i\}} \frac{1}{|S|+1} I_{S\cup\{i\}} \\
&= \sum_{S\subseteq N\setminus\{i\}} \frac{1}{|S|+1} \left[\sum_{L\subseteq S} (-1)^{|S|+1-|L|} v(L) + \sum_{L\subseteq S} (-1)^{|S|-|L|} v(L\cup\{i\}) \right] \\
&= \sum_{S\subseteq N\setminus\{i\}} \frac{1}{|S|+1} \sum_{L\subseteq S} (-1)^{|S|-|L|} \left[v(L\cup\{i\}) - v(L) \right] \\
&= \sum_{L\subseteq N\setminus\{i\}} \sum_{K\subseteq N\setminus L\setminus\{i\}} \frac{(-1)^{|K|}}{|K|+|L|+1} \left[v(L\cup\{i\}) - v(L) \right] \qquad \text{\% Let } K = S\setminus L \\
&= \sum_{L\subseteq N\setminus\{i\}} \left(\sum_{k=0}^{n-1-|L|} \frac{(-1)^k}{k+|L|+1} \binom{n-1-|L|}{k} \right) \left[v(L\cup\{i\}) - v(L) \right] \qquad \text{\% Let } k = |K| \\
&= \sum_{L\subseteq N\setminus\{i\}} \frac{|L|!(n-1-|L|)!}{n!} \left[v(L\cup\{i\}) - v(L) \right] \qquad \text{\% by the property of combinitorial number} \\
&= \phi_i = \text{left}
\end{aligned}
$$

D.3 Proof of Remark2, Theorem 3, and Theorem 4

Remark 2 *Let us consider a function with a single causal pattern $f(\boldsymbol{x}_S) = w_S \prod_{j\in S}(x_j - \delta_j)$. Accordingly, ground-truth baseline values of variables are obviously $\{\delta_j\}$, because setting any variable $\forall j \in S,\ x_j = \delta_j$ will deactivate this pattern. Given the correct baseline values $b_j^* = \delta_j$, we can use a single causal pattern to regress $f(\boldsymbol{x}_S)$,* i.e., *$I_S = f(\boldsymbol{x}_S)$, $\forall\, S' \neq S, I_{S'} = 0$.*

Theorem 3 *For the function $f(\boldsymbol{x}_S) = w_S \prod_{j\in S}(x_j - \delta_j)$, if we use m' incorrect baseline values $\{b'_j | b'_j \neq \delta_j\}$ to replace correct ones to compute causal effects, then the function will be explained to contain at most $2^{m'}$ causal patterns.*

Theorem 4 *If we use m' incorrect baseline values to compute causal effects in the function $f(\boldsymbol{x}_S) = w_S \prod_{j\in S}(x_j - \delta_j)$, a total of $\binom{m'}{k-|S|+m'}$ causal patterns of the k-th order emerge, $k \geq |S| - m'$. A causal pattern of the k-th order means that this causal pattern represents the AND relationship between k variables.*

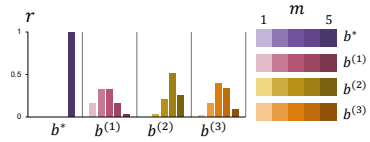
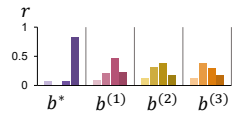
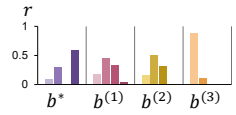
• ***Theoretical proof:*** Without loss of generality, let us consider an input sample $\boldsymbol{x}$, with $\forall j \in S, x_j \neq \delta_j$. Based on the ground-truth baseline value $\{\delta_j\}$, we have

(1) $v(\boldsymbol{x}_S) = f(\boldsymbol{x}_S) = w_S \prod_{j\in S}(x_j - \delta_j) \neq 0$,

(2) $\forall S' \subsetneq S, v(\boldsymbol{x}_{S'}) = w_S \prod_{j\in S'}(x_j - \delta_j) \prod_{k\in S\setminus S'}(\delta_k - \delta_k) = 0$,
Accordingly, we have $I_S = \sum_{S'\subseteq S}(-1)^{|S|-|S'|} v(\boldsymbol{x}_{S'}) = v(\boldsymbol{x}_S) \neq 0$. For $S' \subsetneq S$, we have $I_{S'} = \sum_{L\subseteq S'}(-1)^{|S'|-|L|} v(\boldsymbol{x}_L) = \sum_{L\subseteq S'} 0 = 0$.

(3) $\forall S' \neq S$, let $S' = L \cup M$, where $L \subseteq S$ and $M \cap S = \emptyset$. Then, we have

Table 9: Comparison between ground-truth baseline values and incorrect baseline values. The last column shows ratios of causal patterns of different orders $r_m = \frac{\sum_{S \subseteq N, |S|=m} |I_S|}{\sum_{S \subseteq N, S \neq \emptyset} |I_S|}$. We consider interactions of input samples that activate causal patterns. We find that when models/functions contain a single complex collaborations between multiple variables (*i.e.* high-order causal patterns), incorrect baseline values usually generate a mixture of many low-order causal patterns. In comparison, ground-truth baseline values lead to sparse and high-order causal patterns.

Functions ($\forall i \in N, i \in \{0,1\}$)	Baseline values $\boldsymbol{b}$	Ratios $\boldsymbol{r}$
$f(\boldsymbol{x}) = x_1 x_2 x_3 x_4 x_5$ $\boldsymbol{x} = [1,1,1,1,1]$	ground truth: $\boldsymbol{b}^* = [0,0,0,0,0]$ incorrect: $\boldsymbol{b}^{(1)} = [0.5,0.5,0.5,0.5,0.5]$ incorrect: $\boldsymbol{b}^{(2)} = [0.1,0.2,0.6,0.0,0.1]$ incorrect: $\boldsymbol{b}^{(3)} = [0.7,0.1,0.3,0.5,0.1]$	
$f(\boldsymbol{x}) = \textit{sigmoid}(5x_1x_2x_3 + 5x_4 - 7.5)$ $\boldsymbol{x} = [1,1,1,1]$	ground truth: $\boldsymbol{b}^* = [0,0,0,0]$ incorrect: $\boldsymbol{b}^{(1)} = [0.5,0.5,0.5,0.5]$ incorrect: $\boldsymbol{b}^{(2)} = [0.6,0.4,0.7,0.3]$ incorrect: $\boldsymbol{b}^{(3)} = [0.3,0.6,0.5,0.8]$	
$f(\boldsymbol{x}) = x_1(x_2 + x_3 - x_4)^3$ $\boldsymbol{x} = [1,1,1,0]$	ground truth: $\boldsymbol{b}^* = [0,0,0,1]$ incorrect: $\boldsymbol{b}^{(1)} = [0.5,0.5,0.5,0.5]$ incorrect: $\boldsymbol{b}^{(2)} = [0.2,0.3,0.6,0.1]$ incorrect: $\boldsymbol{b}^{(3)} = [1.0,0.3,1.0,0.1]$	

$$
\begin{aligned}
I_{S'} &= \sum_{T \subseteq S'} (-1)^{|S'|-|T|} v(\boldsymbol{x}_T) \\
&= \sum_{\substack{L' \subseteq L \\ L' \neq \emptyset}} (-1)^{|S'|-|L'|} v(\boldsymbol{x}_{L'}) + \sum_{\substack{M' \subseteq M \\ M' \neq \emptyset}} (-1)^{|S'|-|M'|} \underbrace{v(\boldsymbol{x}_{M'})}_{=v(\boldsymbol{x}_\emptyset)=0} \\
&\quad + \sum_{\substack{L' \subseteq L, M' \subseteq M \\ L' \neq \emptyset, M' \neq \emptyset}} (-1)^{|S|-|L'|-|M'|} \underbrace{v(\boldsymbol{x}_{L' \cup M'})}_{=v(L')} + (-1)^{|S'|} \underbrace{v(\boldsymbol{x}_\emptyset)}_{=0} \\
&= \sum_{\substack{L' \subseteq L \\ L' \neq \emptyset}} (-1)^{|S'|-|L'|} v(\boldsymbol{x}_{L'}) + \sum_{\substack{L' \subseteq L, M' \subseteq M \\ L' \neq \emptyset, M' \neq \emptyset}} (-1)^{|S|-|L'|-|M'|} v(\boldsymbol{x}_{L'}) \\
&= (-1)^{|S'|-|S|} v(\boldsymbol{x}_S) + \sum_{\substack{M' \subseteq M \\ M' \neq \emptyset}} (-1)^{|S'|-|S|-|M'|} v(\boldsymbol{x}_S) \qquad \text{\% } v(\boldsymbol{x}_{L'}) \neq 0 \text{ only if } L' = S \\
&= \sum_{M' \subseteq M} (-1)^{|S'|-|S|-|M'|} v(\boldsymbol{x}_S) = 0
\end{aligned}
$$

Therefore, there is only one causal pattern with non-zero effect I_S.

In comparison, if we use m' incorrect baseline values $\{\delta'_j\}$, where $\sum_{j \in S} \mathbb{1}_{\delta'_j \neq \delta_j} = m'$, then the function will be explained to contain at most $2^{m'}$ causal patterns. For the simplicity of notations, let $S = \{1, 2, ..., m\}$, and $\delta'_1 = \delta_1 + \epsilon_1$, ..., $\delta'_{m'} = \delta_{m'} + \epsilon_{m'}$, where $\epsilon_1, ..., \epsilon_{m'} \neq 0$. Let $T = \{1, 2, \ldots, m'\}$. In this case, we have

(1) $v(\boldsymbol{x}_S) = f(\boldsymbol{x}_S) \neq 0$

(2) $\forall S' \subsetneq S, |S'| < m - m', v(\boldsymbol{x}_{S'}) = w_S \prod_{j \in S'} (x_j - \delta_j) \prod_{l \in S \setminus S'} (\delta'_l - \delta_l)$. Because $|S| - |S'| > m'$, there is at least one variable with ground-truth baseline value in $S \setminus S'$. Therefore, $v(\boldsymbol{x}_{S'}) = 0$. Furthermore, $I_{S'} = \sum_{L \subseteq S'} (-1)^{|S'|-|L|} v(\boldsymbol{x}_L) = 0$

(3) $\forall S' \subsetneq S, |S'| = k \geq m - m', v(\boldsymbol{x}_{S'}) = w_S \prod_{j \in S'} (x_j - \delta_j) \prod_{l \in S \setminus S'} (\delta'_l - \delta_l)$. If $S \setminus T \subseteq S'$, then

$S \setminus S' \subseteq T$ and $v(\boldsymbol{x}_{S'}) \neq 0$. Otherwise, $v(\boldsymbol{x}_{S'}) = 0$. Then,

$$\begin{aligned}
I_{S'} &= \sum_{L \subseteq S'} (-1)^{|S'|-|L|} v(\boldsymbol{x}_L) \\
&= \sum_{L \subseteq S', |L| < m-m'} (-1)^{|S'|-|L|} v(\boldsymbol{x}_L) + \sum_{L \subseteq S', L \geq m-m'} (-1)^{|S'|-|L|} v(\boldsymbol{x}_L) \\
&= 0 + \sum_{L \subseteq S', L \geq m-m', L \supseteq S \setminus T} (-1)^{|S'|-|L|} v(\boldsymbol{x}_L) + \sum_{L \subseteq S', L \geq m-m', L \not\supseteq S \setminus T} (-1)^{|S'|-|L|} v(\boldsymbol{x}_L) \\
&= \sum_{L \subseteq S', L \geq m-m', L \supseteq S \setminus T} (-1)^{|S'|-|L|} v(\boldsymbol{x}_L)
\end{aligned}$$

If the above $I_{S'} = 0$, it indicates that $S \setminus T \nsubseteq S'$. In this case, there is no subset $L \subseteq S'$ s.t. $S \setminus T \subseteq L$. In other words, only if $S \setminus T \subseteq S', I_{S'} \neq 0$. In this way, a total of $\binom{m'}{k-(|S|-m')}$ causal patterns of the k-th order emerge, where the order k of a causal pattern means that this causal pattern S' contains $k = |S'|$ variables. There are totally $\sum_{k=|S|-m'}^{m} \binom{m'}{k-(|S|-m')} = 2^{m'}$ causal patterns in $\boldsymbol{x}$.

For example, if the input $\boldsymbol{x}$ is given as follows,

$$x_i = \begin{cases} \delta_i + 2\epsilon_i, & i \in T = \{1, \ldots, m'\} \\ \delta_i + \epsilon_i, & i \in S \setminus T = \{m'+1, \ldots, m\} \end{cases}$$

where $\epsilon_i \neq 0$ are arbitrary non-zero scalars. In this case, we have $\forall S' \subseteq T, I_{S' \cup \{m'+1,\ldots,m\}} = \epsilon_1 \epsilon_2 ... \epsilon_m \neq 0$. Besides, if $\{m'+1, ..., m\} \nsubseteq S'$, we have $I_{S'} = 0$. In this way, there are totally $2^{m'}$ causal patterns in $\boldsymbol{x}$.

• ***Experimental verification:*** We further conducted experiments to show that the incorrect setting of baseline values makes a model/function consisting of high-order causal patterns be mistakenly explained as a mixture of low-order and high-order causal patterns. To show this phenomenon, we compare causal patterns computed using ground-truth baseline values and incorrect baseline values in Table 9, and the results verify our conclusion. We find that when models/functions contain complex collaborations between multiple variables (*i.e.* high-order causal patterns), incorrect baseline values usually generate fewer high-order causal patterns and more low-order causal patterns than ground-truth baseline values. In other words, the model/function is explained as massive low-order causal patterns. In comparison, ground-truth baseline values lead to sparse and high-order salient patterns.

E PROVING THAT MASKING INPUT VARIABLES REMOVES CAUSAL EFFECTS

In this section, we prove that for the causal pattern $S \ni i$, if the input variable i is masked, then the causal effect $w_S = 0$.

Proof: let $S = S' \cup \{i\}$. If $i \in S$ is masked, then $\forall L \; s.t. \; i \notin L, \boldsymbol{x}_L = \boldsymbol{x}_{L \cup \{i\}}$. Therefore, $v(L \cup \{i\}) = v(L)$. According to the definition of Harsanyi dividend (Harsanyi, 1982), we have

$$\begin{aligned}
w_S &= \sum_{L \subseteq S} (-1)^{|S|-|L|} v(L) \\
&= \sum_{L \subseteq (S' \cup \{i\})} (-1)^{|S'|+1-|L|} v(L) \\
&= \sum_{L \subseteq S'} (-1)^{|S'|+1-|L|} v(L) + \sum_{L \subseteq S'} (-1)^{|S'|-|L|} v(L \cup \{i\}) \\
&= \sum_{L \subseteq S'} (-1)^{|S'|+1-|L|} v(L) + \sum_{L \subseteq S'} (-1)^{|S'|-|L|} v(L) \\
&= \sum_{L \subseteq S'} \left((-1)^{|S'|+1-|L|} + (-1)^{|S'|-|L|} \right) v(L) \\
&= \sum_{L \subseteq S'} (-1+1)(-1)^{|S'|-|L|} v(L) \\
&= 0
\end{aligned}$$

F MORE EXPERIMENTAL DETAILS AND RESULTS

F.1 THE VISUALIZATION OF CAUSAL PATTERNS

This section provides the visualization of causal patterns. Figure 4 visualizes the distribution of textual pixels j that forming causal patterns with a certain pixel i In Figure 4, the pixel value in the heatmap is computed as $p(j|i) = \mathbb{E}_{S\in\Omega}[\mathbb{1}_{j\in S}]$, where $i \in N$ and Ω denotes the set of patterns S whose values $|\Delta v_i(S)|$ are relatively large among all $S \subseteq N\backslash\{i\}$. $\Delta v_i(S)$ is defined in Equation (11) of the main paper.

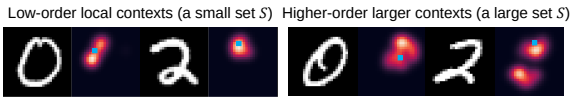


Figure 4: Visualization of the distribution of contextual pixels j that form causal patterns with a certain pixel i (the blue dot).

Due to the exponential computational cost of enumerating all causal patterns to obtain the salient ones in Ω, we used the following greedy strategy for approximation. We first randomly sample an input variable i and a context $S \subseteq N \setminus \{i\}$, and compute the corresponding value of $|\Delta v_i(S)|$. We sample multiple times and find the pair of (i, S) that yields the largest value of $|\Delta v_i(S)|$. Then, we add/remove some variables to/from the context S to obtain a larger $|\Delta v_i(S)|$. Alg. 1 shows the pseudo-code of this algorithm.

F.2 VERIFICATION OF THE SPARSITY OF THE CAUSAL GRAPH

In this subsection, we conducted experiments to verify the sparsity of causal effects, which is introduced in Remark 1. To this end, we computed causal effects U_S of all 2^n causal patterns encoded by a DNN. Specifically, we trained a three-layer MLP on the income dataset and computed causal effects in the model. Figure 5 shows the distribution of absolute causal effects $|U_S|$ of causal patterns in the first five samples of each category of the income dataset. These results show that most causal patterns had insignificant causal effects, $U_S \approx 0$. Only a few causal patterns had salient causal effects.

Moreover, we also conducted experiments to demonstrate the universality of this phenomenon. We trained the five-layer MLP, CNN, LSTM, ResNet-32, and VGG-16 on the UCI census income dataset, the UCI TV news channel commercial detection dataset, the SST-2 dataset, and the MNIST dataset, respectively. Figure 6 shows the absolute causal effects U_S in the descending order. These results show that various DNNs learned on different tasks could be explained by a sparse causal graph.

F.3 VERIFICATION OF USING CAUSAL PATTERNS TO EXAMINE THE STATE OF INPUT VARIABLES

In this subsection, we conducted experiments to verify that causal patterns reflect the states of removing existing patterns. Given causal effects I_S in the normal input image and causal effects

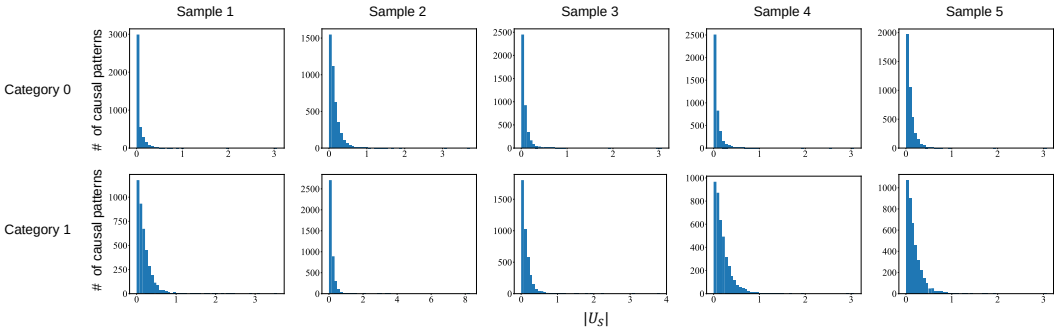


Figure 5: Histograms of absolute causal effects of causal patterns encoded by the three-layer MLP trained on the income dataset.

Algorithm 1: The approximate algorithm to causal patterns whose $\Delta v_i(S)$ are large

Input: The set of input variables N, the reward function (model) $v(\cdot)$, the sampling number of contexts t, the sampling number of noisy variables m, the convergence threshold ϵ, the number k_{max}

Output: a specific input variable i, a set Ω consisting of k_{max} causal patterns

for *all* $i \in N$ **do**
  Randomly sample a set of subsets $\{P_1, P_2, \ldots, P_t\} \subseteq 2^{N \setminus \{i\}}$
  $S_i = \arg\max_P |\Delta v_i(P)|$
end
$i_{\text{max}}, S_{\text{max}} = \arg\max_{(i, S_i)} |\Delta v_i(S_i)|$
Initialize $\Omega = \emptyset$
Randomly sample a set of subsets $\{M_1, M_2, \ldots, M_{k_{\text{max}}}\} \subseteq 2^{N \setminus S_{\text{max}} \setminus \{i_{\text{max}}\}}$ with the size m, *i.e.* $\forall k \in \{1, 2, \ldots, k_{\text{max}}\}, |M_k| = m$
for k *from 1 to* k_{max} **do**
  Let $S^{(k)} = S_{\text{max}} \cup M_k, d^{(k)} = |\Delta v_{i_{\text{max}}}(S^{(k)})|$
  Initial $S_{\text{max}}^{(k)} = S^{(k)}, d_{\text{max}}^{(k)} = d^{(k)}$
  while *True* do
    Let $d_{\text{tmp}}^{(k)} = d_{\text{max}}^{(k)}$
    for *all* $j \in N \setminus \{i\}$ **do**
      if $j \in S_{max}^{(k)}$ **then**
        if $|\Delta v_{i_{max}}(S_{max}^{(k)} \setminus \{j\})| > d_{max}^{(k)}$ **then**
          $S_{\text{max}}^{(k)} \leftarrow S_{\text{max}}^{(k)} \setminus \{j\}$
          $d_{\text{max}}^{(k)} \leftarrow |\Delta v_{i_{\text{max}}}(S_{\text{max}}^{(k)} \setminus \{j\})|$
        end
      else
        if $|\Delta v_{i_{max}}(S_{max}^{(k)} \cup \{j\})| > d_{max}^{(k)}$ **then**
          $S_{\text{max}}^{(k)} \leftarrow S_{\text{max}}^{(k)} \cup \{j\}$
          $d_{\text{max}}^{(k)} \leftarrow |\Delta v_{i_{\text{max}}}(S_{\text{max}}^{(k)} \cup \{j\})|$
        end
      end
    end
    if $|d_{max}^{(k)} - d_{tmp}^{(k)}| < \epsilon$ **then**
      Break
    end
  end
  $\Omega \leftarrow \Omega \cup \{S_{\text{max}}^{(k)}\}$
end
return i_{max}, Ω

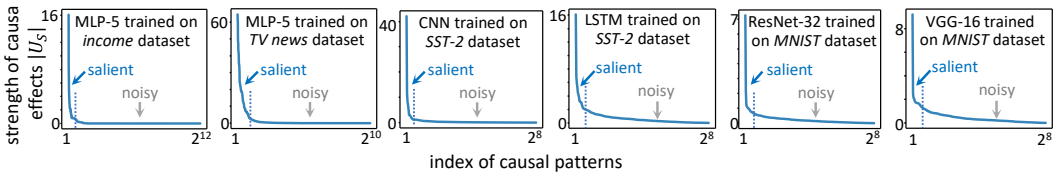


Figure 6: Absolute causal effects of different causal patterns shown in a descending order, which shows that sparse causality is universal for various DNNs.

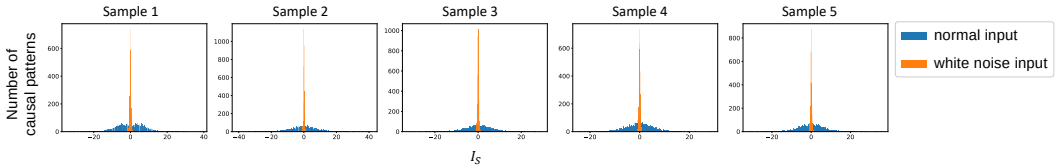


Figure 7: The distribution of causal effects I_S in the normal input, and the distribution of causal effects $I_S^{(\text{noise})}$ in the white noise input.

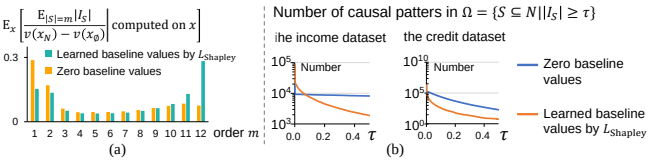


Figure 8: Our method successfully (a) reduced effects of low-order causal patterns on the MLP trained on the census dataset, and (b) reduced the number of salient patterns.

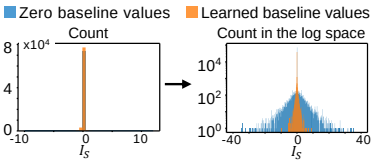


Figure 9: Distribution of causal effects I_S of causal patterns in 20 samples in the credit dataset.

$I_S^{(\text{noise})}$ in the white noise input, we compared their distributions in Figure 7. Note that we assumed that the white noise input naturally contained information for classification than the normal input image. We found that most causal effects in the white noise input were close to zero, and there were few salient causal patterns. Besides, we computed the average strength of causal effects in the above two inputs. In the normal input, the average strength of causal effects $\mathbb{E}_{S\subseteq N}|I_S| = 5.5285$, while in the white noise input, the average strength was much smaller, $\mathbb{E}_{S\subseteq N}|I_S^{(\text{noise})}| = 0.2321$. These results indicated that salient causal patterns could reflect the information encoded in the input.

F.4 Effects of the proposed method on multi-order Shapley values and multi-order marginal benefits

In this section, we conducted experiments to verify that baseline values $\boldsymbol{b}$ learned by the two proposed loss functions in Eq. (8) could effectively reduced causal effects of low-order causal patterns in Eq. (6). Besides, we also conducted experiments to show the effect of the proposed loss in Section 3 on multi-order Shapley values $\phi_i^{(m)}$ and multi-order marginal benefits $\Delta v_i(S)$.

To this end, we computed the metric $\mathbb{E}_{\boldsymbol{x}}[(\mathbb{E}_{|S|=m}|I_S|)/(v(\boldsymbol{x}_N) - v(\boldsymbol{x}_\emptyset))]$ to measure the relative strength of causal patterns of a specific order m, in order to evaluate the effectiveness of baseline values. Fig. 8(a) shows that compared to zero baseline values, **our method effectively reduced low-order causal patterns.** In addition, Fig. 9 and Fig. 8(b) verify that the loss L_{Shapley} in Eq. (8) **reduced the number of salient causal patterns in** Ω, which means L_{Shapley} avoided the exponential number of causal patterns caused by incorrect baseline values.

In order to examine the effect of the proposed loss in Section 3 on multi-order Shapley values $\phi_i^{(m)}$ and multi-order marginal benefits $\Delta v_i(S)$, we compared multi-order Shaplely values $\phi_i^{(m)}$ *w.r.t.* the function proposed in (Tsang et al., 2018) between the following three settings, (1) using zero baseline values, (2) using the baseline values learned with L_{Shapley}, and (3) using the baseline values learned with L_{marginal}, respectively. Similarly, we also compared marginal benefits $\Delta_i v(S)$ *w.r.t.* the function in (Tsang et al., 2018) between the same three settings of baseline values. Figure 10 shows the ratio of multi-order Shapley values and multi-order marginal benefits. Both L_{shapley} and L_{marginal} effectively reduced low-order Shapley values and low-order marginal benefits.

F.5 Discussion about the setting of ground-truth baseline values.

This section discusses the ground truth of baseline values of synthetic functions in Section 4.1 of the main paper. In order to verify the correctness of the learned baseline values, we conducted experiments on synthetic functions with ground-truth baseline values. We randomly generated 100 functions whose causal patterns and ground truth of baseline values could be easily determined. As

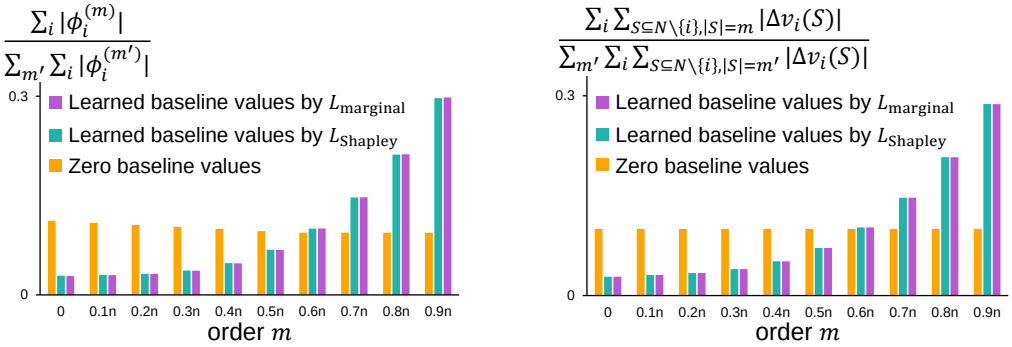


Figure 10: Our method successfully reduced the influence of low-order Shapley values and low-order marginal benefits, computed on a function in (Tsang et al., 2018).

Table 10: Examples of synthetic functions and their ground-truth baseline values.

Functions ($\forall i \in N, x_i \in \{0,1\}$)	The ground truth of baseline values
$-0.185x_1(x_2+x_3)^{2.432} - x_4x_5x_6x_7x_8x_9x_{10}x_{11}$	$b_i^* = 0$ for $i \in \{1,\dots,11\}$
$-\textit{sigmoid}(-4x_1 - 4x_2 - 4x_3 + 2) - 0.011x_4(x_5+x_6+x_7+x_8+x_9+x_{10}+x_{11})^{2.341}$	$b_i^* = 1$ for $i \in \{1,2,3\}, b_i^* = 0$ for $i \in \{4,\dots,11\}$
$0.172x_1x_2x_3(x_4+x_5)^{2.543} - 0.171x_6(x_7+x_8)^{2.545} - 0.093(x_9+x_{10}+x_{11})^{2.157}$	$b_i^* = 0$ for $i \in \{1,\dots,11\}$
$-\textit{sigmoid}(5x_1 + 5x_2x_3x_4 - 7.5) - \textit{sigmoid}(-8x_5x_6 + 8x_7 + 4) + x_8x_9x_{10}x_{11}$	$b_i^* = 1$ for $i = 7, b_i^* = 0$ for $i \in \{1,2,3,4,5,6,8,9,10,11\}$
$-x_1x_2x_3 + 0.156(x_4+x_5+x_6)^{1.693} - x_7x_8x_9x_{10}$	$b_i^* = 0$ for $i \in \{1,\dots,10\}$
$\textit{sigmoid}(-3x_1x_2x_3 + 1.5) + 0.197x_4x_5(x_6+x_7+x_8)^{1.48} + x_9x_{10}x_{11}$	$b_i^* = 0$ for $i \in \{1,\dots,11\}$
$-\textit{sigmoid}(5x_1 - 5x_2x_3 + 2.5) - \textit{sigmoid}(3x_4x_5x_6x_7 - 1.5) - 0.365x_8(x_9+x_{10})^{1.453}$	$b_i^* = 1$ for $i = 1, b_i^* = 0$ for $i \in \{2,\dots,10\}$
$\textit{sigmoid}(4x_1 - 4x_2 - 4x_3 + 6) - \textit{sigmoid}(-6x_4x_5x_6x_7 + 3) - \textit{sigmoid}(6x_8 - 6x_9x_{10} + 3)$	$b_i^* = 1$ for $i \in \{1,8\}, b_i^* = 0$ for $i \in \{2,3,4,5,6,7,9,10\}$
$-x_1x_2x_3 + 0.205x_4(x_5+x_6)^{2.289} - 0.115(x_7+x_8+x_9)^{1.969} + x_{10}x_{11}x_{12}$	$b_i^* = 0$ for $i \in \{1,\dots,12\}$
$-\textit{sigmoid}(-3x_1x_2x_3 - 3x_4x_5 + 4.5) - \textit{sigmoid}(3x_6 + 3x_7 + 3x_8 - 1.5) + x_9x_{10}x_{11}$	$b_i^* = 1$ for $i \in \{6,7,8\}, b_i^* = 0$ for $i \in \{1,\dots,5,9,10,11\}$
$-\textit{sigmoid}(8x_1x_2 - 8x_3 - 8x_4 - 4) - 0.041(x_5+x_6+x_7+x_8)^{2.298} - \textit{sigmoid}(7x_9x_{10} + 7x_{11} - 10.5)$	$b_i^* = 1$ for $i \in \{3,4\}, b_i^* = 0$ for $i \in \{1,2,5,6,\dots,11\}$
$-\textit{sigmoid}(4x_1 - 4x_2 + 4x_3 - 6) - x_4x_5x_6x_7 - x_8x_9x_{10}x_{11}$	$b_i^* = 1$ for $i = 2, b_i^* = 0$ for $i \in \{1,3,4,\dots,11\}$
$\textit{sigmoid}(3x_1x_2 + 3x_3 - 3x_4 - 3x_5 - 3x_6 - 4.5) + \textit{sigmoid}(6x_7x_8 + 6x_9x_{10}x_{11} - 9)$	$b_i^* = 1$ for $i \in \{4,5,6\}, b_i^* = 0$ for $i \in \{1,2,3,7,8,9,10,11\}$
$-\textit{sigmoid}(-7x_1x_2 - 7x_3x_4 + 10.5) + \textit{sigmoid}(-5x_5 + 5x_6 - 5x_7x_8 - 5x_9 + 12.5) + x_{10}x_{11}x_{12}$	$b_i^* = 1$ for $i = 6, b_i^* = 0$ for $i \in \{1,2,3,4,5,7,8,\dots,12\}$
$\textit{sigmoid}(-6x_1 + 6x_2 + 6x_3 + 3) + 0.229x_4x_5x_6(x_7+x_8)^{2.124} + 0.070x_9(x_{10}+x_{11}+x_{12})^{2.418}$	$b_i^* = 1$ for $i \in \{2,3\}, b_i^* = 0$ for $i \in \{1,4,5,\dots,12\}$
$x_1x_2x_3x_4 - \textit{sigmoid}(-6x_5x_6 - 6x_7 + 9) + x_8x_9x_{10}x_{11}$	$b_i^* = 0$ for $i \in \{1,\dots,11\}$
$\textit{sigmoid}(-3x_1 + 3x_2 - 3x_3x_4 - 3x_5x_6 + 7.5) - 0.174x_7(x_8+x_9+x_{10})^{1.594}$	$b_i^* = 1$ for $i = 2, b_i^* = 0$ for $i \in \{1,3,4,\dots,10\}$
$-0.34x_1(x_2+x_3)^{1.557} - \textit{sigmoid}(5x_4x_5x_6 - 5x_7 - 5x_8 - 5x_9 - 5x_{10} - 5x_{11} - 2.5)$	$b_i^* = 1$ for $i \in \{7,\dots,11\}, b_i^* = 0$ for $i \in \{1,\dots,6\}$
$-\textit{sigmoid}(-8x_1x_2 + 8x_3 + 4) - x_4x_5x_6x_7 + 0.457x_8(x_9+x_{10})^{1.13}$	$b_i^* = 1$ for $i = 3, b_i^* = 0$ for $i \in \{1,2,4,5,6,7,8,9,10\}$
$\textit{sigmoid}(-6x_1 + 6x_2 - 6x_3 - 3) + \textit{sigmoid}(-6x_4x_5 - 6x_6 + 6x_7 + 9) + \textit{sigmoid}(4x_8x_9 - 4x_{10} - 2)$	$b_i^* = 1$ for $i \in \{1,3,7,10\}, b_i^* = 0$ for $i \in \{2,4,5,6,8,9\}$

Table 10 shows, the generated functions were composed of addition, subtraction, multiplication, exponentiation, and *sigmoid* operations.

The ground truth of baseline values in these functions was determined based on causal patterns between input variables. In order to represent absence states of variables, baseline values should activate as few salient patterns as possible, where activation states of causal patterns were considered as the most infrequent state. Thus, we first identified the activation states of causal patterns of variables, and the ground-truth of baseline values were set as values that inactivated causal patterns under different masks. We took the following examples to discuss the setting of ground-truth baseline values (in the following examples, $\forall i \in N, x_i \in \{0,1\}$ and $b_i^* \in \{0,1\}$).

• $f(\boldsymbol{x}) = x_1x_2x_3 + \textit{sigmoid}(x_4 + x_5 - 0.5) \cdots$. Let us just focus on the term of $x_1x_2x_3$ in $f(\boldsymbol{x})$. The activation state of this causal pattern is $x_1x_2x_3 = 1$ when $\forall i \in \{1,2,3\}, x_i = 1$. In order to inactivate the causal pattern, we set $\forall i \in \{1,2,3\}, b_i^* = 0$.

• $f(\boldsymbol{x}) = -x_1x_2x_3 + (x_4 + x_5)^3 + \cdots$. Let us just focus on the term of $-x_1x_2x_3$ in $f(\boldsymbol{x})$. The activation state of this causal pattern is $-x_1x_2x_3 = -1$ when $\forall i \in \{1,2,3\}, x_i = 1$. In order to inactivate the causal pattern, we set $\forall i \in \{1,2,3\}, b_i^* = 0$.

• $f(\boldsymbol{x}) = (x_1 + x_2 - x_3)^3 + \cdots$. Let us just focus on the term of $(x_1 + x_2 - x_3)^3$ in $f(\boldsymbol{x})$. The activation state of this causal pattern is $(x_1 + x_2 - x_3)^3 = 8$ when $x_1 = x_2 = 1, x_3 = 0$. In order to inactivate the causal pattern under different masks, we set $b_1^* = b_2^* = 0, b_3^* = 1$.

• $f(\boldsymbol{x}) = \textit{sigmoid}(3x_1x_2 - 3x_3 - 1.5) + \cdots$. Let us just focus on the term of $\textit{sigmoid}(3x_1x_2 - 3x_3 - 1.5)$ in $f(\boldsymbol{x})$. In this case, x_1, x_2, x_3 form a salient causal pattern because $\textit{sigmoid}(3x_1x_2 - 3x_3 - 1.5) > 0.5$ only if $x_1 = x_2 = 1$ and $x_3 = 0$. Thus, in order to inactivate causal patterns, ground-truth baseline values are set to $b_1^* = b_2^* = 0, b_3^* = 1$.

Ground-truth baseline values of functions in (Tsang et al., 2018). This section provides more details about ground-truth baseline values of functions proposed in (Tsang et al., 2018). We evalu-

Table 11: Functions in (Tsang et al., 2018) and their ground-truth baseline values.

Functions ($\forall i \in N, x_i \in \{0.001, 0.999\}$)	The ground truth of baseline values
$\pi^{x_1 x_2}\sqrt{2x_3} - \sin^{-1}(x_4) + \log(x_3 + x_5) - \frac{x_9}{x_{10}}\sqrt{\frac{x_7}{x_8}} - x_2 x_7$	$b_i^* = 0.999$ for $i \in \{5, 8, 10\}, b_i^* = 0.001$ for $i \in \{1, 2, 7, 9\}$
$\pi^{x_1 x_2}\sqrt{2\lvert x_3\rvert} - \sin^{-1}(0.5x_4) + \log(\lvert x_3 + x_5\rvert + 1) + \frac{x_9}{1+\lvert x_{10}\rvert}\sqrt{\frac{x_7}{1+\lvert x_8\rvert}} - x_2 x_7$	$b_i^* = 0.999$ for $i = 5, b_i^* = 0.001$ for $i \in \{1, 2, 7, 9\}$
$\exp\lvert x_1 - x_2\rvert + \lvert x_2 x_3\rvert - x_3^{2\lvert x_4\rvert} + \log(x_4^2 + x_5^2 + x_7^2 + x_8^2) + x_9 + \frac{1}{1+x_{10}^2}$	$b_i^* = 0.999$ for $i \in \{3, 5, 7, 8\}$
$\exp\lvert x_1 - x_2\rvert + \lvert x_2 x_3\rvert - x_3^{2\lvert x_4\rvert} + (x_1 x_4)^2 + \log(x_4^2 + x_5^2 + x_7^2 + x_8^2) + x_9 + \frac{1}{1+x_{10}^2}$	$b_i^* = 0.999$ for $i \in \{3, 5, 7, 8\}$
$\frac{1}{1+x_1^2+x_2^2+x_3^2} + \sqrt{\exp(x_4 + x_5)} + \lvert x_6 + x_7\rvert + x_8 x_9 x_{10}$	$b_i^* = 0.999$ for $i \in \{1, 2, 3\}, b_i^* = 0.001$ for $i \in \{4, 5, 8, 9, 10\}$
$\exp(\lvert x_1 x_2\rvert + 1) - \exp(\lvert x_3 + x_4\rvert + 1) + \cos(x_5 + x_6 - x_8) + \sqrt{x_8^2 + x_9^2 + x_{10}^2}$	$b_i^* = 0.999$ for $i \in \{8, 9, 10\}, b_i^* = 0.001$ for $i \in \{1, 2, 3, 4, 5, 6\}$
$(\arctan(x_1) + \arctan(x_2))^2 + \max(x_3 x_4 + x_6, 0) - \frac{1}{1+(x_4 x_5 x_6 x_7 x_8)^2} + \left(\frac{\lvert x_7\rvert}{1+\lvert x_9\rvert}\right)^5 + \sum_{i=1}^{10} x_i$	$b_i^* = 0.999$ for $i = 9, b_i^* = 0.001$ for $i \in \{1, 2, 3, 4, 5, 6, 7, 8\}$
$x_1 x_2 + 2^{x_3+x_5+x_6} + 2^{x_3+x_4+x_5+x_7} + \sin(x_7 \sin(x_8 + x_9)) + \arccos(0.9x_{10})$	$b_i^* = 0.001$ for $i \in \{1, 2, 3, 4, 5, 6\}$
$\tanh(x_1 x_2 + x_3 x_4)\sqrt{\lvert x_5\rvert} + \exp(x_5 + x_6) + \log((x_6 x_7 x_8)^2 + 1) + x_9 x_{10} + \frac{1}{1+\lvert x_{10}\rvert}$	$b_i^* = 0.001$ for $i \in \{6, 7, 8, 9, 10\}$
$\sinh(x_1 + x_2) + \arccos(\tanh(x_3 + x_5 + x_7)) + \cos(x_4 + x_5) + \sec(x_7 x_9)$	$b_i^* = 0.999$ for $i = 3, b_i^* = 0.001$ for $i \in \{1, 2, 4\}$

ated the correctness of the learned baseline values using functions proposed in (Tsang et al., 2018). Among all the 92 input variables in these functions, the ground truth of 61 variables could be determined and are reported in Table 11. Note that some variables cannot be 0 or 1 (*e.g.* x_8 cannot be zero in the first function), and we set $\forall i \in N, x_i \in \{0.001, 0.999\}$ for variables in these functions instead. Similarly, we set the ground truth of baseline values $\forall i \in N, b_i^* \in \{0.001, 0.999\}$. Some variables did not collaborate/interact with other variables (*e.g.* x_4 in the first function), thereby having no causal patterns. We did not assign ground-truth baseline values for these individual variables, and these variables are not used for evaluation. Some variables formed more than one causal pattern with other variables, and had different ground-truth baseline values *w.r.t.* different patterns. In this case, the collaboration between input variables was complex and hard to analyze, so we did not consider such input variables with conflicting patterns for evaluation, either.

F.6 Discussion about the setting of ground-truth Shapley values.

This section discusses the ground truth of Shapley values in the extended Addition-Multiplication dataset (Zhang et al., 2021), which is used in Section 4.1 of the main paper. In order to verify the correctness of the Shapley values obtained by the optimal baseline values in this paper, we conducted experiments on the extended Addition-Multiplication dataset (Zhang et al., 2021) with ground-truth Shapley values.

The Addition-Multiplication dataset in (Zhang et al., 2021) contained functions that only consisted of addition and multiplication operations. For example, $f(\boldsymbol{x}) = x_1 x_2 + x_3 x_4$ where each input variable $x_i \in \{0, 1\}$ was a binary variable. Given $\boldsymbol{x} = [1, 1, 1, 1]$, the function contained two salient causal patterns, *i.e.* $\{x_1, x_2\}$ and $\{x_3, x_4\}$, and their benefits to the output were $I_{\{x_1,x_2\}} = I_{\{x_3,x_4\}} = 1$, respectively. According to (Harsanyi, 1982), the Shapley value is a uniform distribution of attributions. Therefore, the effect of an causal pattern was supposed to be uniformly assigned to variables in the pattern. Thus, the ground-truth Shapley values of variables were $\hat{\phi}_1 = \hat{\phi}_2 = 1/2$, and $\hat{\phi}_3 = \hat{\phi}_4 = 1/2$. However, if the input $\boldsymbol{x} = [1, 0, 1, 1]$, then the pattern $\{x_1, x_2\}$ was deactivated and $I_{\{x_1,x_2\}} = 0$. In this case, $\hat{\phi}_1 = \hat{\phi}_2 = 0$ while $\hat{\phi}_3 = \hat{\phi}_4 = 1/2$.

According to the analysis in Appendix F.5, ground-truth baseline values in the Addition-Multiplication dataset were all zero. Then our method is equivalent to the zero baseline values. Therefore, in order to avoid all ground-truth baseline values being zero, we added the subtraction operation. We also added a coefficient before each term in the function to boost the diversity of functions. For example, $f(\boldsymbol{x}) = 3.2x_1 x_2 + 1.5x_3(x_4 - 1)$. This function also contained two causal patterns, but the ground-truth baseline values of variables were different from the aforementioned function. Here, $b_0^* = b_0^* = b_3^* = 0$ and $b_4^* = 1$. Given the input $\boldsymbol{x} = [1, 1, 1, 0]$, all patterns were activated and $f(\boldsymbol{x}) = I_{\{x_1,x_2\}} + I_{\{x_3,x_4\}} = 3.2 + (-1.5) = 1.7$. Ground-truth Shapley values of input variables were $\hat{\phi}_1 = \hat{\phi}_2 = 3.2/2$ and $\hat{\phi}_3 = \hat{\phi}_4 = -1.5/2$. However, for the input $\boldsymbol{x} = [1, 1, 1, 1]$, the pattern $\{x_3, x_4\}$ was deactivated, thereby $\hat{\phi}_3 = \hat{\phi}_4 = 0$. Note that the above function can also be considered to contain three patterns ($f(\boldsymbol{x}) = 3.2x_1 x_2 + 1.5x_3 x_4 - 1.5x_3$). According to Occam's Razor, we follow the principle of the most simplified interaction to recognize causal patterns in the function, *i.e.* using the least number of causal patterns. Thus, we consider the above function $f(\boldsymbol{x}) = 3.2x_1 x_2 + 1.5x_3(x_4 - 1)$ containing two salient causal patterns.

Based on the extended Addition-Multiplication dataset, we randomly generated an input sample for each function in the dataset. Each variable x_i in input samples were independently sampled

Table 12: An example for Shapley values computed by using different baseline values. For the function $model(\boldsymbol{x}) = -4.23x_1x_2x_4x_5x_6(x_7 - 0.63)x_8$, if the input $\boldsymbol{x} = [1, 1, 1, 1, 1, 1, 1, 1]$, then other baseline values usually generated incorrect Shapley values.

Baseline values	The computed Shapley values $\{\phi_i\}$
Ground truth/Ours	$\{-0.2236, -0.2236, 0, -0.2236, -0.2236, -0.2236, -0.2236, -0.2236\}$
Zero baseline	$\{-0.1601, -0.1601, 0, -0.1601, -0.1601, -0.1601, -0.6043, -0.1601\}$
Mean baseline	$\{-0.1624, -0.1624, 0, -0.1624, -0.1624, -0.1624, -0.5996, -0.1624\}$
Setting in SHAP	$\{-0.0013, -0.0012, 0, -0.0011, -0.0012, -0.0012, -0.0044, -0.0014\}$

Table 13: An example for Shapley values computed by using different baseline values. For the function $model(\boldsymbol{x}) = -3.49(x_2 - 0.15)(x_5 - 0.78) - 0.88x_3 + 2.24(x_4 - 0.68) - 1.60(x_7 - 0.61) - 4.52x_8$, if the input $\boldsymbol{x} = [1, 1, 1, 1, 1, 1, 1, 0]$, then other baseline values usually generated incorrect Shapley values.

Baseline values	The computed Shapley values $\{\phi_i\}$
Ground truth/Ours	$\{0, -0.3263, -0.8800, 0.7168, -0.3263, 0, -0.6240, 0\}$
Zero baseline	$\{0, 0.9772, -0.8800, 2.2400, -1.2215, 0, -1.6000, 0\}$
Mean baseline	$\{0, 0.0523, -0.4400, 1.12000, -1.0470, 0, -0.8000, 2.2600\}$
Setting in SHAP	$\{0, -0.0005, -0.0021, 0.0051, -0.0059, 0, -0.0037, 0.0246\}$

following the Bernoulli distribution, *i.e.* $p(x_i = 1) = 0.7$. Therefore, for the mean baseline, baseline values of different input variables were all 0.7. For the baseline value based on the marginal distribution, which was used in SHAP (Lundberg and Lee, 2017), $p(x'_i) \sim$ *Bernoulli*(0.7). Then, we compared the accuracy of the computed Shapley values of input variables based on zero baseline values, mean baseline values, baseline values in SHAP, and the optimal baseline values defined in this paper, respectively. The result in Table 4 of the main paper shows that the optimal baseline values correctly generated the ground-truth attributions/Shapley values of input variables.

F.7 More examples of incorrect Shapley values

This section provides more examples of incorrect Shapley values computed by using other baseline values, as discussed in Section 4.1 of the main paper.

Table 12 and Table 13 show two examples of incorrect Shapley values computed by using other baseline values. For the example in Table 12, Shapley values of all input variables should be the same because the output is there production, but the baseline value used in SHAP generated different Shapley values for these variables. For the example in Table 13, Shapley value of the variable x_8 should be zero because $-4.52x_8 = 0$. However, mean baseline values and baseline values in SHAP generated non-zero and large Shapley values for this variable. These results demonstrate the unreliability of other baseline values.

F.8 Other potential settings of $v(x_S)$.

In the computation of Shapley values, people usually use different settings of $v(\boldsymbol{x}_S)$, although the settings of $v(\boldsymbol{x}_S)$ do not affect the applicability of our method. Our method of formulating baseline values is applicable to various settings of $v(\boldsymbol{x}_S)$. Lundberg and Lee (2017) directly set $v(\boldsymbol{x}_S) = p(y^{\text{truth}}|\boldsymbol{x}_S)$. Covert et al. (2020b) used the cross-entropy loss as $v(\boldsymbol{x}_S)$. In this paper, we use $v(\boldsymbol{x}_S) = \log \frac{p(y^{\text{truth}}|\boldsymbol{x}_S)}{1-p(y^{\text{truth}}|\boldsymbol{x}_S)}$ in L_{Shapley}. Besides, we use $|\Delta v_i(S)| = \|h(\boldsymbol{x}_{S\cup\{i\}}) - h(\boldsymbol{x}_S)\|_1$ in L_{marginal} on the MNIST dataset to boost the optimization efficiency, where $h(\boldsymbol{x}_S)$ denotes the intermediate-layer feature. It is because $h(\boldsymbol{x}_{S\cup\{i\}})$ makes the optimization of L_{marginal} receive gradients from all dimensions of the feature.

F.9 Stability of the learning of baseline values

In this section, we conducted experiments to verify the stability of the learning of baseline values. We repeated experiments in Section 4.1, which verified the correctness of the learned baseline values, for three times with different random seeds. Then, we computed the mean and standard deviation of the accuracy of the learned baseline values. Table 14 shows that mean accuracy was close to 100% and the standard deviation was relatively small. Therefore, the learning of baseline values was relatively stable.

Table 14: Stability of the learned baseline values: The mean and standard deviation of accuracy of the learned baseline values over three runs.

	initialize as 0	initialize as 0.5	initialize as 1
Accuracy of baseline values on 100 synthetic functions by using L_{shapley}	98.00±0.04 %	98.84±0.10 %	98.14±0.07 %

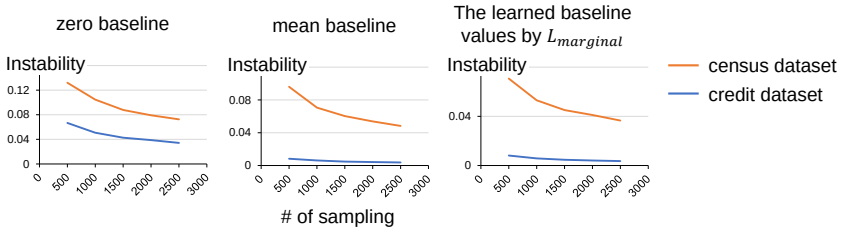


Figure 11: Instability of the Shapley values that were approximated with different sampling numbers. When the sampling number increased in the approximation of the Shapley value, the approximation of Shapley values gradually converged and became increasingly stable.

F.10 Stability of the estimated Shapley values

In this section, we provide more details and discussions about the instability of the estimated Shapley values. Given a specific setting of baseline values, we used the sampling-based approximation (Castro et al., 2009), which provided an unbiased estimation of Shapley values. Besides, for the varying baseline values, we used the code released by (Lundberg and Lee, 2017) and (Covert et al., 2020b). We conducted experiments to examine how to set the sampling number in (Castro et al., 2009) to ensure stable estimation of Shapely values.

Given a certain input sample, we fixed the sampling number and repeatedly computed the Shapley value multiple times. Then, we measured the instability of the estimated Shapley value ϕ_i for a certain input variable i, which indicated that whether we could obtain similar Shapley values considering the randomness in each sampling process. The instability was computed as $\frac{E_{u,v;u\neq v}|\phi_i^{(u)}-\phi_i^{(v)}|}{E_w|\phi_i^{(w)}|}$, where $\phi_i^{(u)}$ denote the estimated Shapley value in the u-th time. Then, we computed the average instability value over Shapley values of all variables in 20 input samples. Figure 11 shows that that when the sampling number was larger than 1000, the instability of the estimated Shapley values was usually lower than 0.1 for all baseline values. This meant that the computed Shapley values were stable for all baseline values. Therefore, we set the sampling number to 1000 in all other experiments in this paper.

F.11 Experimental results on the MNIST and the CIFAR-10 datasets

Experimental results on the MNIST datasets. Experimental settings on the MNIST dataset have been introduced in Section 4.2 of the paper. Figure 12 shows the learned baseline values on the MNIST dataset by using L_{Shapley} and L_{marginal} in Section 3 of the paper, respectively. Figure 13 shows the computed Shapley values using different baseline values on the MNIST dataset. Compared to zero/mean/blurring baseline values, the learned baseline values by our method removed noisy variables on the background, which were far from digits in images. Compared to SHAP, our method yielded more informative attributions. Shapley values computed using baseline values in SAGE were dotted. In comparison, our method generated smoother attributions.

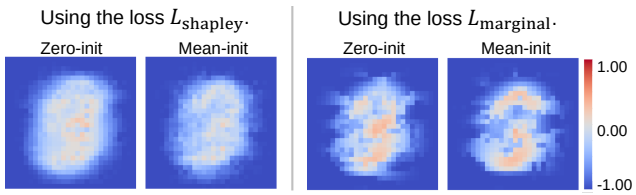


Figure 12: The learned baseline values on the MNIST dataset (better viewed in color).

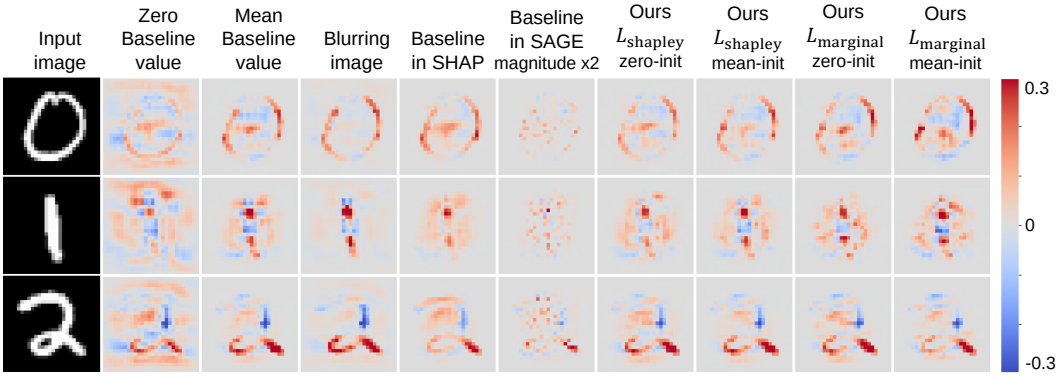


Figure 13: Shapley values computed with different baseline values on the MNIST dataset.

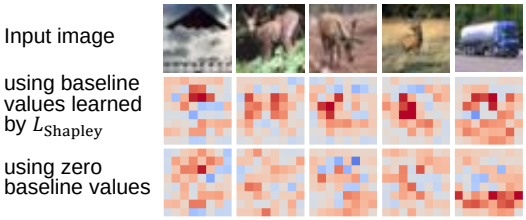


Figure 14: Shapley values computed with different baseline values. Images were selected from the CIFAR-10 dataset.

Experimental results on the CIFAR-10 datasets. First, let us clarify the experiment settings on the CIFAR-10 dataset. We split each image into 8×8 grids. Each grid contained 4×4 pixels, which shared a specific RGB color as their baseline value. Thus, we needed to learn a total of $8 \times 8 \times 3$ color values as baseline values. We learned baseline values by using the loss L_{Shapley}. Then, we computed Shapley values for grids by using zero baseline values and computed Shapley values by using the learned baseline values for comparison. Figure 14 shows that Shapley values computed with the learned baseline values mainly focused on objects in images. In comparison, Shapley values based on zero baselines values mainly focused on the background.

F.12 EXPERIMENTAL RESULTS ON THE UCI SOUTH GERMAN CREDIT DATASET.

This section provides experimental results on the UCI South German Credit dataset (Dua and Graff, 2017). Based on the UCI datasets, we learned MLPs following settings in (Guidotti et al., 2018). Figure 15 shows the learned baseline values by our method, and Figure 16 compares Shapley values computed using different baseline values. Just like results on the UCI Census Income dataset, attributions (Shapley values) generated by our learned baseline values are similar to results of the varying baseline values in SHAP and SAGE. However, the zero/mean baseline values usually generated conflicting results with all other methods.

Besides, we also noticed that baseline values learned with different initialization settings (*zero-init* and *mean-init*) all converged to similar baseline values, except for very few dimensions having multiple local-minimum solutions, which proved the stability of our method. More specifically, an input variable might have different optimal baseline values in real applications from different perspectives. For example, in the function of $y = (a-0.5)(b-3)(c+2)+(a+0.5)(d-3)(e+2)+f$, the variable a had two equivalent baseline value ± 0.5.

F.13 DETAILS ABOUT THE QUANTITATIVE EVALUATION OF ATTRIBUTIONS FOR IMAGE CLASSIFICATION

This section introduces experimental details about the quantitative evaluation of attributions on the MNIST dataset. We constructed a decision tree (Song et al., 2013) for each category in the MNIST dataset. Specifically, for each category (digit), we first computed the average image over all training

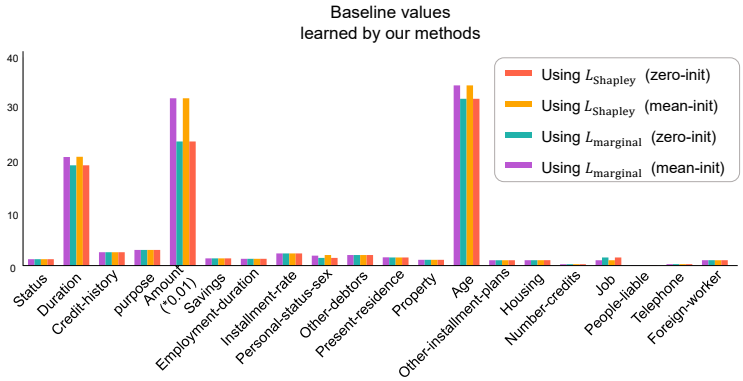


Figure 15: The learned baseline values on the UCI South German Credit dataset.

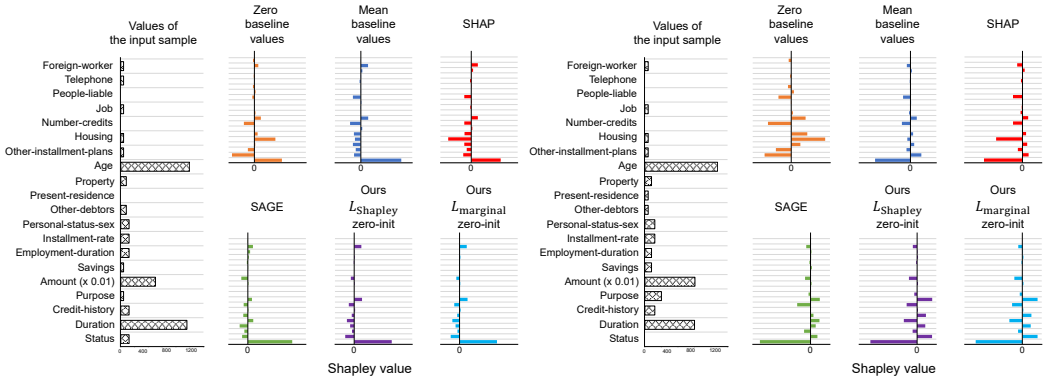


Figure 16: Shapley values computed with different baseline values on the UCI South German Credit dataset.

samples in this category. Let $\bar{\boldsymbol{x}}^{(c)} \in \mathbb{R}^n$ denote the average image of the c-th category. Then, we built a decision tree by considering each pixel as an internal node. The splitting rule for the decision tree was designed as follows. Given an input $\boldsymbol{x}$ in the category c, the splitting criterion at the pixel (node) x_i was designed as $\Big((\bar{x}_i^{(c)} > 0.5)\&(x_i > 0.5)\Big)$[2]. If $(\bar{x}_i^{(c)} > 0.5)\&(x_i > 0.5) = \text{True}$, then the pixel value x_i was added to the output; otherwise, x_i was ignored. In this way, the output of the decision tree was $f(\boldsymbol{x}) = \sum_{i \in V} x_i$, where $V = \{i \in N | (\bar{x}_i^{(c)} > 0.5)\&(x_i > 0.5) = \text{True}\}$ denoted the set of all pixels that satisfied the above equation. For inference, the probability of $\boldsymbol{x}$ belonging to the category c was $p(c|\boldsymbol{x}) = \text{sigmoid}(\gamma(f(\boldsymbol{x}) - \beta))$, where $\gamma = 40$ was a constant and $\beta \propto \sum_{i \in N} \mathbb{1}_{\bar{x}_i^{(c)} > 0.5}$. In this case, we defined $v(\boldsymbol{x}_N) = \log \frac{p(c|\boldsymbol{x})}{1 - p(c|\boldsymbol{x})}$. Thus, the co-appearing of pixels in V formed a causal pattern to contribute for $v(\boldsymbol{x}_N)$. In other words, because $\forall i \in N, x_i \geq 0$, the absence of any pixel in V might deactivate this pattern by leading to a small probability $p(c|\boldsymbol{x}) < 0.5$ and a small v. This pattern can also be understood as an AND node in the And-Or decision tree (Song et al., 2013).

In the above decision tree, the ground-truth Shapley values of input variables (pixels) were easy to determine. The above decision tree ensured that the absence of any variable in V would deactivate the causal pattern. Therefore, according to Theorem 2 in the paper, the output probability should be fairly assigned to pixels in V, *i.e.*, they shared the same Shapley values $\hat{\phi}_i = \frac{v(\boldsymbol{x}_N)}{|V|}$. For other pixels that were not contained in the output, their ground-truth Shapley values were zero.

We estimated Shapley values of input variables in the above decision tree by using zero baseline values, mean baseline values, baseline values in SHAP, and the learned baseline values by our method, respectively. Let ϕ_i denote the estimated Shapley value of the variable i. If $|\phi_i - \hat{\phi}_i| \leq 0.01$, we considered the estimated Shapley value ϕ_i correct; otherwise, incorrect. In this way, we computed

[2]For Table 6 in the paper, the splitting criterion was designed as $(\bar{x}_i^{(c)} > 0.5)$.

Table 15: Accuracy of Shapley values computed on the pre-defined decision tree, which was based on the MNIST dataset.

Baseline value	zero	mean	baseline value in SHAP	ours
Accuracy	90.86	90.86	90.82	**90.87**

the accuracy of the estimated Shapley values, and Table 15 shows that out method achieved the highest accuracy.

F.14 DISCUSSION ABOUT THE BASELINE VALUES LEARNED ON ADVERSARIAL EXAMPLES.

In Section 4.3 of the main paper, we show that the learned baseline values can recover original samples form adversarial examples. This section provides more discussions about this experiment.

Let $\boldsymbol{x}$ denote the normal sample, and let $\boldsymbol{x}^{\text{adv}} = \boldsymbol{x} + \delta$ denote the adversarial example generated by adversarial attacks (Madry et al., 2018). A previous study (Ren et al., 2021b) defined the bivariate interaction with different contextual complexities, and found that adversarial attacks mainly created out-of-distribution bivariate interactions with large contexts. In the scenario of this study, we can consider such sensitive interactions with large contexts related to high-order multi-variate causal patterns. It is because the bivariate causal between (i, j) with large contexts (*i.e.* under many contextual variables S) actually considers the collaboration between i, j and contextual variables in S. Thus, it also partially reflects $I_{S\cup\{i,j\}}$, which is defined in this paper.

Therefore, from the perspective of multi-variate causal patterns, the adversarial utility can be considered as introducing out-of-distribution high-order causal patterns in the model. To this end, setting input variables to baseline values is supposed to remove related causal patterns to represent absence states. Particularly, if we set all input variables to their baseline values, many causal patterns will be eliminated. Thus, if we initialize baseline values as the adversarial example, and optimize baseline values using our method, the learned baseline values are supposed to remove OOD high-order causal patterns in the adversarial example, and recover the original sample.

F.15 COMPUTATIONAL COMPLEXITY OF THE PROPOSED LOSS FUNCTIONS

In this section, we will introduce both the theoretical complexity of the loss function and the real complexity of approximating the loss in real applications.

In terms of the theoretical complexity, the complexity of the loss functions $L_{Shapley}$ and $L_{marginal}$ in Eq. (8) is $Dn \cdot \sum_{m=0}^{\lambda} \binom{n}{m}$, where D denotes the number of samples $\boldsymbol{x}$ in the dataset. n is the number of input variables in the input. $\sum_{m=0}^{\lambda} \binom{n}{m}$ denotes the computational complexity of all terms $\phi_i^{(m)}(x|b)$ of different orders $m \leq \lambda$.

In terms of real implementation, as mentioned in Section 4.2, we used the sampling-based approximation (Castro et al., 2009) to compute the loss function. We randomly sampled K input variables in each epoch, and we randomly sampled T contexts of each order m *w.r.t.* each variable i to approximate $\phi_i^{(m)}(x|b)$. Thus, the real computational cost of $L_{Shapley}$ and $L_{marginal}$ was reduced to $DK \cdot \sum_{m=0}^{\lambda} T$, where we set $T = 100$ in implementation.

We conducted an experiment to show the time cost of learning the optimal baseline values. We used the MLP, LeNet, and ResNet-20 trained on the *income* dataset, the *credit* dataset, the MNIST dataset, and the CIFAR-10 dataset. For the MNIST and the CIFAR-10 datasets, we randomly sampled $K = 100$ pixels as the variable set N to compute the Loss $L_{Shapley}$ and optimize their baseline values.¡/font¿ The following table reports the time cost of optimizing $L_{Shapley}$ for an epoch *w.r.t.* DNNs trained on different datasets. The time cost was measured by using PyTorch 1.10 on Ubuntu 20.04, with the Intel(R) Core(TM) i9-10900X CPU @ 3.70GHz and one NVIDIA Geforce RTX 3090 GPU.

Table 16: Time cost of learning baseline values for an epoch.

Model	Dataset	Number of input variables n	Time per epoch	Overall computational time until convergence
5-layer MLP	UCI Census Income dataset	12	3.81s	1h3min (1000 epochs)
5-layer MLP	UCI South German Credit dataset	20	5.58s	1h33min (1000 epochs)
LeNet	MNIST	28×28	45.69s	2h32min (200 epochs)
ResNet-20	CIFAR-10	32×32	64.44s	3h44min (200 epochs)

G MULTI-ORDER SHAPLEY VALUES AND MARGINAL BENEFITS

In Section 3 of the main paper, we propose the Shapley values of different orders $\phi_i^{(m)}$, and marginal benefits of different orders $\Delta v_i(S)$. Furthermore, we find that high-order causal effects are only contained by high-order Shapley values and marginal benefits. Actually, the loss function on marginal benefits L_{marginal} is more fine-grained than the loss function on the multi-order Shapley value L_{Shapley}. This section provides proofs for the above claims.

First, the Shapley value ϕ_i can be decomposed into the sum of Shapley values of different orders $\phi_i^{(m)}$ and marginal benefits of different orders $\Delta v_i(S)$, as follows.

$$\phi_i = \frac{1}{n}\sum_{m=0}^{n-1}\phi_i^{(m)} = \frac{1}{n}\sum_{m=0}^{n-1}\mathbb{E}_{S\subseteq N\setminus\{i\},|S|=m}\Delta v_i(S) \tag{12}$$

where the Shapley value of m-order $\phi_i^{(m)} \overset{\text{def}}{=} \mathbb{E}_{S\subseteq N\setminus\{i\}|S|=m}\left[v(\boldsymbol{x}_{S\cup\{i\}}) - v(\boldsymbol{x}_S)\right]$, and the marginal benefit $\Delta v_i(S) \overset{\text{def}}{=} v(\boldsymbol{x}_{S\cup\{i\}}) - v(\boldsymbol{x}_S)$.

• *Proof*:

$$\begin{aligned}
\phi_i &= \sum_{S\subseteq N}\frac{|S|!(n-1-|S|!)}{n!}\left[v(\boldsymbol{x}_{S\cup\{i\}}) - v(\boldsymbol{x}_S)\right] \\
&= \sum_{m=0}^{n-1}\sum_{S\subseteq N,|S|=m}\frac{|S|!(n-1-|S|!)}{n!}\left[v(\boldsymbol{x}_{S\cup\{i\}}) - v(\boldsymbol{x}_S)\right] \\
&= \frac{1}{n}\sum_{m=0}^{n-1}\sum_{S\subseteq N,|S|=m}\frac{|S|!(n-1-|S|!)}{(n-1)!}\left[v(\boldsymbol{x}_{S\cup\{i\}}) - v(\boldsymbol{x}_S)\right] \\
&= \frac{1}{n}\sum_{m=0}^{n-1}\mathbb{E}_{S\subseteq N,|S|=m}\left[v(\boldsymbol{x}_{S\cup\{i\}}) - v(\boldsymbol{x}_S)\right] \\
&= \frac{1}{n}\sum_{m=0}^{n-1}\phi_i^{(m)} \\
&= \frac{1}{n}\sum_{m=0}^{n-1}\mathbb{E}_{S\subseteq N\setminus\{i\},|S|=m}\Delta v_i(S)
\end{aligned}$$

Connection between multi-variate interactions and multi-order marginal benefits. The m-order marginal benefit can be decomposed as the sum of multi-variate interaction benefits, as follows. Therefore, high-order causal effects I_S are only contained in high-order marginal benefits.

$$\Delta v_i(S) = \sum_{L\subseteq S} I_{L\cup\{i\}} \tag{13}$$

• *Proof*:

$$\begin{aligned}
\text{right} &= \sum_{L\subseteq S} I_{L\cup\{i\}} \\
&= \sum_{L\subseteq S}\left[\sum_{K\subseteq L}(-1)^{|L|+1-|K|}v(\boldsymbol{x}_K) + \sum_{K\subseteq L}(-1)^{|L|-|K|}v(\boldsymbol{x}_{K\cup\{i\}})\right] \\
&= \sum_{L\subseteq S}\sum_{K\subseteq L}(-1)^{|L|-|K|}\left[v(\boldsymbol{x}_{K\cup\{i\}}) - v(\boldsymbol{x}_K)\right]
\end{aligned}$$

$$
\begin{aligned}
&= \sum_{L \subseteq S} \sum_{K \subseteq L} (-1)^{|L|-|K|} \Delta v_i(K) \\
&= \sum_{K \subseteq S} \sum_{P \subseteq S \setminus K} (-1)^{|P|} \Delta v_i(K) \qquad \text{\% Let } P = L \setminus K \\
&= \sum_{K \subseteq S} \left(\sum_{p=0}^{|S|-|K|} \binom{|S|-|K|}{p} (-1)^p \right) \Delta v_i(K) \qquad \text{\% Let } p = |P| \\
&= \sum_{K \subseteq S} \left[(1 + (-1)^{|S|-|K|}) \right] \Delta v_i(K) \qquad \text{\% Let } p = |P| \\
&= \sum_{K \subsetneq S} 0 \cdot \Delta v_i(K) + \sum_{K = S} \left(\sum_{p=0}^{|S|-|K|} \binom{|S|-|K|}{p} (-1)^p \right) \Delta v_i(K) \\
&= \Delta v_i(S) = \text{left}
\end{aligned}
$$

Connection between multi-order interactions and multi-order Shapley values. The m-order Shapley value can also be decomposed as the sum of interaction benefits, as follows. Therefore, high-order causal effects I_S are only contained in high-order Shapley values.

$$
\phi_i^{(m)} = \mathbb{E}_{\substack{S \subseteq N \setminus \{i\} \\ |S|=m}} \left[\sum_{L \subseteq S} I_{L \cup \{i\}} \right] \tag{14}
$$

• *Proof*:

$$
\begin{aligned}
\phi_i^{(m)} &= \mathbb{E}_{S \subseteq N, |S|=m} \Delta v_i(S) \\
&= \mathbb{E}_{S \subseteq N, |S|=m} \left[\sum_{L \subseteq S} I_{L \cup \{i\}} \right] \\
&= \mathbb{E}_{S \subseteq N, |S|=m} \left[\sum_{L \subseteq S} I_{L \cup \{i\}} \right]
\end{aligned}
$$