Proxy-Based Approximation of Shapley and Banzhaf Interactions

Santo M. A. R. Thies [1,2,3,*] **Hubert Baniecki** [4,5] **R. Teal Witter** [6]
Eyke Hüllermeier [1,2,3] **Maximilian Muschalik** [1,2,†] **Fabian Fumagalli** [1,2,7,†]

[1]LMU Munich [2]MCML [3]DFKI [4]Centre for Credible AI, Warsaw University of Technology
[5]University of Warsaw [6]Claremont McKenna College [7]Bielefeld University

Abstract

Shapley and Banzhaf interactions capture the complex dynamics inherent in modern machine learning applications. However, current estimators for these higher-order interactions trade off between speed and accuracy. To overcome this limitation, we introduce ProxySHAP. ProxySHAP reconciles the high sample efficiency of tree-based proxy models with a situational residual correction recovering consistency. On a theoretical level, we derive a polynomial-time generalization of interventional TreeSHAP that computes exact interaction indices for tree ensembles, thereby bypassing exponential tree-depth dependencies in prior methods. Furthermore, we formally analyze the residual adjustment strategy, characterizing the specific conditions under which Maximum Sample Reuse (MSR) corrects proxy bias without its variance scaling exponentially with interaction size. Extensive benchmarking demonstrates that ProxySHAP sets a new state-of-the-art standard for approximation quality, including in large-scale applications with thousands of features. By achieving the lowest error in both small- and large-budget regimes, ProxySHAP significantly outperforms the prior best estimators ProxySPEX and KernelSHAP-IQ, while also delivering superior performance on downstream explainability tasks.

1 Introduction

With the growing integration of artificial intelligence into critical decision-making processes across healthcare and finance, the demand for transparency and trustworthiness has never been higher. To interpret the often opaque dynamics of these models, the field has coalesced around cooperative game theory as the *de facto* standard [36, 40, 41, 49]. Specifically, Shapley values [53] and Banzhaf values [3], fundamental instances of *cardinal-probabilistic values*, provide a rigorous axiomatic framework for attribution. These concepts are ubiquitous in modern machine learning, serving as the cornerstone for tasks such as feature attribution [36, 55] and data valuation [25, 49].

Formally, we consider a set $N = \{1, \ldots, n\}$ consisting of n entities, such as input features or training data points. The theoretical foundation of these explanations rests on a value function, $\nu : 2^N \to \mathbb{R}$, assigning a scalar score to any subset (or coalition) $S \subseteq N$. The interpretation of ν depends on the application: in feature attribution, $\nu(S)$ typically represents the model's prediction when features in $N \setminus S$ are masked or marginalized out; in data valuation, $\nu(S)$ corresponds to the utility, such as test accuracy or negative loss, of a model trained exclusively on the subset of data points S.

To summarize the complex dynamics of ν, cardinal-probabilistic values quantify the marginal effect of each entity i on the model's output. Formally, the attribution to i is a weighted sum of its marginal

*Corresponding author: s.thies@lmu.de.
†Equal supervision.

Part of shapiq: https://github.com/mmschlk/shapiq.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

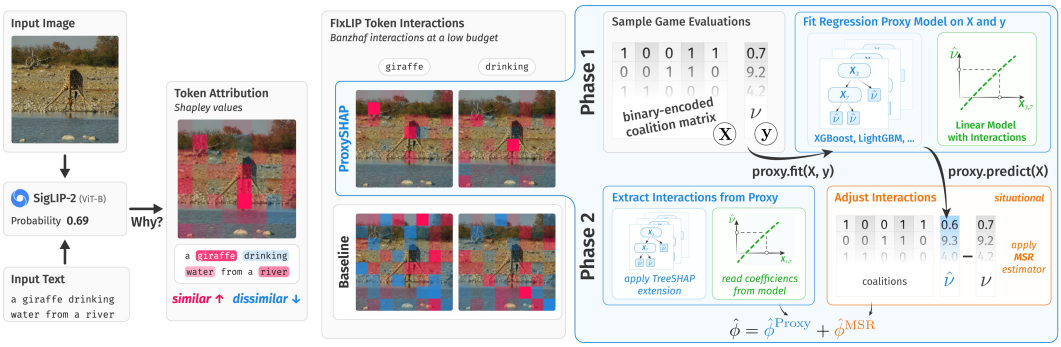


Figure 1: Left: A ProxySHAP explanation of the SigLIP-2 model using only 2048 model calls. **Right:** In *Phase 1*, we fit a regression proxy model using sampled binary coalitions and game values. In *Phase 2*, we extract proxy interactions and, when appropriate, adjust them using residual estimates.

contributions across all possible subsets:

$$\phi_i^p(\nu) := \sum_{T \subseteq N \setminus \{i\}} p_t(n) \Delta_i \nu(T), \tag{1}$$

where $\Delta_i \nu(T) := \nu(T \cup \{i\}) - \nu(T)$ denotes the marginal contribution of ν at coalition T with respect to i, and $t := |T|$ is the cardinality of T. Setting the weights to $p_t(n) = \frac{1}{n\binom{n-1}{t}}$ yields the Shapley value, while $p_t(n) = \frac{1}{2^{n-1}}$ yields the Banzhaf value.

While probabilistic values are useful for understanding individual contributions, they remain limited to singleton effects and often fail to capture the interactions between elements. To capture these complex dynamics, recent research has shifted toward richer explanations grounded in higher-order terms. These concepts have seen rapid adoption for quantifying feature interactions [16, 42, 58], cross-modal interactions in vision-language models [2, 26], data valuation [5], language modelling [52, 54], and even hyperparameter optimization [62]. For instance, in vision-language similarity models such as SigLIP-2 [59], interactions between image and text patches are not captured by probabilistic values [2], as illustrated in Figure 1.

Cardinal-probabilistic interaction indices naturally generalize probabilistic values to arbitrary subsets [15]. Rather than isolating the marginal contribution of a singleton, they quantify the contribution of a subset $S \subseteq N$ as it is added to a coalition T:

$$\phi_S^p(\nu) := \sum_{T \subseteq N \setminus S} p_t^s(n) \Delta_S \nu(T), \tag{2}$$

where $\Delta_S \nu(T)$ denotes the discrete derivative of ν at T with respect to the *subset* S. Akin to the singleton case, specific weight choices allow us to recover the Shapley interaction index [19] and the Banzhaf interaction index [19]. For $s = 1$, the interaction weights reduce to $p_t(n)$.

1.1 Estimating Cardinal-Probabilistic Interactions

As computing probabilistic values and interactions requires evaluating $\nu(T)$ across all 2^n subsets, exact computation is infeasible for large n. Consequently, researchers rely on algorithms that produce estimates using a fixed evaluation budget. For probabilistic values, there are numerous estimators such as Monte Carlo sampling [56], maximum sample reuse (MSR) [60], permutation sampling [7], and regression-based approaches like KernelSHAP [10, 36, 44].

Recently, work on estimating probabilistic *interactions* has expanded to approximate the higher-order interactions inherent in modern machine learning, but current estimators are either slow or not accurate. SHAP-IQ [16] generalizes MSR to interactions, being efficiently computable, but inaccurate, as its variance scales quadratically with the magnitude of ν, which we find compounds further for higher-order interactions. Conversely, regression approaches like KernelSHAP-IQ [17] are accurate but require impractical sample budgets to adequately fit the large number of higher-order terms. Additionally, KernelSHAP-IQ remains computationally expensive, incurring quadratic time complexity in n.

To circumvent these bottlenecks, recent work leverages surrogate models $\hat{\nu}$ to approximate ν [5, 63]. ProxySPEX [5] extracts Fourier coefficients from a tree surrogate, which scales exponentially with depth, necessitating aggressive, accuracy-compromising truncation. Additionally, even with the exact Fourier coefficients, ProxySPEX would return the interaction of $\hat{\nu}$ rather than the underlying function ν. RegressionMSR [63] elegantly resolves this by directly extracting the values from the tree and employing MSR to estimate the residual values of $\nu - \hat{\nu}$, showcasing superior performance for probabilistic *values*, yet remaining restricted to singleton effects. Extending this residual-correction framework to higher-order interactions requires efficient extraction of interactions from tree proxies. While an algorithm for extracting Shapley interaction indices exists [65], it is ill-suited for general cardinal-probabilistic interactions (e.g., those used in vision tasks, as shown in Figure 1).

1.2 Our Contributions

Current interaction estimators are either fast or accurate; we propose *ProxySHAP*, which is both. First, we extend the algorithm of Zern et al. [65] to extract *any* interaction index that exhibits a Möbius representation ((4)), subsuming cardinal-probabilistic interaction indices, from a tree in polynomial time, orders of magnitude faster than ProxySPEX's Fourier-based extraction (see Section 3.1 and Figure 2). Second, we correct the resulting bias with the residual strategy of Witter et al. [63] and generalized MSR [16]. Unlike for probabilistic values, this correction does not always help, as its variance grows with the magnitude of the value function, the number of entities, and exponentially with the interaction order, so we characterize when it does (see Section 3.2 and Figure 3). Across *54 datasets* spanning tabular, vision, language, graph, and vision–language domains, ProxySHAP attains the lowest error at both small budgets, previously held by ProxySPEX, and large budgets, previously held by KernelSHAP-IQ. It remains applicable to thousands of entities and improves the downstream CLIP explanations (see Sections 4.2 and 4.3).

Our main contributions can be summarized as follows:

(1) ProxySHAP. We introduce ProxySHAP, an estimation framework for cardinal-probabilistic interaction indices (Figure 1) that combines the sample efficiency of tree-based proxies with a residual correction applied only where it is most effective.

(2) Theoretical Foundations. We generalize interventional TreeSHAP to compute *exact* cardinal-probabilistic interaction indices for tree ensembles in polynomial time, avoiding the exponential tree-depth dependence of Fourier-based extraction. We further bound the variance of the MSR correction for interactions, separating the regimes in which it removes proxy bias from those in which it dominates the error.

(3) State-of-the-Art Performance. Across 54 datasets, ProxySHAP attains lower error than ProxySPEX and KernelSHAP-IQ, often by orders of magnitude in relative MSE, and Pareto-dominates both on CLIP explanations.

2 Background

Cardinal-Probabilistic Interaction Indices. Fujimoto et al. [15] extended semivalues to interactions between subsets $S \subseteq N$. A *cardinal-probabilistic interaction index* is defined as:

$$\phi_S^p(\nu) := \sum_{T \subseteq N \setminus S} p_t^s(n) \Delta_S \nu(T), \quad \text{where} \quad \Delta_S \nu(T) := \sum_{L \subseteq S} (-1)^{s-\ell} \nu(T \cup L). \tag{3}$$

Here, $\Delta_S \nu(T)$ quantifies the interaction of S in the presence of T. These indices satisfy the axioms of *linearity (additivity)*, *symmetry*, *dummy partnership*, *monotonicity*, and k*-monotonicity* [15].

Möbius Representation. Fujimoto et al. [15] have further shown the equivalent representation in terms of the *Möbius transform* $m_S(\nu) := \Delta_S \nu(\emptyset)$, as

$$\phi_S^p(\nu) = \sum_{T \supseteq S} q_t^s(n) m_T(\nu), \tag{4}$$

where the corresponding weights $q_t^s(n)$ are computable by $p_t^s(n)$ for $s \in \{1, \ldots, n\}$ and all $t \in \{s, \ldots, n\}$. The structure of cardinal-probabilistic interaction indices simplifies in their Möbius representation (see Table 1), which itself is a cardinal-probabilistic interaction index.

Maximum Sample Reuse (MSR). Since evaluating ν on all 2^n coalitions is intractable, Wang and Jia [60] introduced maximum sample reuse (MSR), which was later extended to interactions by Fumagalli et al. [16]. Given a sample collection $\mathcal{T}$ drawn according to $\mathbb{P}_{\text{sampling}}$, MSR estimates:

$$\hat{\phi}_S^{\text{MSR}}(\nu;\mathcal{T}) := \frac{1}{|\mathcal{T}|}\sum_{T\in\mathcal{T}} \nu(T)\frac{(-1)^{s-|S\cap T|}p^s_{t-|S\cap T|}(n)}{\mathbb{P}_{\text{sampling}}(T)}. \tag{5}$$

MSR generalizes Unbiased KernelSHAP [10] and has been refined via stratification [31]. However, the variance of MSR scales with $\nu(T)^2$ [63]. To address this, Witter et al. [63] proposed *RegressionMSR* using a proxy model $\hat{\nu}$. We refine this approach for cardinal-probabilistic interaction indices.

3 Proxy-based Approximation of Shapley and Banzhaf Interactions

We now present *ProxySHAP*, a general framework for approximating cardinal-probabilistic interaction indices via a decomposition into a proxy term and a residual correction term. Given a budget of m coalitions, we sample a collection $\mathcal{T} = \{T_1, \dots, T_m\} \subseteq 2^N$ and query the value function $\nu(T)$ for all $T \in \mathcal{T}$. Using these samples, we fit a proxy model $\hat{\nu}_\mathcal{T} : 2^N \to \mathbb{R}$ to approximate the underlying game ν. The key observation underlying ProxySHAP is that, by linearity of cardinal-probabilistic interaction indices [15], the target interaction admits the decomposition

$$\phi^p_S(\nu) = \underbrace{\phi^p_S(\hat{\nu}_\mathcal{T})}_{\text{Exact Proxy}} + \underbrace{\phi^p_S(\nu - \hat{\nu}_\mathcal{T})}_{\text{Residual}}. \tag{6}$$

This separates the overall approximation problem into two components: *(i)* a modeling problem, namely how well the proxy $\hat{\nu}_\mathcal{T}$ approximates ν, and *(ii)* a correction problem, namely, how accurately the residual interactions can be estimated from the available coalition evaluations.

This perspective makes ProxySHAP a *framework* rather than a single fixed estimator. At a conceptual level, the only requirement for the proxy class is that its cardinal-probabilistic interactions can be efficiently extracted. Learning the proxy itself is a standard supervised regression problem, where the sampled coalitions are represented as binary inputs and the corresponding game values $\nu(T)$ serve as targets. Hence, in principle, any machine learning model can serve as a proxy, provided its interactions remain tractable. Moreover, this viewpoint naturally enables hyperparameter optimization to improve proxy quality, as shown in Figure 5.

In this work, we study two proxy classes. First, we consider a linear proxy with interaction features, which provides a simple and often competitive baseline. Second, we consider tree-based proxies, which form our main instantiation of ProxySHAP. Their appeal is that they are substantially more expressive than linear surrogates, while—as we show in the next subsection—still admitting exact polynomial-time cardinal-probabilistic interaction extraction. This exact extraction is the key to obtaining a substantial runtime improvement over Fourier-based proxy methods.

Linear Proxy. We first consider a linear model with interaction features. Let $\mathcal{S} \subseteq 2^N$ be all interactions of interest, e.g. interactions up to order $k = 1, \dots, n$. We define the linear proxy as

$$\hat{\nu}_{\text{linear}}(T) := \sum_{S\in\mathcal{S}} \beta_S \cdot \mathbb{1}[S \subseteq T] \qquad \text{with } \beta \in \mathbb{R}^{|\mathcal{S}|}. \tag{7}$$

The coefficients $\beta \in \mathbb{R}^{|\mathcal{S}|}$ are determined using standard linear regression. A key advantage of the linear proxy in our setting is that its interactions are directly accessible through the Möbius representation. Each coefficient β_S corresponds to a Möbius coefficient $m_S(\hat{\nu}_{\text{linear}})$, and therefore the proxy interactions can be computed directly via Eq. (4).

Proposition 3.1. *The Möbius transform of the linear proxy is given by* $m_S(\hat{\nu}_{linear}) = \beta_S \mathbb{1}[S \in \mathcal{S}]$. *Hence,*

$$\phi^p_S(\hat{\nu}_{linear}) = \sum_{T\in\mathcal{S}:T\supseteq S} q^s_t(n)\beta_T.$$

Thus, linear proxies provide a simple and effective instantiation of ProxySHAP whenever the interaction basis remains sufficiently small to be fitted reliably. However, they become restrictive

when the value function exhibits a strong nonlinearity, and the number of their parameters grows combinatorially with the interaction order. This makes them increasingly impractical for large n or high-order interactions. Tree-based proxies offer a compelling alternative: they capture complex non-linear relationships while remaining applicable to large feature sets and high-order interactions. Motivated by their strong empirical performance for cardinal-probabilistic values [63], we make tree-based proxies applicable to interaction estimation by deriving an exact polynomial-time extraction procedure for interactions which exhibit a Möbius representation as in Eq. (4).

3.1 Exact Proxy Interactions for Tree-Based Models

Our main proxy classes are tree-based models such as XGBoost [9] and LightGBM [29]. We define such tree-based proxies as the piecewise constant functions induced by decision tree leaf predictions:

$$\hat{\nu}_{\text{tree}}(T) := \sum_{j \in \mathcal{L}} c_j \cdot \mathbb{1}[R_j \subseteq T \subseteq N \setminus L_j]. \tag{8}$$

Here, $\mathcal{L}$ is the set of leaves, $c_j \in \mathbb{R}$ is the prediction of leaf j, and L_j and R_j are the features that split left and right along the path to j. That is, a coalition T reaches leaf j if and only if it contains every feature in R_j and none in L_j. Each leaf term is thus an indicator game [65], whose Möbius transform has a closed form (see Lemma A.2). Combining it with Eq. (4) and summing over the leaves yields the interaction index of the tree proxy.

Proposition 3.2. *For $S \subseteq N$, we have*

$$\phi_S^p(\hat{\nu}_{\text{tree}}) = \sum_{j \in \mathcal{L}: S \subseteq L_j \cup R_j} c_j \cdot \lambda\left(|L_j|, |R_j|, |S \cap L_j|, |S|\right),$$

where $\lambda(\ell, r, u, s) := \sum_{i=0}^{\ell-u} (-1)^{i+u} \binom{\ell-u}{i} q_{i+u+r}^s(n)$.

Proposition 3.2 shows that the interactions of the tree proxy can be computed exactly by aggregating leaf-wise contributions, requiring only a single pass over the tree leaves for a fixed interaction S together with evaluation of the closed-form weight λ. This yields Algorithm 2 with a runtime of $\mathcal{O}(n_{\text{nodes}})$ per interaction and $\mathcal{O}(n_{\text{nodes}} \cdot |\mathcal{S}|)$ for a collection $\mathcal{S}$ of interactions (see Appendix B for details). Crucially, by the linearity of cardinal-probabilistic interaction indices [15], the result extends to ensembles of trees, with runtime scaling linearly in the number of trees. This exact computation is a key computational advantage of tree-based ProxySHAP.

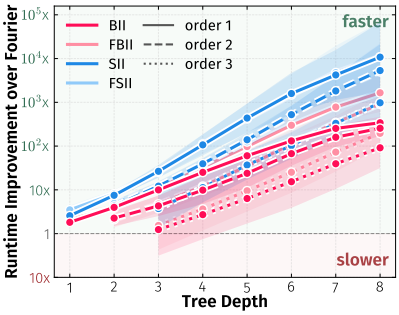


Figure 2: Runtime improvement of extracting interactions using our Algorithm 2 over Fourier-based extraction. Per-dataset speedups and the effect of tree depth on approximation quality are shown in Figures 12a and 13.

In contrast to ProxySPEX, which obtains interactions via Fourier coefficients and exhibits worst-case complexity exponential in the tree depth, $\mathcal{O}(4^d)$ [5, 18], our method remains efficient for deep trees. As shown in Figure 2, interventional tree-based extraction is orders of magnitude faster than Fourier-based extraction (see Appendix D.5 for details), making tree models particularly attractive proxies for ProxySHAP. Further comparison can be found in Appendix D.6.

Additionally, our tree-based ProxySHAP supports computing up to 10 indices, whereas previous methods support at most 4 (see Table 2), yielding practical gains for explaining text-image similarity models such as CLIP (see Figure 6). Furthermore, Proposition 3.2 filters the number of non-zero interactions, as $\phi_S^p(\hat{\nu}_{tree}) = 0$ whenever the features of S do not co-occur on a common root-leaf path. As such, for a tree of depth d and interactions of size s, at most $\mathcal{O}(n_{leaves}\binom{d}{s})$ can be non-zero, being often way smaller than the total number of interactions $\binom{n}{s}$.

3.2 Residual Adjustment and its Practical Limits

While the tree proxy alone already yields a powerful estimator, it is not consistent in general, since the fitted proxy need not perfectly match the underlying value function ν (see, e.g., Figure 4). To obtain consistency, meaning that we recover the exact index once all 2^n coalitions are observed, we correct for proxy bias by estimating the interactions of the residual game $\nu - \hat{\nu}_{\mathcal{T}}$. As previously shown by

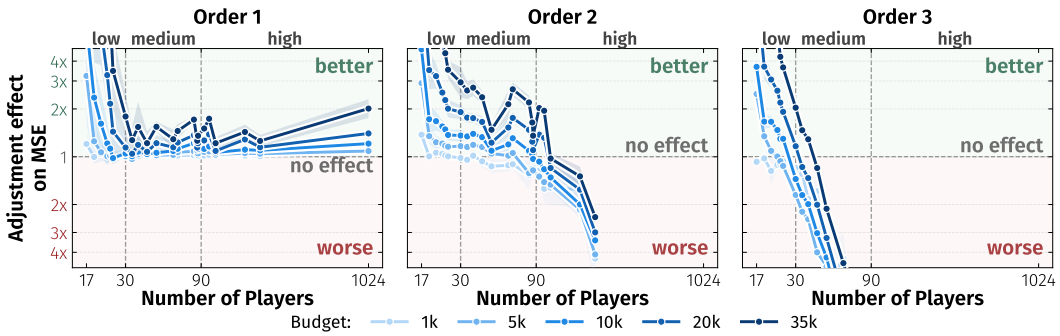


Figure 3: Comparison of ProxySHAP with and without MSR adjustment, measured by the MSE ratio. While MSR improves Shapley value approximation, it can degrade higher-order interaction estimates, as its variance is bounded by $n^{k-1}/|\mathcal{T}|$ for interactions of order k (see Theorem 3.3).

Witter et al. [63], MSR adjustment reliably improves approximation quality for probabilistic values, which our results in Figure 3 confirm. This motivates the adjusted ProxySHAP estimator:

$$\hat{\phi}_S^{\text{ProxySHAP}}(\nu;\mathcal{T}) = \phi_S^p(\hat{\nu}_\mathcal{T}) + \hat{\phi}_S^{\text{MSR}}(\nu - \hat{\nu}_\mathcal{T};\mathcal{T}).$$

While MSR often improves singleton estimates, this *does not* directly translate to improved interaction approximation, as evidenced in Figure 3. This is explained by the variance of the MSR estimator.

Theorem 3.3 (Variance growth of MSR under size-uniform sampling for Shapley interactions)**.** *Suppose coalitions are sampled i.i.d. uniformly over subset size,* $\mathbb{P}_{\text{sampling}}(T) = \frac{1}{(n+1)\binom{n}{|T|}}$.

Then, for any interaction $S \subseteq N$*, the MSR estimator of the Shapley interaction index satisfies*

$$\mathbb{V}\left[\hat{\phi}_S^{\text{MSR}}(\nu;\mathcal{T})\right] \leq \begin{cases} \mathcal{O}\left(\dfrac{\|\nu\|_\infty^2 \log n}{|\mathcal{T}|}\right), & |S| = 1, \\ \mathcal{O}\left(\dfrac{\|\nu\|_\infty^2 n^{|S|-1}}{|\mathcal{T}|}\right), & |S| \geq 2. \end{cases}$$

Theorem 3.3 shows that, for Shapley interactions with uniform over-sizes, the variance upper bound grows rapidly with the interaction order. In particular, when estimating all Shapley interactions up to order k, the dominant variance term scales as $\mathcal{O}\left(\frac{\|\nu\|_\infty^2 n^{k-1}}{|\mathcal{T}|}\right)$. Thus, the variance of MSR is governed by three quantities: the number of sampled coalitions $|\mathcal{T}|$, the number of players n, and the maximal interaction order k. We provide matching lower bounds and a related order-dependent result for Banzhaf interactions in Appendix A.2.

We evaluate the quality of MSR adjustment across all 32 benchmarked *TabArena* [14] datasets in Figure 3; further details are provided in Appendix D.7. Our findings are threefold. First, in the pairwise interaction approximation, MSR substantially reduces MSE as the evaluation budget increases. Second, this benefit diminishes as the number of players grows, making the adjustment impractical for high-dimensional games. Third, for higher-order interactions, adjustments can already harm performance in medium-sized games, even with large budgets, in contrast to the more robust gains observed for pairwise interactions. For Banzhaf interactions, similar insights apply, with adjustment being even less beneficial as player numbers increase (see Figure 15).

Consequently, while MSR adjustment remains effective for first-order interactions, its benefit for higher-order interactions is more nuanced. It substantially improves approximation quality for second-order interactions in small- to medium-sized games, but beyond pairwise interactions, it is mainly beneficial for smaller games. As a practical rule of thumb, we recommend applying MSR adjustment primarily in low-dimensional settings with $n < 30$. For larger games, adjustment should only be used when the sampling budget is sufficiently large relative to the dominant variance term, i.e., when $|\mathcal{T}| \gg n^{k-1}$ for interactions up to order k, which in our experiments occurs for medium-sized games only at budgets above 10,000.

3.3 The ProxySHAP Algorithm

We summarize ProxySHAP in Algorithm 1, using Algorithm 2 to compute the exact proxy interactions. Following Witter et al. [63], we use the same sampled coalitions $\mathcal{T}$ for both proxy fitting and residual estimation. This makes the residual game depend on the coalitions used to correct it, so Theorem 3.3 does not apply verbatim, but we find that reusing coalitions substantially improves the approximation quality for interactions as well (Section D.8). Unless stated otherwise, we sample coalitions uniformly over subset sizes without replacement [44], which can only reduce the variance relative to i.i.d. sampling [22]. ProxySHAP also works with other sampling schemes [17, 36], though these yield similar results (Section D.4).

Algorithm 1 `ProxySHAP(XGBoost, MSR)`

Require: value function ν, weight $p_t^s(n)$, sampling distribution $\mathbb{P}_{\text{sampling}}$, interactions of interest $\mathcal{S} \subseteq 2^N$
Ensure: $\hat{\phi}_S^p$ for all $S \in \mathcal{S}$

$\mathcal{T} \leftarrow$ sample according to $\mathbb{P}_{\text{sampling}}$
▷ ***Phase 1**: fit proxy $\hat{\nu}$ and residual game r* ◁
$\hat{\nu}_{\mathcal{T}} \leftarrow$ train XGBoost on $\{(T, \nu(T)) : T \in \mathcal{T}\}$
$r(T) \leftarrow \nu(T) - \hat{\nu}_{\mathcal{T}}(T)$ for all $T \in \mathcal{T}$
▷ ***Phase 2**: extract interactions* ◁
for $S \in \mathcal{S}$ **do**
  $\hat{\phi}_S^{\text{Proxy}} \leftarrow \phi_S^p(\hat{\nu}_{\mathcal{T}})$ ▷ *exact, Algorithm 2*
  $\hat{\phi}_S^{\text{MSR}} \leftarrow \hat{\phi}_S^{\text{MSR}}(r; \mathcal{T})$ ▷ *situational (Section 3.2)*
  $\hat{\phi}_S^p \leftarrow \hat{\phi}_S^{\text{Proxy}} + \hat{\phi}_S^{\text{MSR}}$
return $\hat{\phi}_S^p$ for all $S \in \mathcal{S}$

Computational Complexity. Fitting the XGBoost proxy $\hat{\nu}$ with m binary-encoded coalitions takes roughly $O(m \log m)$ time [9], and extracting one interaction with Algorithm 2 takes $O(n_{\text{trees}} \ell_{\max})$ time, where n_{trees} is the number of trees and $\ell_{\max}$ the maximum number of leaves per tree. Tree proxies are moreover inherently sparse. By Proposition 3.2, $\phi_S^p(\hat{\nu}_{\text{tree}}) = 0$ unless $S \subseteq L_j \cup R_j$ for some leaf j, so only interactions co-occurring on a root-to-leaf path are non-zero. Iterating over leaves, unadjusted ProxySHAP returns all of them in $O(n_{\text{trees}} \ell_{\max} 2^d)$ time for maximum tree depth d, against $O(n^k)$ interactions in total. The MSR adjustment is dense, since $\hat{\phi}_S^{\text{MSR}}(r; \mathcal{T})$ is generally non-zero for every $S \in \mathcal{S}$, raising the cost to $\mathcal{O}\left(|\mathcal{S}| \cdot (n_{\text{trees}} \ell_{\max} + m)\right)$. ProxySHAP thus offers a sparse but biased variant that scales beyond 1000 features (see Figure 5), and a consistent one that costs $O(n^k m)$ but pays off at lower player counts (see Figure 3). For a linear proxy, the number of fitted parameters equals the number of target interactions, so the demand grows as $O(mn^{2k})$ using ordinary least squares.

4 Experiments

We empirically evaluate ProxySHAP across a range of experimental settings and systematically compare it against standard baselines, including KernelSHAP-IQ [17], ProxySPEX [5], SHAP-IQ [i.e., MSR for interactions, 16], SVARM-IQ [31], and traditional permutation sampling [57, 58]. All experiments are based on `shapiq` [41], and code can be found in `https://github.com/Advueu963/ProxySHAP`.

Games. We evaluate ProxySHAP on local-explanation games across tabular benchmarks [41, 55], including TabArena [14], and established synthetic, vision, language, graph, and vision–language settings [2, 41, 43]. The underlying models include TabPFN [23], XGBoost [9], LightGBM [29], vision transformers [12], language models [51], CLIP [48], DINOv2 [47] and GNNs [64]; Table 3 summarizes all 54 datasets. We use exhaustive evaluation only for games with $n \leq 25$, since exact Shapley and Banzhaf interactions scale exponentially in n. For larger tabular games, tree models allow efficient ground-truth extraction via Algorithm 2, for graphs, we use `GraphSHAP-IQ` [43]. The synthetic SOUM games are built by randomly sampling Möbius coefficients, enabling ground-truth values. We hereby follow the procedure described in [58]. For CLIP, ground-truth interactions are unavailable, so we use task-specific faithfulness metrics. Further details are given in Section C.

Metrics. We measure approximation quality using relative mean squared error (Relative MSE; lower is better), defined as the sum of squared errors normalized by the sum of squared ground-truth interaction values. This normalization makes errors comparable across games with different interaction magnitudes. Unless stated otherwise, we report the mean and standard error of the mean (SEM) over 30 explained instances per dataset. We additionally report computational efficiency in terms of model evaluations and runtime (see Appendix D.3). To broaden the scope of evaluation, for

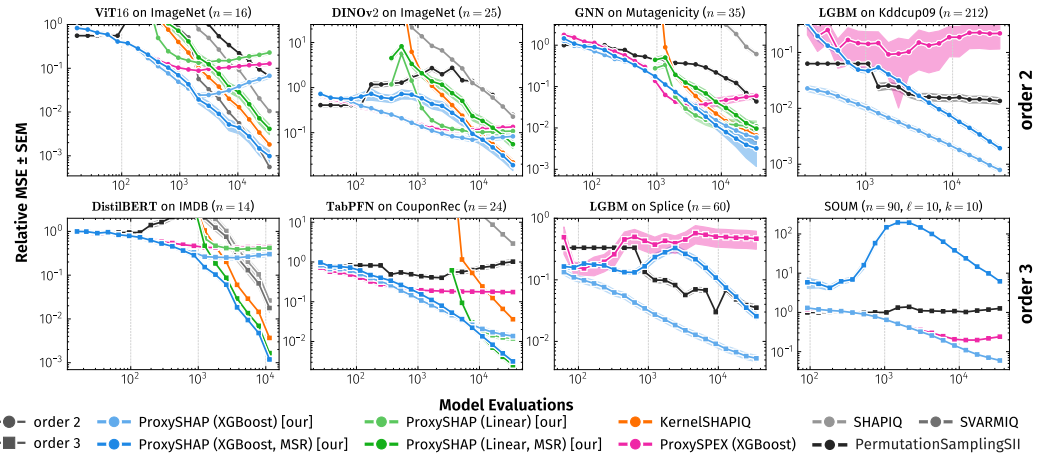


Figure 4: Approximation quality (Relative MSE) for Shapley interactions of ProxySHAP across different configurations and state-of-the-art baselines. Additional results for Shapley and Banzhaf interactions on all 54 datasets can be found in Figure 18 and Figure 19, respectively.

models such as CLIP, we evaluate explanation quality using R^2 faithfulness and the area under the insertion–deletion curve (see Appendix D.1; 2).

4.1 Approximation Quality

We first compare ProxySHAP to state-of-the-art baselines in terms of approximation quality (see Figure 4). We evaluate two proxy classes: a linear model with interaction features, ProxySHAP (Linear), and an XGBoost regressor with default hyperparameters, ProxySHAP (XGBoost), each with and without MSR adjustment.

Across datasets and interaction orders, ProxySHAP consistently outperforms all baselines, often by several orders of magnitude in relative MSE. MSR adjustment improves estimates for small games ($n < 30$) and for larger games when the sampling budget is sufficiently large, as in MUTAGENICITY. However, consistent with Section 3.2, its benefit decreases with interaction order and player count and can hurt performance in larger games, as observed on SPLICE. The linear proxy is competitive and sometimes even outperforms the tree-based proxy (see Figure 18). Yet, it becomes impractical for large feature counts and higher-order interactions due to its computational complexity (see Section 3). Tree-based proxies avoid this bottleneck and remain effective even when KernelSHAP-IQ is no longer feasible. Overall, XGBoost provides a scalable and accurate proxy, while MSR adjustment is most useful when the sampling budget is sufficiently large relative to the player count and interaction order. Additional results for further datasets and for both Shapley and Banzhaf interactions are provided in Section D.9.

4.2 Practical Considerations of ProxySHAP

HPO improves performance. The quality of ProxySHAP's approximation depends directly on the fitted proxy. We validate this via hyperparameter optimization (HPO), denoting the tuned variant as ProxySHAP (XGBoost+HPO) (see Appendix C.3). As shown in Figure 5 (left), HPO substantially improves over the default XGBoost proxy, with further examples in Appendix D.2. Since HPO is costly, we derive a cheaper ProxySHAP (XGBoost+HPO-Informed) variant from recurring strong HPO configurations. It performs on par with full HPO at small to medium budgets, highlighting the importance of proxy configuration.

Scalability of ProxySHAP. We demonstrate the scalability of ProxySHAP beyond 1000 features in Figure 5. Using ProxySHAP, we achieve better approximation quality at lower runtime, with the HPO-informed variant further improving it. Unlike ProxySHAP, ProxySPEX does not improve approximation quality with the HPO-informed configuration.

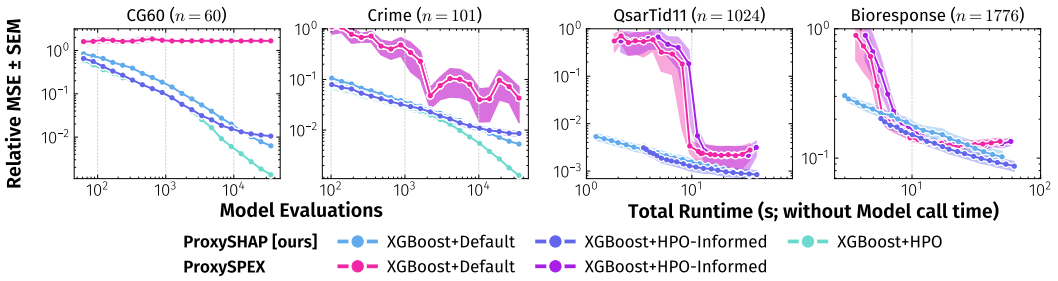


Figure 5: Relative MSE for pairwise Shapley interaction approximation of ProxySHAP with HPO (left) and for large n (right). Further results in Figure 9.

4.3 ProxySHAP Improves the Approximation of Faithful Interaction Explanations of CLIP

We demonstrate the broader applicability of ProxySHAP to improve the approximation of faithful interaction explanations of CLIP [FIxLIP, 2], a popular vision–language encoder architecture [48]. We follow the original experimental setup [2] and compare ProxySHAP to the original linear regression-based approximation, as well as ProxySPEX, in the FIxLIP game, where the goal is to explain the interaction between image and text inputs in CLIP (see Figure 1). Additional details and results are provided in Appendix D.1.

Setup. Similarly to Baniecki et al. [2], we analyze the CLIP model in two vision transformer variants: ViT-32 and ViT-16. We explain a sample of 200 image–text pairs from the MS COCO dataset [33], each having around 10–30 text tokens per input, yielding about 60–72 players in the final approximation game. As in the original work, we estimate the faithful Banzhaf interaction index of order 2 and quantify the quality of explanations with the area between the insertion/deletion curves (AID). Unlike the original work, we use smaller budgets ranging from 10^3 to 10^5 CLIP model calls, where each inference denotes that both the vision and language encoders are called on a single image–text pair.

Results. Figure 6 effectively demonstrates that ProxySHAP Pareto-dominates the original linear regression-based approximation, as well as ProxySPEX, in the FIxLIP game for both models. Overall, we find that the ProxySHAP adjustment incurs substantial computational overhead despite not improving the estimate. Note that in this setup, ProxySPEX and ProxySHAP with adjustment took about $10\times$ as long to compute as ProxySHAP without adjustment and the FIxLIP baseline. In Appendix D.1, we provide additional analysis measuring the R^2 metric, which yields a similar conclusion, and we ablate on FIxLIP's cross-modal estimator.

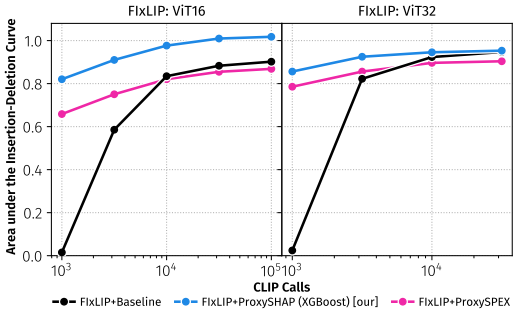


Figure 6: Area between the insertion/deletion curves (AID) for explaining two CLIP ViT variants on MS COCO with ProxySHAP, ProxySPEX, and the baseline.

5 Conclusion

We propose ProxySHAP, an efficient model-agnostic approximation method for the broad class of cardinal-probabilistic interaction indices. At its core, ProxySHAP extracts, in closed form, any cardinal-probabilistic interaction index, including Shapley and Banzhaf variants, from tree-based proxies. Notably, we achieve orders-of-magnitude speedups over Fourier-based extraction. By exploiting the linearity of interaction indices, ProxySHAP admits a clean decomposition into an exact proxy term and an MSR-based residual correction. Based on our extensive empirical evaluation and theoretical variance bound for MSR, we recommend applying the residual correction *situationally*: for interactions of order k, it is most useful in games with $n < 30$ players or when the budget satisfies $|\mathcal{T}| \gg n^{k-1}$. Empirically, ProxySHAP achieves strong approximation quality across an extensive

benchmark, outperforming ProxySPEX and KernelSHAP-IQ in both low- and high-budget regimes, and Pareto-dominating FIxLIP and ProxySPEX when applied to CLIP.

Limitations and future work. The quality of ProxySHAP's approximation depends directly on the trained proxy. We demonstrated that HPO substantially improves the performance of the XGBoost proxy but incurs significant computational overhead. A second caveat concerns inductive bias. In the tabular games, the explained model is often itself a gradient-boosted tree, which may favor a tree-based proxy. We therefore also evaluate on games whose underlying models are not tree-based, namely TabPFN, vision-and-language transformers, graph neural networks, CLIP, and SOUM, where ProxySHAP retains its advantage. In future work, we aim to explore other proxy classes, including their efficient interaction extraction.

Broader impact. We believe ProxySHAP can empower scientific researchers to quantify nonlinear dependencies among variables in complex systems, e.g., in physics [35] and material sciences [6]. We provide a highly scalable C++ implementation of our algorithm, enabling the quantification of interactions in foundation models spanning billions of parameters across large datasets.

Acknowledgments and Disclosure of Funding

Fabian Fumagalli and Maximilian Muschalik acknowledge funding by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation): TRR 318/3 2026 – 438445824. Hubert Baniecki was supported by the Foundation for Polish Science (FNP), and the Polish Ministry of Education and Science within the "Pearls of Science" program, project number PN/01/0087/2022.

References

[1] Hubert Baniecki, Giuseppe Casalicchio, Bernd Bischl, and Przemyslaw Biecek. Efficient and Accurate Explanation Estimation with Distribution Compression. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2025.

[2] Hubert Baniecki, Maximilian Muschalik, Fabian Fumagalli, Barbara Hammer, Eyke Hüllermeier, and Przemyslaw Biecek. Explaining Similarity in Vision-Language Encoders with Weighted Banzhaf Interactions. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

[3] John F Banzhaf III. Weighted voting doesn't work: A mathematical analysis. *Rutgers Law Review*, 19:317, 1964.

[4] Sebastian Bordt and Ulrike von Luxburg. From Shapley Values to Generalized Additive Models and back. In *Proceedings of the International Conference on Artificial Intelligence and Statistics (AISTATS)*, pages 709–745, 2023.

[5] Landon Butler, Abhineet Agarwal, Justin Singh Kang, Yigit Efe Erginbas, Bin Yu, and Kannan Ramchandran. Proxy-SPEX: Sample-efficient interpretability via sparse feature interactions in LLMs. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

[6] Yu-Xuan Cai, Hai-Yan Chen, Ya-Jing Qu, Wen-Hao Zhao, Mei-Ying Wang, Ying Chen, and Jin Ma. Improved vertical distribution prediction of soil vocs contamination in site-scale utilizing ensemble machine learning approach integrated with molecular descriptors. *Journal of Hazardous Materials*, page 139452, 2025.

[7] Javier Castro, Daniel Gómez, and Juan Tejada. Polynomial calculation of the Shapley value based on sampling. *Computers & Operations Research*, 36(5):1726–1730, 2009. doi: 10.1016/j.cor.2008.04.004.

[8] Javier Castro, Daniel Gómez, Elisenda Molina, and Juan Tejada. Improving polynomial estimation of the Shapley value by stratified random sampling with optimum allocation. *Computers & Operations Research*, 82:180–188, 2017. doi: 10.1016/j.cor.2017.01.019.

[9] Tianqi Chen and Carlos Guestrin. XGBoost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, pages 785–794. ACM, 2016. doi: 10.1145/2939672.2939785.

[10] Ian Covert and Su-In Lee. Improving KernelSHAP: Practical Shapley Value Estimation Using Linear Regression. In *Proceedings of the International Conference on Artificial Intelligence and Statistics (AISTATS)*, pages 3457–3465, 2021.

[11] Ian Connick Covert, Chanwoo Kim, Su-In Lee, James Zou, and Tatsunori Hashimoto. Stochastic amortization: A unified approach to accelerate feature and data attribution. In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

[12] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2021.

[13] James Enouen and Yan Liu. InstaSHAP: Interpretable additive models explain shapley values instantly. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2025.

[14] Nick Erickson, Lennart Purucker, Andrej Tschalzev, David Holzmüller, Prateek Mutalik Desai, David Salinas, and Frank Hutter. Tabarena: A living benchmark for machine learning on tabular data. In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2025.

[15] Katsushige Fujimoto, Ivan Kojadinovic, and Jean-Luc Marichal. Axiomatic characterizations of probabilistic and cardinal-probabilistic interaction indices. *Games and Economic Behavior*, 55(1):72–99, 2006. doi: 10.1016/j.geb.2005.03.002.

[16] Fabian Fumagalli, Maximilian Muschalik, Patrick Kolpaczki, Eyke Hüllermeier, and Barbara Hammer. SHAP-IQ: Unified Approximation of any-order Shapley Interactions. In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, pages 11515–11551, 2023.

[17] Fabian Fumagalli, Maximilian Muschalik, Patrick Kolpaczki, Eyke Hüllermeier, and Barbara Hammer. KernelSHAP-IQ: Weighted Least Square Optimization for Shapley Interactions. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 14308–14342, 2024.

[18] Ali Gorji, Andisheh Amrollahi, and Andreas Krause. SHAP values via sparse fourier representation. In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2025.

[19] 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.

[20] Léo Grinsztajn, Klemens Flöge, Oscar Key, Felix Birkel, Philipp Jund, Brendan Roof, Benjamin Jäger, Dominik Safaric, Simone Alessi, Adrian Hayler, Mihir Manium, Rosen Yu, Felix Jablonski, Shi Bin Hoo, Anurag Garg, Jake Robertson, Magnus Bühler, Vladyslav Moroshan, Lennart Purucker, Clara Cornu, Lilly Charlotte Wehrhahn, Alessandro Bonetto, Bernhard Schölkopf, Sauraj Gambhir, Noah Hollmann, and Frank Hutter. Tabpfn-2.5: Advancing the state of the art in tabular foundation models. *CoRR*, abs/2511.08667, 2025. doi: 10.48550/ARXIV.2511.08667.

[21] Naofumi Hama, Masayoshi Mase, and Art B. Owen. Deletion and insertion tests in regression models. *Journal of Machine Learning Research*, 24:290:1–290:38, 2023.

[22] Wassily Hoeffding. Probability inequalities for sums of bounded random variables. *Journal of the American Statistical Association*, 58:13–30, 1963.

[23] Noah Hollmann, Samuel Müller, Lennart Purucker, Arjun Krishnakumar, Max Körfer, Shi Bin Hoo, Robin Tibor Schirrmeister, and Frank Hutter. Accurate predictions on small data with a tabular foundation model. *Nature*, 637(8045):319–326, 2025. doi: 10.1038/s41586-024-08328-6.

[24] Neil Jethani, Mukund Sudarshan, Ian Connick Covert, Su-In Lee, and Rajesh Ranganath. FastSHAP: Real-Time Shapley Value Estimation. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2022.

[25] Ruoxi Jia, David Dao, Boxin Wang, Frances Ann Hubis, Nick Hynes, Nezihe Merve Gürel, Bo Li, Ce Zhang, Dawn Song, and Costas J. Spanos. Towards efficient data valuation based on the shapley value. In *Proceedings of the International Conference on Artificial Intelligence and Statistics (AISTATS)*, pages 1167–1176, 2019.

[26] Peng Jin, Hao Li, Li Yuan, Shuicheng Yan, and Jie Chen. Hierarchical Banzhaf interaction for general video-language representation learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 47(3):2125–2139, 2025.

[27] Justin Singh Kang, Landon Butler, Abhineet Agarwal, Yigit Efe Erginbas, Ramtin Pedarsani, Bin Yu, and Kannan Ramchandran. SPEX: Scaling feature interaction explanations for LLMs. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 28878–28903, 2025.

[28] Jeroen Kazius, Ross McGuire, and Roberta Bursi. Derivation and validation of toxicophores for mutagenicity prediction. *Journal of Medicinal Chemistry*, 48(1):312–320, 2005. doi: 10.1021/jm040835a.

[29] Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. LightGBM: A Highly Efficient Gradient Boosting Decision Tree. In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, pages 3146–3154, 2017.

[30] Patrick Kolpaczki, Viktor Bengs, Maximilian Muschalik, and Eyke Hüllermeier. Approximating the shapley value without marginal contributions. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, pages 13246–13255, 2024.

[31] Patrick Kolpaczki, Maximilian Muschalik, Fabian Fumagalli, Barbara Hammer, and Eyke Hüllermeier. SVARM-IQ: efficient approximation of any-order shapley interactions through stratification. In *Proceedings of the International Conference on Artificial Intelligence and Statistics (AISTATS)*, pages 3520–3528, 2024.

[32] Quentin Lhoest, Albert Villanova del Moral, Patrick von Platen, Thomas Wolf, Mario Šaško, Yacine Jernite, Abhishek Thakur, Lewis Tunstall, Suraj Patil, Mariama Drame, Julien Chaumond, Julien Plu, Joe Davison, Simon Brandeis, Victor Sanh, Teven Le Scao, Kevin Canwen Xu, Nicolas Patry, Steven Liu, Angelina McMillan-Major, Philipp Schmid, Sylvain Gugger, Nathan Raw, Sylvain Lesage, Anton Lozhkov, Matthew Carrigan, Théo Matussière, Leandro von Werra, Lysandre Debut, Stas Bekman, and Clément Delangue. Datasets: A community library for natural language processing. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, (EMNLP 2021)*, pages 175–184. Association for Computational Linguistics, 2021.

[33] Tsung-Yi Lin, Michael Maire, Serge J. Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: common objects in context. In *European Conference on Computer Vision ECCV*, volume 8693, pages 740–755, 2014.

[34] Marius Lindauer, Katharina Eggensperger, Matthias Feurer, André Biedenkapp, Difan Deng, Carolin Benjamins, Tim Ruhkopf, René Sass, and Frank Hutter. Smac3: A versatile bayesian optimization package for hyperparameter optimization. *Journal of Machine Learning Research*, 23(54):1–9, 2022.

[35] Tianran Liu, Nicky Evans, Kangyu Ji, Ronaldo Lee, Aaron Zhu, Vinn Nguyen, James Serdy, Elizabeth M Wall, Yongli Lu, Florian A Formica, et al. Disentangling environmental effects on perovskite solar cell performance via interpretable machine learning. *ACS Energy Letters*, 11: 1609–1617, 2026.

[36] Scott M. Lundberg and Su-In Lee. A Unified Approach to Interpreting Model Predictions. In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, pages 4765–4774, 2017.

[37] Scott M. Lundberg, Gabriel G. Erion, Hugh Chen, Alex J. DeGrave, Jordan M. Prutkin, Bala Nair, Ronit Katz, Jonathan Himmelfarb, Nisha Bansal, and Su-In Lee. From local explanations to global understanding with explainable AI for trees. *Nature Machine Intelligence*, 2(1):56–67, 2020. doi: 10.1038/s42256-019-0138-9.

[38] Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. Learning word vectors for sentiment analysis. In *Proceedings of the Association for Computational Linguistics: Human Language Technologies (HLT)*, pages 142–150, 2011.

[39] Majid Mohammadi, Siu Lun Chau, and Krikamol Muandet. Computing exact Shapley values in polynomial time for product-kernel methods. *arXiv preprint, arXiv:2505.16516*, 2025.

[40] Christoph Molnar, Gunnar König, Julia Herbinger, Timo Freiesleben, Susanne Dandl, Christian A Scholbeck, Giuseppe Casalicchio, Moritz Grosse-Wentrup, and Bernd Bischl. General pitfalls of model-agnostic interpretation methods for machine learning models. In *xxAI-Beyond Explainable AI: International Workshop, Held in Conjunction with ICML 2020*, pages 39–68, 2022.

[41] Maximilian Muschalik, Hubert Baniecki, Fabian Fumagalli, Patrick Kolpaczki, Barbara Hammer, and Eyke Hüllermeier. shapiq: Shapley Interactions for Machine Learning. In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, pages 130324–130357, 2024.

[42] Maximilian Muschalik, Fabian Fumagalli, Barbara Hammer, and Eyke Hüllermeier. Beyond TreeSHAP: Efficient Computation of Any-Order Shapley Interactions for Tree Ensembles. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, pages 14388–14396, 2024. doi: 10.1609/aaai.v38i13.29352.

[43] Maximilian Muschalik, Fabian Fumagalli, Paolo Frazzetto, Janine Strotherm, Luca Hermes, Alessandro Sperduti, Eyke Hüllermeier, and Barbara Hammer. Exact Computation of Any-Order Shapley Interactions for Graph Neural Networks. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2025.

[44] Christopher Musco and R. Teal Witter. Provably Accurate Shapley Value Estimation via Leverage Score Sampling. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2025.

[45] Alexander Nadel and Ron Wettenstein. From decision trees to boolean logic: A fast and unified SHAP algorithm. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, pages 24476–24485, 2026. doi: 10.1609/AAAI.V40I29.39630.

[46] Lars H. B. Olsen, Ingrid K. Glad, Martin Jullum, and Kjersti Aas. Using Shapley values and variational autoencoders to explain predictive models with dependent mixed features. *Journal of Machine Learning Research*, 23(213):1–51, 2022.

[47] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mido Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Hervé Jégou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. Dinov2: Learning robust visual features without supervision. *Trans. Mach. Learn. Res.*, 2024, 2024.

[48] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *Proceedings of the International Conference on Machine Learning ICML*, pages 8748–8763, 2021.

[49] Benedek Rozemberczki, Lauren Watson, Péter Bayer, Hao-Tsung Yang, Oliver Kiss, Sebastian Nilsson, and Rik Sarkar. The shapley value in machine learning. In *Proceedings of International Joint Conference on Artificial Intelligence (IJCAI)*, pages 5572–5579, 2022.

[50] Benjamin Sanchez-Lengeling, Jennifer Wei, Brian Lee, Emily Reif, Peter Wang, Wesley Qian, Kevin McCloskey, Lucy Colwell, and Alexander Wiltschko. Evaluating attribution for graph neural networks. In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, volume 33, pages 5898–5910, 2020.

[51] Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *CoRR*, abs/1910.01108, 2019.

[52] Meghdut Sengupta, Maximilian Muschalik, Fabian Fumagalli, Barbara Hammer, Eyke Hüllermeier, Debanjan Ghosh, and Henning Wachsmuth. Investigating the impact of conceptual metaphors on LLM-based NLI through shapley interactions. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pages 17393–17403, 2025.

[53] L. S. Shapley. A Value for n-Person Games. In *Contributions to the Theory of Games (AM-28), Volume II*, pages 307–318. Princeton University Press, 1953.

[54] Maximilian Spliethöver, Tim Knebler, Fabian Fumagalli, Maximilian Muschalik, Barbara Hammer, Eyke Hüllermeier, and Henning Wachsmuth. Adaptive prompting: Ad-hoc prompt composition for social bias detection. In *Proceedings of the Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics*, 2025.

[55] Erik Strumbelj and Igor Kononenko. An Efficient Explanation of Individual Classifications using Game Theory. *Journal of Machine Learning Research*, 11:1–18, 2010.

[56] Erik Strumbelj and Igor 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.

[57] Mukund Sundararajan, Kedar Dhamdhere, and Ashish Agarwal. The Shapley Taylor Interaction Index. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 9259–9268, 2020.

[58] Che-Ping Tsai, Chih-Kuan Yeh, and Pradeep Ravikumar. Faith-Shap: The Faithful Shapley Interaction Index. *Journal of Machine Learning Research*, 24(94):1–42, 2023.

[59] Michael Tschannen, Alexey Gritsenko, Xiao Wang, Muhammad Ferjad Naeem, Ibrahim Alabdulmohsin, Nikhil Parthasarathy, Talfan Evans, Lucas Beyer, Ye Xia, Basil Mustafa, et al. SigLIP 2: Multilingual vision-language encoders with improved semantic understanding, localization, and dense features. *arXiv preprint arXiv:2502.14786*, 2025.

[60] Jiachen T. Wang and Ruoxi Jia. Data banzhaf: A robust data valuation framework for machine learning. In *Proceedings of the International Conference on Artificial Intelligence and Statistics (AISTATS)*, pages 6388–6421, 2023.

[61] Jiachen T. Wang, Prateek Mittal, and Ruoxi Jia. Efficient data Shapley for weighted nearest neighbor algorithms. In *Proceedings of the International Conference on Artificial Intelligence and Statistics (AISTATS)*, pages 2557–2565, 2024.

[62] Marcel Wever, Maximilian Muschalik, Fabian Fumagalli, and Marius Lindauer. HyperSHAP: Shapley Values and Interactions for Hyperparameter Importance. In *AAAI*, 2026.

[63] R. Teal Witter, Yurong Liu, and Christopher Musco. Regression-adjusted monte carlo estimators for shapley values and probabilistic values. In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2025.

[64] Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. How powerful are graph neural networks? In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2019.

[65] Artjom Zern, Klaus Broelemann, and Gjergji Kasneci. Interventional SHAP values and interaction values for piecewise linear regression trees. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, pages 11164–11173, 2023.

[66] Chenyang Zhao, Kun Wang, Janet H. Hsiao, and Antoni B. Chan. Grad-ECLIP: Gradient-based visual and textual explanations for CLIP. In *Proceedings of the International Conference on Machine Learning ICML*, 2024.

Appendix for "Proxy-Based Approximation of Shapley and Banzhaf Interactions"

A Proofs **16**

A.1 Proof of Proposition 3.2 16

A.2 Proof of Theorem 3.3 18

A.3 Closed Forms for Proposition 3.2 25

B Generalization of Interventional TreeSHAP **31**

C Experimental Details **32**

C.1 Datasets 32

C.2 Computational Resources 33

C.3 Hyperparameter Optimization 33

C.4 Baselines 34

D Additional Approximation Results **35**

D.1 Additional Experiments with FIxLIP 35

D.2 XGBoost Default for Large Player Counts 35

D.3 Runtime 37

D.4 Ablation Details 39

D.5 Fourier Extraction vs. Interventional Extraction 39

D.6 Detailed Comparison of ProxySHAP and ProxySPEX 40

D.7 When to use Adjustment? 42

D.8 Shared vs. Disjoint Subsets in ProxySHAP 43

D.9 Approximation Quality 44

E Additional Related Work **45**

Table 1: Weights of cardinal-probabilistic interaction indices.

Weight	Banzhaf	Shapley	Möbius
$p_t^s(n)$	$(\frac{1}{2})^{n-s}$	$\frac{1}{(n-s+1)\binom{n-s}{t}}$	$\mathbb{1}[t=0]$
$q_t^s(n)$	$(\frac{1}{2})^{t-s}$	$\frac{1}{t-s+1}$	$\mathbb{1}[t=s]$

A Proofs

A central contribution of our method is an extension of the algorithm of Zern et al. [65] to compute cardinal-probabilistic interactions of arbitrary order (see Algorithm 2). We establish a general extraction result in Section A.1, with specialized proofs for commonly used indices—including the Banzhaf Interaction Index (BII)[19], the Chaining Interaction Index (CII) [15], the Faithful Banzhaf Interaction Index (FBII)[58], Faithful Shapley Interaction Index (FSII) [58] and Shapley Taylor Interaction Index (STII) [57]—provided in Section A.3. These results rely repeatedly on Lemma A.1 and Lemma A.2.

We prove the variance growth of MSR for general interaction indices under general sampling schemes in Appendix A.2. We investigate the growth of the variance of MSR for the Shapley Interaction Index and the Banzhaf Interaction Index, using sampling uniformly over sizes. We further showcase that the derived upper bounds are tight by providing matching lower bounds for both indices and sampling schemes.

A.1 Proof of Proposition 3.2

Before proving the main result, we first establish two lemmas that are instrumental for the proof of Proposition 3.2. These will also be used to derive the closed-form expression for the Banzhaf Interaction Index, Chaining Interaction Index, Faithful Banzhaf Interaction Index, and Faithful Shapley Interaction Index in Section A.3.

Lemma A.1. *For $A \subseteq B \subseteq N$, we have*

$$\mathbb{1}_{[A,B]} = \sum_{T \subseteq N \setminus B} (-1)^t \mathbb{1}_{[T \cup A, N]}$$

Proof. See proof of Lemma 1 in Zern et al. [65]. □

Lemma A.2. *Let N be the set of players, $A \subseteq B \subseteq N$, and the game $\mathbb{1}_{[A,B]}(T) = 1$ iff $A \subseteq T \subseteq B$. Then it holds that the Möbius value for a set $S \subseteq N$ equals*

$$m_S(\mathbb{1}_{[A,B]}) = (-1)^{|(N \setminus B) \cap S|} \cdot \mathbb{1}_{A \subseteq S, (B \setminus A) \cap S = \emptyset}$$

Proof. Let N be the set of players, $A, B \subseteq N$ such that $A \subseteq B$, and the game $\mathbb{1}_{[A,B]}$ be defined as above, and $S \subseteq N$. Then

$$m_S(\mathbb{1}_{[A,B]}) = \sum_{L \subseteq S} (-1)^{s-l} \mathbb{1}_{[A,B]}(L).$$

Note that we must have $A \subseteq S$, as otherwise the game value is always zero. We therefore define $F = (N \setminus B) \cap S$ and $G = (B \setminus A) \cap S$ such that $S = A \cup F \cup G$, which yields:

$$m_S(\mathbb{1}_{[A,B]}) = \sum_{L \subseteq G \cup F \cup A} (-1)^{f+g+a-l} \mathbb{1}_{[A,B]}(L).$$

Observe that only those L have non-zero game values for which $A \subseteq L$ and $L \cap F = \emptyset$, as otherwise $L \subseteq B$ would not hold. Therefore, we can equivalently express the sum as

$$\begin{aligned} m_S(\mathbb{1}_{[A,B]}) &= (-1)^{f+g} \sum_{L \subseteq G} (-1)^{-l} \\ &= (-1)^{f+g} \sum_{l=0}^{g} \binom{g}{l} (-1)^{-l}. \end{aligned}$$

The latter term always equals 0 for $g \neq 0$, which yields our second condition $G = (B \setminus A) \cap S = \emptyset$. Finally, we have

$$m_S(\mathbb{1}_{[A,B]}) = (-1)^f = (-1)^{|(N \setminus B) \cap S|}$$

if and only if (i) $A \subseteq S$ and $(B \setminus A) \cap S = \emptyset$, which concludes the proof. □

We now prove the main result of this section, which provides a closed-form expression for the cardinal-probabilistic interaction indices of tree-based proxies.

Proposition 3.2. For $S \subseteq N$ and leaves $j \in \mathcal{L}$, we have

$$\phi_S^p(\hat{\nu}_{\text{tree}}) = \sum_{\substack{j \in \mathcal{L} \\ S \subseteq L_j \cup R_j}} c_j \cdot \lambda\left(|L_j|, |R_j|, |S \cap L_j|, |S|\right),$$

where $\lambda(\ell, r, u, s) := \sum_{i=0}^{\ell-u} (-1)^{i+u} \binom{\ell-u}{i} q_{i+u+r}^s(n)$.

Proof of Proposition 3.2. Let $\mathbb{1}_{[A,B]} : 2^N \to \{0,1\}$ be an indicator game such that $\mathbb{1}[A,B](T) = \mathbb{1}[A \subseteq T \subseteq B]$. We observe that $\hat{\nu}_{\text{tree}}$ can be written as a sum of indicator games :

$$\begin{aligned}
\phi_S^p(\hat{\nu}_{\text{tree}}) &= \phi_S^p\left(\sum_{j \in \mathcal{L}} c_j \cdot \mathbb{1}[R_j,\ N \setminus L_j]\right) \\
&= \sum_{j \in \mathcal{L}} c_j \cdot \phi_S^p\left(\mathbb{1}[R_j,\ N \setminus L_j]\right)
\end{aligned}$$

Using Lemma A.1 we obtain:

$$\begin{aligned}
\phi_S^p(\hat{\nu}_{\text{tree}}) &= \sum_{j \in \mathcal{L}} c_j \phi_S^p\left(\sum_{T \subseteq L_j} (-1)^t \mathbb{1}_{[T \cup R_j, N]}\right) \\
&= \sum_{j \in \mathcal{L}} c_j \sum_{T \subseteq L_j} (-1)^t \phi_S^p(\mathbb{1}_{[T \cup R_j, N]}) \\
&= \sum_{j \in \mathcal{L}} c_j \sum_{T \subseteq L_j} (-1)^t \sum_{V \supseteq S} q_v^s(n) m_V(\mathbb{1}_{[T \cup R_j, N]}) \\
&= \sum_{j \in \mathcal{L}} c_j \sum_{\substack{T \subseteq L_j \\ T \cup R_j \supseteq S}} (-1)^t q_{t+r_j}^s(n)
\end{aligned}$$

In the second-to-last step we use (4); in the last step we use $m_V(\mathbb{1}_{[R_j \cup T, N]}) = 1$ iff $R_j \cup T = V$ and 0 otherwise (a direct consequence of $B = N$ in Lemma A.2). Notice that we can only update those interactions S in leaf j for which it holds $S \subseteq L_j \cup R_j$, since the condition $T \cup R_j \supseteq S$ does not hold otherwise, and as such $\phi_S^p(\mathbb{1}[R_j, N \setminus L_j]) = 0$. We define $S_0 := S \cap L_j$ and let $B \subseteq L_j \setminus S_0$, so that $T = S_0 \cup B$, which gives rise to

$$\begin{aligned}
\phi_S^p(\hat{\nu}_{\text{tree}}) &= \sum_{\substack{j \in \mathcal{L} \\ S \subseteq L_j \cup R_j}} c_j \sum_{B \subseteq L_j \setminus S_0} (-1)^{b+s_0} q_{b+s_0+r_j}^s(n) \\
&= \sum_{\substack{j \in \mathcal{L} \\ S \subseteq L_j \cup R_j}} c_j \sum_{i=0}^{\ell_j - s_0} (-1)^{i+s_0} \binom{\ell_j - s_0}{i} q_{i+s_0+r_j}^s(n)
\end{aligned}$$

Defining $\lambda(\ell, r, u, s) := \sum_{i=0}^{\ell-u} (-1)^{i+u} \binom{\ell-u}{i} q_{i+u+r}^s(n)$ we obtain:

$$\phi_S^p(\hat{\nu}_{\text{tree}}) = \sum_{\substack{j \in \mathcal{L} \\ S \subseteq L_j \cup R_j}} c_j \lambda(|L_j|, |R_j|, |S \cap L_j|, |S|)$$

which concludes the proof. □

A.2 Proof of Theorem 3.3

We first derive a general variance identity for the MSR estimator under arbitrary sampling distributions in Theorem A.3. We then specialize this identity to proportional sampling and, most importantly, uniform over sizes for the Shapley interaction index. The latter specialization yields Theorem 3.3 from the main paper. Corresponding lower bounds are shown in Corollary A.7. The corresponding bound for the Banzhaf interaction index can be found in Corollary A.8.

Theorem A.3 (General variance identity for MSR)**.** *Let $S \subseteq N$ with $s := |S|$, and let $T_1, \dots, T_{|\mathcal{T}|}$ be sampled i.i.d. from a sampling distribution $\mathbb{P}_{\text{sampling}}$ on 2^N with $\mathbb{P}_{\text{sampling}}(T) > 0$ for all $T \subseteq N$. Then the MSR estimator*

$$\hat{\phi}_S^{\text{MSR}}(\nu; \mathcal{T}) = \frac{1}{|\mathcal{T}|} \sum_{i=1}^{|\mathcal{T}|} \nu(T_i) \, \frac{(-1)^{s-|S\cap T_i|} \, p^s_{|T_i|-|S\cap T_i|}(n)}{\mathbb{P}_{\text{sampling}}(T_i)}$$

satisfies

$$\mathbb{V}[\hat{\phi}_S^{\text{MSR}}(\nu; \mathcal{T})] = \frac{1}{|\mathcal{T}|} \left(\sum_{T \in 2^N} \nu(T)^2 \frac{\left(p^s_{|T|-|S\cap T|}(n)\right)^2}{\mathbb{P}_{\text{sampling}}(T)} - \phi_S^p(\nu)^2 \right).$$

In particular,

$$\mathbb{V}[\hat{\phi}_S^{\text{MSR}}(\nu; \mathcal{T})] \leq \frac{\|\nu\|_\infty^2}{|\mathcal{T}|} \Gamma_S(\mathbb{P}_{\text{sampling}}),$$

where

$$\Gamma_S(\mathbb{P}_{\text{sampling}}) := \sum_{T \in 2^N} \frac{\left(p^s_{|T|-|S\cap T|}(n)\right)^2}{\mathbb{P}_{\text{sampling}}(T)}.$$

Proof. Define the random variable

$$X := \nu(T) \, \frac{(-1)^{s-|S\cap T|} \, p^s_{|T|-|S\cap T|}(n)}{\mathbb{P}_{\text{sampling}}(T)}, \qquad T \sim \mathbb{P}_{\text{sampling}}.$$

Since $T_1, \dots, T_{|\mathcal{T}|}$ are i.i.d., the corresponding variables

$$X_i := \nu(T_i) \, \frac{(-1)^{s-|S\cap T_i|} \, p^s_{|T_i|-|S\cap T_i|}(n)}{\mathbb{P}_{\text{sampling}}(T_i)}$$

are i.i.d. as well, and

$$\hat{\phi}_S^{\text{MSR}}(\nu; \mathcal{T}) = \frac{1}{|\mathcal{T}|} \sum_{i=1}^{|\mathcal{T}|} X_i.$$

Using independence, we obtain

$$\begin{aligned} \mathbb{V}[\hat{\phi}_S^{\text{MSR}}(\nu; \mathcal{T})] &= \mathbb{V}\left[\frac{1}{|\mathcal{T}|} \sum_{i=1}^{|\mathcal{T}|} X_i \right] \\ &= \frac{1}{|\mathcal{T}|^2} \sum_{i=1}^{|\mathcal{T}|} \mathbb{V}[X_i] \\ &= \frac{1}{|\mathcal{T}|} \mathbb{V}[X]. \end{aligned} \tag{9}$$

We now expand the variance of X:

$$\mathbb{V}[X] = \mathbb{E}[X^2] - \mathbb{E}[X]^2.$$

For the second moment, we compute

$$\mathbb{E}[X^2] = \sum_{T \in 2^N} \mathbb{P}_{\text{sampling}}(T)\, \nu(T)^2 \left(\frac{p^s_{|T|-|S\cap T|}(n)}{\mathbb{P}_{\text{sampling}}(T)} \right)^2$$
$$= \sum_{T \in 2^N} \nu(T)^2 \frac{\left(p^s_{|T|-|S\cap T|}(n)\right)^2}{\mathbb{P}_{\text{sampling}}(T)}, \tag{10}$$

where the sign disappears since it is squared. For the first moment,

$$\mathbb{E}[X] = \sum_{T \in 2^N} \mathbb{P}_{\text{sampling}}(T)\, \nu(T)\, \frac{(-1)^{s-|S\cap T|}\, p^s_{|T|-|S\cap T|}(n)}{\mathbb{P}_{\text{sampling}}(T)}$$
$$= \sum_{T \in 2^N} \nu(T)\, (-1)^{s-|S\cap T|}\, p^s_{|T|-|S\cap T|}(n)$$
$$= \phi^p_S(\nu). \tag{11}$$

Substituting (10) and (11) into the variance expansion yields

$$\mathbb{V}[X] = \sum_{T \in 2^N} \nu(T)^2 \frac{\left(p^s_{|T|-|S\cap T|}(n)\right)^2}{\mathbb{P}_{\text{sampling}}(T)} - \phi^p_S(\nu)^2.$$

Combining this with (9) proves the exact variance identity. The upper bound follows from $\phi^p_S(\nu)^2 \geq 0$ and $\nu(T)^2 \leq \|\nu\|^2_\infty$ for all $T \subseteq N$. □

Corollary A.4 (Variance of MSR under proportional sampling)**.** *Under the assumptions of Theorem A.3, suppose the coalitions are sampled proportionally to the interaction weights, i.e.*

$$\mathbb{P}_{\text{sampling}}(T) \propto p^s_{|T|-|S\cap T|}(n), \qquad \textit{for all } T \subseteq N.$$

Then

$$\mathbb{V}[\hat{\phi}^{\text{MSR}}_S(\nu; \mathcal{T})] \leq \frac{4^s}{|\mathcal{T}|}\, \|\nu\|^2_\infty.$$

Proof. By Theorem A.3,

$$\mathbb{V}[\hat{\phi}^{\text{MSR}}_S(\nu; \mathcal{T})] \leq \frac{\|\nu\|^2_\infty}{|\mathcal{T}|}\, \Gamma_S(\mathbb{P}_{\text{sampling}}).$$

Under proportional sampling,

$$\mathbb{P}_{\text{sampling}}(T) = \frac{p^s_{|T|-|S\cap T|}(n)}{2^s},$$

and hence

$$\Gamma_S(\mathbb{P}_{\text{sampling}}) = \sum_{T \subseteq N} \frac{\left(p^s_{|T|-|S\cap T|}(n)\right)^2}{\mathbb{P}_{\text{sampling}}(T)}$$
$$= \sum_{T \subseteq N} \mathbb{P}_{\text{sampling}}(T) \left(\frac{p^s_{|T|-|S\cap T|}(n)}{\mathbb{P}_{\text{sampling}}(T)} \right)^2$$
$$= \sum_{T \subseteq N} \mathbb{P}_{\text{sampling}}(T)\, (2^s)^2$$
$$= 4^s \sum_{T \subseteq N} \mathbb{P}_{\text{sampling}}(T)$$
$$= 4^s.$$

Substituting this into the general bound proves the claim. □

Corollary A.5 (Asymptotics of uniform over sizes sampling variance factor for SII)**.** *Suppose* $\mathbb{P}_{\text{sampling}}$ *is given by,*

$$\mathbb{P}_{\text{sampling}}(T) = \frac{1}{(n+1)\binom{n}{|T|}},$$

and let $p_t^s(n)$ *denote the SII weights,*

$$p_t^s(n) = \frac{1}{(n-s+1)\binom{n-s}{t}}.$$

Then, for fixed interaction order s*,*

$$\Gamma_S(\mathbb{P}_{\text{sampling}}) = \begin{cases} O(\log n), & s = 1, \\ O(n^{s-1}), & s \geq 2. \end{cases}$$

Proof. We rewrite $\Gamma_S(\mathbb{P}_{\text{sampling}})$ as a double sum over the overlap size $r = |S \cap T|$ and the outside size $q = |T \setminus S|$, then treat the cases $s = 1$ and $s \geq 2$ separately.

Step 1: Rewriting $\Gamma_S(\mathbb{P}_{\text{sampling}})$**.** Using the definition of $\Gamma_S(\mathbb{P}_{\text{sampling}})$ from Theorem A.3, together with the SII weights and the sampling probabilities, we obtain

$$\begin{aligned} \Gamma_S(\mathbb{P}_{\text{sampling}}) &= \sum_{T \subseteq N} \frac{\left(p^s_{|T|-|S\cap T|}(n)\right)^2}{\mathbb{P}_{\text{sampling}}(T)} \\ &= \sum_{T \subseteq N} \frac{1}{(n-s+1)^2\binom{n-s}{|T\setminus S|}^2} \cdot (n+1)\binom{n}{|T|}. \end{aligned}$$

We group coalitions $T \subseteq N$ according to

$$r := |S \cap T|, \qquad q := |T \setminus S|.$$

Then $|T| = r + q$, and for fixed (r, q) there are exactly

$$\binom{s}{r}\binom{n-s}{q}$$

coalitions with these values. Hence

$$\Gamma_S(\mathbb{P}_{\text{sampling}}) = \frac{n+1}{(n-s+1)^2} \sum_{r=0}^{s} \binom{s}{r} \sum_{q=0}^{n-s} \frac{\binom{n}{r+q}}{\binom{n-s}{q}}.$$

Next, we rewrite the binomial ratio:

$$\begin{aligned} \frac{\binom{n}{r+q}}{\binom{n-s}{q}} &= \frac{n!}{(r+q)!(n-r-q)!} \cdot \frac{q!(n-s-q)!}{(n-s)!} \\ &= \frac{n!}{(n-s)!} \cdot \frac{q!}{(r+q)!} \cdot \frac{(n-s-q)!}{(n-r-q)!}. \end{aligned}$$

Using the rising factorial

$$(a)_k := a(a+1)\cdots(a+k-1),$$

this becomes

$$\frac{\binom{n}{r+q}}{\binom{n-s}{q}} = \frac{(n-s+1)_s}{(q+1)_r\,(n-s-q+1)_{s-r}}.$$

Therefore,

$$\Gamma_S(\mathbb{P}_{\text{sampling}}) = \frac{(n+1)(n-s+1)_s}{(n-s+1)^2} \sum_{r=0}^{s} \binom{s}{r} \sum_{q=0}^{n-s} \frac{1}{(q+1)_r\,(n-s-q+1)_{s-r}}. \tag{12}$$

Step 2: The case $s = 1$. Assume first that $s = 1$. Then $r \in \{0, 1\}$, so (12) becomes

$$\Gamma_S(\mathbb{P}_{\text{sampling}}) = \frac{n+1}{n} \sum_{r=0}^{1} \binom{1}{r} \sum_{q=0}^{n-1} \frac{1}{(q+1)_r \, (n-q)_{1-r}}.$$

We used $(n-s+1)_s = (n)_1 = n$ and $(n-s+1)^2 = n^2$ when $s = 1$. For $r = 0$,

$$\sum_{q=0}^{n-1} \frac{1}{n-q} = \sum_{j=1}^{n} \frac{1}{j} = H_n,$$

and similarly for $r = 1$,

$$\sum_{q=0}^{n-1} \frac{1}{q+1} = \sum_{j=1}^{n} \frac{1}{j} = H_n.$$

Hence

$$\Gamma_S(\mathbb{P}_{\text{sampling}}) = \frac{2(n+1)}{n} H_n = O(\log n).$$

Step 3: The case $s \geq 2$. Now let $s \geq 2$ be fixed. In (12), we bound the inner sum via

$$(q+1)_r \geq (q+1)^r, \qquad (n-s-q+1)_{s-r} \geq (n-s-q+1)^{s-r},$$

which yields

$$\sum_{q=0}^{n-s} \frac{1}{(q+1)_r \, (n-s-q+1)_{s-r}} \leq \sum_{q=0}^{n-s} \frac{1}{(q+1)^r \, (n-s-q+1)^{s-r}}.$$

We distinguish two cases.

Case 1: $r = 0$ or $r = s$. If $r = 0$, then

$$\sum_{q=0}^{n-s} \frac{1}{(n-s-q+1)^s} = \sum_{j=1}^{n-s+1} \frac{1}{j^s} = O(1),$$

since for every $s \geq 2$ and every $j \geq 1$ we have $j^{-s} \leq j^{-2}$, so the sum is bounded by $\sum_{j \geq 1} j^{-2} = \pi^2/6$, a constant that holds uniformly in both n and s. The same argument applies to $r = s$, giving

$$\sum_{q=0}^{n-s} \frac{1}{(q+1)^s} = O(1).$$

Case 2: $1 \leq r \leq s-1$. In this range, both exponents are at least 1, and therefore

$$\frac{1}{(q+1)^r \, (n-s-q+1)^{s-r}} \leq \frac{1}{(q+1)(n-s-q+1)}.$$

Writing $A := n - s$, we get

$$\begin{aligned} \sum_{q=0}^{A} \frac{1}{(q+1)(A-q+1)} &= \frac{1}{A+2} \sum_{q=0}^{A} \left(\frac{1}{q+1} + \frac{1}{A-q+1} \right) \\ &= \frac{2H_{A+1}}{A+2} = O\left(\frac{\log n}{n} \right). \end{aligned}$$

Here we used

$$\frac{1}{(q+1)(A-q+1)} = \frac{1}{A+2} \left(\frac{1}{q+1} + \frac{1}{A-q+1} \right).$$

Combining the two cases, we find that for every fixed $r \in \{0, \ldots, s\}$,

$$\sum_{q=0}^{n-s} \frac{1}{(q+1)_r \, (n-s-q+1)_{s-r}} = \begin{cases} O(1), & r = 0 \text{ or } r = s, \\ O\left(\dfrac{\log n}{n} \right), & 1 \leq r \leq s-1. \end{cases}$$

Thus, the dominant contributions come from the boundary cases $r = 0$ and $r = s$, while all interior contributions are asymptotically smaller. Therefore

$$\sum_{r=0}^{s} \binom{s}{r} \sum_{q=0}^{n-s} \frac{1}{(q+1)_r \, (n-s-q+1)_{s-r}} = O(1).$$

By (12),

$$\Gamma_S(\mathbb{P}_{\text{sampling}}) = O\left(\frac{(n+1)(n-s+1)_s}{(n-s+1)^2} \right).$$

Finally, for fixed s,

$$\frac{(n+1)(n-s+1)_s}{(n-s+1)^2} = O(n^{s-1}),$$

and therefore

$$\Gamma_S(\mathbb{P}_{\text{sampling}}) = O(n^{s-1}).$$

□

Proof of Theorem 3.3. By Theorem A.3,

$$\mathbb{V}[\hat{\phi}_S^{\text{MSR}}(\nu; \mathcal{T})] \leq \frac{\|\nu\|_\infty^2}{|\mathcal{T}|} \Gamma_S(\mathbb{P}_{\text{sampling}}).$$

Under uniform sampling over sizes and SII weights, Corollary A.5 yields

$$\Gamma_S(\mathbb{P}_{\text{sampling}}) = \begin{cases} O(\log n), & |S| = 1, \\ O(n^{|S|-1}), & |S| \geq 2. \end{cases}$$

Substituting this into the general bound gives

$$\mathbb{V}[\hat{\phi}_S^{\text{MSR}}(\nu; \mathcal{T})] \leq \begin{cases} \mathcal{O}\left(\frac{\|\nu\|_\infty^2 \log n}{|\mathcal{T}|} \right), & |S| = 1, \\ \mathcal{O}\left(\frac{\|\nu\|_\infty^2 n^{|S|-1}}{|\mathcal{T}|} \right), & |S| \geq 2, \end{cases}$$

where the constants hidden in the $\mathcal{O}(\cdot)$ notation depend only on $|S|$. This proves Theorem 3.3. □

Lemma A.6 (Lower bound on Γ_S for SII under uniform over sizes)**.** *Suppose $\mathbb{P}_{\text{sampling}}$ is uniform over sizes and $p_t^s(n)$ are the SII weights. Then, for fixed interaction order s,*

$$\Gamma_S(\mathbb{P}_{\text{sampling}}) = \begin{cases} \Theta(\log n), & s = 1, \\ \Theta(n^{s-1}), & s \geq 2. \end{cases}$$

Proof. Recall from the proof of Corollary A.5 that

$$\Gamma_S(\mathbb{P}_{\text{sampling}}) = \frac{(n+1)(n-s+1)_s}{(n-s+1)^2} \sum_{r=0}^{s} \binom{s}{r} \sum_{q=0}^{n-s} \frac{1}{(q+1)_r \, (n-s-q+1)_{s-r}}.$$

Since every term of the double sum is non-negative, we may lower-bound it by retaining only the boundary term $r = 0$:

$$\Gamma_S(\mathbb{P}_{\text{sampling}}) \geq \frac{(n+1)(n-s+1)_s}{(n-s+1)^2} \sum_{q=0}^{n-s} \frac{1}{(n-s-q+1)_s} = \frac{(n+1)(n-s+1)_s}{(n-s+1)^2} \sum_{j=1}^{n-s+1} \frac{1}{(j)_s}.$$

Case $s = 1$. Then $(j)_1 = j$ and

$$\sum_{j=1}^{n} \frac{1}{j} = H_n = \Theta(\log n),$$

so $\Gamma_S(\mathbb{P}_{\text{sampling}}) \geq \frac{n+1}{n} H_n = \Theta(\log n)$.

Case $s \geq 2$**.** The sum $\sum_{j=1}^{n-s+1} 1/(j)_s$ converges to a positive constant as $n \to \infty$ (its $j = 1$ term alone equals $1/s!$), so it is $\Theta(1)$. Combined with $(n-s+1)_s = \Theta(n^s)$ and $(n-s+1)^2 = \Theta(n^2)$ for fixed s, we obtain

$$\Gamma_S(\mathbb{P}_{\text{sampling}}) = \Theta\left(\frac{n \cdot n^s}{n^2}\right) = \Theta(n^{s-1}).$$

□

Corollary A.7 (Matching lower bound on MSR variance for SII uniform over sizes)**.** *Suppose coalitions are sampled i.i.d. according to:*

$$\mathbb{P}_{\textit{sampling}}(T) = \frac{1}{(n+1)\binom{n}{|T|}}.$$

Then there exists a value function ν such that

$$\mathbb{V}\left[\hat{\phi}_S^{\text{MSR}}(\nu;\mathcal{T})\right] = \begin{cases} \Theta\left(\dfrac{\|\nu\|_\infty^2 \log n}{|\mathcal{T}|}\right), & |S| = 1, \\ \Theta\left(\dfrac{\|\nu\|_\infty^2 \, n^{|S|-1}}{|\mathcal{T}|}\right), & |S| \geq 2. \end{cases}$$

Proof. By Theorem A.3,

$$\mathbb{V}\left[\hat{\phi}_S^{\text{MSR}}(\nu;\mathcal{T})\right] = \frac{1}{|\mathcal{T}|}\left(\sum_{T \in 2^N} \nu(T)^2 \, \frac{\left(p^s_{|T|-|S\cap T|}(n)\right)^2}{\mathbb{P}_{\text{sampling}}(T)} \; - \; \phi_S^p(\nu)^2\right).$$

Take $\nu \equiv 1$, so $\phi_S^p(\nu) = 0$ for any $|S| \geq 1$ by the dummy axiom. The variance then equals $\Gamma_S(\mathbb{P}_{\text{sampling}})/|\mathcal{T}| = \|\nu\|_\infty^2 \, \Gamma_S(\mathbb{P}_{\text{sampling}})/|\mathcal{T}|$, and the claim follows from the Lemma A.6 above. □

Corollary A.8 (Variance of MSR under uniform over sizes for BII)**.** *Suppose $\mathbb{P}_{\text{sampling}}$ is given by uniform over sizes,*

$$\mathbb{P}_{\text{sampling}}(T) = \frac{1}{(n+1)\binom{n}{|T|}},$$

and let $p_t^s(n)$ denote the Banzhaf Interaction Index weights,

$$p_t^s(n) = \frac{1}{2^{n-s}}.$$

Then

$$\Gamma_S(\mathbb{P}_{\text{sampling}}) = \frac{n+1}{4^{n-s}}\binom{2n}{n}.$$

In particular,

$$\Gamma_S(\mathbb{P}_{\text{sampling}}) = \Theta(4^s\sqrt{n}),$$

and therefore, by Theorem A.3,

$$\mathbb{V}[\hat{\phi}_S^{\text{MSR}}(\nu;\mathcal{T})] \leq \frac{\|\nu\|_\infty^2}{|\mathcal{T}|}\,\Gamma_S(\mathbb{P}_{\text{sampling}}) = O\left(\frac{\|\nu\|_\infty^2 4^s\sqrt{n}}{|\mathcal{T}|}\right).$$

In particular, for fixed interaction order s,

$$\mathbb{V}[\hat{\phi}_S^{\text{MSR}}(\nu;\mathcal{T})] = O\left(\frac{\|\nu\|_\infty^2\sqrt{n}}{|\mathcal{T}|}\right).$$

Proof. By Theorem A.3,

$$\mathbb{V}[\hat{\phi}_S^{\text{MSR}}(\nu;\mathcal{T})] \leq \frac{\|\nu\|_\infty^2}{|\mathcal{T}|}\,\Gamma_S(\mathbb{P}_{\text{sampling}}),$$

where

$$\Gamma_S(\mathbb{P}_{\text{sampling}}) := \sum_{T \subseteq N} \frac{\left(p^s_{|T|-|S\cap T|}(n)\right)^2}{\mathbb{P}_{\text{sampling}}(T)}.$$

For the Banzhaf Interaction Index, the weights are constant in the coalition size, namely

$$p_t^s(n) = \frac{1}{2^{n-s}}.$$

Hence

$$\begin{aligned}\Gamma_S(\mathbb{P}_{\text{sampling}}) &= \sum_{T \subseteq N} \frac{1}{4^{n-s}} \frac{1}{\mathbb{P}_{\text{sampling}}(T)} \\ &= \frac{1}{4^{n-s}} \sum_{T \subseteq N} (n+1) \binom{n}{|T|}.\end{aligned}$$

We now group coalitions $T \subseteq N$ by their size $t := |T|$. For each $t \in \{0, \dots, n\}$, there are exactly $\binom{n}{t}$ coalitions of size t. Therefore

$$\begin{aligned}\sum_{T \subseteq N} \binom{n}{|T|} &= \sum_{t=0}^{n} \binom{n}{t} \binom{n}{t} \\ &= \sum_{t=0}^{n} \binom{n}{t}^2.\end{aligned}$$

Using Vandermonde's identity,

$$\sum_{t=0}^{n} \binom{n}{t}^2 = \binom{2n}{n}.$$

Substituting this into the previous display yields the exact formula

$$\Gamma_S(\mathbb{P}_{\text{sampling}}) = \frac{n+1}{4^{n-s}} \binom{2n}{n}.$$

To obtain the asymptotic form, we use the central binomial coefficient estimate

$$\binom{2n}{n} = \Theta\left(\frac{4^n}{\sqrt{n}}\right).$$

Hence

$$\Gamma_S(\mathbb{P}_{\text{sampling}}) = \frac{n+1}{4^{n-s}} \binom{2n}{n} = \Theta\left(\frac{n+1}{4^{n-s}} \cdot \frac{4^n}{\sqrt{n}}\right) = \Theta(4^s \sqrt{n}).$$

Finally, substituting this into the general variance bound from Theorem A.3 gives

$$\mathbb{V}[\hat{\phi}_S^{\text{MSR}}(\nu; \mathcal{T})] \leq \frac{\|\nu\|_\infty^2}{|\mathcal{T}|} \Gamma_S(\mathbb{P}_{\text{sampling}}) = O\left(\frac{\|\nu\|_\infty^2 4^s \sqrt{n}}{|\mathcal{T}|}\right).$$

For fixed interaction order s, the factor 4^s is absorbed into the constant, yielding

$$\mathbb{V}[\hat{\phi}_S^{\text{MSR}}(\nu; \mathcal{T})] = O\left(\frac{\|\nu\|_\infty^2 \sqrt{n}}{|\mathcal{T}|}\right).$$

This proves the claim. □

A.3 Closed Forms for Proposition 3.2

A.3.1 Shapley Value and Shapley Interaction Index

Proposition A.9. *The Shapley Value and the Shapley Interaction Index yield the closed form*

$$\lambda(|L_j|, |R_j|, |L_j \cap S|, |S|) = (-1)^u \frac{1}{(a+b+1)\binom{a+b}{a}},$$

where

$$u := |L_j \cap S|, \qquad a := |R_j| - |R_j \cap S|, \qquad b := |L_j| - |L_j \cap S|.$$

Proof. This is exactly Proposition 1 in Zern et al. [65], specialized to our notation. □

A.3.2 Banzhaf Value and Banzhaf Interaction Index

Proposition A.10. *The Banzhaf Value and the Banzhaf Interaction Index yield the closed form*

$$\lambda(\ell, r, u, s) = (-1)^u \frac{1}{2^{\ell+r-s}}.$$

Proof. Starting from Proposition 3.2 and using the Möbius weights for the Banzhaf interaction index from Table 1, we obtain

$$\lambda(\ell, r, u, s) = \sum_{i=0}^{\ell-u} (-1)^{i+u} \binom{\ell-u}{i} \frac{1}{2^{i+u+r-s}}.$$

Define

$$k_0 := u + r - s, \qquad m := \ell - u.$$

Then

$$\begin{aligned}
\lambda(\ell, r, u, s) &= \sum_{i=0}^{m} (-1)^{i+u} \binom{m}{i} \frac{1}{2^{i+k_0}} \\
&= (-1)^u \frac{1}{2^{k_0}} \sum_{i=0}^{m} (-1)^i \binom{m}{i} \frac{1}{2^i} \\
&= (-1)^u \frac{1}{2^{k_0}} \sum_{i=0}^{m} \binom{m}{i} \left(-\frac{1}{2}\right)^i \\
&= (-1)^u \frac{1}{2^{k_0}} \sum_{i=0}^{m} \binom{m}{i} \left(-\frac{1}{2}\right)^i 1^{m-i}.
\end{aligned}$$

By the binomial theorem,

$$\sum_{i=0}^{m} \binom{m}{i} \left(-\frac{1}{2}\right)^i 1^{m-i} = \left(1 - \frac{1}{2}\right)^m = \left(\frac{1}{2}\right)^m.$$

Hence

$$\lambda(\ell, r, u, s) = (-1)^u \frac{1}{2^{k_0}} \left(\frac{1}{2}\right)^m = (-1)^u \frac{1}{2^{k_0+m}}.$$

Substituting the definitions of k_0 and m gives

$$\lambda(\ell, r, u, s) = (-1)^u \frac{1}{2^{\ell+r-s}}.$$

□

A.3.3 Chaining Value and Chaining Interaction Index

Proposition A.11. *The Chaining Value and the Chaining Interaction Index yield the closed form*

$$\lambda(\ell, r, u, s) = s(-1)^u B(u + r, \ell - u + 1),$$

where

$$B(z_1, z_2) := \int_0^1 x^{z_1 - 1}(1 - x)^{z_2 - 1}\, dx$$

denotes the Beta function.

Proof. For the Chaining Interaction Index, the Möbius weights satisfy $q_t^s(n) = \frac{s}{t}$ [15]. Therefore, Proposition 3.2 yields

$$\lambda(\ell, r, u, s) = \sum_{i=0}^{\ell-u} (-1)^{i+u} \binom{\ell - u}{i} \frac{s}{i + u + r}.$$

Factor out the constant terms:

$$\lambda(\ell, r, u, s) = (-1)^u s \sum_{i=0}^{\ell-u} (-1)^i \binom{\ell - u}{i} \frac{1}{i + u + r}.$$

Now define

$$k_0 := u + r, \qquad m := \ell - u,$$

and use the identity

$$\frac{1}{i + k_0} = \int_0^1 x^{i + k_0 - 1}\, dx.$$

Then

$$\begin{aligned}
\lambda(\ell, r, u, s) &= (-1)^u s \sum_{i=0}^{m} (-1)^i \binom{m}{i} \int_0^1 x^{i + k_0 - 1}\, dx \\
&= (-1)^u s \int_0^1 \sum_{i=0}^{m} (-1)^i \binom{m}{i} x^i x^{k_0 - 1}\, dx \\
&= (-1)^u s \int_0^1 \left(\sum_{i=0}^{m} \binom{m}{i} (-x)^i 1^{m-i} \right) x^{k_0 - 1}\, dx \\
&= (-1)^u s \int_0^1 (1 - x)^m x^{k_0 - 1}\, dx \\
&= (-1)^u s\, B(k_0, m + 1).
\end{aligned}$$

Substituting back $k_0 = u + r$ and $m = \ell - u$ gives

$$\lambda(\ell, r, u, s) = (-1)^u s\, B(u + r, \ell - u + 1).$$

□

A.3.4 Faithful Banzhaf Interaction Index (FBII)

Proposition A.12. *For the Faithful Banzhaf Interaction Index (FBII), Proposition 3.2 yields the closed form*

$$\phi_S^{\text{FBII}}(\hat{\nu}_{\text{tree}}) = \sum_{\substack{j \in \mathcal{L} \\ S \subseteq L_j \cup R_j}} c_j \lambda\big(|L_j|, |R_j|, |S \cap L_j|, |S|\big),$$

where

$$\begin{aligned}
\lambda(\ell, r, u, s) = (-1)^u \mathbb{1}[r = s - u] \\
+ \sum_{i=\max(0, k-r-u+1)}^{\ell-u} (-1)^{u+i+k-s} 2^{-(r+i+u-s)} \binom{\ell - u}{i} \binom{r + i + u - s - 1}{k - s}.
\end{aligned}$$

and k denotes the maximal interaction order to be computed.

Proof. We begin with the Möbius representation of FBII [58]:

$$\phi_S^{\mathrm{FBII}}(\nu) = m_S(\nu) + (-1)^{k-s} \sum_{\substack{T \supseteq S \\ |T|>k}} \left(\frac{1}{2}\right)^{|T|-s} \binom{|T|-s-1}{k-s} m_T(\nu).$$

Applying Lemma A.1 to the tree representation yields

$$\begin{aligned}\phi_S^{\mathrm{FBII}}(\hat{\nu}_{\text{tree}}) &= \sum_{j \in \mathcal{L}} c_j \, \phi_S^{\mathrm{FBII}} \left(\sum_{T \subseteq L_j} (-1)^{|T|} \mathbb{1}_{[T \cup R_j, N]} \right) \\ &= \sum_{j \in \mathcal{L}} c_j \sum_{T \subseteq L_j} (-1)^{|T|} \phi_S^{\mathrm{FBII}}(\mathbb{1}_{[T \cup R_j, N]}).\end{aligned}$$

We now analyze a fixed leaf j. For notational brevity, write

$$r := |R_j|, \qquad \ell := |L_j|, \qquad u := |S \cap L_j|, \qquad s := |S|.$$

Using the Möbius representation above, we split the expression into two parts:

$$\begin{aligned}&\sum_{T \subseteq L_j} (-1)^{|T|} \phi_S^{\mathrm{FBII}}(\mathbb{1}_{[T \cup R_j, N]}) \\ &\quad = \sum_{T \subseteq L_j} (-1)^{|T|} m_S(\mathbb{1}_{[R_j \cup T, N]}) \\ &\qquad + (-1)^{k-s} \sum_{T \subseteq L_j} (-1)^{|T|} \sum_{\substack{V \supseteq S \\ |V|>k}} \left(\frac{1}{2}\right)^{|V|-s} \binom{|V|-s-1}{k-s} m_V(\mathbb{1}_{[R_j \cup T, N]}).\end{aligned}$$

Step 1: The Möbius term. Consider

$$\sum_{T \subseteq L_j} (-1)^{|T|} m_S(\mathbb{1}_{[R_j \cup T, N]}).$$

By Lemma A.2 with $B = N$, we have

$$m_S(\mathbb{1}_{[R_j \cup T, N]}) = 1 \quad \Longleftrightarrow \quad R_j \cup T = S,$$

and otherwise it is zero. Moreover, only interactions with $S \subseteq L_j \cup R_j$ can contribute. Under this condition, we must have $R_j \subseteq S$, hence

$$\begin{aligned}\sum_{T \subseteq L_j} (-1)^{|T|} m_S(\mathbb{1}_{[R_j \cup T, N]}) &= \sum_{T \subseteq L_j} (-1)^{|T|} \mathbb{1}[R_j \cup T = S] \\ &= (-1)^u \mathbb{1}[R_j \subseteq S].\end{aligned}$$

Step 2: The faithful tail term. Now consider

$$\begin{aligned}&(-1)^{k-s} \sum_{T \subseteq L_j} (-1)^{|T|} \sum_{\substack{V \supseteq S \\ |V|>k}} \left(\frac{1}{2}\right)^{|V|-s} \binom{|V|-s-1}{k-s} m_V(\mathbb{1}_{[R_j \cup T, N]}) \\ &\quad = (-1)^{k-s} \sum_{\substack{V \supseteq S \\ |V|>k}} \left(\frac{1}{2}\right)^{|V|-s} \binom{|V|-s-1}{k-s} \sum_{T \subseteq L_j} (-1)^{|T|} m_V(\mathbb{1}_{[R_j \cup T, N]}).\end{aligned}$$

Again by Lemma A.2 with $B = N$,

$$m_V(\mathbb{1}_{[R_j \cup T, N]}) = 1 \quad \Longleftrightarrow \quad R_j \cup T = V.$$

Since the outer sum only contains V with $|V| > k$, only subsets T satisfying $|R_j \cup T| > k$ contribute, i.e.

$$|T| > k - r.$$

Furthermore, we must have $S \subseteq R_j \cup T$, which is equivalent to $S \cap L_j \subseteq T$. Therefore, the tail term becomes

$$\sum_{\substack{S\cap L_j \subseteq T \subseteq L_j \\ |T|>k-r}} (-1)^{|T|+k-s} \left(\frac{1}{2}\right)^{r+|T|-s} \binom{r+|T|-s-1}{k-s}.$$

Write

$$S_L := L_j \cap S, \qquad S_R \subseteq L_j \setminus S_L, \qquad T = S_L \cup S_R.$$

Then $|T| = |S_L| + |S_R|$, and summing over all possible S_R gives

$$\sum_{\substack{S_R \subseteq L_j \setminus S_L \\ |S_R|>k-r-u}} (-1)^{|S_R|+u+k-s} \left(\frac{1}{2}\right)^{r+|S_R|+u-s} \binom{r+|S_R|+u-s-1}{k-s}.$$

Grouping terms by $i := |S_R|$, there are $\binom{\ell-u}{i}$ such subsets. Hence, the tail term simplifies to

$$\sum_{i=k-r-u+1}^{\ell-u} (-1)^{u+i+k-s} \left(\frac{1}{2}\right)^{r+i+u-s} \binom{r+i+u-s-1}{k-s} \binom{\ell-u}{i}.$$

Step 3: Combine both parts. When $k-r-u+1 \leq 0$, the constraint $|T| > k-r$ is satisfied by every $S \cap L_j \subseteq T \subseteq L_j$, so the lower summation index collapses to 0; as such, the lower index can be written as $\max(0,\, k-r-u+1)$ as in the proposition statement. Combining the Möbius term and the faithful tail term, summing over all leaves, and incorporating the condition $S \subseteq L_j \cup R_j$, we obtain

$$\phi_S^{\mathrm{FBII}}(\hat{\nu}_{\text{tree}}) = \sum_{\substack{j\in\mathcal{L} \\ S\subseteq L_j \cup R_j}} c_j \cdot \lambda\big(|L_j|, |R_j|, |S\cap L_j|, |S|\big),$$

where

$$\begin{aligned}\lambda(|L_j|, |R_j|, |S\cap L_j|, |S|) =& (-1)^{|S\cap L_j|} \mathbb{1}[R_j \subseteq S] \\ &+ \sum_{i=\max(0,\, k-r-u+1)}^{\ell-u} (-1)^{u+i+k-s} \left(\frac{1}{2}\right)^{r+i+u-s} \binom{r+i+u-s-1}{k-s} \binom{\ell-u}{i}\end{aligned}$$

yields the claimed formula. □

A.3.5 Faithful Shapley Interaction Index (FSII)

Proposition A.13. *For the Faithful Shapley Interaction Index (FSII), Proposition 3.2 yields the closed form*

$$\phi_S^{\mathrm{FSII}}(\hat{\nu}_{\text{tree}}) = \sum_{\substack{j\in\mathcal{L} \\ S\subseteq L_j \cup R_j}} c_j \lambda\big(|L_j|, |R_j|, |S\cap L_j|, |S|\big),$$

where $u := |S \cap L_j|$ *in*

$$\begin{aligned}\lambda(\ell, r, u, s) =& (-1)^u \mathbb{1}[r = s-u] \\ &+ \sum_{i=\max(0, k-r-u+1)}^{\ell-u} (-1)^{u+i+k-s} \frac{s}{k+s} \binom{k}{s} \binom{\ell-u}{i} \frac{\binom{r+i+u-1}{k}}{\binom{r+u+i+k-1}{k+s}}\end{aligned}$$

Proof. We use the Möbius representation of FSII [58]:

$$\phi_S^{\mathrm{FSII}}(\nu) = m_S(\nu) + (-1)^{k-s} \frac{s}{k+s} \binom{k}{s} \sum_{\substack{T\supseteq S \\ |T|>k}} \frac{\binom{|T|-1}{k}}{\binom{|T|+k-1}{k+s}} m_T(\nu).$$

Applying Lemma A.1 to the tree representation gives

$$\phi_S^{\mathrm{FSII}}(\hat{\nu}_{\text{tree}}) = \sum_{j \in \mathcal{L}} c_j \sum_{T \subseteq L_j} (-1)^{|T|} \phi_S^{\mathrm{FSII}}(\mathbb{1}_{[T \cup R_j, N]}).$$

Fix a leaf j, and write

$$r := |R_j|, \qquad \ell := |L_j|, \qquad u := |S \cap L_j|, \qquad s := |S|.$$

The first term is identical to the corresponding term in the proof of Proposition A.12, since it only depends on the Möbius term m_S. Hence it contributes

$$(-1)^u \mathbb{1}[R_j \subseteq S].$$

For the second term, the same Möbius support argument as in the FBII proof shows that only subsets $T \subseteq L_j$ with $|T| > k - r$ contribute, and the outer sum reduces to the case $V = R_j \cup T$. Again, we must have $S \subseteq R_j \cup T$, which is equivalent to $S \cap L_j \subseteq T$. Therefore, the faithful tail term equals

$$(-1)^{k-s} \frac{s}{k+s} \binom{k}{s} \sum_{\substack{S \cap L_j \subseteq T \subseteq L_j \\ |T| > k-r}} (-1)^{|T|} \frac{\binom{r+|T|-1}{k}}{\binom{r+|T|+k-1}{k+s}}.$$

Now write

$$S_L := L_j \cap S, \qquad S_R \subseteq L_j \setminus S_L, \qquad T = S_L \cup S_R.$$

Then $|T| = |S_L| + |S_R|$, and grouping subsets by $i := |S_R|$ yields

$$(-1)^{k-s} \frac{s}{k+s} \binom{k}{s} \sum_{i=k-r-u+1}^{\ell-u} (-1)^{u+i} \binom{\ell-u}{i} \frac{\binom{r+i+u-1}{k}}{\binom{r+u+i+k-1}{k+s}}.$$

When $k - r - u + 1 \leq 0$, the constraint $|T| > k - r$ is satisfied by every $S \cap L_j \subseteq T \subseteq L_j$, so the lower summation index collapses to 0; as such, the lower index can be written as $\max(0,\ k-r-u+1)$ as in the proposition statement. Combining this faithful tail term with the Möbius term and summing over all leaves gives

$$\phi_S^{\mathrm{FSII}}(\hat{\nu}_{\text{tree}}) = \sum_{\substack{j \in \mathcal{L} \\ S \subseteq L_j \cup R_j}} c_j \cdot \lambda\big(|L_j|, |R_j|, |S \cap L_j|, |S|\big)$$

where

$$\begin{aligned}\lambda(|L_j|, |R_j|, |S \cap L_j|, |S|) =& (-1)^{|S \cap L_j|} \mathbb{1}[R_j \subseteq S] \\ &+ \sum_{i=\max(0,k-r-u+1)}^{\ell-u} (-1)^{u+i+k-s} \frac{s}{k+s} \binom{k}{s} \binom{\ell-u}{i} \frac{\binom{r+i+u-1}{k}}{\binom{r+u+i+k-1}{k+s}}\end{aligned}$$

which proves the claim. □

Proposition A.14. *For the Shapley Taylor Interaction Index (STII), Proposition 3.2 yields the closed form*

$$\phi_S^{\mathrm{STII}}(\hat{\nu}_{\text{tree}}) = \sum_{\substack{j \in \mathcal{L} \\ S \subseteq L_j \cup R_j}} c_j \lambda\big(|L_j|, |R_j|, |S \cap L_j|, |S|\big),$$

where $u := |S \cap L_j|$ *in*

$$\begin{aligned}\lambda(|L_j|, |R_j|, |S \cap L_j|, |S|) =& \mathbb{1}_{s<k} \mathbb{1}_{R_j \subseteq S} (-1)^u \\ &+ \mathbb{1}_{s=k} (-1)^u \, s \, B(u + r_j - s + 1, \ell_j - u + s)\end{aligned}$$

and $B(a, b)$ *is the Beta function.*

Proof. Firstly, we observe that the STII is defined to match the Möbius value for $s < k$, as such, Lemma A.2 applies for that case, yielding $m_S(\mathbb{1}_{[R_j, N \setminus L_j]}) = (-1)^u \mathbb{1}_{R_j \subseteq S}$ for $S \subseteq L_j \cup R_j$. We next show that for $s = k$, a closed form is obtainable. We begin with using Proposition 3.2 and obtain

$$\lambda(l, r, u, s) = \sum_{i=0}^{l-u} (-1)^{i+u} \binom{l-u}{i} \frac{1}{\binom{i+u+r}{s}}$$

by directly using the STII Möbius weights $q_t^s(n) = \frac{1}{\binom{t}{s}}$ [57], where $i+u+r \geq s$ since $S \subseteq L_j \cup R_j$. Next, we begin writing out the latter fraction and obtain

$$\begin{aligned}
\lambda(l, r, u, s) &= \sum_{i=0}^{l-u} (-1)^{i+u} \binom{l-u}{i} \frac{s!(i+u+r-s)!}{(i+u+r)!} \\
&= \sum_{i=0}^{n} (-1)^{i+u} \binom{n}{i} \frac{s!(i+m-s)!}{(i+m)!}
\end{aligned}$$

where we rewrite $n := l - u$ and $m := u + r$. By using the formulation of the Beta function $B(a, b) = \frac{(a-1)!(b-1)!}{(a+b-1)!}$, we obtain

$$\begin{aligned}
\lambda(l, r, u, s) &= \sum_{i=0}^{n} (-1)^{i+u} \binom{n}{i} sB(i+m-s+1, s) \\
&= (-1)^u s \sum_{i=0}^{n} (-1)^i \binom{n}{i} B(i+m-s+1, s)
\end{aligned}$$

Now we use the different representation $B(a, b) = \int_0^1 t^{a-1}(1-t)^{b-1} dt$ to remove the sum by

$$\begin{aligned}
\lambda(l, r, u, s) &= (-1)^u s \sum_{i=0}^{n} (-1)^i \binom{n}{i} B(i+m-s+1, s) \\
&= (-1)^u s \sum_{i=0}^{n} (-1)^i \binom{n}{i} \int_0^1 (1-t)^{s-1} t^{i+m-s} dt \\
&= (-1)^u s \int_0^1 (1-t)^{s-1} \Big[\sum_{i=0}^{n} (-1)^i \binom{n}{i} t^i\Big] t^{m-s} dt \\
&= (-1)^u s \int_0^1 (1-t)^{s-1} (1-t)^n t^{m-s} dt \\
&= (-1)^u s \int_0^1 (1-t)^{n+s-1} t^{m-s} dt \\
&= (-1)^u s B(m-s+1, n+s)
\end{aligned}$$

Then the final formula is obtained by substituting $m = u + r$ and $n = l - u$, which concludes the proof. □

Table 2: Interventional Tree-extraction algorithms. Parenthesized check marks indicate pairwise-only.

Algorithm	Values			Interactions					Faithful	
	SV	BV	CV	SII	BII	CII	STII	k-SII	FSII	FBII
Interventional extraction										
TreeSHAP (Lundberg et al. [37])	✓	×	×	×	×	×	×	×	×	×
TreeSHAP (Zern et al. [65])	✓	×	×	✓	×	×	×	×	×	×
Woodelf [45]	✓	✓	×	(✓)	(✓)	×	×	×	×	×
Ours (Algorithm 2)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

B Generalization of Interventional TreeSHAP

Central to ProxySHAP is the ability to efficiently extract exact cardinal-probabilistic interaction indices from tree-based proxies. Building on interventional TreeSHAP [65], we extract the indices of $\hat{\nu}$ for a target set $\mathcal{S} \subseteq 2^N$ in $\mathcal{O}(n_{\text{nodes}}|\mathcal{S}|)$ time. Table 2 summarizes the supported indices and compares our generalized interventional TreeSHAP with existing tree-based extraction algorithms. Note that $k-$SII can be computed by first extracting SII and then using the aggregation rule of [4].

Algorithm 2 `Tree Interaction Extraction`

Require: Tree-based model $\hat{\nu}_{\text{tree}}$, interaction $S \subseteq N$
Ensure: Extracted interaction ϕ_S^p

$\phi_S^p \leftarrow 0$ ▷ initialize
for each leaf $j \in \mathcal{L}$ **do**
  $(L_j, R_j) \leftarrow \textsc{Path}(j)$
  $c_j \leftarrow \textsc{LeafValue}(j)$
  if $S \subseteq L_j \cup R_j$ **then**
    $\phi_S^p \leftarrow \phi_S^p + \lambda(|L_j|, |R_j|, |S \cap L_j|, |S|)\, c_j$
return ϕ_S^p

First, consider a single decision tree with inputs as binary-encoded subsets $T \subseteq N$. Each leaf $j \in \mathcal{L}$ corresponds to a path of node splits, where each split on feature $i \in N$ determines whether $i \in T$ (right branch) or $i \notin T$ (left branch). An input T reaches leaf j if and only if along the path to j: (i) $T \supseteq R_j$, meaning it contains all features that split to the right (R_j), and (ii) $T \subseteq N \setminus L_j$, meaning it contains none of the features that split to the left (L_j). The tree-based proxy is then given by the piecewise constant leaf predictions $c_j \in \mathbb{R}$ as

$$\hat{\nu}_{\text{tree}}(T) := \sum_{j \in \mathcal{L}} c_j \cdot \mathbb{1}[R_j \subseteq T \subseteq N \setminus L_j]. \tag{13}$$

Zern et al. [65] exploited linearity in Eq. (8) and computed the Shapley interactions for each $\mathbb{1}[R_j \subseteq T \subseteq N \setminus L_j]$, based on the Möbius representation of $\mathbb{1}[R_j \subseteq T \subseteq N \setminus L_j]$. In a similar spirit, we can compute the general cardinal-probabilistic interaction indices for $\mathbb{1}[R_j \subseteq T \subseteq N \setminus L_j]$ via the Möbius representation, and then use linearity to obtain the interaction indices for $\hat{\nu}_{\text{tree}}$. This yields Proposition 3.2, which gives a closed-form solution for any cardinal-probabilistic interaction index of $\mathbb{1}[R_j \subseteq T \subseteq N \setminus L_j]$. Then, by using the closed forms of the Möbius weights q_t^s for specific cardinal-probabilistic interaction indices, we can obtain closed-form solutions for the corresponding interventional TreeSHAP interactions, as shown in Propositions A.10, A.11, A.12, A.13 and A.14. The overall procedure is summarized in Algorithm 2. Note that, due to Corollary 1 in Zern et al. [65], Proposition 3.2 also enables exact computation of interactions for piecewise linear regression trees.

Given an interaction S, the algorithm iterates over all leaves j and extracts the corresponding path information (L_j, R_j) and leaf value c_j. Then, it updates the interaction value ϕ_S^p by adding the contribution from leaf j, which is computed using the closed-form solution $\lambda(|L_j|, |R_j|, |S \cap L_j|, |S|)$ from Proposition 3.2 multiplied by the leaf value c_j. Finally, the algorithm returns the extracted interaction ϕ_S^p. The lookups PATH and LEAFVALUE can be pre-computed for all leaves and stored in a hash map for efficient access, resulting in an overall time complexity of $\mathcal{O}(n_{\text{nodes}})$ for extracting a single interaction S.

C Experimental Details

C.1 Datasets

Table 3: Summary of datasets used in the experiments. "Players" denotes the number of features or components considered in the corresponding game.

Dataset	Task	Model(s)	Players	Source	License
Vision and language datasets [41]					
Language (IMDB)	Classification	DistilBERT	14	shapiq / HuggingFace	Public Domain
ResNet18-SP	Image classification	ResNet18	14	shapiq	ImageNet Terms of Access
ViT4by4	Image classification	ViT-B	16	shapiq	ImageNet Terms of Access
DINOv2-16	Image classification	DINOv2-B + linear head	16	ImageNet-1k / HuggingFace	ImageNet Terms of Access
DINOv2-20	Image classification	DINOv2-B + linear head	20	ImageNet-1k / HuggingFace	ImageNet Terms of Access
DINOv2-25	Image classification	DINOv2-B + linear head	25	ImageNet-1k / HuggingFace	ImageNet Terms of Access
Tabular datasets [41, 55]					
Bike	Regression	TabPFN	12	OpenML (42712)	Public Domain
Forest Fires	Regression	TabPFN	13	UCI	CC-BY 4.0
AdultCensus	Classification	TabPFN	14	OpenML (1590)	CC-BY 4.0
Estate	Regression	TabPFN	15	UCI	CC-BY 4.0
Hepatitis	Classification	TabPFN	19	UCI (46)	CC-BY 4.0
Thyroid	Classification	TabPFN	21	UCI (102)	CC-BY 4.0
Mushroom	Classification	TabPFN	22	UCI (73)	CC-BY 4.0
Cancer	Classification	XGBoost	30	scikit-learn	CC-BY 4.0
Ionosphere	Classification	LightGBM	33	UCI (52)	CC-BY 4.0
Soybean	Classification	LightGBM	35	UCI (90)	CC-BY 4.0
CG60	Regression (synthetic)	XGBoost	60	SHAP	MIT
IL60	Regression (synthetic)	XGBoost	60	SHAP	MIT
NHANES	Survival	XGBoost	79	SHAP	Public Domain
Crime	Regression	XGBoost	101	SHAP	CC-BY 4.0
TabArena datasets [14]					
Wine Quality	Regression	TabPFN	12	OpenML (46964)	CC-BY 4.0
HR Analytics	Classification	TabPFN	12	OpenML (46935)	CC0
Bank Marketing	Classification	TabPFN	13	OpenML (46910)	CC-BY 4.0
Good Customer	Classification	TabPFN	13	OpenML (46938)	CC0
Miami Housing	Regression	TabPFN	15	OpenML (46942)	CC-BY-NC-SA
Seismic Bumps	Classification	TabPFN	15	OpenML (46956)	CC-BY 4.0
Online Shoppers	Classification	TabPFN	17	OpenML (46947)	CC-BY 4.0
Churn	Classification	TabPFN	19	OpenML (46915)	MIT
Credit-G	Classification	TabPFN	20	OpenML (46918)	CC-BY 4.0
Airline Satisfaction	Classification	TabPFN	21	OpenML (46920)	CC0
JM1	Classification	TabPFN	21	OpenML (46979)	Public Domain
Credit Card Default	Classification	TabPFN	23	OpenML (46919)	CC-BY 4.0
HELOC	Classification	TabPFN	23	OpenML (46932)	Public Domain
Coupon Recommendation	Classification	TabPFN	24	OpenML (46937)	CC-BY 4.0
Marketing Campaign	Classification	TabPFN	25	OpenML (46940)	Public Domain
Hazelnut	Classification	LightGBM	30	OpenML (46930)	CC-BY-SA
Students Dropout	Classification	LightGBM	36	OpenML (46960)	CC-BY 4.0
Anneal	Classification	LightGBM	38	OpenML (46906)	CC-BY 4.0
QSAR Biodeg	Classification	LightGBM	41	OpenML (46952)	CC-BY 4.0
Diabetes 130-US	Classification	LightGBM	47	OpenML (46922)	CC-BY 4.0
Splice	Classification	LightGBM	60	OpenML (46958)	CC-BY 4.0
Bankruptcy	Classification	LightGBM	64	OpenML (46950)	CC-BY 4.0
Superconductivity	Regression	LightGBM	81	OpenML (46961)	CC-BY 4.0
COIL 2000	Classification	LightGBM	85	OpenML (46916)	CC-BY 4.0
NATICUSdroid	Classification	LightGBM	86	OpenML (46969)	CC-BY 4.0
Taiwanese Bankruptcy	Classification	LightGBM	94	OpenML (46962)	CC-BY 4.0
MIC	Classification	LightGBM	111	OpenML (46980)	CC-BY 4.0
APS Failure	Classification	LightGBM	170	OpenML (46908)	CC-BY 4.0
KDD Cup 09	Classification	LightGBM	212	OpenML (46939)	Public Domain
QSAR TID11	Classification	LightGBM	1024	OpenML (46953)	Public Domain
HivaAgnostic	Classification	LightGBM	1617	OpenML (46933)	Public Domain
Bioresponse	Classification	LightGBM	1776	OpenML (46912)	Public Domain
Graph datasets [43]					
Benzene	Classification	Graph Neural Network	25	Sanchez-Lengeling et al. [50]	CC0 1.0 Universal
Mutagenicity	Classification	Graph Neural Network	35	Kazius et al. [28]	CC0 1.0 Universal

We build an interventional game for each tabular dataset with more than 25 features, based on the underlying trained XGBoost [9] or LightGBM [29]. The interventional game is based on a point to explain x^e and background samples $\mathcal{B} = \{b_1, \ldots, b_{n_{bg}}\}$. Given the coalition S, we construct new

points

$$z_i[j] = \begin{cases} x^e[j] \text{ if } j \in S \\ b_i[j] \text{ otherwise} \end{cases}$$

where we access the feature j via $x[j]$. The game value of T is then obtained by averaging each value from $z_1, \ldots, z_{n_{bg}}$, i.e.

$$\nu(T) = \frac{1}{n_{bg}} \sum_{i=1}^{n_{bg}} f(z_i)$$

where we denote the underlying tree-based models as f. In total, we use $n_{bg} = 50$ background samples drawn from the training dataset. The exact values are calculated using Algorithm 2. In the following, we provide concise descriptions of each domain; an overview of all datasets is presented in Table 3.

Vision and Language For Vision and Language tasks, we use the precomputed games provided by `shapiq` [41], which are based on pretrained models and standard datasets for their respective domains. The language games are based on DistilBERT [51] fine-tuned on the IMDB movie reviews dataset [32, 38], while the image classification games use Vision Transformer and ResNet18 pretrained on ImageNet. The language games aim to predict the sentiment of movie reviews, while the image classification games aim to predict the class of input images.

Tabular Datasets We consider a variety of tabular datasets [14, 41, 55, 63]. For all tabular datasets with up to 25 features, we train TabPFN [20, 23], since this feature count still allows exact computation of the probabilistic indices. For all other tabular datasets, we either train an XGBoost model [9] or a LightGBM model [29] using default hyperparameters to obtain the interventional game. This makes our evaluation more comprehensive and enables us to investigate how the underlying model influences the interactions we obtain.

Graph Datasets For graph datasets, we follow the setup of Muschalik et al. [43] and use their proposed `GraphSHAP-IQ` method to compute exact Shapley interactions for graph neural networks. The MUTAGENICITY dataset [28] consists of 1,768 molecular graphs, categorized into two classes based on their mutagenic properties, specifically their effects on the Gram-negative bacterium *S. typhimurium* [43]. The BENZENE dataset [50] consists of 12,000 molecular graphs, each labeled according to whether it contains a benzene ring. We use the pretrained models from Muschalik et al. [43], which have 2 GNN layers and a hidden dimension of 32.

Sum Of Unanimity (SOUM) For the synthetic games, we follow the procedure described in [58]. A SOUM game ν is expressed as:

$$\nu(T) = \sum_{i=1}^{l} \alpha_i \mathbf{1}_{S_i \subseteq T}$$

where we denote l as the number of unanimity basis games and S_i as uniformly sampled subsets with a maximal size of k. Based on the experimental setup in [1], we build two games: 70 players with $l = 30$ and $k = 5$ and 90 players with $l = 10$ and $k = 10$. Each α_i is sampled uniformly over $[-1, +1]$. The exact Shapley interactions can be obtained via the Möbius transform, as ν is in its Möbius form.

C.2 Computational Resources

Approximation-quality experiments and hyperparameter optimization were conducted on a compute cluster with 96 Intel Xeon Platinum 8480+ (Sapphire Rapids) CPUs and 512 GB of RAM, totaling 3 weeks of runtime. TabPFN-based games were precomputed using NVIDIA V100 GPUs in parallel, enabling exhaustive evaluation of the value function for datasets with $n \leq 25$ features. CLIP experiments were run on a cluster consisting of NVIDIA A100 GPUs.

C.3 Hyperparameter Optimization

To obtain high-quality proxy models, we tune the XGBoost hyperparameters using Bayesian optimization (BO), since the approximation quality of ProxySHAP depends directly on the fitted proxy.

Compared with grid or random search, BO explores the predefined search space more adaptively and can therefore identify strong configurations with fewer evaluations. We use the BO implementation of Lindauer et al. [34].

The integer-valued hyperparameters are the number of estimators ($[500, 2000]$), maximum tree depth ($[2, 8]$), and minimum child weight ($[1, 20]$). The continuous hyperparameters are the subsampling rate ($[0.6, 1.0]$), column subsampling rate ($[0.6, 1.0]$), learning rate ($[0.001, 0.3]$), and ℓ_2 regularization strength ($[0.001, 50]$). We run BO for 200 iterations and evaluate each configuration using 5-fold cross-validation. After optimization, the final proxy model is retrained on the full set of sampled coalitions.

HPO runtime varies substantially across datasets, depending on the number of sampled coalitions and the number of features. Expanding the search space may further improve approximation quality, but would also increase the computational cost.

C.4 Baselines

We briefly describe the baseline methods used for comparison in our experiments. All baselines are based on the implementation of the `shapiq` library [41].

PermutationSamplingSII. PermutationSamplingSII is a Monte Carlo estimator for the Shapley Interaction Index (SII), extending classical permutation sampling for Shapley values [7] to higher-order interactions [16, 58]. The method estimates interactions by averaging marginal contributions over random orderings of the random features. It is applicable exclusively to SII.

SVARM-IQ. SVARM-IQ (Stratified Variable Approximation for Interaction Quantification) [31] is an efficient estimator for cardinal probabilistic interaction indices based on stratified sampling of marginal contributions. It builds on the MSR framework [60] and additionally uses stratified sampling, further reducing the variance of the estimate [8].

SHAP-IQ. SHAP-IQ [16] provides a unified framework for approximating Shapley interactions of arbitrary order by extending unbiased KernelSHAP to interaction indices. It supports all cardinal probabilistic interaction indices.

KernelSHAP-IQ. KernelSHAP-IQ [17] generalizes KernelSHAP to higher-order interactions by solving a weighted least-squares optimization problem with interaction-specific kernels. The method supports estimating both Shapley and Banzhaf interactions but suffers from scalability limitations due to the combinatorial growth of interaction terms.

ProxySPEX. ProxySPEX [5] is an inference-efficient method for estimating sparse Shapley and Banzhaf interactions, improving upon SPEX [27]. By learning a sparse proxy model, ProxySPEX significantly reduces the number of required model evaluations and scales to settings with thousands of features.

D Additional Approximation Results

D.1 Additional Experiments with FIxLIP

Following Baniecki et al. [2], we estimate model explanations with the faithful Banzhaf interaction index (FBII) of second-order and assess approximation quality using the area between the insertion/deletion curves (AID) and R^2 metrics. We rely on the default experimental setup, restricting pairwise interactions to the top 72 clique for ViT-B/16, selected via a greedy increase-in-value strategy. Given estimated interactions ϕ^{FBII}, we sample $m = 1000$ coalitions and define $\hat{\nu}(T) = \sum_{S \subseteq T} \phi_S^{\text{FBII}}$. The R^2 score is computed as

$$R^2 = 1 - \frac{\|\hat{\nu} - \nu\|^2}{\|\nu\|^2},$$

where ν denotes the sampled game values obtained from the model outputs. AID score measures the average increase/decrease in the model's prediction when sequentially inserting/deleting important features based on their ranking as measured with an explanation [21, 66].

Evaluating faithfulness of the game. We measure R^2 on explanations for 30 image–text inputs from MS COCO, with budgets ranging from 10^2 to 10^4. Figure 7 shows that ProxySHAP improves the original approximation in low-budget regimes, improves the R^2 metric more rapidly, and requires substantially fewer model evaluations than ProxySPEX. It effectively solves the challenge of approximating the $m \ll p$ regression matrix using a linear-based model, as reported in the original work.

Ablation on approximating the cross-modal FIxLIP estimator. We run additional ablations with another *cross-modal* estimator proposed in [2], which is more sample-efficient. The standard estimator samples image–text coalitions jointly, i.e., each CLIP call yields a single game evaluation. The cross-modal estimator samples image and text coalitions independently and evaluates the model on all pairwise combinations, yielding more game evaluations for the same CLIP-call budget. Table 4 summarizes the rough relationship between CLIP calls and the resulting number of game evaluations, which varies with vision model size and input text length. Figure 8 reports R^2 approximation performance for ProxySHAP on the FIxLIP game, which remains useful in the low-budget regimes, especially for the larger ViT-16 game.

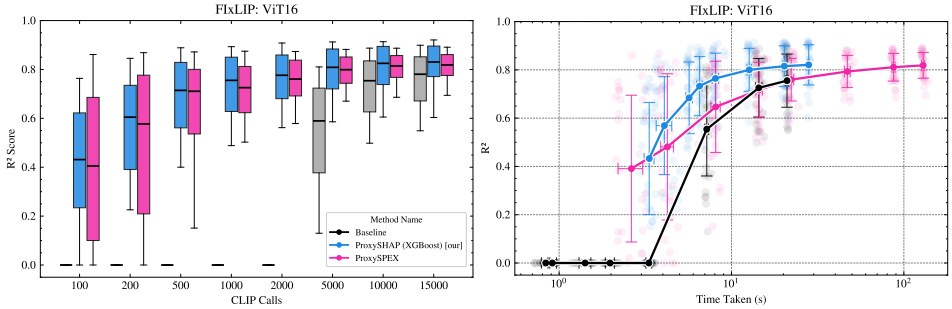


Figure 7: Faithfulness R^2 for explaining CLIP (ViT-16) on the MS COCO dataset with ProxySHAP, ProxySPEX, and the FIxLIP baseline.

D.2 XGBoost Default for Large Player Counts

When games involve many features, hyperparameter optimization (HPO) often selects XGBoost proxies with many shallow trees. In particular, the selected configurations often use approximately 2,000 trees together with a small maximum depth. We use this observation to define an HPO-informed default configuration, denoted by *ProxySHAP (XGBoost+HPO-Informed)*. Table 5 summarizes the differences between the standard XGBoost configuration used

Table 5: XGBoost proxy configurations.

Hyperparameter	Default	HPO-informed
$n_{\text{estimators}}$	100	2,000
`max_depth`	6	3
`learning_rate`	0.3	0.05
`reg_lambda`	1	5

Table 4: Relationship between CLIP calls and the resulting number of game evaluations under the standard and cross-modal estimators, averaged over the explained MS COCO images.

CLIP model	CLIP calls	Approximate Game Evaluations	Crossmodal Game Evaluations
ViT-B/16	100	100	790
	200	200	1 713
	500	500	6 834
	1 000	1 000	26 088
	2 000	2 000	102 107
	5 000	5 000	630 766
	10 000	10 000	2 645 308
ViT-B/32	100	100	2 870
	200	200	11 280
	500	500	69 654
	1 000	1 000	278 522
	2 000	2 000	1 112 185
	5 000	5 000	6 945 324
	10 000	10 000	24 062 728

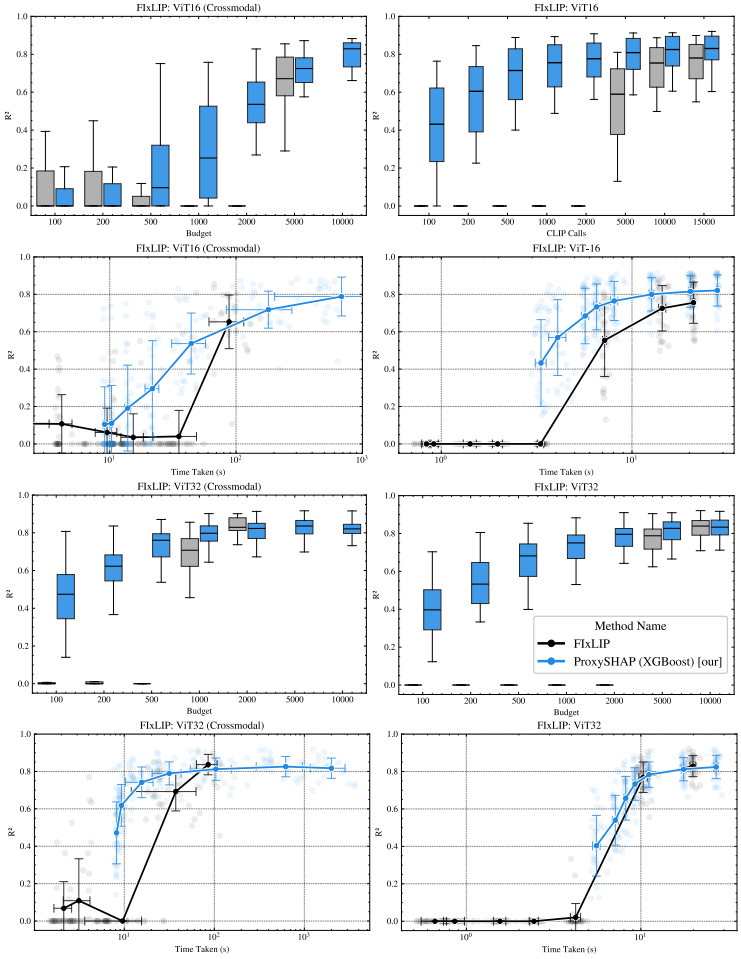


Figure 8: Ablation on approximating the cross-modal FIxLIP estimator. Faithfulness R^2 for explaining two CLIP variants on MS COCO with ProxySHAP and the FIxLIP baseline.

in ProxySHAP and the HPO-informed configuration. Motivated by this observation, we evaluate *ProxySHAP (XGBoost+HPO-Informed)* as an alternative default proxy for large-scale games. The results show that this configuration improves approximation quality in low-budget regimes and for games with many players. As illustrated in Figure 9, the HPO-informed configuration consistently outperforms the standard XGBoost default at low budgets. For moderate feature counts, however, its advantage decreases as the budget increases, and the standard configuration eventually becomes competitive or superior.

For datasets with more than 1,000 features, *ProxySHAP (XGBoost+HPO-Informed)* significantly outperforms the standard default across all considered budgets. This is the case, for example, on HIVAAGNOSTIC and BIORESPONSE, which contain 1,617 and 1,776 features, respectively. These

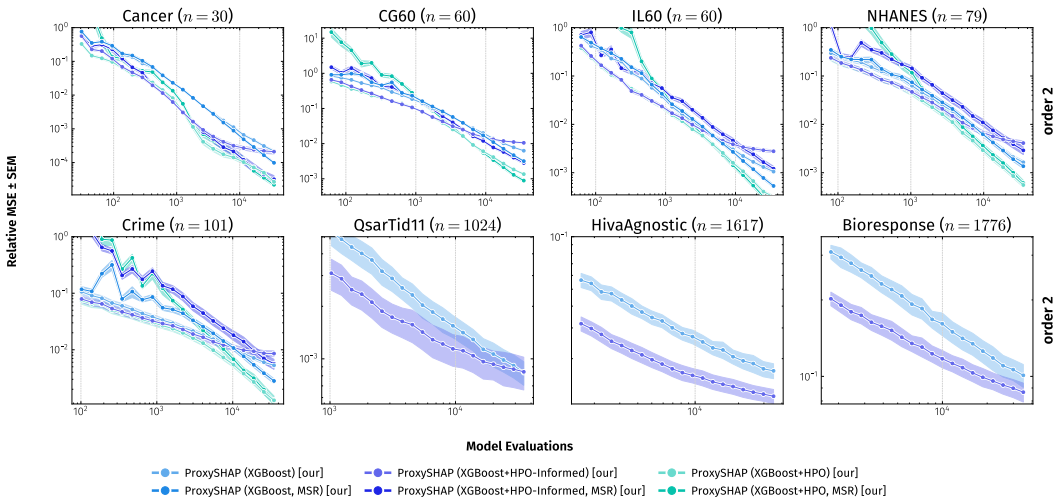


Figure 9: Approximation quality of two different XGBoost defaults. We show that using 2000 trees with a maximum depth of 3 improves estimation quality in low- to medium-budget regimes.

results further highlight the importance of proxy hyperparameters for ProxySHAP, especially in low-budget regimes and for games with many players, where the standard XGBoost default may be insufficient to capture the relevant interaction structure.

D.3 Runtime

We evaluate runtime by translating model evaluations into wall-clock time using fixed per-evaluation costs of $0\,\mathrm{ms}$, $1\,\mathrm{ms}$, $10\,\mathrm{ms}$, and $100\,\mathrm{ms}$. These regimes capture increasingly expensive inference settings, from highly optimized models to large-scale neural networks. Figure 10 reports the resulting runtimes for second- and third-order interaction estimation.

Figure 10 reports approximation quality together with runtime, ranging from zero model-evaluation cost (top row) to $100\,\mathrm{ms}$ per model call. We find that the linear proxy incurs higher computational overhead than the tree-based proxy, especially for higher-order interactions, due to its polynomial scaling in the number of interaction terms. It only outperforms the tree-based proxy on a few games, such as GNN on Mutagenicity.

ProxySHAP is faster than ProxySPEX when model-inference costs are ignored. As model evaluations increasingly dominate the runtime, this difference becomes less pronounced. We observe a similar pattern for MSR adjustment: its overhead grows with the interaction order and number of players, but becomes less visible as the cost of model calls increases.

Overall, ProxySHAP remains an efficient estimator for Shapley and Banzhaf interactions, requiring less computation than current proxy-based baselines, most notably ProxySPEX.

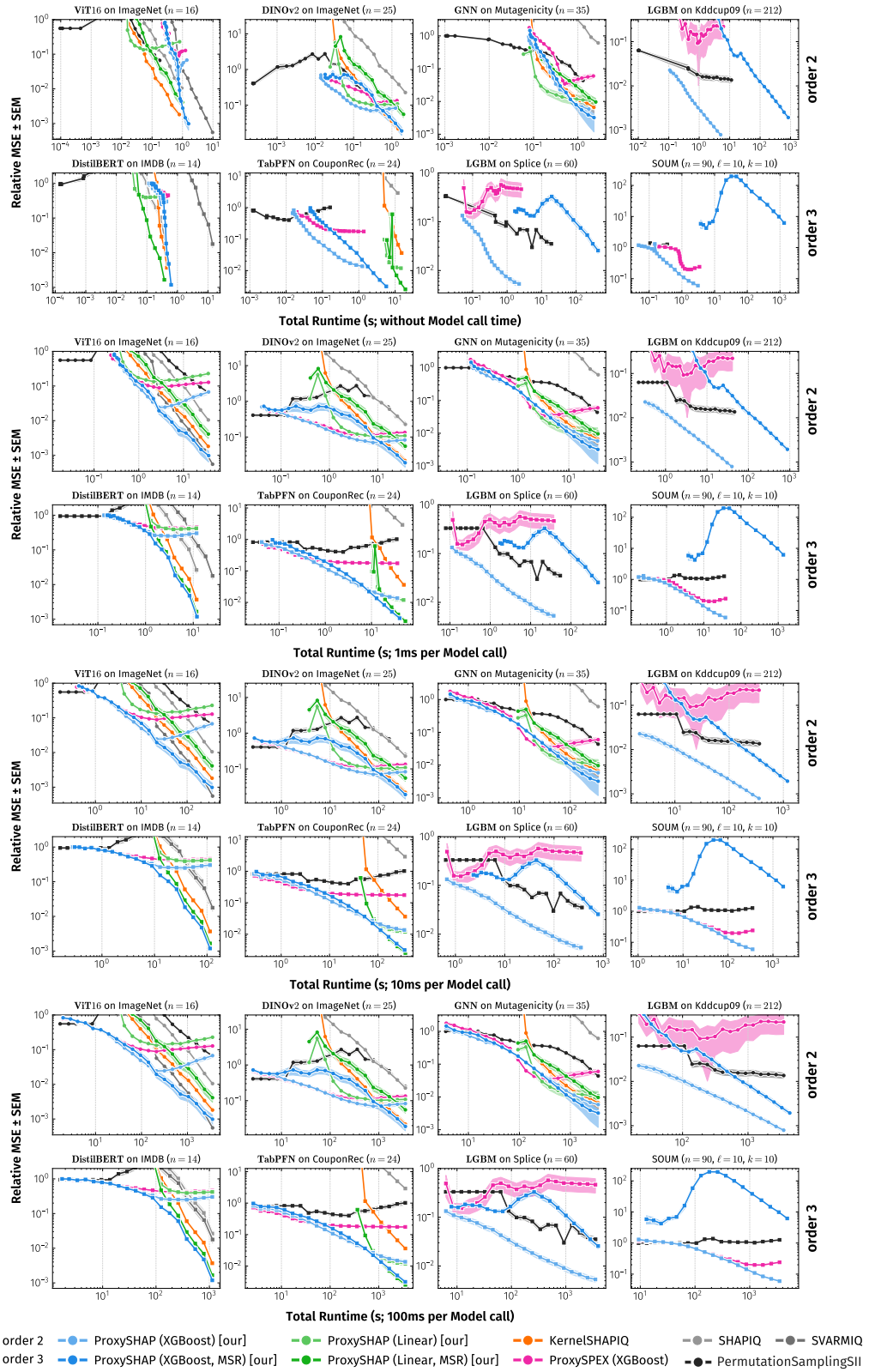


Figure 10: Approximation quality as a function of runtime for second- and third-order interaction estimation across different per-evaluation cost regimes.

D.4 Ablation Details

We further investigate the effect of the residual approximator and sampling weights used in the adjustment step. Specifically, we compare SHAP-IQ [16], KernelSHAP-IQ [17] and SVARM-IQ [31] as model-agnostic residual approximators. We also compare uniform over sizes, as used in LeverageSHAP [44], with KernelSHAP-IQ weights [17]. As underlying games, we use VIT4BY4PATCHES, BIKESHARINGLOCALXAI, CALIFORNIAHOUSINGLOCALXAI, CORRGROUPS60LOCALXAI, and COMMUNITIESANDCRIMELOCALXAI; details on these datasets are provided in Section C.1. For each game, we approximate second-order Shapley interaction indices so that the comparison directly reflects the effect on interaction estimation. As shown in Figure 11, neither the choice of residual approximator nor the choice of sampling weights has a clear systematic effect on approximation quality for these games.

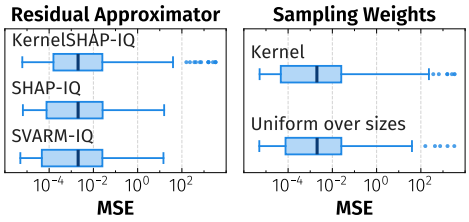
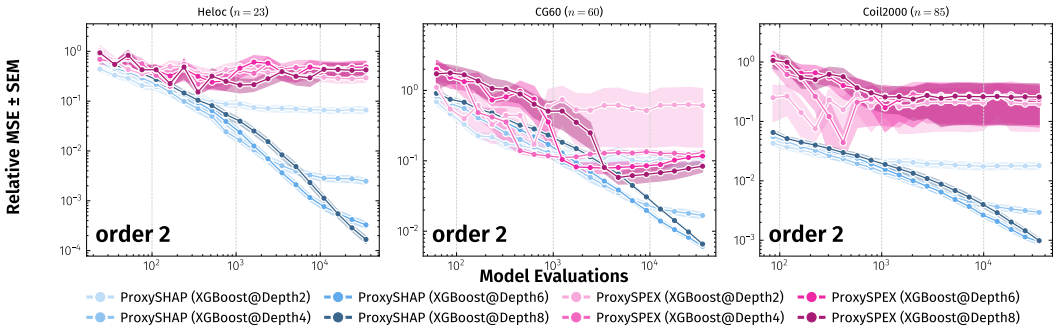
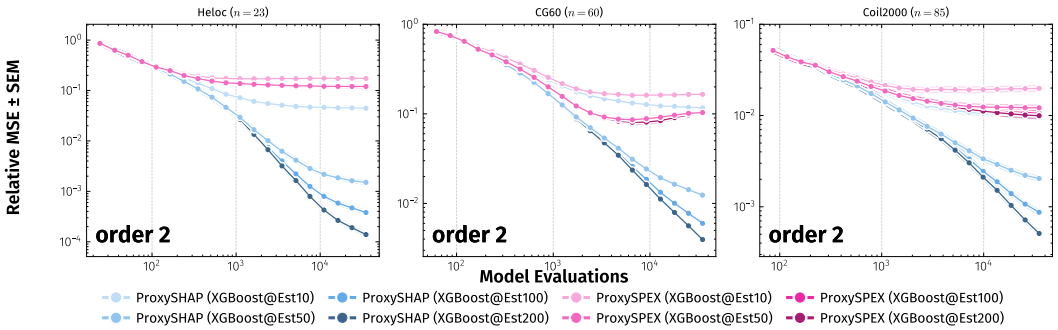


Figure 11: Ablations of sampling weights and residual approximators for ProxySHAP.

D.5 Fourier Extraction vs. Interventional Extraction



(a) Approximation quality (Relative MSE) of ProxySHAP and ProxySPEX using different maximum tree depth options across small, medium, and large player domains.



(b) Approximation quality (Relative MSE) of ProxySHAP and ProxySPEX using different numbers of trees across small, medium, and large player domains.

Our method relies on the ability to efficiently extract exact cardinal-probabilistic interaction indices from the underlying tree-based model. We extend interventional TreeSHAP by Zern et al. [65] to extract the *exact* cardinal-probabilistic interaction indices of $\hat{\nu}$ for a target set $\mathcal{S} \subseteq 2^N$ in $\mathcal{O}(n_{\text{nodes}} \cdot |\mathcal{S}|)$ time, where n_{nodes} denotes the number of tree nodes. An alternative is Fourier extraction, as proposed by Butler et al. [5], Gorji et al. [18], whose cost grows exponentially with the maximum tree depth d. In the worst case, a tree of depth d can induce up to $\mathcal{O}(4^d)$ Fourier coefficients [18], which must then be extracted and converted to the desired interaction indices.

We compare the two extraction approaches on nine datasets: BIKE, ADULTCENSUS, CRIME, FOREST-FIRES, IL60, CG60, NHANES, CANCER, and ESTATE. For each method, we measure the runtime required to extract all target interactions. We exclude preprocessing time, such as storing the tree structure in a hash map, since this is a one-time cost that can be amortized across multiple extraction runs. For Fourier extraction, we measure the time required to extract the Fourier coefficients and convert them to the target interaction indices. For interventional extraction, we measure the runtime of our extraction algorithm directly. We use the Fourier extraction implementation provided by `shapiq` [41] and our interventional extraction implementation based on Algorithm 2. We report the speedup as the ratio between the runtime of Fourier extraction and the runtime of interventional extraction in Figure 13. Since a tree can only contain interactions of order k if its depth is at least k, we report order-k speedups only for trees with depth at least k.

The results show that interventional extraction consistently outperforms Fourier extraction across datasets, with speedups ranging from $2\times$ to more than $10^4\times$, depending on the dataset and maximum tree depth. The only exception occurs for shallow trees of depth 3 when extracting third-order interactions, where Fourier extraction can be slightly faster on large datasets such as CRIME.

We additionally investigate the effect of varying the tree depth and the number of estimators of the XGBoost proxy on the approximation quality of ProxySPEX and ProxySHAP in Figure 12a and 12b.

Tree Depth. To this end, we set the maximum depth of the individual XGBoost trees [9] to 2, 4, 6, and 8. We observe that shallow trees perform particularly well for small coalition budgets, while medium-depth trees are effective across most of the budget range considered. For large coalition budgets, deeper trees become increasingly beneficial, suggesting that the default depth of 6 may no longer capture all relevant interaction structure once sufficient training data are available. This trend is evident in both ProxySHAP and ProxySPEX, underscoring the importance of adjusting the tree depth to the available coalition budget.

Number of Estimators. For the number of estimators, we choose the values 10, 50, 100, and 200. We observe that the effect of more trees only becomes visible when given a sufficient budget. Notably, the more trees we have, the more interaction structure it can capture, and thus the better it approximates the underlying value function. This becomes most prominent as we increase the budget, with more of the underlying game's structure revealed. Interestingly, ProxySPEX does not gain much from an increased number of trees, whereas ProxySHAP clearly does as the budget increases.

D.6 Detailed Comparison of ProxySHAP and ProxySPEX

ProxySPEX [5] is a model-agnostic approximation method for computing any-order cardinal-probabilistic interaction indices, including in settings with large feature counts. It consists of three main steps with an optional fourth step:

1. **Sampling and evaluation.** Coalitions $\mathcal{T} \subseteq 2^N$ are sampled and evaluated, yielding the dataset
$$\mathcal{D} = \{(T, \nu(T))\}_{T \in \mathcal{T}}.$$
2. **Proxy fitting.** A gradient-boosted tree model is fitted on $\mathcal{D}$ by minimizing the mean squared error. By default, `shapiq` [41] uses a LightGBM with an additional hyperparameter sweep. For all experiments, we restrict ourselves to the default XGBoost parameters to ensure the same tree across all proxy-based methods.
3. **Fourier extraction and truncation.** Fourier coefficients are extracted from the fitted tree proxy. ProxySPEX then keeps a minimal subset $C^\star \subseteq \mathcal{F}$ of coefficients that explains at least 95% of the total squared Fourier mass,
$$C^\star = \arg\min_{C \subseteq \mathcal{F}} |C| \quad \text{s.t.} \quad \frac{\sum_{F \in C} F^2}{\sum_{F \in \mathcal{F}} F^2} \geq 0.95,$$
where $\mathcal{F}$ denotes the set of Fourier coefficients extracted from the tree.
4. **Adjustment (optional).** Given the truncated coefficient set $C^\star$, ProxySPEX applies a refinement step to improve the extracted Fourier coefficients. It constructs a design matrix $X \in \{-1, +1\}^{|\mathcal{T}| \times |C^\star|}$ with entries
$$X_{i,j} = (-1)^{|T_i \cap C_j|},$$

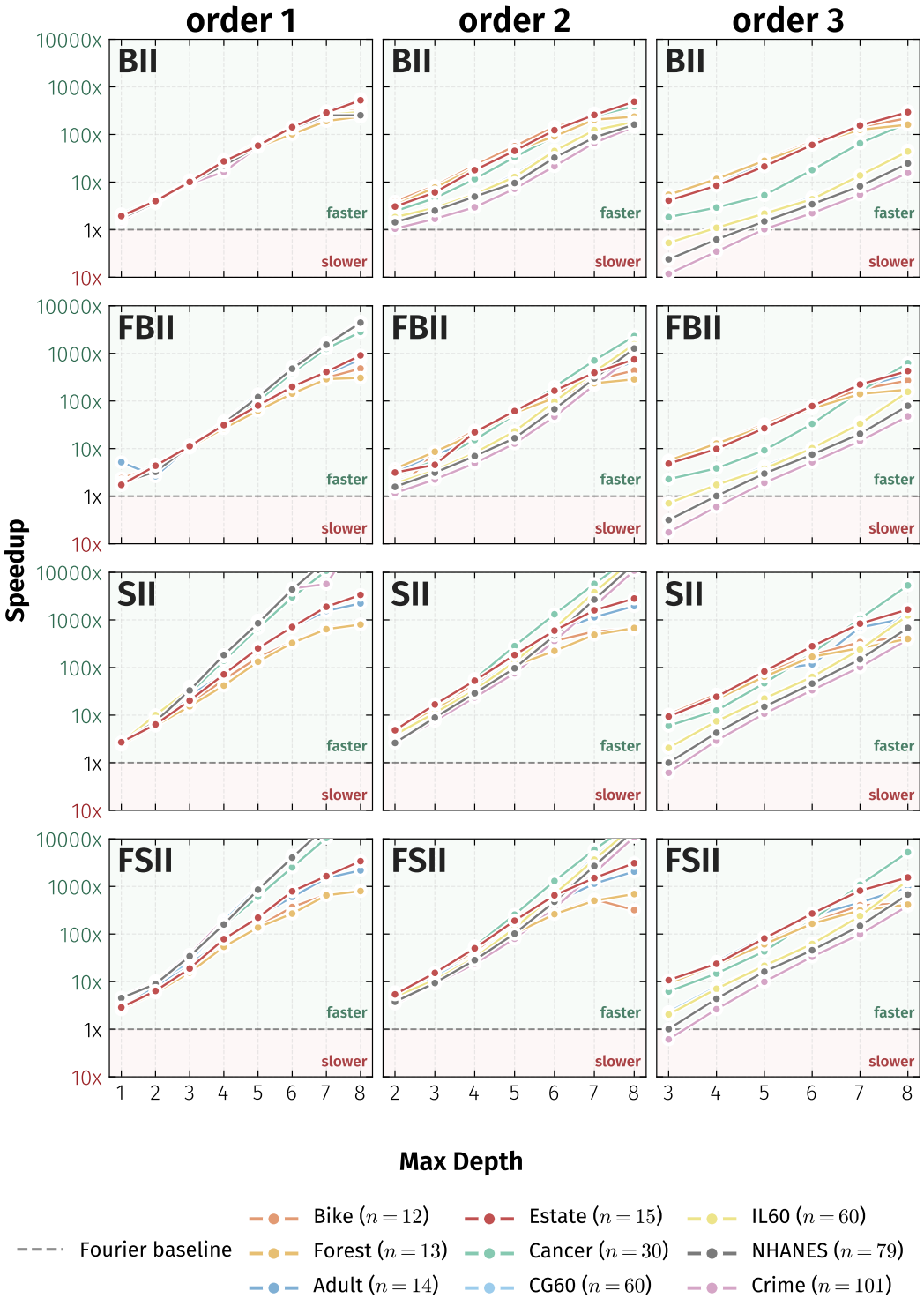


Figure 13: Speedup of interventional extraction compared to Fourier extraction for extracting all interactions of order 1, 2, and 3 across different datasets.

and solves the regularized regression problem

$$F^\star = \arg\min_{F \in \mathbb{R}^{|C^\star|}} \|\nu - XF\|_2^2 + \lambda \|F\|_2^2.$$

The truncation step is essential for making the refinement step computationally feasible, since the number of Fourier coefficients of a tree can grow as $\mathcal{O}(4^d)$ with the tree depth d. Based on the refined Fourier coefficients $F^\star$, ProxySPEX can compute any interaction index having a Möbius transform as in Eq. (4), since the Fourier coefficients can be translated into Möbius coefficients [2]. For the exact transformations, we refer to Appendix A of Butler et al. [5].

ProxySHAP follows a closely related proxy-based strategy, but differs in two important aspects. First, instead of extracting Fourier coefficients and converting them into the desired index, ProxySHAP directly extracts the target cardinal-probabilistic interaction index from the tree proxy. Second, ProxySHAP applies a case-by-case MSR adjustment directly to the residual game. This yields a consistent estimator whenever the residual correction is sufficiently well covered by the sampled coalitions, and it leads to improved performance in regimes where sufficient budget is available (see Figures 4 and 18).

Figure 14 illustrates how the approximation quality of ProxySPEX changes as the cutoff threshold is increased and, consequently, more Fourier energy is retained in C^*. As the cutoff approaches 1, ProxySPEX increasingly recovers the behavior of ProxySHAP, but at substantially higher computational cost, since up to $\mathcal{O}(4^d)$ Fourier coefficients may need to be converted to the desired interaction index. In contrast, ProxySHAP consistently exhibits near-diagonal behavior across budgets and datasets, and is only outperformed by KernelSHAP-IQ once the latter receives a sufficiently large evaluation budget. For COIL2000, even a cutoff of 0.999 does not lead to substantial improvements, since more than 48% of the Fourier coefficients are still discarded. We refer to Figure 18 for a complementary overview of the best-performing methods on these datasets.

D.7 When to use Adjustment?

The adjustment step is designed to make the resulting interaction estimates consistent. To assess its practical relevance, we complement this theoretical motivation with an empirical analysis across all 32 considered TabArena datasets [14], budgets, and interaction orders. Since each experiment is repeated over 30 explained instances per dataset, we measure the instance-wise relative change in approximation quality as

$$r_i = \frac{\text{MSE without adjustment}}{\text{MSE with adjustment}},$$

and report the geometric mean

$$\bar{r} = \exp\left(\frac{1}{N}\sum_{i=1}^{N} \log r_i\right),$$

pooled across all instances with the same number of players. The shaded bands show the geometric standard deviation factor,

$$s = \exp(\mathrm{std}(\log r_i)),$$

drawn as the multiplicative interval $[\bar{r}/s,\ \bar{r} \cdot s]$, corresponding to one standard deviation in log-space. Hence, values above one indicate that adjustment improves approximation quality, whereas values below one indicate that it deteriorates.

For interaction indices, however, the effect of adjustment is more nuanced. While adjustment can reduce proxy bias, it can also introduce substantial variance, especially for higher-order interactions. This is consistent with our theoretical analysis in Section 3, where we show that the dominant variance term of the adjustment estimator scales as n^{k-1}, with n denoting the number of features and k the maximal interaction order. Consequently, adjustment can deteriorate approximation quality when the feature dimension or interaction order is large. Empirically, Figure 3 shows that, for second-order interactions, adjustment remains beneficial up to datasets with 90 features, provided that the budget is sufficiently large. In contrast, for third-order interactions, the adjustment already degrades approximation quality in games with more than 30 features, even at large budgets.

[2] As the Fourier coefficients also form a basis of the games, it is sometimes computationally better to directly use those instead of transforming to Möbius coefficients.

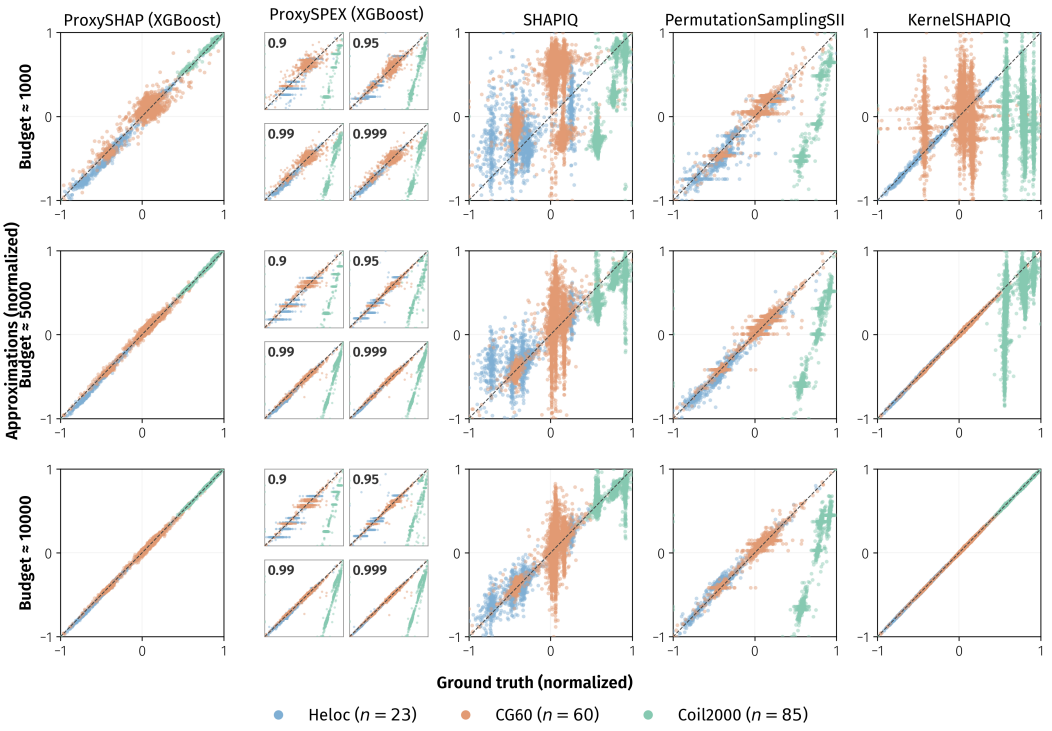


Figure 14: Predicted versus ground-truth normalized interaction values for different approximation methods and sampling budgets. Each point represents an interaction value from a dataset and a benchmark run; points closer to the diagonal indicate better agreement with the exact interaction values. Columns compare ProxySHAP, ProxySPEX, SHAPIQ, PermutationSamplingSII, and KernelSHAPIQ, while rows correspond to increasing evaluation budgets. In ProxySPEX, the central column shows four cutoff thresholds, with larger cutoffs retaining more Fourier energy in the surrogate approximation. As the ProxySPEX cutoff increases, the scatter increasingly aligns with ProxySHAP, indicating that ProxySPEX becomes closer to ProxySHAP as more Fourier energy is retained.

We therefore recommend applying MSR adjustment primarily when the available coalition budget is substantially larger than the dominant variance term, i.e., when $|\mathcal{T}| \gg n^{k-1}$. In practice, this supports using adjustment for second-order interactions at sufficiently large budgets, while omitting it for third-order interactions and beyond unless the budget is exceptionally large.

Banzhaf interactions. For the Banzhaf interaction index, Corollary A.8 gives a variance factor of $\Theta(4^s\sqrt{n})$ with uniform over sizes, which grows more slowly in n than the $\Theta(n^{s-1})$ factor for Shapley interactions with $s \geq 2$. This asymptotic advantage, however, only takes effect once $n > 4^{2s/(2s-3)}$ when comparing the leading terms. Empirically, at a budget of $35{,}000$ coalitions, adjustment for second-order Banzhaf interactions stops being beneficial from roughly $n \approx 35$, whereas for Shapley interactions it remains beneficial up to $n = 90$ (see Figure 15 for the Banzhaf counterpart of Figure 3). The $n < 30$ recommendation therefore applies to Banzhaf interactions as well.

D.8 Shared vs. Disjoint Subsets in ProxySHAP

We compare two strategies for using the sampled coalitions: reusing the same set for both proxy fitting and residual adjustment, or splitting it into disjoint sets. For the disjoint variant, we first sample coalitions $\mathcal{T}$ and then split them into $\mathcal{T}^{\text{Proxy}}$ and $\mathcal{T}^{\text{Adjustment}}$, which are used for fitting the proxy and estimating the residual correction, respectively. As shown in Figure 16, enforcing disjoint sets degrades approximation quality for cardinal-probabilistic interactions. This provides empirical evidence that the observation of Witter et al. [63] also holds for interaction indices: reusing the same sampled coalitions for proxy fitting and adjustment is preferable in practice. Based on this

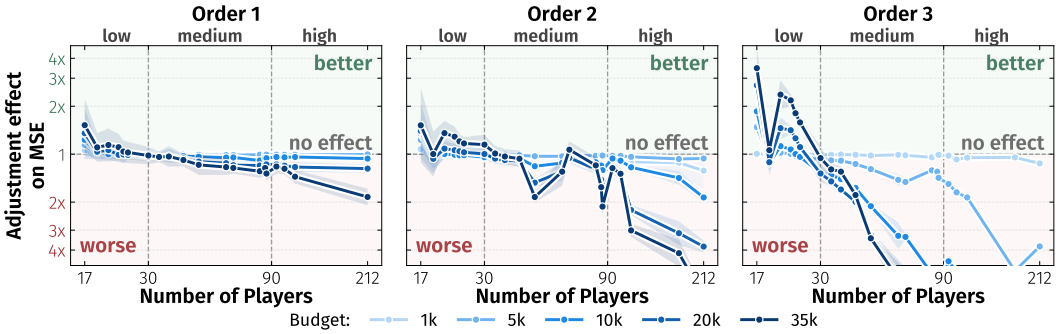


Figure 15: MSR adjustment effect for *Banzhaf* values and interactions of orders one to three, in the format of Figure 3. Values above one indicate that adjustment improves approximation quality. Even for Banzhaf values (order one), the adjustment helps only mildly and only for small player counts; for higher orders, the regime in which adjustment helps shrinks faster with the number of players than for Shapley interactions, consistent with the 4^s-dominated regime of Corollary A.8.

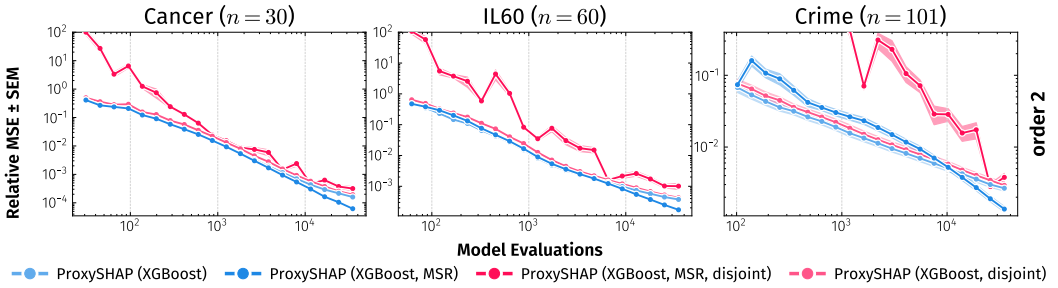


Figure 16: ProxySHAP with disjoint coalition sets for proxy fitting and residual adjustment.

finding, all experiments in the main paper use the same sampled coalitions for both steps, as shown in Algorithm 1.

D.9 Approximation Quality

We provide additional approximation-quality results across datasets and budgets for second- and third-order Shapley interaction indices and Banzhaf interaction indices (Figure 17). We also report winner maps for SII (Figure 18) and BII (Figure 19), where the winner is the method with the lowest average relative MSE across the 30 explained instances for each dataset and budget. A complete collection of approximation curves is available at `https://github.com/Advueu963/ProxySHAP`, alongside additional metrics such as Pearson's correlation.

Banzhaf Interaction Index. As shown in Figure 19, ProxySHAP with an XGBoost proxy almost always outperforms all baselines across datasets and budgets for both second- and third-order interactions. For second-order interactions, the linear proxy can be stronger on some datasets, such as CRIME and SOYBEAN. For third-order interactions, however, the tree-based proxy tends to perform better. As the number of players and target interactions grows, the linear proxy becomes less effective because fitting the interaction basis requires larger budgets. Overall, ProxySHAP with an XGBoost proxy is the strongest method for approximating Banzhaf interaction indices in our experiments.

Shapley Interaction Index. Figure 18 shows that ProxySHAP with an XGBoost proxy consistently outperforms all baselines across budget and player regimes. For smaller datasets, ProxySHAP outperforms for mid-sized budget counts, but shifts to the lower-budget regime as the number of players increases. For pairwise interactions on datasets with roughly 30 features and sufficiently large budgets, KernelSHAP-IQ can outperform both XGBoost and linear ProxySHAP variants, e.g., on SOYBEAN, IONOSPHERE, and STUDENTS DROPOUT. However, as the number of features increases, KernelSHAP-IQ becomes applicable in fewer regimes, and ProxySHAP with XGBoost becomes the

best-performing method. The winner maps also show that the regime in which MSR adjustment is beneficial shrinks with increasing feature count for both second- and third-order interactions. For third-order interactions, ProxySHAP with XGBoost outperforms all baselines across datasets and budgets, whereas the linear proxy is preferable in only a few cases, such as CANCER.

E Additional Related Work

Beyond proxy-based methods, research has focused on variance reduction for Monte Carlo estimates of probabilistic values [30, 56, 60] and interactions [31, 58]. Parallel to our work is *amortization*, where an explanation model is trained to directly predict attributions [11, 24] or interactions [13], rather than explaining the proxy post-hoc. Further related work includes ways to improve estimation of the value function for explanations via marginal distribution compression [1] and conditional distribution generation [46]. Specialized polynomial-time algorithms exist for computing values and interactions for k-nearest neighbours [61], SVMs [39], and graph neural networks [43]. Within tree-based methods, Muschalik et al. [42] derived exact algorithms for *path-dependent* interactions; however, unlike our proposed interventional extension, their method is incompatible with proxy modeling.

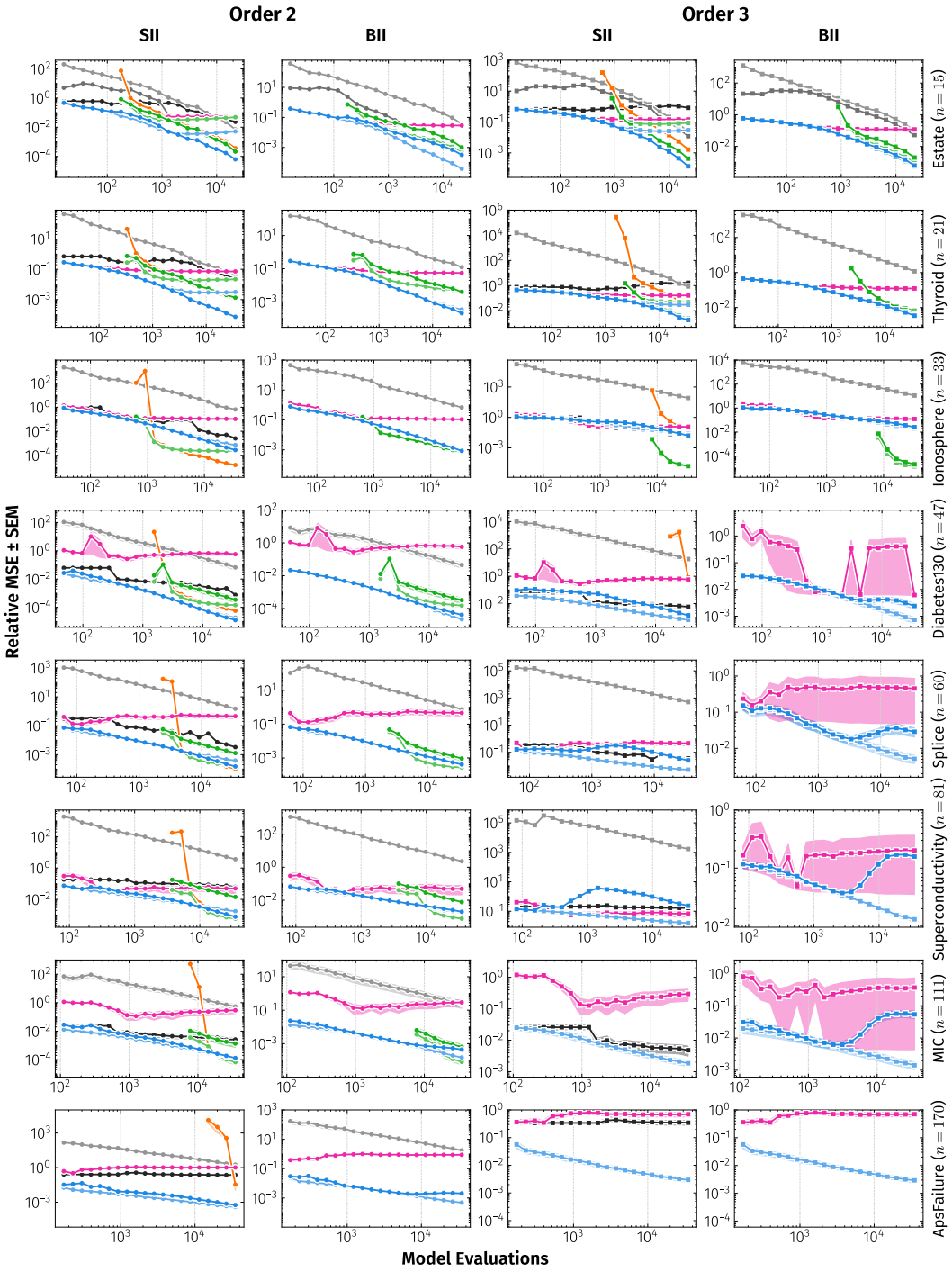


Figure 17: Selection of representative approximation curves for SII and BII at second- and third-order interactions.

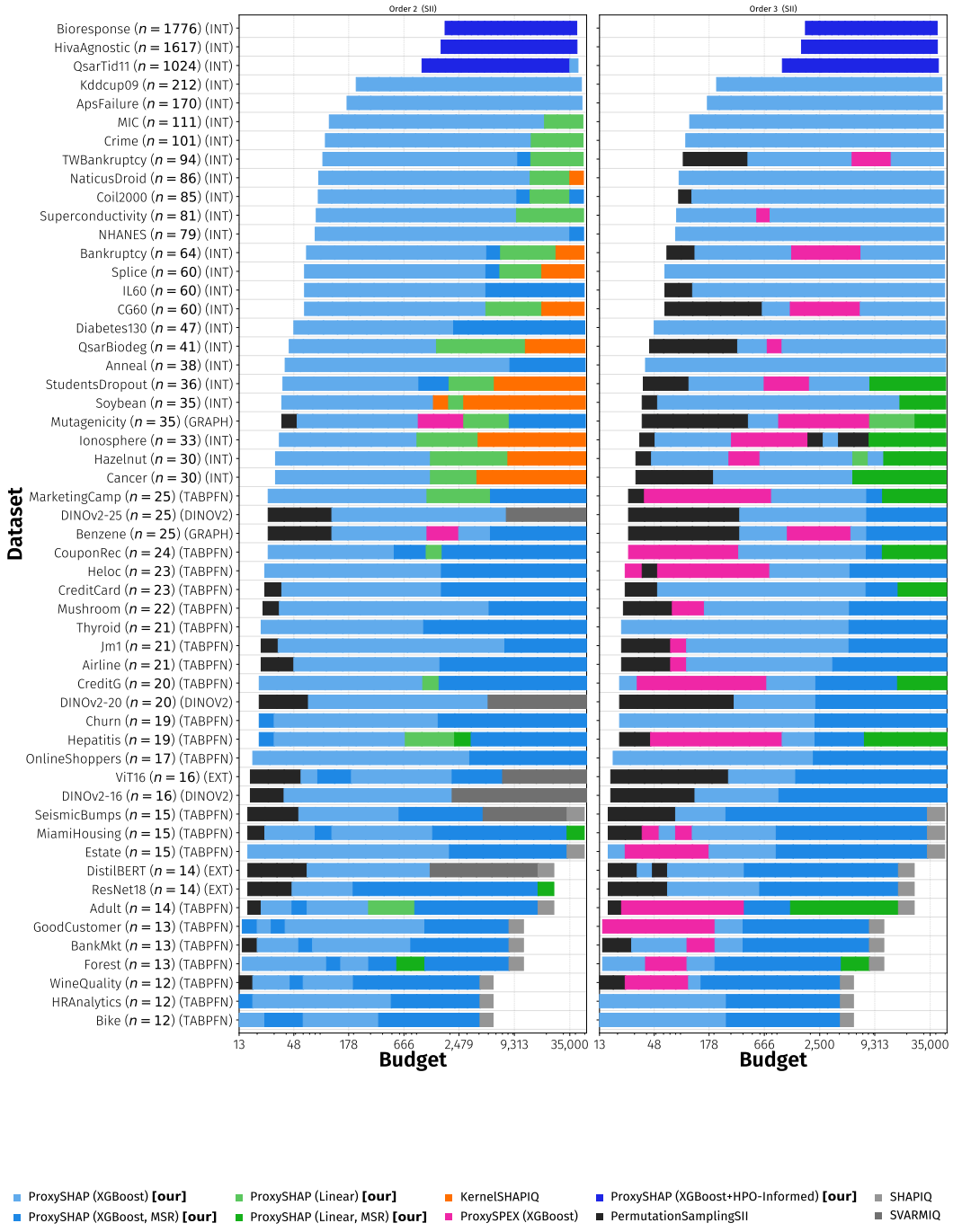


Figure 18: Winnermap comparing the best-performing method for each dataset and budget for SII orders 2 and 3. Note that the HPO-Informed variants are considered only in this overview for datasets with more than 1000 features.

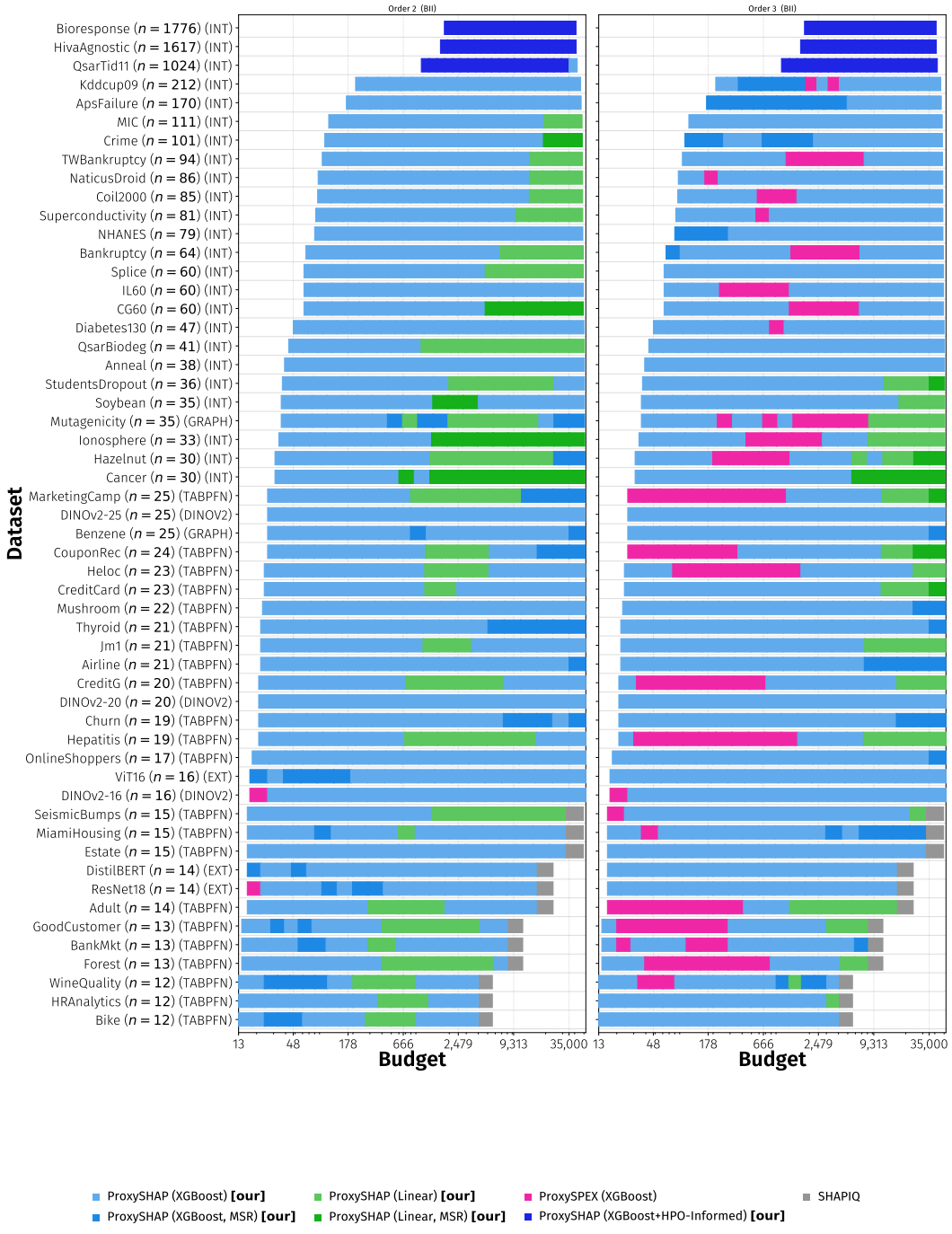


Figure 19: Winnermap comparing the best-performing method for each dataset and budget for BII orders 2 and 3. Note that the HPO-Informed variants are considered only in this overview for datasets with more than 1000 features.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: Please see Section 3, Section 4 and for the proofs see Appendix A.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Please see the end of Section 5.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

Justification: Please see Section 3.2 and Appendix A.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: Please see Section 4 as well as Appendix C.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: Please see the code: https://github.com/Advueu963/ProxySHAP

Guidelines:

- The answer [N/A] means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (https://neurips.cc/public/guides/CodeSubmissionPolicy) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (https://neurips.cc/public/guides/CodeSubmissionPolicy) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Please see Section 4 as well as Appendix C.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: Please see Section 4 as well as Appendix C.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.

- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Please see Appendix C.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: We have reviewed the NeurIPS Code of Ethics and confirm that our research conforms to it in every respect.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: Please consult the Broader Impact statement in Section 5.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: We do not release any data or models with a high risk for misuse, so we have not put in place any safeguards.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: Please see Appendix C.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.

- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: Please see https://github.com/Advueu963/ProxySHAP

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: Our paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: Our paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [N/A]

Justification: The core method development does not involve LLMs as any important, original, or non-standard components.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.