A Overview

In this supplementary material we provide the following information:

- Appendix B discuss other high-quality and Monte Carlo explainers.
- Appendix C discuss a guide to select the coverage α when the agent providing selective explanations has a budget for the average number of inferences to provide an explanation.
- Appendix D shows more experimental results on selective explanations.
- Appendix E shows the proofs for the theoretical results in Section 4.

B Additional Explanation Methods

In this section, we describe high-quality, Monte Carlo, and amortized explainers with further details.

B.1 High-Quality Explainers

Shapley Values (SHAP) [21] is a **high-quality** explainer that attributes a value ϕ_i for each feature x_i in $\boldsymbol{x} = (x_1, ..., x_d)$ which is the marginal contribution of feature x_i if the model was to predict $\boldsymbol{y}$ (2).

$$\phi_i(\boldsymbol{x}, \boldsymbol{y}) = \frac{1}{d} \sum_{S \subset [d]/\{i\}} \binom{d-1}{|S|}^{-1} \left(h_{\boldsymbol{y}}(\boldsymbol{x}_{S \cup \{i\}}) - h_{\boldsymbol{y}}(\boldsymbol{x}_S)\right). \tag{13}$$

SHAP has several desirable properties and is widely used. However, as (2) indicates, computing Shapley values and the attribution vector $\mathsf{HQ}(\boldsymbol{x}, \boldsymbol{y}) = (\phi_1(\boldsymbol{x}, \boldsymbol{y}), ..., \phi_d(\boldsymbol{x}, \boldsymbol{y}))$ requires 2^d inferences from h, making SHAP impractical for large models where inference is costly. This has motivated several approximation methods for SHAP, discussed next[6].

Local Interpretable Explanations (Lime). Lime is another feature attribution method [28] widely used to provide feature attributions. It relies on selecting combinations of features, removing these features from the input to generate perturbations, and using these perturbations to approximate the black box model h locally by a linear model. The coefficients of the linear model are considered to be the attribution of each feature. Formally, given a weighting kernel $\pi(S)$ and a penalty function Ω, the attribution produced by lime are given by

$$(\phi, a) = \underset{\phi \in \mathbb{R}^d, a \in \mathbb{R}}{\operatorname{argmin}} \sum_{S \subset [d]} \pi(S) \left(h(\boldsymbol{x}_S) - a_0 - \sum_{i \in S} \phi_i \right), \tag{14}$$

where $\mathsf{HQ}(\boldsymbol{x}, \boldsymbol{y}) = \phi$. As in SHAP, to compute the feature attributions using lime, we need to perform a large number of model inferences, which is prohibitive for large models.

B.2 Monte Carlo Lime

Shapley Value Sampling (SVS) [23] is a **Monte Carlo** explainer that approximates SHAP by restricting the sum in (2) to specific permutations of feature. SVS computes the attribution scores by uniformly sampling m features permutations $S_1, ..., S_m$ restricting the sum in (2) and performing $n = md + 1$ inferences. We denote SVS that samples m feature permutations by SVS-m.

Kernel Shap (KS) [21] is a **Monte Carlo** explainer that approximate the Shapley values using the fact that SHAP can be computed by solving the optimization problem

$$(\phi, a) = \underset{\phi \in \mathbb{R}^d, a \in \mathbb{R}}{\operatorname{argmin}} \sum_{i=1}^{n} \pi(S_i) \left(h(\boldsymbol{x}_{S_i}) - a_0 - \sum_{j \in S_i} \phi_j \right), \tag{15}$$

using $\pi(S) = \binom{d}{|S|} |S| (d - |S|)$ and where $\mathsf{MC}^n(\boldsymbol{x}, \boldsymbol{y}) = \phi$. Kernel Shap samples $n > 0$ feature combinations $S_1, ..., S_n$ and define the feature attributions to be given by the coefficients ϕ. We refer to Kernel Shap using n inferences as KS-n. We use the KS-n from the Captum library [17] for our experiments.

[6] We also discuss Lime and its amortized version in Appendix B

Sample Constrained Lime. To approximate the attributions from Lime, we consider the sample-contained version of (15). Instead of sampling all feature combinations in $[d]$, we only uniformly sample a fixed number n of feature combinations $S_1, ..., S_n$. For our experiments, shown in the appendix, we use the Sample Constrained Lime from the Captum library [17].

B.3 Amortized Explainers

Stochastic Amortization [6] is a **Amortized** explainer that uses noisy Monte Carlo explanations to learn high-quality explanations. Covert et al. [6] trained an amortized explainer $\mathsf{Amor} \in \mathcal{F}$ in a hypothesis class $\mathcal{F}$ (we use multilayer perceptrons) that takes an input and predicts an explanation. Specifically, taking the amortized explainer to be the solution of the training problem given in (3).

$$\mathsf{Amor} \in \underset{f \in \mathcal{F}}{\operatorname{argmin}} \sum_{(\boldsymbol{x},\boldsymbol{y}) \in \mathcal{D}_{\text{train}}} \|f(\boldsymbol{x},\boldsymbol{y}) - \mathsf{MC}^n(\boldsymbol{x},\boldsymbol{y})\|_2^2. \tag{16}$$

We are interested in explaining the predictions of large models for text classification. However, the approach in (3) is only suitable for numerical inputs. Hence, we follow the approach from Yang et al. [34] to explain the predictions of large language models, explained next.

Amortized Shap for LLMs [34] is a **Amortized** explainer similar to the one in (3) but tailored for LLMs. First, the authors note that they can use the LLM to write all input texts $\boldsymbol{x}$ as a sequence of token embedding $[e_1(\boldsymbol{x}), ..., e_{|\boldsymbol{x}|}(\boldsymbol{x})]$ where $e_i(\boldsymbol{x}) \in \mathbb{R}^d$ denotes the LLM embedding for the i-th token contained in the input text $\boldsymbol{x}$ and $|\boldsymbol{x}|$ is the number of tokens in the input text. Second, they restrict $\mathcal{F}$ in (3) to be the set of all linear regressions that take the token embeddings and output the token attribution score. Then, they solve the optimization problem in

$$W \in \underset{W \in \mathbb{R}^d, b \in \mathbb{R}}{\operatorname{argmin}} \sum_{(\boldsymbol{x},\boldsymbol{y}) \in \mathcal{D}_{\text{train}}} \sum_{j=1}^{|\boldsymbol{x}|} \|W^T e_j(\boldsymbol{x}) + b - \mathsf{MC}^n(\boldsymbol{x},\boldsymbol{y})_j\|_2^2, \tag{17}$$

and define the amortized explainer as $\mathsf{Amor}(\boldsymbol{x}) = (W^T e_1(\boldsymbol{x}) + b, ..., W^T e_{|\boldsymbol{x}|}(\boldsymbol{x}) + b)$.

We use stochastic amortization to produce amortized explainers for tabular datasets and Amortized Shap for LLMs to produce explainers for LLM predictions. Both explainers are trained using SVS-12 as MC^n.

C Selecting Coverage for a Given Inference Budget

Determining Coverage from Inference Budget: Providing explanations with initial guess increases the number of model inferences from 1 when using solely the amortized explainer to $n + 1$. However, a practitioner may have a budget of inferences, i.e., a maximum average number of inferences they are willing to perform to provide an explanation. We formalize the notion of inference budget in Definition 3.

Definition 3 (Inference Budget)**.** Denote by $\mathrm{N}(\mathsf{SE}(\boldsymbol{x},\boldsymbol{y}))$ the number of model inferences to produce the explanation $\mathsf{SE}(\boldsymbol{x},\boldsymbol{y})$. The inference budget $\mathrm{N}_{\texttt{budget}} \in \mathbb{N}$ is the maximum average number of inferences a practitioner is willing to perform per explanation, i.e., it is such that

$$\mathrm{N}_{\texttt{budget}} \geq \mathbb{E}\left[\mathrm{N}(\mathsf{SE}(\boldsymbol{x},\boldsymbol{y}))\right]. \tag{18}$$

Once an inference budget $\mathrm{N}_{\texttt{budget}}$ is defined, the coverage α should be set to follow it. In Proposition 1, we show the minimum coverage for the selective explanations to follow the inference budget.

Proposition 1 (Coverage for Inference Budget)**.** *Let $N_{\texttt{budget}} \geq 1$ be the inference budget, and assume that the Monte Carlo method $\mathsf{MC}^n(\boldsymbol{x},\boldsymbol{y})$ uses n model inferences. Then, the coverage level α should be chosen such that*

$$\frac{n + 1 - N_{\texttt{budget}}}{n} = \min_{\alpha \in [0,1]} \alpha, \text{ such that } \mathbb{E}\left[N(\mathsf{SE}(\boldsymbol{x},\boldsymbol{y}))\right] \leq N_{\texttt{budget}}. \tag{19}$$

Recall that SVS-m performs $n = 1 + dm$ inferences ($\boldsymbol{x} \in \mathbb{R}^d$), and KS-$m$ performs $n = m$ inferences.

D More Experimental Results

In this section, we (i) give further implementation details and (ii) discuss further empirical results.

D.1 More Details on Experimental Setup

High-Quality Explanations: We define the high-quality explanations for the tabular datasets to be given by Kernel Shap with as many inferences as needed for convergence, using the Shapley Regression library [4]. For the textual dataset, following [34], we define the high-quality explanations to be given by Kernel Shap using 8912 model inferences per explanation.

Amortized Explainers: For the tabular datasets, we use the amortized explainer from [6] that we describe in Section 2. Specifically, we use a multilayer perceptrom model architecture to learn the shapley values for the tabular datasets. For the textual datasets, we use the linear regression on token-level textual embeddings to learn the shapley values, as described in Section 2. Both amortized models learn from the training dataset of explanations generated using Shapley Value Sampling from the Captum library [17] with parameter 12, i.e., SVS-12.

Uncertainty Metrics: We test the two proposed uncertainty metrics in Section 3, namely, deep uncertainty and uncertainty learn. For **deep uncertainty**, we run the training pipeline for the amortized explainers 20 times for each dataset we perform experiments on, resulting in 20 different amortized explainer that we use to compute (4). For **uncertainty learn**, we use the multilayer perceptrom as the hypothesis class with only one hidden layer. The hidden layer was composed of $\kappa = 3d$ neurons where d is the dimension of the input vector $\boldsymbol{x} \in \mathbb{R}^d$. The uncertainty learn metric was trained on $\mathcal{D}_{\texttt{train}}$, the same training dataset as the amortized explainers.

Dataset sizes: We use 4000 samples from each dataset due to computational limitations on the computation of high-quality explanations used to evaluate selective explanations. All explanations were computed using the Captum library [17]. The dataset $\mathcal{D}$ with $N = 4000$ samples was partitioned in three parts, $\mathcal{D}_{\texttt{train}}$ with 50% of points, $\mathcal{D}_{\texttt{cal}}$ with 25% of points, and $\mathcal{D}_{\texttt{test}}$ with the other 25% of points.

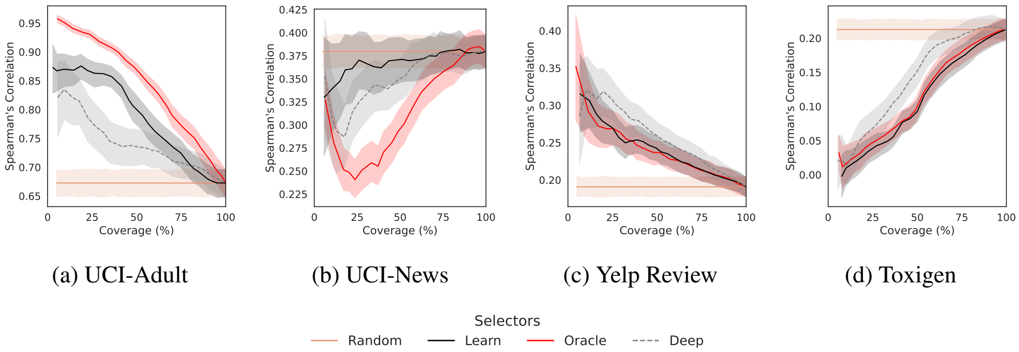
Computational Resources: All experiments were run in a A100 40 GB GPU. For each dataset, we compute different Monte Carlo explanations. For the UCI-News dataset, the high quality explanations took 4:30 hours to be generate until convergence while for UCI-Adult it took 3:46 hours. For the tabular datasets, all other Monte Carlo explainers were generated in less than 1 hour. For the language models, the high-quality explanations with 8192 model inferences, took 18:51 hours for the Toxigen dataset and 20:00 hours for the Yelp Review datasets. The other used Monte Carlo explanations took proportional (to the number of inferences) time to be generated.

D.2 Uncertainty Measures Impact on Spearman's Correlation

Figure 7 shows in the x-axis the coverage (α) and in the y-axis the average Spearman's correlation of the selected amortized explanations from high-quality explanations using deep uncertainty (with 20 models) and the uncertainty learn to select low-quality explanations. The Oracle[7] is computed by sorting examples by the smallest to higher MSE and computing the average Spearman's correlation in the bottom x-axis points accordingly to the MSE and is the best that can be done in terms of MSE.

Figure 7 shows that the Oracle and proposed uncertainty metrics don't always select the points with the smallest Spearman's correlation first. This implies that MSE and Spearman's correlation don't always align, i.e., there are points with high MSE and high Spearman's correlation at the same time. However, we note that the uncertainty learns selector can be applied to **any** metric ℓ as we define in (5) including Spearman's correlation and any combination of Spearman's correlation and MSE aiming to approximate both metrics. Moreover, when the smallest MSE aligns with the highest Spearman's correlation, i.e., the oracle is decreasing in Spearman's correlation when the coverage increases (Figure 7 (a) and (c)), the proposed uncertainty metrics also accurately detect the low-quality explanations in term of Spearman's correlation.

[7] The oracle is computationally expensive because it requires access to high-quality explanations.

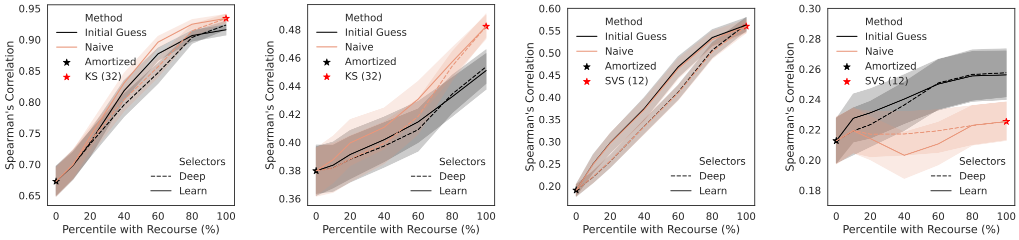


(a) UCI-Adult (b) UCI-News (c) Yelp Review (d) Toxigen

Fig. 7: Coverage vs. Spearman's correlation from the high-quality explanation. Coverage is the percentage of the points that the selection function predicts that will receive a higher-quality explanation, i.e., $\tau_t(\boldsymbol{x}) = 1$. When coverage is 100% Spearman's correlation is the average performance for the amortized explainer.

D.3 The Effect of Explanations with Initial Guess

In Figure 8 we compare explanations with initial guess (Definition 2) to only using the Monte Carlo to provide recourse to the low-quality explanaitons, i.e., $\lambda_h = 0$ we call it Naive. In all tested cases, Spearman's correlation of the Monte Carlo method is comparable to or larger than the amortized explainer. Although selective explanations optimized for MSE by using explanations with initial guess (Definition 2), we observe that the Spearman's correlation of selective explanations is close to or larger than the naive method, once again, demonstrating the efficacy of selective explanations.

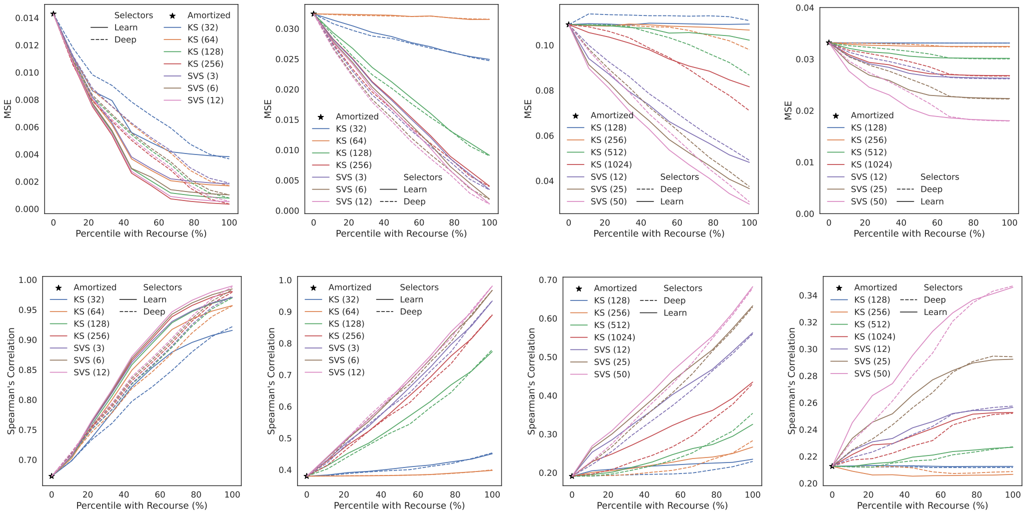


(a) UCI-Adult (b) UCI-News (c) Yelp Review (d) Toxigen

Fig. 8: Fraction of the population that receive explanations with initial guess (x-axis) vs. their Spearman's correlation from the high-quality explanations (y-axis). Naive uses $\lambda_h = 0$ while initial guess uses explanations with initial guess, i.e., when λ_h is given in (12).

D.4 Performance for Different Monte-Carlo Explainers

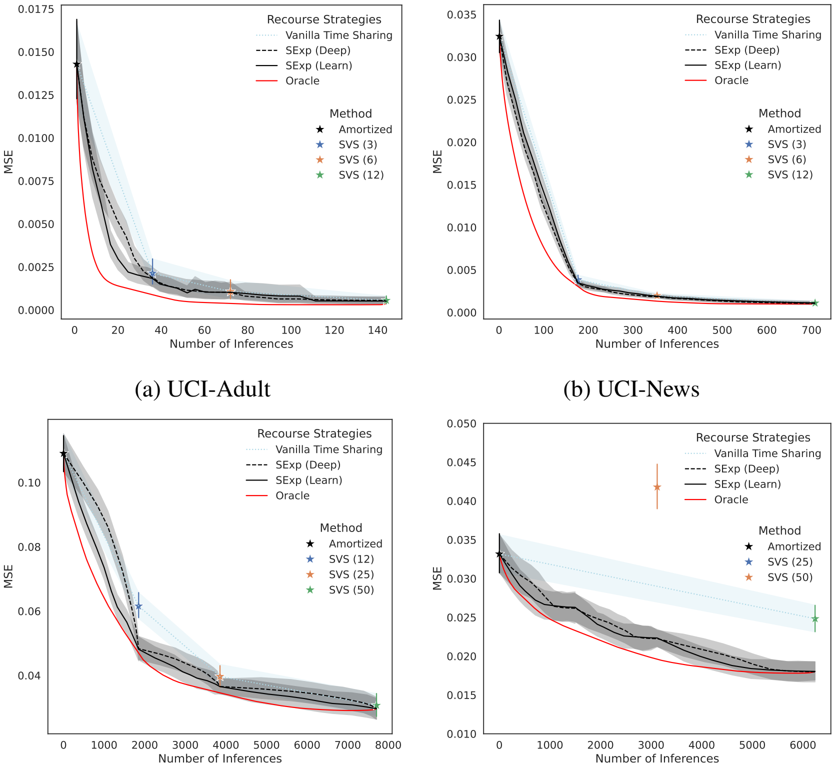
Figure 9 shows how the MSE and Spearman's correlation behave accordingly with the quality of the Monte Carlo explainer. We compare Kernel Shap and Shapley Value Sampling in all experiments. We observe that when the quality of the Monte Carlo explainer increases, the quality of the Selective explanation also increases, i.e., the MSE decreases and the Spearman's correlation increases. Moreover, we also observe diminishing returns, i.e., after a certain point, increasing the quality of the Monte Carlo explanations doesn't lead to a tailored increase in performance. For example, observe the SVS method in the tabular datasets Figure 9 (a) and (b). We also observe that providing explanations with initial guess has a high impact on both Spearman's correlation and MSE when only providing recourse toa small fraction of the population. For example, when providing explanations with initial guess for 20% of the population using SVS-12 in the Yelp Review dataset, Figure 9 (c), increases the Spearman's correlation in more than 50% (from 0.2 to more than 0.3).



(a) UCI-Adult (b) UCI-News (c) Yelp Review (d) Toxigen

Fig. 9: MSE (top) and Spearman's correlation (bottom) for selective explanations using different Monte Carlo explainers.

D.5 Time Sharing Using Selective Explanations



(a) UCI-Adult (b) UCI-News

(c) Yelp Review (d) Toxigen

Fig. 10: Number of model inferences (x-axis) vs. MSE (y-axis) using (i) vanilla time sharing, (ii) time sharing using selective explanations compared to (iii) the oracle when the MSE of the provided explanation is known.

We analyze how selective explanations can be used to improve the quality of Monte Carlo methods by time sharing between methods. When computing explanations using Monte Carlo methods, we perform n model inferences (x-axis in Figure 10) until a desired MSE (y-axis in Figure 10) is achieved. This is done by gradually increasing the number of inferences per points we generate explanations – this is displayed by the blue dotted curve in Figure 10 and we name it vanilla time sharing because the inferences (time) are shared gradually across points. We also compare it with the Oracle given the red curve in Figure 10 where, for each point, we compute Monte Carlo explanations using SVS with parameter 12, 25, and 50, compute their MSE to high-quality explanations and give the best explanation possible for a given number of inferences. Oracle is the best that can be done in terms of MSE vs. Number of Inferences only using Monte Carlo explanations. We compare both Orcle and vanilla time sharing with time sharing using selective explanations given by the black lines in Figure 10. For the time sharing using selective explanations, we also gradually increase the number of inferences but use selective explanations instead of plain Monte Carlo explanations.

Figure 10 shows that selective explanations closely approximate the Oracle curve, indicating the selective explanations have close to optimal trade-off between the number of model inferences and MSE. We highlight the performance of selective explanations in the Toxigen dataset. With only 1000 model inferences, we get better performance than using SVS-50 with about 6000 model inferences. We also note that in both LLMs, using selective explanations closely approximates the oracle and provides a better explanation with the same number of inferences than just using SVS.

E Proofs of Theoretical Results

Theorem 1 (Optimal λ_h)**.** *Let $0 = \alpha_1 < \alpha_2 < ... < \alpha_m = 1$ and define Q_i as in (9). Then, λ_i that solves the optimization problem in (11) is given by*

$$\lambda_i = \frac{\sum_{\substack{(\boldsymbol{x},\boldsymbol{y})\in\mathcal{D}_{val}\\ s_h(\boldsymbol{x})\in Q_i}} \langle \mathit{MC}^n(\boldsymbol{x},\boldsymbol{y}) - \mathit{MC}^{n'}(\boldsymbol{x},\boldsymbol{y}), \mathit{MC}^n(\boldsymbol{x},\boldsymbol{y}) - \mathit{Amor}(\boldsymbol{x},\boldsymbol{y})\rangle}{\sum_{\substack{(\boldsymbol{x},\boldsymbol{y})\in\mathcal{D}_{val}\\ s_h(\boldsymbol{x})\in Q_i}} ||\mathit{Amor}(\boldsymbol{x},\boldsymbol{y}) - \mathit{MC}^n(\boldsymbol{x},\boldsymbol{y})||_2^2}. \tag{20}$$

Proof. First, recall that

$$\lambda_i \triangleq \operatorname*{argmin}_{\lambda\in\mathbb{R}} \sum_{\substack{(\boldsymbol{x},\boldsymbol{y})\in\mathcal{D}_{\mathtt{val}}\\ s_h(\boldsymbol{x})\in Q_i}} \left|\left|\mathsf{SE}(\boldsymbol{x},\boldsymbol{y}) - \mathsf{MC}^{n'}(\boldsymbol{x},\boldsymbol{y})\right|\right|_2^2 \tag{21}$$

$$= \operatorname*{argmin}_{\lambda\in\mathbb{R}} \sum_{\substack{(\boldsymbol{x},\boldsymbol{y})\in\mathcal{D}_{\mathtt{val}}\\ s_h(\boldsymbol{x})\in Q_i}} \left|\left|\lambda\mathsf{Amor}(\boldsymbol{x},\boldsymbol{y}) + (1-\lambda)\mathsf{MC}^n(\boldsymbol{x},\boldsymbol{y}) - \mathsf{MC}^{n'}(\boldsymbol{x},\boldsymbol{y})\right|\right|_2^2. \tag{22}$$

Note that the function in (22) is convex in λ; therefore, if the derivative of it with respect to λ is zero, then the lambda that achieves the zero gradient is the minima. So, let's derivate (22) to find λ_i.

$$0 = \frac{d}{d\lambda} \sum_{\substack{(\boldsymbol{x},\boldsymbol{y})\in\mathcal{D}_{\mathtt{val}}\\ s_h(\boldsymbol{x})\in Q_i}} \left|\left|\lambda\mathsf{Amor}(\boldsymbol{x},\boldsymbol{y}) + (1-\lambda)\mathsf{MC}^n(\boldsymbol{x},\boldsymbol{y}) - \mathsf{MC}^{n'}(\boldsymbol{x},\boldsymbol{y})\right|\right|_2^2 \tag{23}$$

$$= 2\sum_{\substack{(\boldsymbol{x},\boldsymbol{y})\in\mathcal{D}_{\mathtt{val}}\\ s_h(\boldsymbol{x})\in Q_i}} \lambda||\mathsf{MC}^n(\boldsymbol{x},\boldsymbol{y}) - \mathsf{Amor}(\boldsymbol{x},\boldsymbol{y})||^2 \tag{24}$$

$$- 2\sum_{\substack{(\boldsymbol{x},\boldsymbol{y})\in\mathcal{D}_{\mathtt{val}}\\ s_h(\boldsymbol{x})\in Q_i}} \langle\mathsf{MC}^n(\boldsymbol{x},\boldsymbol{y}) - \mathsf{MC}^{n'}(\boldsymbol{x},\boldsymbol{y}), \mathsf{MC}^n(\boldsymbol{x},\boldsymbol{y}) - \mathsf{Amor}(\boldsymbol{x},\boldsymbol{y})\rangle \tag{25}$$

From (25) we conclude the proof by showing that

$$\lambda_i = \lambda = \frac{\sum_{\substack{(\boldsymbol{x},\boldsymbol{y})\in\mathcal{D}_{\mathtt{val}}\\ s_h(\boldsymbol{x})\in Q_i}} \langle\mathsf{MC}^n(\boldsymbol{x},\boldsymbol{y}) - \mathsf{MC}^{n'}(\boldsymbol{x},\boldsymbol{y}), \mathsf{MC}^n(\boldsymbol{x},\boldsymbol{y}) - \mathsf{Amor}(\boldsymbol{x},\boldsymbol{y})\rangle}{\sum_{\substack{(\boldsymbol{x},\boldsymbol{y})\in\mathcal{D}_{\mathtt{val}}\\ s_h(\boldsymbol{x})\in Q_i}} ||\mathsf{MC}^n(\boldsymbol{x},\boldsymbol{y}) - \mathsf{Amor}(\boldsymbol{x},\boldsymbol{y})||^2}. \tag{26}$$

□

Theorem 2 ($\lambda_i \approx \lambda_i^{\texttt{opt}}$)**.** *Let the Monte Carlo explanation used to provide recourse MC^n to be different enough from the amortized explainer, i.e., $\mathbb{E}\left[||\mathsf{MC}^n(X,Y) - \mathsf{Amor}(X,Y)||^2\right] = \mu > 0$. Also, assume that $\mathsf{MC}^{n'}$ is a good Monte Carlo approximation for the high-quality explainer HQ, i.e., $\mathbb{E}\left[||\mathsf{MC}^{n'}(X,Y) - \mathsf{HQ}(X,Y)||^2\right] = \mu^*$ for $\epsilon > \frac{\sqrt{5\mu^*}}{\mu}$. Recall that $\boldsymbol{x} \in \mathbb{R}^d$. If the explanations are bounded, i.e., $||\mathsf{MC}^n(\boldsymbol{x},\boldsymbol{y})||, ||\mathsf{Amor}(\boldsymbol{x},\boldsymbol{y})||, ||\mathsf{HQ}(\boldsymbol{x},\boldsymbol{y})| < Cd$ for some $C > 0$ then*

$$\Pr[|\lambda_i - \lambda_i^{opt}| > \epsilon] \leq e^{\frac{-\mu^2|Q_i|}{4Cd}} + e^{\frac{-\mu^4\epsilon^4|Q_i|}{400Cd}}, \tag{27}$$

where $|Q_i|$ is the number of points $\boldsymbol{x}$ in the validation dataset $\mathcal{D}_{\texttt{val}}$ that are in the bin Q_i.

Proof. Denote $|Q_i| = |\{(\boldsymbol{x},\boldsymbol{y}) \in \mathcal{D}_{\texttt{val}},\ \text{s.t. } s_h(\boldsymbol{x}) \in Q_i\}|$.

We start by showing that if $\mathbb{E}\left[||\mathsf{MC}^n(X,Y) - \mathsf{Amor}(X,Y)||^2\right] = \mu$ then

$$\Pr\left[\frac{1}{|Q_i|}\sum_{\substack{(\boldsymbol{x},\boldsymbol{y})\in\mathcal{D}_{\texttt{val}}\\ s_h(\boldsymbol{x})\in Q_i}} ||\mathsf{MC}^n(\boldsymbol{x},\boldsymbol{y}) - \mathsf{Amor}(\boldsymbol{x},\boldsymbol{y})||^2 \leq \frac{\mu}{2}\right] \tag{28}$$

$$= \Pr\left[\mu - \frac{1}{|Q_i|}\sum_{\substack{(\boldsymbol{x},\boldsymbol{y})\in\mathcal{D}_{\texttt{val}}\\ s_h(\boldsymbol{x})\in Q_i}} ||\mathsf{MC}^n(\boldsymbol{x},\boldsymbol{y}) - \mathsf{Amor}(\boldsymbol{x},\boldsymbol{y})||^2 \geq \frac{\mu}{2}\right] \tag{29}$$

$$\leq e^{\frac{-\mu^2|Q_i|}{4Cd}}. \tag{30}$$

Where the inequality in (30) follows from Hoeffding's inequality and the fact that:

$$||\mathsf{MC}^n(\boldsymbol{x},\boldsymbol{y}) - \mathsf{Amor}(\boldsymbol{x},\boldsymbol{y})||^2 \leq ||\mathsf{MC}^n(\boldsymbol{x},\boldsymbol{y})|| + ||\mathsf{Amor}(\boldsymbol{x},\boldsymbol{y})|| \leq 2Cd. \tag{31}$$

Second, we recall that $\mathbb{E}\left[||\mathsf{MC}^{n'}(X,Y) - \mathsf{HQ}(X,Y)||^2\right] = \mu^* \leq \frac{\mu^2\epsilon^2}{5}$. Then, we have that

$$\Pr\left[\frac{1}{|Q_i|}\sum_{\substack{(\boldsymbol{x},\boldsymbol{y})\in\mathcal{D}_{\texttt{val}}\\ s_h(\boldsymbol{x})\in Q_i}} ||\mathsf{HQ}(\boldsymbol{x},\boldsymbol{y}) - \mathsf{Amor}(\boldsymbol{x},\boldsymbol{y})||^2 \geq \epsilon^2\frac{\mu^2}{4}\right] \tag{32}$$

$$= \Pr\left[\frac{1}{|Q_i|}\sum_{\substack{(\boldsymbol{x},\boldsymbol{y})\in\mathcal{D}_{\texttt{val}}\\ s_h(\boldsymbol{x})\in Q_i}} ||\mathsf{HQ}(\boldsymbol{x},\boldsymbol{y}) - \mathsf{Amor}(\boldsymbol{x},\boldsymbol{y})||^2 - \mu^* \geq \epsilon^2\frac{\mu^2}{4} - \mu^*\right] \tag{33}$$

$$\leq \Pr\left[\frac{1}{|Q_i|}\sum_{\substack{(\boldsymbol{x},\boldsymbol{y})\in\mathcal{D}_{\texttt{val}}\\ s_h(\boldsymbol{x})\in Q_i}} ||\mathsf{HQ}(\boldsymbol{x},\boldsymbol{y}) - \mathsf{Amor}(\boldsymbol{x},\boldsymbol{y})||^2 - \mu^* \geq \epsilon^2\frac{\mu^2}{20}\right] \tag{34}$$

$$\leq e^{\frac{-\mu^4\epsilon^4|Q_i|}{400Cd}}. \tag{35}$$

Where the inequality in (35) follows from Hoeffding's inequality and the fact that:

$$||\mathsf{HQ}(\boldsymbol{x},\boldsymbol{y}) - \mathsf{Amor}(\boldsymbol{x},\boldsymbol{y})||^2 \leq ||\mathsf{HQ}(\boldsymbol{x},\boldsymbol{y})|| + ||\mathsf{Amor}(\boldsymbol{x},\boldsymbol{y})|| \leq 2Cd. \tag{36}$$

Third, notice by directly applying Theorem 1 and replacing the Monte Carlo explanation by the high-quality explanation, we have that

$$\lambda_i^{\texttt{opt}} = \frac{\sum_{\substack{(\boldsymbol{x},\boldsymbol{y})\in\mathcal{D}_{\texttt{val}}\\ s_h(\boldsymbol{x})\in Q_i}} \langle\mathsf{MC}^n(\boldsymbol{x},\boldsymbol{y}) - \mathsf{HQ}(\boldsymbol{x},\boldsymbol{y}), \mathsf{MC}^n(\boldsymbol{x},\boldsymbol{y}) - \mathsf{Amor}(\boldsymbol{x},\boldsymbol{y})\rangle}{\sum_{\substack{(\boldsymbol{x},\boldsymbol{y})\in\mathcal{D}_{\texttt{val}}\\ s_h(\boldsymbol{x})\in Q_i}} ||\mathsf{MC}^n(\boldsymbol{x},\boldsymbol{y}) - \mathsf{Amor}(\boldsymbol{x},\boldsymbol{y})||^2}. \tag{37}$$

Hence, we can write $\lambda_i^{\mathtt{opt}} - \lambda_i$ as

$$|\lambda_i^{\mathtt{opt}} - \lambda_i| \tag{38}$$

$$= \left| \frac{\sum_{\substack{(\boldsymbol{x},\boldsymbol{y})\in\mathcal{D}_{\mathtt{val}}\\ s_h(\boldsymbol{x})\in Q_i}} \langle \mathsf{MC}^{n'}(\boldsymbol{x},\boldsymbol{y}) - \mathsf{HQ}(\boldsymbol{x},\boldsymbol{y}), \mathsf{MC}^{n}(\boldsymbol{x},\boldsymbol{y}) - \mathsf{Amor}(\boldsymbol{x},\boldsymbol{y})\rangle}{\sum_{\substack{(\boldsymbol{x},\boldsymbol{y})\in\mathcal{D}_{\mathtt{val}}\\ s_h(\boldsymbol{x})\in Q_i}} ||\mathsf{MC}^{n}(\boldsymbol{x},\boldsymbol{y}) - \mathsf{Amor}(\boldsymbol{x},\boldsymbol{y})||^2} \right| \tag{39}$$

$$\le \frac{\left(\sum_{\substack{(\boldsymbol{x},\boldsymbol{y})\in\mathcal{D}_{\mathtt{val}}\\ s_h(\boldsymbol{x})\in Q_i}} ||\mathsf{MC}^{n'}(\boldsymbol{x},\boldsymbol{y}) - \mathsf{HQ}(\boldsymbol{x},\boldsymbol{y})||_2^2 ||\mathsf{MC}^{n}(\boldsymbol{x},\boldsymbol{y}) - \mathsf{Amor}(\boldsymbol{x},\boldsymbol{y})||_2^2\right)^{1/2}}{\sum_{\substack{(\boldsymbol{x},\boldsymbol{y})\in\mathcal{D}_{\mathtt{val}}\\ s_h(\boldsymbol{x})\in Q_i}} ||\mathsf{MC}^{n}(\boldsymbol{x},\boldsymbol{y}) - \mathsf{Amor}(\boldsymbol{x},\boldsymbol{y})||^2}, \tag{40}$$

where the last inequality (40) comes from the Cauchy–Schwarz inequality. Denote the denominator in (40) by Δ, i.e.,

$$\sum_{\substack{(\boldsymbol{x},\boldsymbol{y})\in\mathcal{D}_{\mathtt{val}}\\ s_h(\boldsymbol{x})\in Q_i}} ||\mathsf{MC}^{n}(\boldsymbol{x},\boldsymbol{y}) - \mathsf{Amor}(\boldsymbol{x},\boldsymbol{y})||^2 = \Delta.$$

Lastly, notice that $\mathsf{MC}^{n'}(\boldsymbol{x},\boldsymbol{y})$ is sampled independently of $\mathsf{MC}^{n}(\boldsymbol{x},\boldsymbol{y})$ and that $\mathsf{HQ}(\boldsymbol{x},\boldsymbol{y})$ is deterministic. Therefore:

$$\Pr[|\lambda_i^{\mathtt{opt}} - \lambda_i| \ge \epsilon] \tag{41}$$

$$\le \Pr\left[\frac{\left(\sum_{\substack{(\boldsymbol{x},\boldsymbol{y})\in\mathcal{D}_{\mathtt{val}}\\ s_h(\boldsymbol{x})\in Q_i}} ||\mathsf{MC}^{n'}(\boldsymbol{x},\boldsymbol{y}) - \mathsf{HQ}(\boldsymbol{x},\boldsymbol{y})||_2^2 ||\mathsf{MC}^{n}(\boldsymbol{x},\boldsymbol{y}) - \mathsf{Amor}(\boldsymbol{x},\boldsymbol{y})||_2^2\right)^{1/2}}{\sum_{\substack{(\boldsymbol{x},\boldsymbol{y})\in\mathcal{D}_{\mathtt{val}}\\ s_h(\boldsymbol{x})\in Q_i}} ||\mathsf{MC}^{n}(\boldsymbol{x},\boldsymbol{y}) - \mathsf{Amor}(\boldsymbol{x},\boldsymbol{y})||^2} \ge \epsilon \right] \tag{42}$$

$$\le \Pr\left[\frac{\sum_{\substack{(\boldsymbol{x},\boldsymbol{y})\in\mathcal{D}_{\mathtt{val}}\\ s_h(\boldsymbol{x})\in Q_i}} ||\mathsf{MC}^{n'}(\boldsymbol{x},\boldsymbol{y}) - \mathsf{HQ}(\boldsymbol{x},\boldsymbol{y})||_2^2 ||\mathsf{MC}^{n}(\boldsymbol{x},\boldsymbol{y}) - \mathsf{Amor}(\boldsymbol{x},\boldsymbol{y})||_2^2}{\Delta^2} \ge \epsilon^2 \right] \tag{43}$$

$$\begin{aligned}
&\le \Pr\left[\frac{\sum_{\substack{(\boldsymbol{x},\boldsymbol{y})\in\mathcal{D}_{\mathtt{val}}\\ s_h(\boldsymbol{x})\in Q_i}} ||\mathsf{MC}^{n'}(\boldsymbol{x},\boldsymbol{y}) - \mathsf{HQ}(\boldsymbol{x},\boldsymbol{y})||_2^2 ||\mathsf{MC}^{n}(\boldsymbol{x},\boldsymbol{y}) - \mathsf{Amor}(\boldsymbol{x},\boldsymbol{y})||_2^2}{\Delta^2} \ge \epsilon^2 \,\middle|\, \Delta \le \frac{\mu}{2} \right] \\
&\quad \times \Pr\left[\Delta \le \frac{\mu}{2}\right] \\
&+ \Pr\left[\frac{\sum_{\substack{(\boldsymbol{x},\boldsymbol{y})\in\mathcal{D}_{\mathtt{val}}\\ s_h(\boldsymbol{x})\in Q_i}} ||\mathsf{MC}^{n'}(\boldsymbol{x},\boldsymbol{y}) - \mathsf{HQ}(\boldsymbol{x},\boldsymbol{y})||_2^2 ||\mathsf{MC}^{n}(\boldsymbol{x},\boldsymbol{y}) - \mathsf{Amor}(\boldsymbol{x},\boldsymbol{y})||_2^2}{\Delta^2} \ge \epsilon^2 \,\middle|\, \Delta > \frac{\mu}{2} \right] \\
&\quad \times \Pr\left[\Delta > \frac{\mu}{2}\right]
\end{aligned} \tag{44}$$

$$\begin{aligned}
&\le \Pr\left[\sum_{\substack{(\boldsymbol{x},\boldsymbol{y})\in\mathcal{D}_{\mathtt{val}}\\ s_h(\boldsymbol{x})\in Q_i}} ||\mathsf{MC}^{n'}(\boldsymbol{x},\boldsymbol{y}) - \mathsf{HQ}(\boldsymbol{x},\boldsymbol{y})||_2^2 ||\mathsf{MC}^{n}(\boldsymbol{x},\boldsymbol{y}) - \mathsf{Amor}(\boldsymbol{x},\boldsymbol{y})||_2^2 \ge \epsilon^2 \frac{\mu^2}{4} \right] \\
&\quad + \Pr\left[\Delta \le \frac{\mu}{2}\right]
\end{aligned} \tag{45}$$

$$\le e^{\frac{-\mu^2|Q_i|}{4Cd}} + e^{\frac{-\mu^4\epsilon^4|Q_i|}{400Cd}}. \tag{46}$$

Where the inequality in (42) is a direct application of 40, the inequality in (44) comes from simply conditioning, the inequality in (45) comes from the fact that probabilities are bounded by one getting rid of the first term in (45) (first out of lines) and the fourth term in (45) (forth out of lines) and the fact that $\mathsf{MC}^{n'}(\boldsymbol{x},\boldsymbol{y})$ is sampled independently of $\mathsf{MC}^{n}(\boldsymbol{x},\boldsymbol{y})$ and that $\mathsf{HQ}(\boldsymbol{x},\boldsymbol{y})$ is deterministic. Finally, the last inequality in (46) comes from applying (30) and (35).

Hence, from (46), we conclude that

$$\Pr[|\lambda_i^{\mathtt{opt}} - \lambda_i| \geq \epsilon] \leq e^{\frac{-\mu^2|Q_i|}{4Cd}} + e^{\frac{-\mu^4\epsilon^4|Q_i|}{400Cd}}. \tag{47}$$

□

Proposition 2 (Coverage for Inference Budget)**.** *Let $N_{\mathtt{budget}} \geq 1$ be the set inference budget, and assume that the Monte Carlo method $\mathsf{MC}^n(\boldsymbol{x},\boldsymbol{y})$ uses n model inferences. Then, the coverage level α should be chosen such that*

$$\operatorname*{argmin}_{\alpha\in[0,1]} \left\{\mathbb{E}\left[N(\mathsf{SE}(\boldsymbol{x},\boldsymbol{y}))\right] \leq N_{\mathtt{budget}}\right\} = \frac{n+1-N_{\mathtt{budget}}}{n}. \tag{48}$$

Recall that Shapley Value Sampling with parameter m performs $1+dm$ inferences ($\boldsymbol{x}\in\mathbb{R}^d$), and Kernel Shap with parameter m performs m inferences.

Proof. Let $\alpha \in [0,1]$, then an α portion of examples receive explanations from the amortized explainer, i.e., they receive one inference, and $1-\alpha$ portion of examples receive explanations with initial guess, i.e., n model inferences. Therefore, the expected number of model inferences per instance is given by (49).

$$\mathbb{E}\left[\mathrm{N}(\mathsf{SE}(\boldsymbol{x},\boldsymbol{y}))\right] = \alpha + (1-\alpha)(n+1) \tag{49}$$

In order for the inference budget to be followed, it is necessary that

$$\mathbb{E}\left[\mathrm{N}(\mathsf{SE}(\boldsymbol{x},\boldsymbol{y}))\right] = \alpha + (1-\alpha)(n+1) \leq \mathrm{N}_{\mathtt{budget}}. \tag{50}$$

From (50), we conclude that:

$$\alpha \geq \frac{n+1-\mathrm{N}_{\mathtt{budget}}}{n}, \tag{51}$$

Hence,

$$\mathrm{argmin}\,\alpha \in [0,1]\left\{\mathbb{E}\left[\mathrm{N}(\mathsf{SE}(\boldsymbol{x},\boldsymbol{y}))\right] \leq \mathrm{N}_{\mathtt{budget}}\right\} = \frac{n+1-\mathrm{N}_{\mathtt{budget}}}{n}. \tag{52}$$

□