
Improving Perturbation-based Explanations by Understanding the Role of Uncertainty Calibration:

Appendix

Thomas Decker^{1,2,3} Volker Tresp^{2,3} Florian Buettner^{4,5,6}

¹Siemens AG ²LMU Munich ³Munich Center for Machine Learning (MCML)

⁴Goethe University Frankfurt ⁵German Cancer Research Center (DKFZ)

⁶German Cancer Consortium (DKTK)

thomas.decker@siemens.com, volker.tresp@lmu.de, florian.buettner@dkfz.de

A Theoretical Proofs

A.1 Proofs of Results in the Main Paper

In this section, we provide the theoretical proofs of all statements in the main paper. We start by introducing several existing results on which we built on:

Theorem A.1 (Relationship between KL-Divergence, Cross-Entropy and Entropy). *Let P and Q be probability distributions over a common probability space $(\mathcal{X}, \mathcal{F})$. The Kullback-Leibler (KL) divergence between P and Q is defined as:*

$$D_{KL}(P||Q) = \mathbb{E}_{X \sim P} \left[\log \frac{P(X)}{Q(X)} \right].$$

This can be decomposed in terms of the entropy and the cross-entropy as follows:

$$D_{KL}(P||Q) = H(P, Q) - H(P),$$

where:

- $H(P)$ is the entropy of P , defined as:

$$H(P) = -\mathbb{E}_{X \sim P} [\log P(X)] = -\sum_{x \in \mathcal{X}} P(x) \log P(x).$$

- $H(P, Q)$ is the cross-entropy between P and Q , given by:

$$H(P, Q) = -\mathbb{E}_{X \sim P} [\log Q(X)] = -\sum_{x \in \mathcal{X}} P(x) \log Q(x).$$

Theorem A.2 (Hoeffding's inequality). *Let X_1, X_2, \dots, X_n be independent random variables such that for all i , it holds that $X_i \in [a_i, b_i]$ almost surely. Define the empirical mean as $\bar{X} = \frac{1}{n} \sum_{i=1}^n X_i$, then, for any $t > 0$:*

$$P(\bar{X} - \mathbb{E}[\bar{X}] \geq t) \leq \exp \left(-\frac{2nt^2}{\sum_{i=1}^n (b_i - a_i)^2} \right).$$

Hoeffding's inequality provides an upper bound on the probability that the sum of bounded independent random variables deviates from its expectation.

Theorem A.3 (Pinsker's inequality). *Given two probability distributions P and Q over the same sample space, the total variation distance is defined as:*

$$D_{TV}(P, Q) = \sup_A |P(A) - Q(A)|.$$

For measurable sets $A \in \mathcal{F}$. Then it holds:

$$D_{TV}(P, Q) \leq \sqrt{\frac{1}{2} D_{KL}(P \| Q)}.$$

Pinsker's inequality provides a lower bound on the total variation distance between two probability distributions in terms of the Kullback-Leibler (KL) divergence.

Theorem A.4 (Operator norm of bounded summaries [16]). *Let $A \in \mathbb{R}^{d \times 2^d}$ be a matrix satisfying the boundedness property, such that for all $v \in \mathbb{R}^{2^d}$, the attributions are bounded by each feature's smallest and largest contributions:*

$$\min_{i \notin S} (v_{S \cup \{i\}} - v_S) \leq (Av)_i \leq \max_{i \notin S} (v_{S \cup \{i\}} - v_S), \quad \forall i \in [d].$$

Then, the operator norm of A with respect to the $\ell_1 \rightarrow \ell_\infty$ norms is given by:

$$\|A\|_{1, \infty} = 2\sqrt{d}.$$

Note that many popular perturbation-based methods techniques have linear summary techniques satisfying the boundedness property and theorem A.4 [16], such as Shapley Values [19] and LIME [21]. Now we can turn to the proofs of the theorems in the main paper:

Theorem 3.2. Let \mathcal{L}_{CE} be the cross-entropy loss, $D_{KL}(\cdot, \cdot)$ be the KL-Divergence between two distributions, and let $I(\cdot, \cdot)$ denote the mutual information between random variables. Then we have:

$$v_f^\pi(S) = \underbrace{D_{KL}(P_Y \| f_\emptyset^\pi(X))}_{\text{Perturbation Baseline Bias}} + \underbrace{I(f_S^\pi(X), Y)}_{\text{Information in } f_S^\pi \text{ about } Y} - \underbrace{CE_{KL}(f_S^\pi)}_{\text{Calibration Error of } f_S^\pi}$$

Proof. By definition we have

$$v_f^\pi(S) = \mathbb{E}[\mathcal{L}_{CE}(f_\emptyset^\pi(X), Y)] - \mathbb{E}[\mathcal{L}_{CE}(f_S^\pi(X), Y)]$$

Using the notation from Theorem A1, the second term can be rewritten in the following way:

$$\begin{aligned} \mathbb{E}[\mathcal{L}_{CE}(f_S^\pi(X), Y)] &= \mathbb{E}_{X, Y}[-\log(f_S^\pi(X)_Y)] = \mathbb{E}_{f_S^\pi(X), Y}[-\log(f_S^\pi(X)_Y)] \\ &= \mathbb{E}_{f_S^\pi(X)} \mathbb{E}_{Y|f_S^\pi(X)}[-\log(f_S^\pi(X)_Y)] = \mathbb{E}_{f_S^\pi(X)} [H(P_{Y|f_S^\pi(X)}, f_S^\pi(X))] \\ &= \mathbb{E}_{f_S^\pi(X)} [H(P_{Y|f_S^\pi(X)}, f_S^\pi(X))] + \mathbb{E}_{f_S^\pi(X)} [H(P_{Y|f_S^\pi(X)})] \\ &\quad - \mathbb{E}_{f_S^\pi(X)} [H(P_{Y|f_S^\pi(X)})] \\ &= D_{KL}(P_{Y|f_S^\pi(X)} \| f_S^\pi(X)) + \mathbb{E}_{f_S^\pi(X)} [H(P_{Y|f_S^\pi(X)})] \\ &= CE_{KL}(f_S^\pi) + \mathbb{E}_{f_S^\pi(X)} [H(P_{Y|f_S^\pi(X)})] + H(P_Y) - H(P_Y) \end{aligned}$$

Using the tower rule of the conditional expectation we can further reformulate $H(P_Y)$ as follows:

$$\begin{aligned} -H(P_Y) &= \mathbb{E}_Y[\log(P_Y)] = \mathbb{E}_{f_S^\pi(X)} \mathbb{E}_{Y|f_S^\pi(X)}[\log(P_Y)] \\ &= -\mathbb{E}_{f_S^\pi(X)} [H(P_{Y|f_S^\pi(X)}, P_Y)] \end{aligned}$$

Plugging this into the last expression above gives:

$$\begin{aligned} \mathbb{E}[\mathcal{L}_{CE}(f_S^\pi(X), Y)] &= CE_{KL}(f_S^\pi) + \mathbb{E}_{f_S^\pi(X)} [H(P_{Y|f_S^\pi(X)})] + H(P_Y) - H(P_Y) \\ &= CE_{KL}(f_S^\pi) + \mathbb{E}_{f_S^\pi(X)} [H(P_{Y|f_S^\pi(X)})] - \mathbb{E}_{f_S^\pi(X)} [H(P_{Y|f_S^\pi(X)}, P_Y)] + H(P_Y) \\ &= CE_{KL}(f_S^\pi) - \mathbb{E}_{f_S^\pi(X)} [D_{KL}(P_{Y|f_S^\pi(X)} \| P_Y)] + H(P_Y) \\ &= CE_{KL}(f_S^\pi) - I(f_S^\pi(X), Y) + H(P_Y) \end{aligned}$$

Integrating this into the definition of the predictive power finally yields:

$$\begin{aligned} v_f^\pi(S) &= \mathbb{E}[\mathcal{L}_{CE}(f_\emptyset^\pi(X), Y)] - H(P_Y) + I(f_S^\pi(X), Y) - CE_{KL}(f_S^\pi) \\ &= H(P_Y, f_\emptyset^\pi(X)) - H(P_Y) + I(f_S^\pi(X), Y) - CE_{KL}(f_S^\pi) \\ &= D_{KL}(P_Y \| f_\emptyset^\pi(X)) + I(f_S^\pi(X), Y) - CE_{KL}(f_S^\pi) \end{aligned}$$

□

Corollary 3.3. If a model f is perfectly calibrated under all subset perturbations faced during the explanation process, then we have:

$$v_f^\pi(S) = I(f_S^\pi(X), Y)$$

Proof. If a model f is perfectly calibrated under all subset perturbations faced during the explanation process, then we trivially have $CE_{KL}(f_S^\pi) = 0$ for all $S \subset \{1, \dots, d\}$. Therefore the third term on the right hand side of Theorem 3.2 vanishes. Moreover, $f_\emptyset^\pi(X)$ results in a constant prediction and the only constant prediction that is perfectly calibrated is P_Y . Hence, also the first term vanishes since the KL-Divergence between two identical distributions is zero. \square

Theorem 3.4. Let $\phi(x)$ be a local explanation for input x with respect to a class-wise prediction $f(x)_k$ and perturbation π . Let $\phi^*(x)$ denote the explanation that would be obtained if the model f were perfectly calibrated under all subset perturbations. Define the maximum calibration error across all perturbation subsets as: $CE_{KL}^{\max_S} := \max_{S \subset [d]} CE_{KL}(f_S^\pi)$. Then, with probability of at least $(1 - \delta)$, the mean squared difference between the actual and ideal explanations is bounded by:

$$\frac{1}{d} \|\phi(x) - \phi^*(x)\|_2^2 \leq 2CE_{KL}^{\max_S} + \sqrt{8 \log(1/\delta)}$$

Proof.

$$\begin{aligned} \|\phi(x) - \phi^*(x)\|_2 &= \|A\vartheta(x) - A\vartheta^*(x)\|_2 \\ &\leq \|A\|_{1,\infty} \|\vartheta(x) - \vartheta^*(x)\|_\infty \\ &= 2\sqrt{d} \max_{i \in \{1, \dots, 2^d\}} |\vartheta_i(x) - \vartheta_i^*(x)| \\ &= 2\sqrt{d} \max_{S \subset \{1, \dots, d\}} |f_S^\pi(x)_k - f_S^{*,\pi}(x)_k| \\ &\leq 2\sqrt{d} \max_{S \subset \{1, \dots, d\}} D_{TV}(f_S^\pi(x), P_{Y|f_S^\pi(x)}) \end{aligned}$$

Note that we used Theorem A4 to reformulate the operator norm of the linear summary matrix A , which we expect to satisfy the boundedness property (see Theorem A4).

To ease the notation, denote $Z := D_{TV}(f_S^\pi(X), P_{Y|f_S^\pi(X)})^2$ such that $Z \in [0, 1]$ as the total variation distance is bounded in that range. Then we can apply Hoeffding's inequality (Theorem A2) to bound the probability that Z will exceed its expectation by t , since:

$$P(Z > t + \mathbb{E}[Z]) \leq \exp(-2t^2)$$

Consequently, by choosing $t = \sqrt{1/2 \log(1/\delta)}$ we get that:

$$P(Z > \sqrt{1/2 \log(1/\delta)} + \mathbb{E}[Z]) \leq \delta$$

Therefore it holds with probability of at least $(1 - \delta)$ that:

$$\begin{aligned} \frac{1}{d} \|\phi(x) - \phi^*(x)\|_2^2 &\leq 4 \max_{S \subset \{1, \dots, d\}} D_{TV}(f_S^\pi(X), P_{Y|f_S^\pi(X)})^2 \\ &\leq 4 \max_{S \subset \{1, \dots, d\}} \mathbb{E}[D_{TV}(f_S^\pi(X), P_{Y|f_S^\pi(X)})^2] + \sqrt{8 \log(1/\delta)} \\ &\leq 4 \max_{S \subset \{1, \dots, d\}} \mathbb{E}\left[\frac{1}{2} D_{KL}(f_S^\pi(X), P_{Y|f_S^\pi(X)})\right] + \sqrt{8 \log(1/\delta)} \\ &= 2CE_{KL}^{\max_S} + \sqrt{8 \log(1/\delta)} \end{aligned}$$

The last inequality utilizes Pinker's inequality (Theorem A3) to bound the total variation distance in terms of the KL-Divergence. \square

Proposition 4.2 Let $\mathcal{T} : [0, 1]^K \times 2^{\{1, \dots, d\}} \rightarrow [0, 1]^K$ be a deterministic and componentwise strictly monotonic function for every fixed S . Then \mathcal{T} is information-preserving.

Proof. In general, the data processing inequality [4] implies that for every deterministic function \mathcal{T} it holds:

$$I(Y, \mathcal{T}(f_S^\pi(X), S)) \leq I(Y, f_S^\pi(X))$$

and equality holds under the conditional independence assumption that:

$$Y \perp \mathcal{T}(f_S^\pi(X), S) \mid f_S^\pi(X), S$$

which can also be expressed as:

$$P(Y \mid \mathcal{T}(f_S^\pi(X), S), f_S^\pi(X), S) = P(Y \mid f_S^\pi(X), S)$$

Since calibration here can also explicitly depend on the properties of S , this condition intuitively requires that the mere knowledge about S should not encode additional information about the target Y . So in general, any \mathcal{T} that satisfies this property is information-preserving as defined in Definition 4.1. A practical way to obtain appropriate recalibration maps is to consider functions that are strictly monotonic increasing [28] for every fixed Subset S . Since such maps are also injective for every S we get:

$$\forall s_1, s_2 \in [0, 1]^K, S \in 2^{\{1, \dots, d\}} : s_1 \neq s_2 \implies \mathcal{T}(s_1, S) \neq \mathcal{T}(s_2, S)$$

This injectivity ensures that for any fixed S , there exists a one-to-one correspondence between $f_S^\pi(X)$ and $\mathcal{T}(f_S^\pi(X), S)$. Therefore:

$$P(Y = y \mid \mathcal{T}(f_S^\pi(X), S) = t, f_S^\pi(X) = f, S = s) = \frac{P(Y = y, \mathcal{T}(f_S^\pi(X), S) = t, f_S^\pi(X) = f, S = s)}{P(\mathcal{T}(f_S^\pi(X), S) = t, f_S^\pi(X) = f, S = s)}$$

and the joint events in the numerator and denominator have non-zero probability only when $t = \mathcal{T}(f, s)$ due to their deterministic functional relationship. Hence, we can write: and the joint events in the numerator and denominator have non-zero probability only when $t = \mathcal{T}(f, s)$ due to their deterministic functional relationship. Hence, we can write:

$$\begin{aligned} P(Y = y, \mathcal{T}(f_S^\pi(X), S) = t, f_S^\pi(X) = f, S = s) &= P(Y = y, f_S^\pi(X) = f, S = s) \cdot \mathbf{1}_{\{t = \mathcal{T}(f, s)\}} \\ P(\mathcal{T}(f_S^\pi(X), S) = t, f_S^\pi(X) = f, S = s) &= P(f_S^\pi(X) = f, S = s) \cdot \mathbf{1}_{\{t = \mathcal{T}(f, s)\}} \end{aligned}$$

Substituting these expressions back into our original equation:

$$\begin{aligned} P(Y = y \mid \mathcal{T}(f_S^\pi(X), S) = t, f_S^\pi(X) = f, S = s) &= \frac{P(Y = y, f_S^\pi(X) = f, S = s) \cdot \mathbf{1}_{\{t = \mathcal{T}(f, s)\}}}{P(f_S^\pi(X) = f, S = s) \cdot \mathbf{1}_{\{t = \mathcal{T}(f, s)\}}} \\ &= \frac{P(Y = y, f_S^\pi(X) = f, S = s)}{P(f_S^\pi(X) = f, S = s)} \\ &= P(Y = y \mid f_S^\pi(X) = f, S = s) \end{aligned}$$

which shows the proposition. Also note that [28] has also proposed such kinds of recalibration methods and showed that they are also accuracy preserving. \square

A.2 Extension of Theorem 3.4 for non-linear aggregation methods

Theorem 3.4 can naturally be extended to non-linear aggregation methods. For any general non-linear aggregation function G that maps model predictions under perturbations to feature importance vectors, we can reformulate the bound using a more general Lipschitz property. Let G be a non-linear aggregation strategy, and let $\nu(x)$ and $\nu^*(x)$ represent the vectors of model predictions under subset perturbations for the uncalibrated and perfectly calibrated models, respectively. All we additionally need for the non-linear case is the existence of a constant L such that:

$$\|G(\nu(x)) - G(\nu^*(x))\|_2 \leq L\|\nu(x) - \nu^*(x)\|_\infty$$

This leads to the following more general version of Theorem 3.4:

$$\frac{1}{d}\|\phi(x) - \phi^*(x)\|_2^2 \leq \frac{L^2}{2d}CE_{KL}^{\max_S} + \frac{L^2}{d}\sqrt{1/2\log(1/\delta)}$$

Proof.

$$\begin{aligned} \|\phi(x) - \phi^*(x)\|_2 &= \|G(\nu(x)) - G(\nu^*(x))\|_2 \\ &\leq L\|\nu(x) - \nu^*(x)\|_\infty \\ &= L \max_{S \subset \{1, \dots, d\}} |f_S^\pi(x)_k - f_S^{*,\pi}(x)_k| \\ &\leq L \max_{S \subset \{1, \dots, d\}} D_{TV}(f_S^\pi(x), P_{Y|f_S^\pi(x)}) \end{aligned}$$

Squaring both sides and dividing by d :

$$\frac{1}{d}\|\phi(x) - \phi^*(x)\|_2^2 \leq \frac{L^2}{d} \max_{S \subset \{1, \dots, d\}} D_{TV}(f_S^\pi(X), P_{Y|f_S^\pi(X)})^2$$

Applying Hoeffding's inequality and Pinsker's inequality as in Theorem 3.4, with probability at least $(1 - \delta)$:

$$\begin{aligned} \frac{1}{d}\|\phi(x) - \phi^*(x)\|_2^2 &\leq \frac{L^2}{d} \left[\mathbb{E}[D_{TV}(f_S^\pi(X), P_{Y|f_S^\pi(X)})^2] + \sqrt{1/2\log(1/\delta)} \right] \\ &\leq \frac{L^2}{d} \left[\frac{1}{2} \mathbb{E}[D_{KL}(f_S^\pi(X), P_{Y|f_S^\pi(X)})] + \sqrt{1/2\log(1/\delta)} \right] \\ &= \frac{L^2}{2d}CE_{KL}^{\max_S} + \frac{L^2}{d}\sqrt{1/2\log(1/\delta)} \end{aligned}$$

□

For linear aggregation strategies used by methods like SHAP and LIME, one can show that $L = 2\sqrt{d}$ [16], which yields the specific bound presented in Theorem 3.4 of the paper. This demonstrates that our theoretical framework can be generalized to perturbation-based explanation methods based on non-linear aggregation as well.

B Experimental Details

All numerical experiments have been computed on an Nvidia RTX A5000 GPU with CUDA 12.2 and two physical AMD EPYC 7502P 32-Core CPUs running on Linux Ubuntu.

B.1 Dataset and Model Details

Tabular Datasets All tabular datasets are downloaded from [7] and are utilized without any further preprocessing. Across all experiments, we considered train/validation/test splits of 60/10/30.

Tabular Models We considered a Multi-Layer-Perception and a ResNet for tabular classification based on the architecture proposed in [6]. The MLP models use 3 hidden layers of dimension 64 with ReLU activation and dropout (0.2) for regularization, while ResNet adds skip connections between the hidden layers and includes Batch normalization. All models are trained using Adam optimizer with a learning rate of $1e-3$ for up to 30 epochs, implementing early stopping with a patience of 5 epochs based on validation loss.

Image Models We downloaded all convolutional models, including VGG16 [22], ResNet50 [9], and DenseNet121 [12], from torchvision with pre-trained weights. All transformer-based models are downloaded using the timm library [25], and all zero-shot models are obtained from openclip [13]. More precisely, we utilized the following model variants:

DeiT [24]: deit_tiny_patch16_224.fb_in1k
ViT [5]: vit_tiny_patch16_224.augreg_in21k_ft_in1k
SwinT [18]: swin_tiny_patch4_window7_224.ms_in1k
MLPMixer [23]: mixer_b16_224.goog_in21k_ft_in1k
SigLip [27]: ViT-B-16-SigLIP

B.2 ReCalX Algorithmic Details

We implemented ReCalX as an extended version of classical temperature scaling. For a subset $S \subseteq \{1, \dots, d\}$, we define the perturbation level $\lambda(S)$ as the fraction of perturbed features: $\lambda(S) = (d - |S|)/d \in [0, 1]$. To account for different perturbation intensities, we partition $[0, 1]$ into $B = 10$ equal-width bins and learn a specific temperature for each bin. Let $\mathcal{B}_b := [\frac{b-1}{B}, \frac{b}{B}]$, $b \in 1, \dots, B$ denote these bins. Given a validation set $\mathcal{D}_{\text{val}} = (x_i, y_i)_{i=1}^N$, we optimize a temperature T_b for each bin by minimizing the cross-entropy loss \mathcal{L}_{CE} on perturbed samples with corresponding perturbation levels:

Algorithm 1 ReCalX

Require: Model f , validation data \mathcal{D}_{val} , number of bins B , samples per bin M

```

1: Initialize bins  $\mathcal{B}_b = [\frac{b-1}{B}, \frac{b}{B}]$  for  $b \in 1, \dots, B$  and Temperature parameters  $\{T_b = 1\}_{b=1}^B$ 
2: for  $b = 1$  to  $B$  do
3:    $\mathcal{L}_b = 0$  ▷ Initialize loss for bin  $b$ 
4:   for  $(x_i, y_i)$  in  $\mathcal{D}_{\text{val}}$  do
5:     for  $j = 1$  to  $M$  do
6:       Sample  $S \subset \{1, \dots, d\}$  with  $\lambda(S) \in \mathcal{B}_b$  ▷ Random subset with desired pert.level
7:        $\mathcal{L}_b += \mathcal{L}_{\text{CE}}(f_S^\pi(x_i, T_b), y_i)$ 
8:     end for
9:   end for
10:   $T_b^* = \arg \min_{T_b > 0} \mathcal{L}_b$  ▷ Optimize using L-BFGS
11: end for
    return  $\{T_b^*\}_{b=1}^B$ 

```

The algorithm iterates over each bin, sampling M random subsets with appropriate perturbation levels for each validation sample. For each bin, we optimize a temperature parameter using L-BFGS [17] to minimize the cross-entropy loss on the perturbed samples.

B.3 Calibration Error Analysis

To perform temperature scaling, we followed the standard approach introduced in [8], which optimizes a single temperature parameter via L-BFGS for the cross-entropy loss based on 1000 unperturbed samples from the corresponding validation set of each considered dataset. For this, we relied on the implementation provided by [28]. For ReCalX, we performed the algorithm presented in Appendix B, considering $B=10$ perturbation level bins, for each bin 200 distinct data instances with $M=5$ different perturbation samples, such that we again have a total recalibration set of 1000 per bin. For each perturbation level bin, the resulting calibration error is computed based on 10000 fresh samples from the remaining validation dataset of ImageNet for vision models and based on at least 6000 samples of the test set for each tabular dataset. To estimate explicitly the KL-Divergence-based calibration error, we utilized the dedicated estimator provided by [20]. For tabular data we considered fixed baseline replacement with the feature mean in the training dataset as perturbation. For vision data, we analyzed fixed baseline replacement with zeros (the default choice in [14]) as well as pixel replacement based on a significantly blurred image version using a Gaussian blur kernel of size 5 and sigmas of (0.1, 2).

B.4 Global Remove and Retrain Fidelity

The sequential retraining experiments evaluate the global removal and retrain fidelity of explanations by progressively removing the most important features and measuring model performance degradation. The procedure begins with a fully trained model (MLP or ResNet) on the complete feature set. Then, we derive global feature importance rankings from averaging the local feature attribution results of 1000 training samples using Shapley Values based on the implementation provided by SHAP [19]. As a perturbation, we used fixed baseline replacement with feature mean in the training dataset. This process is performed for both uncalibrated models and ReCalX-calibrated ones to compare their effectiveness in identifying truly important features. Subsequently, the top 5 most important features are sequentially removed one by one, and after each feature removal, the model is retrained from scratch using multiple random seeds (0, 123, 456) to ensure statistical robustness. Each retraining run again consists of maximally 30 epochs and might stop early with patience of 5. The retraining performance is measured on the test set based on the cross-entropy loss and is reported as the average over the three seeds [11].

B.5 Explanation Robustness

For the robustness experiments, we considered three popular perturbation-based methods given by LIME [21], KernelSHAP [19], and basic Feature Ablation, all based on their default implementation in Captum [14]. We further utilized a feature mask of 200 segments obtained via the SLIC algorithm [1]. To measure explanation robustness, rely on the Average Sensitivity (S_{AVG}) [2] and Max Sensitivity (S_{MAX}) [26] based on their implementation in Quantus [10]. Such metrics are given by:

$$S_{\text{AVG}} : \mathbb{E}_{\varepsilon} [\|\phi(x) - \phi(x + \varepsilon)\|_2^2] \quad \text{and} \quad S_{\text{MAX}} : \max_{\varepsilon} \|\phi(x) - \phi(x + \varepsilon)\|_2^2$$

where we utilized 50 tiny uniformly distributed corruptions $\varepsilon \sim \mathcal{U}[-0.1, 0.1]$. Each metric is computed as an average over 200 random samples from the remaining ImageNet validation dataset, which has not been used for recalibration.

C Additional Experimental Results

C.1 Calibration Error Analysis for Tabular Models

In Table 1, we provide additional results on the calibration error analysis on tabular classifiers. This Table extends the corresponding one in the main paper and includes 6 additional datasets.

C.2 Applying ReCalX to Regression problems

For regression, we consider a model $f : \mathcal{X} \rightarrow \mathcal{P}(Y)$ that maps inputs to probability distributions over a continuous target space $\mathcal{Y} \subseteq \mathbb{R}$. In our experiments, we use a neural network that outputs the parameters of a Gaussian distribution:

$$f(x) = \mathcal{N}(\mu(x), \sigma^2(x)),$$

where $\mu : \mathcal{X} \rightarrow \mathbb{R}$ predicts the mean and $\sigma : \mathcal{X} \rightarrow \mathbb{R}^+$ predicts the standard deviation. For regression models, calibration is typically assessed using quantile-based calibration error [15]. A regression model is well-calibrated if its predicted cumulative distribution function (CDF) matches the empirical frequency of observations. Formally, for any quantile level $\alpha \in [0, 1]$, a calibrated model should satisfy:

$$P(Y \leq F_{f(X)}^{-1}(\alpha)) = \alpha,$$

where $F_{f(X)}^{-1}(\alpha)$ denotes the α -quantile of the predicted distribution $f(X)$. The quantile-based calibration error measures the deviation from this ideal property:

$$CE_{\text{quantile}}(f) = \mathbb{E}_{\alpha \sim \text{Uniform}[0,1]} |P(Y \leq F_{f(X)}^{-1}(\alpha)) - \alpha|.$$

To apply ReCalX to regression models, we employ affine transformations instead of temperature scaling. For a given perturbation level $\lambda(S) = (d - |S|)/d$, we learn perturbation-specific scaling

Dataset	Uncalibrated		Temperature Scaling		ReCalX		
	$CE_{KL}^{AVG_S}$	$CE_{KL}^{MAX_S}$	$CE_{KL}^{AVG_S}$	$CE_{KL}^{MAX_S}$	$CE_{KL}^{AVG_S}$	$CE_{KL}^{MAX_S}$	$\downarrow_{MAX} (\%)$
MLP Model							
Electricity	0.0423	0.1534	0.0452	0.1664	0.0097	0.0163	89.4%
Coverttype	0.0412	0.0797	0.0564	0.1115	0.0047	0.0061	92.3%
Credit	0.2127	0.4763	0.2638	0.5961	0.0355	0.0533	88.8%
Pol	0.1740	0.6735	0.1689	0.6521	0.0570	0.1679	75.1%
Default-Credit	0.0589	0.2449	0.0593	0.2461	0.0073	0.0135	94.5%
Higgs	0.0203	0.0821	0.0262	0.1094	0.0012	0.0023	97.2%
House16H	0.0017	0.0033	0.0017	0.0035	0.0015	0.0034	3.0%
California	0.0966	0.3985	0.1005	0.4298	0.0161	0.0364	90.9%
MagicTelescope	0.2126	0.5881	0.1954	0.5154	0.0261	0.0374	93.6%
Diabetes130US	0.0035	0.0095	0.0035	0.0093	0.0025	0.0046	51.6%
ResNet Model							
Electricity	0.0032	0.0118	0.0090	0.0355	0.0017	0.0049	58.5%
Coverttype	0.0439	0.0963	0.0614	0.1413	0.0058	0.0080	91.7%
Credit	0.0721	0.0830	0.0778	0.0847	0.0366	0.0773	6.9%
Pol	0.2853	0.8633	0.3187	1.0173	0.0727	0.0910	89.5%
Default-Credit	0.0675	0.2237	0.0532	0.1790	0.0087	0.0126	94.4%
Higgs	0.0214	0.0482	0.0243	0.0543	0.0018	0.0028	94.2%
House16H	0.0339	0.0882	0.0325	0.0818	0.0127	0.0167	81.1%
California	0.0200	0.0344	0.0198	0.0319	0.0078	0.0161	53.2%
MagicTelescope	0.1460	0.4200	0.1347	0.3805	0.0213	0.0276	93.4%
Diabetes130US	0.0087	0.0238	0.0079	0.0203	0.0041	0.0061	74.4%

Table 1: Calibration error comparison across datasets for MLP and ResNet models using mean value replacement perturbation. $CE_{KL}^{AVG_S}$ and $CE_{KL}^{MAX_S}$ represent average and maximum KL divergence-based calibration errors, across perturbation levels. Lower values indicate better calibration. The improvement column (\downarrow_{MAX}) shows the relative reduction in maximum calibration error achieved by ReCalX compared to uncalibrated models. ReCalX consistently outperforms both uncalibrated models and Temperature Scaling across all settings, with improvements of up to 97.2%.

and shift parameters. Specifically, for each perturbation bin $b \in \{1, \dots, B\}$, we optimize parameters $a_b > 0$ (scaling) and $c_b \in \mathbb{R}$ (shift) that transform the predicted mean:

$$f_{\text{ReCalX}}^\pi(x, S; \{a_b, c_b\}_{b=1}^B) = a_{b(S)} \cdot \mathcal{N}(\mu(\pi(x, S)), \sigma^2(\pi(x, S))) + c_{b(S)}$$

where $b(S)$ denotes the bin index corresponding to the perturbation level $\lambda(S)$. This affine transformation is strictly monotonic in the predicted mean and thus satisfies the information-preserving property (Proposition 4.2). The parameters $\{a_b, c_b\}_{b=1}^B$ are optimized on a validation set by minimizing the negative log-likelihood under different perturbation levels, analogous to the classification case. To validate the effectiveness of ReCalX for regression, we conducted experiments on two tabular regression datasets: `wine quality` and `houses` from [7]. We trained simple MLP regression models with one hidden layer of size 64 and Gaussian output distributions [3]. Following the setup of our main experiments, we used fixed baseline replacement with the feature mean as the perturbation strategy π . We evaluated calibration using the quantile-based calibration error by sampling 100 predictions from the predictive distribution for each test instance and computing the empirical coverage at uniformly spaced quantile levels. We report both the average calibration error across all perturbation strengths ($CE_{\text{quantile}}^{AVG_S}$) and the maximum calibration error across perturbation levels ($CE_{\text{quantile}}^{MAX_S}$), which directly corresponds to the quantity CE^{max_S} analyzed in our theoretical results (Theorem 3.4). Table 2 presents the calibration errors before and after applying ReCalX. The results demonstrate that ReCalX substantially reduces both average and maximum calibration errors under explainability-specific perturbations, confirming that perturbation-specific recalibration is also effective for regression tasks. These results indicate that ReCalX is capable of reducing miscalibration under explainability-specific perturbations for regression tasks, supporting the generality of our theoretical framework beyond classification problems.

Table 2: Quantile-based calibration errors on regression datasets before and after applying ReCalX. Results show average ($CE_{quantile}^{AVG_S}$) and maximum ($CE_{quantile}^{MAX_S}$) calibration errors across perturbation strengths.

Dataset	Uncalibrated $CE_{quantile}^{AVG_S}$	ReCalX $CE_{quantile}^{AVG_S}$	Uncalibrated $CE_{quantile}^{MAX_S}$	ReCalX $CE_{quantile}^{MAX_S}$
wine quality	0.06495	0.01182	0.1053	0.0183
houses	0.18078	0.00741	0.2512	0.0178

C.3 Sensitivity to the Number of Bins

An important design choice in ReCalX is the number of bins B used to partition the perturbation space. Each bin corresponds to a range of perturbation levels $\lambda(S)$ and is assigned a separate temperature parameter. In this section, we investigate the sensitivity of ReCalX to this hyperparameter and discuss the associated trade-offs.

We conducted a dedicated ablation study using a Vision Transformer (ViT) model on ImageNet with the same experimental setup described in the main paper. We systematically varied the number of bins from 1 to 25 and measured the resulting calibration errors. Table 3 presents the average and maximum KL-based calibration errors across perturbation levels for different bin configurations.

Table 3: Impact of the number of bins on ReCalX calibration performance. Results show average (CE_{KL}^{avg}) and maximum (CE_{KL}^{max}) calibration errors for a ViT model on ImageNet.

Number of Bins	ReCalX CE_{KL}^{avg}	ReCalX CE_{KL}^{max}
1	0.0506	0.0915
3	0.0137	0.0440
5	0.0092	0.0215
10	0.0075	0.0138
15	0.0070	0.0128
20	0.0070	0.0116
25	0.0067	0.0116

The results reveal that increasing the number of bins generally improves calibration performance. This is expected because a finer-grained binning strategy allows ReCalX to learn a more nuanced set of temperatures that better match the model’s varying calibration properties across different perturbation levels. The most significant improvements occur when increasing from 1 bin (equivalent to standard temperature scaling) to approximately 10 bins. Beyond this point, the reduction in calibration error exhibits diminishing returns and eventually saturates.

This presents a trade-off between calibration granularity and computational cost during the one-time setup phase. A higher number of bins enables ReCalX to learn more fine-grained temperature parameters, better adapting to the model’s specific miscalibration at different perturbation levels. As shown in Table 3, this generally leads to lower maximum calibration error. However, each bin requires a separate temperature parameter to be optimized on the validation set. Therefore, increasing the number of bins linearly increases the computational effort associated with the calibration procedure.

Our empirical results indicate that the performance gains from adding more bins diminish and eventually saturate around 10 bins. To manage this trade-off, we recommend choosing a moderate number of bins (e.g. 10) that captures most of the potential calibration improvement without incurring unnecessarily high computational cost during setup.

C.4 Sensitivity to Calibration Set Size

To investigate the sensitivity of ReCalX to the amount of validation data, we performed an ablation study on a Vision Transformer (ViT) model on ImageNet, systematically varying the number of samples in the calibration set from 10 to 1000. For each calibration set size, we randomly sampled the specified number of validation examples and optimized the ReCalX temperatures using the procedure

described in Section 4. To account for sampling variability, we repeated this process with 10 different random seeds and report the average calibration errors.

The uncalibrated baseline model exhibited calibration errors of 0.0936 (average) and 0.2618 (maximum) across perturbation levels. Table 4 presents the resulting calibration errors after applying ReCalX with different calibration set sizes.

Table 4: Impact of calibration set size on ReCalX performance. Results show average (CE_{KL}^{avg}) and maximum (CE_{KL}^{max}) calibration errors for a ViT model on ImageNet, averaged over 10 random seeds. The uncalibrated baseline errors were 0.0936 (avg) and 0.2618 (max).

Calibration Set Size	ReCalX CE_{KL}^{avg}	ReCalX CE_{KL}^{max}
10	0.0175	0.0743
25	0.0099	0.0346
50	0.0045	0.0115
75	0.0035	0.0088
100	0.0031	0.0063
200	0.0028	0.0060
500	0.0021	0.0027
1000	0.0021	0.0025

The table shows that even a small, randomly selected validation set of 50-100 samples is sufficient to achieve the vast majority of the potential calibration improvement. Beyond 100 samples, the gains continue but with diminishing returns, and performance largely saturates around 500 samples.

C.5 Temperature Relationship Across Perturbation Types

An important question is whether the temperature parameters learned by ReCalX for one perturbation strategy may even generalize to other perturbation types. To investigate this, we leveraged the experiments already conducted in the main paper, where we optimized two separate sets of ReCalX temperatures for different classifiers on ImageNet, one using blur perturbations and another using zero-baseline replacement perturbations. This allows us to compute the correlation between the resulting sets of bin-specific temperatures across the perturbation levels to assess their correspondence.

Table 5 presents the Pearson correlation coefficients between the temperature parameters computed across all perturbation bins for each model architecture.

Table 5: Pearson correlation between ReCalX temperature parameters learned for blur perturbations versus zero-baseline replacement perturbations across different model architectures on ImageNet.

Model	Pearson Correlation
ResNet50	0.984
DenseNet	0.975
ViT	0.914
SigLip	0.956

The results show strong positive correlations across all architectures, ranging from 0.914 to 0.984. This suggests that a model’s underlying miscalibration patterns exhibit substantial similarity across different perturbation types. While these findings suggest that ReCalX temperatures may offer some degree of generalization across perturbation strategies, we recommend explicitly calibrating ReCalX for the specific perturbation strategy being used. This ensures that the theoretical benefits of our method are fully realized for the explanation method at hand.

C.6 ReCalX with Domain-Specific Perturbations

To further validate the generality of ReCalX beyond simple perturbation strategies, we conducted additional experiments evaluating its performance with more complex, domain-aware perturbation methods. Specifically, we chose two advanced inpainting techniques available in the official SHAP

library [19], which leverage the `inpaint_telega` and `inpaint_ns` algorithms provided by the OpenCV package. Unlike simple blurring or zero-filling, these methods are designed to fill corrupted image regions in a semantically plausible way by incorporating information from the surrounding pixels through partial differential equation-based reconstruction. Both approaches aim to generate realistic replacements that better preserve the natural image statistics compared to fixed baseline strategies, making them particularly suitable for explaining vision models where context matters. We evaluated the performance of ReCalX on a Vision Transformer (ViT) model when using these inpainting methods on ImageNet. Table 6 shows the average and maximum KL-divergence calibration errors before calibration (Initial), after applying standard temperature scaling (TS), and after applying ReCalX with 10 perturbation-specific bins.

Table 6: Calibration errors for domain-specific inpainting perturbations on ViT. Results compare uncalibrated models (Initial), standard temperature scaling (TS), and ReCalX across two advanced inpainting methods.

Perturbation	Initial _{Avg}	Initial _{Max}	TS _{Avg}	TS _{Max}	ReCalX _{Avg}	ReCalX _{Max}
<code>inpaint_ns</code>	0.3796	1.0337	0.2327	0.6890	0.0305	0.1012
<code>inpaint_telega</code>	0.3843	1.0612	0.2370	0.6993	0.0293	0.0991

The results demonstrate that ReCalX remains highly effective and substantially reduces calibration error even for these sophisticated, domain-specific perturbations.

C.7 Calibration Error Plots for Vision Models

In Figure 1 we provide calibration error plots for additional vision models obtained based on fixed baseline perturbation with zeros, and in Figure 2 corresponding ones using the blur perturbation.

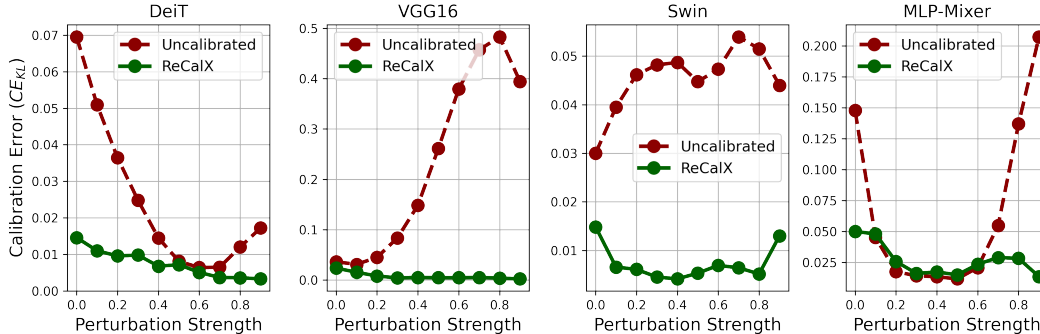


Figure 1: Calibration error results for additional image classifiers on ImageNet under fixed baseline perturbation with zeros. Across all methods, miscalibration varies significantly across the perturbation severity. While also for most image models the error tends to grow with perturbation level, this behavior is not always consistent. This flexible behavior highlights the importance of calibration strategies that are adaptive to the perturbation strength.

C.8 Global Remove and Retrain Fidelity

In Figure 3 we provide additional remove and retrain fidelity results for additional datasets based on Shapley Values, and in Figure 4 we report corresponding results using LIME as the underlying explanation method.

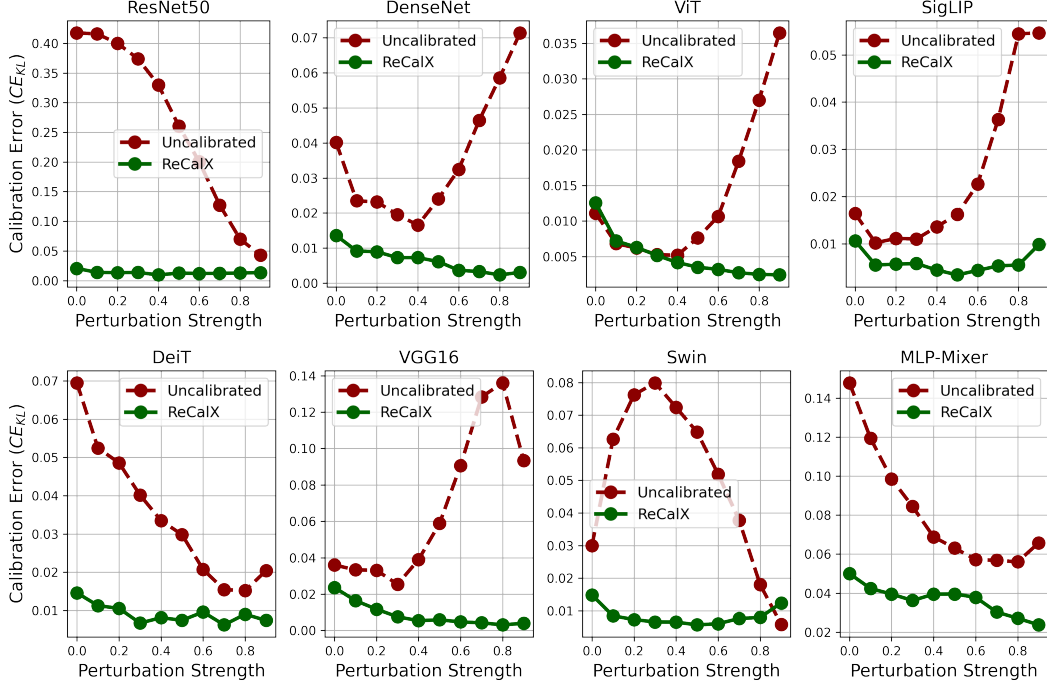


Figure 2: Calibration error results for popular image classifiers on ImageNet under blur perturbation with zeros. Across all methods, miscalibration varies significantly across the perturbation severity. While also for most image models the error tends to grow with perturbation level, this behavior is not always consistent. This flexible behavior highlights the importance of calibration strategies that are adaptive to the perturbation strength.

D Computational Costs of ReCalX

In this section, we demonstrate that the additional computational cost of ReCalX at inference time when an explanation is actually generated is negligible. This is because the total explanation time is overwhelmingly dominated by the repeated model queries required by the perturbation-based method itself. In contrast, ReCalX only adds two trivial operations: a quick temperature lookup based on the perturbation level and a single division operation on the model’s logits. To quantify this overhead, we measured the per-explanation computation time for various models on ImageNet when generating explanations with and without ReCalX. Table 7 shows the additional computation time in seconds induced by ReCalX when computing different explanation methods, averaged over 100 runs. The results confirm that the extra time is minor and typically in the order of milliseconds, representing a negligible fraction of the total explanation time.

Table 7: Additional inference time (in seconds) induced by ReCalX when generating explanations, averaged over 100 runs on ImageNet. The overhead is negligible compared to the total explanation computation time.

Method	ResNet18	DenseNet	ViT	SigLip
LIME	0.0193s	0.0219s	0.0184s	0.0210s
Shapley Values	0.2587s	0.2517s	0.1754s	0.3431s

References

- [1] Radhakrishna Achanta, Appu Shaji, Kevin Smith, Aurelien Lucchi, Pascal Fua, and Sabine Süsstrunk. Slic superpixels compared to state-of-the-art superpixel methods. *IEEE transactions*

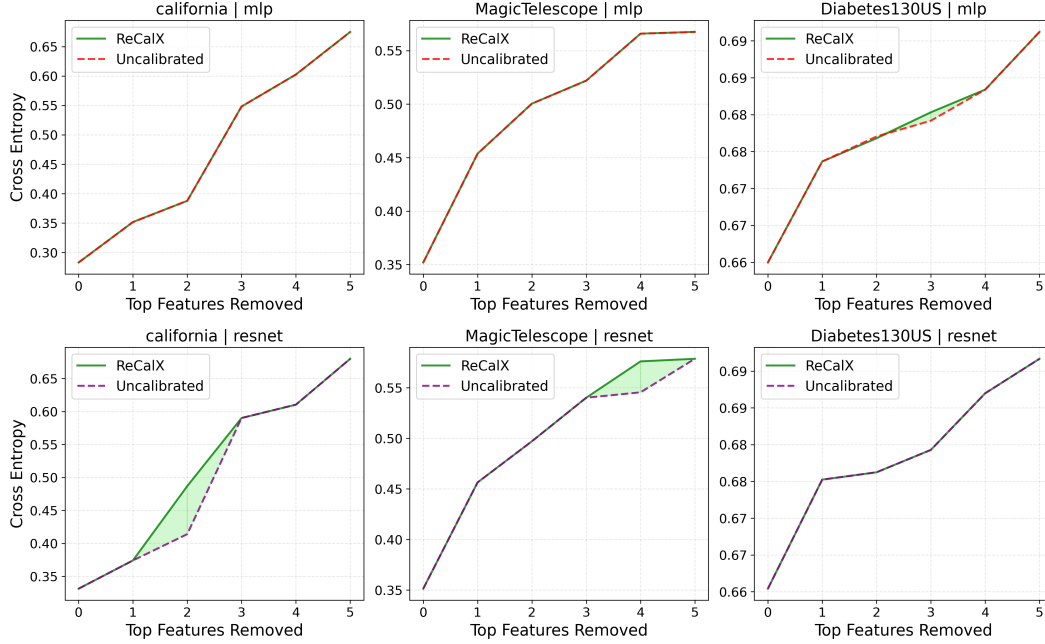


Figure 3: Retraining results on additional tabular datasets for an MLP (top row) and a ResNet (bottom row) when the features are removed based on their global importance estimated via Shapley Values. Whenever calibrated explanations imply a different importance ranking (green area), the resulting performance loss is consistently higher compared to the uncalibrated importance indications. Hence, ReCalX enables better identification of truly relevant features that are crucial for good performance.

on pattern analysis and machine intelligence, 34(11):2274–2282, 2012.

- [2] Umang Bhatt, Adrian Weller, and José MF Moura. Evaluating and aggregating feature-based model explanations. In *Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence*, pages 3016–3022, 2021.
- [3] Christopher M Bishop. Mixture density networks. *Aston University*, 1994.
- [4] Thomas M Cover and Joy A Thomas. *Elements of Information Theory*. John Wiley & Sons, 2012.
- [5] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2020.
- [6] Yury Gorishniy, Ivan Rubachev, Valentin Khruikov, and Artem Babenko. Revisiting deep learning models for tabular data. *Advances in neural information processing systems*, 34:18932–18943, 2021.
- [7] Léo Grinsztajn, Edouard Oyallon, and Gaël Varoquaux. Why do tree-based models still outperform deep learning on typical tabular data? *Advances in neural information processing systems*, 35:507–520, 2022.
- [8] Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. On calibration of modern neural networks. In *International conference on machine learning*, pages 1321–1330. PMLR, 2017.
- [9] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

- [10] Anna Hedström, Leander Weber, Daniel Krakowczyk, Dilyara Bareeva, Franz Motzkus, Wojciech Samek, Sebastian Lapuschkin, and Marina M-C Höhne. Quantus: An explainable ai toolkit for responsible evaluation of neural network explanations and beyond. *Journal of Machine Learning Research*, 24(34):1–11, 2023.
- [11] Sara Hooker, Dumitru Erhan, Pieter-Jan Kindermans, and Been Kim. A benchmark for interpretability methods in deep neural networks. *Advances in neural information processing systems*, 32, 2019.
- [12] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4700–4708, 2017.
- [13] Gabriel Ilharco, Mitchell Wortsman, Ross Wightman, Cade Gordon, Nicholas Carlini, Rohan Taori, Achal Dave, Vaishaal Shankar, Hongseok Namkoong, John Miller, Hannaneh Hajishirzi, Ali Farhadi, and Ludwig Schmidt. Openclip, July 2021.
- [14] Narine Kokhlikyan, Vivek Miglani, Miguel Martin, Edward Wang, Bilal Alsallakh, Jonathan Reynolds, Alexander Melnikov, Natalia Kliushkina, Carlos Araya, Siqi Yan, et al. Captum: A unified and generic model interpretability library for pytorch. *arXiv preprint arXiv:2009.07896*, 2020.
- [15] Volodymyr Kuleshov, Nathan Fenner, and Stefano Ermon. Accurate uncertainties for deep learning using calibrated regression. In *International conference on machine learning*, pages 2796–2804. PMLR, 2018.
- [16] Chris Lin, Ian Covert, and Su-In Lee. On the robustness of removal-based feature attributions. *Advances in Neural Information Processing Systems*, 36, 2024.
- [17] Dong C Liu and Jorge Nocedal. On the limited memory bfgs method for large scale optimization. *Mathematical programming*, 45(1):503–528, 1989.
- [18] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10012–10022, 2021.
- [19] Scott M Lundberg and Su-In Lee. A unified approach to interpreting model predictions. *Advances in Neural Information Processing Systems*, 30, 2017.
- [20] Teodora Popordanoska, Sebastian Gregor Gruber, Aleksei Tiulpin, Florian Buettner, and Matthew B Blaschko. Consistent and asymptotically unbiased estimation of proper calibration errors. In *International Conference on Artificial Intelligence and Statistics*, pages 3466–3474. PMLR, 2024.
- [21] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. " why should i trust you?" explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1135–1144, 2016.
- [22] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.
- [23] Ilya O Tolstikhin, Neil Houlsby, Alexander Kolesnikov, Lucas Beyer, Xiaohua Zhai, Thomas Unterthiner, Jessica Yung, Andreas Steiner, Daniel Keysers, Jakob Uszkoreit, et al. Mlp-mixer: An all-mlp architecture for vision. *Advances in neural information processing systems*, 34:24261–24272, 2021.
- [24] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In *International conference on machine learning*, pages 10347–10357. PMLR, 2021.
- [25] Ross Wightman. Pytorch image models. <https://github.com/rwightman/pytorch-image-models>, 2019.

- [26] Chih-Kuan Yeh, Cheng-Yu Hsieh, Arun Suggala, David I Inouye, and Pradeep K Ravikumar. On the (in) fidelity and sensitivity of explanations. *Advances in Neural Information Processing Systems*, 32, 2019.
- [27] Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11975–11986, 2023.
- [28] Jize Zhang, Bhavya Kailkhura, and T Yong-Jin Han. Mix-n-match: Ensemble and compositional methods for uncertainty calibration in deep learning. In *International conference on machine learning*, pages 11117–11128. PMLR, 2020.

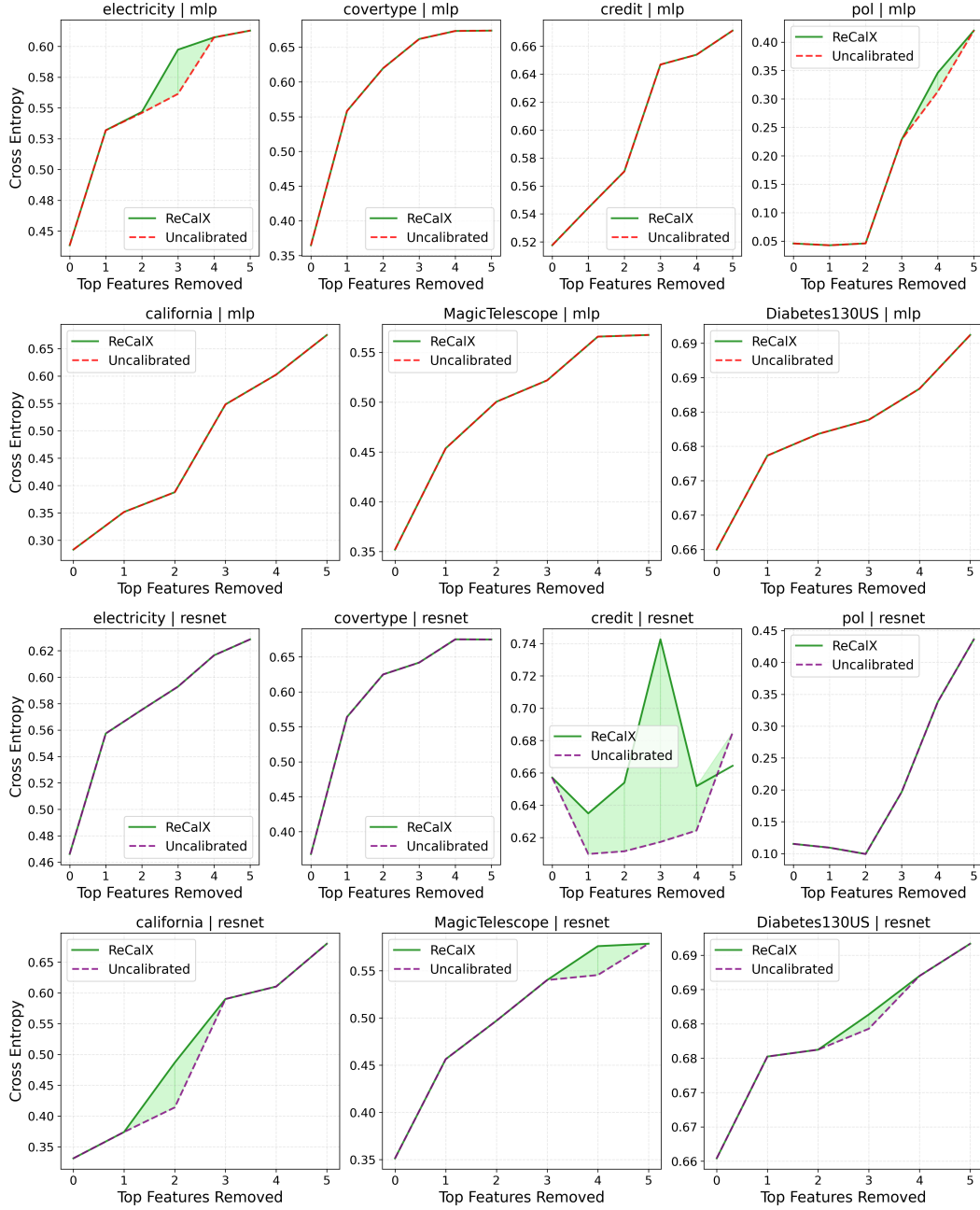


Figure 4: Retraining results on different tabular datasets for an MLP (first and second row) and a ResNet (third and fourth row) when the features are removed based on their global importance estimated via LIME. Whenever calibrated explanations imply a different importance ranking (green area), the resulting performance loss is consistently higher compared to the uncalibrated importance indications. Hence, ReCalX enables better identification of truly relevant features that are crucial for good performance.