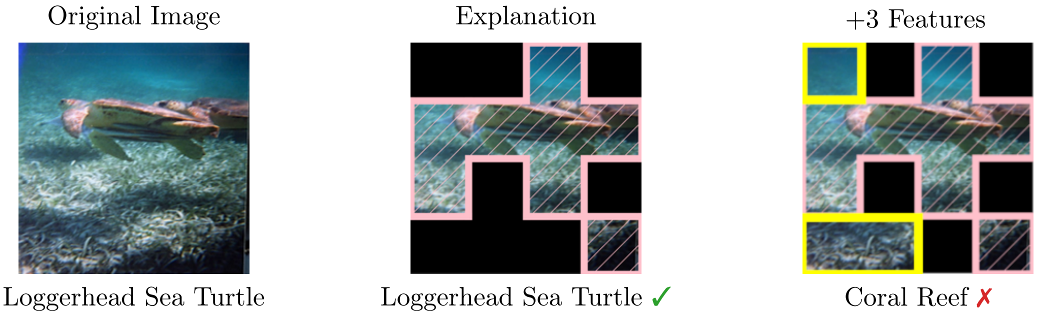
Probabilistic Stability Guarantees for Feature Attributions

Helen Jin[*], Anton Xue[*], Weiqiu You, Surbhi Goel, and Eric Wong
Department of Computer and Information Science, University of Pennsylvania

April 9, 2025

Abstract

Stability guarantees are an emerging tool for feature attributions, but existing certification methods rely on specialized architectures and yield conservative bounds that are not useful in practice. To address this gap, we introduce soft stability and present a model-agnostic, sample-efficient certification algorithm that provides non-trivial guarantees that are practically interpretable. Interestingly, we show that mild smoothing enhances soft stability without incurring the accuracy degradation observed in existing smoothing-based stability certifications. To explain this phenomenon, we leverage techniques from Boolean function analysis to characterize and provide insights into the impact of smoothing on classifier behavior. We demonstrate the improvement of soft stability over hard stability through experiments on vision and language tasks with various feature attribution methods.



Original Image

Explanation

+3 Features

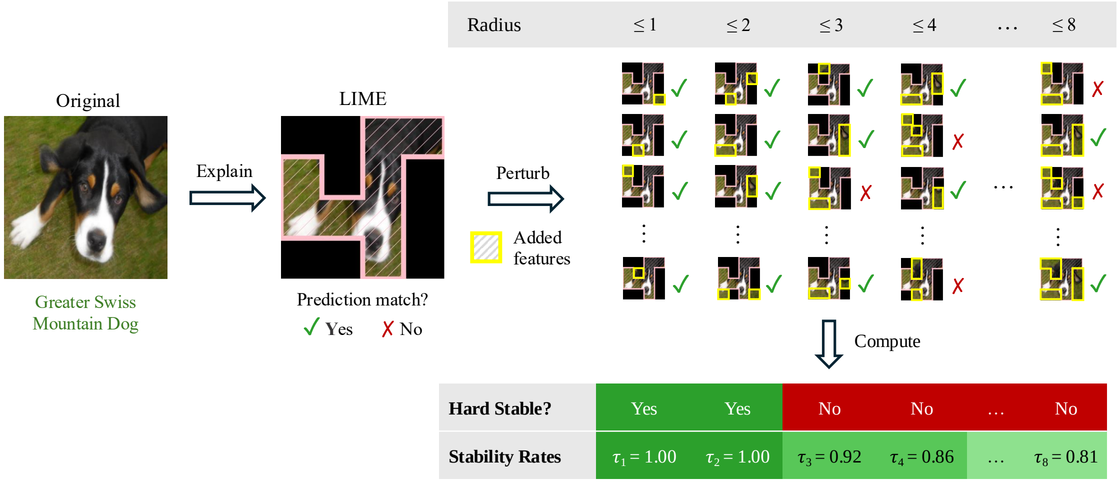
Figure 1: **An unstable explanation.** Given an input image (left), the features selected by LIME [44] (middle, in pink) are enough to preserve Vision Transformer's [15] prediction. However, including three additional features (right, in yellow) flips the prediction, suggesting that the explanation is not robust.

1 Introduction

Powerful machine learning models are increasingly deployed in practice. However, their opacity presents a major challenge in being adopted in high-stake domains, where transparent explanations are needed in decision making. In healthcare, for instance, doctors require insights into the diagnostic steps to trust the model and integrate them into clinical practice effectively [28]. Similarly, in the legal domain, attorneys must ensure that decisions reached with the assistance of models meet stringent judicial standards [45].

There is much interest in explaining the behavior of complex models. One popular class of explanation methods are *feature attributions* [34, 44], which aim to select the input features most important to a model's prediction. However, many explanations are *unstable*, as illustrated in Figure 1: additionally including a few features may change the model's prediction. This kind of non-robustness is often undesirable because

[*] Equal contribution.



Radius	≤ 1	≤ 2	≤ 3	≤ 4	…	≤ 8
Hard Stable?	Yes	Yes	No	No	…	No
Stability Rates	$\tau_1 = 1.00$	$\tau_2 = 1.00$	$\tau_3 = 0.92$	$\tau_4 = 0.86$	…	$\tau_8 = 0.81$

Figure 2: **Soft stability provides a fine-grained measure of robustness.** For the Vision Transformer classifier [15], LIME's explanation [44] is only hard stable at radius $r \leq 2$. In contrast, the stability rate — the key metric of soft stability — offers a more nuanced measure of sensitivity to added features.

it suggests that the given explanation is unreliable [40, 53, 58]. Such phenomena has motivated efforts to quantify how model predictions vary with explanations, including the effects of adding or removing features [47, 54] and the influence of the selection's shape [20, 46]. However, most existing works focus on empirical measures [3], with limited formal, mathematical guarantees on robustness.

To address this gap, prior work in Xue et al. [56] considers *stability* as a formal certification framework for robust explanations. In particular, a *stable explanation* is one where adding any small number of features does not alter the model's prediction, up to some maximum tolerance. However, finding this tolerance is non-trivial: for an arbitrary model, one must exhaustively enumerate and check all possible perturbations in a computationally intractable manner. To overcome this, Xue et al. [56] apply adversarial robustness techniques [12, 30] to construct *smoothed classifiers*, which have mathematical properties for efficiently and non-trivially lower-bounding the maximum tolerance. While this is a first step towards certifiably robust explanations, such notions of tolerance are coarse, and smoothing-based certificates are often conservative.

In this work, we introduce *soft stability*, a new variant of stability defined in contrast to *hard stability* [56]. As illustrated in Figure 2, hard stability certifies whether *all* small perturbations to an explanation yield the same prediction, whereas soft stability quantifies *how often* the prediction remains unchanged. Soft stability may thus be interpreted as a probabilistic relaxation of hard stability, and enables a more fine-grained analysis of explanation robustness. Crucially, this shift in perspective allows for model-agnostic applicability and admits efficient certification algorithms that provide stronger guarantees. This work advances our understanding of robust feature-based explanations, and we summarize our contributions below.

Soft Stability is Practical and Attains Useful Guarantees We introduce soft stability to address the limitations of hard stability. Soft stability certification is model-agnostic and provides non-trivial guarantees that are practically interpretable. In contrast, hard stability certification relies on heavily smoothed classifiers, which not only suffer from accuracy degradation but also yield overly conservative guarantees. In Section 3, we examine the drawbacks of hard stability and introduce an efficient soft stability certification algorithm.

Smoothing Can Improve Soft Stability Although soft stability certification does not require smoothing, we found that mild smoothing can improve guarantees without incurring the severe accuracy degradation typically associated with the smoothing-based certification of hard stability. To explain this phenomenon, we developed new techniques for analyzing the effects of smoothing on stability using Boolean function analysis. We give an overview of our main theoretical results in Section 4.

Soft Stability Outperforms Hard Stability We empirically validated the improvement of soft stability over hard stability on a combination of vision and language tasks and various feature attribution methods. In particular, the attainable soft stability certificates are often two orders of magnitude stronger than the hard stability certificates. Moreover, we show that mildly smoothed classifiers attain a good trade-off between accuracy and stability. We present our experimental results in Section 5.

2 Background and Overview

First, we will give an overview of feature attributions. We then discuss the existing work on hard stability and introduce the notion of soft stability.

2.1 Feature Attributions as Explanations

Feature attributions are widely used in explainability due to their simplicity and generality. To formalize our discussion, we consider classifiers of the form $f : \mathbb{R}^n \to \mathbb{R}^m$, which map n-dimensional inputs to m logits representing class probabilities. A feature attribution method assigns a score α_i to each input feature x_i to indicate its importance to the model's prediction $f(x)$. The definition of importance depends on the method. In gradient-based methods [49, 51], each α_i might be dependent on $\nabla_{x_i} f(x)$, whereas in Shapley value-based methods [34, 50], they are used to measure the Shapley value at x_i. Although attribution scores are typically real-valued, it is common to binarize them to $\alpha \in \{0,1\}^n$ by selecting only the top-k highest-scoring features [44], thereby aligning with the general preference for concise explanations [39].

2.2 Hard Stability and Soft Stability

Many evaluation metrics exist for binary-valued feature attributions [3]. To compare two attributions $\alpha, \alpha' \in \{0,1\}^n$, it is common to study whether they *induce* the same prediction with respect to a given classifier $f : \mathbb{R}^n \to \mathbb{R}^m$ and input $x \in \mathbb{R}^n$. Let $(x \odot \alpha) \in \mathbb{R}^n$ be the α-masked variant of x, where $\odot$ is the coordinate-wise product of two vectors. Next, we write $f(x \odot \alpha) \cong f(x \odot \alpha')$ to mean that the masked inputs $x \odot \alpha$ and $x \odot \alpha'$ have the same prediction on f, which holds if:

$$\arg\max_k f(x \odot \alpha)_k = \arg\max_{k'} f(x \odot \alpha')_{k'}, \tag{1}$$

where $k \in \{1, \dots, m\}$ indexes the predicted class. This form of evaluating feature sets is related to notions of *fidelity*, *consistency*, and *preservation* in the explainability literature [40], but the specific terminology and definition vary by author and source. Furthermore, attribution-masked evaluation is more commonly seen in vision tasks [23], though it is also present in language modeling [35, 57].

It is often desirable that two "similar" attributions induce the same prediction [58]. Although various measures of similarity exist, we are interested in the notion of additive perturbations. Specifically, we conceptualize an additively perturbed attribution α' as one that contains *more information* (features) than α, where the desiderata is that adding more features to a "good quality" α should not easily alter the prediction.

Definition 2.1 (Additive Perturbations)**.** For an attribution α and integer-valued radius $r \geq 0$, define r-additive perturbation set of α as:

$$\Delta_r(\alpha) = \{\alpha' \in \{0,1\}^n : \alpha' \geq \alpha,\ |\alpha' - \alpha| \leq r\}, \tag{2}$$

where $\alpha' \geq \alpha$ iff each $\alpha'_i \geq \alpha_i$ and $|\cdot|$ counts the non-zeros in a binary vector (i.e., the ℓ^0 norm).

Intuitively, $\Delta_r(\alpha)$ represents the set of attributions that are at least as informative as α, differing by at most r features. This allows us to study the robustness of explanations by analyzing whether small modifications in feature selection affect the model's prediction. A natural way to formalize such robustness is through *stability*: an attribution α should be considered stable if adding a small number of features does not alter the classifier's decision. We now define *hard stability*, which reinforces this concept strictly.

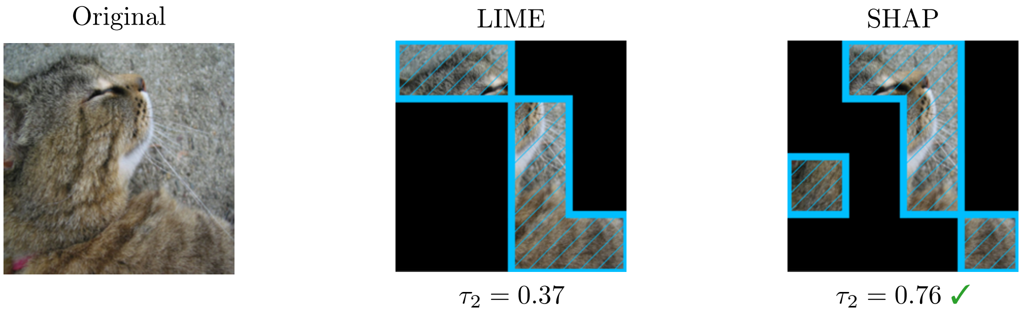


Figure 3: **Similar explanations may have different stability rates.** Despite visual similarities, the explanations generated by LIME [44] (middle) and SHAP [34] (right) have different stability rates at radius $r = 2$. In this example, SHAP's explanation is considerably more stable than LIME's.

Definition 2.2 (Hard Stability [1] [56])**.** For a classifier f and input x, the explanation α is *hard-stable* with radius r if: $f(x \odot \alpha') \cong f(x \odot \alpha)$ for all $\alpha' \in \Delta_r$.

However, hard stability is not straightforward to certify, and existing algorithms suffer from costly tradeoffs that we later discuss in Section 3.1. This motivates us to investigate *relaxations* that admit efficient certification algorithms while remaining practically useful. In particular, we are motivated by the increasing usage of probabilistic guarantees in domains such as medical imaging [16], drug discovery [6], and autonomous driving [32], which are often formulated in terms of confidence [7, 11]. We thus present a probabilistic relaxation of hard stability, quantified by the *stability rate*, as follows.

Definition 2.3 (Soft Stability)**.** For a classifier f and input x, define the *stability rate* of attribution α at radius r as:

$$\tau_r(f, x, \alpha) = \Pr_{\alpha' \sim \Delta_r} [f(x \odot \alpha') \cong f(x \odot \alpha)], \quad \text{where } \alpha' \sim \Delta_r \text{ is uniformly sampled.} \tag{3}$$

A higher stability rate τ_r indicates a greater likelihood that a perturbation of at most r features preserves the prediction. In fact, soft stability generalizes hard stability, as the extreme case of $\tau_r = 1$ recovers the hard stability condition. In the context of robustness, soft stability may differ greatly between two explanations that appear similar, as shown in Figure 3, where two explanations that differ at only two features have drastically different stability rates.

Alternative Formulations Our definition of soft stability is one of many possible variants. For example, one might define $\tau_{=k}$ as the probability that the prediction remains unchanged under an *exactly* k-sized additive perturbation. A conservative variant could then take the minimum over $\tau_{=1}, \ldots, \tau_{=r}$. The choice of formulation affects the implementation of the certification algorithm.

3 Certifying Soft Stability

We first discuss the limitations of existing methods for certifying hard stability. We then introduce a sampling-based algorithm to efficiently certify the soft stability of any model.

3.1 Challenges in Certifying Hard Stability

Existing approaches to certifying hard stability rely on a classifier's *Lipschitz constant*, which is a measure of function smoothness. While useful for robustness certification [12], the Lipschitz constant is often intractable to compute [52] and challenging to approximate [17, 55]. To address this, Xue et al. [56] derive smoothed

[1] Xue et al. [56] equivalently call this property "incrementally stable" and define "stable" as a stricter property.

classifiers with a known Lipschitz constant by construction. Starting with any classifier f, one defines the smoothed classifier $\tilde{f}$ as the expectation over randomly perturbed inputs:

$$\tilde{f}(x) = \frac{1}{N}\big[f(x^{(1)}) + \cdots + f(x^{(N)})\big], \tag{4}$$

where $x^{(1)}, \dots, x^{(N)} \sim \mathcal{D}(x)$ are sampled perturbations of x. If $\mathcal{D}$ is properly chosen, then the smoothed classifier $\tilde{f}$ has a Lipschitz constant κ that is explicitly known in expectation. [2]

Since κ measures a function's sensitivity to input perturbations, a smaller κ implies a smoother (i.e., more robust) classifier. Crucially, because $\tilde{f}$ is designed to have a known Lipschitz constant, this enables efficient computation of hard stability guarantees: in general, a smaller κ leads to a larger certified radius. We describe how the certified radius is computed in Theorem C.1.

Smoothing has Performance Trade-offs A key limitation of smoothing-based certificates is that the stability guarantees apply to $\tilde{f}$, not the original classifier f. Typically, the smoother the model, the stronger its guarantees (larger certified radii), but this comes at the cost of accuracy. This tension arises because excessive smoothing reduces a model's sensitivity, making it harder to distinguish between classes [5, 22].

Smoothing-based Hard Stability is Conservative Even when smoothing-based certification is feasible, the resulting certified radii are often conservative. The main reason is that these radii depend on a global property (the Lipschitz constant κ) to make local guarantees about feature perturbations. In hard stability certification [56], the certified radius of $\tilde{f}$ scales as $\mathcal{O}(1/\kappa)$ for any input x and attribution α, and we elaborate on this in Theorem C.1.

3.2 Estimating Soft Stability

Unlike hard stability, which requires destructively smoothing the classifier and often yields conservative guarantees, soft stability can be estimated efficiently for any classifier. Its key measure, the *stability rate* τ_r, can be efficiently estimated via the following algorithm.

Theorem 3.1 (Estimation Algorithm). *Let $N \geq \frac{\log(2/\delta)}{2\varepsilon^2}$ for any $\varepsilon > 0$ and $\delta > 0$. For a classifier f, input x, explanation α, and radius r, define the stability rate estimator:*

$$\hat{\tau}_r = \frac{1}{N}\sum_{i=1}^{N} \mathbf{1}\big[f(x \odot \alpha^{(i)}) \cong f(x \odot \alpha)\big], \quad \textit{where } \alpha^{(1)}, \dots, \alpha^{(N)} \sim \Delta_r(\alpha) \textit{ are i.i.d. samples.} \tag{5}$$

Then, with probability $\geq 1 - \delta$, it holds that $|\tau_r - \hat{\tau}_r| \leq \varepsilon$.

Proof. The sample complexity N follows from applying Hoeffding's inequality to the mean estimation of independent Bernoulli random variables $X^{(1)}, \dots, X^{(N)}$, where each $X^{(i)} = \mathbf{1}[f(x \odot \alpha^{(i)}) \cong f(x \odot \alpha)]$. □

We illustrate this algorithm in Figure 4, where note that the sample size N depends only on ε and δ because τ_r is a one-dimensional statistic. Moreover, certifying soft stability does not require deriving a smoothed classifier. Unlike hard stability, which applies to the smoothed classifier $\tilde{f}$, soft stability provides robustness guarantees directly on the original classifier f. This eliminates the need for a destructive smoothing process that risks degrading accuracy.

Suppose the maximum perturbation radius satisfies $r \leq n - |\alpha|$, then sampling from $\Delta_r(\alpha)$ may be done in two steps: first, sample the perturbation size $k \sim \{0, 1, \dots, r\}$ with probability $\binom{n-|\alpha|}{k}/|\Delta_r(\alpha)|$, where $|\Delta_r(\alpha)| = \sum_{i=0}^{r}\binom{n-|\alpha|}{i}$; then, uniformly select k among the $n - |\alpha|$ zero positions available in α and set them to one. We caution that naively computing these terms may lead to numerical instability and so we recommend using a Gumbel softmax reparametrization [24] to sample in the log probability space.

[2] Specifically, $\tilde{f}$ is Lipschitz with respect to the masking of features. For any $f : \mathbb{R}^n \to [0, 1]$, Xue et al. [56] yields a $\tilde{f} : \mathbb{R}^n \to [0, 1]$ where: $\|\tilde{f}(x \odot \alpha) - \tilde{f}(x \odot \alpha')\| \leq \kappa\|\alpha - \alpha'\|_0$. This is then lifted to m classes by standard robustness arguments.

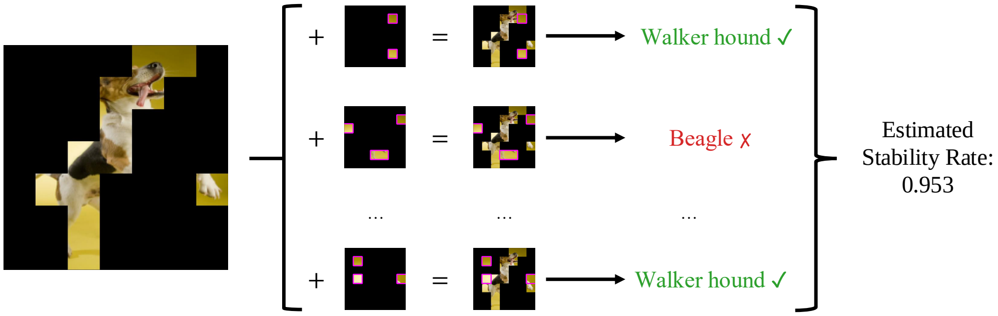


Figure 4: **Estimating the stability rate τ_r.** Given an explanation $\alpha \in \{0,1\}^n$ for a model f and input $x \in \mathbb{R}^n$: First, uniformly sample with replacement the additively perturbed explanations $\alpha' \sim \Delta_r(\alpha)$. Then, let $\hat{\tau}_r$ be the rate at which the perturbed explanation match the explanation's prediction, i.e., $\hat{\tau}_r = \frac{1}{N}\sum_{\alpha'} \mathbf{1}[f(x \odot \alpha') \cong f(x \odot \alpha)]$, where N is the number of samples. If $N \geq \frac{\log(2/\delta)}{2\varepsilon^2}$, then with probability at least $1-\delta$, the estimation accuracy is $|\hat{\tau}_r - \tau_r| \leq \varepsilon$. In this example, $\hat{\tau}_r = 0.953$.

4 Smoothing Can Improve Soft Stability

Smoothing is commonly used to certify robustness guarantees, but often at a high cost to the smoothed classifier's accuracy. Interestingly, however, we found that a milder variant of the smoothing proposed in [56] can improve soft stability while incurring only a minor accuracy trade-off. We emphasize that the soft stability certification algorithm in Theorem 3.1 does *not* require smoothing. Rather, mildly smoothing the model can empirically improve stability rates.

We now introduce the smoothing operator used to certify hard stability in Xue et al. [56], wherein the main idea is for the smoothed classifier to be more robust to the inclusion and exclusion of features. This is achieved by randomly masking features in the following process.

Definition 4.1 (Random Masking[3])**.** For any classifier f and smoothing parameter $\lambda \in [0,1]$, define the random masking operator M_λ as:

$$M_\lambda f(x) = \mathop{\mathbb{E}}_{z \sim \mathrm{Bern}(\lambda)^n} f(x \odot z), \quad \text{where } z_1, \ldots, z_n \sim \mathrm{Bern}(\lambda) \text{ are i.i.d. samples.} \tag{6}$$

The smoothing parameter λ is the probability that any given feature is kept. That is, each feature is randomly masked (zeroed, dropped) with probability $1-\lambda$. We say that smoothing becomes stronger as λ shrinks: at $\lambda = 1$, no smoothing occurs because $M_1 f(x) = f(x)$; at $\lambda = 1/2$, half the features of $x \odot z$ are zeroed out on average; at $\lambda = 0$, the classifier predicts on an entirely zeroed input because $M_0 f(x) = f(\mathbf{0}_n)$. In the following, we give an overview of our results in Section 4.1 and a more technical presentation in Section 4.2.

4.1 Summary of Theoretical Results

Our main theoretical tooling is Boolean function analysis [41], which studies real-valued functions of Boolean-valued inputs. To connect this with attribution-masked classification: for any classifier $f : \mathbb{R}^n \to \mathbb{R}^m$ and input $x \in \mathbb{R}^n$, define the function $f_x(\alpha) = f(x \odot \alpha)$. Such $f_x : \{0,1\}^n \to \mathbb{R}^m$ is then a Boolean function, for which M_λ is well-defined because of the identity $M_\lambda f(x \odot \alpha) = M_\lambda f_x(\alpha)$.

To simplify our analysis, we consider a relaxed notion of prediction agreement. Specifically, we consider classifiers of the form $f_x : \{0,1\}^n \to \mathbb{R}$, where for $\alpha' \sim \Delta_r(\alpha)$ let:

$$f_x(\alpha') \cong f_x(\alpha) \quad \text{if} \quad |f_x(\alpha') - f_x(\alpha)| \leq \gamma. \tag{7}$$

[3] This is also called multiplicative smoothing because the noise scales the input, unlike standard additive noising [12]. In Xue et al. [56], the noise distribution is not restricted to coordinate-wise i.i.d. Bernoulli sampling over a 2^n-sized space. Instead, introducing structured statistical dependencies enables a deterministic, sample-efficient variant of random masking.

This setup, including the decision boundary distance γ, can be derived from a general m-class classifier once the x and α are given. In summary, we establish the following.

Theorem 4.2 (Smoothed Stability, Informal)**.** *Smoothing improves the stability rate lower bound by a factor of λ. Consider any classifier f_x and attribution α that satisfy Equation* (7)*, and let Q be a quantity that depends on f_x (specifically, its monotone weights of degree $\leq r$), then:*

$$1 - \frac{Q}{\gamma} \leq \tau_r(f_x, \alpha) \implies 1 - \frac{\lambda Q}{\gamma} \leq \tau_r(M_\lambda f_x, \alpha). \tag{8}$$

We present the full version in Theorem B.4. Theoretically, smoothing improves the worst-case stability rate by a factor of λ. Empirically, we observe that smoothed classifiers tend to be more stable. Interestingly, we found it challenging to bound the stability rate of M_λ-smoothed classifiers using standard Boolean analytic techniques, such as those presented in widely used references like [41]. This motivated us to develop novel analytic tooling. We describe these challenges and developments next.

4.2 Challenges with Standard Boolean Analytic Tooling and New Techniques

We now describe the challenges encountered with standard Boolean analytic tooling and introduce novel techniques for analyzing the random masking operator M_λ. We refer to Appendix A for a more extensive exposition on Boolean function analysis and defer additional technical details to Appendix B.

It is common to study Boolean functions through their Fourier expansion. For any function $h : \{0,1\}^n \to \mathbb{R}$, the Fourier expansion exists uniquely and is a summation over the subsets of $[n] = \{1, \dots, n\}$ of the form:

$$h(\alpha) = \sum_{S \subseteq [n]} \widehat{h}(S) \chi_S(\alpha), \quad \text{where } \chi_S(\alpha) = \prod_{i \in S} (-1)^{\alpha_i}, \quad \chi_\emptyset(\alpha) = 1. \tag{9}$$

Here, $\chi_S : \{0,1\}^n \to \{\pm 1\}$ is the standard Fourier basis function with weight $\widehat{h}(S) \in \mathbb{R}$ at subset $S \subseteq [n]$. A key benefit of writing h as a linear combination over $S \subseteq [n]$ is that all the $k = 0, 1, \dots, n$ degree (order) interactions between input bits are made explicit. For example, the 2-bit conjunction (AND) $h(\alpha_1, \alpha_2) = \alpha_1 \wedge \alpha_2$ is uniquely expressible as:

$$h(\alpha_1, \alpha_2) = \frac{1}{4}\chi_\emptyset(\alpha) - \frac{1}{4}\chi_{\{1\}}(\alpha) - \frac{1}{4}\chi_{\{2\}}(\alpha) + \frac{1}{4}\chi_{\{1,2\}}(\alpha), \tag{10}$$

A common way to study operators on Boolean functions is to examine how they affect each basis function in an expansion. The random masking operator M_λ acts as follows with respect to the standard basis.

Theorem 4.3. *For any standard basis function χ_S and smoothing parameter $\lambda \in [0, 1]$,*

$$M_\lambda \chi_S(\alpha) = \sum_{T \subseteq S} \lambda^{|T|} (1 - \lambda)^{|S - T|} \chi_T(\alpha). \tag{11}$$

For any function $h : \{0,1\}^n \to \mathbb{R}$, its smoothed variant M_λ has expansion

$$M_\lambda h(\alpha) = \sum_{T \subseteq [n]} \widehat{M_\lambda h}(T) \chi_T(\alpha), \quad \textit{where } \widehat{M_\lambda h}(T) = \lambda^{|T|} \sum_{S \supseteq T} (1 - \lambda)^{|S - T|} \widehat{h}(S). \tag{12}$$

Qualitatively, M_λ rapidly decays the weights at higher-order terms by shifting them onto lower-order terms according to a binomial distribution $\mathsf{Bin}(|S|, \lambda)$. However, this behavior introduces significant complexity in the algebraic manipulations and is distinct from that of other operators commonly studied in literature, which makes it hard to adapt existing techniques for stability analysis. [4]

[4] The prototypical smoothing operator is random flipping: for $0 \leq \rho \leq 1$, define $T_\rho h(\alpha) = \mathbb{E}_{z \sim \mathsf{Bern}(q)^n}[h((\alpha + z) \bmod 2)]$, where $q = (1 - \rho)/2$. This point-wise contracts the spectral weight at S via $T_\rho \chi_S(\alpha) = \rho^{|S|} \chi_S(\alpha)$, which is distinct from redistribution.

Although stability results could, in principle, be derived using the standard Boolean basis, we introduce a novel *monotone basis* that yields cleaner analytical expressions. This development is motivated by the fact that the monotone basis is better equipped to describe properties that depend on the inclusion and exclusion of features, such as the additive perturbations from $\Delta_r(\alpha)$ or the random deletions from M_λ.

Definition 4.4 (Monotone Basis)**.** For any subset $T \subseteq [n]$, define its respective monotone basis function as:

$$\mathbf{1}_T(\alpha) = \begin{cases} 1, & \text{if } \alpha_i = 1 \text{ for all } i \in T \text{ (all features of } T \text{ are present)}, \\ 0, & \text{otherwise (any feature of } T \text{ is absent).} \end{cases} \tag{13}$$

The monotone basis provides a direct encoding of set inclusion, where the earlier example of $h(\alpha_1, \alpha_2) = \alpha_1 \wedge \alpha_2$ is now concisely represented as $h(\alpha) = \mathbf{1}_{\{1,2\}}(\alpha)$. Similar to the standard basis, the monotone basis also admits a unique *monotone expansion* for any $h : \{0,1\}^n \to \mathbb{R}$ of the form:

$$h(\alpha) = \sum_{T \subseteq [n]} \widetilde{h}(T) \mathbf{1}_T(\alpha), \quad \widetilde{h}(T) = h(T) - \sum_{S \subsetneq T} \widetilde{h}(S), \quad \widetilde{h}(\emptyset) = h(\mathbf{0}_n), \tag{14}$$

where $\widetilde{h}(T)$ are the recursively defined monotone weights at each $T \subseteq [n]$, with $h(T)$ being the evaluation of h on the natural $\{0,1\}^n$-valued representation of T. A key property of the monotone basis is that the action of M_λ is now point-wise at each T.

Theorem 4.5. *For any function $h : \{0,1\}^n \to \mathbb{R}$, subset $T \subseteq [n]$, and smoothing parameter $\lambda \in [0,1]$,*

$$\widetilde{M_\lambda h}(T) = \lambda^{|T|} \widetilde{h}(T), \tag{15}$$

where $\widetilde{M_\lambda h}(T)$ and $\widetilde{h}(T)$ are the monotone basis coefficients of $M_\lambda h$ and h at T, respectively.

In the monotone basis, smoothing exponentially decays the weight at each $T \subseteq [n]$ by a factor of $\lambda^{|T|}$. This is algebraically simpler than the redistribution of weights in Theorem 4.3 and aligns more closely with the motifs in existing techniques. As previewed in Theorem 4.2 (full version in Theorem B.4), we use the monotone basis to bound the stability rate of smoothed classifiers, where Q is a value that depends on $\{\widetilde{h}(T) : |T| \leq r\}$, i.e., the monotone weights of degree $\leq r$. We refer to Appendix B for additional details.

5 Experiments

We evaluate soft and hard stability guarantees on vision and language models, investigate how smoothing affects classifier stability and accuracy, and analyze how different feature attribution methods perform under these conditions. Our findings reinforce that soft stability provides significantly larger certificates than hard stability, while mild smoothing often improves soft stability without substantial accuracy degradation. We summarize our results here and defer full experimental details to Appendix C.

Setup We evaluated a combination of vision and language models: for vision models, we used Vision Transformer (VIT) [15] and ResNet18 [21], while for language models, we used RoBERTa[33]. Our datasets included a 2000-image subset of ImageNet and six subsets of TweetEval (emoji, emotion, hate, irony, offensive, sentiment; total 10653 samples). The images were of size $3 \times 224 \times 224$, which we segmented into patches of size 16×16, for a total of $n = (224/16)^2 = 196$ features per image. For text, each token was treated as one feature. We considered five feature attribution methods: LIME [44], SHAP [34], Integrated Gradients (IntGrad) [51], MFABA [61], and a baseline where random features are selected. We binarized real-valued attributions by selecting the top-25% features in their ranking unless specified otherwise. Our experiments were conducted using NVIDIA GeForce RTX 3090 and NVIDIA RTX A6000 GPUs.

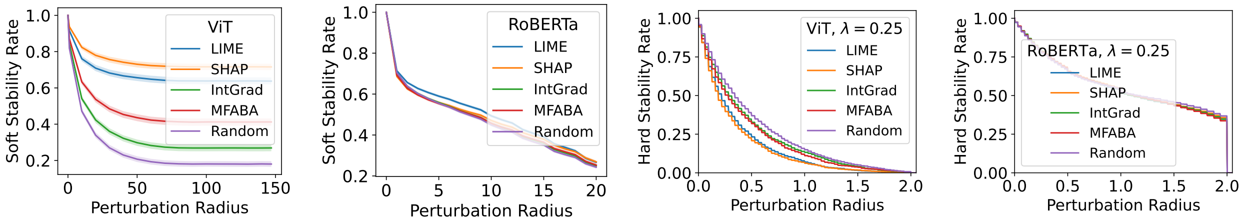


Figure 5: **Soft stability certifies more than hard stability.** We show the attainable soft (far left, center left) and hard (center right, far right) rates, where we observe that hard stability certificates are only effective at smaller radii. We used a smoothing parameter of $\lambda = 0.25$ to certify hard stability. We selected 25% of the top-scoring features from each method as the explanation. We used 2000 samples from ImageNet for ViT, and we used TweetEval for RoBERTa. Because we implement soft stability estimation via a randomized algorithm ($\varepsilon = \delta = 0.1$), we used 1000 bootstrap iterations to compute a 95% confidence interval.

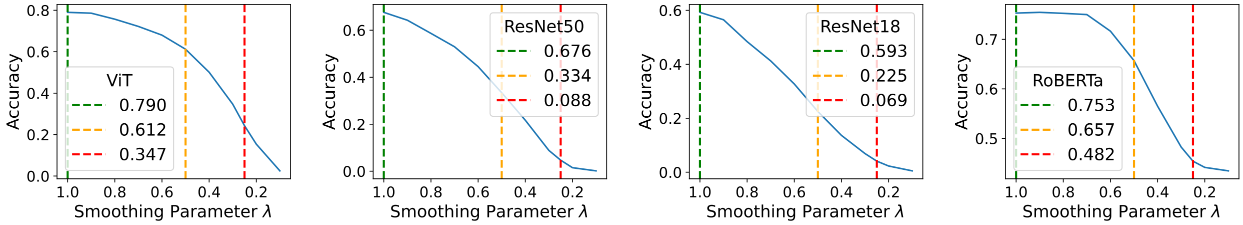


Figure 6: **Accuracy decreases as smoothing intensifies.** We report accuracy values at key thresholds: ($\lambda = 1.0$, green) the original, unmodified classifier; ($\lambda = 0.5$, orange) the threshold above which hard stability certificates are not attainable; ($\lambda = 0.25$, red) at which hard stability can only certify additive perturbations of up to 2 features. We used 2000 samples from ImageNet and 2000 samples from TweetEval.

Question 1: How much do soft stability certificates improve over hard stability? We compare soft stability guarantees with hard stability across different inputs and radii. While soft stability is measured via the (soft) stability rate, hard stability is defined by its (hard) certifiable radius. To facilitate comparison, we define a hard stability rate as follows:

$$\text{Hard stability rate at radius } r = \frac{|\{(x, \alpha) : \text{CertifiedRadius}(M_\lambda f_x, \alpha) \geq r\}|}{\text{Total number of } x\text{'s}}. \tag{16}$$

We present results in Figure 5 and describe the certified radius computation in Theorem C.1. We use $\delta = \varepsilon = 0.1$ to estimate soft stability, which means that 150 samples are needed per Theorem 3.1 to ensure an estimation accuracy of ε with probability at least $1 - \delta$. For hard stability, we set $\lambda = 0.25$, meaning that the maximum certifiable radius is only 2.0 by Theorem C.1. We note that because hard stability certification requires smoothed classifiers, the shown guarantees are with respect to a less accurate classifier.

We find that soft stability consistently yields larger certified radii than hard stability across all models and attribution methods. For vision, LIME and SHAP achieve the highest soft stability values. For language, this distinction is less clear, suggesting that these explanation methods are approximately comparable on RoBERTa and TweetEval with respect to this metric. We give further comparison of soft and hard stability in Appendix C.1 and Appendix C.2.

Question 2: How much does smoothing degrade accuracy? We analyze the impact of smoothing on classifier accuracy and plot our results in Figure 6, where we note three key values: the original, unmodified classifier accuracy ($\lambda = 1.0$), the largest smoothing parameter usable in the certification of hard stability ($\lambda = 0.5$), and ($\lambda = 0.25$), the smoothing parameter used in many hard stability experiments of [56].

For evaluation, we used 64 Bernoulli samples when computing the smoothed classifier. We observe that transformer-based models (ViT, RoBERTa) experience a more graceful accuracy decline as smoothing

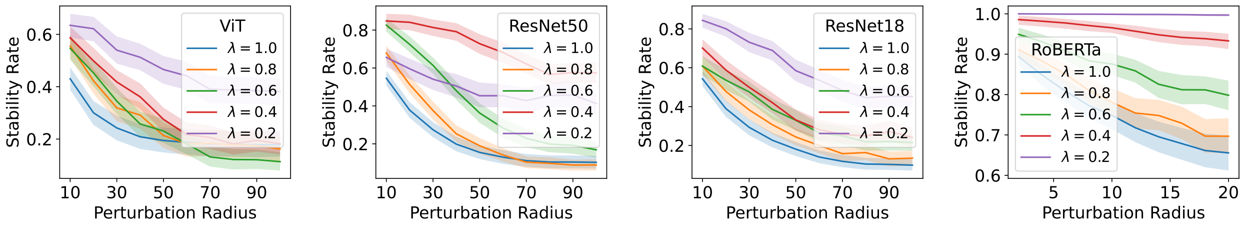


Figure 7: **Smoothing can improve stability.** However, the improvement is not necessarily monotonic, and some models benefit more than others. We used 200 samples from ImageNet and 200 samples from TweetEval that had at least 40 tokens. For each input, we randomly selected 25% of the input features as the initial attribution. We ran bootstrap with 1000 iterations to compute the 95% confidence interval.

intensifies. We suspect this is because random masking strategies similar to our perturbation scheme are commonly employed in their training. We remark that fine-tuning more aggressively on masked inputs is known to improve a model's resilience to this perturbation, as shown in [56].

Question 3: How does smoothing affect soft stability? We study the effect of the smoothing parameter on the stability rate. We show results in Figure 7, where we used 200 samples from our subset of ImageNet and 200 samples from TweetEval that had at least 40 tokens. We randomly selected 25% of the features from each input as our attribution. We show the confidence 95% confidence interval over 1000 iterations of bootstrap sampling. Our conservative lower bound filter on the text token length was to ensure that up to 20 additive perturbations could be meaningfully applied.

We observe that smoothing generally improves stability, particularly for ResNet50 and ResNet18. Mildly smoothed ($\lambda = 0.8, 0.6$) models typically have good stability gains for smaller perturbations. While improvement exists for every model, the improvement is not necessarily monotone in the smoothing parameter λ nor uniform in the perturbation radius r.

6 Related Work

Feature-based Explanations Feature attributions have long been used in explainability and remain popular. Early examples include gradient saliency [49], LIME [44], SHAP [34], and Integrated Gradients [51]. More recent works include DIME [37], LAFA [59], CAFE [13], DoRaR [43], MFABA [61], various Shapley value-based methods [50], and methods based on influence functions [8, 29]. Moreover, while feature attributions are commonly associated with vision models, they are also used in language modeling [36]. For general surveys on explainability, we refer to Milani et al. [38], Schwalbe and Finzel [48]. For explainability in medicine, we refer to Klauschen et al. [28], Patrício et al. [42]. For explainability in law, we refer to [4, 45].

Evaluating Feature Attributions Although feature attributions are popular, their correctness and usefulness have often been called into question [1, 14, 27]. This is because each attribution method computes importance by a different measure, which may not necessarily be indicative of the underlying model behavior [2, 60], as well as theoretical results on their limitations [9]. This has prompted a large number of evaluation metrics for feature attributions [3, 25, 40, 46], in particular for various notions of robustness [18, 26]. Perturbation-based robustness metrics similar to ours include incremental insertion [40] and ranking stability [19].

Certifying Feature Attributions While many empirical metrics exist, there is also growing interest in ensuring that feature attributions are well-behaved through formal guarantees. There exists work on certifying the robustness properties of adding [56] and removing [31] features from an attribution. There is also work on selecting feature sets that are provably optimal with respect to some metric [10], such as in their ranking [19]. However, the literature on explicit guarantees for feature attributions is still emergent.

7 Discussion

This work aims to make post hoc explanations more reliable by introducing soft stability, a probabilistic and model-agnostic notion of robustness. While prior certification methods tend to be overly conservative or tied to smoothed models, our approach yields stronger, practically useful guarantees. We now discuss some broader implications and promising extensions.

Boolean Function Analysis in Explainability Boolean analytic techniques are well-suited for explainability, as many manipulations in this domain are often discrete. This makes Boolean function analysis a natural tool for both developing new algorithms and analyzing existing ones. In our case, this approach enabled us to shift away from traditional continuous Lipschitz-based robustness analysis to provide a discrete perspective. Our findings suggest that similar techniques could be valuable in other machine learning tasks, particularly those involving voting, aggregation, or other discrete perturbation schemes.

Future Directions An exciting direction for future work is adaptive smoothing, where the smoothing parameter is tuned based on feature importance or model confidence, for example, applying stronger smoothing to uncertain predictions. Another is stability-aware training, where models are explicitly regularized to produce more stable attributions. Exploring connections between stability and generalization may also yield insights, particularly if higher soft stability aligns with better model reliability. Alternative smoothing schemes based on attribution rankings [19] merit investigation as well. Finally, since stability and adversarial robustness both capture sensitivity to perturbations, future work could study when stability naturally emerges as the appropriate robustness notion.

8 Conclusion

We introduce soft stability, a probabilistic relaxation of hard stability that provides a more flexible and efficient way to certify the robustness of feature attributions. Unlike hard stability, soft stability is model-agnostic, sample-efficient, and does not require destructively modifying the classifier. Interestingly, we show that mild smoothing can improve the soft stability certificate of classifiers while incurring only a small cost to accuracy. We study this phenomenon from the perspective of Boolean function analysis and present novel characterizations and techniques that would be of interest to explainability researchers. Furthermore, we validate our theory through experiments on vision and language tasks.

Acknowledgements This research was partially supported by the ARPA-H program on Safe and Explainable AI under the grant D24AC00253-00, by NSF award CCF 2313010, by the AI2050 program at Schmidt Sciences, by an Amazon Research Award Fall 2023, and by an OpenAI SuperAlignment grant.

References

[1] Julius Adebayo, Justin Gilmer, Michael Muelly, Ian Goodfellow, Moritz Hardt, and Been Kim. Sanity checks for saliency maps. *Advances in neural information processing systems*, 31, 2018.

[2] Julius Adebayo, Michael Muelly, Harold Abelson, and Been Kim. Post hoc explanations may be ineffective for detecting unknown spurious correlation. In *International Conference on Learning Representations*, 2022.

[3] Chirag Agarwal, Satyapriya Krishna, Eshika Saxena, Martin Pawelczyk, Nari Johnson, Isha Puri, Marinka Zitnik, and Himabindu Lakkaraju. Openxai: Towards a transparent evaluation of model explanations. *Advances in neural information processing systems*, 35:15784–15799, 2022.

[4] Kasun Amarasinghe, Kit T Rodolfa, Hemank Lamba, and Rayid Ghani. Explainable machine learning for public policy: Use cases, gaps, and research directions. *Data & Policy*, 5:e5, 2023.

[5] Cem Anil, James Lucas, and Roger Grosse. Sorting out lipschitz function approximation. In *International Conference on Machine Learning*, pages 291–301. PMLR, 2019.

[6] Staffan Arvidsson McShane, Ulf Norinder, Jonathan Alvarsson, Ernst Ahlberg, Lars Carlsson, and Ola Spjuth. Cpsign: conformal prediction for cheminformatics modeling. *Journal of Cheminformatics*, 16(1): 75, 2024.

[7] Pepa Atanasova. A diagnostic study of explainability techniques for text classification. In *Accountable and Explainable Methods for Complex Reasoning over Text*, pages 155–187. Springer, 2024.

[8] Samyadeep Basu, Philip Pope, and Soheil Feizi. Influence functions in deep learning are fragile. *arXiv preprint arXiv:2006.14651*, 2020.

[9] Blair Bilodeau, Natasha Jaques, Pang Wei Koh, and Been Kim. Impossibility theorems for feature attribution. *Proceedings of the National Academy of Sciences*, 121(2):e2304406120, 2024.

[10] Guy Blanc, Jane Lange, and Li-Yang Tan. Provably efficient, succinct, and precise explanations. *Advances in Neural Information Processing Systems*, 34:6129–6141, 2021.

[11] Diogo V Carvalho, Eduardo M Pereira, and Jaime S Cardoso. Machine learning interpretability: A survey on methods and metrics. *Electronics*, 8(8):832, 2019.

[12] Jeremy Cohen, Elan Rosenfeld, and Zico Kolter. Certified adversarial robustness via randomized smoothing. In *international conference on machine learning*, pages 1310–1320. PMLR, 2019.

[13] Adam Dejl, Hamed Ayoobi, Matthew Williams, and Francesca Toni. Cafe: Conflict-aware feature-wise explanations. *arXiv preprint arXiv:2310.20363*, 2023.

[14] Jonathan Dinu, Jeffrey Bigham, and J Zico Kolter. Challenging common interpretability assumptions in feature attribution explanations. *arXiv preprint arXiv:2012.02748*, 2020.

[15] Alexey Dosovitskiy. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[16] Jamil Fayyad, Shadi Alijani, and Homayoun Najjaran. Empirical validation of conformal prediction for trustworthy skin lesions classification. *Computer Methods and Programs in Biomedicine*, page 108231, 2024.

[17] Mahyar Fazlyab, Alexander Robey, Hamed Hassani, Manfred Morari, and George Pappas. Efficient and accurate estimation of lipschitz constants for deep neural networks. *Advances in neural information processing systems*, 32, 2019.

[18] Yuyou Gan, Yuhao Mao, Xuhong Zhang, Shouling Ji, Yuwen Pu, Meng Han, Jianwei Yin, and Ting Wang. "is your explanation stable?" a robustness evaluation framework for feature attribution. In *Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security*, pages 1157–1171, 2022.

[19] Jeremy Goldwasser and Giles Hooker. Provably stable feature rankings with shap and lime. *arXiv preprint arXiv:2401.15800*, 2024.

[20] Peter Hase, Harry Xie, and Mohit Bansal. The out-of-distribution problem in explainability and search methods for feature importance explanations. *Advances in neural information processing systems*, 34: 3650–3666, 2021.

[21] 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.

[22] Todd Huster, Cho-Yu Jason Chiang, and Ritu Chadha. Limitations of the lipschitz constant as a defense against adversarial examples. In *ECML PKDD 2018 Workshops: Nemesis 2018, UrbReas 2018, SoGood 2018, IWAISe 2018, and Green Data Mining 2018, Dublin, Ireland, September 10-14, 2018, Proceedings 18*, pages 16–29. Springer, 2019.

[23] Saachi Jain, Hadi Salman, Eric Wong, Pengchuan Zhang, Vibhav Vineet, Sai Vemprala, and Aleksander Madry. Missingness bias in model debugging. In *International Conference on Learning Representations*, 2022.

[24] Eric Jang, Shixiang Gu, and Ben Poole. Categorical reparameterization with gumbel-softmax. *arXiv preprint arXiv:1611.01144*, 2016.

[25] Helen Jin, Shreya Havaldar, Chaehyeon Kim, Anton Xue, Weiqiu You, Helen Qu, Marco Gatti, Daniel Hashimoto, Bhuvnesh Jain, Amin Madani, Masao Sako, Lyle Ungar, and Eric Wong. The fix benchmark: Extracting features interpretable to experts. *arXiv preprint arXiv:2409.13684*, 2024.

[26] Sandesh Kamath, Sankalp Mittal, Amit Deshpande, and Vineeth N Balasubramanian. Rethinking robustness of model attributions. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 2688–2696, 2024.

[27] Pieter-Jan Kindermans, Sara Hooker, Julius Adebayo, Maximilian Alber, Kristof T Schütt, Sven Dähne, Dumitru Erhan, and Been Kim. The (un) reliability of saliency methods. In *Explainable AI: Interpreting, Explaining and Visualizing Deep Learning*, pages 267–280. Springer, 2019.

[28] Frederick Klauschen, Jonas Dippel, Philipp Keyl, Philipp Jurmeister, Michael Bockmayr, Andreas Mock, Oliver Buchstab, Maximilian Alber, Lukas Ruff, Grégoire Montavon, et al. Toward explainable artificial intelligence for precision pathology. *Annual Review of Pathology: Mechanisms of Disease*, 19(1):541–570, 2024.

[29] Pang Wei Koh and Percy Liang. Understanding black-box predictions via influence functions. In *International conference on machine learning*, pages 1885–1894. PMLR, 2017.

[30] Alexander J Levine and Soheil Feizi. Improved, deterministic smoothing for l_1 certified robustness. In *International Conference on Machine Learning*, pages 6254–6264. PMLR, 2021.

[31] Chris Lin, Ian Covert, and Su-In Lee. On the robustness of removal-based feature attributions. *Advances in Neural Information Processing Systems*, 36, 2024.

[32] Lars Lindemann, Matthew Cleaveland, Gihyun Shim, and George J Pappas. Safe planning in dynamic environments using conformal prediction. *IEEE Robotics and Automation Letters*, 2023.

[33] Yinhan Liu. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 364, 2019.

[34] Scott M Lundberg and Su-In Lee. A unified approach to interpreting model predictions. *Advances in neural information processing systems*, 30, 2017.

[35] Qing Lyu, Shreya Havaldar, Adam Stein, Li Zhang, Delip Rao, Eric Wong, Marianna Apidianaki, and Chris Callison-Burch. Faithful chain-of-thought reasoning. In *Proceedings of the 13th International Joint Conference on Natural Language Processing and the 3rd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 305–329, 2023.

[36] Qing Lyu, Marianna Apidianaki, and Chris Callison-Burch. Towards faithful model explanation in nlp: A survey. *Computational Linguistics*, pages 1–67, 2024.

[37] Yiwei Lyu, Paul Pu Liang, Zihao Deng, Ruslan Salakhutdinov, and Louis-Philippe Morency. Dime: Fine-grained interpretations of multimodal models via disentangled local explanations. In *Proceedings of the 2022 AAAI/ACM Conference on AI, Ethics, and Society*, pages 455–467, 2022.

[38] Stephanie Milani, Nicholay Topin, Manuela Veloso, and Fei Fang. Explainable reinforcement learning: A survey and comparative review. *ACM Computing Surveys*, 56(7):1–36, 2024.

[39] Tim Miller. Explanation in artificial intelligence: Insights from the social sciences. *Artificial intelligence*, 267:1–38, 2019.

[40] Meike Nauta, Jan Trienes, Shreyasi Pathak, Elisa Nguyen, Michelle Peters, Yasmin Schmitt, Jörg Schlötterer, Maurice Van Keulen, and Christin Seifert. From anecdotal evidence to quantitative evaluation methods: A systematic review on evaluating explainable ai. *ACM Computing Surveys*, 55 (13s):1–42, 2023.

[41] Ryan O'Donnell. *Analysis of boolean functions*. Cambridge University Press, 2014.

[42] Cristiano Patrício, João C Neves, and Luís F Teixeira. Explainable deep learning methods in medical image classification: A survey. *ACM Computing Surveys*, 56(4):1–41, 2023.

[43] Dong Qin, George T Amariucai, Daji Qiao, Yong Guan, and Shen Fu. A comprehensive and reliable feature attribution method: Double-sided remove and reconstruct (dorar). *Neural Networks*, 173:106166, 2024.

[44] 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.

[45] Karen McGregor Richmond, Satya M Muddamsetty, Thomas Gammeltoft-Hansen, Henrik Palmer Olsen, and Thomas B Moeslund. Explainable ai and law: an evidential survey. *Digital Society*, 3(1):1, 2024.

[46] Yao Rong, Tobias Leemann, Vadim Borisov, Gjergji Kasneci, and Enkelejda Kasneci. A consistent and efficient evaluation strategy for attribution methods. In *International Conference on Machine Learning*, pages 18770–18795. PMLR, 2022.

[47] Wojciech Samek, Alexander Binder, Grégoire Montavon, Sebastian Lapuschkin, and Klaus-Robert Müller. Evaluating the visualization of what a deep neural network has learned. *IEEE transactions on neural networks and learning systems*, 28(11):2660–2673, 2016.

[48] Gesina Schwalbe and Bettina Finzel. A comprehensive taxonomy for explainable artificial intelligence: a systematic survey of surveys on methods and concepts. *Data Mining and Knowledge Discovery*, 38(5): 3043–3101, 2024.

[49] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. *arXiv preprint arXiv:1312.6034*, 2013.

[50] Mukund Sundararajan and Amir Najmi. The many shapley values for model explanation. In *International conference on machine learning*, pages 9269–9278. PMLR, 2020.

[51] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *International conference on machine learning*, pages 3319–3328. PMLR, 2017.

[52] Aladin Virmaux and Kevin Scaman. Lipschitz regularity of deep neural networks: analysis and efficient estimation. *Advances in Neural Information Processing Systems*, 31, 2018.

[53] Jorg Wagner, Jan Mathias Kohler, Tobias Gindele, Leon Hetzel, Jakob Thaddaus Wiedemer, and Sven Behnke. Interpretable and fine-grained visual explanations for convolutional neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9097–9107, 2019.

[54] Weibin Wu, Yuxin Su, Xixian Chen, Shenglin Zhao, Irwin King, Michael R Lyu, and Yu-Wing Tai. Towards global explanations of convolutional neural networks with concept attribution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8652–8661, 2020.

[55] Anton Xue, Lars Lindemann, Alexander Robey, Hamed Hassani, George J Pappas, and Rajeev Alur. Chordal sparsity for lipschitz constant estimation of deep neural networks. In *2022 IEEE 61st Conference on Decision and Control (CDC)*, pages 3389–3396. IEEE, 2022.

[56] Anton Xue, Rajeev Alur, and Eric Wong. Stability guarantees for feature attributions with multiplicative smoothing. *Advances in Neural Information Processing Systems*, 36, 2024.

[57] Fanghua Ye, Mingming Yang, Jianhui Pang, Longyue Wang, Derek F Wong, Emine Yilmaz, Shuming Shi, and Zhaopeng Tu. Benchmarking llms via uncertainty quantification. *arXiv preprint arXiv:2401.12794*, 2024.

[58] 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.

[59] Sheng Zhang, Jin Wang, Haitao Jiang, and Rui Song. Locally aggregated feature attribution on natural language model understanding. *arXiv preprint arXiv:2204.10893*, 2022.

[60] Yilun Zhou, Serena Booth, Marco Tulio Ribeiro, and Julie Shah. Do feature attribution methods correctly attribute features? In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 9623–9633, 2022.

[61] Zhiyu Zhu, Huaming Chen, Jiayu Zhang, Xinyi Wang, Zhibo Jin, Minhui Xue, Dongxiao Zhu, and Kim-Kwang Raymond Choo. Mfaba: A more faithful and accelerated boundary-based attribution method for deep neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 17228–17236, 2024.

A Analysis of Smoothing with Standard Techniques

In this appendix, we analyze the smoothing operator M_λ using classical tools from Boolean function analysis. Specifically, we study how smoothing redistributes the spectral mass of a function by examining its action on standard Fourier basis functions. This sets up the foundation for our later motivation to introduce a more natural basis in Appendix B. First, recall the definition of the random masking-based smoothing operator.

Definition A.1 (Xue et al. [56])**.** For any classifier $f : \mathbb{R}^n \to \mathbb{R}^m$ and smoothing parameter $\lambda \in [0,1]$, define the random masking operator M_λ as:

$$M_\lambda f(x) = \mathop{\mathbb{E}}_{z \sim \mathrm{Bern}(\lambda)^n} f(x \odot z), \quad \text{where } z_1, \dots, z_n \sim \mathrm{Bern}(\lambda) \text{ are i.i.d. samples.} \tag{17}$$

To analyze $M\lambda$ in the Boolean function setting, we fix the input $x \in \mathbb{R}^n$ and view the masked classifier $f_x(\alpha) = f(x \odot \alpha)$ as a Boolean function $f_x : 0, 1^n \to \mathbb{R}^m$. This perspective allows us to apply tools from Fourier analysis on $0, 1^n$ to study the effect of smoothing on the classifier's output. In particular, we have the following relation for M_λ:

$$M_\lambda f(x \odot \alpha) = M_\lambda f_x(\alpha) = M_\lambda f_{x \odot \alpha}(\mathbf{1}_n). \tag{18}$$

This relation is useful from an explainability perspective because it means that features not selected by α (the x_i where $\alpha_i = 0$) will not be seen by the classifier. In other words, this prevents a form of information leakage when evaluating the informativeness of a feature selection.

A.1 Background on Boolean Function Analysis

A key approach in Boolean function analysis is to study functions of the form $h : \{0,1\}^n \to \mathbb{R}$ by their unique *Fourier expansion*:

$$h(\alpha) = \sum_{S \subseteq [n]} \widehat{h}(S) \chi_S(\alpha), \quad \text{where } \chi_S(\alpha) = \prod_{i \in S} (-1)^{\alpha_i}, \quad \chi_\emptyset(\alpha) = 1 \tag{19}$$

where each subset $S \subseteq [n]$ indexes a *basis function* $\chi_S : \{0,1\}^n \to \{\pm 1\}$ with *weight* $\widehat{h}(S) \in \mathbb{R}$. Crucially, the basis functions are orthonormal on $\{0,1\}^n$ in that:

$$\langle \chi_S, \chi_T \rangle = \mathop{\mathbb{E}}_{\alpha \sim \mathsf{Bern}(1/2)^n} [\chi_S(\alpha)\chi_T(\alpha)] = \frac{1}{2^n} \sum_{\alpha \in \{0,1\}^n} \chi_S(\alpha)\chi_T(\alpha) = \begin{cases} 1, & \text{if } S = T, \\ 0, & \text{if } S \neq T. \end{cases} \tag{20}$$

This lets us express the relation between h and its weights $\widehat{h}(S)$ as follows:

$$\widehat{h}(S) = \langle h, \chi_S \rangle = \frac{1}{2^n} \sum_{\alpha \in \{0,1\}^n} h(\alpha)\chi_S(\alpha). \tag{21}$$

For example, one can check that the function $h(\alpha_1, \alpha_2) = \alpha_1 \wedge \alpha_2$ may be uniquely expressed as:

$$h(\alpha_1, \alpha_2) = \frac{1}{4}\chi_\emptyset(\alpha) - \frac{1}{4}\chi_{\{1\}}(\alpha) - \frac{1}{4}\chi_{\{2\}}(\alpha) + \frac{1}{4}\chi_{\{1,2\}}(\alpha). \tag{22}$$

A.2 Basic Results in the Standard Basis

We now study how smoothing affects stability by analyzing how M_λ transforms Boolean functions in the standard Fourier basis. A common approach is to examine how M_λ acts on each basis function χ_S, and we show that smoothing causes a spectral mass shift from higher-order to lower-order terms.

Lemma A.2. *For any standard basis function χ_S and smoothing parameter $\lambda \in [0,1]$,*

$$M_\lambda \chi_S(\alpha) = \sum_{T \subseteq S} \lambda^{|T|}(1-\lambda)^{|S-T|}\chi_T(\alpha). \tag{23}$$

Proof. We first expand the definition of $\chi_S(\alpha)$ to derive:

$$
\begin{aligned}
M_\lambda \chi_S(\alpha) &= \mathop{\mathbb{E}}_{z} \prod_{i \in S} (-1)^{\alpha_i z_i} && (24) \\
&= \prod_{i \in S} \mathop{\mathbb{E}}_{z} (-1)^{\alpha_i z_i} && \text{(by independence of } z_1, \dots, z_n\text{)} \\
&= \prod_{i \in S} [(1-\lambda) + \lambda(-1)^{\alpha_i}], && (25)
\end{aligned}
$$

We then use the distributive property (i.e., expanding products over sums) to rewrite the product $\prod_{i \in S}(\cdots)$ as a summation over $T \subseteq S$ to get

$$
\begin{aligned}
M_\lambda \chi_S(\alpha) &= \sum_{T \subseteq S} \left(\prod_{j \in S - T} (1-\lambda) \right) \left(\prod_{i \in T} \lambda(-1)^{\alpha_i} \right) && (26) \\
&= \sum_{T \subseteq S} (1-\lambda)^{|S-T|} \lambda^{|T|} \chi_T(\alpha), && (27)
\end{aligned}
$$

where T acts like an enumeration over the 2^n choices of $z \in \{0,1\}^n$ and recall that $\chi_T(\alpha) = \prod_{i \in T}(\alpha)$. □

In other words, M_λ redistributes the Fourier weight at each basis χ_S over to the $2^{|S|}$ subsets $T \subseteq S$ according to a binomial distribution $\mathsf{Bin}(|S|, \lambda)$. Because this redistribution acts linearly on the input, we can visualize M_λ as a $\mathbb{R}^{2^n \times 2^n}$ upper-triangular matrix whose entries are indexed by $T, S \subseteq [n]$, such that

$$
(M_\lambda)_{T,S} = \begin{cases} \lambda^{|T|}(1-\lambda)^{|S-T|}, & \text{if } T \subseteq S, \\ 0, & \text{otherwise.} \end{cases} \tag{28}
$$

Using the earlier example of $h(\alpha_1, \alpha_2) = \alpha_1 \wedge \alpha_2$, the Fourier coefficients of $M_\lambda h$ may be expressed as:

$$
\begin{bmatrix} \widehat{M_\lambda h}(\emptyset) \\ \widehat{M_\lambda h}(\{1\}) \\ \widehat{M_\lambda h}(\{2\}) \\ \widehat{M_\lambda h}(\{1,2\}) \end{bmatrix}
=
\begin{bmatrix} 1 & (1-\lambda) & (1-\lambda) & (1-\lambda)^2 \\ & \lambda & & \lambda(1-\lambda) \\ & & \lambda & \lambda(1-\lambda) \\ & & & \lambda^2 \end{bmatrix}
\begin{bmatrix} \widehat{h}(\emptyset) \\ \widehat{h}(\{1\}) \\ \widehat{h}(\{2\}) \\ \widehat{h}(\{1,2\}) \end{bmatrix}
= \frac{1}{4} \begin{bmatrix} (2-\lambda)^2 \\ -\lambda(2-\lambda) \\ -\lambda(2-\lambda) \\ \lambda^2 \end{bmatrix} \tag{29}
$$

where recall that $\widehat{h}(S) = 1/4$ for all $S \subseteq \{1,2\}$. For visualization, it is useful to sort the rows and columns of M_λ by inclusion and partition them by degree. As an example with $n = 3$, one may write $M_\lambda \in \mathbb{R}^{8 \times 8}$ as:

$$
M_\lambda \quad \cong \quad
\begin{array}{c||c|ccc|ccc|c}
 & \emptyset & \{1\} & \{2\} & \{3\} & \{1,2\} & \{1,3\} & \{2,3\} & \{1,2,3\} \\
\hline\hline
\emptyset & 1 & (1-\lambda) & (1-\lambda) & (1-\lambda) & (1-\lambda)^2 & (1-\lambda)^2 & (1-\lambda)^2 & (1-\lambda)^3 \\
\hline
\{1\} & & \lambda & & & \lambda(1-\lambda) & \lambda(1-\lambda) & & \lambda(1-\lambda)^2 \\
\{2\} & & & \lambda & & \lambda(1-\lambda) & & \lambda(1-\lambda) & \lambda(1-\lambda)^2 \\
\{3\} & & & & \lambda & & \lambda(1-\lambda) & \lambda(1-\lambda) & \lambda(1-\lambda)^2 \\
\hline
\{1,2\} & & & & & \lambda^2 & & & \lambda^2(1-\lambda) \\
\{1,3\} & & & & & & \lambda^2 & & \lambda^2(1-\lambda) \\
\{2,3\} & & & & & & & \lambda^2 & \lambda^2(1-\lambda) \\
\hline
\{1,2,3\} & & & & & & & & \lambda^3
\end{array} \tag{30}
$$

Because the columns of M_λ sum to 1, we have the identity:

$$
\sum_{T \subseteq [n]} \widehat{M_\lambda h}(T) = \sum_{S \subseteq [n]} \widehat{h}(S), \quad \text{for any function } h : \{0,1\}^n \to \mathbb{R}. \tag{31}
$$

Moreover, M_λ may be interpreted as a downshift operator in the sense that: for each $T \subseteq [n]$, the Fourier coefficient $\widehat{M_\lambda h}(T)$ depends only on those of $\widehat{h}(S)$ for $S \supseteq T$. The following result gives a more precise characterization of each $\widehat{M_\lambda h}(T)$ in the standard basis.

Lemma A.3. *For any function $h : \{0,1\}^n \to \mathbb{R}$ and smoothing parameter $\lambda \in [0,1]$,*

$$M_\lambda h(\alpha) = \sum_{T \subseteq [n]} \widehat{M_\lambda h}(T) \chi_T(\alpha), \quad \textit{where } \widehat{M_\lambda h}(T) = \lambda^{|T|} \sum_{S \supseteq T} (1-\lambda)^{|S-T|} \widehat{h}(S). \tag{32}$$

Proof. This follows by analyzing the T-th row of M_λ as in Equation (30). More specifically, we have:

$$\begin{aligned} M_\lambda h(\alpha) &= \sum_{S \subseteq [n]} \widehat{h}(S) M_\lambda \chi_S(\alpha) && (33) \\ &= \sum_{S \subseteq [n]} \widehat{h}(S) \sum_{T \subseteq S} \lambda^{|T|} (1-\lambda)^{|S-T|} \chi_T(\alpha) && \text{(Lemma A.2)} \\ &= \sum_{T \subseteq [n]} \chi_T(\alpha) \underbrace{\sum_{S \supseteq T} \lambda^{|T|} (1-\lambda)^{|S-T|} \widehat{h}(S)}_{\widehat{M_\lambda h}(T)}, && (34) \end{aligned}$$

where the final step follows by noting that each $\widehat{M_\lambda h}(T)$ depends only on $\widehat{h}(S)$ for $S \supseteq T$. □

The expression derived in Lemma A.3 shows how spectral mass gets redistributed from higher-degree to lower-degree terms. To understand how smoothing affects model robustness, it is helpful to quantify how much of the original function's complexity (i.e., higher-degree interactions) survive after smoothing. The following result upper-bounds the remaining "tail mass" on large sets after smoothing.

Theorem A.4 (Tail Mass Contraction). *For any function $h : \{0,1\}^n \to \mathbb{R}$, smoothing parameter $\lambda \in [0,1]$, and $0 \leq k \leq n$,*

$$\sum_{T : |T| \geq k} |\widehat{M_\lambda h}(T)| \leq \Pr_{X \sim \mathsf{Bin}(n,\lambda)} [X \geq k] \sum_{S : |S| \geq k} |\widehat{h}(S)|. \tag{35}$$

Proof. We first apply Lemma A.3 to expand each $\widehat{M_\lambda h}(T)$ and derive

$$\begin{aligned} \sum_{T : |T| \geq k} |\widehat{M_\lambda h}(T)| &\leq \sum_{T : |T| \geq k} \sum_{S \supseteq T} \lambda^{|T|} (1-\lambda)^{|S-T|} |\widehat{h}(S)| && (36) \\ &= \sum_{S : |S| \geq k} |\widehat{h}(S)| \underbrace{\sum_{j=k}^{|S|} \binom{|S|}{j} \lambda^j (1-\lambda)^{|S|-j}}_{\Pr_{Y \sim \mathsf{Bin}(|S|,\lambda)} [Y \geq k]} && (37) \end{aligned}$$

where we re-indexed the summations to track the contribution of each $|\widehat{h}(S)|$ for $|S| \geq k$. To yield the desired result, we next apply the following inequality of binomial tail CDFs given $|S| \leq n$:

$$\Pr_{Y \sim \mathsf{Bin}(|S|,\lambda)} [Y \geq k] \leq \Pr_{X \sim \mathsf{Bin}(n,\lambda)} [X \geq k]. \tag{38}$$

□

Our analyses with respect to the standard basis provide a first step towards understanding the random masking operator M_λ. However, the weight-mixing from our initial calculations suggests that the standard basis may be challenging to work with.

A.3 Analysis in the p-Biased Basis

While analysis on the standard Fourier basis reveals interesting properties about M_λ, it suggests that this may not be the natural choice of basis in which to analyze random masking. Principally, this is because each $M_\lambda \chi_S$ is expressed as a linear combination of χ_T where $T \subseteq S$. By "natural", we instead aim to express the image of M_λ as a single term. One partial attempt is an extension of the standard basis, known as the p-biased basis, which is defined as follows.

Definition A.5 (p-Biased Basis)**.** For any subset $S \subseteq [n]$, define its corresponding p-biased function basis as:

$$\chi_S^p(\alpha) = \prod_{i \in S} \frac{p - \alpha_i}{\sqrt{p - p^2}}. \tag{39}$$

Observe that when $p = 1/2$, this is the standard basis discussed earlier. We first rederive a basic fact that the p-biased basis functions are orthonormal.

Proposition A.6 (Orthonormality of p-Biased Basis)**.** *For any subsets $S, T \subseteq [n]$,*

$$\mathop{\mathbb{E}}_{\alpha \sim \mathrm{Bern}(p)^n} [\chi_S^p(\alpha) \chi_T^p(\alpha)] = \begin{cases} 1, & \text{if } S = T, \\ 0, & \text{if } S \neq T. \end{cases} \tag{40}$$

Proof. For any coordinate $i \in [n]$, note the following identities for the first and second moments:

$$\mathop{\mathbb{E}}_{\alpha \sim \mathrm{Bern}(p)} \left[\frac{p - \alpha_i}{\sqrt{p - p^2}} \right] = 0, \qquad \mathop{\mathbb{E}}_{\alpha \sim \mathrm{Bern}(p)} \left[\left(\frac{p - \alpha_i}{\sqrt{p - p^2}} \right)^2 \right] = 1. \tag{41}$$

The inner product is then:

$$\mathop{\mathbb{E}}_{\alpha \sim \mathrm{Bern}(p)^n} [\chi_S^p(\alpha) \chi_T^p(\alpha)] = \mathop{\mathbb{E}}_{\alpha} \left[\prod_{i \in S} \frac{p - \alpha_i}{\sqrt{p - p^2}} \prod_{j \in T} \frac{p - \alpha_j}{\sqrt{p - p^2}} \right] \tag{42}$$

$$= \underbrace{\prod_{i \in S \cap T} \mathop{\mathbb{E}}_{\alpha} \left[\left(\frac{p - \alpha_i}{\sqrt{p - p^2}} \right)^2 \right]}_{= 1, \text{ for any } S \text{ and } T} \underbrace{\prod_{j \in S \triangle T} \mathop{\mathbb{E}}_{\alpha} \left[\frac{p - \alpha_j}{\sqrt{p - p^2}} \right]}_{= 0, \text{ if } S \triangle T \neq \emptyset} \tag{43}$$

where Δ is the symmetric set difference, and we have used the coordinate-wise independence of $\alpha_1, \ldots, \alpha_n$ to swap the expectation and products. □

We remark that setting $p = 1/2$ recovers the proof of orthonormality for the standard basis. On the p-biased basis, smoothing with a well-chosen λ induces a change-of-basis effect.

Lemma A.7 (Change-of-Basis)**.** *For any p-biased basis function χ_S^p and smoothing parameter $\lambda \in [p, 1]$,*

$$M_\lambda \chi_S^p(\alpha) = \left(\frac{\lambda - p}{1 - p} \right)^{|S|/2} \chi_S^{p/\lambda}(\alpha). \tag{44}$$

Proof. Expanding the definition of M_λ, we first derive:

$$M_\lambda \chi_S^p(\alpha) = \mathop{\mathbb{E}}_{z \sim \mathrm{Bern}(\lambda)^n} \left[\prod_{i \in S} \frac{p - \alpha_i z_i}{\sqrt{p - p^2}} \right] \tag{45}$$

$$= \prod_{i \in S} \mathop{\mathbb{E}}_{z} \left[\frac{p - \alpha_i z_i}{\sqrt{p - p^2}} \right] \qquad \text{(by independence of } z_1, \ldots, z_n\text{)}$$

$$= \prod_{i \in S} \frac{p - \lambda \alpha_i}{\sqrt{p - p^2}}, \tag{46}$$

We then rewrite the above in terms of a (p/λ)-biased basis function as follows:

$$\begin{aligned}
M_\lambda \chi_S^p(\alpha) &= \prod_{i\in S} \lambda \frac{(p/\lambda)-\alpha_i}{\sqrt{p-p^2}} && (47)\\
&= \prod_{i\in S} \lambda \frac{\sqrt{(p/\lambda)-(p/\lambda)^2}}{\sqrt{p-p^2}} \frac{(p/\lambda)-\alpha_i}{\sqrt{(p/\lambda)-(p/\lambda)^2}} && (\lambda \geq p)\\
&= \prod_{i\in S} \sqrt{\frac{\lambda-p}{1-p}} \frac{(p/\lambda)-\alpha_i}{\sqrt{(p/\lambda)-(p/\lambda)^2}} && (48)\\
&= \left(\frac{\lambda-p}{1-p}\right)^{|S|/2} \underbrace{\prod_{i\in S} \frac{(p/\lambda)-\alpha_i}{\sqrt{(p/\lambda)-(p/\lambda)^2}}}_{\chi_S^{p/\lambda}(\alpha)} && (49)
\end{aligned}$$

□

When measured with respect to this changed basis, we can show that M_λ provably contracts the variance.

Theorem A.8 (Variance Reduction)**.** *For any function $h : \{0,1\}^n \to \mathbb{R}$ and smoothing parameter $\lambda \in [p, 1]$,*

$$\operatorname*{Var}_{\alpha\sim\text{Bern}(p/\lambda)^n} [M_\lambda h(\alpha)] \leq \left(\frac{\lambda-p}{1-p}\right) \operatorname*{Var}_{\alpha\sim\text{Bern}(p)^n} [h(\alpha)]. \tag{50}$$

If the function is centered at $\mathbb{E}_{\alpha\sim\mathsf{Bern}(p)^n}[h(\alpha)] = 0$, then we also have:

$$\mathop{\mathbb{E}}_{\alpha\sim\text{Bern}(p/\lambda)^n} \left[M_\lambda h(\alpha)^2\right] \leq \mathop{\mathbb{E}}_{\alpha\sim\text{Bern}(p)} \left[h(\alpha)^2\right]. \tag{51}$$

Proof. We use the previous results to compute:

$$\begin{aligned}
\operatorname*{Var}_{\alpha\sim\text{Bern}(p/\lambda)^n} [M_\lambda h(\alpha)] &= \operatorname*{Var}_{\alpha\sim\text{Bern}(p/\lambda)^n} \left[M_\lambda \sum_{S\subseteq[n]} \widehat{h}(S)\chi_S^p(\alpha)\right] && \text{(by unique } p\text{-biased representation of } h)\\
&= \operatorname*{Var}_{\alpha\sim\text{Bern}(p/\lambda)^n} \left[\sum_{S\subseteq[n]} \left(\frac{\lambda-p}{1-p}\right)^{|S|/2} \widehat{h}(S)\chi_S^{p/\lambda}(\alpha)\right] && \text{(by linearity and Lemma A.7)}\\
&= \sum_{S\neq\emptyset} \left(\frac{\lambda-p}{1-p}\right)^{|S|} \widehat{h}(S)^2 && \text{(Parseval's theorem by orthonormality of } \chi_S^{p/\lambda})\\
&\leq \left(\frac{\lambda-p}{1-p}\right) \sum_{S\neq\emptyset} \widehat{h}(S)^2 && (0 \leq \tfrac{\lambda-p}{1-p} \leq 1 \text{ because } p \leq \lambda \leq 1)\\
&= \left(\frac{\lambda-p}{1-p}\right) \operatorname*{Var}_{\alpha\sim\text{Bern}(p)^n} [h(\alpha)] && \text{(Parseval's by orthonormality of } \chi_S^p)
\end{aligned}$$

which leads to the first desired inequality. For the second inequality, we have:

$$\mathop{\mathbb{E}}_{\alpha\sim\mathsf{Bern}(p)^n} [h(\alpha)^2] = \widehat{h}(\emptyset)^2 + \underbrace{\sum_{S\neq\emptyset} \widehat{h}(S)^2}_{\text{Var}[h(\alpha)]}, \tag{52}$$

$$\mathop{\mathbb{E}}_{\alpha\sim\mathsf{Bern}(p/\lambda)^n} [M_\lambda h(\alpha)^2] = \widehat{M_\lambda h}(\emptyset)^2 + \underbrace{\sum_{S\neq\emptyset} \widehat{M_\lambda h}(S)^2}_{\text{Var}[M_\lambda h(\alpha)]}, \tag{53}$$

where recall that $\widehat{h}(\emptyset) = \mathbb{E}_\alpha[h(\alpha)]$ is zero by assumption. □

In general, the smoothing operator M_λ acts like a downshift operator on the Fourier spectrum for both the standard and biased basis. However, the redistribution of spectral mass is algebraically cumbersome to manipulate. To address this, we next introduce in Appendix B a new set of basis functions, the *monotone basis* that allows for a more tractable characterization of how smoothing affects the structure and stability of Boolean functions in the context of classification.

B Analysis of Stability and Smoothing in the Monotone Basis

While the standard Fourier basis is a common starting point for studying Boolean functions, its interaction with the smoothing operator M_λ is algebraically complex. The main reason is that the Fourier basis treats $0 \to 1$ and $1 \to 0$ perturbations symmetrically. In contrast, we wish to analyze perturbations that add features (i.e., $\beta \geq \alpha$) and smoothing operations that remove features. This mismatch results in a complex redistribution of terms that is algebraically inconvenient to manipulate. We were thus motivated to develop a new set of analytical tooling, principally in a new set of basis functions that we call the *monotone basis*.

B.1 Monotone Basis for Boolean Functions

For any subset $T \subseteq [n]$, define its corresponding *monotone basis function* $\mathbf{1}_T : \{0,1\}^n \to \{0,1\}$ as:

$$\mathbf{1}_T(\alpha) = \begin{cases} 1, & \text{if } \alpha_i = 1 \text{ for all } i \in T \text{ (all features in } S \text{ present)}, \\ 0, & \text{otherwise (any feature in } T \text{ is absent)}, \end{cases} \tag{54}$$

where let $\mathbf{1}_\emptyset(\alpha) = 1$. First, we flexibly identify subsets of $[n]$ with binary vectors in $\{0,1\}^n$, which lets us write $T \subseteq \alpha$ if $i \in T$ implies $\alpha_i = 1$. This gives us some useful ways to equivalently express $\mathbf{1}_T(\alpha)$:

$$\mathbf{1}_T(\alpha) = \prod_{i \in T} \alpha_i = \begin{cases} 1, & T \subseteq \alpha, \\ 0, & \text{otherwise.} \end{cases} \tag{55}$$

The monotone basis lets us more compactly express properties that depend on the inclusion or exclusion of features. For instance, the earlier example of conjunction $h(\alpha) = \alpha_1 \wedge \alpha_2$ may be equivalently written as:

$$\begin{aligned} \alpha_1 \wedge \alpha_2 &= \mathbf{1}_{\{1,2\}}(\alpha) && \text{(monotone basis)} \\ &= \frac{1}{4}\chi_\emptyset(\alpha) - \frac{1}{4}\chi_{\{1\}}(\alpha) - \frac{1}{4}\chi_{\{2\}}(\alpha) + \frac{1}{4}\chi_{\{1,2\}}(\alpha) && \text{(standard basis)} \end{aligned}$$

Unlike the standard bases (both standard Fourier and p-biased Fourier), the monotone basis is not orthonormal. However, it satisfies interesting properties.

Theorem B.1. *Any Boolean function $h : \{0,1\}^n \to \mathbb{R}^n$ can be uniquely expressed in the monotone basis as:*

$$h(\alpha) = \sum_{T \subseteq [n]} \widetilde{h}(T) \mathbf{1}_T(\alpha), \tag{56}$$

where $\widetilde{h}(T) \in \mathbb{R}$ are the monotone basis coefficients of h. For any $S, T \subseteq [n]$, we have:

$$\mathop{\mathbb{E}}_{\alpha \sim \{0,1\}^n} [\mathbf{1}_S(\alpha) \mathbf{1}_T(\alpha)] = 2^{-|S \cup T|}, \tag{57}$$

and the monotone coefficients can be recursively computed by the formula:

$$\widetilde{h}(T) = h(T) - \sum_{S \subsetneq T} \widetilde{h}(S), \tag{58}$$

where $h(T)$ denotes the evaluation of h on the binary vectorized representation of T.

Proof. We first prove existence and uniqueness. By definition of $\mathbf{1}_T$, we have the simplification:

$$h(\alpha) = \sum_{T \subseteq [n]} \widetilde{h}(T)\mathbf{1}_T(\alpha) = \sum_{T \subseteq \alpha} \widetilde{h}(T). \tag{59}$$

This yields a system of 2^n linear equations (one for each $h(\alpha)$) in 2^n unknowns (one for each $\widetilde{h}(T)$). We may treat this as a matrix of size $2^n \times 2^n$ with rows indexed by $h(\alpha)$ and columns indexed by $\widetilde{h}(T)$, sorted by inclusion and degree (e.g., Equation (30)). This matrix is then lower-triangular with ones on the diagonal ($\mathbf{1}_T(T) = 1$ and $\mathbf{1}_T(\alpha) = 0$ for $|T| > \alpha$), meaning that the 2^n values of $h(\alpha)$ uniquely determine $\widetilde{h}(T)$.

Next, we prove the inner product formula by using the fact that $S \cup T \subseteq \alpha$ iff both $S \subseteq \alpha$ and $T \subseteq \alpha$:

$$\mathop{\mathbb{E}}_{\alpha \sim \{0,1\}^n} [\mathbf{1}_S(\alpha)\mathbf{1}_T(\alpha)] = \Pr_{\alpha \sim \{0,1\}^n} [S \cup T \subseteq \alpha] = 2^{-|S \cup T|}. \tag{60}$$

Finally, for the recursive formula, recall that:

$$h(T) = \widetilde{h}(T) + \sum_{S \subsetneq T} \widetilde{h}(S), \tag{61}$$

and re-ordering terms yields the desired result. □

B.2 Smoothing and Stability in the Monotone Basis

A key advantage of the monotone basis is that it yields a convenient analytical expression for how smoothing affects the spectrum.

Theorem B.2 (Smoothing in the Monotone Basis)**.** *Let M_λ be the smoothing operator as in Definition A.1. Then, for any Boolean function $h : \{0,1\}^n \to \mathbb{R}$ and subset $T \subseteq [n]$, we have the spectral contraction:*

$$\widetilde{M_\lambda h}(T) = \lambda^{|T|}\widetilde{h}(T),$$

where $\widetilde{M_\lambda h}(T)$ and $\widetilde{h}(T)$ are the monotone basis coefficients of $M_\lambda h$ and h at T, respectively.

Proof. By linearity of expectation, it suffices to study the action of M_λ on each monotone basis function:

$$\begin{aligned}
M_\lambda \mathbf{1}_T(\alpha) &= \mathop{\mathbb{E}}_{z \sim \mathsf{Bern}(\lambda)^n} [\mathbf{1}_T(\alpha \odot z)] && \text{(by definition of } M_\lambda) \\
&= \mathop{\mathbb{E}}_{z \sim \mathsf{Bern}(\lambda)^n} \left[\prod_{i \in T} (\alpha_i z_i)\right] && \text{(by definition of } \mathbf{1}_T(\alpha)) \\
&= \prod_{i \in T} \left(\alpha_i \mathop{\mathbb{E}}_{z_i \sim \mathsf{Bern}(\lambda)} [z_i]\right) && \text{(by independence of } z_1, \ldots, z_n) \\
&= \lambda^{|T|}\mathbf{1}_T(\alpha) && (\mathbb{E}\,[z_i] = \lambda)
\end{aligned}$$

□

The monotone basis also gives a computationally tractable way of bounding the stability rate. Crucially, the difference between two Boolean functions is easier to characterize. As a simplified setup, we consider classifiers of form $h : \{0,1\}^n \to \mathbb{R}$, where for $\beta \sim \Delta_r(\alpha)$ let:

$$h(\beta) \cong h(\alpha) \quad \text{if} \quad |h(\beta) - h(\alpha)| \leq \gamma. \tag{62}$$

Such h and its decision boundary γ may be derived from a general classifier $f : \mathbb{R}^n \to \mathbb{R}^m$ once x and α are known. This relation of the decision boundary then motivates the difference computation:

$$h(\beta) - h(\alpha) = \sum_{T \subseteq [n]} \widetilde{h}(T)(\mathbf{1}_T(\beta) - \mathbf{1}_T(\alpha) = \sum_{T \subseteq \beta \setminus \alpha, T \neq \emptyset} \widetilde{h}(T), \tag{63}$$

where recall that $\mathbf{1}_T(\beta) - \mathbf{1}_T(\alpha) = 1$ iff $T \neq \emptyset$ and $T \subseteq \beta \setminus \alpha$. This algebraic property plays a key role in tractably bounding the stability rate. More precisely, we upper-bound the *instability rate* $1 - \tau_r$, which is

$$1 - \tau_r = \Pr_{\beta \sim \Delta_r(\alpha)} [|h(\beta) - h(\alpha)| > \gamma]. \tag{64}$$

An upper bound of form $1 - \tau_r \leq Q$, where Q depends on the monotone coefficients of h, then implies a lower bound on the stability rate $1 - Q \leq \tau_r$. We show this next.

Lemma B.3 (Soft Stability Bound)**.** *For any Boolean function $h : \{0,1\}^n \to [0,1]$ and attribution $\alpha \in \{0,1\}^n$ that satisfy Equation* (62)*, the stability rate τ_r is bounded by:*

$$1 - \tau_r \leq \frac{1}{\gamma} \sum_{k=1}^{r} \sum_{\substack{T \subseteq [n] \setminus \alpha \\ |T| = k}} |\widetilde{h}(T)| \cdot \Pr_{\beta \sim \Delta_r} [|\beta \setminus \alpha| \geq k], \tag{65}$$

where

$$\Pr_{\beta \sim \Delta_r} [|\beta \setminus \alpha| \geq k] = \frac{1}{|\Delta_r|} \sum_{j=k}^{r} \binom{n - |\alpha| - k}{j - k}. \tag{66}$$

Proof. We can directly bound the stability rate as follows:

$$\begin{aligned}
1 - \tau_r &= \Pr_{\beta \sim \Delta_r} [|h(\beta) - h(\alpha)| > \gamma] && (67) \\
&\leq \frac{1}{\gamma} \mathop{\mathbb{E}}_{\beta \sim \Delta_r} [|h(\beta) - h(\alpha)|] && \text{(Markov's inequality)} \\
&\leq \frac{1}{\gamma} \mathop{\mathbb{E}}_{\beta \sim \Delta_r} \sum_{\substack{T \subseteq \beta \setminus \alpha \\ T \neq \emptyset}} |\widetilde{h}(T)| && \text{(by Equation (63), triangle inequality)} \\
&= \frac{1}{\gamma |\Delta_r|} \sum_{k=0}^{r} \sum_{|\beta \setminus \alpha| = k} \sum_{\substack{T \subseteq \beta \setminus \alpha \\ T \neq \emptyset}} |\widetilde{h}(T)| && \text{(enumerate } \beta \in \Delta_r(\alpha) \text{ by its size, } k) \\
&= \frac{1}{\gamma |\Delta_r|} \sum_{k=1}^{r} \sum_{\substack{S \subseteq [n] \setminus \alpha \\ |S| = k}} \sum_{\substack{T \subseteq S \\ T \neq \emptyset}} |\widetilde{h}(T)| && \text{(the } k = 0 \text{ term is zero, and let } S = \beta \setminus \alpha) \\
&= \frac{1}{\gamma |\Delta_r|} \sum_{k=1}^{r} \sum_{\substack{T \subseteq [n] \setminus \alpha \\ |T| = k}} |\widetilde{h}(T)| \cdot \underbrace{|\{S \subseteq [n] \setminus \alpha : S \supseteq T, |S| \leq r\}|}_{\text{Total times that } \widetilde{h}(T) \text{ appears}} && \text{(re-index by } T) \\
&= \frac{1}{\gamma} \sum_{k=1}^{r} \sum_{\substack{T \subseteq [n] \setminus \alpha \\ |T| = k}} |\widetilde{h}(T)| \cdot \Pr_{\beta \sim \Delta_r} [|\beta \setminus \alpha| \geq k] && (68)
\end{aligned}$$

□

An immediate consequence from Theorem B.2 is a stability rate bound on smoothed functions.

Theorem B.4 (Stability of Smoothed Functions)**.** *Consider any Boolean function $h : \{0,1\}^n \to [0,1]$ and attribution $\alpha \in \{0,1\}^n$ that satisfy Equation* (62)*. Then, for any smoothing parameter $\lambda \in [0,1]$,*

$$1 - \frac{Q}{\gamma} \leq \tau_r(h, \alpha) \implies 1 - \frac{\lambda Q}{\gamma} \leq \tau_r(M_\lambda h, \alpha), \tag{69}$$

where

$$Q = \sum_{k=1}^{r} \sum_{\substack{T \subseteq [n] \setminus \alpha \\ |T| = k}} |\widetilde{h}(T)| \cdot \Pr_{\beta \sim \Delta_r} [|\beta \setminus \alpha| \geq k]. \tag{70}$$

Proof. This follows by noting that:

$$1 - \tau_r(M_\lambda h, \alpha) \leq \frac{1}{\gamma} \sum_{k=1}^{r} \lambda^k \sum_{\substack{T \subseteq [n] \setminus \alpha \\ |T| = k}} |\widetilde{h}(T)| \cdot \Pr_{\beta \sim \Delta_r} [|\beta \setminus \alpha| \geq k]. \tag{71}$$

□

Moreover, we also present the following result on hard stability in the monotone basis.

Theorem B.5 (Hard Stability Bound)**.** *For any Boolean function $h : \{0,1\}^n \to [0,1]$ and attribution $\alpha \in \{0,1\}^n$ that satisfy Equation* (62)*, let*

$$r^\star = \operatorname*{arg\,max}_{r \geq 0} \max_{\beta : |\beta \setminus \alpha| \leq r} \left[\left| \sum_{T \subseteq \beta \setminus \alpha, T \neq \emptyset} \widetilde{h}(T) \right| \leq \frac{1}{2} \right]. \tag{72}$$

Then, h is hard stable at α with radius $r^\star$.

Proof. This follows from Equation (63) because it is equivalent to stating that:

$$r^\star = \operatorname*{arg\,max}_{r \geq 0} \max_{\beta : |\beta \setminus \alpha| \leq r} \underbrace{\left[|h(\beta) - h(\alpha)| \leq \frac{1}{2} \right]}_{h(\beta) \cong h(\alpha)}. \tag{73}$$

□

In summary, the monotone basis provides a more natural setting in which to study the smoothing operator M_λ. While M_λ yields an algebraically complex weight redistribution under the standard basis, its effect is more compactly described in the monotone basis as a point-wise contraction at each $T \subseteq [n]$. In particular, we are able to derive a lower-bound improvement on the stability of smoothed functions in Theorem B.4.

C Additional Experiments and Figures

Experiment Setup For vision models, we use Vision Transformer (ViT) [15], ResNet50, and ResNet18 [21]. For language models, we use RoBERTa [33]. For the vision dataset, we used a subset of ImageNet that contains two classes per sample, for a total of 2000 images. The images are of size 3 × 224 × 224, which we segmented into grids with patches of size 16 × 16, for a total of $n = (224/16)^2 = 196$ features. For the language dataset, we used six subsets of TweetEval (emoji, emotion, hate, irony, offensive, sentiment) for a total of 10653 items; we omitted the stance subset because their corresponding fine-tuned models were not readily available. For feature attribution methods, we used LIME [44], SHAP [34], Integrated Gradients [51], and MFABA [61] using the implementation from exlib [5] Each attribution method outputs a ranking of features by their importance score, and we selected the top-25% of features as the binarized explanation. For GPUs, we had access to a combination of NVIDIA GeForce RTX 3090 and NVIDIA RTX A6000.

[5] `https://github.com/BrachioLab/exlib`

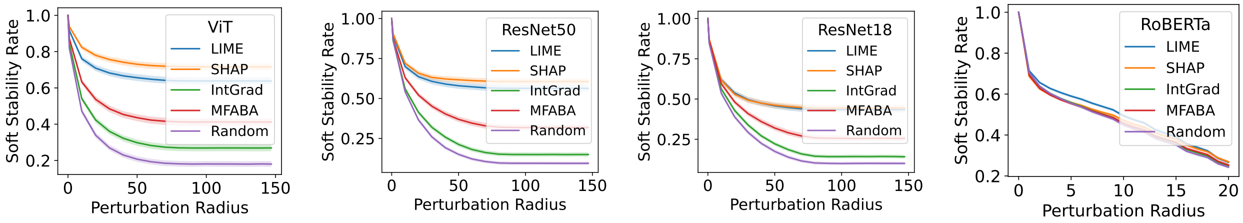


Figure 8: **Soft stability rate of different explanation methods**. For ViT, ResNet50, and ResNet18, we used 2000 samples from ImageNet. For RoBERTa, we used TweetEval. We used bootstrapping with 1000 iterations to compute a confidence interval of 95%, shaded.

Computing the Hard Certified Stability Radius An important part of hard stability is in computing the (hard) certified radius. Below, we describe how Xue et al. [56] compute this for a smoothed classifier.

Theorem C.1 (Hard Stability Radius [56])**.** *For any classifier $f : \mathbb{R}^n \to [0,1]$ and smoothing parameter $\lambda \in [0,1]$, let $\tilde{f} = M_\lambda f$ be the smoothed classifier. For any input $x \in \mathbb{R}^n$ and explanation $\alpha \in \{0,1\}^n$, the certifiable hard stability radius is given by:*

$$r_{\text{cert}} = \frac{\tilde{f}_1(x \odot \alpha) - \tilde{f}_2(x \odot \alpha)}{2\lambda}, \tag{74}$$

where $\tilde{f}_1(x \odot \alpha)$ and $\tilde{f}_2(x \odot \alpha)$ denote the top-1 and top-2 class probabilities of the smoothed output $\tilde{f}(x \odot \alpha)$.

Each output coordinate $\tilde{f}_1, \ldots, \tilde{f}_m$ is also λ-Lipschitz to the masking of features:

$$|\tilde{f}_i(x \odot \alpha) - \tilde{f}_i(x \odot \alpha')| \leq \lambda|\alpha - \alpha'|, \quad \text{for all } \alpha, \alpha' \in \{0,1\}^n \text{ and } i = 1, \ldots, m. \tag{75}$$

That is, the keep-probability of each feature is also the Lipschitz constant (per earlier discussion: $\kappa = \lambda$). Note that deterministically evaluating $M_\lambda f_x$ would require 2^n samples in total, as there are 2^n possibilities for $\mathsf{Bern}(\lambda)^n$. Interestingly, distributions other than $\mathsf{Bern}(\lambda)^n$ also suffice to attain the desired Lipschitz, and thus hard certified radius, guarantees. In fact, Xue et al. [56] constructs such a distribution for which a smoothed classifier can be deterministically evaluated in $\ll 2^n$ samples. However, our Boolean analytic results do not readily extend to classifiers constructed from non-$\mathsf{Bern}(\lambda)^n$ distributions.

C.1 Soft Stability of Different Explanation Methods

We give an expanded view of how the soft stability rate varies across different explanation methods in Figure 8 for all models. As before, LIME and SHAP tend to have the best soft stability rates across all radii.

C.2 Hard Stability of Different Explanation Methods

We show the certified hard stability radii across different models and explanation methods in Figure 9, and we show the same information ordered by smoothing parameter in Figure 10. We see that all explanation methods attain similar hard stability certificates for their respective domains (i.e., vision and language).

Because smoothed classifier accuracy degrades as λ decreases (i.e., smoothing intensifies), one must necessarily degrade the classifier in order to get stronger guarantees. To facilitate direct comparison with soft stability, we may define the hard stability rate over a collection of input-attribution pairs (x, α) as:

$$\text{Hard stability rate at radius } r = \frac{|\{(x,\alpha) : \text{CertifiedRadius}(M_\lambda f_x, \alpha) \geq r\}|}{\text{Total number of } x\text{'s}}. \tag{76}$$

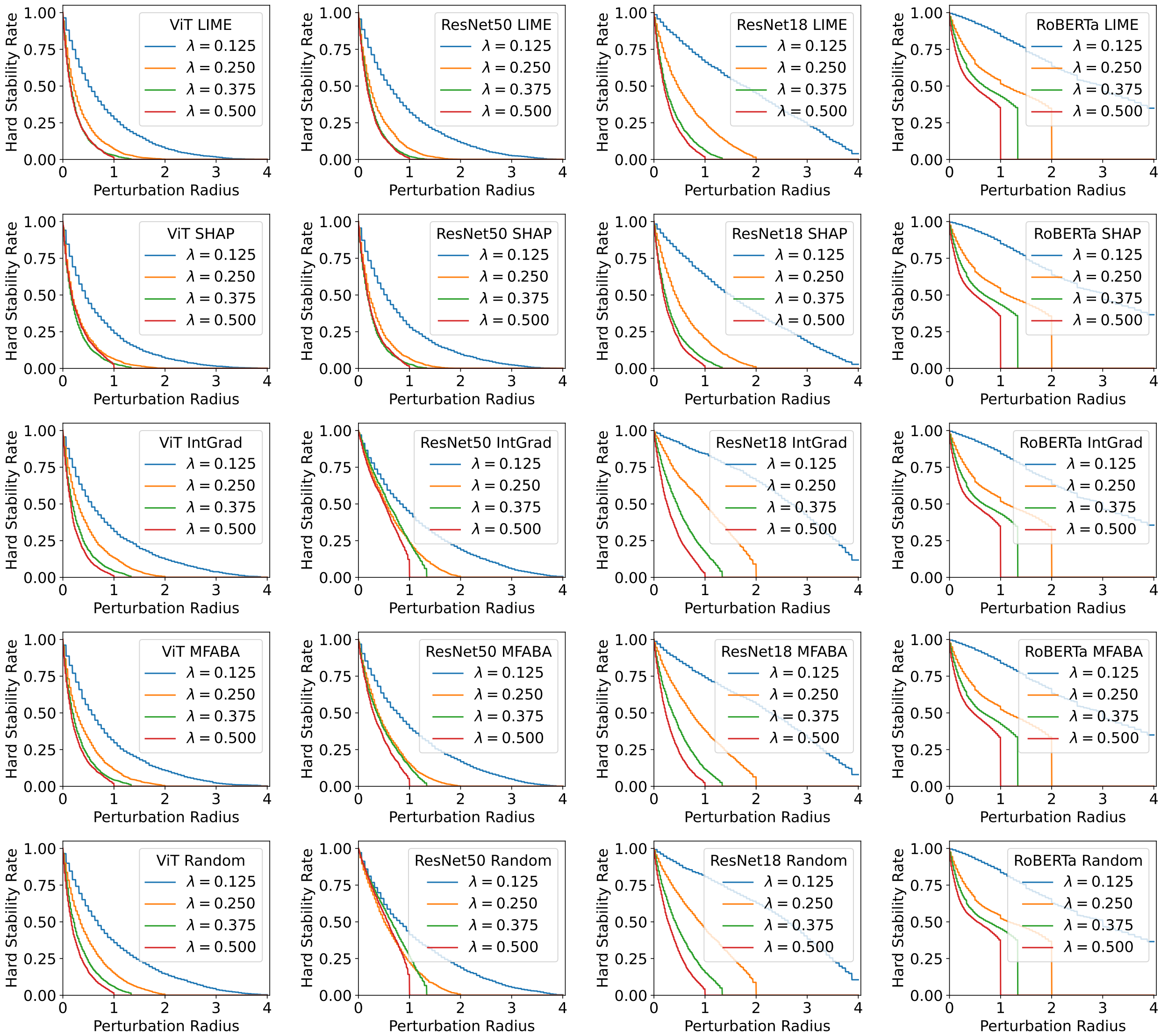


Figure 9: **Hard stability certificates, by explanation methods.** At any smoothing parameter λ, the maximum certifiable hard stability radius is only $1/2\lambda$ by Theorem C.1.

C.3 Random Masking vs. Random Flipping

We next study how the Fourier spectrum is affected by random masking and random flipping (i.e., the noise operator), which are respectively defined for Boolean functions as follows:

$$M_\lambda h(\alpha) = \mathop{\mathbb{E}}_{z \sim \mathsf{Bern}(\lambda)^n} [h(\alpha \odot z)] \qquad \text{(random masking)}$$

$$T_\lambda h(\alpha) = \mathop{\mathbb{E}}_{z \sim \mathsf{Bern}(q)^n} [h((\alpha + z) \bmod 2)], \quad q = \frac{1-\lambda}{2} \qquad \text{(random flipping)}$$

In both cases, $\lambda \approx 1$ corresponds to the lightly noised setting, whereas $\lambda \approx 0$ is the heavily noised setting.

To study this, we randomly generated a spectrum via $\widehat{h}(S) \sim N(0,1)$ for each $S \subseteq [n]$. We then average the mass of the randomly masked and randomly flipped spectrum at each degree, which are respectively:

$$\sum_{S:|S|=k} |\widehat{M_\lambda h}(S)|, \qquad \sum_{S:|S|=k} |\widehat{T_\lambda h}(S)|, \qquad \text{for each degree } k = 0, 1, \ldots, n. \tag{77}$$

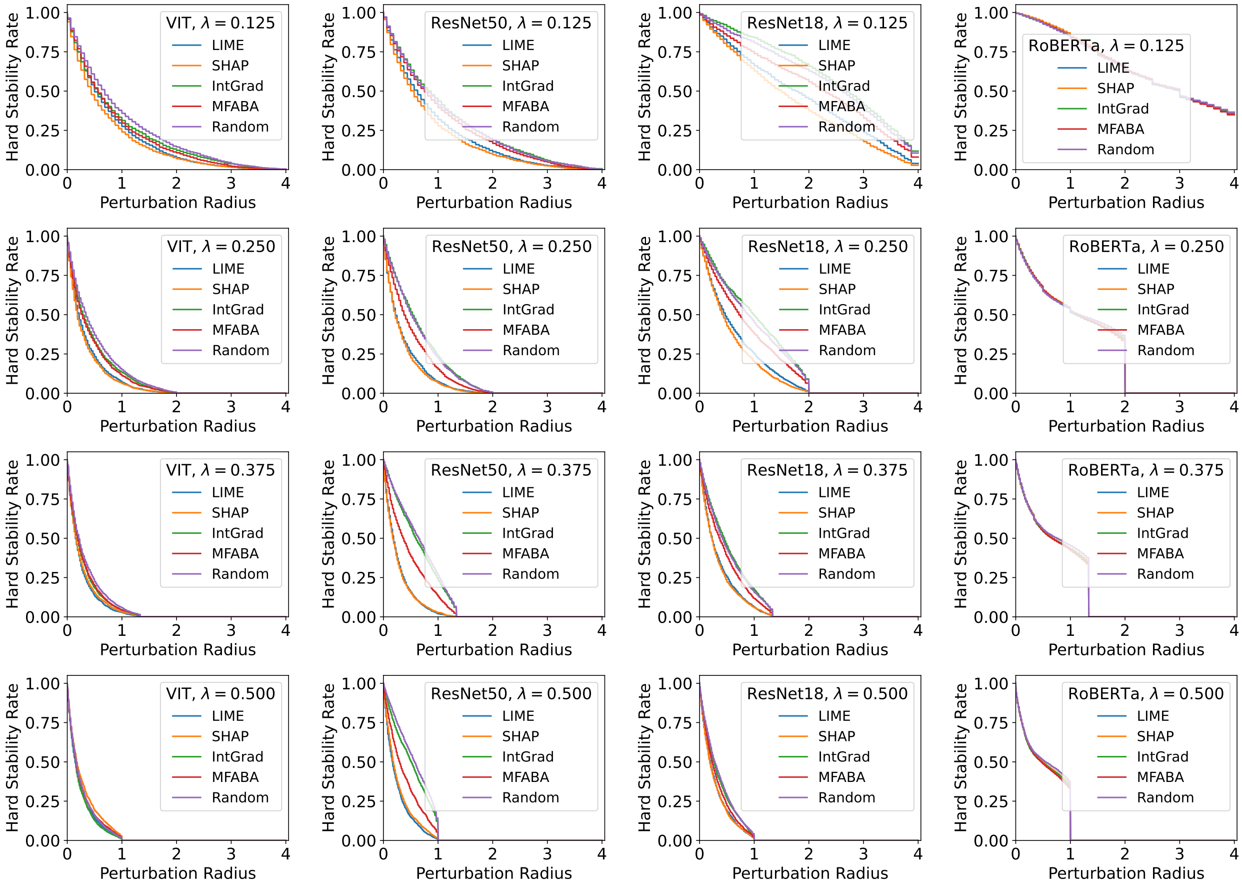


Figure 10: **Hard stability certificates, by smoothing parameter.** At any smoothing parameter λ, the maximum certifiable hard stability radius is only $1/2\lambda$ by Theorem C.1.

We plot the result in Figure 11 for $n = 10$ features. Note that random masking causes the spectrum to down-shift, whereas random flipping induces a point-wise contraction at each S.

D Additional Discussion

Our primary goal is to investigate reliable explanations for machine learning models, with stability serving as a key measure of reliability. While hard stability offers deterministic guarantees, it is highly conservative and limited to small certified radii, making it less practical for distinguishing between feature attribution methods. In contrast, soft stability leverages probabilistic certification to provide significantly larger guarantees while maintaining strong reliability. Our results also indicate that mild smoothing enhances soft stability without substantial accuracy degradation, suggesting broader applicability beyond robustness certification. These findings suggest the possibility of studying stability-aware training and adaptive smoothing techniques to improve the reliability and interpretability of feature-based explanations.

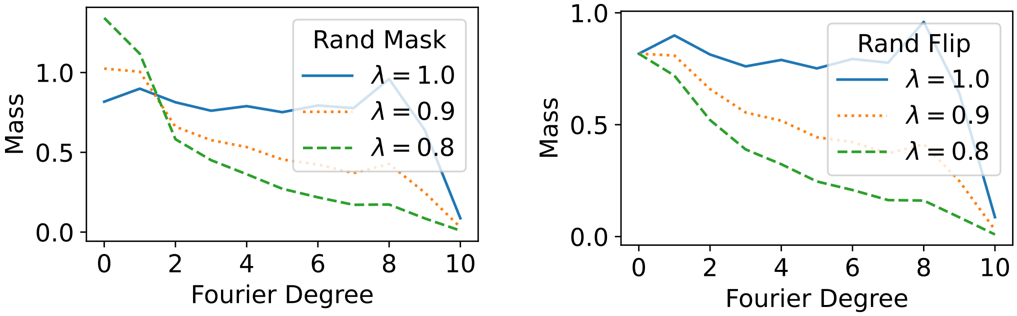


Figure 11: **Spectral effects of random masking (left) and random flipping (right).** With respect to the standard Fourier basis, random masking causes a down-shift in spectral mass, whereas the more commonly studied random flipping causes a point-wise contraction. This shows that the two smoothing operators are qualitatively different.