Probabilistic Stability Guarantees for Feature Attributions

Anonymous Authors[1]

Abstract

Stability guarantees are important for feature attributions, but existing certification methods rely on smoothed classifiers and yield overly conservative bounds. To address this, we introduce the concept of soft stability and propose a sampling-based certification algorithm that is both model-agnostic and sample-efficient. Interestingly, we demonstrate that mild smoothing can improve the soft stability certificate without incurring the severe accuracy degradation that heavily smoothed classifiers typically exhibit. 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 validate our approach through experiments on vision and language tasks with various feature attribution methods.

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 (Klauschen et al., 2024). Similarly, in the legal domain, attorneys must ensure that decisions reached with the assistance of models meet stringent judicial standards (Richmond et al., 2024).

There has been great interest in using explanation methods to understand opaque model behaviors. One popular class of explanation methods are *feature attributions* (Lundberg and Lee, 2017; Ribeiro et al., 2016), which aim to identify the most important input features that contribute to a model's prediction. We show such an example in Figure 1 using the top 44% features selected by LIME (Ribeiro et al., 2016).

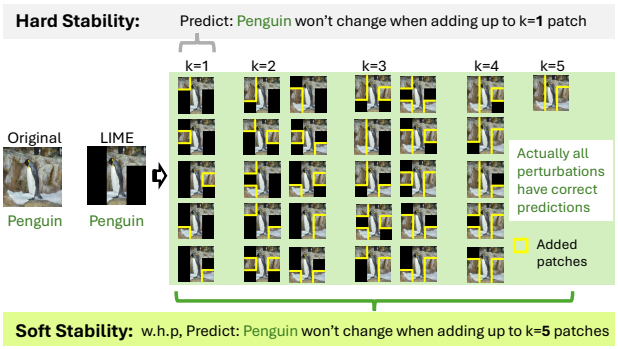

Figure 1. A visual example of certified radii by hard stability vs. soft stability. For an image of penguin masked to show only top 44% explanation by LIME, hard stability certifies that adding 1 patch won't change the prediction, whereas soft stability can certify adding up to 5 patches with a probabilistic guarantee.

A common way to evaluate an attribution method is to test whether the selected features contain enough information to recover the original prediction (Nauta et al., 2023; Wagner et al., 2019). The selection in this example can do so, but including a single additional patch can drastically alter the predicted label. Despite the complexity of modern classifiers, this behavior is often undesirable because it suggests that providing more information can paradoxically decrease the model's confidence (Yeh et al., 2019).

Such phenomenon has sparked interest in quantifying how model predictions vary with attributions, such as the effect of adding or removing features (Samek et al., 2016; Wu et al., 2020) and the impact of the selection's shape (Hase et al., 2021; Rong et al., 2022). However, most existing works focus on empirical measures (Agarwal et al., 2022), with limited formal, mathematical guarantees on the robustness of attribution-induced predictions.

To address this gap, Xue et al. (2024) consider *stability* as a formal certification framework for feature selection. In particular, a *stable* explanation is one in which adding a small number of features does not alter the model's prediction, thereby eliminating the undesirable behavior illustrated in Figure 2. This property is quantified by its *certified radius*, which measures the maximum number of additional features that can be included while preserving the prediction.

[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

However, certifying stability is non-trivial. If the classifier lacks favorable properties, one must exhaustively check predictions for all possible feature additions, a computationally intractable task. To overcome this, Xue et al. (2024) apply smoothing techniques from the adversarial robustness (Cohen et al., 2019; Levine and Feizi, 2021) to transform arbitrary models into *smoothed classifiers* with convenient properties for efficiently computing certified radii. However, these radii are often small and apply only to the smoothed classifier rather than the original model. Moreover, smoothing inherently degrades a classifier's accuracy. While these guarantees are meaningful, they remain conservative and impose a harsh accuracy trade-off on the smoothed classifier.

In this work, we present a new variant of stability that we call *soft stability*. We define this in contrast to that of Xue et al. (2024), which we refer to as *hard stability* from this point forward. While hard stability certifies whether *all* small perturbations to an attribution yield the same prediction, soft stability instead quantifies *how often* the prediction remains consistent. This is a probabilistic relaxation of hard stability that avoids the need to smooth the classifier. In general, probabilistic guarantees are flexible to apply and efficient to compute compared to their hard variants. Consequently, they have gained traction in machine learning applications such as medical imaging (Fayyad et al., 2024), drug discovery (Arvidsson McShane et al., 2024), and autonomous driving (Lindemann et al., 2023). Conveniently, probabilistic guarantees are also often formulated in terms of *confidence*, which is widely explored in machine learning and explainability literature (Angelopoulos et al., 2023; Atanasova, 2024; Carvalho et al., 2019).

In this work, we advance the understanding of robust feature-based explanations by extending probabilistic guarantees to stability. Our analyses and experiments provide new insights into attribution robustness, especially on the role of smoothed classifiers. Our key contributions are as follows.

Soft Stability is Model-Agnostic and Sample-Efficient We introduce soft stability as a measure for certifying the robustness of feature attributions. Unlike hard stability, which relies on a destructive smoothing process and yields conservative guarantees, soft stability applies non-destructively to any classifier and is sample-efficient to certify. This contributes to the sparse literature on formal guarantees for feature attributions. We further examine the computational challenges of hard stability and introduce an algorithm for certifying soft stability in Section 3.

Mild Smoothing Improves Soft Stability Interestingly, a milder version of smoothing from Xue et al. (2024) enhances a classifier's soft stability guarantees without significantly compromising accuracy. Using techniques from Boolean function analysis (O'Donnell, 2014), we provide a novel characterization of smoothing and develop new analytic tools to establish theoretical results. This expands the robustness toolkit beyond standard Lipschitz-based approaches and provides new insights for analyzing feature attributions, which we explore in more detail in Section 4.

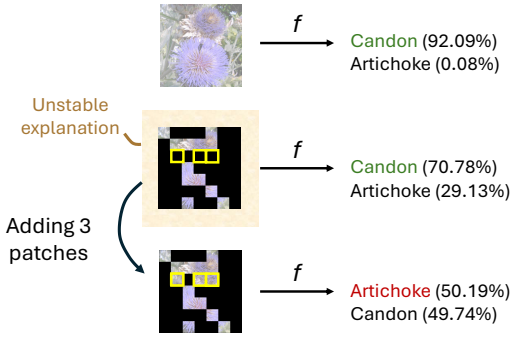

Figure 2. An unstable selection of features from SHAP. Although the image masked by the original explanation makes the same prediction as the original image (the second row vs. the first row), adding one patch to the explanation changes the highest predicted class from "candon" to "artichoke".

Empirical Validation We conduct experiments on vision and language tasks to validate our theoretical developments. Specifically, we compare the guarantees of soft and hard stability and analyze the effect of smoothing on classifier performance. These experiments provide empirical support for our claims and are detailed 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 (Simonyan et al., 2013; Sundararajan et al., 2017), each α_i might be dependent on $\nabla_{x_i} f(x)$, whereas in Shapley value-based methods (Lundberg and Lee, 2017; Sundararajan and Najmi, 2020), the α_i might to measure the Shapley value at x_i. Although attribution scores are typically real-valued, it is common to simplify them to binary values ($\alpha \in \{0, 1\}^n$) by selecting only the top-k most relevant features (Ribeiro et al., 2016). This aligns with the human preference for

concise and interpretable explanations (Miller, 2019).

2.2. Hard Stability and Soft Stability

Many evaluation metrics exist for binary-valued feature attributions (Agarwal et al., 2022). 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 $x \in \mathbb{R}^n$. To formalize this, 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:

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

This form of evaluating feature sets is related to notions of *fidelity*, *consistency*, and *preservation* in the explainability literature (Nauta et al., 2023), but the specific terminology and definition vary by author and source. Furthermore, attribution-masked evaluation is more commonly seen in vision tasks (Jain et al., 2022), though it is also present in language modeling (Lyu et al., 2023; Ye et al., 2024).

It is often desirable that two "similar" attributions induce the same prediction (Yeh et al., 2019). Although various measures of similarity exist, we are interested in the notion of additive perturbations. Specifically, we conceptualize a 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. We formalize such perturbations as follows.

Definition 2.1 (Additive Perturbations)**.** For an attribution α and radius $r > 0$, define its r-additive perturbation set as:

$$\Delta_r(\alpha) = \{\alpha' \in \{0,1\}^n : \alpha' \geq \alpha,\ \|\alpha' - \alpha\|_0 \leq r\},$$

where $\alpha' \geq \alpha$ iff each $\alpha'_i \geq \alpha_i$ and $\|\cdot\|_0$ denotes the ℓ^0 norm, which measures the number of non-zero coordinates.

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.

Definition 2.2 (Hard Stability (Xue et al., 2024))**.** For a classifier f and input x, the explanation α is *hard-stable* [1] with radius r if: $f(x \odot \alpha') \cong f(x \odot \alpha)$ for all $\alpha' \in \Delta_r$.

[1] Xue et al. (2024) equivalently call this property "incrementally stable" and more broadly define "stable" as a stricter property.

However, hard stability is non-trivial to certify, and existing algorithms suffer from costly trade-offs 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 (Fayyad et al., 2024), drug discovery (Arvidsson McShane et al., 2024), and autonomous driving (Lindemann et al., 2023), which are often formulated in terms of confidence (Atanasova, 2024; Carvalho et al., 2019). 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:

$$\tau_r(f, x, \alpha) = \Pr_{\alpha' \sim \Delta_r}[f(x \odot \alpha') \cong f(x \odot \alpha)],$$

where $\alpha' \sim \Delta_r$ is uniformly sampled.

A higher stability rate τ_r indicates a greater likelihood that a perturbation of at most r features preserves the prediction. Notably, soft stability generalizes hard stability, as the extreme case of $\tau_r = 1$ recovers the hard stability condition.

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 perturbation of α. 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 (Cohen et al., 2019), the Lipschitz constant is often intractable to compute (Virmaux and Scaman, 2018) and challenging to approximate (Fazlyab et al., 2019; Xue et al., 2022). To address this, Xue et al. (2024) derive *smoothed classifiers*, which have known Lipschitz constant by construction. Starting with any classifier f, one defines the *smoothed classifier* $\tilde{f}$ as the expectation over randomly perturbed inputs:

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

where $x^{(1)}, \ldots, x^{(N)} \sim \mathcal{D}(x)$ 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.

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 larger certified radii.

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. Additionally, since smoothing relies on evaluation with perturbed inputs, it inevitably leads to accuracy degradation compared to f. This relation between smoothness, certified radii, and accuracy follows a well-known trade-off:

$$\text{Smoothness}(\tilde{f}) \approx \text{CertRadius}(\tilde{f}) \approx \big(1 - \text{Accuracy}(\tilde{f})\big),$$

where $\approx$ indicates a general trend rather than an exact numerical relation. In other words, increased smoothness leads to larger certified radii (stronger hard stability guarantees) but at the cost of accuracy. This trade-off arises because excessive smoothing reduces a model's sensitivity, making it harder to distinguish between classes (Anil et al., 2019; Huster et al., 2019).

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 general, the certified radius of $\tilde{f}$ scales as $\mathcal{O}(1/\kappa)$ for any input x and attribution α.

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. The 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:*

$$\hat{\tau}_r = \frac{1}{N} \sum_{i=1}^{N} \mathbf{1}\big[f(x \odot \alpha^{(i)}) \cong f(x \odot \alpha)\big],$$

where $\alpha^{(1)}, \ldots, \alpha^{(N)} \sim \Delta_r(\alpha)$ are i.i.d. samples. Then, with probability $\geq 1 - \delta$, it holds that $|\tau_r - \hat{\tau}_r| \leq \varepsilon$.

Proof. Apply Hoeffding's inequality to estimate the mean of independently distributed Bernoulli $X^{(1)}, \ldots, X^{(N)}$, where let each $X^{(i)} = \mathbf{1}[f(x \odot \alpha^{(i)}) \cong f(x \odot \alpha)]$. □

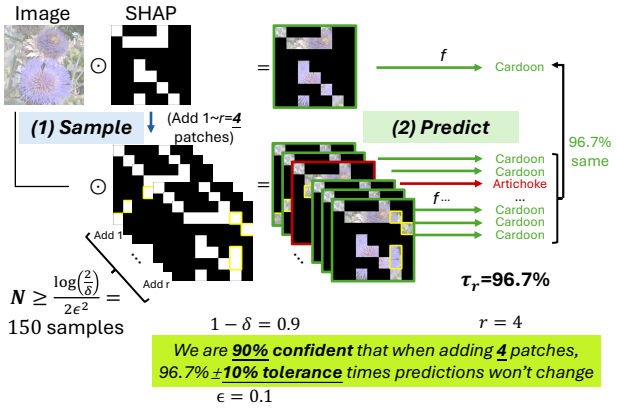

Figure 3. (Algorithm for Estimating Stability Rate.) Given an explanation α: (1) **Sample** masks from $\Delta_r(\alpha)$ by adding up to r patches. (2) **Predict** on the masked images and compute the stability rate τ_r, the fraction of predictions matching $f(x \odot \alpha)$. To ensure $1 - \delta$ confidence that the computed τ_r is within $\pm\epsilon$ tolerance, we sample at least $N \geq \frac{\log(2/\delta)}{2\epsilon^2}$ masks.

We illustrate this algorithm in Figure 3. Notably, the required sample size N depends only on ε and δ, since τ_r is a one-dimensional statistic. Because N is independent of f, the estimation algorithm scales linearly in the cost of evaluating f. Moreover, certifying soft stability does not require deriving a smoothed classifier through a destructive smoothing 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.

4. Mild Smoothing Improves Soft Stability

Smoothing is commonly used to certify robustness guarantees, such as hard stability (Xue et al., 2024), but often at a high cost to the smoothed classifier's accuracy. Interestingly, however, we find that a milder variant of the smoothing proposed in (Xue et al., 2024) 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, we observe that mildly smoothing the model empirically improves stability rates. To formalize this, we now introduce the multiplicative smoothing operator originally used to certify hard stability in Xue et al. (2024).

Definition 4.1 (Multiplicative Smoothing)**.** For any classifier f and smoothing parameter $\lambda \in [0, 1]$, define the multiplicative smoothing operator M_λ as:

$$M_\lambda f(x) = \mathop{\mathbb{E}}_{z \sim \text{Bern}(\lambda)^n} f(x \odot z),$$

where $z_1, \ldots, z_n \sim \text{Bern}(\lambda)$ are i.i.d. samples.

This method is termed multiplicative smoothing because the noise multiplies the input, unlike standard additive noise smoothing (Cohen et al., 2019). The parameter λ controls the probability of keeping each feature, wherein each feature is i.i.d. dropped (zeroed out) with probability $1 - \lambda$. Note that at $\lambda = 1$, we recover the original classifier $M_1 f(x) = f(x)$, while at $\lambda = 0$, the classifier is forced to predict on a fully masked input: $M_0 f(x) = f(\mathbf{0}_n)$.

4.1. Summary of Main Theoretical Findings

To effectively apply smoothing, we must avoid the drawbacks observed in hard-stability certification. Our first observation is the following:

> (**Finding 1**) Heavy smoothing ($\lambda \leq 1/2$) is required to obtain non-trivial hard stability guarantees.

Xue et al. (2024) only give hard stability guarantees when $\lambda \leq 1/2$. Critically, such a small λ means that at least half of the features are dropped from x on average, and so f must operate on a heavily masked image. Under such intense masking, it is expected that the smoothed classifier suffers a drop in accuracy, which was observed in Xue et al. (2024) and we also show in Section 5. However, such values of λ are far smaller than what is needed to improve the stability rate, as we later show in our experiments.

To understand the effect of multiplicative smoothing on soft stability, we draw on techniques from Boolean function analysis (O'Donnell, 2014), which studies functions of Boolean-valued inputs. This is related to attribution-masked evaluation as follows: for a classifier f and input x, let $f_x(\alpha) = f(x \odot \alpha)$ be the α-masked evaluation, such that $f_x : \{0,1\}^n \to \mathbb{R}^m$ is then a Boolean function. Section 4.2 gives an overview of Boolean function analysis and uses standard techniques to characterize the effect of smoothing on the classifier's spectrum, wherein our main result is that:

> (**Finding 2**) Mild smoothing ($\lambda \approx 1$) already suffices to rapidly decay the classifier's spectrum, making it less sensitive to perturbations in the selected features.

Although this gives a novel and insightful perspective on how smoothing affects the classifier's spectrum, it does not yield direct results on how smoothing affects the stability rate. To analyze this, we introduce novel analytic techniques in Section 4.3 to derive an explicit bound in terms of the smoothing parameter λ.

> (**Finding 3**) Smoothing improves the stability rate *lower bound* by a factor of λ: if $1 - \mathcal{Q} \leq \tau_r(f_x)$, then $1 - \lambda\mathcal{Q} \leq \tau_r(M_\lambda f_x)$, where $\mathcal{Q}$ is a quantity that depends on the spectrum of f_x and r.

In other words, the lower bound on the smoothed classifier's stability rate shrinks by a factor of λ compared to the original classifier's. We emphasize that the derivation of this bound required the investigation of novel Boolean function analytic tooling, which would be of interest for mathematically studying feature attributions. We give some background on Boolean function analysis and an overview of our results in the following, and we refer to Appendix A and Appendix B for more extensive details.

4.2. General Results via Boolean Function Analysis

Boolean functions are often analyzed as linear combinations of basis functions, with a standard choice being the p-biased Fourier basis, defined as follows.

Definition 4.2 (p-Biased Basis)**.** For any subset $S \subseteq [n]$, define its corresponding p-biased Fourier basis function as:

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

When $p = 1/2$, this is the standard basis. For example, the function $h(\alpha_1, \alpha_2) = \alpha_1 \wedge \alpha_2$ may be expressed as:

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

where we omit p for brevity, and we let the empty product equal 1 by convention so that $\chi_\emptyset(\alpha) = 1$ for any α. Importantly, these χ_S^p form an orthonormal basis: for any $S, T \subseteq [n]$, we have $\mathbb{E}_{\alpha \sim \mathsf{Bern}(p)^n}[\chi_S^p(\alpha)\chi_T^p(\alpha)] = 1$ when $S = T$, and zero otherwise. This lets us uniquely express $f_x : \{0,1\}^n \to \mathbb{R}^m$ as a linear combination of χ_S^p via

$$f_x(\alpha) = \sum_{S \subseteq [n]} \widehat{f}_x(S)\chi_S^p(\alpha), \quad \widehat{f}_x(S) = \mathop{\mathbb{E}}_{\alpha}[f_x(\alpha)\chi_S^p(\alpha)],$$

where $\widehat{f}_x(S) \in \mathbb{R}^m$ is the p-biased Fourier coefficient of f_x at the index set S, whose *degree* is its size $|S|$. Our first result characterizes how M_λ acts on each χ_S^p.

Lemma 4.3 (Spectrum Decay)**.** *For any p-biased basis function χ_S^p and smoothing parameter $\lambda \in [p, 1]$,*

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

For $\lambda \geq p$, smoothing can be understood as a change-of-basis from χ_S^p to $\chi_S^{p/\lambda}$ while scaling the coefficient by a factor that decays exponentially with $|S|$. In particular,

$$M_\lambda f_x(\alpha) = \sum_{S \subseteq [n]} \left(\frac{\lambda - p}{1 - p}\right)^{|S|/n} \widehat{f}_x(S)\chi_S^{p/\lambda}(\alpha).$$

This contraction means that $M_\lambda f_x$ has lower variance than f_x and is, therefore, more robust than f_x with respect to the appropriate input distribution. To make this concrete:

Lemma 4.4 (Variance Reduction)**.** *For any function* $h : \{0,1\}^n \to \mathbb{R}$ *and smoothing parameter* $\lambda \in [p, 1]$*,*

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

If the function is centered at $h(\alpha) = 0$*, then we also have:*

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

The above result is established for scalar-valued functions, which is relevant when analyzing individual coordinates of f_x or considering f_x as a binary classifier. The above result provides a novel characterization of how multiplicative smoothing modifies the classifier's spectrum in terms of a variance reduction. However, there remains a gap between spectral decay and its impact on soft stability: while smoothing reduces variance, it is not immediately clear how this impacts the stability rate $\tau_r(f_x)$. To address this, we move beyond the standard Fourier basis and introduce novel analytic tooling next in Section 4.3, which allows us to derive explicit lower-bounds on the stability rate.

4.3. Lower Bounding the Stability Rate

Analyzing the stability of attributions leads to the study of Boolean functions under one-way perturbations. While the standard Fourier basis is a powerful theoretical tool, it has two key limitations in our setting. First, it treats the $0 \to 1$ and $1 \to 0$ transitions symmetrically, whereas stability concerns only additive perturbations of $\alpha' \ge \alpha$. Second, standard spectral analysis focuses on global function properties rather than local behavior about a particular α. This asymmetry of tooling and setting, combined with our focus on mild smoothing, motivates the development of new analytical tools beyond standard Fourier analysis. In particular, we introduce a monotone function basis as follows.

Definition 4.5 (Monotone Basis)**.** For any subset $S \subseteq [n]$, define its corresponding monotone basis function as:

$$\mathbf{1}_S(\alpha) = \begin{cases} 1, & \alpha_i = 1 \text{ for all } i \in S, \\ 0, & \text{otherwise.} \end{cases}$$

The monotone basis provides a direct encoding of set inclusion. For example, the function $h(\alpha_1, \alpha_2) = \alpha_1 \wedge \alpha_2$ is concisely represented as $h(\alpha) = \mathbf{1}_{\{1,2\}}(\alpha)$. More generally, any f_x also admits a unique expansion:

$$f_x(\alpha) = \sum_{S\subseteq[n]} \widetilde{f}_x(S)\mathbf{1}_S(\alpha),$$

where let $\widetilde{f}_x(S)$ denote the monotone coefficient of S. Crucially, the monotone basis exhibits the properties that let us establish direct lower bounds on the stability rate of the smoothed classifier.

Lemma 4.6 (Smoothing Helps Stability)**.** *The stability rate of a binary classifier* $h : \{0,1\}^n \to [0,1]$ *is bounded by*

$$1 - \tau_r(h) \le \mathcal{Q}\Big(\{\widetilde{h}(S) : |S| \le r\}\Big),$$

where $\mathcal{Q}$ *depends on the monotone weights of degree* $\le r$*. For any* $\lambda \in [0,1]$*, the stability rate of* $M_\lambda h$ *is bounded by*

$$1 - \tau_r(M_\lambda h) \le \lambda\mathcal{Q}\Big(\{\widetilde{h}(S) : |S| \le r\}\Big).$$

This result quantifies the improvement in soft stability due to smoothing, wherein the worst-case bound shrinks by a factor of λ. We present extensive details in Appendix B.

5. Experiments

We evaluate the attainable soft stability rates on different classification models, and explore how multiplicative smoothing can improve them.

Setup For vision models, we use Vision Transformer (ViT) (Dosovitskiy, 2020) and ResNet (He et al., 2016). For language models, we use RoBERTa (Liu, 2019). For the vision dataset, we use a 1000-sized subset [2] of ImageNet (Deng et al., 2009) that contains one sample per each of its 1000 classes. For the language dataset, we use the emotion subset of TweetEval (Mohammad et al., 2018), which consists of four classes: anger, joy, optimism, and sadness. For feature attribution methods, we used LIME (Ribeiro et al., 2016), SHAP (Lundberg and Lee, 2017), Integrated Gradients (Sundararajan et al., 2017), and MFABA (Zhu et al., 2024). We convert real-valued attribution to binary-valued ones by selecting the top-k features.

Result 1: Soft Stability Certifies More Than Hard Stability We first investigate the quality of stability guarantees that different feature attribution methods can give us with respect to off-the-shelf, non-fine-tuned classifiers. We plot in Figure 4 the average soft and hard stability rates obtained by taking the top-25% of features ranked by LIME, SHAP, Integrated Gradients, MFABA, and random explanation methods. We use $\varepsilon = \delta = 0.1$ in our experiments, such that $N = 150$ according to our sampling method in Theorem 3.1. The plotted soft stability rates are estimates of the true soft stability rates within ε distance with probability 1 - δ, that is, we can guarantee that the estimated soft stability rates are within the 0.1 interval of the true soft stability rates with 90% probability.

We observe that the attainable radii are much larger by soft stability than hard stability, for both Vision Transformer and RoBERTa. In particular, for Vision Transformer, soft stability attains radii up to two orders of magnitude larger than

[2] github.com/EliSchwartz/imagenet-sample-images

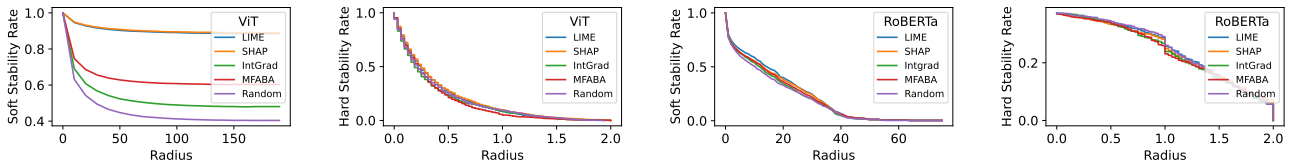

Figure 4. Soft Stability Certifies More Than Hard Stability. Given a classifier (VIT or Roberta) and an explanation method (e.g., SHAP 25%), we show soft stability rates vs. hard stability rates that are attainable. The soft stability rates that are shown are estimates of the true soft stability rates within ε distance with probability $1-\delta$, where $\varepsilon = \delta = 0.1$. (Far Left) The soft stability rates for Vision Transformer on different explanation methods as a function of the radius. (Center Left) The hard stability rates for Vision Transformer on different explanation methods as a function of the radius. (Center Right) The soft stability rates for RoBERTa on different explanation methods as a function of the radius. (Far Right) The hard stability rates for RoBERTa on different explanation methods as a function of the radius.

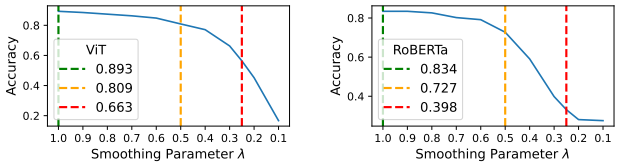

Figure 5. Accuracy decreases as smoothing intensifies. We report accuracy values at key thresholds: ($\lambda = 1.0$) the original, unmodified classifier; ($\lambda = 0.5$) the threshold above which hard stability certificates are not attainable; and ($\lambda = 0.25$) at which hard stability can only certify additive perturbations of up to 2 features.

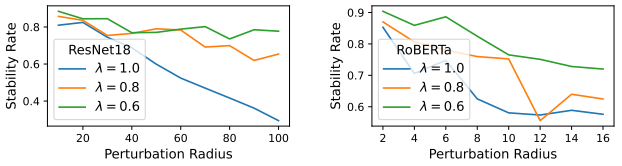

Figure 6. Mild smoothing improves the stability rate, particularly for weaker models. However, the improvement is not necessarily monotonic. The values reported are for when α is a random selection of 25% of the input features.

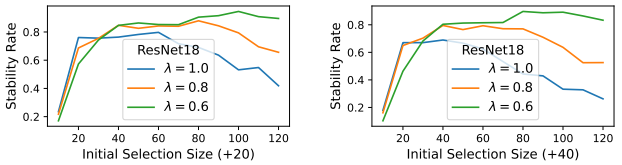

Figure 7. Larger initial feature selections lead to higher stability rates, which improve with greater smoothing (smaller λ). (Left) Stability rates at perturbation radius $r = 20$ for different initial selection sizes of α. (Right) Stability rates at perturbation $r = 40$.

hard stability does. We can see that soft stability effectively differentiates attribution methods, with LIME and SHAP showing a sizable advantage over IntGrad, MFABA, and random baselines across all radii. In contrast, hard stability certifies overly low radii for all methods, making it ineffective for distinguishing stability differences. Note that a caveat of the soft stability bounds is that they are inherently probabilistic, which directly contrasts with the deterministic style of hard stability. To remedy this, one can always take more samples to get closer to the true soft stability rate.

Result 2: Mildly Smoothing Preserves Accuracy We observe that mild amounts of smoothing suffices to maintain accuracy. We first study the effect of smoothing on the classifier accuracy, wherein we observe that mildly smoothing suffices to maintain accuracy. We plot our results in Figure 5, 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 the smoothing parameter close to what is used in parameter used in many hard stability experiments, We use $N = 64$ samples from the Bernoulli distribution when evaluating the smoothed classifier $M_\lambda f$.

Result 3: Smoothing Improves Stability We study the effect of the smoothing parameter on the stability rate. We observe that smoothing improves stability but that this gain is not necessarily uniform. We show our results in Figure 6.

For the vision dataset, we use a random sample of $N = 50$ images from our ImageNet subset, and for each image, we randomly select 25% of the input features to be α. For the language dataset, we use only those in the emotions dataset with input token sequences of length ≥ 40, of which there were 50 items, where we similarly choose to include 25% of them in α. We do this because the average token length is only 28, so having too great of a perturbation radius might accidentally reveal too many features.

Result 4: Stability Improves with Larger Selections We analyze how the size of the initial feature selection α affects the stability rate across different levels of smoothing. As shown in Figure 7, larger initial feature sizes have diminishing returns on stability for an unsmoothed classifier. However, applying smoothing mitigates this drop. We use a random sample of $N = 50$ images.

6. Related Work

Feature-based Explanations Feature attributions have long been used in explainability and remain popular. Early examples include gradient saliency (Simonyan et al., 2013), LIME (Ribeiro et al., 2016), SHAP (Lundberg and Lee, 2017), and Integrated Gradients (Sundararajan et al., 2017). More recent works include DIME (Lyu et al., 2022), LAFA (Zhang et al., 2022), CAFE (Dejl et al., 2023), DoRaR (Qin et al., 2024), MFABA (Zhu et al., 2024), various Shapley value-based methods (Sundararajan and Najmi, 2020), and methods based on influence functions (Basu et al., 2020; Koh and Liang, 2017). Moreover, while feature attributions are commonly associated with vision models, they are also used in language modeling (Lyu et al., 2024). For general surveys on explainability, we refer to Milani et al. (2024); Schwalbe and Finzel (2024). For explainability in medicine, we refer to Klauschen et al. (2024); Patrício et al. (2023). For explainability in law, we refer to (Amarasinghe et al., 2023; Richmond et al., 2024). Furthermore, "stability" is a widely used and overloaded term in the explainability literature, but many definitions relate to some notion of robustness (Nauta et al., 2023).

Evaluating Feature Attributions Although feature attributions are popular, their correctness and usefulness have often been called into question (Adebayo et al., 2018; Dinu et al., 2020; Kindermans et al., 2019). This is because each attribution method computes importance by a different measure, which may not necessarily be indicative of the underlying model behavior (Adebayo et al., 2022; Zhou et al., 2022), as well as theoretical results on their limitations (Bilodeau et al., 2024). This has prompted a large number of evaluation metrics for feature attributions (Agarwal et al., 2022; Jin et al., 2024; Nauta et al., 2023; Rong et al., 2022), in particular for various notions of robustness (Gan et al., 2022; Kamath et al., 2024).

Certifying Feature Attributions While many empirical metrics exist, there is also growing interest in ensuring that feature attributions are well-behaved through formal, mathematical guarantees. In particular, there is interest in certifying the robustness properties of adding (Xue et al., 2024) and removing (Lin et al., 2024) features from an attribution. There is also work on selecting feature sets that are provably optimal in some sense (Blanc et al., 2021). However, the literature on explicit guarantees for feature attributions is still emerging, largely because formalizing desirable properties and algorithmically certifying them is difficult.

7. Discussion

Probabilistic Guarantees We investigate probabilistic guarantees, which are different than the kinds of guarantees usually studied in robust certification. Namely, probabilistic guarantees only require that a certain property holds with high probability, whereas certified robustness guarantees desire that *all* things hold.

Boolean Function Analysis for Explainability Our analysis leverages Boolean function techniques to better understand stability. The monotone basis perspective reveals that instability is largely driven by high-order feature interactions, which grow combinatorially with perturbation radius. Our analysis shows that smoothing can exponentially suppress high-order interactions while preserving key low-order terms, leading to improved stability. This explains why small amounts of smoothing can yield disproportionately large gains in stability.

Towards More Stable Explanations Our work suggests an alternative in robustness research: rather than focusing solely on model Lipschitzness, we should analyze the distribution of monotone basis coefficients. This perspective enables new approaches, such as regularizing high-order terms or constructing explanations from simpler components, to enhance stability without compromising accuracy.

Balancing Stability and Fidelity Our analysis shows that mild smoothing ($\lambda \approx 1$) improves stability by exponentially suppressing high-order interactions while preserving essential low-order structure. However, stronger smoothing could also distort attributions, potentially reducing their fidelity to the model's true decision process. The choice of λ is therefore crucial: while it guarantees at least a factor of λ improvement in stability, its effect on fidelity depends on the distribution of $|\widetilde{h}(T)|$ across different set sizes. Our experiments confirm that multiplicative smoothing enhances stability without significantly degrading accuracy, suggesting that careful tuning of λ allows for a balance between stability and faithful attribution.

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.

Impact Statement

This paper presents work whose goal is to advance the field of Machine Learning by making machine learning models more interpretable and trustworthy to human practitioners. There are many potential societal consequences of our work, none which we feel must be specifically highlighted here.

References

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.

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.

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.

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.

Anastasios N Angelopoulos, Stephen Bates, et al. Conformal prediction: A gentle introduction. *Foundations and Trends® in Machine Learning*, 16(4):494–591, 2023.

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

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.

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.

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

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.

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

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

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

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

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

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

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

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.

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.

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.

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.

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.

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.

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.

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.

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.

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.

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.

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

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.

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

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

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

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

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.

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

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.

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.

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

Saif Mohammad, Felipe Bravo-Marquez, Mohammad Salameh, and Svetlana Kiritchenko. Semeval-2018 task 1: Affect in tweets. In *Proceedings of the 12th international workshop on semantic evaluation*, pages 1–17, 2018.

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.

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

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.

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.

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.

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.

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.

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.

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.

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

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

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

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

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.

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.

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.

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

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.

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.

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

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.

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

We give an analysis of the smoothing operator as described in Section 4. Recall that this is defined as follows.

Definition A.1 (Multiplicative Smoothing)**.** For any function $f : \mathbb{R}^n \to \mathbb{R}^m$ and smoothing parameter $\lambda \in [0, 1]$, define the multiplicative smoothing operator M_λ as:

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

Definition A.2 (p-Biased Basis)**.** For any $S \subseteq [n]$, define its corresponding p-biased Fourier basis function as:

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

Proposition A.3 (Orthonormality of p-Biased Basis)**.** *The p-biased basis functions χ_S^p are orthonormal with respect to the distribution $\mathrm{Bern}(p)^n$. Specifically, for any $S, T \subseteq [n]$,*

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

Proof. For any coordinate $i \in [n]$, note the following identities:

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

The inner product is then:

$$\begin{aligned} \mathop{\mathbb{E}}_{\alpha\sim \mathrm{Bern}(p)^n} [\chi_S^p(\alpha)\chi_T^p(\alpha)] &= \mathop{\mathbb{E}}_{\alpha} \left[\prod_{i\in S} \frac{p - \alpha_i}{\sqrt{p - p^2}} \prod_{j\in T} \frac{p - \alpha_j}{\sqrt{p - p^2}} \right] \\ &= \underbrace{\prod_{i\in S\cap T} \mathop{\mathbb{E}}_{\alpha} \left[\left(\frac{p - \alpha_i}{\sqrt{p - p^2}} \right)^2 \right]}_{=1, \text{ for any } S \text{ and } T} \underbrace{\prod_{j\in S\triangle T} \mathop{\mathbb{E}}_{\alpha} \left[\frac{p - \alpha_j}{\sqrt{p - p^2}} \right]}_{=0, \text{ if } S\triangle T \neq \emptyset} \end{aligned}$$

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

Lemma A.4 (Change-of-Basis via Smoothing)**.** *For any p-biased basis function χ_S^p and smoothing parameter $\lambda \in [p, 1]$,*

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

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

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

where we swapped the expectation and products using the coordinate-wise independence of $z_1, \ldots, z_n$. We then rewrite the above in terms of a (p/λ)-biased basis function as follows:

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

□

Theorem A.5 (Smoothing Reduces Variance)**.** *For any function $h : \{0,1\}^n \to \mathbb{R}$ and smoothing parameter $\lambda \in [p, 1]$,*

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

Proof. We use the previous results to compute:

$$\begin{aligned} \operatorname*{Var}_{\alpha \sim \mathrm{Bern}(p/\lambda)^n} [M_\lambda h(\alpha)] &= \operatorname*{Var}_{\alpha \sim \mathrm{Bern}(p/\lambda)^n} \left[M_\lambda \sum_{S \subseteq [n]} \widehat{h}(S) \chi_S^p(\alpha) \right] && \text{(Unique } p\text{-biased representation of } h) \\ &= \operatorname*{Var}_{\alpha \sim \mathrm{Bern}(p/\lambda)^n} \left[\sum_{S \subseteq [n]} \left(\frac{\lambda - p}{1 - p} \right)^{|S|/2} \widehat{h}(S) \chi_S^{p/\lambda}(\alpha) \right] && \text{(Linearity and Lemma A.4)} \\ &= \sum_{S \neq \emptyset} \left(\frac{\lambda - p}{1 - p} \right)^{|S|} \widehat{h}(S)^2 && \text{(Parseval's theorem by orthonormality of } \chi_S^{p/\lambda}) \\ &\leq \left(\frac{\lambda - p}{1 - p} \right) \sum_{S \neq \emptyset} \widehat{h}(S)^2 && (0 \leq \tfrac{\lambda - p}{1 - p} \leq 1 \text{ because } p \leq \lambda \leq 1) \\ &= \left(\frac{\lambda - p}{1 - p} \right) \operatorname*{Var}_{\alpha \sim \mathrm{Bern}(p)^n} [h(\alpha)] && \text{(Parseval's by orthonormality of } \chi_S^p) \end{aligned}$$

□

B. Analysis of Stability and Smoothing in the Monotone Basis

The analysis of feature attribution stability naturally leads to studying Boolean functions under one-way perturbations. While Fourier analysis is the standard tool for Boolean function analysis, it has key limitations for our setting. First, it treats 0→1 and 1→0 transitions symmetrically, making it harder to analyze perturbations that only add features ($\beta \geq \alpha$) and smoothing operations that only remove features (via masking). Second, traditional spectral analysis focuses on global properties, while our stability guarantees are inherently local (they depend on the specific attribution α). This asymmetry in our setting, combined with our focus on mild smoothing ($\lambda \approx 1$), motivates the development of new analytical tools beyond standard Fourier analysis.

B.1. Monotone Basis for Boolean Functions

To respect this one-way nature of perturbations, we introduce a monotone basis. For any set $T \subseteq [n]$:

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

Unlike the standard Fourier basis, the monotone basis is not orthonormal. However, it satisfies certain desirable properties:

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

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

where $\widetilde{h}(T)$ are the monotone basis coefficients of h, $\widetilde{h}(\emptyset)$ is a constant term, and $\mathbf{1}_\emptyset(\alpha) = 1$ for all α. The basis functions satisfy:

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

and the coefficients can be computed recursively:

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

where $h(T)$ means evaluating h on the attribution with 1's exactly at positions in T.

Proof of Lemma B.1. First, we prove existence and uniqueness. For any attribution α, let $S_\alpha = \{i : \alpha_i = 1\}$ be its support. By definition of $\mathbf{1}_T$:

$$\begin{aligned} h(\alpha) &= \widetilde{h}(\emptyset) + \sum_{\substack{T \subseteq [n] \\ T \neq \emptyset}} \widetilde{h}(T)\mathbf{1}_T(\alpha) \\ &= \widetilde{h}(\emptyset) + \sum_{T \subseteq S_\alpha} \widetilde{h}(T) && \text{(support restriction)} \end{aligned}$$

This gives a system of 2^n linear equations (one for each α) in 2^n unknowns (the coefficients $\widetilde{h}(T)$). When we order both attributions and sets by inclusion, for each set T, all proper subsets $S \subsetneq T$ appear before T in the ordering. This creates an upper triangular matrix with 1's on the diagonal (since $\mathbf{1}_T(T) = 1$ and $\mathbf{1}_T(S) = 0$ for $|S| < |T|$), proving existence and uniqueness.

For the inner product formula:

$$\begin{aligned} \mathop{\mathbb{E}}_{\alpha} \left[\mathbf{1}_S(\alpha)\mathbf{1}_T(\alpha)\right] &= \Pr_{\alpha} \left[\alpha_i = 1 \text{ for all } i \in S \cup T\right] && \text{(product rule)} \\ &= 2^{-|S \cup T|} && \text{(uniform distribution)} \end{aligned}$$

For the recursive formula, fix a set T and consider $h(T)$. By the expansion:

$$\begin{aligned} h(T) &= \widetilde{h}(\emptyset) + \sum_{S \subseteq T} \widetilde{h}(S) && \text{(basis expansion)} \\ &= \widetilde{h}(T) + \widetilde{h}(\emptyset) + \sum_{S \subsetneq T} \widetilde{h}(S) && \text{(split largest term)} \end{aligned}$$

Rearranging gives the recursive formula:

$$\widetilde{h}(T) = h(T) - \sum_{S \subsetneq T} \widetilde{h}(S) \qquad \text{(recursion)}$$

□

To build intuition for this basis, consider the following example:

Example B.2 (Conjunction vs Fourier). Consider the conjunction of two features: $h(\alpha) = \alpha_1 \wedge \alpha_2$. This function outputs 1 only when both features are present.

In the standard Fourier basis with $\chi_T(\alpha) = (-1)^{|\operatorname{supp}(\alpha) \cap T|}$:

$$h(\alpha) = \frac{1}{4} + \frac{1}{4}\chi_{\{1\}}(\alpha) + \frac{1}{4}\chi_{\{2\}}(\alpha) + \frac{1}{4}\chi_{\{1,2\}}(\alpha)$$

showing complex interactions between features.

In contrast, in the monotone basis:

$$h(\alpha) = \mathbf{1}_{\{1,2\}}(\alpha)$$

This single coefficient directly captures the AND operation: the function is 1 exactly when both features are present.

B.2. Connection between Stability and Monotone Basis Expansion

We will focus on $f : \mathbb{R}^n \to \mathbb{R}$ (binary classification) and consider the following notion of model prediction equivalence.

Definition B.3 (Model Prediction Equivalence)**.** For a classifier $f : \mathcal{X} \to \mathcal{Y}$ and input x, we say two predictions are equivalent, denoted $f(x \odot \beta) \cong f(x \odot \alpha)$, if:

$$|f(x \odot \beta) - f(x \odot \alpha)| \leq 1/2$$

For binary classifiers where $\mathcal{Y} = \{0, 1\}$, this means the predictions must be identical. For probabilistic classifiers where $\mathcal{Y} = [0, 1]$, this allows for small variations in confidence while preserving the predicted class.

The monotone basis allows us to derive tight bounds on both soft and hard stability. We begin with soft stability:

Lemma B.4 (Soft Stability)**.** *For any Boolean function $h : \{0,1\}^n \to [0,1]$ and attribution α, the stability rate τ_r satisfies:*

$$1 - \tau_r \leq \frac{2}{\sum_{i=0}^{r} \binom{n-|\alpha|}{i}} \sum_{\substack{j=1 \\ S:|S|=j \\ S \cap \operatorname{supp}(\alpha) = \emptyset}}^{r} \left| \sum_{T \subseteq S \setminus \{\emptyset\}} \widetilde{h}(T) \right|$$

where $\widetilde{h}(T)$ are the coefficients in the monotone basis.

Proof of Lemma B.4. We begin with the definition of stability rate. By Markov's inequality:

$$\begin{aligned} 1 - \tau_r &= \Pr_{\beta \sim \Delta_r(\alpha)} \left[|h(\beta) - h(\alpha)| > 1/2 \right] \\ &\leq 2 \mathop{\mathbb{E}}_{\beta \sim \Delta_r(\alpha)} \left[|h(\beta) - h(\alpha)| \right] \qquad \text{(Markov)} \end{aligned}$$

To analyze the difference $h(\beta) - h(\alpha)$, we express it using the monotone basis:

$$h(\beta) - h(\alpha) = \sum_{T \subseteq [n]} \widetilde{h}(T) \big(\mathbf{1}_T(\beta) - \mathbf{1}_T(\alpha) \big)$$

Since perturbations only add features ($\beta \geq \alpha$), the difference in indicator functions simplifies considerably. For any set T:

$$\mathbf{1}_T(\beta) - \mathbf{1}_T(\alpha) = \begin{cases} 1 & \text{if } T \neq \emptyset \text{ and } \alpha_i = 0, \beta_i = 1 \text{ for all } i \in T \\ 0 & \text{otherwise} \end{cases}$$

This allows us to rewrite the difference as a sum over only the relevant sets:

$$h(\beta) - h(\alpha) = \sum_{T : \alpha_i = 0, \beta_i = 1 \text{ for all } i \in T \setminus \{\emptyset\}} \widetilde{h}(T)$$

To compute the expectation of $|h(\beta) - h(\alpha)|$, we first need to understand the structure of this difference for any fixed β. Note that β is completely determined by the set of positions S where it differs from α (where zeros become ones). By construction of $\Delta_r(\alpha)$, this set S must satisfy two properties: $|S| = j$ for some $j \leq r$, and $S \cap \text{supp}(\alpha) = \emptyset$ since we can only flip zeros to ones.

For such a fixed set S, we can simplify our expression for $h(\beta) - h(\alpha)$:

$$h(\beta) - h(\alpha) = \sum_{T \subseteq S \setminus \{\emptyset\}} \widetilde{h}(T)$$

This simplification follows because a set T contributes to the difference if and only if it is contained in S (the positions where β differs from α).

Now we can compute the expectation by considering how β is sampled under $\Delta_r(\alpha)$. The sampling process has two steps: first choose the number of positions j to flip with probability $\frac{\binom{n-|\alpha|}{j}}{\sum_{i=0}^{r}\binom{n-|\alpha|}{i}}$, then uniformly select j positions from the zeros in α. This gives us:

$$\mathop{\mathbb{E}}_{\beta}\left[|h(\beta) - h(\alpha)|\right] = \sum_{j=1}^{r} \sum_{\substack{S:|S|=j \\ S\cap\text{supp}(\alpha)=\emptyset}} \frac{\binom{n-|\alpha|}{j}}{\sum_{i=0}^{r}\binom{n-|\alpha|}{i}} \left| \sum_{T \subseteq S \setminus \{\emptyset\}} \widetilde{h}(T) \right|$$

Combining this with our initial Markov inequality bound completes the proof:

$$1 - \tau_r \leq \frac{2}{\sum_{i=0}^{r}\binom{n-|\alpha|}{i}} \sum_{\substack{j=1 \\ S:|S|=j \\ S\cap\text{supp}(\alpha)=\emptyset}}^{r} \left| \sum_{T \subseteq S \setminus \{\emptyset\}} \widetilde{h}(T) \right|$$

□

Here we present a simplification of the soft-stability bound above to make it easier to parse.

Lemma B.5 (Simplified Soft Stability Bound)**.** *Under the same conditions, we also have:*

$$1 - \tau_r \leq 2\sum_{k=1}^{r} \sum_{\substack{T:|T|=k \\ T\cap\text{supp}(\alpha)=\emptyset}} |\widetilde{h}(T)| \cdot \frac{\sum_{j=k}^{r}\binom{n-|\alpha|-k}{j-k}}{\sum_{j=0}^{r}\binom{n-|\alpha|}{i}}$$

Proof. Starting from the bound in Lemma B.4:

$$\begin{aligned}
1 - \tau_r &\leq \frac{2}{\sum_{i=0}^{r}\binom{n-|\alpha|}{i}} \sum_{j=1}^{r} \sum_{\substack{S:|S|=j \\ S\cap\text{supp}(\alpha)=\emptyset}} \left| \sum_{T \subseteq S \setminus \{\emptyset\}} \widetilde{h}(T) \right| \\
&\leq \frac{2}{\sum_{i=0}^{r}\binom{n-|\alpha|}{i}} \sum_{j=1}^{r} \sum_{\substack{S:|S|=j \\ S\cap\text{supp}(\alpha)=\emptyset}} \sum_{T \subseteq S \setminus \{\emptyset\}} |\widetilde{h}(T)| && \text{(triangle inequality)} \\
&= \frac{2}{\sum_{i=0}^{r}\binom{n-|\alpha|}{i}} \sum_{k=1}^{r} \sum_{\substack{T:|T|=k \\ T\cap\text{supp}(\alpha)=\emptyset}} |\widetilde{h}(T)| \sum_{j=k}^{r} \binom{n-|\alpha|-k}{j-k} && \text{(reorder sums)} \\
&= 2\sum_{k=1}^{r} \sum_{\substack{T:|T|=k \\ T\cap\text{supp}(\alpha)=\emptyset}} |\widetilde{h}(T)| \cdot \frac{\sum_{j=k}^{r}\binom{n-|\alpha|-k}{j-k}}{\sum_{j=0}^{r}\binom{n-|\alpha|}{i}} && \text{(rearrange)}
\end{aligned}$$

The derivation proceeds in three steps. We begin by applying the triangle inequality to separate the coefficients. Next, we reorder the summation to group terms by coefficient size. Finally, we count the occurrences of each coefficient in the sum. The final expression weights each coefficient $\widetilde{h}(T)$ by $\frac{\sum_{j=k}^{r}\binom{n-|\alpha|-k}{j-k}}{\sum_{j=0}^{r}\binom{n-|\alpha|}{i}}$, which is the probability that a random perturbation contains set T of size k. □

Observe that the bound depends only on the monotone expansion terms of degree $\leq r$.

We can use the same technique above to derive a hard stability bound in terms of the monomial expansion as well.

Lemma B.6 (Hard Stability Bound)**.** *For any Boolean function $h : \{0,1\}^n \to [0,1]$ and attribution α, let*

$$r^* = \max\left\{ r \geq 0 : \max_{\substack{S\subseteq[n]\\ 1\leq|S|\leq r\\ S\cap\mathrm{supp}(\alpha)=\emptyset}} \left| \sum_{T\subseteq S\setminus\{\emptyset\}} \widetilde{h}(T) \right| \leq \frac{1}{2} \right\}$$

Then h is hard-stable at radius r^*.

Proof. For any $\beta \in \Delta_r(\alpha)$:

$$|h(\beta) - h(\alpha)| = \left| \sum_{T\subseteq \mathrm{diff}(\beta,\alpha)\setminus\{\emptyset\}} \widetilde{h}(T) \right|$$

since $\mathrm{diff}(\beta,\alpha)$ is always a non-empty subset of size at most r disjoint from $\mathrm{supp}(\alpha)$. By definition of r^*, $|h(\beta) - h(\alpha)| \leq 1/2$ for all $\beta \in \Delta_{r^*}(\alpha)$, proving hard stability. □

B.3. Stability Bound for Smoothed Distribution

The monotone basis allows us to capture the smoothing operator as a simple transformation of the monomial expansion.

Theorem B.7 (Smoothing in Monotone Basis)**.** *Let M_λ be the smoothing operator that randomly masks features with probability $1-\lambda$:*

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

where z represents a random mask and $\odot$ denotes element-wise multiplication as defined in Section **??**.

For any Boolean function $h : \{0,1\}^n \to [0,1]$, the smoothed function $M_\lambda h$ in the monotone basis satisfies:

$$\widetilde{M_\lambda h}(T) = \begin{cases} \widetilde{h}(\emptyset) & \text{if } T = \emptyset \text{ (constant term preserved)} \\ \lambda^{|T|}\widetilde{h}(T) & \text{if } T \neq \emptyset \text{ (coefficients damped)} \end{cases}$$

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

Proof of Theorem B.7. First, note that M_λ is a linear operator since expectation is linear. For the empty set, $\widetilde{M_\lambda h}(\emptyset) = \widetilde{h}(\emptyset)$ since smoothing preserves constants.

For any non-empty set T:

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

The result follows by linearity of expectation. □

With the above theorem in hand, we can now compute the stability of the smoothed classifier:

Corollary B.8 (Stability of Smoothed Functions)**.** *For any Boolean function $h : \{0,1\}^n \to [0,1]$, attribution α, and smoothing parameter $\lambda \in [0,1]$, the stability rate of the smoothed function satisfies:*

$$1 - \tau_r(M_\lambda h) \leq 2 \sum_{k=1}^{r} \lambda^k \sum_{\substack{T:|T|=k \\ T \cap \mathrm{supp}(\alpha)=\emptyset}} |\widetilde{h}(T)| \cdot \frac{\sum_{j=k}^{r} \binom{n-|\alpha|-k}{j-k}}{\sum_{j=0}^{r} \binom{n-|\alpha|}{i}}$$

This bound reveals that smoothing improves stability by exponentially dampening the influence of larger feature sets.

Proof of Corollary B.8. Apply the stability bound from Lemma B.4 to $M_\lambda h$ and use Theorem B.7 which shows that $\widetilde{M_\lambda h}(T) = \lambda^{|T|}\widetilde{h}(T)$ for non-empty T:

$$\begin{aligned}
1 - \tau_r(M_\lambda h) &\leq 2 \sum_{k=1}^{r} \sum_{\substack{T:|T|=k \\ T \cap \mathrm{supp}(\alpha)=\emptyset}} |\widetilde{M_\lambda h}(T)| \cdot \frac{\sum_{j=k}^{r} \binom{n-|\alpha|-k}{j-k}}{\sum_{j=0}^{r} \binom{n-|\alpha|}{i}} && \text{(by Lemma B.4)} \\
&= 2 \sum_{k=1}^{r} \sum_{\substack{T:|T|=k \\ T \cap \mathrm{supp}(\alpha)=\emptyset}} \lambda^k |\widetilde{h}(T)| \cdot \frac{\sum_{j=k}^{r} \binom{n-|\alpha|-k}{j-k}}{\sum_{j=0}^{r} \binom{n-|\alpha|}{i}} && \text{(by Theorem B.7)} \\
&= 2 \sum_{k=1}^{r} \lambda^k \sum_{\substack{T:|T|=k \\ T \cap \mathrm{supp}(\alpha)=\emptyset}} |\widetilde{h}(T)| \cdot \frac{\sum_{j=k}^{r} \binom{n-|\alpha|-k}{j-k}}{\sum_{j=0}^{r} \binom{n-|\alpha|}{i}} && \text{(rearrange terms)}
\end{aligned}$$

The final expression shows how smoothing affects stability through three key mechanisms. First, each coefficient $\widetilde{h}(T)$ is weighted by λ^k where $k = |T|$. Second, larger sets T are dampened more strongly since λ^k decreases exponentially with k. Finally, the combinatorial terms $\frac{\sum_{j=k}^{r} \binom{n-|\alpha|-k}{j-k}}{\sum_{j=0}^{r} \binom{n-|\alpha|}{i}}$ represent the probability of including set T in a random perturbation. □

Remark B.9 (Smoothing Effect). The upper bound for the smoothed function in Corollary B.8 is at least a factor of λ smaller than the upper bound for the original function, since $\lambda^k \leq \lambda$ for all $k \geq 1$. However, these are only upper bounds - the actual improvement from smoothing could be either better or worse than suggested by comparing these bounds.

B.4. Discussion and Practical Implications

Our analysis through the monotone basis reveals some key mechanisms affecting stability. First, mild smoothing ($\lambda \approx 1$) can be effective because it exponentially dampens higher-order terms while preserving essential low-order structure—for instance, with $\lambda = 0.9$, single-feature terms are dampened by 0.9 while five-feature terms are dampened by $0.9^5 \approx 0.59$. While our bounds guarantee at least a factor of λ improvement in stability (since $\lambda^k \leq \lambda$ for all $k \geq 1$), the actual improvement could be either better or worse in practice. Second, stability becomes harder to maintain at larger radii because both the number of terms and their combinatorial weights grow with r, suggesting that λ should be chosen based on the distribution of $|\widetilde{h}(T)|$ across different set sizes. These insights are validated by our experiments in Section 5, where we show that our multiplicative smoothing improves stability without significantly degrading accuracy (Q2).

While this work establishes the theoretical foundations, we could use these insights to design new attribution methods that explicitly control the monotone basis expansion of their output—for instance, by regularizing higher-order coefficients or by constructing explanations primarily from small low-order terms. This suggests a new shift approach to attribution stability: rather than focusing solely on Lipschitz constants of the model, we should study the distribution of monotone basis coefficients, as these more directly capture the stability properties we care about.

C. Additional Experiments and Figures

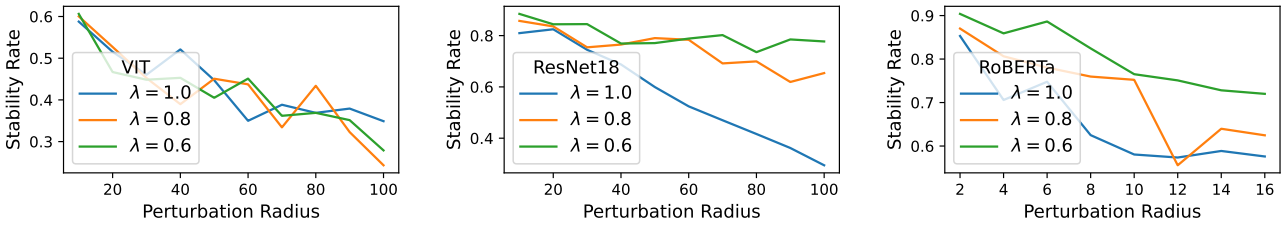

Figure 8. The stability rates of different classifiers when 25% of the features are selected. Smoothing tends to be more effective in improving the stability rate on weaker models.

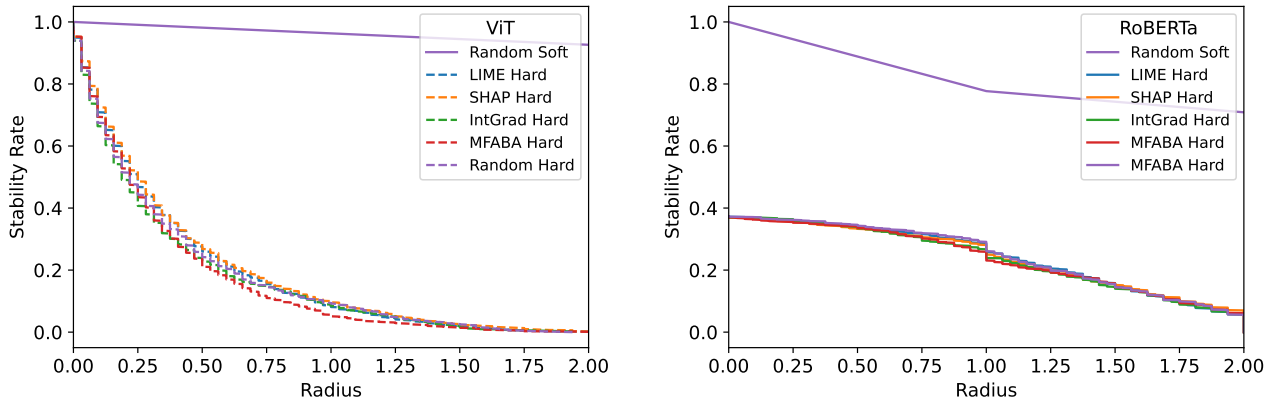

Figure 9. Soft vs. Hard Stability. (Left) The hard stability rates for Vision Transformer on different explanation methods compared to the soft stability rate on a random explanation as a function of the radius. (Right) The hard stability rates for RoBERTa on different explanation methods compared to the soft stability rate on a random explanation as a function of the radius.

We zoom in and focus on the hard stability rates for different explanation methods (e.g., SHAP 25%), and show how they compare to the soft stability rate for a random explanation at that radius range. All explanations are 25% of the entire input, and curves are averaged over the entire dataset. Note that the soft stability rates for the different explanation methods (LIME, SHAP, IntGrad, MFABA) are similarly high up in comparison to the hard stability rates. While the hard stability curves stop at radius = 2, the soft stability curves continue and fill in a much larger radius range. We did not show the full curves of the soft stability so that we can showcase the hard stability cases more closely. The gap between hard stability and soft stability is significant, showcasing that soft stability is much more powerful and capable, even for larger radii.

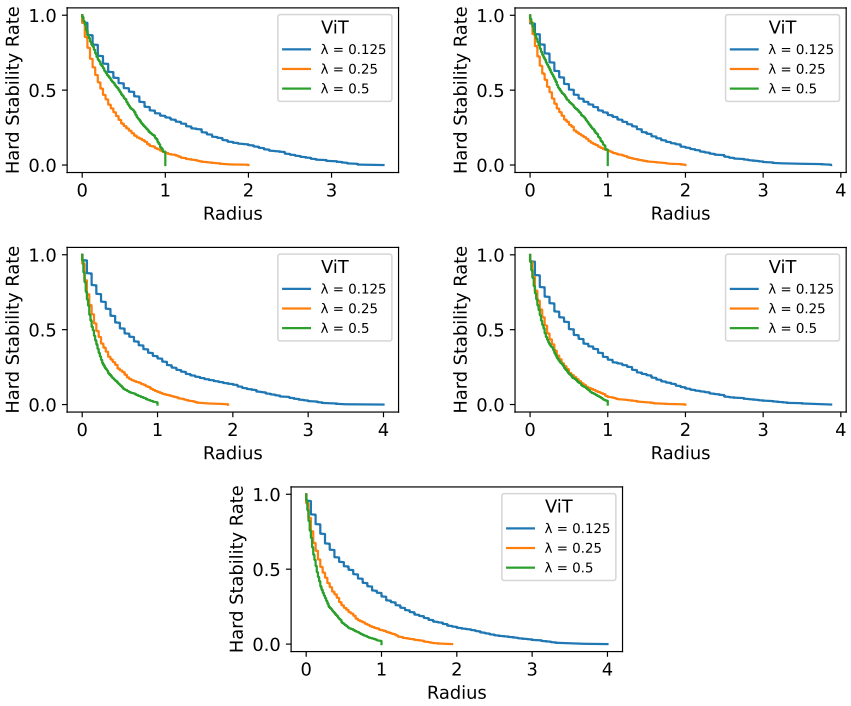

Figure 10. Hard stability rates for varying lambda parameters, Vision Transformer.

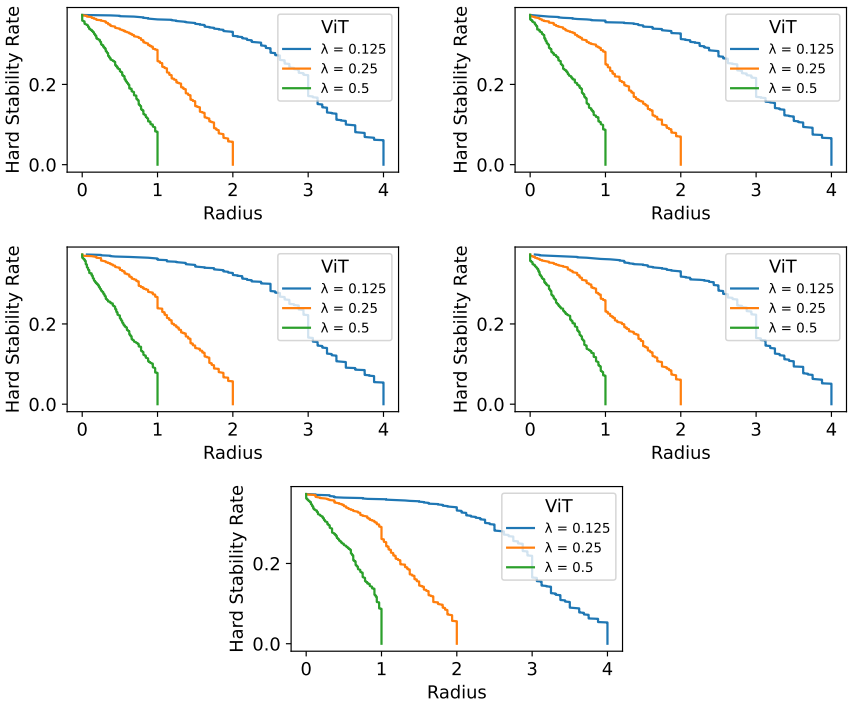

Figure 11. Hard stability rates for varying lambda parameters, RoBERTa.