RETHINKING STABILITY FOR ATTRIBUTION-BASED EXPLANATIONS

Chirag Agarwal[1], **Nari Johnson**[2], **Martin Pawelczyk**[3], **Satyapriya Krishna**[4],
Eshika Saxena[4], **Marinka Zitnik**[4] & **Himabindu Lakkaraju**[4]
[1] Media and Data Science Research Lab, Adobe
[2] Carnegie Mellon University
[3] University of Tübingen
[4] Harvard University

ABSTRACT

As attribution-based explanation methods are increasingly used to establish model trustworthiness in high-stakes situations, it is critical to ensure that these explanations are stable, e.g., robust to infinitesimal perturbations to an input. However, previous works have shown that state-of-the-art explanation methods generate unstable explanations. Here, we introduce metrics to quantify the stability of an explanation and show that several popular explanation methods are unstable. In particular, we propose new *Relative Stability* metrics that measure the change in output explanation with respect to change in input, model representation, or output of the underlying predictor. Finally, our experimental evaluation with three real-world datasets demonstrates interesting insights for seven explanation methods and different stability metrics.

1 INTRODUCTION

With machine learning (ML) models being increasingly employed in high-stakes domain such as criminal justice, finance, and healthcare, it is essential to ensure that the relevant stakeholders understand these models' decisions. However, existing approaches to explain the predictions of complex machine learning (ML) models suffer from several critical shortcomings. Recent works have shown that explanations generated using attribution-based methods are not stable (Ghorbani et al., 2019; Slack et al., 2020; Dombrowski et al., 2019; Adebayo et al., 2018; Alvarez-Melis & Jaakkola, 2018; Bansal et al., 2020), e.g. that infinitesimal perturbations to an input can result in substantially different explanations.

Existing metrics (Alvarez-Melis & Jaakkola, 2018) measure the change in explanation only with respect to the input perturbations, e.g., they only assume black-box access to the predictive model, and don't leverage potentially meaningful information such as the model's internal representations to evaluate stability. To address these limitations of existing stability metrics, we propose *Relative Stability* that measures the change in output explanation with respect to the behavior of the underlying predictive model (Section 3.3). Finally, we present extensive theoretical and empirical analysis (Section 4.2) for comparing the stability of seven state-of-the-art explanation methods using multiple real-world datasets.

2 RELATED WORKS

This paper draws from two main areas of prior work: 1) attribution-based explanation methods, and 2) stability analysis of explanations.

Attribution-based Explanation Methods. While a variety of approaches have been proposed to explain model decisions for classifiers, our work focuses on *local feature attribution explanations*, which measure the contribution of each feature to the model's prediction on a point. In particular, we study two broad types of feature attribution explanations: gradient-based and approximation-based. Gradient-based feature attribution methods like VanillaGrad (Simonyan et al., 2014), SmoothGrad (Smilkov et al., 2017), Integrated Gradients (Sundararajan et al., 2017), and Gradient×Input (Shrikumar et al., 2017) leverage model gradients to quantify how a change in each feature would affect the model's prediction. Approximation-based methods like LIME (Ribeiro et al., 2016), SHAP (Lundberg & Lee, 2017), Anchors (Ribeiro et al., 2018), BayesLIME, and BayesSHAP (Slack et al., 2021) leverage perturbations of individual inputs to construct a local approximation model from which feature attributions are derived.

Explanation Stability. Recent works have formalized desirable properties for feature attribution explanations (Agarwal et al., 2022). Our work specifically focuses on the *stability* of explanations. Alvarez-Melis & Jaakkola (2018) argued that "similar inputs should lead to similar explanations" and is the first work to formalize a metric to measure the stability of local explanation methods. We highlight potential issues with this stability metric that measures stability only *w.r.t.* the change in *input*.

3 Stability Analysis for Evaluating Explanations

3.1 Notation and Preliminaries

Machine Learning Model. Given a feature domain $\mathcal{X}$ and label domain $\mathcal{Y}$, we denote a classification model $f\colon \mathcal{X} \rightarrow \mathcal{Y}$ that maps a set of features $\mathbf{x} {\in} \mathcal{X}$ to labels $\mathbf{y} {\in} \mathcal{Y}$, where $\mathbf{x} \in \mathbb{R}^d$ is a d-dimensional feature vector, $\mathbf{y} \in \{0, 1, \ldots, \mathrm{C}\}$ where C is the total number of classes in the dataset. We use $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N\}$ to denote all the N instances in the dataset. In addition, we define $f(\mathbf{x}){=}\sigma(h(\mathbf{x}))$, where $h : \mathcal{X} \rightarrow \mathbb{R}$ is a scoring function (e.g., logits) and $\sigma : \mathbb{R} \rightarrow \mathcal{Y}$ is an activation function that maps output logit scores to discrete labels. Finally, for a given input $\mathbf{x}$, the output predicted class label is: $\hat{y}_{\mathbf{x}}{=} \arg\max_c f(\mathbf{x})$. We assume access to the gradients and intermediate representations of model f.

Explainability Methods. An attribution-based explanation method $\mathcal{E}$ generates an explanation $\mathbf{e}_{\mathbf{x}} \in \mathbb{R}^d$ to explain model prediction $f(\mathbf{x})$. To calculate our stability metrics, we generate perturbations $\mathbf{x}'$ by adding infinitesimal noise to $\mathbf{x}$, and denote their respective explanation as $\mathbf{e}_{\mathbf{x}'}$.

3.2 Existing Definition and Problems

Alvarez-Melis & Jaakkola (2018) formalize the first stability metric for local explanation methods, arguing that explanations should be robust to local perturbations of an input. To evaluate the stability of an explanation for instance $\mathbf{x}$, perturbed instances $\mathbf{x}'$ are generated by adding infinitesimally small noise to the clean instance $\mathbf{x}$ such that $\hat{y}_{\mathbf{x}} = \hat{y}_{\mathbf{x}'}$:

$$\mathrm{S}(\mathbf{x}, \mathbf{x}', \mathbf{e}_{\mathbf{x}}, \mathbf{e}_{\mathbf{x}'}) = \max_{\mathbf{x}'} \frac{|| \, \mathbf{e}_{\mathbf{x}} - \mathbf{e}_{\mathbf{x}'} \, ||}{|| \, \mathbf{x} - \mathbf{x}' \, ||}, \; \forall \mathbf{x}' \text{ s.t. } \mathbf{x}' \in \mathcal{N}_{\mathbf{x}}; \; \hat{y}_{\mathbf{x}} = \hat{y}_{\mathbf{x}'} \tag{1}$$

where $\mathcal{N}_{\mathbf{x}}$ is a neighborhood of instances $\mathbf{x}'$ similar to $\mathbf{x}$, and $\mathbf{e}_{\mathbf{x}}$ and $\mathbf{e}_{\mathbf{x}'}$ denote the explanations corresponding to instances $\mathbf{x}$ and $\mathbf{x}'$, respectively. For each point $\mathbf{x}'$, the stability ratio in Equation 1 measures how the output explanation varies with respect to the change in the *input*. Because the neighborhood of instances $\mathcal{N}_{\mathbf{x}}$ are sampled to be similar to the original instance $\mathbf{x}$, the authors argue that points that are similar should have similar model explanations, e.g., we desire the ratio in Equation 1 to be close to 1 (Alvarez-Melis et al., 2021). This stability definition relies on the point-wise neighborhood-based local Lipschitz continuity of the explanation method $\mathbf{e}_{\mathbf{x}}$ around $\mathbf{x}$.

Problems. We note two key problems with the existing stability definition: i) it only assumes black-box access to the prediction model f, and does not leverage potentially meaningful information such as the model's internal representations for evaluating stability; and ii) it implicitly assumes that f has the same *behavior* on inputs $\mathbf{x}$ and $\mathbf{x}'$ that are similar. While this may be the case for underlying prediction models that are smooth or robust, this assumption may not hold in a large number of cases. In Figure 1, we discuss a toy example where perturbed samples $\mathcal{N}_{\mathbf{x}}$ have drastically different intermediate representations than the original point $\mathbf{x}$. Note that since the goal of an explanation is to faithfully and accurately represent the behavior of the underlying prediction model (Agarwal et al., 2022), we argue that an explanation method *should* vary for points $\mathbf{x}$ and $\mathbf{x}'$ where the prediction model's behavior differs. Thus, we argue for the inclusion of new stability metrics that measure how much explanations vary with respect to the behavior of the underlying prediction model.

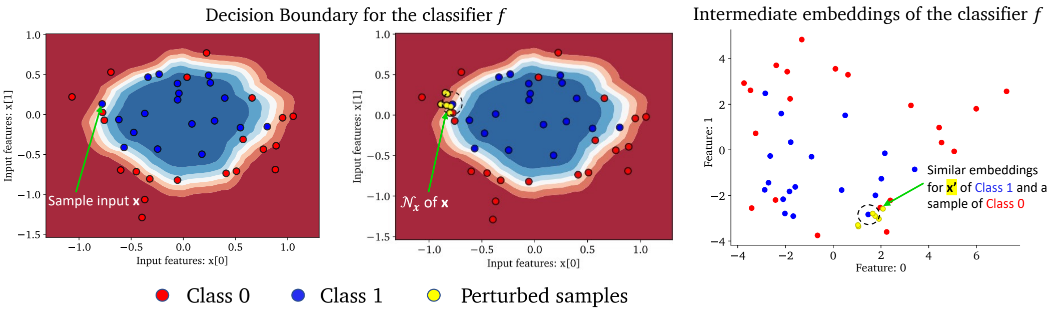


Figure 1: Decision boundaries and embeddings of a two-layer neural network predictor f with 100 units trained on the circles dataset. The heatmaps (left and middle column) shows the models' confidence for the positive-class (in blue), test set examples $\mathbf{x}$ (●, ●), and a set of perturbed samples $\mathbf{x}'$ (●). While all perturbed samples $\mathbf{x}'$ are predicted to the same class as $\mathbf{x}'$, the embeddings (right column) for some $\mathbf{x}'$ are far from the embeddings of $\mathbf{x}'$ and similar to the embeddings of Class 0, highlighting the need of incorporating the model behavior using its internal embeddings (Equations 3,5).

3.3 Proposed metric: Relative Stability

To address the aforementioned challenges, we propose *Relative Stability* that leverages model information to evaluate the stability of an explanation with respect to the change in the a) input data, b) intermediate representations, and c) output logits of the underlying prediction model.

a) Relative Input Stability. We extend the stability metric in Equation 1 and define *Relative Input Stability* that measures the relative distance between explanations $\mathbf{e}_\mathbf{x}$ and $\mathbf{e}_{\mathbf{x}'}$ with respect to the distance between the two inputs $\mathbf{x}$ and $\mathbf{x}'$.

$$\text{RIS}(\mathbf{x}, \mathbf{x}', \mathbf{e}_\mathbf{x}, \mathbf{e}_{\mathbf{x}'}) = \max_{\mathbf{x}'} \frac{||\frac{(\mathbf{e}_\mathbf{x} - \mathbf{e}_{\mathbf{x}'})}{\mathbf{e}_\mathbf{x}}||_p}{\max(||\frac{(\mathbf{x}-\mathbf{x}')}{\mathbf{x}}||_p, \epsilon_{min})}, \; \forall \mathbf{x}' \text{ s.t. } \mathbf{x}' \in \mathcal{N}_\mathbf{x};\; \hat{y}_\mathbf{x} = \hat{y}_{\mathbf{x}'} \tag{2}$$

where the numerator of the metric measures the ℓ_p norm of the *percent change* of explanation $\mathbf{e}_{\mathbf{x}'}$ on the perturbed instance $\mathbf{x}'$ with respect to the explanation $\mathbf{e}_\mathbf{x}$ on the original point $\mathbf{x}$, the denominator measures the ℓ_p norm between (normalized) inputs $\mathbf{x}$ and $\mathbf{x}'$ and the max term prevents division by zero in cases when norm $||\frac{(\mathbf{x}-\mathbf{x}')}{\mathbf{x}}||_p$ is less than some small ϵ_{min}>0. Here, we use the percent change from the explanation on the original point to the explanation on the perturbed instance in contrast to the absolute difference between the explanations (as in Equation 1) to enable comparison across different attribution-based explanation methods that have vastly different ranges or magnitudes. Intuitively, one can expect *similar* explanations for points that are similar – the percent change in explanations (numerator) should be *small* for points that are close, or have a *small* l_p norm (denominator). Note that the metric in Equation 2 measures instability of an explanation and higher values indicate higher instability.

b) Relative Representation Stability. Previous stability definitions in Equation 1-2 do not cater to cases where the model uses different logic paths (e.g., activating different neurons in a deep neural network) to predict the same label for the original and perturbed instance. In addition, past works have presented empirical evidence that the intermediate representations of a model are related to the underlying behavior or reasoning of the model (Agarwal et al., 2021). Thus, we leverage the internal features or representation learned by the underlying model and propose *Relative Representation Stability* as:

$$\text{RRS}(\mathbf{x}, \mathbf{x}', \mathbf{e}_\mathbf{x}, \mathbf{e}_{\mathbf{x}'}) = \max_{\mathbf{x}'} \frac{||\frac{(\mathbf{e}_\mathbf{x} - \mathbf{e}_{\mathbf{x}'})}{\mathbf{e}_\mathbf{x}}||_p}{\max(||\frac{(\mathcal{L}_\mathbf{x}-\mathcal{L}_{\mathbf{x}'})}{\mathcal{L}_\mathbf{x}}||_p, \epsilon_{min})}, \; \forall \mathbf{x}' \text{ s.t. } \mathbf{x}' \in \mathcal{N}_\mathbf{x};\; \hat{y}_\mathbf{x} {=} \hat{y}_{\mathbf{x}'} \tag{3}$$

where $\mathcal{L}(\cdot)$ denotes the internal model representation, e.g., output embeddings of hidden layers, and δ is an infinitesimal constant. Due to insufficient knowledge about the data generating mechanism, we follow the perturbation mechanisms described above to generate perturbed samples $\mathbf{x}'$ but use additional checks to ensure that for certain perturbations the model behaves similar to its training behavior. For any given instance $\mathbf{x}$, we generate m local perturbed samples such that $||\mathbf{x} - \mathbf{x}'||_p \leq \epsilon$, and $\hat{y}_\mathbf{x}{=}\hat{y}_{\mathbf{x}'}$. For every perturbed sample, we calculate the difference in their respective explanations and using Equation 3 calculate the relative stability of an explanation. Note that, as before, the metric in Equation 3 measures instability of an explanation and higher values indicate higher instability.

Finally, we show that the *Relative Input Stability* can be bounded using the Lipschitzness of the underlying model. In particular, we proof that RIS is upper bounded by a product of the Lipschitz constant L_1 of the intermediate model layer (assuming a neural network classifier) and our proposed *Relative Representation Stability*. See Appendix A for the complete proof.

$$\text{RIS}(\mathbf{x}, \mathbf{x}', \mathbf{e_x}, \mathbf{e_{x'}}) < \lambda_1 L_1 \times \text{ RRS}(\mathbf{x}, \mathbf{x}', \mathbf{e_x}, \mathbf{e_{x'}}) \tag{4}$$

c) Relative Output Stability. Note that Relative Representation Stability assumes that the underlying ML model is white-box, i.e., explanation method has access to the internal model knowledge. Hence, for black-box ML models we define *Relative Output Stability* as:

$$\text{ROS}(\mathbf{x}, \mathbf{x}', \mathbf{e_x}, \mathbf{e_{x'}}) = \max_{\mathbf{x}'} \frac{||\frac{(\mathbf{e_x} - \mathbf{e_{x'}})}{\mathbf{e_x}}||_p}{\max(||h(\mathbf{x}) - h(\mathbf{x}')||_p, \epsilon_{min})}, \; \forall \mathbf{x}' \text{ s.t. } \mathbf{x}' \in \mathcal{N}_{\mathbf{x}}; \; \hat{y}_{\mathbf{x}} = \hat{y}_{\mathbf{x}'} \tag{5}$$

where $h(\mathbf{x})$ and $h(\mathbf{x}')$ are the output logits for $\mathbf{x}$ and $\mathbf{x}'$, respectively. Again, we proof that RRS is upper bounded by a product of the Lipschitz constant L_2 of the output model layer and our proposed *Relative Output Stability*. See Appendix A for the complete proof.

$$\text{RRS}(\mathbf{x}, \mathbf{x}', \mathbf{e_x}, \mathbf{e_{x'}}) < \lambda_2 L_2 \; \times \; \text{ROS}(\mathbf{x}, \mathbf{x}', \mathbf{e_x}, \mathbf{e_{x'}}) \tag{6}$$

4 Experiments

To demonstrate the utility of relative stability, we systematically compare and evaluate the stability of seven explanation methods using three real-world datasets using equations defined in Section 3. Further, we show that, in contrast to relative input stability, relative representation and output stability better captures the stability of the underlying black-box model.

4.1 Datasets and Experimental Setup

Datasets. We use real-world structured datasets to empirically analyze the stability behavior of explanation methods and consider 3 benchmark datasets from high-stakes domains: i) the *German Credit* dataset (Dua & Graff, 2017) which has records of 1,000 clients in a German bank. The downstream task is to classify clients into good or bad credit risks, ii) the *COMPAS* dataset (Mattu et al., 2016) which has records of 18,876 defendants who got released on bail at the U.S state courts during 1990-2009. The dataset comprises of features representing past criminal records and demographics of the defendants and the goal is to classify them into bail or no bail, and iii) the *Adult* dataset (Dua & Graff, 2017) which has records of 48,842 individuals including demographic, education, employment, personal, and financial features. The downstream task is to predict whether an individual's income exceeds $50K per year.

Predictors. We train logistic regression (LR) and artificial neural network (ANN) as our predictive models. Details in Appendix B.

Explanation methods. We evaluate seven attribution-based explanation methods, including VanillaGrad (Simonyan et al., 2014), Integrated Gradients (Sundararajan et al., 2017), SmoothGrad (Smilkov et al., 2017), Input×Gradients (Shrikumar et al., 2017), LIME (Ribeiro et al., 2016), and SHAP (Lundberg & Lee, 2017). Following Agarwal et al. (2022), we also include a random assignment of importance as a control setting. Details on implementation and hyperparameter selection for the explanation methods are in Appendix B.

Setup. For each dataset and predictor, we: (1) train the prediction model on the respective dataset; (2) randomly sample 100 points from the test set; (3) generate 50 perturbations for each point in the test set; (4) generate explanations $\mathbf{e_{x'}}$ for each test set point and its perturbations using seven explanation methods; and (5) evaluate the stability of the explanations for these test points using all stability metrics (Equations 2,3,5).

4.2 Results

Empirically verifying our theoretical bound. We empirically evaluated our theoretical bounds by computing the LHS of Equation 4 for all seven explanation methods. Results in Figure 2 illustrate

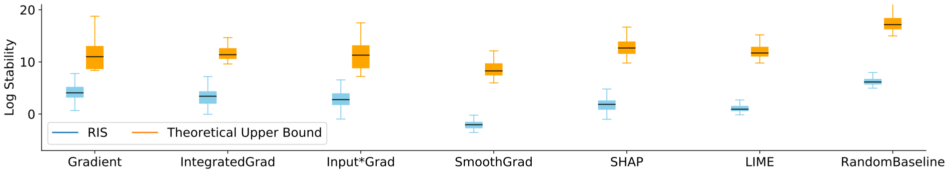
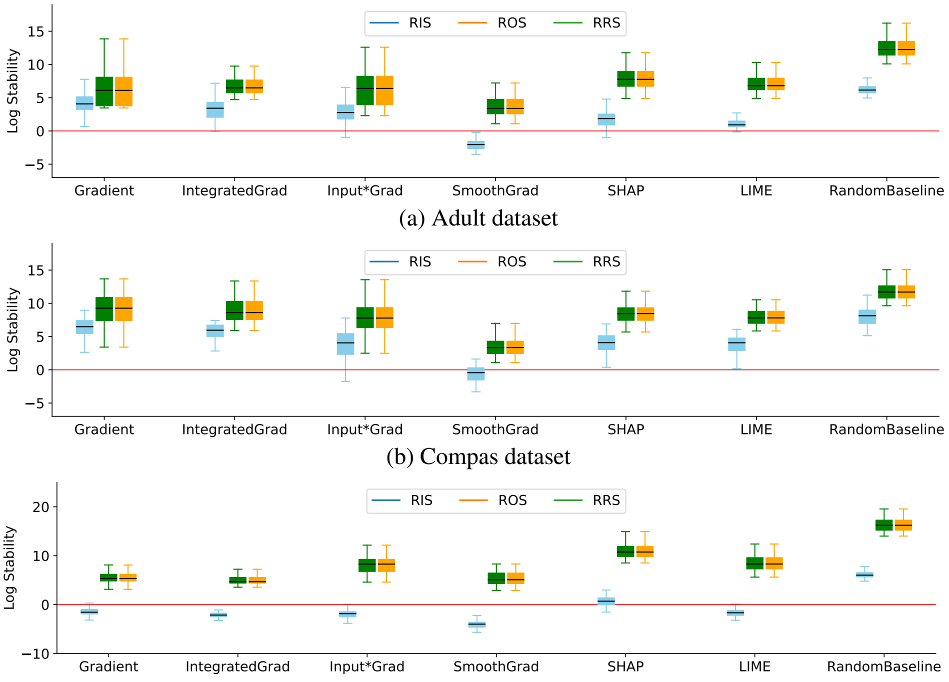


Figure 2: Theoretical upper bounds for the (log) relative input stability (RIS) computed using the right-hand-side of Equation 4 across seven explanation methods for an ANN predictor trained on Adult dataset. Results show that RIS is upper bounded by the product of L_1 and RRS (relative representation stability), where L_1 is the Lipschitz constant between the input and hidden layer of the ANN model. Results for the Compas and German dataset are shown in Appendix 5.



(a) Adult dataset

(b) Compas dataset

(c) German Credit dataset

Figure 3: Empirically calculated log stability of relative stability variants (Equations 2-5) across seven explanation methods. Results on the Adult (a), Compas (b), and German (c) dataset trained with ANN predictor show that SmoothGrad generates the most stable explanation across RRS and ROS variants. Results for all datasets trained on Logistic Regression models are shown in Appendix 4.

the empirical and theoretical bounds for the Relative Input Stability, confirming that none of our theoretical bounds are violated. In addition, we observe that, on average across all explanation methods, our upper bounds are tight with the mean theoretical bounds being 233% higher than that of the empirical values. Similar results are found for other datasets in Appendix 5.

Evaluating the stability of explanation methods. We compare the stability of explanation methods by computing instability using all three variants as described in Section 3.3. Results in Figure 3 show that the median instability of all explanation methods using *Relative Input Stability* (Figure 3; blue) are lower than that for the *Representation* (Figure 3; green) and *Output* Stability (Figure 3; orange) because the relative input stability (Equation 2) scores are highly influenced by the input differences $(\mathbf{x} - \mathbf{x}')$, i.e., the median RIS scores across all explanation methods are always lower than RRS and ROS. Finally, we observe that while no explanation method is completely stable, on average across all datasets and representation stability variants, the SmoothGrad explanation method generates the most stable explanation and outperforms other methods by 12.7%.

5 Conclusion

We introduce *Relative Stability* metrics that measure the change in output explanation with respect to the behavior of the underlying predictive model. To this end, we analyze the stability performance of seven state-of-the-art explanation methods using multiple real-world datasets and predictive models. Our theoretical and empirical analysis demonstrate that representation and output stability indicates that SmoothGrad explanation method generates the most stable explanation. We believe that our work is an important step towards developing a broader set of evaluation metrics that incorporate the behavior of the underlying prediction model.

Acknowledgments

We would like to thank the anonymous reviewers for their insightful feedback. M.Z. is supported, in part by NSF under nos. IIS-2030459 and IIS-2033384, Harvard Data Science Initiative, Amazon Research Award, Bayer Early Excellence in Science Award, AstraZeneca Research, Roche Alliance with Distinguished Scientists (ROADS) Award, Department of the Air Force, and MIT Lincoln National Laboratory. H.L. is supported, in part by the NSF awards IIS-2008461 and IIS-2040989, and research awards from the Harvard Data Science Institute, Amazon, Bayer, and Google. H.L. would like to thank Mohan and Sujatha Lakkaraju, and Pracheer Gupta for all their inputs and support. The views expressed are those of the authors and do not reflect the official policy or position of the funding agencies.

References

Julius Adebayo, Justin Gilmer, Michael Muelly, Ian Goodfellow, Moritz Hardt, and Been Kim. Sanity checks for saliency maps. In *NeurIPS*, 2018.

Chirag Agarwal, Himabindu Lakkaraju, and Marinka Zitnik. Towards a unified framework for fair and stable graph representation learning. In *UAI*, 2021.

Chirag Agarwal, Marinka Zitnik, and Himabindu Lakkaraju. Probing gnn explainers: A rigorous theoretical and empirical analysis of gnn explanation methods. In *AISTATS*, 2022.

David Alvarez-Melis and Tommi S Jaakkola. On the robustness of interpretability methods. *ICML Workshop on Human Interpretability in Machine Learning*, 2018.

David Alvarez-Melis, Harmanpreet Kaur, Hal Daum ée II, Hanna Wallach, and Jennifer Wortman Vaughan. From human explanation to model interpretability: A framework based on weight of evidence. In *HCOMP*, 2021.

Naman Bansal, Chirag Agarwal, and Anh Nguyen. Sam: The sensitivity of attribution methods to hyperparameters. In *CVPR*, 2020.

Ann-Kathrin Dombrowski, Maximilian Alber, Christopher J Anders, Marcel Ackermann, Klaus-Robert Müller, and Pan Kessel. Explanations can be manipulated and geometry is to blame. *arXiv*, 2019.

Dheeru Dua and Casey Graff. UCI machine learning repository, 2017. URL `http://archive.ics.uci.edu/ml`.

Amirata Ghorbani, Abubakar Abid, and James Zou. Interpretation of neural networks is fragile. In *AAAI*, 2019.

Henry Gouk, Eibe Frank, Bernhard Pfahringer, and Michael J Cree. Regularisation of neural networks by enforcing lipschitz continuity. In *Machine Learning*. Springer, 2021.

Scott M Lundberg and Su-In Lee. A unified approach to interpreting model predictions. In *NeurIPS*, 2017.

S Mattu, L Kirchner, and J Angwin. How we analyzed the compas recidivism algorithm. *ProPublica*, 2016.

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. " why should i trust you?" explaining the predictions of any classifier. In *KDD*, 2016.

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. Anchors: High-precision model-agnostic explanations. In *AAAI*, 2018.

Avanti Shrikumar, Peyton Greenside, and Anshul Kundaje. Learning important features through propagating activation differences. In *ICML*, 2017.

Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. In *ICLR*, 2014.

Dylan Slack, Sophie Hilgard, Emily Jia, Sameer Singh, and Himabindu Lakkaraju. How can we fool lime and shap? adversarial attacks on post hoc explanation methods. In *AIES*, 2020.

Dylan Slack, Anna Hilgard, Sameer Singh, and Himabindu Lakkaraju. Reliable post hoc explanations: Modeling uncertainty in explainability. In *NeurIPS*, 2021.

Daniel Smilkov, Nikhil Thorat, Been Kim, Fernanda B. Viégas, and Martin Wattenberg. SmoothGrad: removing noise by adding noise. In *ICML Workshop on Visualization for Deep Learning*, 2017.

Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *ICML*, 2017.

A Theoretical Interpretation

Prior works have shown that commonly used artificial neural network (ANN) models comprise of linear and activation layers which satisfy Lipschitz continuity (Gouk et al., 2021). Let us consider a 2-layer ANN model f, where $h_1(\cdot)$ and $h_2(\cdot)$ represent the outputs of the first and second hidden layers, respectively. For a given input $\mathbf{x}$ and its perturbed counterpart $\mathbf{x}'$, we can write the Lipschitz form for the first hidden layer as:

$$|| \, h_1(\mathbf{x}) - h_1(\mathbf{x}') \, ||_p \leq L_1 \, || \, \mathbf{x} - \mathbf{x}' \, ||_p, \tag{7}$$

where L is the Lipschitz constant of the hidden layer $h_1(\cdot)$. Taking the reciprocal of Equation 7, we get:

$$\frac{1}{|| \, h_1(\mathbf{x}) - h_1(\mathbf{x}') \, ||_p} > \frac{1}{L_1} \, \frac{1}{|| \, \mathbf{x} - \mathbf{x}' \, ||_p}, \tag{8}$$

Multiplying both sides with $|| \, \frac{(\mathbf{e_x} - \mathbf{e_{x'}})}{\mathbf{e_x}} ||_p$, we get:

$$\frac{|| \, \frac{(\mathbf{e_x} - \mathbf{e_{x'}})}{\mathbf{e_x}} ||_p}{|| \, h_1(\mathbf{x}) - h_1(\mathbf{x}') \, ||_p} > \frac{1}{L_1} \, \frac{|| \, \frac{(\mathbf{e_x} - \mathbf{e_{x'}})}{\mathbf{e_x}} ||_p}{|| \, \mathbf{x} - \mathbf{x}' \, ||_p}, \tag{9}$$

With further simplifications, we get:

$$\frac{|| \, \frac{(\mathbf{e_x} - \mathbf{e_{x'}})}{\mathbf{e_x}} ||_p}{||h_1(\mathbf{x})||_p || \frac{h_1(\mathbf{x}) - h_1(\mathbf{x}')}{h_1(\mathbf{x})} ||_p} > \frac{1}{L_1} \, \frac{|| \, \frac{(\mathbf{e_x} - \mathbf{e_{x'}})}{\mathbf{e_x}} ||_p}{||\mathbf{x}||_p || \frac{\mathbf{x} - \mathbf{x}'}{\mathbf{x}} ||_p} \tag{10}$$

For a given $\mathbf{x}'$ and representations from model layer $h_1(\cdot)$, using Equations 2-3, we get:

$$\frac{\text{RRS}(\mathbf{x}, \mathbf{x}', \mathbf{e_x}, \mathbf{e_{x'}})}{||h_1(\mathbf{x})||_p} > \frac{1}{L_1} \, \frac{\text{RIS}(\mathbf{x}, \mathbf{x}', \mathbf{e_x}, \mathbf{e_{x'}})}{||\mathbf{x}||_p}, \tag{11}$$

$$\Rightarrow \text{RIS}(\mathbf{x}, \mathbf{x}', \mathbf{e_x}, \mathbf{e_{x'}}) < \big(L_1 \frac{||h_1(\mathbf{x})||_p}{||\mathbf{x}||_p}\big) \; \times \; \text{RRS}(\mathbf{x}, \mathbf{x}', \mathbf{e_x}, \mathbf{e_{x'}}), \tag{12}$$

where we find that the Relative Input Stability score is upper bounded by L_1 times $\lambda_1 {=} \frac{||h_1(\mathbf{x})||_p}{||\mathbf{x}||_p}$ times the Relative Representation Stability score. Finally, we can also extend the above analysis by substituting $h_1(\cdot)$ with the output logit layer $h_2(\cdot)$ and show that the same relation holds for Relative Output Stability:

$$\text{RRS}(\mathbf{x}, \mathbf{x}', \mathbf{e_x}, \mathbf{e_{x'}}) < \lambda_2 L_2 \; \times \; \text{ROS}(\mathbf{x}, \mathbf{x}', \mathbf{e_x}, \mathbf{e_{x'}}), \tag{13}$$

where $\lambda_2 = ||h_1(\mathbf{x})||_p$.

B IMPLEMENTATION DETAILS

Predictors. We train logistic regression (LR) and artificial neural network (ANN) models. Details in Appendix B. The ANN models have 1 hidden layer of width 100 followed by a ReLU activation function and the output Softmax layer.

Predictor Training. To train all predictive models, we used a 80-10-10 train-test-validation split. We used the RMSProp optimizer with learning rate $2e-03$, the binary cross entropy loss function, and batchsize 32. We trained for 100 epochs and selected the model at the epoch with the highest validation set accuracy as the final prediction model to be explained in our experiments.

Explanation Method Implementations. We used existing public implementations of all explanation methods in our experiments. We used the following `captum` software package classes: i) `captum.attr.Saliency` for VanillaGrad; ii) `captum.attr.IntegratedGradients` for IntegratedGradients; iii) `captum.attr.NoiseTunnel`; iv) `captum.attr.Saliency` for SmoothGrad; v) `captum.attr.InputXGradient` for Gradients×Input; and vi) `captum.attr.KernelShap` for SHAP. We use the authors' LIME python package for LIME.

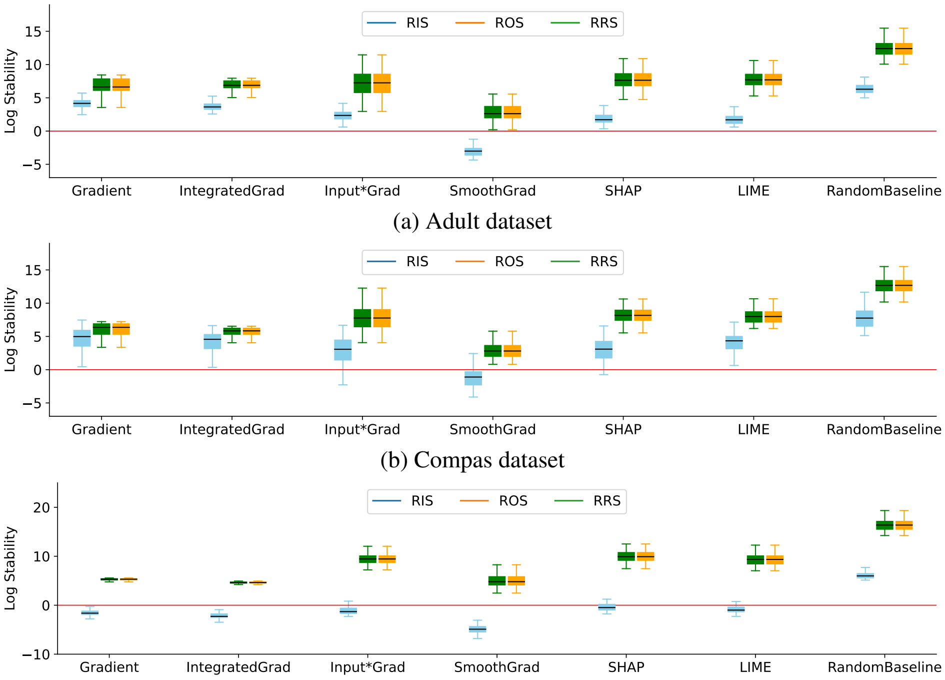
Metric hyperparameters. For all metrics, we generate a neighborhood $\mathcal{N}_{\mathbf{x}}$ of size 50 for each point $\mathbf{x}$. The neighborhood points were generated by perturbing the clean sample $\mathbf{x}$ with noise from $\mathcal{N}(\mathbf{x}, 0.05)$. For data sets with with discrete binary inputs we used independent Bernoulli random variables for the pertubations: for each discrete dimension, we replaced the original values with those that were drawn from a Bernoulli distribution with parameter $p = 0.03$. Choosing a small p ensures that only a small fraction of samples are perturbed to reduce the likelihood of sampling an out-of-distribution point. For internal model representations $\mathcal{L}_{\mathbf{x}}$ we use the pre-softmax input linear layer output embedding for the LR models, and the pre-ReLU output embedding of the first hidden layer for the ANN.

Explanation Method	Hyperparameter	Value

LIME	`n_samples`
	`kernel_width`
	`std`
		1000
		0.75
		0.05

SHAP	`n_samples`	500
SmoothGrad	`std`	0.05
Integrated Gradients	`baseline`	train data means
Random Baseline	attributions from $\mathcal{N}(0, 1)$	

Table 1: Hyperparameters used for explanation methods. For hyperparameters not listed, we used their package defaults.

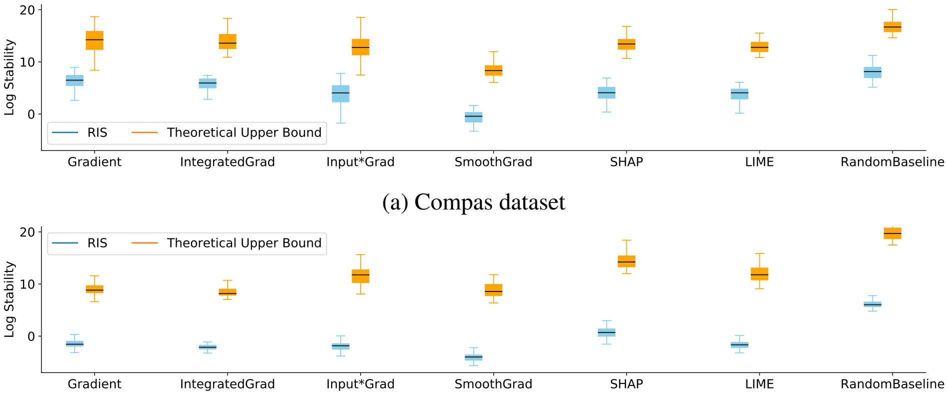


(a) Adult dataset

(b) Compas dataset

(c) German dataset

Figure 4: Empirically calculated log stability of all three relative stability variants (Equations 2-5) across seven explanation methods. Results on the Adult dataset trained with Logistic Regression predictor show that SmoothGrad generates the most stable explanation across representation and output stability variants.



(a) Compas dataset

(b) German credit dataset

Figure 5: Theoretical upper bounds for the (log) relative input stability (RIS) computed using the right-hand-side of Equation 4 across seven explanation methods for an ANN predictor trained on the Compas and German credit datasets. Results show that RIS is upper bounded by the product of L_1 and RRS (relative representation stability), where L_1 is the Lipschitz constant between the input and hidden layer of the ANN model.