Optimizing Explanations:
Nuances Matter When Evaluation Metrics Become Loss Functions

Jonas B. Raedler [1] **Hiwot Belay Tadesse** [1] **Weiwei Pan** [1] **Finale Doshi-Velez** [1]

Abstract

Recent work has introduced a framework that allows users to directly optimize explanations for desired properties and their trade-offs. While powerful in principle, this method repurposes evaluation metrics as loss functions – an approach whose implications are not yet well understood. In this paper, we study how different robustness metrics influence the outcome of explanation optimization, holding faithfulness constant. We do this in the transductive setting, in which all points are available in advance. Contrary to our expectations, we observe that the choice of robustness metric can lead to highly divergent explanations, particularly in higher-dimensional settings. We trace this behavior to the use of metrics that evaluate the explanation set as a whole, rather than imposing constraints on individual points, and to how these "global" metrics interact with other optimization objectives. These interactions can allow the optimizer to produce locally inconsistent, unintuitive, and even undesirable explanations, despite satisfying the desired trade-offs. Our findings highlight the need for metrics whose mathematical structure more closely aligns with their intended use in optimization, and we advocate for future work that rigorously investigates metrics that incorporate a pointwise evaluation and their influence on the optimization landscape.

1. Introduction

To assess the quality of explanations, the interpretable machine learning community relies on properties such as *faithfulness* (how accurately the explanation reflects the model's behavior) and *robustness* (how stable the explanation remains under input perturbations). Recent work by Tadesse et al. (2024) proposes a framework that enables users to tailor feature attribution explanations to their needs by directly optimizing explanations for desired levels of faithfulness and robustness. Their work addresses a limitation of standard explanation algorithms such as LIME (Ribeiro et al., 2016) and SmoothGrad (Smilkov et al., 2017) which implicitly encode fixed trade-offs (LIME favoring faithfulness, SmoothGrad robustness).

However, this framework still requires users to choose specific faithfulness and robustness metrics from an abundance of mathematical formalizations for these properties (for a collection, see Chen et al. (2024)'s work), making this a complex and underexplored task. To examine how significantly the choice of optimization metrics affects outcomes in practice, we study the influence of different robustness metrics on the resulting explanations when used as optimization objectives (in the transductive setting). We did this for varying function types, data dimensionalities, and perturbation regions.

We initially conjectured that the choice of robustness metric would not matter significantly – that as long as the explanations were fixed to a specific faithfulness loss, they would be consistent across different robustness metrics with little variance. Contrary to our initial expectations, we found that optimizing for desired properties can lead to counterintuitive and undesirable behaviors, especially in higher-dimensional settings. We hypothesize that this behavior arises from the use of metrics that evaluate the explanation set as a whole, rather than at the level of individual points, and from how such "global" metrics interact with other objectives in the optimization. When using such metrics in an optimization setting, the optimizer can exploit this flexibility of only considering the entire set as a whole and, e.g., focus on faithfulness in some regions of the function and robustness in others. This can result in explanations that appear faithful (according to the specified "global" metric), but that are highly imbalanced locally.

[*]Equal contribution [1]Harvard University. Correspondence to: Jonas B. Raedler <jraedler at g.harvard.edu>.

Accepted at Methods and Opportunities at Small Scale (MOSS), ICML 2025, Vancouver, Canada.

2. Related Works

Previous work has repeatedly shown that humans prefer different explanation properties for different tasks (Zhou et al., 2021; Liao et al., 2022; Nofshin et al., 2024), motivating the need for explanation methods that allow users to specify which properties are most important. However, enabling such customization is challenging because explanation properties are often in tension. Previous work, for example, has documented trade-offs between faithfulness and robustness (Bansal et al., 2020), faithfulness and complexity (Bhatt et al., 2020b), and faithfulness and homogeneity (Balagopalan et al., 2022). To give concrete examples of this tension, several works have found that LIME (Ribeiro et al., 2016), a method that encourages faithfulness, often lacks robustness (Alvarez-Melis & Jaakkola, 2018b; Ghorbani et al., 2018; Slack et al., 2020). Conversely, methods like SmoothGrad (Smilkov et al., 2017) and GradCAM (Selvaraju et al., 2019), which prioritize robustness in their explanations, often exhibit lower faithfulness (Adebayo et al., 2020; Tan & Tian, 2023).

An additional issue that emerges in the need for explanations with specific properties is that most existing methods do not explicitly optimize for explanation quality with respect to a certain property. Decker et al. (2024) have proposed to linearly aggregate explanations from multiple methods to obtain explanations with more optimal properties, but the optimality of the resulting explanations is limited by the linear span of the initial set of explanations. Similarly, Wang et al. (2024) proposed a framework in which explanations are directly optimized for multiple properties simultaneously using a genetic algorithm. However, because the explanations are generated through a stochastic process, the method is ineffective at targeting a specific balance of properties.

More recently, Tadesse et al. (2024) introduced a framework for directly optimizing explanations with respect to user-specified property metrics and trade-offs between them, representing the first method to formalize this kind of control.

Since there exists a wide range of available property metrics (Chen et al., 2024) that users can (and have to) choose from to use this framework – often with subtle yet important differences in how they evaluate explanations – we identify a lot of value in determining how the different formalizations influence the explanations resulting from this optimization framework.

3. Methods

3.1. Experimental Setup

We use a simple toy example as the basis for our experiment. Specifically, we consider a model f that maps a multidimensional input $x \in \mathbb{R}^D$ to an output $f(x) \in \mathbb{R}$. We then generate feature attribution explanations, which approximate the contribution of each input feature (i.e., the different dimensions of x) to the output of the model, $f(x)$. In other words, we are estimating the gradient of f with respect to x.

In our experiment, we focus on the transductive setting in which all input points that explanations are generated for are known in advance. To evaluate the consistency of our results, we compare explanations across five different functions (see Table 1) that vary in steepness and periodicity, allowing us to assess whether observed behaviors generalize or whether behavior differs for certain families of functions.

Name	Function
x^2	$f(x) = \sum_{d=1}^{D} x_d^2$
x^3	$f(x) = \sum_{d=1}^{D} 0.35x_d^3$
$sin\ x + e^x - x^2$	$f(x) = \sum_{d=1}^{D} \sin 2x_d + 0.1e^{x_d} - 0.5x_d^2$
$sin\ x$	$f(x) = \sum_{d=1}^{D} \sin x_d$
$sin\ x^3$	$f(x) = \sum_{d=1}^{D} \sin 0.35x_d^3$

Table 1: Function Classes with Corresponding Functions and Dimensions. We introduced the coefficients to ensure that the specified fixed faithfulness loss mentioned in section 3 was achievable. Note that the name just states the general function type and does not include the coefficients.

Our experiments use input points sampled from the range $[-5, 5]^D$, with $D \in \{1, 2, 3, 4\}$; higher dimensions were excluded due to computational constraints. We also vary the perturbation radius u, a hyperparameter that all three robustness metrics depend on. We consider the values $u \in \{1, 2, 3, 5, 10\}$, subject to the constraint that the minimum Euclidean pairwise distance between input points exceeds u. For further implementation details, see Appendix A.

3.2. Comparing the Influence of Robustness Metrics

Choice of Metrics. To analyze the impact of different property metrics on explanations, we consider one faithfulness and three robustness metrics and compare pairing the faithfulness metric with each of the robustness metrics as loss functions in our direct optimization. The Faithfulness metric as a loss minimizes the distance between the explanation to the original function. Using Max-Sensitivity as a robustness loss function minimizes the maximum difference between explanations across nearby points within a perturbation region u. Local-Stability minimizes the maximum rate of change in explanations with respect to input perturbations within a perturbation region u. Average-Sensitivity minimizes the average change in explanations within a perturbation region u. Table 2 provides the formalizations for our metrics.

Metric	Formalization
Faithfulness	$\mathcal{L}_{\text{faithful}}(\mathbf{E}) = \sum_{n=1}^{N} \|\nabla \mathbf{f_n} - \mathbf{E_n}\|_2^2$
Max-Sensitivity (Yeh et al., 2022)	$\mathcal{L}_{\text{max\_sensitivity}}(\mathbf{E}) = \sum_{n=1}^{N} \max_{\substack{n' \in N s.t. \\ \|X_n - X_{n'}\| \leq u}} \|\mathbf{E_n} - \mathbf{E_{n'}}\|$
Local-Stability (Alvarez-Melis & Jaakkola, 2018a)	$\mathcal{L}_{\text{local\_stability}}(\mathbf{E}) = \sum_{n=1}^{N} \max_{\substack{n' \in N s.t. \\ \|X_n - X_{n'}\| \leq u}} \frac{\|\mathbf{E_n} - \mathbf{E_{n'}}\|}{\|X_n - X_{n'}\|}$
Average-Sensitivity (Bhatt et al., 2020a)	$\mathcal{L}_{\text{average\_sensitivity}}(\mathbf{E}) = \sum_{n=1}^{N} \sum_{n'=1}^{N} \|\mathbf{E_n} - \mathbf{E_{n'}}\|^2 \cdot p_n(n')$

Table 2: Property metrics used in this work. In all formalizations, E are the explanations, N are the total number of considered points, and X_n refers to the n^{th} input point (X is sorted). f is the underlying model that is being explained. For average-sensitivity, p_n is the uniform probability distribution $p_n = U(\ \{n' \mid \|X_n - X_{n'}\| \leq u\}\ )$.

Isolating the Effects of the Robustness Metrics. We want to determine how the choice of robustness metric in our considered optimization framework affects the explanations of a model f. Therefore, to help isolate this effect as much as possible, we not only keep the considered faithfulness metric constant, but we also fix the total loss $\mathcal{L}_{\text{faithful}}$ for every set of explanations. This ensures that the explanations optimized for different robustness metrics all have the same faithfulness loss, enabling a more direct comparison. We achieve this by first determining the range of possible $\mathcal{L}_{\text{faithful}}$ values for a given model f. This range always starts at $\mathcal{L}_{\text{faithful}}^{\text{min}} = 0$ (with the explanation $E = \nabla f$) and ends at $\mathcal{L}_{\text{faithful}}^{\text{max}}$, which comes from having a perfectly robust explanation (i.e., the average value of the gradient over all inputs). To remain consistent across all our considered robustness metrics, we need to fix $\mathcal{L}_{\text{faithful}}$ to be a specific value, so we chose it to be $\eta = 0.2 \cdot \mathcal{L}_{\text{faithful}}^{\text{max}}$. We then use $\mathcal{L}_{\text{faithful}} \leq \eta$ as a constraint in the optimization problem. For a given robustness metric, we solve:

$$E_{\text{robust}}^{*} = arg \min_{E} \mathcal{L}_{\text{robust}}(E)$$

$$s.t. \quad \mathcal{L}_{\text{faithful}}(E) \leq \eta$$

Since the optimization aims to minimize the robustness loss, and since robustness and faithfulness operate in a natural trade-off, this results in all explanations having $\mathcal{L}_{\text{faithful}} \approx \eta$.

We solve this problem using the Python package *cvxpy* (Diamond & Boyd, 2016; Agrawal et al., 2018) with the MOSEK solver (ApS, 2025). See Appendix C for our derivations on how we turn the metrics into convex second-order cone programs, which enables us to use the *cvxpy* package to solve for the optimal explanation.

Comparing the Effects of Robustness Metrics via Agreement Rates. To quantify each robustness metric's impact on the resulting explanations, we compute the agreement rate over all considered input points: for each point, we record whether the feature deemed most important is the same across all explanations. Higher agreement rates indicate more similar explanations. Note that agreement is only defined for input dimensions $D \geq 2$ (the agreement rate for $D = 1$ is trivially 1).

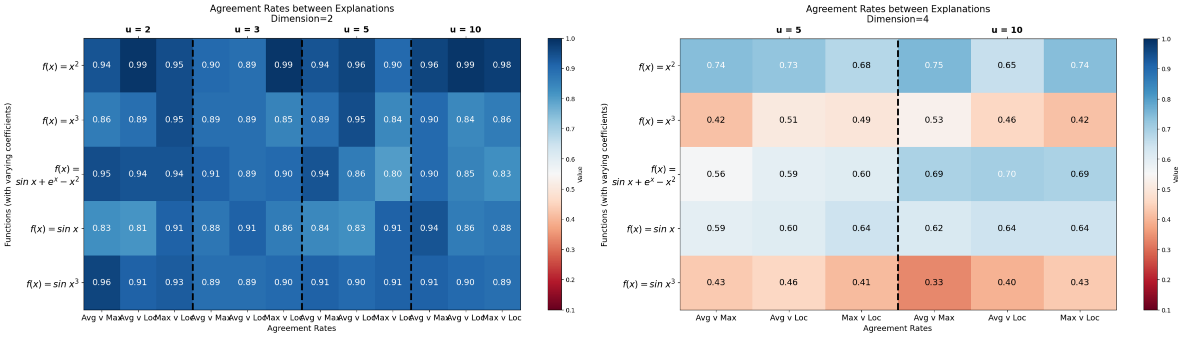

Figure 1: Agreement rates between explanations optimized for various robustness objectives, across increasing input dimension (we present $D = 2$ and $D = 4$ here. For the agreement rates table for $D = 3$, refer to Appendix B) and across all mentioned functions in Table 1 (listed in Appendix A).

4. Results and Analysis

We performed the experiment as described and find that different robustness metrics result in divergent explanations, even when constrained to equal overall faithfulness. This seems to be particularly true in higher dimensions and for functions with steep or oscillatory gradients.

This is shown by the agreement rates we obtain from the explanations that have been optimized under different robustness metrics (see Fig. 1): as the dimensionality of the input data increases, the agreement rates between different explanations steadily decrease – sinking from relatively high rates ($\gtrsim 0.8$ in $D = 2$) to rates consistently below 0.5 in $D = 4$ for functions with highly oscillatory ($sin\ x^3$) and steeper (x^3) gradients.

This divergence in explanations is particularly striking given the simplicity of our experimental setup: picking the most important feature out of only four candidates. Real-world models often involve dozens or even hundreds of input features; if agreement among just four drops below 50%, discrepancies in higher dimensions could be much worse.

Why do explanations optimized for different robustness metrics diverge so dramatically, even when they have the same faithfulness loss? We find this empirical result to be counterintuitive, so we investigate further:

Let $\Omega \subset \mathbb{R}^d$ be our input domain and let $f : \Omega \to \mathbb{R}$ be a function with gradient $\nabla f(x) \in \mathbb{R}^d$. Suppose that an explanation $E : \Omega \to \mathbb{R}^d$ approximates ∇f. We have a faithfulness loss:

$$\mathcal{L}_{\text{faithful}}(E, \nabla f) = \frac{1}{|\Omega|} \int_{\Omega} \|E(x) - \nabla f(x)\|_2^2 \, dx$$

We want the faithfulness loss to be bounded, so we require $\mathcal{L}_{\text{faithful}}(E, \nabla f) \leq \eta$ for some $\eta > 0$. This, however, constrains E only in an average sense: as long as the overall average faithfulness loss remains $\leq \eta$, explanations may deviate substantially from ∇f; these deviations just have to "cancel out" elsewhere. An example of this phenomenon can be seen in Fig. 2, where we consider x^3 with $D = 1$ and perturbation region $u = 1$. We can clearly see how the optimization under average-sensitivity, with the constraint $\mathcal{L}_{\text{faithful}}(E, \nabla f) \leq \eta$, results in explanations that represent the original model very faithfully in certain regions (-4 to -2, as well as 2 to 4), but that also has completely flat (and robust) parts elsewhere. On average, the robust regions cancel the faithful ones out, such that the robustness metric is minimized while still satisfying the $\mathcal{L}_{\text{faithful}}(E, \nabla f) \leq \eta$ constraint.

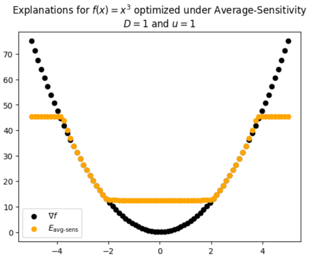

Figure 2: A depiction of the "faithful in some areas, robust in others"-phenomenon. The optimization with a global faithfulness constraint can result in unbalanced explanations.

Because the faithfulness constraint is global (i.e. acting on E as a whole, not on individual explanation points), it does not prevent a robustness-driven optimizer from, e.g., concentrating faithfulness in certain regions (where ∇f is large or highly variable) while smoothing or flattening explanations elsewhere. The result can be "uneven" (and undesirable) explanations that are very faithful in some areas and highly robust in others. In geometric terms, this can be seen as the set of admissible explanations E lying within an L^2 "ball" of radius $\sqrt{\eta}$, whose center is the concatenation of explanations that have $\mathcal{L}_{\text{faithful}} = 0$ (i.e., ∇f_n for all input points N).

This counterintuitive phenomenon highlights the importance of carefully investigating and developing an understanding of how evaluation metrics influence the optimization before using them as loss functions.

Crucially, the number of degrees of freedom available to deviate from ∇f grows with D. In low dimensions, there are relatively few directions in which the optimizer can "shift" the explanation to minimize robustness while preserving global faithfulness. As D increases, however, the optimizer can exploit a much larger space of directions, leading to very different optima for different robustness metrics. We believe that this is why agreement rates decrease so significantly as D grows.

Conclusion

In this work, we demonstrate that – while intuitive – it is not advisable to naively repurpose evaluation metrics as loss functions within Tadesse et al. (2024)'s optimization framework in the transductive setting. This appears to be particularly true when the metric – used as a loss function – operates over a set of explanations as a whole, rather than at the level of individual points, as there are otherwise no guarantees about the properties of individual explanations - just about the entire set. This can lead to inconsistent, unintuitive and even undesirable explanations.

To better address the challenge of generating explanations that fulfill user-specified properties in the transductive setting, we need metrics with mathematic formalizations that – when used as loss functions – work in the way we intend and expect. To get us closer to that goal, we identify the need to rigorously explore and evaluate how the optimization is affected by metrics that consider individual points. We believe that such metrics will result in more stable and consistent explanations, as they introduce restrictions on individual explanations and limit the optimizer's excessive flexibility to find weird minima.

References

Adebayo, J., Gilmer, J., Muelly, M., Goodfellow, I., Hardt, M., and Kim, B. Sanity checks for saliency maps, 2020. URL https://arxiv.org/abs/1810.03292.

Agrawal, A., Verschueren, R., Diamond, S., and Boyd, S. A rewriting system for convex optimization problems. *Journal of Control and Decision*, 5(1):42–60, 2018.

Alvarez-Melis, D. and Jaakkola, T. S. Towards robust interpretability with self-explaining neural networks. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, NIPS'18, pp. 7786–7795, Red Hook, NY, USA, 2018a. Curran Associates Inc.

Alvarez-Melis, D. and Jaakkola, T. S. On the robustness of interpretability methods, 2018b. URL https://arxiv.org/abs/1806.08049.

ApS, M. *MOSEK Optimizer API for Python 11.0.20*, 2025. URL https://docs.mosek.com/latest/pythonapi/index.html.

Balagopalan, A., Zhang, H., Hamidieh, K., Hartvigsen, T., Rudzicz, F., and Ghassemi, M. The road to explainability is paved with bias: Measuring the fairness of explanations. In *Proceedings of the 2022 ACM Conference on Fairness, Accountability, and Transparency*, FAccT '22, pp. 1194–1206, New York, NY, USA, 2022. Association for Computing Machinery. ISBN 9781450393522. doi: 10.1145/3531146.3533179. URL https://doi.org/10.1145/3531146.3533179.

Bansal, N., Agarwal, C., and Nguyen, A. Sam: The sensitivity of attribution methods to hyperparameters, 2020. URL https://arxiv.org/abs/2003.08754.

Bhatt, U., Weller, A., and Moura, J. M. F. Evaluating and aggregating feature-based model explanations. In Bessiere, C. (ed.), *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20*, pp. 3016–3022. International Joint Conferences on Artificial Intelligence Organization, 7 2020a. doi: 10.24963/ijcai.2020/417. URL https://doi.org/10.24963/ijcai.2020/417. Main track.

Bhatt, U., Weller, A., and Moura, J. M. F. Evaluating and aggregating feature-based model explanations, 2020b. URL https://arxiv.org/abs/2005.00631.

Chen, Z., Subhash, V., Havasi, M., Pan, W., and Doshi-Velez, F. What makes a good explanation?: A harmonized view of properties of explanations, 2024. URL https://arxiv.org/abs/2211.05667.

Decker, T., Bhattarai, A. R., Gu, J., Tresp, V., and Buettner, F. Provably better explanations with optimized aggregation of feature attributions, 2024. URL https://arxiv.org/abs/2406.05090.

Diamond, S. and Boyd, S. CVXPY: A Python-embedded modeling language for convex optimization. *Journal of Machine Learning Research*, 17(83):1–5, 2016.

Ghorbani, A., Abid, A., and Zou, J. Interpretation of neural networks is fragile, 2018. URL https://arxiv.org/abs/1710.10547.

Liao, Q. V., Zhang, Y., Luss, R., Doshi-Velez, F., and Dhurandhar, A. Connecting algorithmic research and usage contexts: A perspective of contextualized evaluation for explainable ai, 2022. URL https://arxiv.org/abs/2206.10847.

Nofshin, E., Brown, E., Lim, B., Pan, W., and Doshi-Velez, F. A sim2real approach for identifying task-relevant properties in interpretable machine learning, 2024. URL https://arxiv.org/abs/2406.00116.

Ribeiro, M. T., Singh, S., and Guestrin, C. ”why should i trust you?”: Explaining the predictions of any classifier, 2016. URL https://arxiv.org/abs/1602.04938.

Selvaraju, R. R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., and Batra, D. Grad-cam: Visual explanations from deep networks via gradient-based localization. *International Journal of Computer Vision*, 128(2):336–359, October 2019. ISSN 1573-1405. doi: 10.1007/s11263-019-01228-7. URL http://dx.doi.org/10.1007/s11263-019-01228-7.

Slack, D., Hilgard, S., Jia, E., Singh, S., and Lakkaraju, H. Fooling lime and shap: Adversarial attacks on post hoc explanation methods, 2020. URL https://arxiv.org/abs/1911.02508.

Smilkov, D., Thorat, N., Kim, B., Viégas, F., and Wattenberg, M. Smoothgrad: removing noise by adding noise, 2017. URL https://arxiv.org/abs/1706.03825.

Tadesse, H. B., Hüyük, A., Pan, W., and Doshi-Velez, F. Directly optimizing explanations for desired properties, 2024. URL https://arxiv.org/abs/2410.23880.

Tan, Z. and Tian, Y. Robust explanation for free or at the cost of faithfulness. In Krause, A., Brunskill, E., Cho, K., Engelhardt, B., Sabato, S., and Scarlett, J. (eds.), *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 33534–33562. PMLR, 23–29 Jul 2023. URL https://proceedings.mlr.press/v202/tan23a.html.

Wang, Z., Huang, C., Li, Y., and Yao, X. Multi-objective feature attribution explanation for explainable machine learning. *ACM Transactions on Evolutionary Learning and Optimization*, 4(1):1–32, 2024.

Yeh, C.-K., Kim, B., Arik, S. O., Li, C.-L., Pfister, T., and Ravikumar, P. On completeness-aware concept-based explanations in deep neural networks, 2022. URL https://arxiv.org/abs/1910.07969.

Zhou, J., Gandomi, A. H., Chen, F., and Holzinger, A. Evaluating the quality of machine learning explanations: A survey on methods and metrics. *Electronics*, 10(5), 2021. ISSN 2079-9292. doi: 10.3390/electronics10050593. URL https://www.mdpi.com/2079-9292/10/5/593.

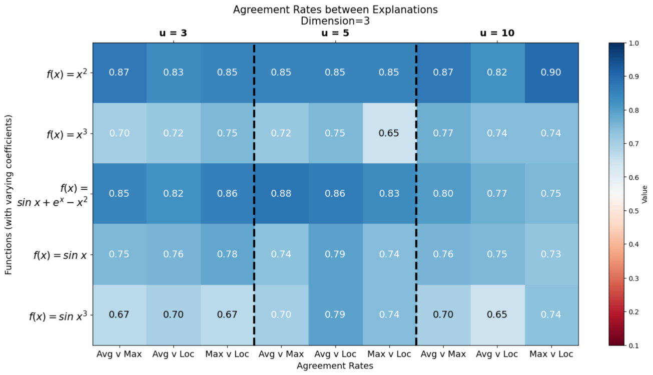

Figure 3: Agreement rates for $D = 3$ between explanations optimized for various robustness objectives across all mentioned functions in Table 1.

A. Experimental Setup

This section includes more detail on our exact methodology.

Number of Input Points Considered. We did not use the same amount of input samples in each dimension, as we wanted equidistant samples across the entire input range to approximate the entire space as faithfully as possible. Computational constraints forced us to adaptively choose the maximum number of points N that still allowed for equidistant sampling. This resulted in $N = 80$ for $D = 1$, $N = 81$ for $D = 2$, $N = 125$ for $D = 3$, and $N = 81$ for $D = 4$.

Perturbation Regions Considered. We consider perturbation regions $u \in \{1, 2, 3, 5, 10\}$ in our experiments, subject to the constraint that the minimum (euclidean) pairwise distance between input points is larger than u. This means that $u = 1$ is only used for $D = 1$, $u = 2$ for $D \leq 2$ and $u = 3$ for $D \leq 3$).

B. Agreement Rates for Dimension 3

In addition to the agreement rate tables for dimension 2 and 4 in Figure 1, we also provide the agreement rate table for dimension 3, which depicts the gradual decrease in agreement as dimensionality increases (see Figure 3).

C. Derivations

This section provides our derivations for how we turned the three metrics – max-sensitivity, local-stability, and average-sensitivity – into convex second-order cone programs, which can be solved via the MOSEK solver in cvxpy.

C.1. Derivation of Max-Sensitivity as a Convex Second-Order Cone Program

This is the original equation:

$$\mathcal{L}_{\text{max\_sensitivity}}(\mathbf{E}) = \sum_{n=1}^{N} \max_{\substack{n' \in N s.t. \\ \|X_n - X_{n'}\| \leq u}} \|\mathbf{E_n} - \mathbf{E_{n'}}\| \tag{1}$$

We want to form $\mathcal{L}_{\text{max\_sensitivity}}$ into an optimization problem that we can feed into a solver. Our overarching goal is to find $E_{opt} = \text{arg} \min_{E} \mathcal{L}_{\text{max}}(E)$.

First, we need to figure out how to obtain the max in a way that allows us to use an optimization problem with constraints. For a fixed E_n we want to find the explanation $E_{n'}$ — under the restriction that $\|X_n - X_{n'}\| \leq u$ — that will result in the highest possible L2-norm between the two explanations (i.e. $\max_{\substack{n' \in N s.t. \\ \|X_n - X_{n'}\| \leq r}} \|\mathbf{E_n} - \mathbf{E_{n'}}\|$).

Let's create the variable l_n that denotes this highest possible squared L2-norm for E_n. Put formally:

$$l_n = \max_{\substack{n' \in N s.t. \\ \|X_n - X_{n'}\| \leq u}} \|\mathbf{E_n} - \mathbf{E_{n'}}\| \tag{2}$$

In order to form this into an optimization problem that we can feed into a solver, we want to get rid of the max statement. However, we still want to ensure that l_n really does represent that max value. We can do that by putting some constraints on l_n: it needs to be at least equal the value of every possible squared L2-norm between the fixed E_n and all other $E_{n'}$ (given the restriction $\|X_n - X_{n'}\| \leq u$). Thus, we will have a total of y constraints for l_n, where $1 \leq y \leq N$.

With the assumption that $\|X_n - X_{n'}\| \leq u$ holds true for all n, n' pairs, we would get the constraints:

$$\begin{aligned} l_n &\geq \|E_n - E_1\| \\ l_n &\geq \|E_n - E_2\| \\ &\vdots \\ l_n &\geq \|E_n - E_n\| \end{aligned}$$

These constraints establish the correct lower bound for l_n, i.e.

$$l_n \geq \max_{\substack{n' \in N s.t. \\ \|X_n - X_{n'}\| \leq u}} \|\mathbf{E_n} - \mathbf{E_{n'}}\| \tag{3}$$

From this follows:

$$\sum_{n=1}^{N} l_n \;\geq\; \sum_{n=1}^{N} \max_{\substack{n' \in N s.t. \\ \|X_n - X_{n'}\| \leq u}} \|\mathbf{E_n} - \mathbf{E_{n'}}\| \tag{4}$$

$$\sum_{n=1}^{N} l_n \;\geq\; \mathcal{L}_{\text{max\_sensitivity}}(\mathbf{E}) \tag{5}$$

Since we want $\sum_{n=1}^{N} l_n$ to be *equal* to $\mathcal{L}_{\text{max\_sensitivity}}(\mathbf{E})$, we have to make sure that we also define a rigorous upper bound for each l_n. We can indirectly achieve this by simply minimizing this sum with respect to all the l_n values. Thus, we get:

$$\arg\min_{l_n} \sum_{n=1}^{N} l_n \;=\; \mathcal{L}_{\text{max\_sensitivity}}(\mathbf{E}) \tag{6}$$

Now that we have rephrased $\mathcal{L}_{\text{max\_sensitivity}}(\mathbf{E})$ in a way that no longer involves the max expression over multiple iterations, we can now minimize $\mathcal{L}_{\text{max\_sensitivity}}(\mathbf{E})$ with respect to E in a way that allows us to use a solver! We simply do:

$$\arg\min_{E} \mathcal{L}_{\text{max\_sensitivity}}(\mathbf{E}) \;=\; \arg\min_{l_n, E} \sum_{n=1}^{N} l_n \tag{7}$$

This will give us the explanations E_{opt} that minimize $\mathcal{L}_{\text{max\_sensitivity}}$.

B. Derivation of Local-Stability as a as a Convex Second-Order Cone Program

This is the original equation:

$$\mathcal{L}_{\text{local\_stability}}(\mathbf{W_E}) = \sum_{n=1}^{N} \max_{\substack{n' \in N s.t. \\ \|X_n - X_{n'}\| \leq u}} \frac{\|\mathbf{E_n} - \mathbf{E_{n'}}\|}{\|X_n - X_{n'}\|} \tag{8}$$

Given that Max-Sensitivity and Local-Stability are extremely similar to each other (l_n and all its constraints are just divided by $\|X_n - X_{n'}\|$), we can simply follow the proof for max-sensitivity and adjust it slightly with this change. Everything else stays the same.

C. Derivation of Average-Sensitivity as a Convex Second-Order Cone Program

This is the original equation:

$$\mathcal{L}_{\text{average\_sensitivity}}(\mathbf{E}) = \sum_{n=1}^{N} \sum_{n'=1}^{N} \|\mathbf{E_n} - \mathbf{E_{n'}}\| \cdot p_n(n') \tag{9}$$

where p_n is the uniform probability distribution p_n = $U(\ \{n' \mid \|X_n - X_{n'}\| \leq u\}\ )$.

The formation of this robustness metric into an optimization problem is a lot less complex: we want to minimize $\mathcal{L}_{\text{average\_sensitivity}}(\mathbf{E})$ with respect to E. This is the same as finding E that minimizes the sum of $\|\mathbf{E_n} - \mathbf{E_{n'}}\| \cdot p_n(n')$ for all possible combinations of n, n'. Thus, we have no constraints.