Robust Counterfactual Explanations for Neural Networks With Probabilistic Guarantees

Faisal Hamman [1] **Erfaun Noorani** [1] **Saumitra Mishra** [2] **Daniele Magazzeni** [2] **Sanghamitra Dutta** [1]

Abstract

There is an emerging interest in generating robust counterfactual explanations that would remain valid if the model is updated or changed even slightly. Towards finding robust counterfactuals, existing literature often assumes that the original model m and the new model M are bounded in the parameter space, i.e., $\|\text{Params}(M) - \text{Params}(m)\| < \Delta$. However, models can often change significantly in the parameter space with little to no change in their predictions or accuracy on the given dataset. In this work, we introduce a mathematical abstraction termed *naturally-occurring* model change, which allows for arbitrary changes in the parameter space such that the change in predictions on points that lie on the data manifold is limited. Next, we propose a measure – that we call *Stability* – to quantify the robustness of counterfactuals to potential model changes for differentiable models, e.g., neural networks. Our main contribution is to show that counterfactuals with sufficiently high value of *Stability* as defined by our measure will remain valid after potential "naturally-occurring" model changes with high probability (leveraging concentration bounds for Lipschitz function of independent Gaussians). Since our quantification depends on the local Lipschitz constant around a data point which is not always available, we also examine practical relaxations of our proposed measure and demonstrate experimentally how they can be incorporated to find robust counterfactuals for neural networks that are close, realistic, and remain valid after potential model changes.

[1]Department of Electrical and Computer Engineering, University of Maryland, College Park. [2]JP Morgan Chase AI Research. Correspondence to: Faisal Hamman <fhamman@umd.edu>.

Proceedings of the 40th International Conference on Machine Learning, Honolulu, Hawaii, USA. PMLR 202, 2023. Copyright 2023 by the author(s).

1. Introduction

Counterfactual explanations (Wachter et al., 2017; Karimi et al., 2020; Barocas et al., 2020) have garnered significant interest in various high-stakes applications, such as lending, hiring, etc. Counterfactual explanations aim to guide an applicant on how they can change a model outcome by providing suggestions for improvement. Given an original data-point (e.g., an applicant who is denied a loan), the goal is to try to find a point on the other (desired) side of the decision boundary (a hypothetical applicant who is approved for the loan) which also satisfies several other preferred constraints, such as, (i) proximity to the original point; (ii) changes in as few features as possible; and (iii) conforming to the data manifold. Such a data-point that alters the model decision is widely referred to as a "counterfactual."

However, in several real-world scenarios, such as credit lending, the models making these high-stakes decisions have to be updated due to various reasons (Upadhyay et al., 2021; Black et al., 2021), e.g., to retrain on a few additional data points, change the hyper-parameters or seed, or transition to a different model class (Pawelczyk et al., 2020b). Such model changes can often cause the counterfactuals to become invalid because typically they are quite close to the original data point, and hence, also quite close to the decision boundary. For instance, suppose the counterfactual explanation suggests an applicant to increase their income by $10K$ to get approved for a loan and they actually act upon that, but now, due to updates to the original model, they are still denied by the updated model.

If counterfactuals become invalid due to model updates, this can lead to confusion and distrust in the use of algorithms in high-stakes applications altogether. Users would typically act on the suggested counterfactuals over a period of time, e.g., increase their income for credit lending, but only to find that it is no longer enough since the model has slightly changed (perhaps due to retraining with a new seed or hyperparameter). This cycle of invalidation and regenerating new counterfactuals can not only be frustrating and time-consuming for users but also potentially hurts an institution's reputation. This motivates our primary question: *How do we provide theoretical guarantees on the robustness of counterfactuals to potential model changes?*

Towards addressing this question, in this work, we introduce the abstraction of "naturally-occurring" model change for differentiable models. Our abstraction allows for arbitrary changes in the parameter space such that the change in predictions on points that lie on the data manifold is limited. This abstraction motivates a measure of robustness for counterfactuals that arrives with provable probabilistic guarantees on their validity under naturally-occurring model change. We also introduce the notion of "targeted" model change and provide an impossibility result for such model change. Next, by leveraging a *computable relaxation* of our proposed measure of robustness, we then design and implement algorithms to find robust counterfactuals for neural networks. Our experimental results validate our theoretical understanding and illustrate the efficacy of our proposed algorithms. We summarize our contributions here:

- **Abstraction of "naturally-occurring" model change for differentiable models:** Existing literature (Upadhyay et al., 2021; Black et al., 2021) on robust counterfactuals often assumes that the original model m and the new model M are bounded in the parameter space, i.e., $\|\text{Params}(M) - \text{Params}(m)\| < \Delta$. Building on Dutta et al. (2022) for tree-based models, we note that models can often change significantly in the parameter space with little to no change on their predictions or accuracy on the given dataset. To capture this, we introduce an abstraction (see Definition 5), that we call *naturally-occurring* model change, which instead allows for arbitrary changes in the parameter space such that the change in predictions on points that lie on the data manifold is limited.
- **A measure of robustness with probabilistic guarantees on validity:** Next, we propose a novel mathematical measure – that we call Stability – to quantify the robustness of counterfactuals to potential model changes for differentiable models. Stability of a counterfactual $x \in \mathbb{R}^d$ with respect to a model $m(\cdot)$ is given by:

$$R_{k,\sigma^2}(x, m) = \frac{1}{k} \sum_{x_i \in N_{x,k}} \left(m(x_i) - \gamma_x \|x - x_i\| \right),$$

  where $N_{x,k}$ is a set of k points in $\mathbb{R}^d$ drawn from the Gaussian distribution $\mathcal{N}(x, \sigma^2 \mathrm{I}_d)$ with I_d being the identity matrix, and γ_x is the local Lipschitz constant of the model $m(\cdot)$ around x (Definition 6).

  Our main contribution in this work is to provide a theoretical guarantee (see Theorem 1) that counterfactuals with a sufficiently high value of Stability (as defined by our measure) will remain valid with high probability after "naturally-occurring" model change. Our result leverages concentration bounds for Lipschitz functions of independent Gaussian random variables (see Lemma 3).

  Since our proposed Stability measure depends on the local Lipschitz constant which is not always available, we also examine practical relaxations of our measure of the form:

$$\hat{R}_{k,\sigma^2}(x, m) = \frac{1}{k} \sum_{x_i \in N_{x,k}} \left(m(x_i) - |m(x) - m(x_i)| \right).$$

  The first term essentially captures the mean value of the model output in a region around it (higher mean is expected to be more robust and reliable). The second term captures the local variability of the model output in around it (lower variability is expected to be more reliable). This intuition is in alignment with the results in Dutta et al. (2022) for tree-based models.
- **Impossibility under targeted model change:** We also make a clear distinction between our proposed naturally-occurring and targeted model change. Under targeted model change, we provide an impossibility result (see Theorem 2) that given any counterfactual for a model, one can always design a new model that is quite similar to the original model and that renders that particular counterfactual invalid. However, in this work, our focus is on non-targeted model change such as retraining on a few additional data points, changing some hyperparameters or seed, etc., for which we have defined the abstraction of "naturally-occurring" model change (see Definition 5).
- **Experimental results:** We explore methods for incorporating our relaxed measure into generating robust counterfactuals for neural networks. We introduce T-Rex:I (Algorithm 1), which finds robust counterfactuals that are close to the original data point. T-Rex:I can be integrated into any base technique for generating counterfactuals to improve robustness. We also propose T-Rex:NN (Algorithm 2), which generates robust counterfactuals that are data-supported, making them more realistic (along the lines of Dutta et al. (2022) for tree-based models). Our experiments show that T-Rex:I can improve robustness for neural netwroks without significantly increasing the cost, and T-Rex:NN consistently generates counterfactuals that are similar to the data manifold, as measured using the local outlier factor (LOF).

Related Works: Counterfactual explanations have seen growing interest in recent years (Verma et al., 2020; Karimi et al., 2020; Wachter et al., 2017). Regarding the robustness of these counterfactual to model changes, Pawelczyk et al. (2020a); Kanamori et al. (2020); Poyiadzi et al. (2020) argue that counterfactuals situated on the data manifold are more likely to be more robust than the closest counterfactuals. Later, Dutta et al. (2022) demonstrate that generating counterfactuals on the data manifold may not be sufficient for robustness. While the importance of robustness in local explanation methods has been emphasized (Hancox-Li, 2020), the problem of specifically generating robust counterfactuals has been less explored, with the notable exceptions of some recent works (Upadhyay et al., 2021; Rawal et al., 2020;

Black et al., 2021; Dutta et al., 2022; Jiang et al., 2022). In Upadhyay et al. (2021), the authors propose an algorithm called ROAR that uses min-max optimization to find the *closest* counterfactuals that are also robust. In Rawal et al. (2020), the focus is on analytical trade-offs between validity and cost. Jiang et al. (2022) introduces a method for identifying close and robust counterfactuals based on a framework that utilizes interval neural networks. Black et al. (2021) propose that local Lipschitzness can be leveraged to generate consistent counterfactuals and propose an algorithm called Stable Neighbor Search to generate consistent counterfactuals for neural networks. Our research builds on this perspective introduced in Black et al. (2021) and further performs Gaussian sampling around the counterfactual, leading to a novel estimator for which we are also able to provide probabilistic guarantees going beyond the bounded model change assumption. Furthermore, examining all three performance metrics, namely, cost, validity (robustness), and likeness to the data-manifold has received less attention with the notable exception of Dutta et al. (2022) but they focus only on tree-based models (non-differentiable). We also refer to Mishra et al. (2021) for a survey.

We note that Laugel et al. (2019); Alvarez-Melis & Jaakkola (2018) propose an alternate perspective of robustness in explanations (called L-stability in Alvarez-Melis & Jaakkola (2018)) which is built on similar individuals receiving similar explanations. Pawelczyk et al. (2022); Maragno et al. (2023); Dominguez-Olmedo et al. (2022) focus on finding counterfactuals that are robust to small input perturbations (noisy counterfactuals). In contrast, our focus is on counterfactuals remaining valid after some changes to the model, and providing theoretical guarantees thereof.

2. Preliminaries

Let $\mathcal{X} \subseteq \mathbb{R}^d$ denote the input space and let $\mathcal{S}=\{x_i \in \mathcal{X}\}_{i=1}^n$ be a dataset consisting of n independent and identically distributed data points generated from a density q over $\mathcal{X}$. We also let $m(\cdot) : \mathbb{R}^d \to [0, 1]$ denote the original machine learning model that takes a d-dimensional input value and produces an output probability lying between 0 and 1. The final decision is denoted by $\mathbb{1}(m(x) \geq 0.5)$ where $\mathbb{1}(\cdot)$ denotes the indicator function.

Definition 1 ($\gamma-$Lipschitz)**.** *A function $m(\cdot)$ is said to be $\gamma-$Lipschitz if $|m(x)-m(x')|\leq\gamma\|x-x'\|$ for all $x, x'\in\mathbb{R}^d$.*

Here $\|\cdot\|$ denotes the Euclidean norm, i.e., for $u \in \mathbb{R}^d$, we have $\|u\| = \sqrt{u_1^2 + u_2^2 + \ldots + u_d^2}$. In Remark 2, we also discuss relaxations to local Lipschitz constants from global Lipschitz constants. We denote the updated or changed model as $M(\cdot) : \mathbb{R}^d \to [0, 1]$ where M is a random entity. We mostly use capital letters to denote random entities, e.g., M, X, etc., and small letters to denote non-random entities, e.g., m, x, γ, n, etc.

2.1. Background on Counterfactuals

Definition 2 (Closest Counterfactual $\mathcal{C}_p(x, m)$)**.** *Given $x \in \mathbb{R}^d$ such that $m(x) < 0.5$, its closest counterfactual (in terms of l_p-norm) with respect to the model $m(\cdot)$ is defined as a point $x' \in \mathbb{R}^d$ that minimizes the l_p norm $\|x - x'\|_p$ such that $m(x') \geq 0.5$.*

$$\mathcal{C}_p(x, m) = \arg\min_{x' \in \mathbb{R}^d} \|x - x'\|_p \text{ such that } m(x') \geq 0.5.$$

When one is interested in finding counterfactuals by changing as few features as possible, the l_1 norm is used (enforcing a sparsity constraint). These counterfactuals are also called *sparse* counterfactuals (Pawelczyk et al., 2020a).

Closest counterfactuals often fall too far from the data manifold, resulting in unrealistic and anomalous instances, as noted in Poyiadzi et al. (2020); Pawelczyk et al. (2020a); Kanamori et al. (2020); Verma et al. (2020); Karimi et al. (2020); Albini et al. (2022). This highlights the need for generating counterfactuals that lie on the data manifold.

Definition 3 (Closest Data-Manifold Counterfactual $\mathcal{C}_{p,\mathcal{X}}(x, m)$)**.** *Given $x \in \mathbb{R}^d$ such that $m(x) < 0.5$, its closest data-manifold counterfactual $\mathcal{C}_{p,\mathcal{X}}(x, m)$ with respect to the model $m(\cdot)$ and data manifold $\mathcal{X}$ is defined as a point $x' \in \mathcal{X}$ that minimizes the l_p norm $\|x - x'\|_p$ such that $m(x') \geq 0.5$.*

$$\mathcal{C}_{p,\mathcal{X}}(x, m) = \arg\min_{x' \in \mathcal{X}} \|x - x'\|_p \text{ such that } m(x') \geq 0.5.$$

In order to assess the similarity or anomalous nature of a point concerning the given dataset $\mathcal{S} \subseteq \mathcal{X}$, various metrics can be employed, e.g., K-nearest neighbors, Mahalanobis distance, Kernel density, LOF. These metrics play a crucial role in understanding the quality of counterfactual explanations generated by a model. One widely used metric in the literature on counterfactual explanations (Pawelczyk et al., 2020a; Kanamori et al., 2020; Dutta et al., 2022) is the Local Outlier Factor (LOF).

Definition 4 (Local Outlier Factor (Breunig et al., 2000))**.** *For $x \in \mathcal{S}$, let $L_k(x)$ be its k-Nearest Neighbors (k-NN) in $\mathcal{S}$. The k-reachability distance rd_k of x with respect to x' is defined by $rd_k(x, x') = \max\{\delta(x, x'), d_k(x')\}$, where $d_k(x')$ is the distance δ between x' and its the k-th nearest instance on $\mathcal{S}$. The k-local reachability density of x is defined by $lrd_k(x) = |L_k(x)|(\sum_{x' \in L_k(x)} rd_k(x, x'))^{-1}$. Then, the k-LOF of x on $\mathcal{S}$ is defined as follows:*

$$LOF_{k,\mathcal{S}}(x) = \frac{1}{|L_k(x)|} \sum_{x' \in L_k(x)} \frac{lrd_k(x')}{lrd_k(x)}.$$

Here, $\delta(x, x')$ is the distance between two d-dimensional feature vectors. The LOF Predicts -1 for anomalous points and $+1$ for inlier points.

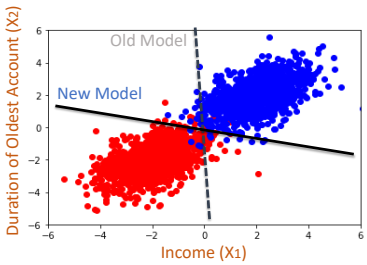


Figure 1. Models can often change drastically in the parameter space causing little to no change in the actual decisions on the points on the data manifold.

Goals: In this work, our main goal is to provide *probabilistic guarantees* on the robustness of counterfactuals to potential model changes for differential models such as neural networks. Towards achieving this goal, our objective involves: (i) introducing an abstraction that rigorously defines the class of model changes that we are interested in; and (ii) establishing a measure, denoted as $R_\Phi(x, m)$, for a counterfactual x and a given model $m(\cdot)$, that quantifies its robustness to potential model changes. Here, Φ represents the hyperparameters of the robustness measure. Ideally, we desire that the measure $R_\Phi(x, m)$ should be high if the counterfactual x is less likely to be invalidated by potential model changes. We seek to provide: (i) theoretical guarantees on the validity of counterfactuals with sufficiently high value of $R_\Phi(x, m)$; and also (ii) incorporate our measure $R_\Phi(x, m)$ into an algorithmic framework for generating robust counterfactuals which also meet other requirements, such as, low cost or likeness to the data manifold.

3. Main Theoretical Contributions

In this section, we first introduce our proposed abstraction of *naturally-occurring* model change and then propose a novel measure – that we call *Stability* – to quantify the robustness of counterfactuals to potential model changes. We derive a theoretical guarantee that counterfactuals that have a sufficiently high value of *Stability* will remain valid after potential *naturally-occurring* model change with high probability. But since our quantification would depend on the local Lipschitz constant around a data point, which is not always known, we also examine a practical relaxation of our proposed measure and demonstrate its applicability.

3.1. Naturally-Occurring Model Change

A popular assumption in existing literature (Upadhyay et al., 2021; Black et al., 2021) to quantify potential model changes is to assume that the model changes are bounded in the parameter space, i.e.,

$$\|\text{Params}(M) - \text{Params}(m)\| < \Delta \text{ for a constant } \Delta.$$

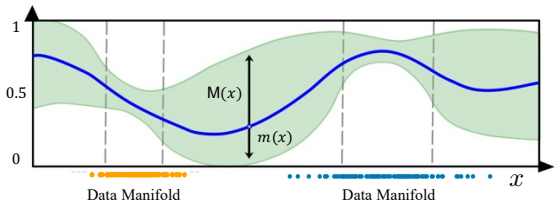


Figure 2. Illustrates our proposed abstraction of naturally-occurring model change: The distribution of the changed model outputs $M(x)$ (stochastic) is centered around the original model output $m(x)$. The points specifically lying on the data-manifold acting as anchors without much change as they exhibit lower variance in model outputs compared to points outside the manifold. This visualization also connects with the Rashomon effect, encapsulating the diverse yet similarly accurate models that can be learned from a given dataset.

Here, Params(M) denote the parameters of the model M, e.g., weights of a neural network. However, we note that models can often change drastically in the parameter space causing little to no change on the actual decisions on the points on the data manifold (see Fig. 1 for an example). In this work, we relax the bounded-model-change assumption, and instead introduce the notion of a naturally-occurring model change as defined in Definition 5. Our abstraction allows for arbitrary model changes such that the change in predictions on points that lie on the data manifold is limited (see Fig. 2 for an illustration).

Definition 5 (Naturally-Occurring Model Change)**.** *A naturally-occurring model change is defined as follows:*

1. $\mathbb{E}[M(X)|X = x] = \mathbb{E}[M(x)] = m(x)$ *where the expectation is over the randomness of* M *given a fixed value of* $X = x \in \mathbb{R}^d$.
2. *Whenever* $m(x)$ *is* γ_m*-Lipschitz, any updated model* $M(x)$ *is also* $\gamma-$*Lipschitz for some constant* γ*. Note that, this constant* γ *does not depend on* M *since we may define* γ *to be an upper bound on the Lipschitz constants for all possible* M *as well as* m.
3. $\text{Var}[M(X)|X = x] = \text{Var}[M(x)] = \nu_x$ *which depends on the fixed value of* $X = x \in \mathbb{R}^d$*. Furthermore, whenever* x *lies on the data manifold* $\mathcal{X}$*, we have* $\nu_x \leq \nu$ *for a small constant* ν.

Closely connected to naturally-occurring model change is the idea of Rashomon effect (Breiman, 2001; Pawelczyk et al., 2020a; Marx et al., 2020; Hsu & Calmon, 2022) which suggests that models can be very different from each other but have almost similar performance on the data manifold, e.g., $\frac{1}{n}\sum_{i=1}^{n} |M(x_i) - m(x_i)|$ is small when the points x_i lie on the data manifold. Under naturally-occurring model change, this holds in expectation as follows:

Lemma 1 (Connection to Roshomon Effect)**.** *For points* $x_1, \ldots, x_n \in \mathcal{X}$ *(lying on the data-manifold) under*

naturally-occurring model change, the following holds:

$$\mathbb{E}\left[\frac{1}{n}\sum_{i=1}^{n}|M(x_i)-m(x_i)|\right]\leq\sqrt{\nu}. \quad (1)$$

Thus, Definition 5 might be better suited over boundedness in the parameter space. Proof of Lemma 1 is in Appendix B.

Remark 1 (Targeted Model Change)**.** *In contrast with naturally-occurring model change, we also introduce the notion of targeted model change (adversarial, worst-case) which essentially refers to a model change that is more deliberately targeted to make a particular counterfactual invalid. For example, one could have a new model $M(x)=m(x)$ almost everywhere except at or around the targeted point x', i.e., $M(x')=1-m(x')$. See Section 3.3 for more details.*

3.2. A Measure of Robustness With Probabilistic Guarantees on Validity

3.2.1. Proposed Measure: Stability

Definition 6 (Stability)**.** *The stability of a counterfactual $x\in\mathbb{R}^d$ is defined as follows:*

$$R_{k,\sigma^2}(x,m)=\frac{1}{k}\sum_{x_i\in N_{x,k}}\left(m(x_i)-\gamma\|x-x_i\|\right), \quad (2)$$

where $N_{x,k}$ is a set of k points drawn from the Gaussian distribution $\mathcal{N}(x,\sigma^2\mathrm{I}_d)$ with I_d being the identity matrix, and γ is an upper bound on the Lipschitz constant for all models $M(\cdot)$ under naturally-occurring change.

Remark 2 (Relaxations to local Lipschitz)**.** *While we prove our theoretical result (Theorem 1) with the global Lipschitz constant γ, we can relax this to local Lipschitz constants γ_x, around a given point x. This is because we sample from a Gaussian centered around the point x and hence mainly capture the variability around x. So most points will be very close to x but a few points can still lie far away. Potential extensions of our guarantees could apply to truncated Gaussian and uniform sampling methods, given their sub-Gaussian properties. This is because Lipschitz concentration inherently extends to sub-Gaussian random variables (Baraniuk et al., 2008).*

3.2.2. Probabilistic Guarantee

Theorem 1 (Probabilistic Guarantee)**.** *Let $X_1,X_2,\ldots,X_k$ be k iid random variables with distribution $\mathcal{N}(x,\sigma^2I_d)$ and $Z=\frac{1}{k}\sum_{i=1}^{k}(m(X_i)-M(X_i))$. Suppose $|\mathbb{E}[Z|M]-\mathbb{E}[Z]|<\epsilon'$. Then, for any $\epsilon>2\epsilon'$, a counterfactual $x\in\mathcal{X}$ under naturally-occurring model change satisfies:*

$$\Pr(M(x)\geq R_{k,\sigma^2}(x,m)-\epsilon)\geq 1-\exp\left(\frac{-k\epsilon^2}{8(\gamma_m+\gamma)^2\sigma^2}\right).$$

The probability is over the randomness of both M and X_i's.

Intuition Behind Our Result: This stability metric (Definition 6) is a way to measure the robustness of counterfactuals that are subject to natural model changes (see Definition 5). The first term in the metric, represented by $\frac{1}{k}\sum_{i=1}^{k}m(X_i)$, captures the average model outputs for a group of points centered around the counterfactual x. The second term, represented by $\gamma\|x-X_i\|$, is an upper bound on the potential difference in outputs of any new model on the points x and X_i (Recall the Lipschitz property of M around the point x). Using our measure, the guarantee in Theorem 1 can be rewritten as:

$$\Pr\left(\frac{1}{k}\sum_{i=1}^{k}m(X_i)-M(x)\leq\frac{\gamma}{k}\sum_{i=1}^{k}\|x-X_i\|+\epsilon\right)\geq 1-\exp\left(\frac{-k\epsilon^2}{8(\gamma+\gamma_m)^2\sigma^2}\right). \quad (3)$$

This form of the inequality allows for the following interpretation of Theorem 1: The distance between the output of the new model on an input x, i.e., $M(x)$, and the average prediction of the neighborhood of the given input by the old model, i.e., $\frac{1}{k}\sum m(X_i)$ is upper bounded by ϵ-corrected, γ multiplied average distance of the datapoints within the neighborhood of the input x, i.e., $\frac{1}{k}\sum\|x-X_i\|$.

Proof Sketch: The complete proof of Theorem 1 is provided in Appendix C.1. Here, we include a proof sketch. Notice that, using the Lipschitz property of $M(\cdot)$ around x, we have $M(x)\geq M(X_i)-\gamma\|x-X_i\|$ for all X_i. Thus,

$$M(x)\geq\frac{1}{k}\sum_{i=1}^{k}(M(X_i)-\gamma\|x-X_i\|) \quad (4)$$

$$\overset{(a)}{\geq}\frac{1}{k}\sum_{i=1}^{k}(m(X_i)-\gamma\|x-X_i\|)-\epsilon, \quad (5)$$

where (a) holds from Lemma 2 with probability at least $1-\exp\left(\frac{-k\epsilon^2}{8(\gamma+\gamma_m)^2\sigma^2}\right)$.

Lemma 2 (Deviation Bound)**.** *Let $X_1,X_2,\ldots,X_k\sim\mathcal{N}(x,\sigma^2I_d)$ and $Z=\frac{1}{k}\sum_{i=1}^{k}(m(X_i)-M(X_i))$. Suppose $|\mathbb{E}[Z|M]-\mathbb{E}[Z]|<\epsilon'$. Then, under naturally-occurring model change, we have $\mathbb{E}[Z]=0$. Moreover, for any $\epsilon>2\epsilon'$,*

$$\Pr(Z\geq\epsilon)\leq\exp\left(\frac{-k\epsilon^2}{8(\gamma+\gamma_m)^2\sigma^2}\right). \quad (6)$$

Proof Sketch: The proof of Lemma 2 leverages concentration bounds for Lipschitz functions of independent Gaussian random variables (see Lemma 3). The complete proof of Lemma 2 is provided in Appendix C. □

Lemma 3 (Gaussian Concentration Inequality)**.** *Let $W=(W_1,W_2,\ldots,W_n)$ consist of n i.i.d. random variables be-*

longing to $\mathcal{N}(0, \sigma^2)$, and $Z = f(W)$ be a γ-Lipschitz function, i.e., $|f(W) - f(W')| \leq \gamma\|W - W'\|$. Then, we have,

$$\Pr(Z - \mathbb{E}[Z] \geq \epsilon) \leq \exp\left(\frac{-\epsilon^2}{2\gamma^2\sigma^2}\right) \text{ for all } \epsilon > 0. \quad (7)$$

For the proof of Lemma 3 refer to Boucheron et al. (2013) in p.125. Our robustness guarantee (Theorem 1) essentially states that $\Pr(M(x) \leq R_{k,\sigma^2}(x, m) - \epsilon) \leq \exp\left(\frac{-k\epsilon^2}{8(\gamma+\gamma_m)^2\sigma^2}\right)$ under naturally-occurring model change. For instance, if we find a counterfactual x such that $R_{k,\sigma^2}(x, m) - \epsilon$ is greater or equal to 0.5, then $M(x)$ would also be greater than 0.5 with high probability. The term $\exp\left(\frac{-k\epsilon^2}{8(\gamma+\gamma_m)^2\sigma^2}\right)$ decays with k.

3.2.3. Practical Relaxation of Stability and Its Properties

While our proposed measure Stability (Definition 6) has probabilistic guarantees, we note that it relies on the Lipschitz constant γ (or the local Lipschitz constant γ_x around the point x), which is often unknown. Therefore, we next propose a practical relaxation of the measure as follows:

Definition 7 (Stability (Relaxed))**.** *The stability (relaxed) of a counterfactual $x \in \mathbb{R}^d$ is defined as follows:*

$$\hat{R}_{k,\sigma^2}(x, m) = \frac{1}{k} \sum_{x_i \in N_{x,k}} (m(x_i) - |m(x) - m(x_i)|),$$

where $N_{x,k}$ is a set of k points drawn from the Gaussian distribution $\mathcal{N}(x, \sigma^2 \mathrm{I}_d)$ with I_d being the identity matrix.

To arrive at this relaxation, we utilize the Lipschitz property to approximate the aspect that involves the Lipschitz constant, specifically, by approximating $\gamma_x||x - x_i||$ with $|m(x) - m(x_i)|$. Another possibility is to consider an estimate of γ_x given by:

$$\hat{\gamma}_x = \max_{x_i \in N_{x,k}} \frac{|m(x) - m(x_i)|}{\|x - x_i\|}. \quad (8)$$

We observed that the experimental results with both these stability estimates are in the same ballpark.

To gain a deeper understanding of the relaxed stability measure, we now consider some desirable properties of counterfactuals that make them more robust and then demonstrate that our proposed relaxation of Stability (Definition 7) satisfies those desirable properties. These properties are inspired from Dutta et al. (2022) which proposed these properties for tree-based ensembles. The first property is based on the fact that the output of a model $m(x) \in [0, 1]$ is expected to be higher if the model has more confidence in that prediction.

Property 1. *For any $x \in \mathbb{R}^d$, a higher value of $m(x)$ makes it less likely to be invalidated due to model changes.*

Having a high $m(x)$ alone does not guarantee robustness, as local variability around x can make predictions less reliable. E.g., points with high $m(x)$ near the decision boundary are also vulnerable to invalidation with model changes.

Property 2. *An $x \in \mathbb{R}^d$ is less likely to be invalidated if several points close to x (denoted by x') have a high value of $m(x')$.*

Counterfactuals may also be more likely to be invalidated if it lies in a highly variable region of the model output function. This is because the confidence of the model predictions in that region may be less reliable.

Property 3. *An $x \in \mathbb{R}^d$ is less likely to be invalidated if model outputs around x have low variability.*

Our stability measure aligns with these desired properties. Given a point $x \in \mathbb{R}^d$, it generates a set of k points centered around x. The first term $\frac{1}{k}\sum_{x' \in N_{x,k}} m(x')$ is expected to be high if the model output value $m(x)$ is high for x as well as several points close to x. But the mean value of $m(x')$ around a point x may not always capture the variability in that region, hence, the second term of our stability measure, i.e., $\frac{1}{k}\sum_{x' \in N_{x,k}} |m(x) - m(x')|$. This term captures the variability of the model output values in a region around x.

It is worth noting that the variability term is only useful in conjunction with the mean term. This is because even points on the opposite side of the decision boundary can have varying levels of variability, regardless of whether $m(x')$ is less or greater than 0.5.

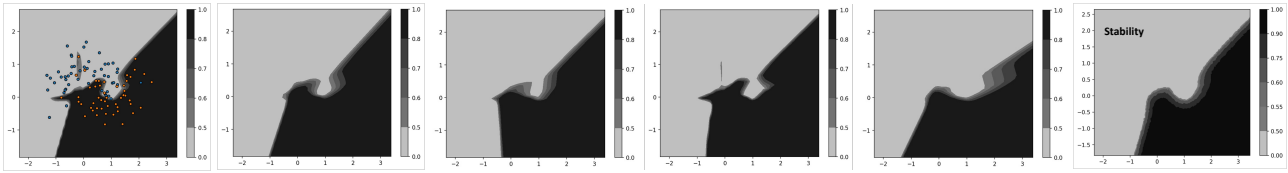
In Fig. 3, we provide an example on the synthetic moon dataset to observe the effect of our stability measure on naturally-changed models. Note that these changed models were realized from actual experiments by retraining with different weight initializations.

3.3. Impossibility Under Targeted Model Change

In this work, we make a key distinction between naturally-occurring and targeted model changes. While we are able to provide probabilistic guarantees for naturally-occurring model change, we also demonstrate an impossibility result for targeted model change. What this result essentially demonstrates is that for a given model, one can design another similar model such that any particular targeted counterfactual can be invalidated.

Theorem 2 (Impossibility Under Targeted Change)**.** *Given a model and a counterfactual, one can design another similar model such that the particular targeted counterfactual can be invalidated.*

The proof relies on the possibility that one could have a new model $M(x) = m(x)$ almost everywhere except at or around the targeted point x', i.e., $M(x') = 1 - m(x')$.



(a) m (b) $M = m_1$ (c) $M = m_2$ (d) $M = m_3$ (e) $M = m_4$ (f) $\hat{R}_{k,\sigma^2}(x, m)$

Figure 3. Effect of stability measure on naturally-occurring model changes: (a) corresponds to the original data distribution and the trained model. (b)-(e) demonstrate some examples of changed models obtained on retraining with different weight initializations. One may notice that the model decision boundary is changing a lot in the sparse regions of the data-manifold (few data-points), possibly violating the bounded-parameter change assumption but the predictions on the dense regions of the data-manifold do not change much (in alignment with Rashomon effect). This motivates our proposed abstraction of naturally-occurring model change which allows for arbitrary changes in the parameter space with little change in the actual predictions on the dense regions of the data manifold. (f) demonstrates our proposed measure of stability $\hat{R}_{k,\sigma^2}(x, m)$ (high mean model output, low variability, *almost* like a Gaussian filter) for which we derive probabilistic guarantees on validity. In essence, we show that under the abstraction of naturally-occurring model change, the stability measure captures the reliable intersecting region of changed models with high probability. In the original model, we observe that certain non-robust regions (i.e., those caused by overfitting to certain data points in the original model) have higher local Lipschitz values and variability. Counterfactuals assigned to these regions (even if $m(x)$ is high) would be invalidated in the changed models. The stability measure, which samples around a region, penalizes these higher local Lipschitz values.

4. Generating Robust Counterfactuals using Our Proposed Measure: Stability

In this section, we examine two techniques of incorporating our proposed measure, Stability (relaxed; see Definition 7), for generating robust counterfactuals for neural networks.

To begin, along the lines of Dutta et al. (2022), we first define a counterfactual robustness test.

Definition 8 (Counterfactual Robustness Test)**.** *A counterfactual $x \in \mathbb{R}^d$ satisfies the robustness test if:*

$$\hat{R}_{k,\sigma^2}(x, m) \geq \tau. \tag{9}$$

Now, we would like to find a reasonable point x' that receives a positive prediction from the model (essentially $m(x') \geq 0.5$), while also satisfying the robustness test, $\hat{R}_{k,\sigma^2}(x', m) \geq \tau$. The threshold value of τ can be adjusted based on the desired effective validity (recall Theorem 1). Hence, a larger threshold would likely ensure that the new model, M, remains valid with high probability. In trying to find a reasonable point x', one may strive to generate robust counterfactuals that are as close as possible to the original point. One might also want the generated counterfactuals to be as realistic as possible, i.e., lie on the data manifold. Toward that end, we propose two algorithms.

We propose Algorithm 1, T-Rex:I, which incorporates our measure to find robust counterfactuals that are close to the original data point. T-Rex:I works with any preferred base method for generating counterfactuals. It evaluates the stability of the generated counterfactual and, if necessary, iteratively updates the generated counterfactual through a gradient ascent process until a robust counterfactual that meets the desired criteria is obtained.

Remark 3 (Gradient of Stability)**.** *In Algorithm 1, we compute the gradient of $R(x, m)$ with respect to x (not model parameters m). Such gradients w.r.t. x instead of m are also computed commonly in adversarial machine learning and also in feature-attributions for explainability. We use TensorFlow* `tf.GradientTape` *for automatic differentiation, which allows for the computation of gradients with respect to certain inputs.*

To ensure that the counterfactuals are as realistic as possible, we also define the T-Rex:NN Counterfactual, which considers counterfactuals that lie within a dataset to avoid any unrealistic or anomalous results (see Algorithm 2).

Definition 9 (Robust Nearest Neighbor Counterfactual)**.** *Given $x \in \mathbb{R}^d$ such that $m(x) < 0.5$, its robust nearest neighbor counterfactual $\mathcal{C}^{(\tau)}_{p,\mathcal{S}}(x, m)$ with respect to the model $m(\cdot)$ and dataset $\mathcal{S}$ is defined as another point $x' \in \mathcal{S}$ that minimizes the l_p norm $\|x - x'\|_p$ such that $m(x') \geq 0.5$ and $\hat{R}_{k,\sigma^2}(x', m) \geq \tau$.*

The closest data-supported counterfactual serves as a reliable reference, as it inherently has a high Local Outlier Factor (LOF). The T-Rex:I algorithm may find counterfactuals with lower costs, but they may compromise on the LOF and result in unrealistic samples.

To address this, we propose Algorithm 2, T-Rex:NN, for finding data-supported counterfactuals. The algorithm first finds k nearest neighbors counterfactuals of x in dataset $\mathcal{S}$, checks through each of them to see if they satisfy the robustness test, $\hat{R}_{k,\sigma^2}(x', m) \geq \tau$, and terminates once such a counterfactual is found.

Algorithm 1 T-Rex:I: Theoretically Robust EXplanations: Iterative Version

Input: Model $m(\cdot)$, Datapoint x with $m(x) < 0.5$, Algorithm parameters (k, σ^2, η, τ, max_steps).
Generate initial counterfactual x' using any technique.
Initialize robust counterfactual $x_c = x'$ and steps = 0.
while $\hat{R}_{k,\sigma^2}(x_c, m) < \tau$ and steps $<$ max_steps **do**
  Compute $\hat{R}_{k,\sigma^2}(x_c, m)$
  Compute gradient $\Delta_{x_c}\hat{R}_{k,\sigma^2}(x_c, m)$
  Update x_c via gradient ascent:
    $x_c = x_c + \eta\Delta_{x_c}\hat{R}_{k,\sigma^2}(x_c, m)$
  Increment steps
end while
Output x_c and exit

Algorithm 2 T-Rex:NN: Theoretically Robust EXplanations: Nearest Neighbor Version

Input: Model $m(\cdot)$, Datapoint x with $m(x)<0.5$, Dataset $\mathcal{S}$, Algorithm parameters (K, σ^2, k, τ).
Find K nearest neighbor counterfactuals $x'_i \in \mathcal{S}$ to x with respect to model $m(\cdot)$, i.e., $\text{NN}_x = (x'_1, x'_2, \ldots, x'_K)$.
for $x'_i \in \text{NN}_x$ **do**
  Perform counterfactual robustness test on x'_i:
    Check if $\hat{R}_{k,\sigma^2}(x'_i, m) \geq \tau$
  if counterfactual robustness test is satisfied: **then**
    Output x'_i and exit
  end if
end for
Output: *No robust counterfactual found* and exit

5. Experiments

In this section, we present experimental results to demonstrate the effectiveness of our proposed measure in capturing robustness and then generating robust counterfactuals that remain valid after potential model changes. We illustrate how our proposed Algorithm 1 & 2 utilizes our stability measure to effectively generate robust counterfactuals.

Datasets: We conduct experiments on several benchmark datasets, namely, HELOC (FICO, 2018), German Credit, Cardiotocography (CTG), Adult (Dua & Graff, 2017), and Taiwanese Credit (Yeh & hui Lien, 2009). These have two classes, with one class representing the most favorable outcome, and the other representing the least desirable outcome for which we aim to generate counterfactuals. For simplicity, we normalize the features to lie between $[0, 1]$.

Performance Metrics: Our metrics of interest are:

- Cost: Average l_1 or l_2 distance between counterfactuals x' and original points x.
- Validity (%): Percentage of counterfactuals that remain valid under the new model M.
- LOF: Predicts -1 for anomalous points, and $+1$ for inliers. A high average LOF essentially suggests the points lie on the data manifold and hence more realistic, i.e., *higher is better* (see Definition 4). We use an existing implementation from Scikit-Learn to compute the LOF.

Methodology. We begin by training a baseline neural network model and aim to find counterfactuals for data points with true negative predictions. To test the robustness of these counterfactual examples, we then train 50 new models (M) and evaluate the $validity$ of the counterfactuals under different model change scenarios, which include: (i) Weight Initialization (WI): Retraining new models using the same hyperparameters but with different weight initialization by using different random seeds for each new model; and (ii) Leave Out (LO): Retraining new models by randomly removing a small portion (1%) of the training data each time (with replacement) as well as different weight initialization.

Hyperparameter selection: Our theoretical findings indicate that higher k improves robustness, but comes at the cost of increased computational cost. We determine $k = 1000$ was sufficient. The value of σ^2 was determined by analyzing the variance of the features. In the dataset with the features between $[0, 1]$, we found that a value of $\sigma^2 = 0.01$ produced good results. The threshold τ is a critical aspect of our method and can be adjusted based on the desired effective validity. A higher τ value improves validity at the expense of l_1 or l_2 cost. See Appendix D for more details.

Baseline: We compare our approaches with established baselines. First, we find the min Cost (l_1 and l_2) counterfactual (Wachter et al., 2017) and use it as our base method for generating counterfactuals. We then compare T-Rex:I to the Stable Neighbor Search (SNS) (Black et al., 2021) and Robust Algorithmic Recourse (ROAR) (Upadhyay et al., 2021). We evaluate the performance of our Robust Nearest Neighbor (Algorithm 2:T-Rex:NN) against the Nearest Neighbor (NN) counterfactuals (closest data-support robust counterfactual in Definition 9). We choose a value of τ to get high validity and compare cost and LOF with baselines.

Results: Results for HELOC, German Credit, and CTG datasets are summarized in Table 1. Observe that the min Cost counterfactual is not robust to variations in the training data or weight initialization as expected. ROAR generates counterfactuals with high validity, albeit at the expense of a higher cost. Our proposed method, T-Rex:I, significantly improves the validity of the counterfactuals compared to the minimum cost. The T-Rex:I algorithm achieves comparable validity results to the SNS method for both types of model changes, and often accomplishes this with lower costs and higher LOF. This can be observed across all three datasets for both l_1 and l_2 cost metrics. The T-Rex:NN algorithm

Table 1. Experimental results.

	Method	l_1 *based*				l_2 *based*			
		COST	LOF	WI VAL.	LO VAL.	COST	LOF	WI VAL.	LO VAL.
HELOC	min Cost	0.40	0.49	38.8%	35.2%	0.11	0.75	13.5%	13.5%
	min Cost+T-Rex:I (Ours)	1.02	0.38	98.2%	98.1%	0.29	0.68	98.5%	98.2%
	min Cost+SNS	1.20	0.30	98.0%	97.8%	0.31	0.64	97.9%	97.0%
	ROAR	1.69	0.41	92.6%	91.2%	1.91	0.43	86.3 %	84.8%
	NN	1.91	0.80	51.1%	50.3%	0.56	0.80	51.1%	50.3%
	T-Rex:NN (Ours)	2.50	0.92	84.0%	84.0%	0.77	0.92	84.0%	84.0%
GERMAN	min Cost	1.42	0.77	58.8%	56.7%	0.48	0.81	26.6%	26.6%
	min Cost+T-Rex:I (Ours)	4.81	0.72	98.0%	96.5%	1.20	0.75	99.2%	98.7%
	min Cost+SNS	5.71	0.67	97.5%	98.1%	1.44	0.68	99.9%	98.9%
	ROAR	7.63	0.54	96.3%	92.3%	6.81	0.58	87.8%	85.2%
	NN	7.05	1.00	95.3%	95.4%	2.50	1.00	95.3%	95.3%
	T-Rex:NN (Ours)	10.13	1.00	100%	100%	3.04	1.00	100%	100%
CTG	min Cost	0.21	0.94	74.6%	70.2%	0.08	1.00	19.7%	14.1%
	min Cost+T-Rex:I (Ours)	1.11	0.83	100%	98.8%	0.42	0.94	100%	99.7%
	min Cost+SNS	3.34	-1.00	100%	98.2%	1.07	-1.00	100%	99.3%
	ROAR	3.68	0.64	98.7%	96.4%	1.35	0.59	98.9%	97.2%
	NN	0.39	1.00	70.5%	67.5%	0.15	1.00	70.5%	67.5%
	T-Rex:NN (Ours)	2.22	-0.33	100%	100%	1.00	-0.33	100%	100%

also significantly improves the validity of the counterfactuals compared to the traditional Nearest Neighbor (NN) method and maintains a high LOF. It comes at a price of increased cost, but the counterfactuals are guaranteed to be realistic since they are data supported. Refer to Appendix D for additional results for Adult and Taiwanese credit datasets.

Ablation: To evaluate the efficacy of our proposed stability measure, we conduct an ablation study on the German credit dataset. We first evaluate a robustness measure that solely relies on the model's prediction of the counterfactual, denoted as $r(x', m) = m(x')$. We then examine a measure that only incorporates the mean, the average predictions for k points sampled from the distribution $N(x', \sigma^2 I_d)$, denoted as $r_{k,\sigma^2}(x', m) = \frac{1}{k}\sum_{x'_i \in N_{x',k}} m(x'_i)$. We compare these with our proposed robustness measure $\hat{R}_{k,\sigma^2}(x', m)$, which also takes into account the variability around the counterfactual. The results of the ablation study, for various τ thresholds, are summarized in Table 7 in Appendix D.

6. Discussion

We introduce an abstraction called naturally-occurring model change and propose a measure, Stability, to quantify the robustness of counterfactuals with probabilistic guarantees. We show that counterfactuals with high Stability will remain valid after potential model changes with high probability. We investigate various techniques for incorporating stability in generating robust counterfactuals and introduce the T-Rex:I and T-Rex:NN algorithms.

Limitations and Broader Impact: The naturally-occurring model changes rest on assumptions that may not apply to all models or datasets. Our relaxed stability, although practically implementable, lacks the same theoretical guarantees as the initial stability measure. Estimating the Lipschitz constant around a counterfactual can be computationally demanding, particularly when leveraging gradient descent to optimize stability. Though generating robust counterfactuals is a key step towards trustworthy AI, it can fall short of other important factors such as fairness (Sharma et al., 2019; Gupta et al., 2019; Ley et al., 2022; Raman et al., 2023; Ehyaei et al., 2023). Future research could explore links between robustness and fairness, improving the estimation of stability, or integrating Stability into training-time-based approaches for generating robust counterfactuals.

Disclaimer This paper was prepared for informational purposes in part by the Artificial Intelligence Research group of JPMorgan Chase & Co. and its affiliates ("JP Morgan"), and is not a product of the Research Department of JP Morgan. JP Morgan makes no representation and warranty whatsoever and disclaims all liability, for the completeness, accuracy, or reliability of the information contained herein. This document is not intended as investment research or investment advice, or a recommendation, offer or solicitation for the purchase or sale of any security, financial instrument, financial product or service, or to be used in any way for evaluating the merits of participating in any transaction, and shall not constitute a solicitation under any jurisdiction or to any person, if such solicitation under such jurisdiction or to such person would be unlawful.

References

Albini, E., Long, J., Dervovic, D., and Magazzeni, D. Counterfactual shapley additive explanations. *ACM Conference on Fairness, Accountability, and Transparency*, 2022.

Alvarez-Melis, D. and Jaakkola, T. S. On the robustness of interpretability methods. *arXiv preprint arXiv:1806.08049*, 2018.

Baraniuk, R., Davenport, M., DeVore, R., and Wakin, M. A simple proof of the restricted isometry property for random matrices. *Constructive approximation*, 28:253–263, 2008.

Barocas, S., Selbst, A. D., and Raghavan, M. The hidden assumptions behind counterfactual explanations and principal reasons. In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, pp. 80–89, 2020.

Black, E., Wang, Z., Fredrikson, M., and Datta, A. Consistent counterfactuals for deep models. *arXiv preprint arXiv:2110.03109*, 2021.

Boucheron, S., Lugosi, G., and Massart, P. *Concentration inequalities: A nonasymptotic theory of independence*. Oxford university press, 2013.

Breiman, L. Statistical modeling: The two cultures. *Quality Engineering*, 48:81–82, 2001.

Breunig, M. M., Kriegel, H.-P., Ng, R. T., and Sander, J. Lof: identifying density-based local outliers. In *Proceedings of the 2000 ACM SIGMOD international conference on Management of data*, pp. 93–104, 2000.

Dominguez-Olmedo, R., Karimi, A. H., and Schölkopf, B. On the adversarial robustness of causal algorithmic recourse. In Chaudhuri, K., Jegelka, S., Song, L., Szepesvari, C., Niu, G., and Sabato, S. (eds.), *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 5324–5342. PMLR, 17–23 Jul 2022. URL https://proceedings.mlr.press/v162/dominguez-olmedo22a.html.

Dua, D. and Graff, C. UCI machine learning repository, 2017. URL http://archive.ics.uci.edu/ml.

Dutta, S., Long, J., Mishra, S., Tilli, C., and Magazzeni, D. Robust counterfactual explanations for tree-based ensembles. In *International Conference on Machine Learning*, pp. 5742–5756. PMLR, 2022.

Ehyaei, A.-R., Karimi, A.-H., Schölkopf, B., and Maghsudi, S. Robustness implies fairness in casual algorithmic recourse. *arXiv preprint arXiv:2302.03465*, 2023.

FICO. FICO XML Challenge. https://community.fico.com/s/explainable-machine-learning-challenge, 2018.

Gupta, V., Nokhiz, P., Roy, C. D., and Venkatasubramanian, S. Equalizing recourse across groups. *arXiv preprint arXiv:1909.03166*, 2019.

Hancox-Li, L. Robustness in Machine Learning Explanations: Does It Matter? In *Proceedings of the 3rd ACM Conference on Fairness, Accountability, and Transparency (FAT*)*, pp. 640–647. Barcelona, Spain, January 27–30 2020.

Hsu, H. and Calmon, F. d. P. Rashomon capacity: A metric for predictive multiplicity in probabilistic classification. *arXiv preprint arXiv:2206.01295*, 2022.

Jiang, J., Leofante, F., Rago, A., and Toni, F. Formalising the robustness of counterfactual explanations for neural networks. *arXiv preprint arXiv:2208.14878*, 2022.

Kanamori, K., Takagi, T., Kobayashi, K., and Arimura, H. Dace: Distribution-aware counterfactual explanation by mixed-integer linear optimization. In *IJCAI*, pp. 2855–2862, 2020.

Karimi, A., Barthe, G., Schölkopf, B., and Valera, I. A survey of algorithmic recourse: definitions, formulations, solutions, and prospects. *CoRR*, abs/2010.04050, 2020. URL https://arxiv.org/abs/2010.04050.

Laugel, T., Lesot, M.-J., Marsala, C., and Detyniecki, M. Issues with post-hoc counterfactual explanations: a discussion. *arXiv preprint arXiv:1906.04774*, 2019.

Ley, D., Mishra, S., and Magazzeni, D. Global counterfactual explanations: Investigations, implementations and improvements, 2022.

Maragno, D., Kurtz, J., Röber, T. E., Goedhart, R., Birbil, Ş. I., and Hertog, D. d. Finding regions of counterfactual explanations via robust optimization. *arXiv preprint arXiv:2301.11113*, 2023.

Marx, C., Calmon, F., and Ustun, B. Predictive multiplicity in classification. In *International Conference on Machine Learning*, pp. 6765–6774. PMLR, 2020.

Mishra, S., Dutta, S., Long, J., and Magazzeni, D. A Survey on the Robustness of Feature Importance and Counterfactual Explanations. *arXiv e-prints*, arXiv:2111.00358, 2021.

Pawelczyk, M., Broelemann, K., and Kasneci, G. On counterfactual explanations under predictive multiplicity. In *Conference on Uncertainty in Artificial Intelligence*, pp. 809–818. PMLR, 2020a.

Pawelczyk, M., Broelemann, K., and Kasneci, G. Learning model-agnostic counterfactual explanations for tabular data. In *Proceedings of The Web Conference 2020*, pp. 3126–3132, 2020b.

Pawelczyk, M., Datta, T., van-den Heuvel, J., Kasneci, G., and Lakkaraju, H. Probabilistically robust recourse: Navigating the trade-offs between costs and robustness in algorithmic recourse. *arXiv preprint arXiv:2203.06768*, 2022.

Poyiadzi, R., Sokol, K., Santos-Rodriguez, R., De Bie, T., and Flach, P. Face: Feasible and actionable counterfactual explanations. In *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*, pp. 344–350, 2020.

Raman, N., Magazzeni, D., and Shah, S. Bayesian hierarchical models for counterfactual estimation. In *International Conference on Artificial Intelligence and Statistics*, pp. 1115–1128. PMLR, 2023.

Rawal, K., Kamar, E., and Lakkaraju, H. Can I still trust you?: Understanding the impact of distribution shifts on algorithmic recourses. *arXiv preprint arXiv:2012.11788*, 2020.

Scikit-Learn. LOF Implementation. URL `https://scikit-learn.org/stable/modules/generated/sklearn.neighbors.LocalOutlierFactor.html`.

Sharma, S., Henderson, J., and Ghosh, J. Certifai: Counterfactual explanations for robustness, transparency, interpretability, and fairness of artificial intelligence models. *arXiv preprint arXiv:1905.07857*, 2019.

Upadhyay, S., Joshi, S., and Lakkaraju, H. Towards robust and reliable algorithmic recourse. *Advances in Neural Information Processing Systems*, 34, 2021.

Verma, S., Dickerson, J., and Hines, K. Counterfactual explanations for machine learning: A review. *arXiv preprint arXiv:2010.10596*, 2020.

Wachter, S., Mittelstadt, B., and Russell, C. Counterfactual explanations without opening the black box: Automated decisions and the gdpr. *Harv. JL & Tech.*, 31:841, 2017.

Yeh, I.-C. and hui Lien, C. The comparisons of data mining techniques for the predictive accuracy of probability of default of credit card clients. *Expert Systems with Applications*, 36(2, Part 1):2473–2480, 2009. ISSN 0957-4174. doi: https://doi.org/10.1016/j.eswa.2007.12.020. URL `https://www.sciencedirect.com/science/article/pii/S0957417407006719`.

A. Relevant Inequalities

Lemma 4 (Cauchy-Schwarz Inequality)**.** *If $\mathbf{u}, \mathbf{v} \in V$, where V is a vector space, then*

$$|\langle \mathbf{u}, \mathbf{v} \rangle| \leq \|\mathbf{u}\| \|\mathbf{v}\|.$$

This inequality is an equality if and only if one of $\mathbf{u}, \mathbf{v}$ is a scalar multiple of the other.

Lemma 5 (Jensens Inequality)**.** *Let X be an integrable random variable. Let $g : \mathbb{R} \to \mathbb{R}$ be a convex function such that $Y = g(X)$ is also integrable. Then, the following inequality, called Jensen's inequality, holds:*

$$\mathrm{E}[g(X)] \geq g(\mathrm{E}[X]).$$

B. Proof of Lemma 1

Lemma 1 (Connection to Roshomon Effect)**.** *For points $x_1, \ldots, x_n \in \mathcal{X}$ (lying on the data-manifold) under naturally-occurring model change, the following holds:*

$$\mathbb{E}\left[\frac{1}{n}\sum_{i=1}^{n} |M(x_i) - m(x_i)|\right] \leq \sqrt{\nu}. \tag{1}$$

Proof.

$$\mathbb{E}\left[\frac{1}{n}\sum_{i=1}^{n} |m(x_i) - M(x_i)|\right] \overset{(a)}{\leq} \mathbb{E}\left[\sqrt{\sum_{i=1}^{n}\frac{1}{n^2}}\sqrt{\sum_{i=1}^{n}|m(x_i) - M(x_i)|^2}\ \right] \tag{10}$$

$$= \frac{1}{\sqrt{n}}\,\mathbb{E}\left[\sqrt{\sum_{i=1}^{n}|m(x_i) - M(x_i)|^2}\ \right] \tag{11}$$

$$\overset{(b)}{\leq} \frac{1}{\sqrt{n}}\sqrt{\mathbb{E}\left[\sum_{i=1}^{n}|m(x_i) - M(x_i)|^2\right]} \tag{12}$$

$$= \frac{1}{\sqrt{n}}\sqrt{\sum_{i=0}^{n}\nu_{x_i}} \tag{13}$$

$$\overset{(c)}{\leq} \frac{1}{\sqrt{n}}\sqrt{\sum_{i=0}^{n}\nu} = \sqrt{\nu}. \tag{14}$$

Here (a) holds from Cauchy-Schwarz Inequality (Lemma 4) applied on the dot product of the two vectors $[1/n, 1/n, \ldots, 1/n]$ and $[|m(x_1) - M(x_1)|, |m(x_2) - M(x_2)|, \ldots, |m(x_n) - M(x_n)|]$. Next, (b) holds from Jensen's Inequality (Lemma 5) applied on concave function $f(u) = \sqrt{u}$. Finally, (c) holds because the points $x_1, x_2, \ldots, x_n$ lie on the data-manifold and hence the variance $\nu_{x_i} \leq \nu$ from Definition 5.

□

C. Proof of Probabilistic Guarantee

To prove Theorem 1, we begin with Lemma 2.

Assume the changed model M comes from a discrete class of random variables. A possible realization of a model is denoted by $\tilde{m}_i$, with $i = 1, 2, \ldots, n$. The set of all possible models is denoted by $\mathcal{M} = \{\tilde{m}_1, \tilde{m}_2, \ldots, \tilde{m}_n\}$. Let $M = \tilde{m}_i$ with probability p_i such that $\sum_{i=1}^{n} p_i = 1$.

Lemma 2 (Deviation Bound)**.** *Let $X_1, X_2, \ldots, X_k \sim \mathcal{N}(x, \sigma^2 I_d)$ and $Z{=}\frac{1}{k}\sum_{i=1}^{k}(m(X_i){-}M(X_i))$. Suppose $|\mathbb{E}[Z|M]{-}\mathbb{E}[Z]| < \epsilon'$. Then, under naturally-occurring model change, we have $\mathbb{E}[Z]{=}0$. Moreover, for any $\epsilon{>}2\epsilon'$,*

$$\Pr(Z \geq \epsilon) \leq \exp\left(\frac{-k\epsilon^2}{8(\gamma + \gamma_m)^2\sigma^2}\right). \tag{6}$$

Proof. To prove Lemma 2, notice that,

$$\begin{aligned}\mathbb{E}\left[Z\right] &= \frac{1}{k}\sum_{i=1}^{k}\mathbb{E}_{X_i}\left[\mathbb{E}_{M|X_i}\left[m(X_i)-M(X_i)\right]\right] \\ &\overset{(a)}{=} \frac{1}{k}\sum_{i=1}^{k}\mathbb{E}_{X_i}\left[m(X_i)-m(X_i)\right]=0,\end{aligned} \tag{15}$$

where (a) holds from the naturally occurring model change assumption (see Definition 1). The remaining part of the proof leverages concentration bounds for Lipschitz functions of independent Gaussian random variables outlined in Lemma 3.

Lemma 3 (Gaussian Concentration Inequality)**.** *Let $W=(W_1,W_2,\ldots,W_n)$ consist of n i.i.d. random variables belonging to $\mathcal{N}(0,\sigma^2)$, and $Z=f(W)$ be a γ-Lipschitz function, i.e., $|f(W)-f(W')|\leq\gamma\|W-W'\|$. Then, we have,*

$$\Pr(Z-\mathbb{E}\left[Z\right]\geq\epsilon)\leq\exp\left(\frac{-\epsilon^2}{2\gamma^2\sigma^2}\right) \textit{ for all } \epsilon>0. \tag{7}$$

Now, let X_{ij} denote the j-th element of $X_i\in\mathbb{R}^d$ and x_j denote the j-th element of $x\in\mathbb{R}^d$. We define a $k\times d$ matrix $W=[W_{ij}]_{i=1,2,\ldots,k,\text{ and }j=1,2,\ldots,d}$ with $W_{ij}=X_{ij}-x_j$. Notice that, $W_{ij}\sim\mathcal{N}(0,\sigma^2)$ for $i=1,2,\ldots,k$ and $j=1,2,\ldots,d$ and, we can write $Z=f(W)$ with Lipschitz constant $(\gamma_m+\gamma)/\sqrt{k}$.

$$\begin{aligned}&|f(W)-f(W')| \\ &=\left|\frac{1}{k}\sum_{i=1}^{k}(m(X_i)-M(X_i)-m(X_i')+M(X_i'))\right| \\ &\overset{(a)}{\leq}\frac{1}{k}\sum_{i=1}^{k}(|m(X_i)-m(X_i')|+|M(X_i)-M(X_i')|) \\ &\overset{(b)}{\leq}\frac{1}{k}\sum_{i=1}^{k}(\gamma_m+\gamma)\|X_i-X_i'\|_2 \\ &\overset{(c)}{\leq}\frac{(\gamma_m+\gamma)\sqrt{k}}{k}\sqrt{\sum_{i=1}^{k}\|X_i-X_i'\|_2^2} \\ &=\frac{(\gamma_m+\gamma)}{\sqrt{k}}\sqrt{\sum_{i=1}^{k}\sum_{j=1}^{d}|W_{ij}-W_{ij}'|^2} \\ &=\frac{(\gamma_m+\gamma)}{\sqrt{k}}\|W-W'\|_2.\end{aligned} \tag{16}$$

□

Here, (a) holds from the triangle inequality. (b) follows directly from the definition of γ-Lipschitz (Definition 1). (c) can be obtained by using the Cauchy-Schwarz Inequality (Lemma 4).

Now, we substitute these expressions in the Gaussian concentration bound (Lemma 3).

$$\Pr(Z-\mathbb{E}\left[Z|M=\tilde{m}\right]\geq\tilde{\epsilon}|M=\tilde{m})\leq\exp\left(\frac{-k\tilde{\epsilon}^2}{2(\gamma+\gamma_m)^2\sigma^2}\right). \tag{17}$$

Since $|\mathbb{E}\left[Z|M\right]-\mathbb{E}\left[Z\right]|<\epsilon'$ and $\mathbb{E}\left[Z\right]=0$, we have $-\epsilon'<\mathbb{E}\left[Z|M=\tilde{m}\right]<\epsilon'\ \forall\tilde{m}\in\mathcal{M}$. Now observe that,

$$\Pr(Z \geq \epsilon' + \tilde{\epsilon} | M = \tilde{m}) \overset{\text{(a)}}{\leq} \Pr(Z \geq \mathbb{E}\left[Z | M = \tilde{m}\right] + \tilde{\epsilon} | M = \tilde{m}) \tag{18}$$

$$\leq \exp\left(\frac{-k\tilde{\epsilon}^2}{2(\gamma + \gamma_m)^2\sigma^2}\right). \tag{19}$$

Here, (a) holds since $\mathbb{E}\left[Z|M = \tilde{m}\right] < \epsilon'$. The event on the left is a subset of that on the right. Therefore, the probability of the event $\{Z \geq \epsilon' + \tilde{\epsilon}\}$ occurring cannot be more than the probability of the event $\{Z \geq \mathbb{E}\left[Z|M = \tilde{m}\right] + \tilde{\epsilon}\}$ occurring.

$$\Pr(Z \geq \epsilon' + \tilde{\epsilon}) \overset{\text{(b)}}{=} \sum_{i=1}^{n} \Pr(Z \geq \tilde{\epsilon} + \epsilon | M = \tilde{m}_i) \Pr(M = \tilde{m}_i) \tag{20}$$

$$\overset{\text{(c)}}{\leq} \exp\left(\frac{-k\tilde{\epsilon}^2}{2(\gamma_m + \gamma)^2\sigma^2}\right) \sum_{i=1}^{n} \Pr(M = \tilde{m}_i) \tag{21}$$

$$= \exp\left(\frac{-k\tilde{\epsilon}^2}{2(\gamma_m + \gamma)^2\sigma^2}\right) \tag{22}$$

$$\overset{\text{(d)}}{\leq} \exp\left(\frac{-k(\tilde{\epsilon} + \epsilon')^2}{8(\gamma_m + \gamma)^2\sigma^2}\right) \tag{23}$$

Here, (b) holds from the law of total probability. Next, (c) follows from bound in (18). Finally, (d) holds from the inequality $4\tilde{\epsilon}^2 > (\tilde{\epsilon} + \epsilon')^2$ which holds for $\tilde{\epsilon} > \epsilon' > 0$. Setting $\epsilon = \tilde{\epsilon} + \epsilon'$ completes the proof of Lemma 2

C.1. Proof of Theorem 1

Theorem 1 (Probabilistic Guarantee)**.** *Let $X_1, X_2, \ldots, X_k$ be k iid random variables with distribution $\mathcal{N}(x, \sigma^2 I_d)$ and $Z = \frac{1}{k}\sum_{i=1}^{k}(m(X_i) - M(X_i))$. Suppose $|\mathbb{E}\left[Z|M\right] - \mathbb{E}\left[Z\right]| < \epsilon'$. Then, for any $\epsilon > 2\epsilon'$, a counterfactual $x \in \mathcal{X}$ under naturally-occurring model change satisfies:*

$$\Pr(M(x) \geq R_{k,\sigma^2}(x, m) - \epsilon) \geq 1 - \exp\left(\frac{-k\epsilon^2}{8(\gamma_m + \gamma)^2\sigma^2}\right).$$

The probability is over the randomness of both M and X_i's.

Proof. The Lipschitz property of $M(\cdot)$ around x is given by,

$$|M(x) - M(x')| \leq \gamma\|x - x'\| \text{ for all } x, x' \in \mathbb{R}^d$$

Therefore,

$$M(x) \geq M(X_i) - \gamma\|x - X_i\| \tag{24}$$

$$M(x) \overset{(a)}{\geq} \frac{1}{k}\sum_{i=1}^{k}(M(X_i) - \gamma\|x - X_i\|) \tag{25}$$

where (a) holds from taking the average of the inequality (24) over all i from 1 to k.
From Lemma 2, for $X_1, X_2, \ldots, X_k \sim \mathcal{N}(x, \sigma^2 I_d)$,

$$\frac{1}{k}\sum_{i=1}^{k} M(X_i) \geq \frac{1}{k}\sum_{i=1}^{k} m(X_i) - \epsilon \tag{26}$$

with probability at least $1 - \exp\left(-\frac{k\epsilon^2}{8(\gamma+\gamma_m)^2\sigma^2}\right)$.

Hence, plugging (26) into (25), we have:

$$\Pr\left(M(x) \geq \frac{1}{k}\sum_{i=1}^{k}(m(X_i) - \gamma\|x - X_i\|) - \epsilon\right) \geq 1 - \exp\left(\frac{-k\epsilon^2}{8(\gamma + \gamma_m)^2\sigma^2}\right).$$

Recall from Definition 6, stability $R_{k,\sigma^2}(x,m) = \frac{1}{k}\sum_{i=1}^{k}(m(X_i) - \gamma\|x - X_i\|)$. Hence, we have:

$$\Pr(M(x) \geq R_{k,\sigma^2}(x,m) - \epsilon) \geq 1 - \exp\left(\frac{-k\epsilon^2}{8(\gamma+\gamma_m)^2\sigma^2}\right).$$

□

D. Appendix to Experiments in Section 5

D.1. Datasets

HELOC. The FICO HELOC (FICO, 2018) dataset contains anonymized information about home equity line of credit applications made by homeowners in the US, with a binary response indicating whether or not the applicant has ever been more than 90 days delinquent for a payment. It can be used to train a machine learning model to predict whether the homeowner qualifies for a line of credit or not. The dataset consists of 10459 rows and 40 features, which we have normalized to be between zero and one.

German Credit. The German Credit Dataset (Dua & Graff, 2017) comprises 1000 entries, each representing an individual who has taken credit from a bank. These entries are characterized by 20 categorical features, which are used to classify each person as a good or bad credit risk. To prepare the dataset for analysis, we one-hot encoded the data and normalized it such that all features fall between 0 and 1. Additionally, we partitioned the dataset into a training set and a test set, with a 70:30 ratio respectively.

CTG. The CTG dataset (Dua & Graff, 2017) consists of 2126 fetal cardiotocograms, which have been evaluated and categorized by experienced obstetricians into three categories: healthy, suspect, and pathological. We process this dataset based on Black et al. (2021). The problem was transformed into a binary classification task, where healthy fetuses are distinguished from the other two categories. We divided the dataset into a training set of 1,700 instances and a validation set of 425 instances. Each instance is described by 21 features, which we normalized to have values between zero and one.

Adult. The Adult dataset (Dua & Graff, 2017) is a publicly available dataset in the UCI repository based on 1994 U.S. census data. This dataset is a classification task to successfully predict whether an individual earns more or less than $50,000$ per year based on features such as occupation, marital status, and education, etc. The dataset consists of approximately 48,842 instances, split into a training set of 32,561 instances and a test set of 16,281 instances.

Taiwanese Credit. The Taiwanese dataset (Yeh & hui Lien, 2009) consists of 30,000 instances with 24 features that include individuals' financial data, with a binary response indicating their creditworthiness. We use one-hot encoding on the data and normalize it to be between zero and one. This dataset is processed based on Black et al. (2021). We partition the data into a training set of 22,500 and a test set of 7,500.

D.2. Model Architecture

We first trained a base neural network model for which we aim to find counterfactuals for instances with false negative predictions. The architecture of our base model consisted of two hidden layers, each containing 128 hidden units. We employed the rectified linear unit (ReLU) activation function and Adam optimizer. The model was trained for 50 epochs using a batch size of 32. We employed this model architecture and training setup on all datasets since it yielded a satisfactory level of accuracy on all of them. To evaluate the robustness of these counterfactual examples, we then trained 50 new models and assessed the validity of the counterfactuals under various model change scenarios. All 50 models had the same architecture and training setup as the base model except for some slight changes which include: Weight Initialization (WI): Retraining new models with the same hyperparameters but different weight initialization by using different random seeds for each model. Leave Out (LO): Retraining new models by randomly removing a small portion (1%) of the training data each time and using different weight initialization.

D.3. Implementation Details

The stability measure has the number of samples k and the variance σ^2 as hyperparameters. Our theoretical findings suggest that a higher value of k improves the robustness of the counterfactuals but at the expense of increased cost and computational complexity. In our experiments, we found that a value of $k = 1000$ was sufficient. The value of σ^2 was determined by analyzing the variance of the features in the dataset, and we found that a value of $\sigma^2 = 0.01$ produced good results for the

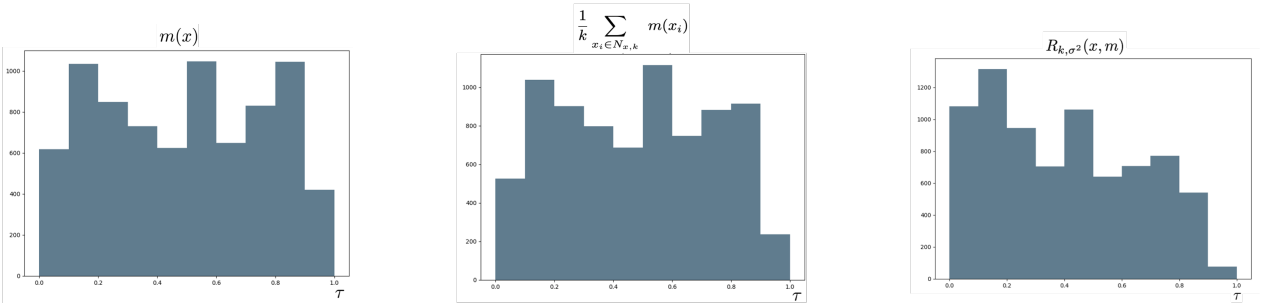


Figure 4. Histograms on the HELOC dataset to visualize the proposed stability measure.

features that lie between $[0, 1]$. These hyperparameters were kept constant across all our experiments and datasets. The baseline technique for generating counterfactuals used in T-rex was the min Cost counterfactual (Wachter et al., 2017). For other algorithm parameters, a step size of 0.01 was fixed for all datasets and experiments. The maximum number of iterations (max_steps) varied depending on the dataset, with 50 for HELOC, 200 for German Credit, and 100 for CTG, Adult, and Taiwanese. An appropriate value of τ balances the trade-off between validity and cost. We choose a value of τ to guarantee high validity and compare cost and LOF with the baselines. Another method of choosing τ is to use the histogram of $R(\cdot)$ on the training dataset (e.g., see Fig. 4 for HELOC dataset). To implement ROAR and SNS, we adhered to techniques and algorithm parameters discussed in the original works (Upadhyay et al., 2021; Black et al., 2021). In NN and T-Rex:NN implementation, a crucial consideration is determining the appropriate number of neighbors K, to search for a robust counterfactual. For larger datasets such as HELOC, Adult, and Taiwanese datasets $K = 1000$, while for the German credit and CTG datasets $K = 100$. Note that if K is too small, a counterfactual robustness test might not be satisfied, and hence T-Rex:NN returns no robust counterfactuals.

D.4. Additional Experimental Results

Table 2. Experimental results for HELOC dataset with standard deviations

HELOC	l_1 *based*				l_2 *based*			
	COST	LOF	WI VAL.	LO VAL.	COST	LOF	WI VAL.	LO VAL.
min Cost	0.40±0.48	0.49±0.93	38.8±2.5%	35.2±2.5%	0.11±0.08	0.75±0.61	13.5±3.4%	13.5±3.5%
+T-Rex:I	1.02±0.41	0.38±0.95	98.2±1.2%	98.1±1.2%	0.29±0.07	0.68±0.64	98.5±1.9%	98.2±2.0%
+SNS	1.20±0.46	0.30±0.46	98.0±1.2%	97.8±1.1%	0.31±0.08	0.64±0.78	97.9±0.8%	97.0±0.8%
ROAR	1.69±1.59	0.41±0.73	92.6±3.9%	91.2±4.3%	1.91±2.22	0.43±0.83	86.3±0.2%	84.8±0.3%
NN	1.91±2.27	0.80±2.27	51.1±12.6%	50.3±12.4%	0.56±0.59	0.80±2.27	51.1±12.6%	50.3±12.4%
T-Rex:NN	2.50±1.83	0.92±0.56	84.0±0.8%	84.0±0.8%	0.77±0.61	0.92±0.56	84.0±0.8%	84.0±0.8%

Table 3. Experimental results for German Credit dataset with standard deviations.

GERMAN	l_1 *based*				l_2 *based*			
	COST	LOF	WI VAL.	LO VAL.	COST	LOF	WI VAL.	LO VAL.
min Cost	1.42±4.16	0.77±0.65	58.8±7.9%	56.7±7.3%	0.48±1.38	0.81±0.71	26.6±15%	26.6±14%
+T-Rex:I	4.81±3.93	0.72±0.73	98.0±4.6%	96.5±3.8%	1.20±1.37	0.75±0.71	99.2±0.9%	98.7±0.9%
+SNS	5.71±4.10	0.67±0.39	97.5±0.3%	98.1±0.2%	1.44±1.38	0.68±0.73	99.9±0.0%	98.9±0.1%
ROAR	7.63±4.00	0.54±0.67	96.3±0.2%	92.3±0.3%	6.81±1.00	0.58±0.98	87.8±1.4%	85.2±1.6%
NN	7.05±3.96	1.00±0.00	95.3±0.9%	95.4±1.0%	2.50±1.20	1.00±0.00	95.3±0.9	95.3±1.0%
T-Rex:NN	10.13±4.10	1.00±0.00	100±0.0%	100±0.0%	3.04±1.45	1.00±0.00	100±0.0%	100±0.0%

Table 4. Experimental results for CTG dataset with standard deviations.

CTG	l_1 *based*				l_2 *based*			
	COST	LOF	WI VAL.	LO VAL.	COST	LOF	WI VAL.	LO VAL.
min Cost	0.21±0.18	0.94±0.91	74.6±0.1%	70.2±0.2%	0.08±0.04	1.00±0.00	19.7±30%	14.1±31%
+T-Rex:I	1.11±0.11	0.83±0.91	100±0.0%	98.8±0.1%	0.42±0.04	0.94±0.63	100±0.0%	99.7±0.1%
+SNS	3.34±0.18	−1.00±0.0	100±0.0%	98.2±0.1%	1.07±0.04	−1.00±0.0	100±0.0%	99.3±0.1%
ROAR	3.68±3.48	0.64±0.78	98.7±0.5%	96.4±0.3%	1.35±2.01	0.59±0.90	98.8±0.0%	97.2±0.0%
NN	0.39±0.23	1.00±0.00	70.5±0.2%	67.5±0.1%	0.15±0.07	1.00±0.00	70.5±0.2%	67.5±0.1%
T-Rex:NN	2.22±0.12	−0.33±0.67	100±0.0%	100±0.0%	1.00±0.80	−0.33±0.67	100±0.0%	100±0.0%

Table 5. Experimental results for Taiwanese Credit dataset with standard deviations.

TAIWANESE	l_1 *based*				l_2 *based*			
	COST	LOF	WI VAL.	LO VAL.	COST	LOF	WI VAL.	LO VAL.
min Cost	3.95±3.42	−0.37±0.92	38.4±6.0%	38.4±5.9%	2.84±1.16	−0.68±0.72	21.1±2.1%	20.0±2.2%
+T-Rex:I	6.34±3.26	0.48±0.67	96.9±0.7%	96.2±0.7%	3.06±1.11	0.40±0.31	96.8±2.1%	96.4±1.9%
+SNS	6.51±3.37	0.39±0.39	97.2±1.3%	96.9±1.4%	3.10±1.16	0.43±0.62	96.7±3.1%	96.1±2.7%
NN	5.80±3.82	0.84±0.76	53.5±9.9%	51.6±8.9%	2.18±1.27	0.84±0.76	53.5±9.9%	51.6±8.9%
T-Rex:NN	6.89±4.51	0.86±0.61	98.8±0.9%	98.4±0.8%	3.54±1.48	0.86±0.61	98.8±0.9%	98.4±0.8%

Table 6. Experimental results for Adult dataset with standard deviations.

ADULT	l_1 *based*				l_2 *based*			
	COST	LOF	WI VAL.	LO VAL.	COST	LOF	WI VAL.	LO VAL.
min Cost	0.12±0.54	0.09±0.99	80.8±5.0%	78.2±4.7%	0.18±0.65	0.09±0.99	50.0±8.0%	49.2±8.2%
+T-Rex:I	0.51±0.53	0.08±0.99	98.4±1.6%	98.1±1.3%	0.24±0.64	0.09±0.99	98.2±0.0%	98.2±0.0%
+SNS	0.92±0.58	−0.2±0.21	98.6±0.0%	97.9±0.1%	0.37±0.62	−0.22±0.97	97.9±0.0%	97.8±0.0%
NN	2.16±1.43	0.91±0.03	81.0±2.8%	81.0±2.7%	1.21±0.65	0.91±0.03	81.0±2.8%	81.0±2.7%
T-Rex:NN	3.25±1.6	0.85±0.02	99.2±0.0%	99.0±0.0%	1.59±0.57	0.85±0.02	99.2±0.0%	99.0±0.0%

Table 7. Ablation study on German Credit Dataset.

		l_1 based			l_2 based		
τ	Measure	COST	LOF	WI VAL.(%)	COST	LOF	WI VAL.(%)
0.5	$r(x,m)$	1.12	0.78	57.0	0.54	0.81	32.6
	$r_{k,\sigma^2}(x,m)$	2.06	0.81	57.4	0.57	0.80	37.3
	$R_{k,\sigma^2}(x,m)$	2.33	0.80	66.9	0.65	0.80	52.0
0.6	$r(x,m)$	1.48	0.77	61.7	0.56	0.80	38.9
	$r_{k,\sigma^2}(x,m)$	2.11	0.77	62.5	0.58	0.79	43.0
	$R_{k,\sigma^2}(x,m)$	2.40	0.76	72.9	0.68	0.81	61.7
0.7	$r(x,m)$	1.73	0.71	61.3	0.63	0.87	42.8
	$r_{k,\sigma^2}(x,m)$	2.61	0.75	63.7	0.67	0.87	51.8
	$R_{k,\sigma^2}(x,m)$	2.90	0.76	72.6	0.76	0.88	67.0
0.8	$r(x,m)$	1.69	0.76	74.4	0.68	0.85	61.0
	$r_{k,\sigma^2}(x,m)$	2.50	0.79	79.0	0.74	0.85	73.3
	$R_{k,\sigma^2}(x,m)$	2.77	0.77	86.0	0.82	0.84	84.2
0.9	$r(x,m)$	2.24	0.73	79.5	0.76	0.81	71.0
	$r_{k,\sigma^2}(x,m)$	3.01	0.74	84.8	0.83	0.79	82.7
	$R_{k,\sigma^2}(x,m)$	3.30	0.74	89.6	0.91	0.78	89.0