On Minimizing the Impact of Dataset Shifts on Actionable Explanations

Anna P. Meyer[*1] **Dan Ley**[*2] **Suraj Srinivas**[2] **Himabindu Lakkaraju**[2]

[1]Dept. of Computer Sciences, University of Wisconsin - Madison, USA
[2] Harvard University, Boston, MA, USA

Abstract

The Right to Explanation is an important regulatory principle that allows individuals to request actionable explanations for algorithmic decisions. However, several technical challenges arise when providing such actionable explanations in practice. For instance, models are periodically retrained to handle dataset shifts. This process may invalidate some of the previously prescribed explanations, thus rendering them unactionable. But, it is unclear if and when such invalidations occur, and what factors determine explanation stability i.e., if an explanation remains unchanged amidst model retraining due to dataset shifts. In this paper, we address the aforementioned gaps and provide one of the first theoretical and empirical characterizations of the factors influencing explanation stability. To this end, we conduct rigorous theoretical analysis to demonstrate that model curvature, weight decay parameters while training, and the magnitude of the dataset shift are key factors that determine the extent of explanation (in)stability. Extensive experimentation with real-world datasets not only validates our theoretical results, but also demonstrates that the aforementioned factors dramatically impact the stability of explanations produced by various state-of-the-art methods.

1 INTRODUCTION

Machine learning (ML) models have recently witnessed increased utility in critical real-world applications. This, in turn, led to the introduction of several regulatory principles that aim to safeguard the practice of algorithmic decision making in such settings [13, 32]. The Right to Explanation is one such important regulatory principle that allows individuals to request actionable explanations for algorithmic decisions that adversely impact them. Post-hoc explanation methods such as LIME, SHAP, input gradients, and Smoothgrad [27, 37, 40, 42] have commonly been employed in practice to operationalize this principle. These methods explain complex model predictions by assigning importance scores to input features, typically via learning local linear approximations of the underlying functions [18].

Providing actionable explanations to end-users is often hindered by operational challenges in practice. For instance, models are periodically retrained to handle dataset shifts, and the original explanations may no longer be valid under the new model. If so, the original explanations do not remain actionable. For example, a user may be informed that their low salary was the primary reason for the rejection of their loan application, thus prompting them to increase their income. However, an update to the model could result in a shift of model internals such that the user's credit score, and not their income, is now the primary factor for the rejection. In this case, the user's action of increasing their income would be less likely to result in a positive outcome once the updated model was deployed.

The aforementioned scenario could have been avoided if the underlying model and its explanations remained (relatively) unchanged after model retraining due to dataset shifts. Such *explanation stability* implies actionability because if explanations remain (relatively) unchanged despite model retraining, they are likely to be actionable over an extended period of time. Therefore, it is necessary to develop a systematic understanding of explanation stability in the face of dataset shifts, characterize the conditions that lead to unstable explanations, and find ways to mitigate explanation instability after retraining. Despite its significance, there is little to no research on characterizing the factors that influence explanation stability.

Our work addresses the aforementioned critical gaps by providing one of the first theoretical and empirical characterizations of the factors influencing explanation stability

[*]Equal Contribution

Accepted for the 39th Conference on Uncertainty in Artificial Intelligence (UAI 2023).

after dataset shifts. To this end, we conduct rigorous theoretical analysis to demonstrate that model curvature, weight decay parameters while training, and the magnitude of the dataset shift are key factors that determine the extent of explanation (in)stability. Our theoretical analysis emphasizes how seemingly inconsequential modeling decisions that may not impact predictive accuracy can heavily influence other critical aspects such as explanation stability. We also conduct extensive experimentation with real-world datasets to validate our theoretical insights with various state-of-the-art explanation methods (e.g., gradient and perturbation based methods such as LIME, SHAP, SmoothGrad etc.) and explanation stability metrics (e.g., ℓ_2, top-k consistency). We also empirically analyze how other training decisions (e.g., learning rate and number of training epochs) impact explanation stability. In summary, our work makes the following key contributions:

1. We provide one of the first theoretical and empirical characterizations of the factors influencing explanation stability in the face of dataset shifts.
2. We conduct a rigorous theoretical analysis to demonstrate that model curvature, weight decay parameters during training, and the magnitude of the dataset shift are key factors influencing the extent of explanation (in)stability (§3).
3. We carry out extensive experimentation with multiple real-world datasets to validate our theoretical results (§4.1). Our empirical findings suggest that standard neural network training pipelines exhibit low explanation stability, and confirm that our theoretical results are valid even when some of the underlying assumptions do not hold in practice (§4.2).
4. We empirically analyze the impact of other training hyperparameters such as learning rate, batch size, and number of training epochs on explanation stability (§4.3).

2 RELATED WORK

Explanation Methods A variety of post-hoc techniques have been proposed to explain complex models [10, 23, 38]. These techniques differ in their access to the complex model (i.e., black box vs. access to internals), scope of approximation (e.g., global vs. local), search technique (e.g., perturbation-based vs. gradient-based), explanation families (e.g., linear vs. non-linear), etc. For instance, LIME [37] and SHAP [27] are *perturbation-based*, *local explanation* approaches that learn a linear model locally around each prediction. Other *local explanation* methods capture feature importances by computing the gradient with respect to the input [39, 40, 42, 44]. Counterfactual explanation methods, on the other hand, capture the changes that need to be made to a given instance in order to flip its prediction [21, 22, 26, 34, 46, 47, 48]. In this work, we focus on analyzing the stability of explanations output by perturbation-based and gradient-based local explanation methods.

Explanation Stability and Robustness Various notions of explanation stability and robustness have been suggested in the literature.[1] One line of work shows that explanations are not robust by crafting adversarial manipulations of test inputs that change their explanations but not predictions [8, 17]. Another set of approaches studies explanation stability by altering the underlying model to maintain accuracy, but change explanations to a desired target [1, 20, 41]. Multiple works suggest using smooth or low-curvature models to improve both model and explanation stability [8, 43]. However, all of these works focus on adversarial manipulations of either the model or the input, while we are interested in more realistic model shifts occurring due to natural dataset shifts. In contrast, other works have proposed algorithmic techniques to generate counterfactual explanations or algorithmic recourses that are stable under model shifts [6, 12, 16, 30, 31, 36, 45]. However, these works differ from ours in three ways: first, we focus on feature attribution-based explanations that highlight influential features, while the existing work focuses on counterfactual explanations or algorithmic recourse. Second, we adapt the model to yield more stable explanations after retraining due to dataset shifts, whereas most of the existing work aims to adapt the explanation method. And finally, we theoretically characterize the factors that impact explanation stability.

More generally, neural networks are known to be non-robust to small training modifications, e.g., different random initializations or alternate model selection due to underspecification [2, 7, 25, 28]. However, relatively few works have studied how *explanations* change in the same setting. These works only show empirically that explanations are not stable with respect to underspecification [5] or random model initializations [3]. These works are orthogonal to ours, as we focus on theoretically and empirically characterizing specific properties of models and datasets that impact explanation stability.

Algorithmic Stability Algorithmic stability is a classical learning-theoretic framework that characterizes the consistency of outputs of learning algorithms when trained on different but similarly distributed data [4]. In contrast, our study does not make the assumption that the shifted data is similarly distributed. [19] studied the hypothesis stability of stochastic gradient descent (SGD) and showed that practical choices such as learning rate and number of epochs of training can impact the stability of model parameters. Recent work has applied the algorithmic stability perspective to

[1] The literature is inconsistent on the distinction between explanation robustness and stability; for clarity we use "robustness" when the model is fixed (e.g., robustness to input perturbations) and "stability" when the model is changing.

post-hoc explanations [14] by requiring that feature attributions be stable to a re-sampling of the underlying dataset. However, none of these prior works study realistic model shifts occurring due to natural dataset shifts or characterize the factors influencing explanation stability, which is the main focus of our work.

3 THEORETICAL ANALYSIS

In this section, we provide an analytical characterization of explanation shifts $e_1 \rightarrow e_2$ and model shifts $f_1 \rightarrow f_2$ that happen due to underlying dataset shifts $\mathcal{D}_1 \rightarrow \mathcal{D}_2$. Concretely, we address the following question:

How much do the explanations for a model f_1 trained on dataset $\mathcal{D}_1$ change when fine-tuning[2]on a slightly shifted dataset $\mathcal{D}_2$, resulting in a new model f_2?

Let the dataset $\mathcal{D}_1$ be composed of input-output pairs $(\mathbf{x}, y)$, where $\mathbf{x} \in \mathbb{R}^d$, and y is either a real value or a one-hot vector depending on the task (regression or classification). For simplicity of analysis, in this section we shall focus on gradient explanations $e = \nabla_{\mathbf{x}} f \in \mathbb{R}^d$, where f is a scalar-valued neural network function $f(\cdot, \theta) : \mathbb{R}^d \rightarrow \mathbb{R}$ mapping D-dimensional inputs to scalar outputs. Let $\ell(\cdot, \theta) : \mathbb{R}^d \rightarrow \mathbb{R}$ denote f composed with a loss function such as cross-entropy loss. Here, θ represents the parameters of the function being minimized, e.g., a deep neural network.

Our goal here is to relate the explanation shift to the dataset shift (i.e., $d(\mathcal{D}_1, \mathcal{D}_2)$, for some distance measure d) and the learning algorithm. To this end, our overall strategy is:

1. In §3.1, we bound the *parameter shift* $\|\theta_2 - \theta_1\|_2$ in terms of the *dataset shift* $d(\mathcal{D}_1, \mathcal{D}_2)$.
2. In §3.2, we bound the *explanation shift* $\|\nabla_{\mathbf{x}} f(\mathbf{x}, \theta_2) - \nabla_{\mathbf{x}} f(\mathbf{x}, \theta_1)\|_2$ in terms of the *parameter shift* $\|\theta_2 - \theta_1\|_2$ derived above.

While the existence of these shifts are qualitatively evident, we make novel contributions in that (a) we quantify the parameter and gradient changes and (b) we show that these changes are affected by factors in the modelling process.

3.1 BOUNDING THE PARAMETER SHIFT

We begin by defining the distance between datasets $d(\mathcal{D}_1, \mathcal{D}_2)$ using Hungarian distance, as defined in Defn. 1. The idea behind the Hungarian distance is simple: we find the "alignment" between data points of two datasets that minimizes the average ℓ_2 distance between a point in $\mathcal{D}_1$ and its counterpart in $\mathcal{D}_2$.

[2]Here, fine-tuning refers to initializing model parameters of f_2 with those of f_1, and then training f_2 using D_2 until its loss converges. We provide practical details of this in §4.1.

Definition 1. *Given datasets $\mathcal{D}_1 = \{(x_i, y_i)_{i=1,\ldots,N}\}$ and $\mathcal{D}_2 = \{(x'_i, y'_i)_{i=1,\ldots,N}\}$, their* Hungarian distance *is*

$$d(\mathcal{D}_1, \mathcal{D}_2) = \min_{P(\mathcal{D}_2)} \sum_{i=1}^{N} \|x_i - x'_i\|_2$$

where the minimum is taken over $P(\mathcal{D}_2)$, all permutations on the data points of $\mathcal{D}_2$.

Using this distance metric, we derive an expression for the parameter change $\|\theta_2 - \theta_1\|_2$ in terms of the dataset distance $d(\mathcal{D}_1, \mathcal{D}_2)$. We use the following assumptions to derive this result: (1) we minimize a regularized loss ℓ_{reg} that involves weight decay, i.e, $\ell_{reg}(\theta) = \ell(\theta) + \gamma\|\theta\|_2^2$; (2) ℓ is locally quadratic; (3) the learning algorithm returns a unique minimum θ given a dataset $\mathcal{D}$.

Theorem 1. *Given the assumptions stated above, and that $\mathcal{L}_x(\theta_1)$ is the Lipschitz constant of the model with parameters θ_1, we have*

$$\|\theta_2 - \theta_1\|_2 \leq \sqrt{\frac{\mathcal{L}_x(\theta_1) d(\mathcal{D}_1, \mathcal{D}_2)}{\gamma}} + C$$

where γ is the weight decay regularization constant, and C is a small problem-dependent constant.

Proof Idea. The idea of the proof is to first estimate $\ell_{\mathcal{D}_2}(\theta_1)$, i.e., the initial loss of the model θ_1, before fine-tuning on $\mathcal{D}_2$. Assuming, for the sake of explanation, that the optimal loss value obtained by θ_2 on $\mathcal{D}_2$ is zero, we have obtained the change in loss value from $\theta_1 \rightarrow \theta_2$. To derive the change in parameters from the change in loss derived above, we use a second order Taylor series expansion (with the assumption that the loss is locally quadratic). In particular, we lower bound the Hessian with its lowest eigenvalue using the weight decay term γ. The "problem-dependent constant" C arises from deviating from the assumption that the optimal loss value is zero. The complete proof is provided in the supplementary material. □

The above theorem achieves our first goal of relating parameter change to dataset shift. Intuitively, this tells us that parameter shift depends directly on dataset shift (as expected), directly on the Lipschitz constant (which depends on the model's robustness), and inversely on the weight decay parameter.

3.2 BOUNDING THE EXPLANATION SHIFT

We now consider the problem of bounding the explanation shift $\|\nabla_{\mathbf{x}} f(\cdot, \theta_2) - \nabla_{\mathbf{x}} f(\cdot, \theta_1)\|_2$ given an estimate of the parameter shift $\|\theta_2 - \theta_1\|_2$. To this end, we first define a

quantity called the *gradient-parameter Lipschitz constant*, defined in Defn. 2. To the best of our knowledge, this quantity has not been considered in previous works.

Definition 2. *We define the "gradient-parameter Lipschitz constant" $\mathcal{L}_{\Theta,\mathcal{D}}$ w.r.t. an input distribution $\mathcal{D}$, and parameter set Θ as follows:*

$$\mathcal{L}_{\Theta,\mathcal{D}} = \mathbb{E}_{\theta\in\Theta}\mathbb{E}_{\mathbf{x}\sim\mathcal{D}}\|\nabla_\theta\nabla_x f(x;\theta)\|_2$$

Intuitively, this quantity captures the sensitivity of gradient explanations to small changes in the parameter θ. In contrast to usual definitions of Lipschitz constants, this is defined (1) locally with respect a particular distribution and parameter set, and (2) using the mean instead of the $\sup$. This allows us to derive the following relationship between average gradient difference and parameter shift.

Lemma 1. *The gradient-parameter Lipschitz has the following property:*

$$\mathbb{E}_{\mathbf{x}\sim\mathcal{D}}\|\nabla_x f(x;\theta_1) - \nabla_x f(x;\theta_2)\|_2 \leq \mathcal{L}_{\Theta,\mathcal{D}} \times \|\theta_2 - \theta_1\|_2$$

where $\Theta = \{\lambda\theta_1 + (1-\lambda)\theta_2 \mid \lambda \in [0,1]\}$.

This relation follows immediately from the fundamental theorem of integral calculus, and the full proof is given in the supplementary material. This result reveals an intuitive fact that larger parameter shifts lead to larger gradient shifts, but mediated by the gradient-parameter Lipschitz. This quantity is difficult to analyse for general neural networks, and we provide below an analysis for the case of 1-hidden layer neural networks, with a specific simplifying assumption on the data distribution.

Theorem 2. *Assume that we have a 1-hidden layer neural network with weights θ, and random inputs $\mathbf{x} \sim \mathcal{N}(0, I)$[3]. Further assume that we use an activation function σ with well-defined second derivatives (e.g: softplus). For this case, the gradient-parameter Lipschitz constant is*

$$\mathbb{E}_{\mathbf{x},\theta}\|\nabla_\theta\nabla_{\mathbf{x}} f(\mathbf{x},\theta)\|_2 \leq (\mathbb{E}_\theta\|\theta\|_2) + \beta\,(\mathbb{E}_\theta\phi(\theta))$$

where β is the maximum curvature of activation function σ, and $\phi(\theta)$ is the path-norm [29] of the model.

Proof Idea. The proof involves computing the gradient-parameter derivatives for the 1-hidden layer neural network case. Our assumption regarding smoothness of activation function ensures that $\sigma'' \leq \beta$, i.e., that the second derivative of the activation function is upper bounded, which is true for smooth non-linearities like softplus. Another simplifying assumption regarding the distribution of the inputs ($\mathbf{x} \sim \mathcal{N}(0, I)$) helps us dramatically simplify the expression for the expected value of the gradient norm terms.

The path-norm for a 1-hidden layer neural network (for weights W_2, W_1) is given by $\phi(\theta) = \sqrt{\sum_{i,j}(W_2^j W_1^{i,j})^2}$ and has been linked to model generalization [29]. The complete proof is provided in the supplementary material. □

These results achieve our goal of relating the change in gradients to parameter change. Taken together with the results in §3.1, this achieves our overall goal of relating gradient change to dataset shift. Intuitively, this tells us the following regarding gradient shift:

1. It depends directly on the *dataset distance* $d(\mathcal{D}_2, \mathcal{D}_1)$, which is as expected: the larger the dataset shift, the larger can be the parameter shift, and explanation shift.
2. It depends inversely on the *weight decay* parameter γ, and directly on the norm of weights $\|\theta\|_2$. Intuitively, large weight decay shifts the optima to be closer to zero, reducing the norm of weights, which makes all optima closer to each other.
3. It depends directly on the *Lipschitz constant* of the first model, which is connected to the model's robustness. Intuitively, if the model is already robust to small input changes, we don't expect it to shift too much upon fine-tuning on slightly shifted data.
4. It depends directly on the *smoothness constant* β of the activation function used in the model. The smoother the model, the smaller are its second derivatives, and thus its gradient-input Lipschitz.

These insights motivate the core thesis of the paper, i.e., that explanation stability depends on properties beyond predictive performance. Specifically, we see that specific algorithm choices that do not necessarily improve accuracy, such as weight decay, smoothness constant and robustness-inducing losses, affect explanation and model stability to data shifts.

We conclude by making two remarks regarding the theory. First, it only highlights *sufficient conditions* for explanation stability, not necessary conditions. Meaning, there can be ways to achieve gradient stability without the parameters θ_1, θ_2 being close, or without neural nets being Lipschitz, or when ReLU or weight decay are not used, but these lie outside the scope of this theory. Second, the focus of these bounds is not to produce numerically tight estimates for explanation stability, rather, it is intended as a tool to make *qualitative predictions* regarding specific modelling interventions that can improve explanation stability in practice. Section 4 explores this in more detail.

[3] Covariance of I is chosen for notational brevity

4 EXPERIMENTAL EVALUATION

In this section, we describe our general evaluation setup. In §4.1, we experimentally validate that the modeling interventions identified in §3 improve explanation stability. Next, in §4.2, we show that these relationships extend to realistic training setups and more-sophisticated explanation techniques (i.e., top-K feature attributions of techniques like SmoothGrad, LIME, and SHAP). Finally, in §4.3, we show that other hyperparameters not discussed in our theory – e.g., learning rate, learning rate decay, batch size, and number of training epochs – also impact gradient stability.

Datasets We evaluate explanation stability on three binary datasets with continuous features. We use the WHO life expectancy dataset (n=2928, number of features d=18) [35], which evaluates whether a country's life expectancy is above the global median based on health and economic factors; the HELOC dataset (n=9871, d=23) [15], which evaluates loan acceptance given the applicant's financial information; and the Adult Income dataset (n=32561, d=6) [11], which evaluates whether a person's income is above $50,000. We model dataset shifts in two ways:

- *Synthetic Noise.* We add Gaussian $\mathcal{N}(0, \sigma^2)$ noise to all training samples, with varying levels of noise σ.
- *Temporal Shift.* We create a temporal shift dataset from the WHO data by using pre-2012 data as the "original" dataset and the full dataset as the "shifted" dataset.

Models We evaluate the stability of medium-sized neural networks containing 5 layers of 50 nodes. By default, we use ReLU activation; however, to see the effects of model curvature on explanation stability, we use softplus (SP) for some experiments. For the experiments using SP, we vary the parameter β from 2 to 10 depending on the dataset and the experiment's goals. Smaller values of β correspond to lower-curvature models, while SP with $\beta = \infty$ is equivalent to ReLU. Unless otherwise specified, all models we compare within each section achieve similar predictive accuracy.

Experimental procedure For each experiment, we average over 10 trials to reduce the impact that random seeds have on the outcomes. Each trial in the synthetic noise experiments consists of training one "base" model plus 10 "noisy" models, each corresponding to a different random perturbation of the data. In the temporal shift experiments, each trial compares the base model (trained on the partial dataset) to a new model (trained on the full dataset). We compute explanation stability for all test samples, except for LIME and SHAP results, where we limit computations to 100 randomly-chosen test samples. In the results, we display the mean of all trials along with the middle 50% of values.

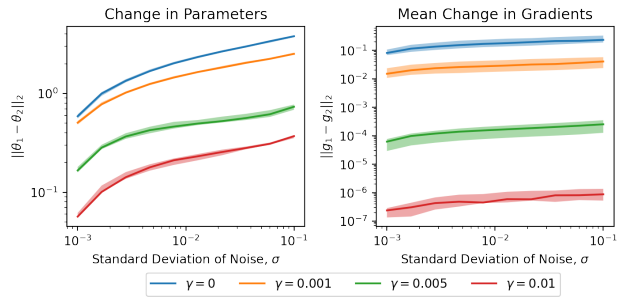


Figure 1: Effect of weight decay value on the mean gradient and parameter changes for HELOC after fine-tuning on a shifted dataset. All models use ReLU activation. The x-axis is the size of the data shift, represented by the standard deviation of noise, σ.

Table 1: Effect of weight decay (γ) on WHO (with fine-tuning).

	$\gamma = 0$	0.001	0.01
$\|\theta_1 - \theta_2\|_2$	1.69±0.09	1.53±0.03	0.95±0.02
$\|g_1 - g_2\|_2$	0.201±0.065	0.122±0.034	0.010±0.002

4.1 VALIDATING OUR THEORETICAL RESULTS

Experiment setup In this section, we test our theoretical results from §3 by evaluating the impact of *weight decay*, *model curvature*, and *size of the data shift* on explanation stability. To ensure that our analysis is sound, we adhere to the theoretical assumptions as closely as possible. In particular, we focus on *model parameters* (θ) and *gradients* (g) rather than top-K feature attributions, since top-K explanations are not differentiable. We train each base model for a large number of epochs to ensure convergence of the training loss to a minimum (i.e., we ignore overfitting on the test set). To fine-tune, we copy the parameters of the base model and then train on the shifted data. Also, we start fine-tuning with a lower learning rate and train for fewer epochs, given that the initial loss is already close to a minimum.

For brevity, we omit similar results from the Adult dataset, with complete results in the appendix. We also provide the training hyperparameters for each dataset in the appendix.

Weight decay results Figure 1 and Table 1 show the effect of weight decay for the synthetically and naturally-shifted datasets, respectively. We see that larger weight decay values correspond to orders of magnitude smaller mean gradient and parameter changes, which is in line with the theoretical relationship between weight decay and stability. As the amount of synthetic noise grows, we observe that both parameters and gradients diverge more between the base model and fine-tuned model. This pattern supports our hypotheses that (a) the loss landscape changes more with larger data shift and (b) a larger change in model parameters is correlated with a larger change in test sample gradients.

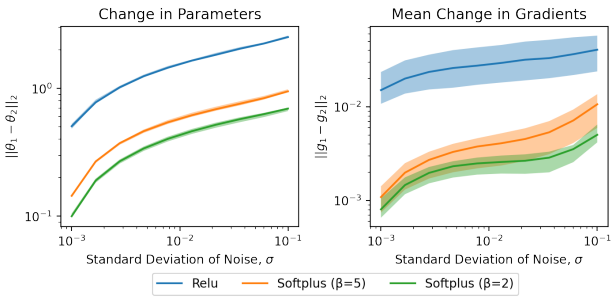


Figure 2: Effect of model curvature on the mean gradient and parameter changes for the HELOC dataset after **fine-tuning** on a shifted dataset. The x-axis is the size of the data shift, represented by the standard deviation of noise, σ.

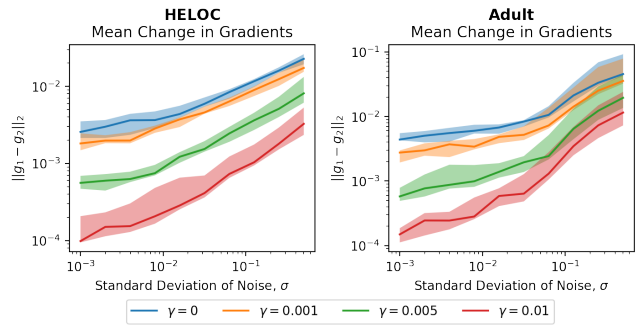


Figure 3: Effect of weight decay on parameter stability when **retraining** on HELOC and Adult datasets. The x-axis is the size of the data shift, represented by the standard deviation of noise, σ.

Table 2: Effect of curvature on WHO (with **fine-tuning**).

	ReLU	SP ($\beta = 10$)	SP ($\beta = 5$)
$\|\theta_1 - \theta_2\|_2$	1.53±0.03	1.37±0.05	1.23±0.02
$\|g_1 - g_2\|_2$	0.122±0.034	0.096±0.026	0.070±0.024

Table 3: Effect of weight decay (γ) on WHO (with **retraining**).

	$\gamma = 0$	0.001	0.01
$\|\theta_1 - \theta_2\|_2$	3.97±0.13	3.50±0.19	1.79±0.17
$\|g_1 - g_2\|_2$	0.295±0.075	0.150±0.034	0.003±0.001

Curvature results Figure 2 and Table 2 show that using low-curvature training techniques increases gradient stability. In particular, SP does better than ReLU, particularly with smaller values of β. For both the real-world and synthetic shifts, the gradient changes are less distinct (i.e., the confidence intervals overlap more) than the parameter changes, but still show a trend in the predicted direction.

4.2 GOING BEYOND THEORETICAL RESULTS

In this section, we evaluate whether our theoretical predictions still hold if we deviate from the theoretical assumptions, namely: fine-tuning, exactly reaching a local minimum, and using non-differentiable top-K feature attribution metrics rather than gradients ℓ_2 distance. The top-K features for a model's predictions are the k features with attribution values (e.g., gradients or functions of gradients for gradient-based methods; weights of linear models for LIME and SHAP) of highest magnitude. For example, if an instance $\mathbf{x}$ comprises of four features indexed by $[1 \cdots 4]$, and the gradient (feature attribution) of a given model w.r.t. $\mathbf{x}$ is $[-0.1, -0.4, 0.2, 0.3]$, then the top-2 features are features 2 and 4 since their corresponding feature attribution magnitudes -0.4 and 0.3 are the highest.

Experiment setup We maintain our experimental setup from §4.1, except that we train each model for a smaller number of epochs (until accuracy stabilizes), before retraining a model from scratch on the shifted data, rather than fine-tuning (hyperparameters are located in the appendix). We use the same random network initialization for both training and retraining, as prior work has shown that feature attributions are not stable to changes in random seed [3].

To complete the analysis, we consider four feature attribution techniques: input gradients, input gradients with SmoothGrad, LIME, and kernel SHAP [27, 37, 42, 40]. We use three top-K stability metrics adapted from Brunet et al. [5]. They each take in a pair of top-K feature sets corresponding to two models. Sign Agreement (SA) returns the fraction of top-K features that appear in both models' top-K features and have the same signed value. Consistent Direction of Contribution (CDC) is binary (per-sample) and measures whether all features in the top-K (for either model) have the same signed value in the other model. Signed-Set Agreement (SSA) is also binary (per-sample) and is 1 if the two model's top-K feature sets contain the same features and the features have the same signed value (the order of the top-K features does not matter).

Relaxing Training Assumptions From Table 3, we see that increasing the algorithm's weight decay significantly decreases both parameters and gradients. Figure 3 confirms that a larger weight decay during training also increases gradient stability for models with synthetic noise (analogous results for parameter stability are included in the appendix). Figure 4 and Table 4 show that the curvature trends discovered in §4.1 still hold, i.e., lower model curvature induces higher gradient stability.

Although curvature and weight decay increase explanation stability upon retraining, we note parameter stability scores are much lower for retraining than for fine-tuning given fixed curvatures and weight decays. For example, the mean change in parameters when using ReLU with a weight decay of 0.001 is 1.53±0.03 when fine-tuning, versus 3.50±0.19 when retraining. The difference in gradient change between the two techniques is less stark, however, suggesting that

Table 4: Effect of curvature on WHO (with **retraining**).

	ReLU	SP ($\beta = 10$)	SP ($\beta = 5$)
$\|\theta_1 - \theta_2\|_2$	3.50±0.19	2.60±0.05	1.97±0.12
$\|g_1 - g_2\|_2$	0.150±0.034	0.102±0.040	0.076±0.024

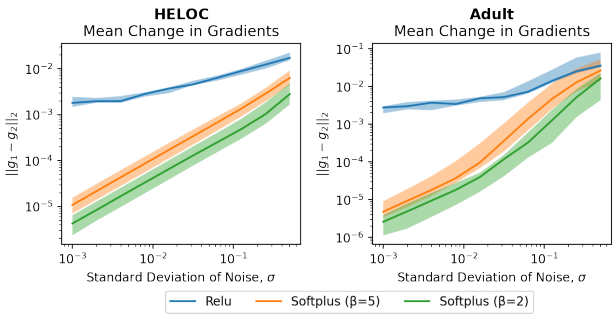


Figure 4: Effect of model curvature on gradient stability when **retraining** on HELOC and Adult datasets. The x-axis is the size of the data shift, represented by the standard deviation of noise, σ.

even though the fine-tuned models are more similar to each other overall, the retrained models are sufficiently similar on the parts of the input domain represented in the test set.

Using Diverse Explanation Techniques Figure 5 shows Top-5 SA stability for two popular explanation techniques, namely, Saliency (input gradients) and SmoothGrad [24]. As in the previous experiments, models with low curvature and large weight decay are more effective at preserving explanation stability. In case of saliency, the explanation stability starts to decrease sharply for shift magnitudes larger than $\mathcal{N}(0, 0.1^2)$. For SmoothGrad, the noise has less of an effect, which we hypothesize is because SmoothGrad is more stable overall as it computes an averaged gradient over a small local neighborhood of a given instance (see Table 5). Lowering the model's curvature and increasing the weight decay similarly improve explanation stability under data shifts for the other metrics and explanation techniques (see the appendix for a full set of graphs).

Table 5 gives a more detailed breakdown of top-K feature attribution stability across all datasets and explanation techniques. The table corresponds to a single modeling choice – ReLU with weight decay 0.001. We notice trends based on the *explanation technique*, *explanation metric*, and *dataset*.

The different explanation techniques exhibit vastly different stabilities under dataset shifts. Notably, SmoothGrad outperforms all other explanation techniques, obtaining scores of over 90% for both Top-3 and Top-5 explanation stability for all of the metrics we used. That is, over 90% of test samples' attributions had the same signs for all of each model's top-K features (CDC), over 90% of test samples' attributions had the same signed features in the top-K (SSA), and on average, over 90% of the top-K features for each test sample appeared in the other model's top-K features (SA).

To compare the explanation metrics, we look at the relative difference between SSA and each of SA and CDC (since SSA is strictly stronger than either of the other metrics). We see that neither SA nor CDC is consistently more stable, suggesting that the failure to have identical top-K feature sets stems both from different features ranking as the most influential (i.e., what SA captures) and also whether the top features have the same sign in the other model (i.e., CDC).

We also see that different datasets exhibit different levels of explanation stability. The top-3 and top-5 scores for the Adult datasets are very high; however, this dataset only has six features, so there are fewer possible feature combinations. We see that SSA scores for Adult are much lower than the SA scores, indicating that even with only 6 features, the learnt models often disagree on different features' impacts.

4.3 SENSITIVITY ANALYSIS

In this section, we analyze the impact of other training hyperparameters on explanation stability, epoch-by-epoch throughout the entire training process.

Given our observation that parameter stability promotes explanation stability, we expect a lower number of epochs and a lower learning rate to both increase explanation stability under dataset shift by decreasing the divergence between optimization paths in parameter space. Based on the geometry of two divergent optimization paths, we anticipate that if the base model is trained for a long period of time, the retrained model's parameters that are most similar to the base model will occur relatively sooner in the retraining process. This emphasizes the challenge of retraining models that both perform well and remain close to the base model.

We track explanations during each epoch of retraining, measuring similarity using gradient ℓ_2 distance and top-K SA consistency. Each trial consists of varying a single hyperparameter while keeping all others fixed. For each value of the varied hyperparameter, we compare one base model with 10 models retrained on synthetically shifted HELOC data $\sim \mathcal{N}(0, 0.1^2)$. Recall that [19] studied the hypothesis stability of stochastic gradient descent (SGD) and showed that practical choices such as learning rate and number of epochs of training can impact the stability of model parameters; the following experiments confirm these findings in the context of explanation stability.

Learning Rate We verify in Figure 6a that increasing the learning rate of all models effectively worsens gradient stability. We observe that the maximal gradient instability occurs sooner when the learning rate is higher. We hypothesize that this is because a high learning rate stretches the optimization path of the base model, taking it further away from the initialization in parameter space. Since the

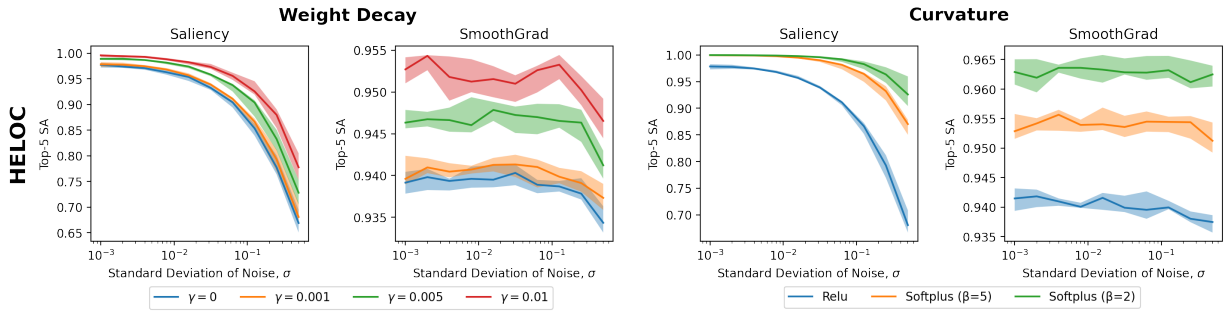


Figure 5: Top-5 SA for the HELOC dataset under model retraining. The left two graphs show the effect of weight decay and the right two graphs show the effect of curvature. Graphs 1 and 3 show Top-5 SA for saliency, while graphs 2 and 4 show Top-5 SA for SmoothGrad. Confidence intervals represent the middle 50% of values. See the appendix for other datasets and explanation techniques.

Table 5: Explanation stability scores for all datasets. The synthetically-shifted datasets (HELOC and Adult) are modified by $\mathcal{N}(0, 0.1^2)$ noise. Models use ReLU activation and weight decay $\gamma = 0.001$. All values averaged across all test samples and then across all trials. Error bounds are such that the lower and upper bound values represent the 25th and 75th percentile respectively.

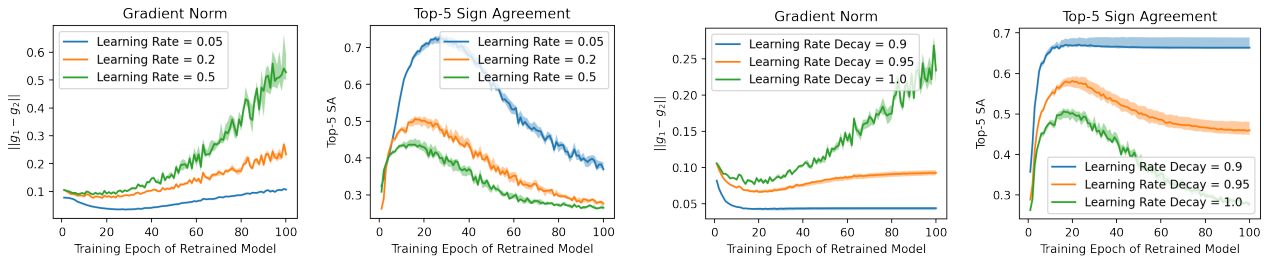
Dataset	Explanation technique	Top-3 SA	Top-3 CDC	Top-3 SSA	Top-5 SA	Top-5 CDC	Top-5 SSA
WHO	Saliency	0.63±0.01	0.83±0.03	0.18±0.03	0.61±0.03	0.61±0.10	0.03±0.01
	SmoothGrad	0.94±0.00	0.94±0.01	0.91±0.00	0.94±0.00	0.92±0.00	0.90±0.00
	LIME	0.69±0.09	0.35±0.24	0.27±0.20	0.59±0.57	0.08±0.05	0.06±0.05
	K.SHAP	0.58±0.10	0.60±0.39	0.13±0.08	0.61±0.10	0.41±0.31	0.05±0.05
HELOC	Saliency	0.51±0.02	0.88±0.03	0.07±0.01	0.55±0.02	0.71±0.05	0.01±0.00
	SmoothGrad	0.94±0.00	0.96±0.00	0.90±0.00	0.94±0.00	0.94±0.00	0.90±0.00
	LIME	0.63±0.02	0.45±0.04	0.18±0.03	0.66±0.02	0.16±0.03	0.07±0.02
	K.SHAP	0.64±0.02	0.91±0.03	0.19±0.03	0.70±0.02	0.79±0.05	0.09±0.02
Adult	Saliency	0.87±0.01	0.92±0.01	0.62±0.04	0.85±0.01	0.57±0.04	0.38±0.03
	SmoothGrad	0.98±0.00	0.98±0.00	0.96±0.00	0.98±0.00	0.96±0.00	0.94±0.00
	LIME	0.95±0.00	0.86±0.02	0.86±0.02	0.97±0.00	0.84±0.02	0.84±0.02
	K.SHAP	0.88±0.01	0.88±0.04	0.65±0.04	0.84±0.02	0.47±0.10	0.32±0.08

retrained models take divergent paths from the base model (due to the dataset shifts), the likelihood that they achieve similar parameters decreases as learning rate increases.

Learning Rate Decay Building upon the understanding of the effect of learning rate on gradient stability, we delve into its counterpart, learning rate decay, and its influence, as depicted in Figure 6b. The results are as expected: decreasing the decay value, i.e. increasing the amount of decay, shows similar behaviour to decreasing the learning rate. Namely, the optimization trajectories have less chance to diverge, and the resultant base model and retrained models are thus more likely to have similar parameters. While a higher amount of decay (lower decay value) is preferable, this can inhibit the training process (notice how a decay value of 0.9 causes training to stop after around 20 epochs).

Batch Size In Figure 6c, we see that decreasing batch size worsens gradient stability. We hypothesize that this behaviour is because there are a greater number of steps in each epoch, inducing a higher level of randomness in the optimization trajectory. Each step adds an opportunity for the retrained model's path to deviate from the initial base model's path, due to the dataset shifts. As such, the likelihood of achieving similar parameters between the retrained models and the base models decreases with smaller batch sizes. Additionally, the points at which explanation stability is maximized occur after fewer epochs for smaller batch sizes, since those correspond to both longer and more divergent optimization trajectories.

Base Model Epochs Figure 6d verifies that as the number of base model epochs is increased, explanation stability suffers, with significantly larger gradient changes and lower top-5 consistency scores. For a sufficiently small base model epoch of 20, we observe that similarity also peaks at around 20 training epochs. However, if we were to increase the base model epoch to 60, as shown in green, we observe that the peak still occurs at around the same value. This means that, after just 20 epochs, the optimization path of the retrained model must have diverged so much from the base model that any further training leads only to a decrease in stability.



(a) Sensitivity to **learning rate**.

(b) Sensitivity to **learning rate decay**.

(c) Sensitivity to **batch size**.

(d) Sensitivity to the base model's number of **epochs**.

Figure 6: Raw gradient similarity (y-axis) of retrained models are shown per epoch of training (x-axis). For gradient norm, lower values are better, while for top-5 SA, higher values (closer to 1) are better. Default values are: learning rate of 0.2; learning rate decay of 1.0 (no decay); batch size of 128; base models are trained for 30 epochs (besides the bottom right); and synthetic noise, $\sigma = 0.1$.

5 CONCLUSIONS

In this paper, we theoretically and empirically characterize how modelling choices impact model explanations' stability to retraining under data shifts. Our formal results suggest, and our experimental results confirm, that model explanations can be made more stable to retraining after data shifts by using low-curvature activation functions and large weight decay values. We also took a first step towards empirically characterizing how different explanation techniques yield variable explanation stabilities under dataset shifts, noting that SmoothGrad outperforms other methods.

The message for ML practitioners is clear: in order to preserve explanations upon model retraining due to data shifts at little cost to accuracy, use softplus with the lowest possible β, and the largest weight decay that does not affect predictive accuracy. If possible, practitioners should fine-tune the model (§4.1) rather than retrain from scratch to avoid instability due to the multiplicity of optimal solutions. However, as shown in §4.3, training hyperparameters can also play a large role in explanation stability, so care must be taken to choose hyperparameters in a way that balances accuracy and explanation stability.

There are several directions for future work. First, our theoretical results are only applicable to differentiable models. In the appendix, we empirically evaluate explanation stability when retraining decision trees, random forests, and XGBoost models, but additional work is needed to formalize explanation stability to retraining for these and other classes of models. Second, while we do perform experiments with a temporal dataset shift, we do not evaluate other types of naturally-occurring data shifts like geographic shifts or data corrections (e.g., as with the German Credit Statlog dataset [11]). Future work should focus on understanding explanation stability for various types of real-world data shifts. Third, we limit our evaluation to tabular datasets due to the commonality of these datasets where laws like a Right to Explanation are relevant. However, other sources of data may become more relevant to user explanations in the future and validating our theoretical results on non-tabular data is of independent interest. Finally, rather than sticking to existing explanation techniques, future work can explore how to develop novel explanation techniques that preserve explanation stability despite small changes in the underlying models and data.

Acknowledgements

The authors would like to thank the anonymous reviewers for their helpful feedback and all the funding agencies listed below for supporting this work. This work is supported in part by the NSF awards #IIS-2008461, #IIS-2040989, and #IIS-2238714, and research awards from Google, JP Morgan, Amazon, Harvard Data Science Initiative, and D$\hat{3}$ Institute at Harvard. The views expressed here are those of the authors and do not reflect the official policy or position of the funding agencies.

References

[1] C. J. Anders, P. Pasliev, A.-K. Dombrowski, K.-R. Müller, and P. Kessel. Fairwashing Explanations with Off-Manifold Detergent. In *Proceedings of the 37th International Conference on Machine Learning*, ICML'20. JMLR.org, 2020.

[2] E. Black and M. Fredrikson. Leave-One-out Unfairness. In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, FAccT '21, page 285–295. Association for Computing Machinery, 2021.

[3] E. Black, K. Leino, and M. Fredrikson. Selective Ensembles for Consistent Predictions. In *International Conference on Learning Representations*, 2022.

[4] O. Bousquet and A. Elisseeff. Stability and Generalization. *The Journal of Machine Learning Research*, 2:499–526, 2002.

[5] M.-E. Brunet, A. Anderson, and R. Zemel. Implications of Model Indeterminacy for Explanations of Automated Decisions. In A. H. Oh, A. Agarwal, D. Belgrave, and K. Cho, editors, *Advances in Neural Information Processing Systems*, 2022.

[6] N. Bui, D. Nguyen, and V. A. Nguyen. Counterfactual Plans under Distributional Ambiguity. In *International Conference on Learning Representations*, 2022.

[7] A. D'Amour, K. Heller, D. Moldovan, B. Adlam, B. Alipanahi, A. Beutel, C. Chen, J. Deaton, J. Eisenstein, M. D. Hoffman, F. Hormozdiari, N. Houlsby, S. Hou, G. Jerfel, A. Karthikesalingam, M. Lucic, Y. Ma, C. McLean, D. Mincu, A. Mitani, A. Montanari, Z. Nado, V. Natarajan, C. Nielson, T. F. Osborne, R. Raman, K. Ramasamy, R. Sayres, J. Schrouff, M. Seneviratne, S. Sequeira, H. Suresh, V. Veitch, M. Vladymyrov, X. Wang, K. Webster, S. Yadlowsky, T. Yun, X. Zhai, and D. Sculley. Underspecification Presents Challenges for Credibility in Modern Machine Learning. *Journal of Machine Learning Research*, 23(226):1–61, 2022.

[8] A.-K. Dombrowski, M. Alber, C. Anders, M. Ackermann, K.-R. Müller, and P. Kessel. Explanations can be manipulated and geometry is to blame. *Advances in Neural Information Processing Systems*, 32, 2019.

[9] R. Dominguez-Olmedo, A. H. Karimi, and B. Schölkopf. On the Adversarial Robustness of Causal Algorithmic Recourse. In K. Chaudhuri, S. Jegelka, L. Song, C. Szepesvari, G. Niu, and S. Sabato, editors, *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 5324–5342. PMLR, 17–23 Jul 2022.

[10] F. Doshi-Velez and B. Kim. Towards A Rigorous Science of Interpretable Machine Learning. *arXiv: Machine Learning*, 2017.

[11] D. Dua and C. Graff. UCI Machine Learning Repository, 2017.

[12] S. Dutta, J. Long, S. Mishra, C. Tilli, and D. Magazzeni. Robust Counterfactual Explanations for Tree-Based Ensembles. In *International Conference on Machine Learning*, pages 5742–5756. PMLR, 2022.

[13] European Commission. General Data Protection Regulation (GDPR).

[14] T. Fel, D. Vigouroux, R. Cadène, and T. Serre. How Good Is Your Explanation? Algorithmic Stability Measures To Assess the Quality of Explanations for Deep Neural Networks. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 720–730, January 2022.

[15] FICO. Explainable Machine Learning Challenge, 2018.

[16] A. Forel, A. Parmentier, and T. Vidal. Robust Counterfactual Explanations for Random Forests. *arXiv preprint arXiv:2205.14116*, 2022.

[17] A. Ghorbani, A. Abid, and J. Zou. Interpretation of Neural Networks is Fragile. In *Proceedings of the Thirty-Third AAAI Conference on Artificial Intelligence and Thirty-First Innovative Applications of Artificial Intelligence Conference and Ninth AAAI Symposium on Educational Advances in Artificial Intelligence*, AAAI'19/IAAI'19/EAAI'19. AAAI Press, 2019.

[18] T. Han, S. Srinivas, and H. Lakkaraju. Which Explanation Should I Choose? A Function Approximation Perspective to Characterizing Post Hoc Explanations. In A. H. Oh, A. Agarwal, D. Belgrave, and K. Cho, editors, *Advances in Neural Information Processing Systems*, 2022.

[19] M. Hardt, B. Recht, and Y. Singer. Train faster, generalize better: Stability of stochastic gradient descent. In *International conference on machine learning*, pages 1225–1234. PMLR, 2016.

[20] J. Heo, S. Joo, and T. Moon. Fooling Neural Network Interpretations via Adversarial Model Manipulation. *Advances in Neural Information Processing Systems*, 32, 2019.

[21] A.-H. Karimi, B. Schölkopf, and I. Valera. Algorithmic Recourse: From Counterfactual Explanations to Interventions. In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, FAccT '21, page 353–362, New York, NY, USA, 2021. Association for Computing Machinery.

[22] A.-H. Karimi, J. von Kügelgen, B. Schölkopf, and I. Valera. Algorithmic recourse under imperfect causal knowledge: a probabilistic approach. In H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 265–277. Curran Associates, Inc., 2020.

[23] P. W. Koh and P. Liang. Understanding Black-Box Predictions via Influence Functions. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML'17, page 1885–1894. JMLR.org, 2017.

[24] N. Kokhlikyan, V. Miglani, M. Martin, E. Wang, B. Alsallakh, J. Reynolds, A. Melnikov, N. Kliushkina, C. Araya, S. Yan, and O. Reblitz-Richardson. Captum: A unified and generic model interpretability library for PyTorch, 2020.

[25] J. F. Kolen and J. B. Pollack. Back Propagation is Sensitive to Initial Conditions. In *Proceedings of the 1990 Conference on Advances in Neural Information Processing Systems 3*, NIPS-3, page 860–867, San Francisco, CA, USA, 1990. Morgan Kaufmann Publishers Inc.

[26] A. V. Looveren and J. Klaise. Interpretable Counterfactual Explanations Guided by Prototypes. *ArXiv*, abs/1907.02584, 2019.

[27] S. M. Lundberg and S.-I. Lee. A Unified Approach to Interpreting Model Predictions. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

[28] J. Mehrer, C. J. Spoerer, N. Kriegeskorte, and T. C. Kietzmann. Individual differences among deep neural network models. *Nature communications*, 11(1):5725, November 2020.

[29] B. Neyshabur, R. Tomioka, and N. Srebro. Norm-based capacity control in neural networks. In *Conference on learning theory*, pages 1376–1401. PMLR, 2015.

[30] D. Nguyen, N. Bui, and V. A. Nguyen. Distributionally Robust Recourse Action, 2022.

[31] T.-D. H. Nguyen, N. Bui, D. Nguyen, M.-C. Yue, and V. A. Nguyen. Robust Bayesian Recourse. In *Uncertainty in Artificial Intelligence*, pages 1498–1508. PMLR, 2022.

[32] Office of Science and Technology Policy. Blueprint for an AI Bill of Rights.

[33] M. Pawelczyk, C. Agarwal, S. Joshi, S. Upadhyay, and H. Lakkaraju. Exploring Counterfactual Explanations Through the Lens of Adversarial Examples: A Theoretical and Empirical Analysis. In G. Camps-Valls, F. J. R. Ruiz, and I. Valera, editors, *Proceedings of The 25th International Conference on Artificial Intelligence and Statistics*, volume 151 of *Proceedings of Machine Learning Research*, pages 4574–4594. PMLR, 28–30 Mar 2022.

[34] R. Poyiadzi, K. Sokol, R. Santos-Rodriguez, T. De Bie, and P. Flach. FACE: Feasible and Actionable Counterfactual Explanations. In *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*, AIES '20, page 344–350, New York, NY, USA, 2020. Association for Computing Machinery.

[35] K. Rajarshi. Life Expectancy (WHO), 2017.

[36] K. Rawal, E. Kamar, and H. Lakkaraju. Algorithmic Recourse in the Wild: Understanding the Impact of Data and Model Shifts, 2020.

[37] M. T. Ribeiro, S. Singh, and C. Guestrin. "Why Should I Trust You?": Explaining the Predictions of Any Classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, page 1135–1144, New York, NY, USA, 2016. Association for Computing Machinery.

[38] M. T. Ribeiro, S. Singh, and C. Guestrin. Anchors: High-Precision Model-Agnostic Explanations. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence and Thirtieth Innovative Applications of Artificial Intelligence Conference and Eighth AAAI Symposium on Educational Advances in Artificial Intelligence*, AAAI'18/IAAI'18/EAAI'18. AAAI Press, 2018.

[39] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra. Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 618–626, 2017.

[40] K. Simonyan, A. Vedaldi, and A. Zisserman. Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps. *CoRR*, abs/1312.6034, 2013.

[41] D. Slack, S. Hilgard, E. Jia, S. Singh, and H. Lakkaraju. Fooling LIME and SHAP: Adversarial Attacks on Post hoc Explanation Methods. In *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*, pages 180–186, 2020.

[42] D. Smilkov, N. Thorat, B. Kim, F. Viegas, and M. M. Wattenberg. SmoothGrad: removing noise by adding noise. In *ICML Workshop on Visualization for Deep Learning*, Sydney, Australia, 2017.

[43] S. Srinivas, K. Matoba, H. Lakkaraju, and F. Fleuret. Efficient Training of Low-Curvature Neural Networks. In A. H. Oh, A. Agarwal, D. Belgrave, and K. Cho, editors, *Advances in Neural Information Processing Systems*, 2022.

[44] M. Sundararajan, A. Taly, and Q. Yan. Axiomatic Attribution for Deep Networks. In D. Precup and Y. W. Teh, editors, *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 3319–3328. PMLR, 06–11 Aug 2017.

[45] S. Upadhyay, S. Joshi, and H. Lakkaraju. Towards Robust and Reliable Algorithmic Recourse. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P. Liang, and J. W. Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 16926–16937. Curran Associates, Inc., 2021.

[46] B. Ustun, A. Spanghler, and Y. Liu. Actionable Recourse in Linear Classification. In *ACM Conference on Fairness, Accountability, and Transparency (FAT '19)*, 2019.

[47] S. Wachter, B. Mittelstadt, and C. Russell. Counterfactual Explanations Without Opening the Black Box: Automated Decisions and the GDPR. *Harvard Journal of Law & Technology*, 31:841–887, 04 2018.

[48] W. Yang, J. Li, C. Xiong, and S. C. H. Hoi. MACE: An Efficient Model-Agnostic Framework for Counterfactual Explanation. *ArXiv*, abs/2205.15540, 2022.