

# RoCourseNet: Robust Training of a Prediction Aware Recourse Model

Hangzhi Guo  
hangz@psu.edu  
The Pennsylvania State University  
University Park, PA, USA

Feiran Jia  
fzj5059@psu.edu  
The Pennsylvania State University  
University Park, PA, USA

Jinghui Chen  
jzc5917@psu.edu  
The Pennsylvania State University  
University Park, PA, USA

Anna Squicciarini  
acs20@psu.edu  
The Pennsylvania State University  
University Park, PA, USA

Amulya Yadav  
amulya@psu.edu  
The Pennsylvania State University  
University Park, PA, USA

## ABSTRACT

Counterfactual (CF) explanations for machine learning (ML) models are preferred by end-users, as they explain the predictions of ML models by providing a recourse (or contrastive) case to individuals who are adversely impacted by predicted outcomes. Existing CF explanation methods generate recourses under the assumption that the underlying target ML model remains stationary over time. However, due to commonly occurring distributional shifts in training data, ML models constantly get updated in practice, which might render previously generated recourses invalid and diminish end-users trust in our algorithmic framework. To address this problem, we propose RoCourseNet, a training framework that jointly optimizes predictions and recourses that are robust to future data shifts. This work contains four key contributions: (1) We formulate the robust recourse generation problem as a tri-level optimization problem which consists of two sub-problems: (i) a bi-level problem that finds the worst-case adversarial shift in the training data, and (ii) an outer minimization problem to generate robust recourses against this worst-case shift. (2) We leverage adversarial training to solve this tri-level optimization problem by: (i) proposing a novel *virtual data shift (VDS)* algorithm to find worst-case shifted ML models via explicitly considering the worst-case data shift in the training dataset, and (ii) a block-wise coordinate descent procedure to optimize for prediction and corresponding robust recourses. (3) We evaluate RoCourseNet’s performance on three real-world datasets, and show that RoCourseNet consistently achieves more than 96% robust validity and outperforms state-of-the-art baselines by at least 10% in generating robust CF explanations. (4) Finally, we generalize the RoCourseNet framework to accommodate any parametric post-hoc methods for improving robust validity.

## CCS CONCEPTS

• **Computing methodologies** → *Machine learning*.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or to publish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from [permissions@acm.org](mailto:permissions@acm.org).  
*CIKM '23, October 21–25, 2023, Birmingham, United Kingdom.*

© 2023 Copyright held by the owner/author(s). Publication rights licensed to ACM.  
ACM ISBN 979-8-4007-0124-5/23/10...\$15.00  
<https://doi.org/10.1145/3583780.3615040>

## KEYWORDS

Counterfactual Explanation, Algorithmic Recourse, Explainable Artificial Intelligence, Interpretability

### ACM Reference Format:

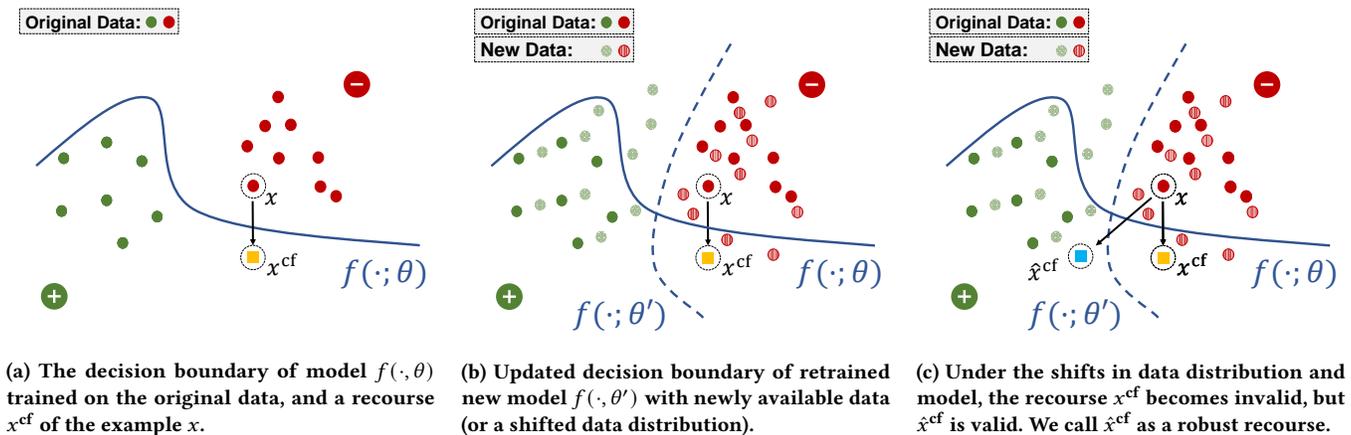
Hangzhi Guo, Feiran Jia, Jinghui Chen, Anna Squicciarini, and Amulya Yadav. 2023. RoCourseNet: Robust Training of a Prediction Aware Recourse Model. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management (CIKM '23)*, October 21–25, 2023, Birmingham, United Kingdom. ACM, New York, NY, USA, 15 pages. <https://doi.org/10.1145/3583780.3615040>

## 1 INTRODUCTION

Existing work in Explainable Artificial Intelligence (XAI) has been focused on developing techniques to interpret decisions made by black-box machine learning (ML) models [24, 25, 29, 40]. In particular, counterfactual (CF) explanation methods find a new *counterfactual* example  $x^{cf}$ , which is similar to input instance  $x$  but gets a different/opposite prediction from the ML model. Counterfactual explanation techniques [23, 34, 49, 52] are often preferred by human end-users because of their ability to provide actionable recourse<sup>1</sup> to individuals who are negatively impacted by algorithm-mediated decisions. For example, CF explanation techniques can be used to provide algorithmic recourse for impoverished loan applicants who have been denied a loan by a bank’s ML algorithm, etc.

Most CF explanation techniques assume that the underlying ML model is stationary and does not change over time [3]. However, in practice, ML models are often updated regularly when new data is available to improve predictive accuracy on the new shifted data distribution. This shifted ML model might render previously recommended recourses ineffective [39], and in turn, diminish end users’ trust towards our system. For example, when providing a recourse to a loan applicant who was denied a loan by the bank’s ML algorithm, it is critical to ensure that the bank can honor that recourse and approve re-applications that fully follow recourse recommendations, even if the bank updates their ML model in the meantime. This necessitates the development of robust algorithms that can generate recourses that remain effective (or valid) for an end-user in the face of ML models being frequently updated. Figure 1 illustrates this challenge of generating robust recourses.

<sup>1</sup>Note that counterfactual explanation [52] and algorithmic recourse [49] are closely related [47, 51]. Hence, we use these terms interchangeably.



**Figure 1: Illustration of the robust recourse generation process.** (a) Given an input data point  $x$ , CF explanation methods generate a new recourse  $x^{cf}$  which lies on the opposite side of decision boundary  $f(\cdot; \theta)$ . (b) As new data is made available, the ML model’s decision boundary is updated as  $f(\cdot; \theta')$ . This shifted decision boundary  $f(\cdot; \theta')$  invalidates the chosen recourse  $x^{cf}$  (as  $x$  and  $x^{cf}$  lie on the same side of the shifted model  $f(\cdot; \theta')$ ). (c) However, robust CF explanation methods generate a **robust recourse**  $\hat{x}^{cf}$  for input  $x$  by anticipating the future shifted model  $f(\cdot; \theta')$ .

**Limitations of Prior Work.** To our knowledge, only two studies [36, 48] propose methods to generate robust recourses. Unfortunately, both these studies suffer from two major limitations. First, both methods are based on strong modeling assumptions which degrade their effectiveness at finding robust recourses (as we show in Section 4). For example, Upadhyay et al. [48] assume that the ML model’s decision boundary can be locally approximated via a linear function, and adopt LIME [40] to find this linear approximation. However, recent works show that the local approximation generated from LIME is unfaithful [26, 41] and inconsistent [1, 46]. Similarly, Nguyen et al. [36] assumes that the underlying data distribution can be approximated via kernel density estimation [5]. However, kernel density estimation suffers from the *curse of dimensionality* [4], which performs exponentially worse with increasing dimensionality of data [10, 35]. This limits its usability for estimating data distributions in real-world high-dimensional datasets.

Second, these two techniques are post-hoc methods designed for use with proprietary black-box ML models whose training data and model weights are not available. However, with the advent of data regulations that enshrine the “*Right to Explanation*” (e.g., EU-GDPR [52]), service providers are required by law to communicate both the decision outcome (i.e., the ML model’s prediction) and its actionable implications (i.e., a recourse for this prediction) to an end-user. In these scenarios, the post-hoc assumption is overly limiting, as service providers can build recourse models that leverage the knowledge of their ML model to generate higher-quality recourses. In fact, prior work [20] has shown that post-hoc CF explanation approaches are unable to balance the cost-invalidity trade-off [39], which is an important consideration in generating recourses. To date, very little prior work departs from the post-hoc paradigm; [20] propose one such approach, unfortunately, it does not consider the robustness of generated recourses.

**Contributions.** We propose **Robust ReCourse Neural Network** (or RoCourseNet), a novel framework for generating recourses which: (i) departs from the paradigm of post-hoc explainability in generating recourses; while (ii) optimizing the robustness of recourse explanations. RoCourseNet presents four key contributions:

- (Formulation-wise) We formulate the robust recourse generation problem as a tri-level (min-max-min) optimization problem, which consists of two sub-problems: (i) a bi-level (max-min) problem which simulates a worst-case attacker to find an adversarially shifted ML model by explicitly simulating the *worst-case data shift* in the training dataset; and (ii) an outer minimization problem which simulates an ML model designer who wants to generate robust recourses against this worst-case bi-level attacker. Unlike prior approaches, our bi-level attacker formulation explicitly connects shifts in the underlying data distribution to corresponding shifts in the ML model parameters.
- (Methodology-wise) We propose *RoCourseNet* for solving our tri-level optimization problem for generating robust recourses. RoCourseNet relies on two key ideas: (i) we propose a novel *Virtual Data Shift (VDS)* algorithm to optimize for the inner bi-level (max-min) attacker problem, which results in an adversarially shifted model; and (ii) inspired by Guo et al. [20], RoCourseNet leverages a block-wise coordinate descent training procedure to optimize the robustness of generated recourses against these adversarially shifted models. Unlike prior methods [36, 48], our method requires no intermediate steps in approximating the underlying model or data distribution.
- (Experiment-wise) We conduct rigorous experiments on three real-world datasets to evaluate the robustness of several popular recourse generation methods under data shifts. Our results show that RoCourseNet generates highly robust CF explanations against data shifts, as it consistently achieves >96% robust validity, outperforming state-of-the-art baselines by ~10%.

- (Framework-wise) Finally, we extend the RoCourseNet training as a generalized robust training framework to be used with *any* parametric post-hoc explanation method. By applying the RoCourseNet training, we witness ~25% robust validity improvements to existing CF parametric methods.

## 2 RELATED WORK

**Counterfactual Explanation Techniques.** A significant body of literature exists on CF explanation techniques, which focuses on generating recourses that lead to different (and often more preferable) predicted outcomes [22, 51, 52]. We categorize prior work on CF explanation techniques into *non-parametric methods* [22, 23, 34, 48–52], which aim to find recourses without involving parameterized models, and *parametric methods* [20, 32, 37, 55], which adopt parametric models (e.g., a neural network model) to generate recourses. In particular, our work is most closely related to CounterNet [20], which unlike post-hoc methods, jointly trains the predictive model and a CF explanation generator. This joint-training procedure leads to significantly better alignment between the generated predictions and corresponding CF explanations. *However, all aforementioned CF explanation techniques (including CounterNet) do not optimize for robustness against adversarial model shifts. In contrast, we devise a novel tri-level adversarial training approach to ensure the robustness of CF explanations generated by RoCourseNet.*

**Robustness in Recourse Explanations.** Our method is closely related to the model shift problem in algorithmic recourse [39], i.e., how to ensure that the generated recourse is robust to shifts in the underlying predictive model. However, existing approaches [36, 48] rely on simplifying assumptions: (i) Upadhyay et al. [48] propose ROAR which relies on a locally linear approximation (via LIME [40]) to construct a shifted model, which is known to suffer from inconsistency [1, 46] and unfaithfulness issues [26, 41]. (ii) Similarly, Nguyen et al. [36] propose RBR which assumes that kernel density estimators can approximate the underlying data distribution. In particular, RBR uses Gaussian kernels for multivariate density estimation, which suffers from the curse of dimensionality [4, 10, 35]. In contrast, our work relaxes these assumptions by constructing adversarial shifted models via simulating the worst-case data shift, and conducting adversarial training for robust CF generation.

Orthogonal to our work, Pawelczyk et al. [38] analyze the *model multiplicity* problem, which studies the validity of recourses under different ML models trained on the *same* data, and Black et al. [6] propose methods to ensure consistency under the model multiplicity setting. In addition, some prior work focus on ensuring robustness to small perturbations in the recourse features [12, 14, 33].

**Adversarial training.** We leverage adversarial robustness techniques to protect ML models from adversarial examples [8, 17, 31, 45, 54]. In addition, recent works [15, 16] also leverage adversarial training to defend against data poisoning [21] and backdoor attacks [42]. In general, adversarial training solves a bi-level (min-max) optimization problem. In our work, we formulate RoCourseNet’s objective as a tri-level (min-max-min) optimization problem, which we can decompose into a game played between a model designer and a worst-case (hypothetical) adversary. The inner worst-case data and

model shifts are assumed to be generated by a bi-level worst-case attacker. The defender trains a robust CF generator against this bi-level attacker by following an adversarial training procedure.

## 3 ROCOURSENET: END-TO-END ROBUST RECURSE GENERATION

RoCourseNet is an end-to-end training framework for simultaneously generating accurate predictions and corresponding recourses (or CF explanations) that are robust to model shifts induced by shifts in the training dataset. We describe the RoCourseNet framework in two stages. First, we discuss the attacker’s problem: (i) we propose a novel bi-level attacker problem to find the worst-case data shift that leads to an adversarially shifted ML model; and (ii) we propose a novel Virtual Data Shift (VDS) algorithm for solving this bi-level attacker problem. Second, we discuss the defender’s problem: (i) we derive a novel tri-level learning problem based on the attacker’s bi-level problem; and (ii) we propose the RoCourseNet training framework for optimizing this tri-level optimization problem, which leads to the simultaneous generation of accurate predictions and robust recourses.

### 3.1 Virtual Data Shift: Constructing Worst-case Data Shifts

We define a predictive model  $f : \mathcal{X} \rightarrow \mathcal{Y}$ . Let  $\mathcal{D} = \{(x_i, y_i) \mid i \in \{1, \dots, N\}\}$  represent our training dataset containing  $N$  points. We denote  $f(x, \theta)$  as the prediction generated by predictive model  $f$  on point  $x$ , parameterized by  $\theta$ . Next, we denote  $\theta$  and  $\theta'$  as parameters of an (original) ML model  $f(\cdot; \theta)$  and its shifted counterpart  $f(\cdot; \theta')$ , respectively. Also, let  $x^{\text{cf}}$  denote a CF explanation (or recourse) for input point  $x$ . Finally, a recourse  $x^{\text{cf}}$  is *valid* iff it gets an opposite prediction from the original data point  $x$ , i.e.,  $f(x^{\text{cf}}; \theta) = 1 - f(x, \theta)$ . On the other hand, a recourse  $x^{\text{cf}}$  is *robustly valid* w.r.t. a shifted model  $f(\cdot; \theta')$  iff  $x^{\text{cf}}$  gets an opposite prediction from the shifted model (as compared to the prediction received by  $x$  on the original model), i.e.,  $f(x^{\text{cf}}; \theta') = 1 - f(x; \theta)$ . This definition aligns with the notion of robustness to model shifts in the literature [48]. Finally,  $\mathcal{L}(\cdot, \cdot)$  represents a loss function formulation, e.g., binary cross-entropy, mean squared error, etc.

**Model Shift as an optimization problem.** To motivate the need of introducing our bi-level attacker problem, we first discuss an optimization problem for a worst-case attacker which directly perturbs model parameters to find an adversarially shifted model (denoted by  $f(\cdot; \theta'_{adv})$ ). The goal of the attacker is to find an adversarially shifted model which minimizes the robustness of the generated recourses. More formally, given our training dataset  $\mathcal{D}$  and a CF explanation method (that can generate recourses  $x^{\text{cf}}$  for each  $(x, y) \in \mathcal{D}$ ), the attacker’s problem aims to find the worst-case shifted model  $f(\cdot; \theta'_{adv})$  which minimizes the robust validity of the generated recourses, i.e.,  $f(x^{\text{cf}}; \theta'_{adv}) \neq 1 - f(x; \theta)$ . We can find an adversarial shifted model by solving Equation 1.

$$\theta'_{adv} = \operatorname{argmax}_{\theta' \in \mathcal{F}} \frac{1}{N} \sum_{(x_i, y_i) \in \mathcal{D}} \left[ \mathcal{L} \left( f \left( x_i^{\text{cf}}; \theta' \right), 1 - f \left( x_i; \theta \right) \right) \right] \quad (1)$$

where  $x^{\text{cf}}$  correspond to CF explanations of input  $x$  produced by a CF generator, and  $\mathcal{F} = \{\theta' \mid \theta + \delta_f\}$  denotes a plausible set of the parameters of all possible shifted models.

Unfortunately, it is non-trivial to construct a plausible model set  $\mathcal{F}$  by directly perturbing the ML model's parameters  $\theta$ , especially when  $f(\cdot; \theta)$  is represented using a neural network. Unlike a linear model, quantifying the importance of neurons is challenging [11, 27], which leads to difficulty in applying weight perturbations. To overcome these challenges, prior work [48] adopts a simplified linear model to approximate the target model, and perturbs this linear model accordingly. Unfortunately, this simplified local linear model introduces approximation errors into the system, which leads to poor performance (as shown in Section 4). Instead of directly perturbing the model's weights, we explicitly consider a worst-case data shift, which then leads to an adversarial model shift.

**Data Shift as a bilevel optimization problem.** We identify distributional data shifts as the fundamental cause of non-robust recourses. Unlike prior work, we propose a bi-level optimization problem for the attacker which explicitly connects shifts in the underlying training data to corresponding shifts in the ML model parameters. Specifically, in response to a shift in the training data (from  $\mathcal{D}$  to  $\mathcal{D}_{\text{shifted}}$ ), defenders update (or shift) their predictive model by optimizing the prediction loss:

$$\theta'_{opt} = \operatorname{argmin}_{\theta'} \frac{1}{N} \sum_{(x_i, y_i) \in \mathcal{D}_{\text{shifted}}} \left[ \mathcal{L}(f(x_i; \theta'), y_i) \right]. \quad (2)$$

Crucially, this updated ML model  $f(\cdot, \theta'_{opt})$  (caused by the shifted data  $\mathcal{D}_{\text{shifted}}$ ) is the key to the non-robustness of recourses (i.e.,  $f(x^{\text{cf}}; \theta'_{opt}) \neq 1 - f(x; \theta)$ ). Therefore, to generate robust recourses, we optimize against an adversary who creates a worst-case shifted dataset  $\mathcal{D}^*_{\text{shifted}}$  such that the correspondingly updated model (found by solving Eq. 2) minimizes the robust validity of CF examples  $x^{\text{cf}}$ . Then, this data shift problem becomes a bi-level problem:

$$\begin{aligned} \delta^* &= \operatorname{argmax}_{\delta, \forall \delta_i \in \Delta} \frac{1}{N} \sum_{(x_i, y_i) \in \mathcal{D}} \left[ \mathcal{L}(f(x_i^{\text{cf}}; \theta'_{opt}(\delta)), 1 - f(x_i; \theta)) \right] \\ \text{s.t.}, \theta'_{opt}(\delta) &= \operatorname{argmin}_{\theta'} \frac{1}{N} \sum_{(x_i, y_i) \in \mathcal{D}} \left[ \mathcal{L}(f(x_i + \delta_i; \theta'), y_i) \right]. \end{aligned} \quad (3)$$

where  $\delta_i \in \Delta$  denotes the data shift for a single data point  $x_i \in \mathcal{D}$ , and  $\delta = \{\delta_i \mid \forall (x_i, y_i) \in \mathcal{D}\}$  denotes the data shift across all data points in the entire dataset. We define  $\Delta$  as the  $l_\infty$ -norm ball  $\Delta = \{\delta \in \mathbb{R}^n \mid \|\delta\|_\infty \leq \epsilon\}$ . Intuitively, the outer problem minimizes the robust validity to construct the worst-case data shift  $\delta^*$ , and the inner problem learns a shifted model  $f(\cdot; \theta'_{opt})$  on the shifted dataset  $\mathcal{D}^*_{\text{shifted}}$ . Once we get the optimal  $\delta^* = \{\delta_1^*, \delta_2^*, \dots, \delta_N^*\}$  by solving Equation 3, the worst-case shifted dataset  $\mathcal{D}^*_{\text{shifted}} = \{(x_i + \delta_i^*, y_i) \mid \forall (x_i, y_i) \in \mathcal{D}\}$ .

**Virtual Data Shift (VDS).** Unfortunately, solving Eq. 3 is computationally intractable due to its nested structure. To approximate this bi-level problem, we devise *Virtual Data Shift (VDS)* (Algorithm 1), a gradient-based algorithm with an unrolling optimization pipeline. At a high level, VDS iteratively approximates the inner problem by

---

### Algorithm 1 Virtual Data Shift (VDS)

---

- 1: **Hyperparameters:** learning rates  $\eta$ , step size  $\alpha$ , # of attacker steps  $T$ , # of unrolling steps  $K$
  - 2: **Input:** model weights  $\theta$ , perturbation constraints  $\epsilon$ , batch  $B = (\mathbf{x}, \mathbf{y})$ , CF examples  $\mathbf{x}^{\text{cf}}$
  - 3: **Initialize:** virtual shifted model weights  $\theta' = \theta$ ,  $\delta \sim \mathcal{U}(-\epsilon, +\epsilon)$
  - 4: **for**  $i = 1 \rightarrow T$  steps **do**
  - 5:     **for**  $k = 1 \rightarrow K$  unroll steps **do**
  - 6:          $\theta' \leftarrow \theta' - \eta \cdot \nabla_{\theta'} \mathcal{L}(f(\mathbf{x} + \delta); \theta'), \mathbf{y}$
  - 7:     **end for**
  - 8:      $\delta \leftarrow \delta + \alpha \cdot \operatorname{sign}(\nabla_{\delta} \mathcal{L}(f(\mathbf{x}^{\text{cf}}; \theta'), 1 - f(\mathbf{x}; \theta)))$
  - 9:     Project  $\delta$  onto the  $l_\infty$ -norm ball.
  - 10: **end for**
  - 11: **return**  $\theta', \delta$
- 

unrolling  $K$ -steps of gradient descent for each outer optimization step. Similar unrolling pipelines are adopted in many ML problems with a bi-level formulation [19, 44], e.g., meta-learning [13], hyperparameter search [30], and poisoning attacks [21].

Algorithm 1 layouts the VDS algorithm which outputs the worst-case data shift  $\delta^*$ , and the corresponding shifted model  $f(\cdot; \theta'_{opt})$ . VDS makes two design choices. First, it uniformly randomizes the data shift  $\delta \sim \mathcal{U}(-\epsilon, +\epsilon)$ , where  $\delta = \{\delta_1, \dots, \delta_N\}$  (Line 3), following practices of Wong et al. [54]. Uniform randomization is critical to adversarial model performance as it increases the smoothness of the objective function, leading to improved convergence of gradient-based algorithms [8]. Then, we iteratively solve this bi-level optimization problem via  $T$  outer attack steps. At each step, we first update the predictor  $f(\cdot; \theta')$  using the shifted data  $\mathbf{x} + \delta$  via  $K$  unrolling steps of gradient descent (Line 6). Next, similar to the fast sign gradient method (FSGM) [17], we maximize the adversarial loss and project  $\delta$  into the feasible region  $\Delta$  (i.e.,  $l_\infty$  norm ball; Line 8-9). Crucially, when computing the gradient of adversarial loss (outer problem) w.r.t. data shift  $\delta$  (Line 8), we look ahead a few steps in the inner problem before back-propagating to the initial unrolling step. This approach stems from applying  $K$  unrolling steps of gradient descent, as opposed to full-blown gradient descent until convergence. Note that the gradient w.r.t.  $\delta$  depends on  $\theta(\delta)$ , where  $\theta(\delta)$  is a function derived from LINE 5-7.

## 3.2 Block-wise Coordinate Descent with Adversarial Training

**Choice of CF Explanation Technique.** Note that our bi-level attacker formulation assumes that CF explanations  $x_i^{\text{cf}}$  for all data points  $x_i$  are provided as input to the VDS algorithm. Thus, a key design choice inside the RoCourseNet framework is the selection of an appropriate CF explanation technique, which can be used to generate recourses for input data points in Algorithm 1.

As mentioned in Section 2, most existing CF explanation techniques follow the post-hoc paradigm, which makes them unsuitable for use inside the RoCourseNet framework for two reasons: (ii) *misaligned motivations*: post-hoc CF explanation methods are mainly designed for use with proprietary black-box ML models whose training data and model weights are not available; instead, the VDS

algorithm relies on having access to the training data. Thus, the motivations and use cases of VDS and post-hoc methods are misaligned. (ii) the post-hoc paradigm is overly limiting in many real-world scenarios. With the advent of data regulations that enshrine the “*Right to Explanation*” (e.g., EU-GDPR [52]), service providers are required by law to communicate both the decision outcome (i.e., the ML model’s prediction) and its actionable implications (i.e., a recourse for this prediction) to an end-user. In these scenarios, the post-hoc assumption is overly limiting. Service providers can build specialized CF explanation techniques that leverage the knowledge of their specific ML model to generate higher-quality recourses.

Motivated by these reasons, we choose CounterNet [20] as our CF explanation model of choice inside the RoCourseNet framework, as that is an end-to-end approach that departs from the post-hoc explanation paradigm by jointly training predictions and recourses. In fact, [20] show that CounterNet can better balance the cost-inequality trade-off [39] than state-of-the-art post-hoc approaches.

**RoCourseNet Objective Function.** We describe how we combine the bi-level attacker problem with an outer minimization, which represents RoCourseNet’s tri-level objective function. Inspired by [20], RoCourseNet has three objectives: (i) high *predictive accuracy* - we expect RoCourseNet to output accurate predictions; (ii) high *robust validity* - we expect that generated recourses in RoCourseNet are robustly valid on shifted models<sup>2</sup>; (iii) low *proximity* - we desire minimal changes required to modify input instance  $x$  to corresponding recourse  $x^{\text{cf}}$ . Given these objectives, RoCourseNet solves the following min-max-min problem to optimize parameters for its predictor  $f(\cdot; \theta)$  and recourse generator  $g(\cdot; \theta_g)$ . Note that similar to [20], both  $f(\cdot; \theta)$  and  $g(\cdot; \theta_g)$  are neural networks.

$$\begin{aligned} & \underset{\theta, \theta_g}{\operatorname{argmin}} \left( \frac{1}{N} \sum_{(x_i, y_i) \in \mathcal{D}} \left[ \underbrace{\lambda_1 \cdot \mathcal{L}(f(x_i; \theta), y_i)}_{\text{Prediction Loss } (L_1)} + \lambda_3 \cdot \underbrace{\mathcal{L}(x_i, x_i^{\text{cf}})}_{\text{Proximity Loss } (L_3)} \right] \right. \\ & \left. + \max_{\delta, \forall \delta_i \in \Delta} \frac{1}{N} \sum_{(x_i, y_i) \in \mathcal{D}} \left[ \underbrace{\lambda_2 \cdot \mathcal{L}(f(x_i^{\text{cf}}; \theta'_{\text{opt}}(\delta)), 1 - f(x_i; \theta))}_{\text{Robust Validity Loss } (L_2)} \right] \right) \\ \text{s.t.}, \theta'_{\text{opt}}(\delta) &= \underset{\theta'}{\operatorname{argmin}} \frac{1}{N} \sum_{(x_i, y_i) \in \mathcal{D}} \left[ \mathcal{L}(f(x_i + \delta_i; \theta'), y_i) \right], \\ x_i^{\text{cf}} &= g(x_i; \theta_g). \end{aligned} \quad (4)$$

**RoCourseNet Training.** A common practice in solving a min-max formulation is to first solve the inner maximization problem, and then solve the outer minimization problem [31]. Thus, we solve Eq. 4 as follows (see Algorithm 2): (i) To solve the inner bi-level problem, we find the worst-case model shift by solving Eq. 3 using VDS (Algorithm 1). (ii) To solve the outer minimization problem, we adopt block-wise coordinate descent optimization by distributing gradient descent backpropagation on the objective function into two stages (as suggested in [20]) – at stage one, we optimize for predictive accuracy, i.e.,  $L_1$  in Eq. 4 (Line 7), and at stage two,

<sup>2</sup>Note that robust validity also ensures validity on the original predictive model (i.e.,  $f(x; \theta) = 1 - f(x^{\text{cf}}; \theta)$ ), as the worst-case model shift case encompasses the unshifted model case. We also empirically validate this design choice in Section 4.

---

### Algorithm 2 Tri-level Robust CF Training

---

```

1: Hyperparameters: learning rates  $\eta$ , # of epochs  $N$ , maximum
   perturbation  $E$ 
2: Input: dataset  $(x, y) \in \mathcal{D}$ 
3: Initialize:  $\theta$ .
4: for epoch = 1  $\rightarrow$   $N$  do
5:    $\epsilon = E \cdot \text{epoch}/N$  ▷ Linearly schedule  $\epsilon$ .
6:   for each minibatch  $B$  in  $\mathcal{D}$  do
7:      $\theta \leftarrow \theta - \eta \cdot \nabla_{\theta} L_1$ 
8:      $\theta', \delta \leftarrow \text{VDS}(B, x^{\text{cf}}, \theta, \epsilon)$ 
9:      $\theta_g \leftarrow \theta_g - \eta \cdot \nabla_{\theta_g} (\lambda_2 \cdot L_2 + \lambda_3 \cdot L_3)$ 
10:  end for
11: end for

```

---

optimize the quality of CF explanations, i.e.,  $\lambda_2 \cdot L_2 + \lambda_3 \cdot L_3$  in Eq. 4 (Line 9). Note that in Line 9, the gradient of  $\lambda_2 \cdot L_2 + \lambda_3 \cdot L_3$  is calculated using the adversarially shifted model parameters  $\theta'$  found by VDS in Line 8. [20] show that this block-wise coordinate descent algorithm efficiently optimizes the outer minimization problem by alleviating the problem of divergent gradients.

In addition, we linearly increase the perturbation constraints  $\epsilon$  for improved convergence of our robust CF generator. Intuitively, linearly increasing  $\epsilon$  values correspond to increasingly strong adversaries. Prior work in curriculum adversarial training [7, 53] suggests that adaptively adjusting the strength of the adversary improves the convergence of adversarial training (we verify this in Section 4).

## 4 EXPERIMENTAL EVALUATION

**Baselines.** To our knowledge, RoCourseNet is the first method that optimizes an end-to-end model for generating predictions and robust recourses. Hence, there exist no previous approaches which we can directly compare against. Nevertheless, we compare RoCourseNet against four state-of-the-art baselines.

- VANILLACF [52] is a popular post-hoc non-parametric method optimizing validity and proximity.
- COUNTERNET [20] is the first end-to-end model for simultaneously generating predictions and recourses, which inspires the development of RoCourseNet.
- ROAR-LIME [48] generates robust recourses by perturbing parameters of locally approximated linear models.
- RBR [36] generates robust recourses from a Gaussian kernel.

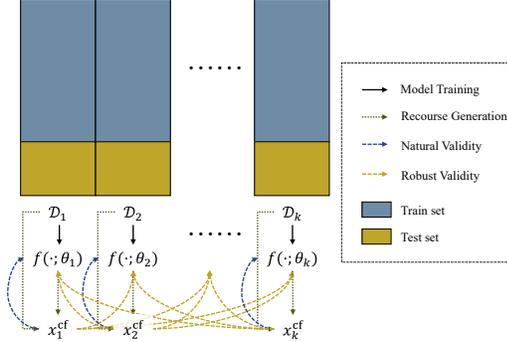
Other than *CounterNet*, all remaining baselines require a trained ML model as an input. Similar to [20], we train a neural network model as the base ML model for all baselines. For each dataset, we separately tune hyperparameters via grid search.

**Datasets.** To remain consistent with prior work on robust recourses [36, 48], we evaluate RoCourseNet on three benchmarked real-world datasets. Table 1 summarizes these three datasets.

- *Loan* [28] captures *temporal shifts* in loan application records. It predicts whether a business defaulted on a loan ( $Y=1$ ) or not ( $Y=0$ ). This large-sized dataset (i.e., ~450k data points) has loan approval records across the U.S. during 1994 to 2009 (i.e.,  $\mathcal{D}_{\text{all}} = \{\mathcal{D}_1, \mathcal{D}_2, \dots, \mathcal{D}_k\}$ , where each subset  $\mathcal{D}_i$  corresponds to a particular year, and  $k = 16$  is the total number of years).

**Table 1: Summary of Datasets used for Evaluation. Each dataset consists of  $k$  subsets. Size represents the number of data points for the entire dataset.**

Dataset	Size	$k$	#Continuous	#Categorical
Loan	449,152	16	7	5
German Credit	2,000	2	6	3
Student	649	2	2	13



**Figure 2: Illustration of evaluating the recourse robustness under the distributional shift.**

- *German Credit* [2] captures *data correction shifts*. It predicts whether the credit score of a customer is good ( $Y=1$ ) or bad ( $Y=0$ ). This dataset has 2,000 data points with two versions; each version contains 1,000 data points (i.e.,  $\mathcal{D}_{\text{all}} = \{\mathcal{D}_1, \mathcal{D}_2\}$ , where  $\mathcal{D}_1, \mathcal{D}_2$  corresponds to the original and corrected datasets, respectively).
- Finally, we use the *Student* dataset [9], which captures *geospatial shifts*. It predicts whether a student will pass ( $Y=1$ ) or fail ( $Y=0$ ) the exam. It contains 649 student records in two places (i.e.,  $\mathcal{D}_{\text{all}} = \{\mathcal{D}_1, \mathcal{D}_2\}$ , and  $\mathcal{D}_1, \mathcal{D}_2$  represent a particular school).

**Evaluation Procedure & Metrics.** Figure 2 illustrates the evaluation procedure of the experiment. Each dataset is partitioned into  $k$  subsets  $\mathcal{D}_{\text{all}} = \{\mathcal{D}_1, \mathcal{D}_2, \dots, \mathcal{D}_k\}$ . This partitioning enables us to create  $k$  original datasets  $\mathcal{D}_i \in \mathcal{D}_{\text{all}}$ , and  $k$  corresponding shifted datasets,  $\mathcal{D}_{\text{shifted},i} = \mathcal{D}_{\text{all}} \setminus \mathcal{D}_i$ , where each shifted dataset  $\mathcal{D}_{\text{shifted},i}$  contains all subsets of the dataset  $\mathcal{D}_{\text{all}}$  except for the original dataset  $\mathcal{D}_i$ . Next, we do a train/test split on each  $\mathcal{D}_i \in \mathcal{D}_{\text{all}}$  (i.e.,  $\mathcal{D}_i = \{\mathcal{D}_i^{\text{train}}, \mathcal{D}_i^{\text{test}}\}$ ). We train a separate model (i.e., the entire models for *RoCourseNet* and *CounterNet*, and predictive models for other baselines) on each train set  $\mathcal{D}_i^{\text{train}}, \forall i \in \{1, \dots, k\}$ . Then, we use the model trained on  $\mathcal{D}_i^{\text{train}}$  to generate recourses on the hold-out sets  $\mathcal{D}_i^{\text{test}}, \forall i \in \{1, \dots, k\}$ . Finally, we evaluate the robustness (against the model shift) of recourses generated on  $\mathcal{D}_i^{\text{test}}, \forall i \in \{1, \dots, k\}$  as follows: for each recourse  $x^{\text{cf}}$  (corresponding to an input instance  $x$  in  $\mathcal{D}_i^{\text{test}}$ ), we evaluate its robust validity by measuring the fraction of shifted models (i.e.,  $k-1$  models trained on all shifted training sets) on which  $x^{\text{cf}}$  remains *robustly valid* (see definitions in Section 3.1).

Finally, we use three metrics to evaluate a CF explanation: (i) *Validity* is the fraction of valid CF examples on the original model  $f(\cdot; \theta)$ . (ii) *Robust validity* is the fraction of robustly valid CF examples on a *shifted* predictive model  $f(\cdot; \theta')$ . We calculate the robust validity on all possible shifted models (as described above). (iii) *Proximity* is the  $l_1$  distance between the input and the CF example. We report the averaged results across all subsets  $\mathcal{D}_{\text{all}}$  (see Table 2).

## 4.1 Experimental Results

**Validity & Robust Validity.** Table 2 compares the validity and robust validity achieved by *RoCourseNet* and other baselines. *RoCourseNet* achieves the highest *robust validity* on each dataset - it outperforms *ROAR-LIME* by 10% (the next best performing baseline), and consistently achieves at least 96.5% robust validity. This illustrates *RoCourseNet*'s effectiveness at generating highly robust recourses. Also, *RoCourseNet* achieves the highest *validity* on each dataset. This result shows that optimizing the worst-case shifted model (i.e.,  $L_2$  in Eq. 4) is sufficient to achieve high validity, without the need to explicitly optimize for an additional validity loss.

**Proximity.** Table 2 compares the proximity achieved by *RoCourseNet* and baselines. In particular, *RoCourseNet* performs exceedingly well on the *Loan* application dataset (our largest dataset with  $\sim 450k$  data points), as it is the second-best method in terms of proximity (just behind *CounterNet*), and outperforms all our post-hoc baseline methods (*RBR*, *ROAR-LIME*, and *VanillaCF*). Perhaps understandably, *RoCourseNet* achieves poorer proximity on the *German Credit* and *Student* dataset, given that the limited size of these datasets (less than 1000 data points) precludes efficient training.

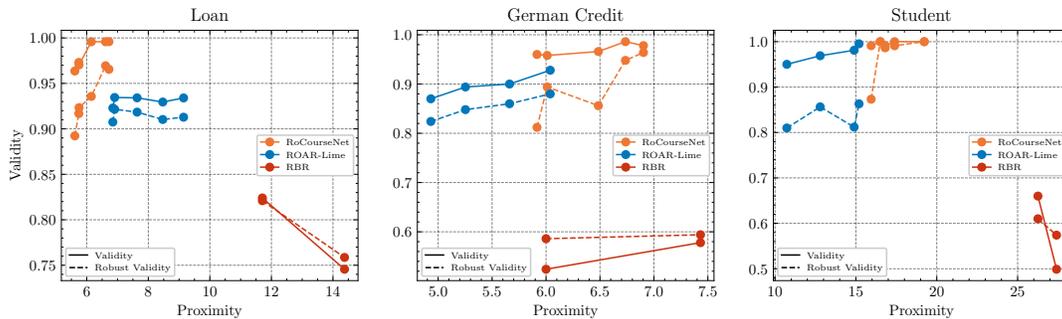
**Cost-Validity Trade-Off.** We compare *RoCourseNet*, *ROAR-LIME*, and *RBR* (three recourse methods explicitly optimizing for distributional shift) in their trade-off between the cost (measured by proximity) and their original and robust validity [39]. For each method, we plot the Pareto frontier of the cost-validity trade-off as follows: (i) For *RBR*, we obtain the Pareto frontier by varying the ambiguity sizes  $\epsilon_1, \epsilon_2$  (i.e., hyperparameters of *RBR* [36]); (ii) For *ROAR-LIME* and *CounterNet*, we obtain the Pareto frontier by varying the trade-off hyperparameter  $\lambda$  that controls the proximity loss term in their respective loss functions (e.g.,  $\lambda_3$  in Equation 4). Figure 3 shows that on the large-sized *Loan* dataset, *RoCourseNet*'s Pareto frontiers either dominate or are comparable to frontiers achieved by *ROAR-LIME* and *RBR*. On the other hand, we observe a clear trade-off on the *German Credit* and *Student* datasets, where *RoCourseNet* (and other methods) can increase their normal and robust validity, but only at the cost of a poorer proximity score. This is consistent with prior literature, which shows that proximity needs to be sacrificed to achieve higher validity [36].

## 4.2 Further Analysis

***RoCourseNet* Training Requires More Data.** We further delve into the impact of data size on the cost-invalidity trade-off. Figure 5 illustrates the Pareto Frontier plot of *RoCourseNet* trained on various fractions of the *Loan* dataset. This figure shows that increasing the training data leads to a better balance between normal and robust cost-invalidity trade-offs. This supports our hypothesis

**Table 2: Evaluation of recourse robustness under model shift. It is desirable for recourse methods to have *low* proximity (prox.) with *high* validity (Val.) and *high* robust validity (Rob-Val.).**

Methods	Loan			German Credit			Student		
	Prox.	Val.	Rob-Val.	Prox.	Val.	Rob-Val.	Prox.	Val.	Rob-Val.
VANILLACF	7.390 ± 1.860	0.942 ± 0.026	0.885 ± 0.121	<b>4.635</b> ± 0.197	0.940 ± 0.011	0.772 ± 0.008	15.236 ± 0.383	0.915 ± 0.056	0.673 ± 0.006
COUNTERNET	6.746 ± 0.723	0.964 ± 0.085	0.639 ± 0.222	5.719 ± 0.130	0.960 ± 0.028	0.706 ± 0.074	18.619 ± 0.131	0.967 ± 0.033	0.859 ± 0.071
ROAR-LIME	7.648 ± 1.951	0.934 ± 0.024	0.918 ± 0.066	4.862 ± 0.117	0.910 ± 0.0255	0.792 ± 0.052	<b>11.931</b> ± 1.396	0.967 ± 0.026	0.820 ± 0.075
RBR	11.71 ± 1.633	0.824 ± 0.130	0.821 ± 0.132	6.005 ± 2.099	0.524 ± 0.148	0.586 ± 0.046	26.255 ± 3.089	0.660 ± 0.014	0.611 ± 0.073
RoCOURSENET	<b>6.611</b> ± 0.418	<b>0.996</b> ± 0.002	<b>0.969</b> ± 0.106	6.903 ± 0.250	<b>0.978</b> ± 0.002	<b>0.964</b> ± 0.008	16.508 ± 0.281	<b>1.000</b> ± 0.000	<b>1.000</b> ± 0.000



**Figure 3: Pareto frontiers of the cost-invalidity trade-off for ROAR-LIME, RBR, and RoCOURSENET. Methods located in the upper-left region are preferable, as they exhibit a favorable balance between cost and invalidity. On the *Loan* (left) dataset, RoCOURSENET exhibits a superior balance between cost and invalidity compared to ROAR-LIME and RBR. On the *German Credit* (middle) and *Student* (right) dataset, RoCOURSENET achieves high normal and robust validity at the cost of proximity score.**

that the underperformance of RoCourseNet on the *German Credit* and *Student* datasets can be attributed to their limited number of data points, making adversarial training more challenging. This result also echoes findings in prior literature [43], which shows that the sample complexity of robust learning can be significantly larger.

**Evaluating VDS Attacker.** We demonstrate the effectiveness of VDS on solving the inner maximization problem in Eq. 4, i.e., how often VDS succeeds in finding an adversarial shifted model  $f(\cdot; \theta')$ , such that the generated recourses  $x^{cf}$  are not robustly valid. To evaluate the performance of VDS, we apply Algorithm 1 to find shifted model parameters  $\theta'$ , and compute the robust validity of all hold-out test-sets with respect to this shifted model.

Figure 4a and 4d compares the effectiveness of attacking RoCourseNet and CounterNet via the VDS algorithm, which highlights three important findings: (i) First, the VDS algorithm is effective in finding a shifted model which invalidates a given recourse - the average robust validity drops to 74.8% when the attacker steps  $T = 20$ , as compared to 99.8% validity on the original model. (ii) Figure 4a shows that when  $T$  is increased, the robust validity of both CounterNet and RoCourseNet is degraded, which indicates that increasing attack steps improves the effectiveness of solving

the bi-level problem in Eq. 3. Similarly, Figure 4d shows that increasing  $E$  also improves effectiveness of solving Eq.3. (iii) Finally, RoCourseNet is more robust than CounterNet when attacked by the VDS algorithm, as RoCourseNet vastly outperforms CounterNet in robust validity (e.g., ~28%, ~10% improved robust validity when  $T = 20$ ,  $E = 0.5$  in Figure 4a and 4d, respectively).

**Understanding the Tri-level Robust CF Training.** We further analyze our tri-level robust training procedure. First, a stronger attacker (i.e., more effective in solving Eq. 3) leads to the training of a more robust CF generator. In Figure 4b, we observe that increasing  $T$  (which results in a stronger attacker as shown in Figure 4a) leads to improved robust validity, which indicates a more robust CF generator. Similarly, Figure 4e illustrates that increasing  $E$  values leads to improved robust validity. These results show that having an appropriately strong attacker (i.e., effectively optimizing Eq. 3) is crucial to train for a robust CF generator.

**Epsilon Scheduler.** Figure 6 illustrates the importance of linearly scheduling  $\epsilon$  values inside VDS. By linearly increasing  $\epsilon$ , we observe ~0.88% improved robust validity (on average) compared to using a static  $\epsilon$  during the entire adversarial training. This shows that this curriculum training strategy can boost the robustness of recourses.

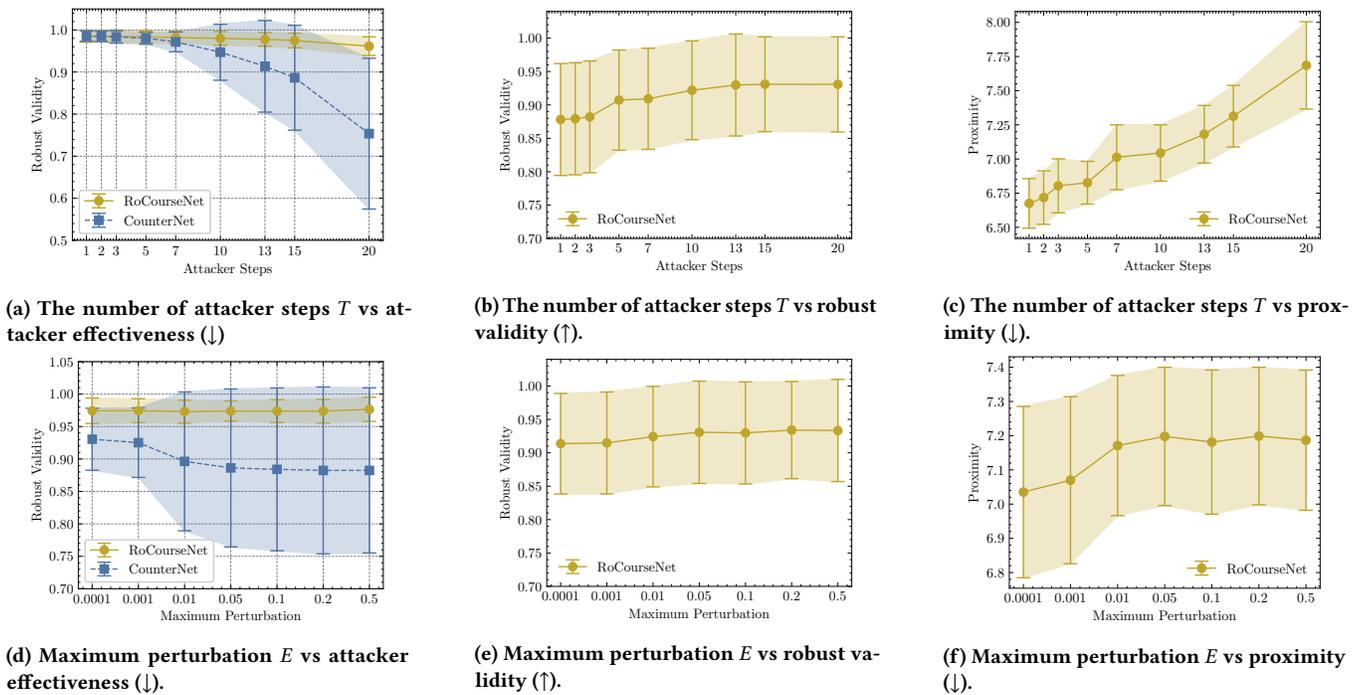


Figure 4: The impact of the number of attacker steps  $T$  (4a-4c) and max-perturbation  $E$  (4d-4f) to robustness on the *Loan* dataset ( $\uparrow$  or  $\downarrow$  means that higher or lower value is preferable, respectively).

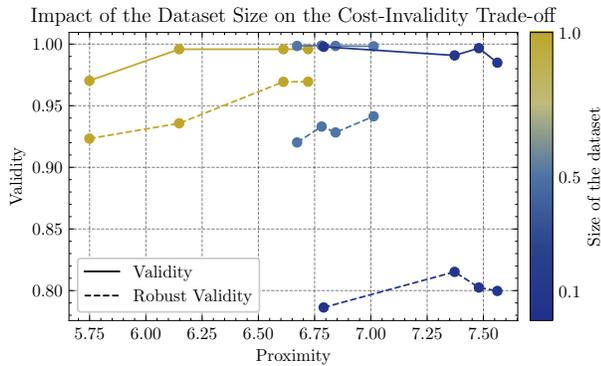


Figure 5: The influence of dataset size on the cost-inequality trade-off. We analyze RoCOURSENET on various fractions of the *Loan* Dataset. The Pareto Frontier plot reveals improvement in both normal and robust cost-inequality trade-offs when increasing training data.

## 5 GENERALIZING ROCOURSENET TO PARAMETRIC CF EXPLANATION METHODS

We now discuss how the RoCourseNet framework is general enough to be used with other parametric CF explanation methods; in fact, we illustrate how the *RoCourseNet* framework can be used to improve the robustness of any parametric CF explanation method.

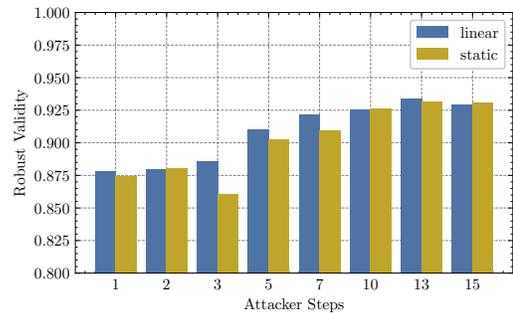


Figure 6: Impact of epsilon scheduler on the robustness. The curriculum training boosts the robustness of recourses.

Note that the VDS algorithm and its corresponding adversarial training procedure (Algorithms 1 & 2) require access to (i) the training dataset  $(x, y) \in \mathcal{D}$  and (ii) the weights of the predictive models  $\theta_f$ . Therefore, any parametric CF explanation method that satisfies these assumptions can leverage these algorithms to improve their robustness to data shift.

Thus, while in this paper, we have made a conscious decision of choosing CounterNet as the CF explanation method of choice within the RoCourseNet framework (since CounterNet’s end-to-end architecture addresses the limitations of post-hoc approaches); in general, the RoCourseNet framework is model-agnostic, and it can work with any parametric model based CF explanation method.

**Table 3: Evaluating robustness under model shift with generalized parametric models on the *Adult* dataset. The **ROBUST** training improves both validity and robust validity.**

Methods	Metrics		
	Prox.	Val.	Rob-Val.
CF MODEL	5.753 ± 0.647	0.826 ± 0.1825	0.607 ± 0.167
ROBUST CF MODEL	0.739 ± 1.078	<b>0.949 ± 0.022</b>	<b>0.906 ± 0.119</b>
VAE-CF	8.531 ± 1.241	0.745 ± 0.272	0.692 ± 0.251
ROBUST VAE-CF	9.473 ± 0.962	<b>0.818 ± 0.186</b>	<b>0.807 ± 0.189</b>
RoCOURSENET	6.611 ± 0.418	<b>0.996 ± 0.002</b>	<b>0.969 ± 0.106</b>

We formulate a slightly altered version of Eq. 4 as the objective function to use the RoCourseNet framework:

$$\begin{aligned}
 & \underset{\theta, \theta_g}{\operatorname{argmin}} \frac{1}{N} \sum_{(x_i, y_i) \in \mathcal{D}} \left[ \lambda_3 \cdot \underbrace{\mathcal{L}(x_i, g(x_i; \theta_g))}_{\text{Proximity Loss } (L_3)} \right] \\
 & + \max_{\delta, \forall \delta_i \in \Delta} \frac{1}{N} \sum_{(x_i, y_i) \in \mathcal{D}} \left[ \lambda_2 \cdot \underbrace{\mathcal{L}(f(x_i^{\text{cf}}, \theta'_{\text{opt}}(\delta)), 1 - f(x_i; \theta))}_{\text{Robust Validity Loss } (L_2)} \right] \\
 & \text{s.t., } \theta'_{\text{opt}}(\delta) = \underset{\theta'}{\operatorname{argmin}} \frac{1}{N} \sum_{(x_i, y_i) \in \mathcal{D}} \left[ \mathcal{L}(f(x_i + \delta_i; \theta'), y_i) \right]. \tag{5}
 \end{aligned}$$

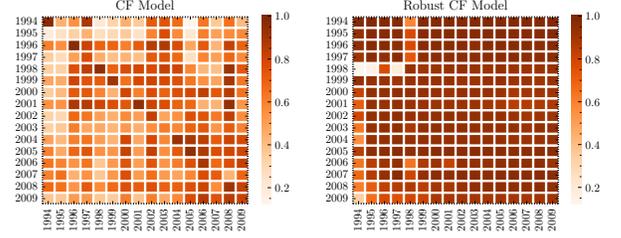
where  $g(\cdot; \theta_g)$  represents a CF model  $g: \mathcal{X} \rightarrow \mathcal{X}^{\text{cf}}$  parametrized by  $\theta_g$ . Note that Eq. 5 is almost identical to Eq. 4, except that the prediction loss ( $\mathcal{L}_1$ ) is ignored, as post-hoc parametric CF explanation methods do not jointly train predictions and CF explanations (instead, they assume access to a pre-trained ML model).

To solve Eq. 5, we can apply a similar strategy used while training RoCourseNet: for each mini-batch datapoints  $\{x^{(i)}, y^{(i)}\}^m$ , (i) we solve the inner bi-level problem to obtain the worst-shift weight of the predictive model  $\theta'_f$  by applying the VDS algorithm (outlined in Algorithm 1); (ii) next, we solve the outer minimization problem by optimizing against the shifted predictive model.

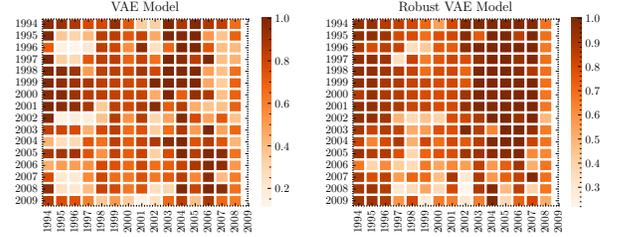
**Experiment Setting.** To demonstrate the generalizability of the RoCourseNet framework, we experiment with two post-hoc parametric CF explanation methods and their “robustified” models.

- CF MODEL is a multi-layer perceptron model which outputs recourses  $x^{\text{cf}}$  given  $x$ . This model is optimized for validity and proximity. We also train a ROBUST CF MODEL to generate robust recourses using the RoCourseNet framework with CF MODEL.
- VAE-CF is a well-known parametric method that uses variational auto-encoder (VAE) to generate recourses  $x^{\text{cf}}$  given input  $x$ . Similarly, ROBUST VAE-CF utilizes the RoCourseNet framework to generate robust recourses with CF MODEL.

For fair comparison, the architecture of CF MODEL is the same as the network that combines the encoder and CF generator. Our experiment is evaluated on the *Loan* dataset, which follows the same experiment settings in Section 4 (e.g., data partitioning, metrics, etc).



(a) Robust validity matrix of CF Model & Robust CF Model.



(b) Robust validity matrix of VAE-CF & robust VAE-CF.

**Figure 7: Comparing the robust validity matrix between (left) normal and (right) adversarial training on two parametric CF methods. Darker color indicates higher robust validity. The proposed adversarial training improves recourse robustness.**

**Empirical Results.** Table 3 compares the (robust) CF MODEL and VAE-CF with RoCourseNet (the best performing method). The results demonstrate two key findings: (i) First, our proposed tri-level robust training (in Algorithm 2) is general purpose and can be plugged in as-is to improve the robustness of post-hoc parametric CF methods. In particular, applying robust training to both CF MODEL and VAE-CF improves the robust validity of these models by  $\sim 32.5\%$  and  $17.3\%$  (on average), respectively. Figure 7 further illustrates this finding, where robustly trained methods (via the generalized RoCourseNet framework) generate recourses with higher robust validity (i.e., the robust validity matrix has more darker colored elements). (ii) Additionally, the design of RoCourseNet, utilizing the design of CounterNet [20], proves to be effective in balancing the cost-invalidity trade-off, as we observe that RoCourseNet outperforms ROBUST CF MODEL and ROBUST VAE-CF in proximity, validity, and robust validity. The result shows the advantages of RoCourseNet’s joint training of prediction and robust recourses.

## 6 CONCLUSION

We present *RoCourseNet*, an end-to-end training framework to generate predictions and robust CF explanations. We formulate this robust end-to-end training as a tri-level optimization problem, and leverage novel adversarial training techniques to solve this problem. Empirical results show that RoCourseNet outperforms state-of-the-art baselines in robust validity, and achieves better balance on the cost-validity trade-off. We further demonstrate that the RoCourseNet training framework is generalizable to be applied with any parametric CF explanation method.

## REFERENCES

- [1] David Alvarez-Melis and Tommi S Jaakkola. 2018. On the robustness of interpretability methods. *arXiv preprint arXiv:1806.08049* (2018).
- [2] Arthur Asuncion and David Newman. 2007. UCI machine learning repository.
- [3] Solon Barocas, Andrew D Selbst, and Manish Raghavan. 2020. The hidden assumptions behind counterfactual explanations and principal reasons. In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*. 80–89.
- [4] Richard Bellman. 1961. *Adaptive Control Processes*. Princeton University Press.
- [5] P Bickel, P Diggle, S Fienberg, U Gather, I Olkin, and S Zeger. 2009. *Springer series in statistics*. Springer.
- [6] Emily Black, Zifan Wang, and Matt Fredrikson. 2022. Consistent Counterfactuals for Deep Models. In *International Conference on Learning Representations*. <https://openreview.net/forum?id=St6eyiTEHnG>
- [7] Qi-Zhi Cai, Chang Liu, and Dawn Song. 2018. Curriculum adversarial training. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence*. 3740–3747.
- [8] Jinghui Chen, Yu Cheng, Zhe Gan, Quanquan Gu, and Jingjing Liu. 2022. Efficient robust training via backward smoothing. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*.
- [9] Paulo Cortez and Alice Maria Gonçalves Silva. 2008. Using data mining to predict secondary school student performance. (2008).
- [10] Jordan Jimmy Crabbe. 2013. *Handling the curse of dimensionality in multivariate kernel density estimation*. Oklahoma State University.
- [11] Kedar Dhamdhere, Mukund Sundararajan, and Qi Qi Yan. 2018. How important is a neuron? *arXiv preprint arXiv:1805.12233* (2018).
- [12] Ricardo Dominguez-Olmedo, Amir H Karimi, and Bernhard Schölkopf. 2022. On the adversarial robustness of causal algorithmic recourse. In *International Conference on Machine Learning*. PMLR, 5324–5342.
- [13] Chelsea Finn, Pieter Abbeel, and Sergey Levine. 2017. Model-agnostic meta-learning for fast adaptation of deep networks. In *International conference on machine learning*. PMLR, 1126–1135.
- [14] Hidde Fokkema, Rianne de Heide, and Tim van Erven. 2022. Attribution-based Explanations that Provide Recourse Cannot be Robust. *arXiv preprint arXiv:2205.15834* (2022).
- [15] Yinghua Gao, Dongxian Wu, Jingfeng Zhang, Guanhao Gan, Shu-Tao Xia, Gang Niu, and Masashi Sugiyama. 2022. On the Effectiveness of Adversarial Training against Backdoor Attacks. *arXiv preprint arXiv:2202.10627* (2022).
- [16] Jonas Geiping, Liam Fowl, Gowthami Somepalli, Micah Goldblum, Michael Moeller, and Tom Goldstein. 2021. What Doesn't Kill You Makes You Robust (er): Adversarial Training against Poisons and Backdoors. *arXiv preprint arXiv:2102.13624* (2021).
- [17] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. 2014. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572* (2014).
- [18] Edward Grefenstette, Brandon Amos, Denis Yarats, Phu Mon Htut, Artem Molchanov, Franziska Meier, Douwe Kiela, Kyunghyun Cho, and Soumith Chintala. 2019. Generalized Inner Loop Meta-Learning. *arXiv preprint arXiv:1910.01727* (2019).
- [19] Alex Gu, Songtao Lu, Parikshit Ram, and Lily Weng. 2022. Min-Max Bilevel Multi-objective Optimization with Applications in Machine Learning. *arXiv preprint arXiv:2203.01924* (2022).
- [20] Hangzhi Guo, Thanh Nguyen, and Amulya Yadav. 2021. CounterNet: End-to-End Training of Counterfactual Aware Predictions. In *ICML 2021 Workshop on Algorithmic Recourse*.
- [21] W Ronny Huang, Jonas Geiping, Liam Fowl, Gavin Taylor, and Tom Goldstein. 2020. Metapoisson: Practical general-purpose clean-label data poisoning. *Advances in Neural Information Processing Systems* 33 (2020), 12080–12091.
- [22] Amir-Hossein Karimi, Gilles Barthe, Bernhard Schölkopf, and Isabel Valera. 2020. A survey of algorithmic recourse: definitions, formulations, solutions, and prospects. *arXiv preprint arXiv:2010.04050* (2020).
- [23] Amir-Hossein Karimi, Bernhard Schölkopf, and Isabel Valera. 2021. Algorithmic recourse: from counterfactual explanations to interventions. In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*. 353–362.
- [24] Been Kim, Martin Wattenberg, Justin Gilmer, Carrie Cai, James Wexler, Fernanda Viegas, et al. 2018. Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (tcav). In *International conference on machine learning*. PMLR, 2668–2677.
- [25] Pang Wei Koh and Percy Liang. 2017. Understanding black-box predictions via influence functions. In *International conference on machine learning*. PMLR, 1885–1894.
- [26] Thibault Laugel, Xavier Renard, Marie-Jeanne Lesot, Christophe Marsala, and Marcin Detyniecki. 2018. Defining locality for surrogates in post-hoc interpretability. *arXiv preprint arXiv:1806.07498* (2018).
- [27] Klas Leino, Shayak Sen, Anupam Datta, Matt Fredrikson, and Linyi Li. 2018. Influence-directed explanations for deep convolutional networks. In *2018 IEEE International Test Conference (ITC)*. IEEE, 1–8.
- [28] Min Li, Amy Mickel, and Stanley Taylor. 2018. “Should This Loan be Approved or Denied?”: A Large Dataset with Class Assignment Guidelines. *Journal of Statistics Education* 26, 1 (2018), 55–66.
- [29] Scott M Lundberg and Su-In Lee. 2017. A unified approach to interpreting model predictions. In *Advances in neural information processing systems*. 4765–4774.
- [30] Dougal Maclaurin, David Duvenaud, and Ryan Adams. 2015. Gradient-based hyperparameter optimization through reversible learning. In *International conference on machine learning*. PMLR, 2113–2122.
- [31] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. 2017. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083* (2017).
- [32] Divyat Mahajan, Chenhao Tan, and Amit Sharma. 2019. Preserving causal constraints in counterfactual explanations for machine learning classifiers. *arXiv preprint arXiv:1912.03277* (2019).
- [33] Saumitra Mishra, Sanghamitra Dutta, Jason Long, and Daniele Magazzeni. 2021. A survey on the robustness of feature importance and counterfactual explanations. *arXiv preprint arXiv:2111.00358* (2021).
- [34] Ramaravind K Mothilal, Amit Sharma, and Chenhao Tan. 2020. Explaining machine learning classifiers through diverse counterfactual explanations. In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*. 607–617.
- [35] Thomas Nagler and Claudia Czado. 2016. Evading the curse of dimensionality in nonparametric density estimation with simplified vine copulas. *Journal of Multivariate Analysis* 151 (2016), 69–89.
- [36] Tuan-Duy Hien Nguyen, Ngoc Bui, Duy Nguyen, Man-Chung Yue, and Viet Anh Nguyen. 2022. Robust Bayesian Recourse. In *The 38th Conference on Uncertainty in Artificial Intelligence*.
- [37] Martin Pawelczyk, Klaus Broelemann, and Gjergji Kasneci. 2020. Learning model-agnostic counterfactual explanations for tabular data. In *Proceedings of The Web Conference 2020*. 3126–3132.
- [38] Martin Pawelczyk, Klaus Broelemann, and Gjergji Kasneci. 2020. On counterfactual explanations under predictive multiplicity. In *Conference on Uncertainty in Artificial Intelligence*. PMLR, 809–818.
- [39] Kaivalya Rawal, Ece Kamar, and Himabindu Lakkaraju. 2020. Algorithmic Recourse in the Wild: Understanding the Impact of Data and Model Shifts. *arXiv preprint arXiv:2012.11788* (2020).
- [40] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2016. “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*. 1135–1144.
- [41] Cynthia Rudin. 2019. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature Machine Intelligence* 1, 5 (2019), 206–215.
- [42] Aniruddha Saha, Akshayvarun Subramanya, and Hamed Pirsiavash. 2020. Hidden trigger backdoor attacks. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 34. 11957–11965.
- [43] Ludwig Schmidt, Shibani Santurkar, Dimitris Tsipras, Kunal Talwar, and Aleksander Madry. 2018. Adversarially robust generalization requires more data. *Advances in neural information processing systems* 31 (2018).
- [44] Amirreza Shaban, Ching-An Cheng, Nathan Hatch, and Byron Boots. 2019. Truncated back-propagation for bilevel optimization. In *The 22nd International Conference on Artificial Intelligence and Statistics*. PMLR, 1723–1732.
- [45] Ali Shafahi, Mahyar Najibi, Mohammad Amin Ghiasi, Zheng Xu, John Dickerson, Christoph Studer, Larry S Davis, Gavin Taylor, and Tom Goldstein. 2019. Adversarial training for free! *Advances in Neural Information Processing Systems* 32 (2019).
- [46] Dylan Slack, Sophie Hilgard, Emily Jia, Sameer Singh, and Himabindu Lakkaraju. 2020. Fooling lime and shap: Adversarial attacks on post hoc explanation methods. In *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*. 180–186.
- [47] Ilija Stepin, Jose M Alonso, Alejandro Catala, and Martín Pereira-Fariña. 2021. A survey of contrastive and counterfactual explanation generation methods for explainable artificial intelligence. *IEEE Access* 9 (2021), 11974–12001.
- [48] Sohini Upadhyay, Shalmali Joshi, and Himabindu Lakkaraju. 2021. Towards robust and reliable algorithmic recourse. *Advances in Neural Information Processing Systems* 34 (2021), 16926–16937.
- [49] Berk Ustun, Alexander Spangher, and Yang Liu. 2019. Actionable recourse in linear classification. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*. 10–19.
- [50] Arnaud Van Looveren and Janis Klaise. 2019. Interpretable counterfactual explanations guided by prototypes. *arXiv preprint arXiv:1907.02584* (2019).
- [51] Sahil Verma, John Dickerson, and Keegan Hines. 2020. Counterfactual Explanations for Machine Learning: A Review. *arXiv preprint arXiv:2010.10596* (2020).
- [52] Sandra Wachter, Brent Mittelstadt, and Chris Russell. 2017. Counterfactual explanations without opening the black box: Automated decisions and the GDPR. *Harv. JL & Tech.* 31 (2017), 841.
- [53] Yisen Wang, Xingjun Ma, James Bailey, Jinfeng Yi, Bowen Zhou, and Quanquan Gu. 2019. On the Convergence and Robustness of Adversarial Training. In *International Conference on Machine Learning*. PMLR, 6586–6595.
- [54] Eric Wong, Leslie Rice, and J Zico Kolter. 2019. Fast is better than free: Revisiting adversarial training. In *International Conference on Learning Representations*.

- [55] Fan Yang, Sahan Suresh Alva, Jiahao Chen, and Xia Hu. 2021. Model-Based Counterfactual Synthesizer for Interpretation. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining* (Virtual Event,

Singapore) (*KDD '21*). Association for Computing Machinery, New York, NY, USA, 1964–1974. <https://doi.org/10.1145/3447548.3467333>

## A IMPLEMENTATION DETAILS

Here we provide implementation details of RoCourseNet and three baseline methods on three datasets listed in Section 4. The code can be found through this anonymous repository (<https://www.dropbox.com/s/gsrpt55hf2ik7v4/RoCourseNet.zip?dl=0>).

**Feature Engineering.** We follow the feature engineering procedure of CounterNet [20]. Specifically, for continuous features, we scale all feature values into the  $[0, 1]$  range. To handle the categorical features, we customize model architecture for each dataset. First, we transform the categorical features into numerical representations via one-hot encoding. In addition, for each categorical feature, we add a softmax layer after the final output layer in the CF generator, which ensures that the generated CF examples respect the one-hot encoding format.

**Hyperparameters.** For all three datasets, we train the model for up to 50 epochs with Adam. We set dropout rate to 0.3 to prevent overfitting. We use  $T = 7$  and  $E = 0.1$  to report results in Table 2, and report the impact of attacker steps  $T$  and maximum perturbation  $E$  to robustness in Figure 4. We use  $K = 2$  unrolling steps (same as [21]) with the step size  $\alpha = 2.5 \times \delta/T$  (based on [31]) for solving the bi-level problem in Equation 3 (via VDS). In addition, Table 4 reports the hyperparameters chosen for each dataset, and Table 5 specifies the architecture used for each dataset.

**Software and Hardware Specifications.** We use Python (v3.7) with Pytorch (v1.82), Pytorch Lightning (v1.10), numpy (v1.19.3), pandas (v1.1.1), scikit-learn (v0.23.2) and higher (v0.2.1) [18] for the implementations. All our experiments were run on a Debian-10 Linux-based Deep Learning Image on the Google Cloud Platform. The RoCourseNet and baseline methods are trained (or optimized) on a 16-core Intel machine with 64 GB of RAM.

## B ADDITIONAL EXPERIMENTAL ANALYSIS

### B.1 Predictive Performance

We first show that, similar to CounterNet, the training of RoCourseNet does not come at the cost of degraded predictive accuracy. Table 6 & 7 compare RoCourseNet’s predictive accuracy and AUC score against the base prediction model used by baselines. This table shows that RoCourseNet achieves competitive predictive performance – it achieves marginally better accuracy than the base model ( $\sim 2\%$ ). Thus, we conclude that the joint training of RoCourseNet does not come at a cost of reduced predictive performance.

### B.2 Heuristic Baselines

We provide two heuristic baseline methods to further illustrate the challenge of generating robust recourses under the distribution shift scenarios. *VanillaCF-Random* aims to generate robust recourse by adding a small perturbation to input. In addition, *RoCourseNet-Random* optimizes for robust CF generator against a random perturbation attacker.

Table 8 compares heuristic baselines with RoCourseNet. Both baseline methods perform significantly worse than RoCourseNet in validity, robust validity and proximity. This experiment further highlights the hardness of generating robust recourses as simple heuristics drastically underperform as compared to RoCourseNet.

## B.3 Simulated Data Shift

We conduct simulated experiments with covariant and label shift.

*Covariant Shift.* We simulate the covariant shift via this Bayesian network:

$$\begin{aligned} x_1 &\sim \mathcal{N}(\mu_1, \sigma_1) \\ x_2 &\sim \mathcal{N}(\mu_2, \sigma_2) \\ y &= -x_2 + x_1^3 + \varepsilon, \quad \varepsilon \sim \mathcal{N}(-0.1, 0.1) \end{aligned}$$

where we set  $\mu_1 = 0.5, \sigma_1 = 0.5, \mu_2 = 0, \sigma_2 = 0.3$  for  $D_1$ , and  $\mu_1 = 0, \sigma_1 = 0.3, \mu_2 = 0.5, \sigma_2 = 0.5$  for  $D_2$ . Figure 8 illustrates this simulation dataset.

*Label Shift.* Similarly, we simulate the label shift via this Bayesian network:

$$\begin{aligned} y &\sim \text{binomial}(p) \\ z &= 2y - 1 + \varepsilon, \quad \varepsilon \sim \mathcal{N}(0.1, 0.1) \\ x_1 &\sim \mathcal{N}(-z + z^3, 0.3) \\ x_2 &\sim \mathcal{N}(z + z^3 - 3y, 0.3) \end{aligned}$$

where we set  $p = 0.6$  for  $D_1$ , and  $p = 0.3$  for  $D_2$ . Figure 9 illustrates this simulation dataset.

*Experimental Results.* Table 9 shows that RoCourseNet achieves 100% validity and robust validity under both covariate and label shifts.

*Validity Matrix of CounterNet and RoCourseNet.* Figure 10 shows the validity matrix of CounterNet and RoCourseNet.

## C ABLATIONS OF ROCOURSENET

### C.1 Training Loss Curve of RoCourseNet

Figure 11 shows RoCourseNet’s training curve. Importantly, the prediction loss  $\mathcal{L}_1$  and the proximity loss  $\mathcal{L}_3$  are smoothly optimized during the training. The robust validity loss  $\mathcal{L}_2$  encounters fluctuations in the early stage of training, but starts to converge after 10 epochs.

### C.2 $l_2$ -norm Projection in Algorithm 1

We provide supplementary results on adopting  $\Delta$  as the  $l_2$ -norm ball (i.e.,  $\Delta = \{\delta \in \mathbb{R}^n \mid \|\delta\|_2 \leq \epsilon\}$ ) for the maximum perturbation constrains. Figure 12 highlights the results of using the  $l_2$ -norm ball in attacking and adversarial training. We observe similar patterns in Figure 4. Thus, this result shows that  $l_\infty$ -norm constrain can be substitute to other feasible region.

## D TIME-COMPLEXITY ANALYSIS

### D.1 Training Time

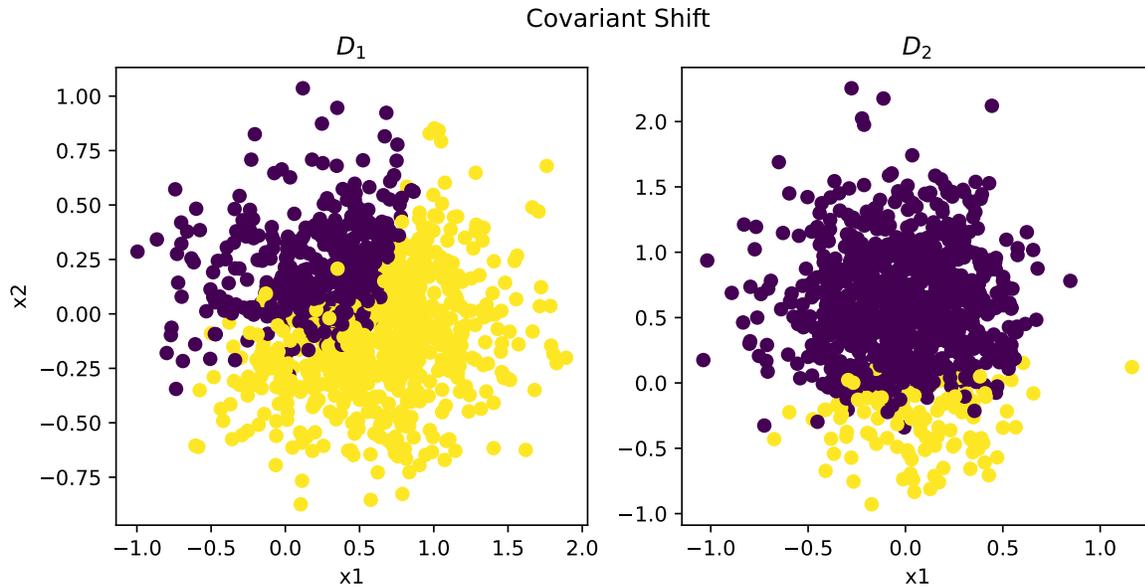
RoCourseNet takes only  $\sim 10$  more minutes of training (as compared to CounterNet) on the Loan dataset (our largest-sized dataset). This is quite reasonable since training time is a one-time up-front cost; after RoCourseNet is trained, test-time inference happens in milliseconds. Table 10 shows the training time of these two models.

**Table 4: Hyperparameters setting for each dataset.**

Dataset	Learning Rate	$\eta$	Batch Size	$\lambda_1$	$\lambda_2$	$\lambda_3$
Loan	0.003	0.03	128	1.0	0.2	0.1
German Credit	0.003	0.03	256	1.0	1.0	0.1
Student	0.01	0.01	128	1.0	0.2	0.1

**Table 5: Architecture specification of RoCourseNet for each dataset.**

Dataset	Encoder Dims	Predictor Dims	CF Generator Dims
Loan	[110,200,10]	[10, 10]	[10, 10]
German Credit	[19, 100,10]	[10, 20]	[10, 20]
Student	[83,50,10]	[10, 10]	[10, 50]

**Figure 8: Illustration of covariant shift.****Table 7: AUC score for each dataset.**

Dataset	Base Model	RoCourseNet
Loan	$0.897 \pm 0.026$	$0.900 \pm 0.027$
German Credit	$0.662 \pm 0.018$	$0.729 \pm 0.010$
Student	$0.913 \pm 0.012$	$0.947 \pm 0.018$

**Table 6: Predictive accuracy for each dataset.**

Dataset	Base Model	RoCourseNet
Loan	$0.886 \pm 0.036$	$0.885 \pm 0.035$
German Credit	$0.714 \pm 0.003$	$0.742 \pm 0.014$
Student	$0.914 \pm 0.028$	$0.906 \pm 0.066$

## D.2 Inference Time

Inference runtime is an important metric as recourses are user-facing. Table 11 shows the inference runtime of RoCourseNet and baseline methods. Importantly, CounterNet and RoCourseNet achieve the same amount of inference time (as they share the same network structure). On the other hand, ROAR and RBR (two non-parametric methods) take significantly more time (i.e.,  $\sim 200X$  and  $\sim 500X$  runtime as compared to RoCourseNet, respectively).

## E DISCUSSION ABOUT MULTI-CLASS CLASSIFICATION

Existing CF explanation literature focuses on evaluating methods under the binary classification settings [20, 32, 34, 48]. However, these CF explanation methods can be adapted to the multi-class classification settings. Given an input instance  $x \in \mathbb{R}^d$ , the RoCourseNet generates (i) a prediction  $\hat{y}_x \in \mathbb{R}^k$  for input instance  $x$ ,

**Table 8: Heuristic baselines as compared to RoCourseNet on Loan dataset. Simple heuristic does not defend against distribution shift.**

Method	Validity	Robust Validity	Proximity
VANILLACF-RANDOM	0.634±0.270	0.510±0.251	9.446±1.042
ROCOURSENET-RANDOM	0.856 ±0.127	0.856 ±0.127	10.405 ± 1.857
ROCOURSENET	<b>0.996 ± 0.002</b>	<b>0.969 ± 0.106</b>	<b>6.611 ± 0.418</b>

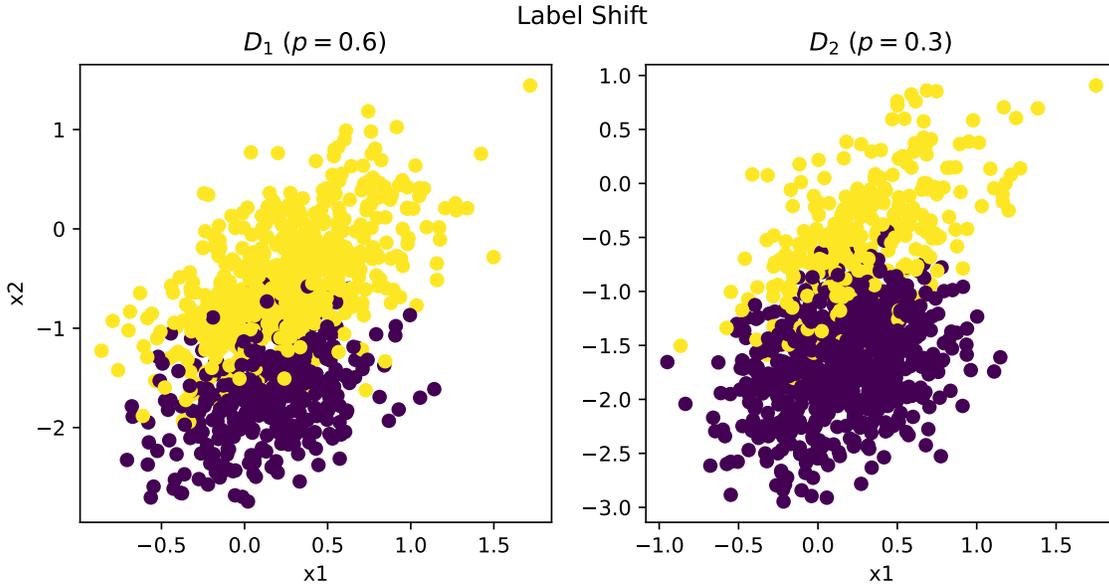


Figure 9: Illustration of label shift.

Table 9: RoCourseNet on simulated data shifts.

Data Shift	Validity	Robust Validity	Proximity
Covariant	1.00	1.00	0.389 ± 0.051
Label	1.00	1.00	0.425 ± 0.012

Table 10: Training time of CounterNet and RoCourseNet on the Loan dataset (the largest dataset).

Dataset	CounterNet	RoCourseNet
Loan	22m 6s	32m 16s
German Credit	46s	1m 37s
Student	55s	2m 19s

Table 11: Inference time for generating a single CF example on the Loan dataset (in milliseconds).

Dataset	ROAR	RBR	CounterNet	RoCourseNet
Loan	131.38	345.91	0.67	0.67
German Credit	104.87	271.34	0.51	0.51
Student	213.58	555.91	1.00	1.01

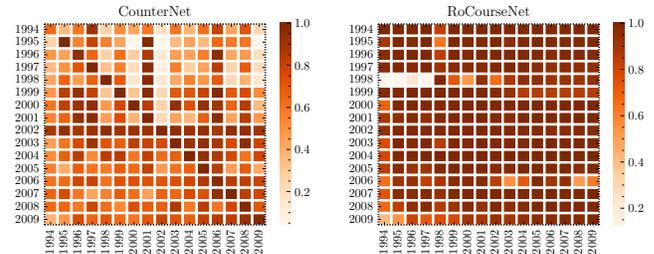
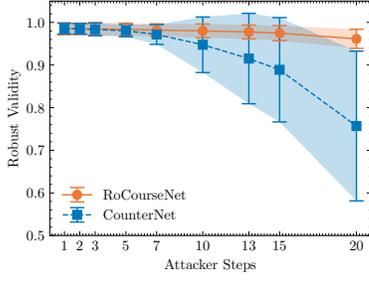
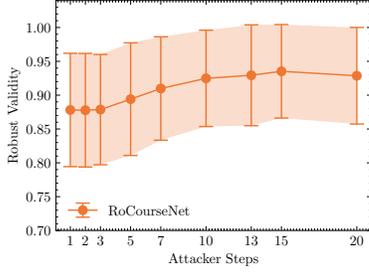


Figure 10: Validity matrix of CounterNet and RoCourseNet. RoCourseNet significantly improves the recourse robustness.

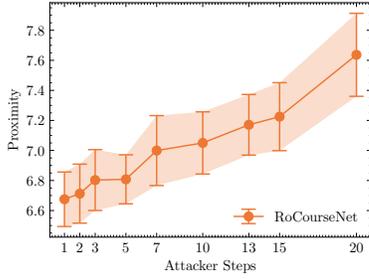
and (ii) a CF example  $x^{cf}$  as an explanation for input instance  $x$ . The prediction  $\hat{y}_x \in \mathbb{R}^k$  is encoded as one-hot format as  $\hat{y}_x \in \{0, 1\}^k$ , where  $\sum_i \hat{y}_x^{(i)} = 1$ ,  $k$  denotes the number of classes. In addition, we assume a desired outcome  $y'$  for every input instances  $x$ . As such,



(a) The number of attacker steps  $T$  vs attacker effectiveness ( $\downarrow$ )

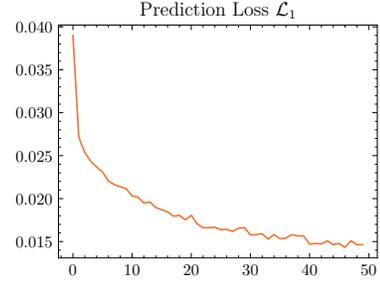


(b) The number of attacker steps  $T$  vs robust validity ( $\uparrow$ ).

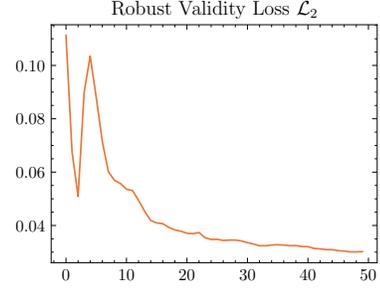


(c) The number of attacker steps  $T$  vs proximity ( $\downarrow$ ).

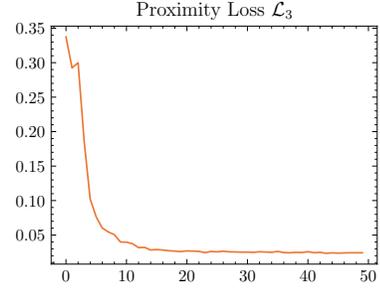
Figure 12: The impact of the number of attacker steps  $T$  under the  $l_2$ -norm constrains.



(a)  $\mathcal{L}_1$  training curve.



(b)  $\mathcal{L}_2$  training curve.



(c)  $\mathcal{L}_3$  training curve.

Figure 11: Training loss curves of RoCourseNet on the Loan dataset.

we can adapt Eq. 4 for binary settings to the multi-class settings as follows:

$$\begin{aligned}
 & \underset{\theta, \theta_g}{\operatorname{argmin}} \frac{1}{N} \sum_{(x_i, y_i) \in \mathcal{D}} \left[ \underbrace{\lambda_1 \cdot \mathcal{L}(f(x_i; \theta), y_i)}_{\text{Prediction Loss } (L_1)} + \lambda_3 \cdot \underbrace{\mathcal{L}(x_i, x_i^{\text{cf}})}_{\text{Proximity Loss } (L_3)} \right] \\
 & + \max_{\delta, \forall \delta_i \in \Delta} \frac{1}{N} \sum_{(x_i, y_i) \in \mathcal{D}} \left[ \underbrace{\lambda_2 \cdot \mathcal{L}(f(x_i^{\text{cf}}; \theta'_{\text{opt}}(\delta)), y')}_{\text{Robust Validity Loss } (L_2)} \right] \\
 & \text{s.t. } \theta'_{\text{opt}}(\delta) = \underset{\theta'}{\operatorname{argmin}} \frac{1}{N} \sum_{(x_i, y_i) \in \mathcal{D}} \left[ \mathcal{L}(f(x_i + \delta; \theta'), y_i) \right], \\
 & x_i^{\text{cf}} = g(x_i; \theta_g).
 \end{aligned} \tag{6}$$

To optimize for Eq. 6, we can follow the same procedure outlined in Algorithm 2. For each sampled batch, we first optimize for the predictive accuracy  $\theta' = \theta - \nabla_{\theta}(\lambda_1 \cdot L_1)$ . Next, we use the VDS algorithm to optimize for the inner max-min bi-level problem (in Algorithm 1). Finally, we optimize for the CF explanations by updating the model's weight as  $\theta''_g = \theta'_g - \nabla_{\theta'_g}(\lambda_2 \cdot L_2 + \lambda_3 \cdot L_3)$ .