Generating robust counterfactual explanations

Victor Guyomard, Françoise Fessant, Thomas Guyet, Tassadit Bouadi, Alexandre Termier

Generating robust counterfactual explanations

Victor Guyomard[1,2], Françoise Fessant[1], Thomas Guyet[3], Tassadit Bouadi[2], and Alexandre Termier[2]

[1] Orange Innovation, Lannion, France
victor.guyomard@orange.com
[2] Univ Rennes, Inria, CNRS, IRISA, Rennes, France
[3] Inria, AIstroSight, France

Abstract. Counterfactual explanations have become a mainstay of the XAI field. This particularly intuitive statement allows the user to understand what small but necessary changes would have to be made to a given situation in order to change a model prediction. The quality of a counterfactual depends on several criteria: realism, actionability, validity, robustness, etc. In this paper, we are interested in the notion of robustness of a counterfactual. More precisely, we focus on robustness to counterfactual input changes. This form of robustness is particularly challenging as it involves a trade-off between the robustness of the counterfactual and the proximity with the example to explain. We propose a new framework, CROCO, that generates robust counterfactuals while managing effectively this trade-off, and guarantees the user a minimal robustness. An empirical evaluation on tabular datasets confirms the relevance and effectiveness of our approach.

Keywords: Counterfactual explanation · Robustness · Algorithmic recourse

1 Introduction

The ever-increasing use of machine learning models in critical decision-making contexts, such as health care, hiring processes or credit allocation, makes it essential to provide explanations for the individual decisions made by these models. To this end, Wachter et al. proposed counterfactual explanation [22]. A counterfactual is defined as the smallest modification of feature values that changes the prediction of a model to a given output. The counterfactual can provide actions (or recourse) for individuals to attain more desirable outcomes. This is particularly important in areas where decisions made by algorithms can have significant impacts on people's lives such as finance, health care or criminal justice. Many methods have been proposed to generate counterfactuals, focusing on some specific properties such as realism [14,20,7], actionability [19,16] or sparsity [3,22,11]. According to Artelt et al. [1], many counterfactual generation methods are vulnerable to small changes, where even a minor change in the value of a counterfactual feature can cause the counterfactual to have a different outcome. Such a situation may arise for example in practical implementation of the

counterfactual, due to various factors such as unexpected noise, or adversarial manipulation. As an illustration, a counterfactual may suggest to an individual to raise its salary by 200$ to obtain a credit, but in practice, the salary is increased by 199$ or 201$, potentially resulting in a negative decision (a rejected credit) regarding the decision model. This line of discussions falls into the topic of robustness [15,4,21,9]. To address robustness in the context of counterfactual explanation, Pawelcyk et al. [15] introduce the notion of recourse invalidation rate which represents the probability of obtaining a counterfactual with a different predicted class, when small changes (sampled from a noise distribution) are applied to it. They presented an estimator of the recourse invalidation rate in the context of Gaussian distributions, and also a framework (PROBE) that guarantees the recourse invalidation rate to be no greater than a target specified by the user. A limitation of their approach is that the satisfaction of the user condition is dependent of the estimator quality, which means that in practice, the recourse invalidation rate can be greater than the target fixed by the user. Moreover, PROBE leads in practice to a poor trade-off management between proximity and robustness i.e the counterfactual is robust but far from the example to explain. In this paper, we introduce a framework called CROCO (Cost-efficient RObust COunterfactuals), which is based on a new minimization problem inspired by PROBE [15]. Our framework introduces the novel concept of soft recourse invalidation rate, as well as an estimator of it. It enables us to derive an upper-bound for the recourse invalidation rate with almost certain probability. This ensures that the user obtains a solution with a recourse invalidation rate lower than the predetermined target. An experimental evaluation on different tabular datasets confirms these theoretical results, and shows that our method better optimizes the two criteria of robustness and proximity.

2 Related work

Since Wachter et al. seminal paper [22], a variety of counterfactual explanation technics have been proposed. These methods seek to enhance the quality of counterfactuals by incorporating additional properties, such as constraining the counterfactual to support the data distribution in order to produce realistic examples, freezing immutable features (such as race or gender), producing multiple counterfactuals at once, or even adding causality constraints. We refer the readers to Guidotti et al. [6] for a detailed review about counterfactual explanation properties and methods. The property of robustness has been studied recently in the context of counterfactual explanations, where the validity of a counterfactual is determined by its ability to maintain the same predicted class in the presence of changes. Mishra et al. [10] distinguish various types of robustness:

Robustness to model change refers to the evolution of the validity of the counterfactual explanation when machine learning models are re-trained or when training parameters settings are slightly modified. Rawal et al. [17] have demonstrated that state-of-the-art counterfactual generation methods have the tendency to produce solutions that are not robust to model retraining.

To address this problem, Ferrario and Loi [5] proposed to use counterfactual data augmentation every time machine learning models are retrained. Upadhyay et al. [18] for their part developed an adversarial training objective that produces counterfactuals that are robust regarding changes in the training data. More specifically, they evaluated the robustness on different types of training data shift which are data correction shift, temporal shift, and geospatial shift. However, the counterfactuals that are generated suffer from a much higher cost of change regarding state-of-the art counterfactual generation methods [15]. In the context of slightly changed training settings, Black et al. [2] achieved robust counterfactual explanations with a regularization method based upon a K-Lipschitz constant.

Robustness to input perturbations refers to how counterfactuals explanations are sensitive to slight input changes. According to Dominguez-Olmedo et al. [4], a counterfactual is said robust if small changes in the example to explain result in valid counterfactuals. They proposed an optimization problem that applies to linear models and neural networks to generate robust counterfactuals in this context. For Artelt et al. [1] robustness means that two examples that are close, must result in two similar counterfactuals. To address this issue they propose to solve an optimization problem that includes a density constraint [1]. They empirically show that having a counterfactual that lies in a dense area has the effect of improving the robustness. Laugel et al. [8] pointed out that such a type of robustness issue cannot solely be attributed to the explainer, but also arises from the decision boundary of the classifier, thus increasing the problem complexity.

Robustness to counterfactual input changes refers to the ability of a counterfactual explanation to remain valid when small feature changes are applied (two similar counterfactuals should have the same predicted class). In this context, Pawelcyk et al. [15] presented PROBE a framework to produce robust counterfactuals that is based on an optimization problem. This framework aims to find a trade-off between two criteria that are the recourse invalidation rate and the proximity, i.e. the distance between the counterfactual and the example to explain. From their side, Maragno et al. [9] introduced an adversarial robust approach that generates counterfactuals that remain valid in an uncertainty set, meaning that for a given example to explain, all the solutions in the set are valid counterfactuals. This approach works for non-differentiable model unlike PROBE. However there is no trade-off between the recourse invalidation rate and the proximity as all the counterfactuals in the uncertainty set are valid. In such a scenario, the robustness constraint cannot be relaxed, then allowing the generation of counterfactuals that are far from the example to explain. Our approach, CROCO, is part of this category of methods. It is inspired by the PROBE framework, and improves its limitations. Indeed, the major criticism that we can make to PROBE is that the guarantees in terms of robustness that it offers to the user are completely dependent on the quality of their estimator (i.e. the guarantee is based on a recourse invalidation rate approximation rather than the true recourse invalidation rate). Our method introduces a new optimiza-

tion problem that is proved to induce an almost-sure upper bound on the true recourse invalidation rate. This leads to a significant improvement in the trade-off between the robustness of the counterfactual and the proximity with the example to explain.

3 Problem statement

In this section, we define some notations related to the generation of counterfactuals, and we formalize the robustness of counterfactual generation by introducing the notion of *recourse invalidation rate*.

3.1 Generation of counterfactuals

We consider the generation of counterfactuals for a binary classifier. Let $\mathcal{X} \subseteq \mathbb{R}^n$ represents the n-dimensional feature space. A binary classifier is a function $h : \mathcal{X} \to \mathcal{Y}$ where $\mathcal{Y} = \{0, 1\}$. We assume that the classification is obtained from a probabilistic prediction i.e. a function $f : \mathcal{X} \to [0, 1]$ that returns $\hat{p}$ which is the predicted probability for the class 1. Then, the predicted class is the most likely class according to $\hat{p}$. For a given example x, $h(x) = g \circ f(x)$ where $g : [0, 1] \to \mathcal{Y}$ is a function that returns the predicted class from the probability vector. We take $g(u) = \mathbb{1}_{>t}(u)$, where t is the decision threshold. $\mathbb{1}_{>t}(u)$ equals 1 if $u > t$ and 0 otherwise.

In this article, we do post-hoc counterfactual generation, meaning that f (and thus h) are given. And for a given example to explain $x \in \mathcal{X}$, whose decision is $h(x)$, we want to generate a counterfactual $\breve{x} \in \mathcal{X}$. A counterfactual is a new example close to the example to explain x, and with a different prediction, *i.e.* $h(\breve{x}) \neq h(x)$. If it is true that $h(\breve{x}) \neq h(x)$, then $\breve{x}$ is said to be *valid*. A counterfactual $\breve{x}$ is also seen as a change to apply to x: $\breve{x} = x + \delta$ where $\delta \in \mathbb{R}^n$. Thus, a counterfactual is associated to a small change δ that modifies the decision returned by h. Generating a counterfactual is basically solving the following optimization problem:

$$\min_{\delta} \ell\left(f\left(x+\delta\right), 1 - h(x)\right) + \lambda \left\|\delta\right\|_1 \tag{1}$$

where $\ell : [0, 1]^2 \mapsto \mathbb{R}^+$ quantifies the distance between the predicted probability, $f(\breve{x})$, and $1 - h(x)$ that is the opposite of the predicted class for example x. For instance, Wachter et al. suggested ℓ as the L_2 distance, so as to produce counterfactuals that are close to the desired decision [22]. The other term in the optimization problem, constraints the change δ applied to the example x to be small.

In what follows, we will focus specifically on the generation of counterfactuals in the case of instances that have received a negative decision (which corresponds to instances predicted as class 0). This choice has no limitation and is motivated by the fact that the majority of robustness methods are defined in a recourse context [15,17,18] where the goal is to provide explanations only for negatively predicted instances. We will also assume that the classifier f is differentiable.

3.2 Recourse invalidation rate

In order to quantify the robustness of the counterfactual to an input perturbation, the notion of recourse invalidation rate has been introduced by Pawelczyk et al. [15].

Definition 1 (Recourse invalidation rate). *The recourse invalidation rate for a counterfactual $\breve{x}$, of an example x predicted as class 0 can be expressed as:*

$$\Gamma\left(\breve{x}; p_\varepsilon\right) = \mathbb{E}_{\varepsilon \sim p_\epsilon}\left[1 - h\left(\breve{x} + \varepsilon\right)\right]$$

where $\varepsilon \in \mathbb{R}^n$ is a random variable that follows a probability distribution p_ε. Since $h\left(\breve{x} + \varepsilon\right) \in \{0, 1\}$, it ensues $\Gamma(\breve{x}; p_\varepsilon) \in [0, 1]$.

Assuming p_ε is centered, then p_ε defines a region around a counterfactual $\breve{x}$ for *similar* counterfactuals $\breve{x} + \varepsilon$. Intuitively, $\Gamma(\breve{x}; p_\varepsilon)$ gives the rate of *similar* counterfactuals that are not valid, *i.e.* that belong to class 0. Thus, the lower $\Gamma(\breve{x}; p_\varepsilon)$, the more robust is the counterfactual. If $\Gamma\left(\breve{x}; p_\varepsilon\right) = 0$, the counterfactual is considered perfectly robust, given that all the perturbed counterfactuals result in positive outcomes (i.e., there are all predicted as class 1). However, if $\Gamma\left(\breve{x}; p_\varepsilon\right) = 1$, the counterfactual is not at all considered robust, since no noisy counterfactuals lead to positive outcomes (i.e., there are all predicted as class 0).

Figure 1 illustrates the intuition of the recourse invalidation rate. $\Gamma(\breve{x}; p_\varepsilon)$ can be seen as the surface of the neighborhood that overlaps the region, split by the decision frontier, on the side of the example. This neighborhood represents the perturbations on the counterfactuals that we would like to accept without changing its validity. The Figure also shows that finding a robust counterfactual requires to make a trade-off between the robustness and the magnitude of the change.

3.3 The PROBE framework for generating robust counterfactuals

Pawelczyk et al.[15] have developed a framework named PROBE that generates robust counterfactuals regarding the recourse invalidation rate. It adapts the minimization problem of equation 1 by adding a new term that enforces the recourse invalidation rate to be under a target value Γ_t. This target value is chosen by the user. More formally, generating a counterfactual relies on solving the following minimization problem:

$$\min_{\delta} \quad \max\left[\Gamma\left(x + \delta; p_\varepsilon\right) - \Gamma_t,\, 0\right] + \ell\left(f\left(x + \delta\right),\, 1 - h(x)\right) + \lambda \left\|\delta\right\|_1 \tag{2}$$

There are some difficulties with the additional constraint on recourse invalidation rate. Indeed, the true value of Γ can not be evaluated in practice. Then, PROBE proposes a Monte-Carlo estimator of Γ. This means that it is estimated by computing the mean of a sample of perturbations in p_ε:

$$\tilde{\Gamma}\left(\breve{x}; K, p_\varepsilon\right) = \frac{1}{K} \sum_{k=1}^{K} \left(1 - h\left(\breve{x} + \varepsilon_k\right)\right) \tag{3}$$

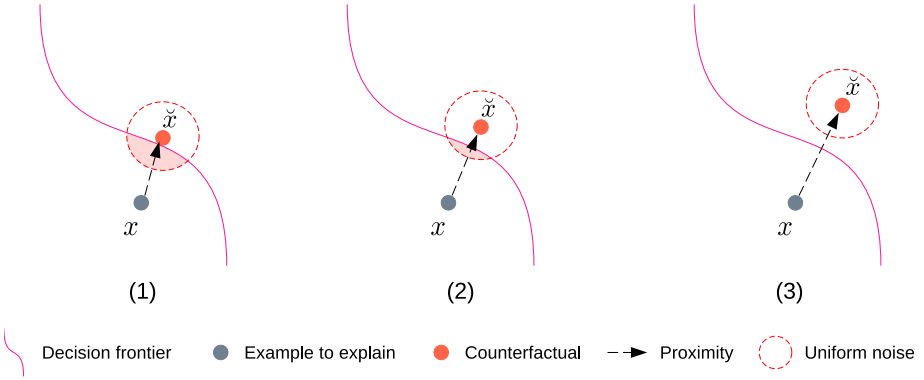

Fig. 1. Illustration of the recourse invalidation rate with a uniform distribution p_ε (dashed-red circle). The recourse invalidation rate is figured out by the area of the region in red. In **(1)** the counterfactual has a low robustness and is at a low distance from the example. In **(2)** the counterfactual has a medium robustness and is at a medium distance, and in **(3)** the counterfactual has a perfect robustness but is far from the example (large distance).

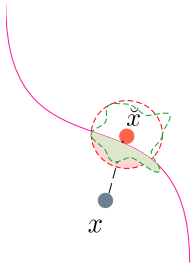

Fig. 2. Illustration of the potential problem with PROBE. The red region illustrates the true recourse invalidation rate (see Figure 1) while the green region illustrates the approximated recourse invalidation rate through the approximation of the red region. In this case, the approximation under-estimates the red region and misleadingly encourages finding a $\breve{x}$ that would break the robustness constraint.

However, $\tilde{\Gamma}$ is non-differentiable, because $h(x) = g \circ f(x)$ and $g(u) = \mathbb{1}_{>t}$. Then, it can not be part of a loss of an optimization problem. To overcome this limitation, the authors proposed a first-order approximation of the true recourse invalidation rate Γ in the context of a Gaussian distribution noise $p_\varepsilon = \mathcal{N}(\mathbf{0}, \sigma\mathbf{I})$, named $\tilde{\Gamma}_{\text{PROBE}}$.

Then, the optimization algorithm solves the problem in eq. 2, replacing Γ by $\tilde{\Gamma}_{\text{PROBE}}$ and stops when the approximation of recourse invalidation rate is under the target value, i.e. when $\tilde{\Gamma}_{\text{PROBE}}(x; \breve{p}_\varepsilon) \leq \Gamma_t$.

Thus, for a given counterfactual $\breve{x}$ returned by PROBE, the user is guaranteed that $\tilde{\Gamma}_{\text{PROBE}}(\breve{x}; p_\varepsilon) \leq \Gamma_t$. However, this means that the guarantee depends on the quality of the estimator. Indeed, it is possible to generate a counterfactual

where $\tilde{\Gamma}_{\text{PROBE}}(\breve{x}; p_\varepsilon) \leq \Gamma_t \leq \Gamma(\breve{x}; p_\varepsilon)$ which would then violate the user-selected guarantee. The intuition behind this situation is depicted in Figure 2.

To sum up, PROBE has two limitations: 1) It offers users a guarantee based on the recourse invalidation rate approximation rather than the true recourse invalidation rate; 2) the approximation applies only for Gaussian distribution of counterfactual perturbation. This makes the approach not applicable to dataset with categorical attributes.

Our contribution overcomes the first limitation by introducing a new estimator that is proved to induce an almost-sure upper bound on the true recourse invalidation rate. Furthermore, our approach is independent to the noise distribution, thus enabling the use of various noise distributions.

4 Our contribution

In this section, we present our method, named CROCO standing for *Cost-efficient RObust COunterfactuals*. It improves the generation of robust counterfactuals according to the recourse invalidation rate.

This method, inspired from PROBE, introduces a new robustness term to the optimization problem presented in Equation 1. This term is based on an upper-bound of the recourse invalidation rate.

4.1 An upper bound of the recourse invalidation rate

As it is not feasible to derive a closed-form expression of Γ without making any assumption about the noise distribution, and given that $\tilde{\Gamma}$ is not differentiable, our idea is to compute an upper-bound of Γ.

Let $\breve{x}$ be a counterfactual for an example $x \in \mathcal{X}$, then we define the soft recourse invalidation rate, $\Theta(\breve{x})$ by:

$$\Theta(\breve{x}; p_\varepsilon) = \mathbb{E}_{\varepsilon \sim p_\varepsilon} \left[1 - f\left(\breve{x} + \varepsilon\right)\right].$$

The proposition 1 states that the soft recourse invalidation rate, Θ, induces an upper-bound of the recourse invalidation rate, Γ.

Proposition 1. [4] *Let $t \in [0, 1]$ be a decision threshold and $\breve{x}$ be a counterfactual for an example $x \in \mathcal{X}$, an upper bound of the true recourse invalidation rate is given by:*

$$\Gamma\left(\breve{x}; p_\varepsilon\right) \leq \frac{\Theta\left(\breve{x}; p_\varepsilon\right)}{(1-t)} \tag{4}$$

Similarly to Γ, Θ can not be evaluated directly. However, we can use the following Monte-Carlo estimator, where K is the number of random samples:

$$\tilde{\Theta}\left(\breve{x}; K, p_\varepsilon\right) = \frac{1}{K} \sum_{k=1}^{K} (1 - f(\breve{x} + \varepsilon_k)) \tag{5}$$

[4] All proofs are provided in Section A.1 of supplementary material.

This quantity can be seen as the mean predicted probability for class 0, computed on perturbed samples that are randomly drawn from the p_ϵ distribution. The proposed estimator is close to the recourse invalidation rate estimation outlined in equation 3, but it differs in that it is differentiable as a composition of differentiable functions, thus can be included in an objective function.

Moreover, the proposition 2 shows that our estimator, $\tilde{\Theta}$, defines an almost-sure upper bound of the true recourse invalidation rate. This means that $\frac{m+\tilde{\Theta}}{1-t}$ has a high probability to be an upper-bound of Γ.

Proposition 2. *Let $t \in [0,1]$ be a decision threshold, p_ε a noise distribution, $\breve{x}$ be a counterfactual for an example $x \in \mathcal{X}$, then an almost-sure upper-bound of the recourse invalidation rate is given by:*

$$\mathbb{P}\left(\Gamma\left(\breve{x}; p_\varepsilon\right) \leq \frac{m + \tilde{\Theta}\left(\breve{x}; K, p_\varepsilon\right)}{1-t}\right) \geq 1 - \exp\left(-2m^2K\right) \tag{6}$$

where $m > 0$ and K is the number of random samples.

With a high number of random samples and a given value of m, the exponential term of proposition 2 can be arbitrarily small. Then for a given value of our estimator $\tilde{\Theta}\left(\breve{x}; K, p_\varepsilon\right)$, we have almost surely that the true recourse invalidation rate will be in the worst case equals to $\frac{m + \tilde{\Theta}\left(\breve{x}; K, p_\varepsilon\right)}{1-t}$. It ensues that if we enforce $\frac{m + \tilde{\Theta}\left(\breve{x}; K, p_\varepsilon\right)}{1-t}$ to be lower than a given threshold $\bar{\Gamma}_t$, then we are almost-sure that the true recourse invalidation rate is lower than $\bar{\Gamma}_t$, *i.e.* that the counterfactual is more robust than the given threshold.

Note that $m \in \mathbb{R}_{>0}$ is a parameter that defines the tightness of the upper-bound. The lower m, the better the upper-bound. In return, low m requires a higher K (*i.e.* more computational resource) to keep the confidence in the bound. Section A.2 in supplementary material provides a table to choose the values of m and K with respect to the desired level of confidence.

For instance, with $K = 500$ and $m = 0.1$, and $t = 0.5$, the inequation of the proposition 2 gives:

$$\mathbb{P}\left(\Gamma\left(\breve{x}\right) \leq 0.2 + 2\tilde{\Theta}\left(\breve{x}\right)\right) \geq 0.999 \tag{7}$$

4.2 Generate robust counterfactuals

We propose a minimization problem for the generation of robust counterfactuals according to the recourse invalidation rate.

Given a neighborhood distribution p_ε, a number of samples K, a tightness value $m > 0$ and a target upper-bound $\bar{\Gamma}_t$, a counterfactual $\breve{x} = x + \delta$ is found by minimizing the following objective function:

$$\min_{\delta} \underbrace{\left(\frac{\tilde{\Theta}\left(x+\delta; K, p_\varepsilon\right) + m}{1-t} - \bar{\Gamma}_t\right)^2}_{\text{Robustness}} + \underbrace{\ell\left(f\left(x+\delta\right), 1 - h(x)\right)}_{\text{Validity}} + \underbrace{\lambda \left\|\delta\right\|_1}_{\text{Proximity}} \tag{8}$$

Algorithm 1 CROCO optimization for counterfactual generation

Input: x s.t. $f(x) < t$, f, $\lambda > 0$, α, $\bar{\Gamma}_t > 0$, K,p_ε
Output: $x + \delta$
$\delta \leftarrow 0$;
Compute $\tilde{\Theta}\left(x + \delta; K, p_\varepsilon\right)$
while $f(x+\delta) < t$ **and** $\frac{m+\tilde{\Theta}(x+\delta;K,p_\varepsilon)}{1-t} > \bar{\Gamma}_t$ **do**
  $\delta \leftarrow \delta - \alpha \cdot \nabla_\delta \mathcal{L}_{\mathsf{CROCO}}(x+\delta; \Theta_t, p_\varepsilon, \lambda)$ ▷ From equation 8
  Update $\tilde{\Theta}\left(x + \delta; K, p_\varepsilon\right)$
end while
Return: $x + \delta$

The last two terms implement the classical trade-off for counterfactual generation. Indeed, the second term pushes the counterfactual class toward a class that differs from the example class (if $h(x) = 0$ then we want $h(\breve{x}) = 1$), while the last term minimizes the distance between the counterfactual and the example to explain.

The first term encourages our new estimator to be close to a target value $\bar{\Gamma}_t$, *i.e.* the target upper-bound of the recourse invalidation rate. This pushes to choose a counterfactual that has an upper bound close to the objective.

Algorithm 1 describes the optimization process for CROCO. Gradient steps are performed until the counterfactual predicted class is flipped ($f\left(x + \delta\right) \geq t$), and the value of the upper-bound $\frac{m+\tilde{\Theta}(x+\delta;K,p_\varepsilon)}{1-t}$ is below the target value $\bar{\Gamma}_t$.

CROCO has several benefits, it allows the user to generate counterfactuals with almost surely a minimal robustness, and this agnostically to the noise distribution. Moreover, our optimization problem relies on an almost-sure upper bound of the true recourse invalidation rate instead of relying on an approximation as Pawelcyk et al. did with PROBE [15]. Our intuition is that this will in practice improve the trade-off between proximity and robustness.

5 Experiments and results

We have divided our experiments into two sections. After experimentally confirming that our approach preserves the validity of the counterfactuals, the purpose of the first section is to demonstrate empirically that CROCO provides an effective management of the trade-off between proximity and robustness in comparison to PROBE. In the second section, we demonstrate experimentally that the counterfactuals returned by CROCO exhibits a lower degree of invalidation with respect to the user-defined target than PROBE do.

First of all, we describe the datasets that we used for evaluation, along with the metrics we employed as well as the predictive model details.

5.1 Experimental setting

For a fair comparison, we used the CARLA library [13], which was also used for evaluating PROBE. It contains three binary classification datasets: *Adult*, *Give*

Me Some Credit (GSC), and *COMPAS*. These datasets contain both numerical and categorical features. Both numerical and categorical variables are used to train the classifier, but the counterfactuals are generated by modifying only the numerical variables. The proportion of categorical variables for each dataset are respectively 3/7, 1/12 and 25/40. Additional details about these datasets are available in the section A.4 of the supplementary material. For every dataset, the classification model, f, is the fully connected neural network implemented in the CARLA library[5]. It is composed of 50 hidden layers and ReLU activation functions.

We used for evaluation the following metrics:

Validity A counterfactual $\breve{x}$ of an example x is valid if the classification model predicts different classes for x and $\breve{x}$ [11,12]. Formally:

$$\text{Validity} = \begin{cases} 0, \text{ if } f(\breve{x}) = f(x) \\ 1, \text{ if } f(\breve{x}) \neq f(x) \end{cases}$$

The validity measure lies in $[0, 1]$. The higher it is, the better.

Distance The distance is the L_1 distance between an example, x and its counterfactual, $\breve{x}$ [11,22].

$$\text{Distance} = \|\breve{x} - x\|_1 = \|\delta\|_1$$

A low value indicates fewer changes of features to apply to the original example to obtain the counterfactual. As the distance decreases, the proximity increases. In the context of counterfactual generation, we assume that the lower the distance, the more actionable the counterfactual, the better.

Recourse invalidation rate We used $\tilde{\Gamma}$ (see equation 3) to evaluate recourse invalidation rate, i.e. the robustness of the counterfactual. This value indicates the risk to have an invalid counterfactual in case the counterfactual is slightly changing wrt to the automatically recommended counterfactual. The lower, the better.

The recourse invalidation rate makes the assumption of a neighborhood represented by a distribution, p_ε. CROCO makes no hypothesis on this distribution but PROBE requires a Gaussian distribution. For the sake of fairness, we use a centered Gaussian distribution with a parameterized variance σ for the two methods.

For each dataset, we run PROBE with $\sigma^2 \in \{0.005, 0.01, 0.015, 0.02\}$ and $\Gamma_t \in \{0.05, 0.10, 0.15, 0.2, 0.25, 0.3, 0.35\}$. Regarding the setting of CROCO, we choose $K = 500$, $m = 0.1$, $t = 0.5$. λ is found through an iterative procedure that is described in section A.5.2 of supplementary material. For each dataset, we run CROCO with the same parameters as PROBE: $\sigma^2 \in \{0.005, 0.01, 0.015, 0.02\}$ and $\bar{\Gamma}_t \in \{0.05, 0.10, 0.15, 0.2, 0.25, 0.3, 0.35\}$.

We also include the approach of Wachter et *al.* [22] (referred to as *Wachter*) in our experiment. This counterfactual generation method establishes a baseline for recourse invalidation rate.

[5] Function carla.models.catalog.MLModelCatalog of the CARLA library.

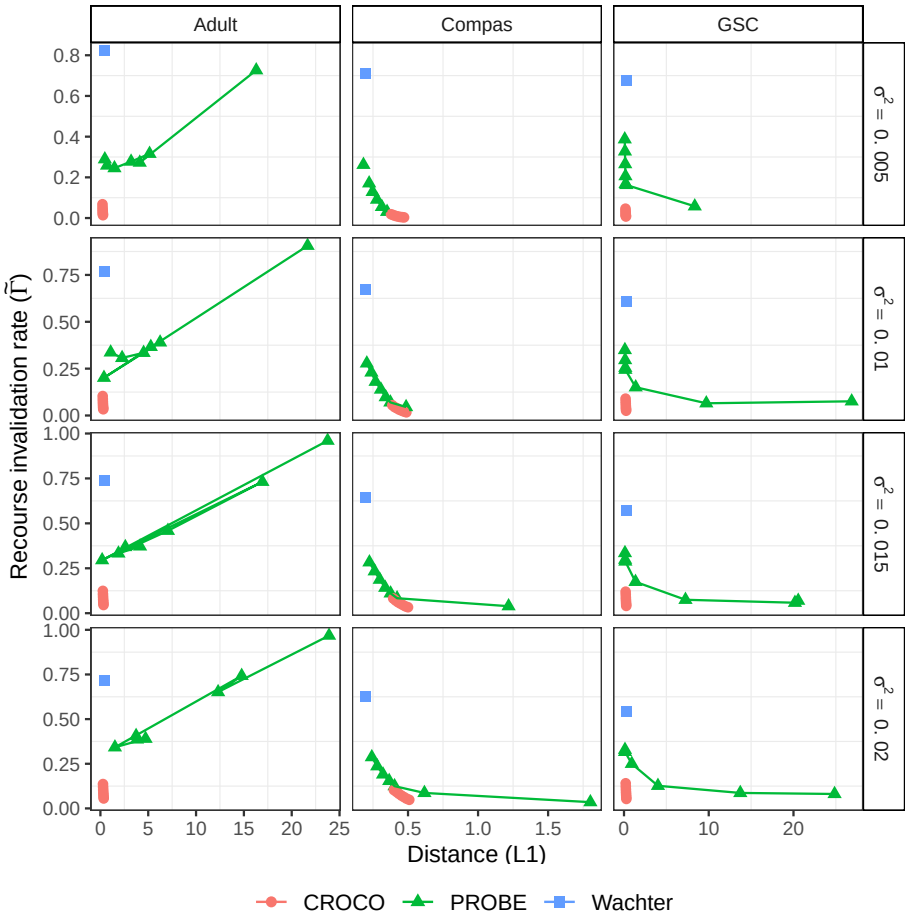

Fig. 3. Trade-off between recourse invalidation rate and distance with Gaussian distribution noises. Each column corresponds to a dataset and each line to a value of $\sigma^2 \in \{0.005, 0.01, 0.015, 0.02\}$. In each subplot the value of σ^2 is fixed. Each point of a curve corresponds to a mean recourse invalidation rate and a mean distance for a given target, we have $target \in \{0.05, 0.10, 0.15, 0.2, 0.25, 0.3, 0.35\}$. The points are connected by target order.

In our experiments, we generated 500 counterfactuals for each dataset and each parameterized method. We collected their recourse invalidation rate, distance and validity, that are discussed in the following.

5.2 Comparisons between PROBE and CROCO

In this section, the quality of the counterfactuals generated using CROCO, PROBE and *Watcher* is compared.

First of all, *Watcher* and CROCO achieves a perfect validity for all datasets. PROBE achieved a perfect validity on all datasets, except for two counterfactual

sets, that corresponds to the COMPAS dataset where $\sigma^2 = 0.005$ and $\Gamma_t = 0.3$ and also the GSC dataset where $\sigma^2 = 0.02$ and $\Gamma_t = 0.05$. As a consequence, in the following, we focus the analysis on the trade-off between the distance and the recourse invalidation rate. The section A.3.1 of the supplementary material contains details regarding the validity obtained for each dataset, and counterfactual sets that are generated.

Figure 3 compares *Watcher*, PROBE and CROCO regarding the distance and recourse invalidation rate on the three different datasets. Each point of a given curve corresponds to the mean recourse invalidation rate and the mean distance that is obtained from CROCO or PROBE by fixing a target value. Note that *Watcher* has only one point as it has no recourse invalidation rate target parameter. The standard-deviation values are provided in section A.3.2 of supplementary material. Note that for a given curve, the points are linked by order of increasing target value.

For the GSC dataset, CROCO achieves both smaller distances (higher proximities) and lower recourse invalidation rates compared to PROBE, regardless of the value of σ^2. The same conclusion can be drawn for the COMPAS dataset, except for $\sigma^2 = 0.005$ where CROCO achieves smaller recourse invalidation rates but at the cost of higher distances.

Regarding the Adult dataset, we observe that PROBE is unstable, as it can produce solutions with higher recourse invalidation rate than the target fixed by the user (where $\tilde{\Gamma} \geq \Gamma_t$). On the other hand, CROCO is stable and achieves both smaller distances (higher proximities) and lower recourse invalidation rates. We also noticed that on all the datasets, distance values increase when σ^2 increased, thus confirming the presence of a trade-off between the two quantities.

When solutions are closely clustered together in terms of mean distances, both PROBE and CROCO exhibit similar standard deviation values. However, when solutions are more widely dispersed, PROBE tends to have higher standard deviation values compared to CROCO (see section A.3.2 of supplementary material).

We observed that for all datasets and values of σ^2, PROBE and CROCO outperform *Wachter* in terms of recourse invalidation rates. The only exception is the Adult dataset when $\Gamma_t = 0.35$, where PROBE produces higher recourse invalidation rates due to instability issues.

5.3 Target invalidation study

For each counterfactual that is obtained from PROBE or CROCO, we computed the recourse invalidation rate and compared it with the targeted recourse invalidation rate.[6] The results are provided in Figure 4. The graphics figure out the diagonal representing the exact match between the targeted and the recourse invalidation rate. All points that are above this diagonal correspond to counterfactuals that do not achieve the robustness requested by the user. We notice that with PROBE, the recourse invalidation rates frequently exceed the target

[6] Watcher is not figured out as it does not set a target for recourse invalidation rate.

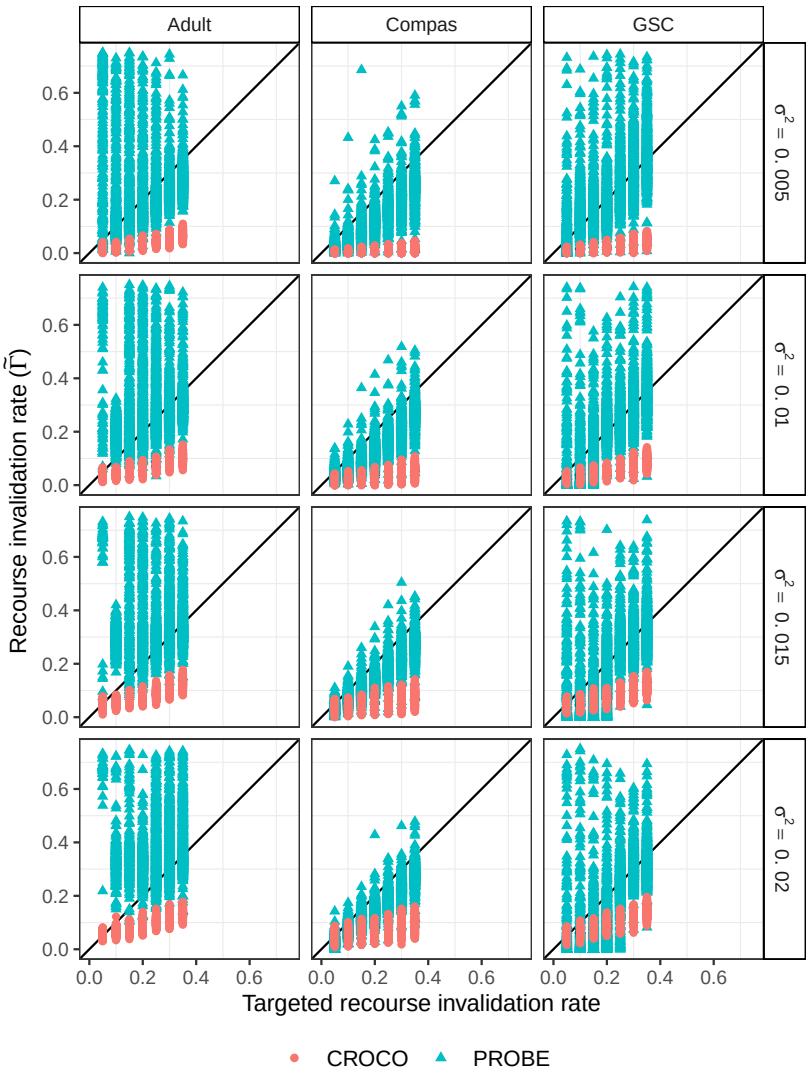

Fig. 4. Comparison between targeted recourse invalidation rate and recourse invalidation rate. Each column corresponds to a dataset and each line to a value of $\sigma^2 \in \{0.005, 0.01, 0.015, 0.02\}$. In each subplot, the value of σ^2 is fixed. Each point corresponds to a counterfactual, on the x-axis is presented the target recourse invalidation rate for the counterfactual, and on the y-axis the recourse invalidation rate that is computed.

fixed by the user. It illustrates that the approximation of Γ made by PROBE is too loose. In contrast, for CROCO, the recourse invalidation rates are typically lower, indicating that the user-specified target is less invalidated.

We computed the upper bound value derived in proposition 2 for each counterfactual obtained from CROCO.

Figure 5 of section A.3.3 of the supplementary material illustrates the evolution of the upper bound value ($\frac{m+\tilde{\Theta}}{1-t}$) with regard to the recourse invalidation rate for different values of σ^2. Our analysis show that the theoretical bound is not violated. This means that even in cases where CROCO failed to find a solution that matches the user target (i.e., where $\frac{m+\tilde{\Theta}}{1-t} > \bar{\Gamma}_t$), we can still provide the user a guarantee on the true recourse invalidation rate. This guarantee is based on the value of $\tilde{\Theta}$ that is obtained at the end of the optimization.

6 Conclusion

In this paper, we introduce CROCO, a novel framework for generating counterfactuals that are robust to input changes. A robust method guarantees that the slightly perturbed counterfactual is still valid. Our approach leverages a new estimator that provides a theoretical guarantee on the true recourse invalidation rate of the generated counterfactuals. Through experiments comparing CROCO to the state-of-the-art PROBE method, we demonstrate that our approach achieves a better trade-off between recourse invalidation rate and proximity, while also leading to less invalidation regarding the user-specified target. While these initial results are promising, it is necessary to evaluate CROCO on a larger number of datasets to confirm the robustness of the performance obtained. Moving forward, we plan to extend the capabilities of CROCO by adapting it to handle categorical variables. Since our approach is independent to the noise distribution, it seems reasonably possible to generate robust counterfactuals for data with both numerical and categorical variables. CROCO is implemented in the CARLA framework and will be soon available for practical usage.

References

1. Artelt, A., Vaquet, V., Velioglu, R., Hinder, F., Brinkrolf, J., Schilling, M., Hammer, B.: Evaluating robustness of counterfactual explanations. In: Proceedings of the Symposium Series on Computational Intelligence (SSCI). pp. 01–09. IEEE (2021)
2. Black, E., Wang, Z., Fredrikson, M.: Consistent counterfactuals for deep models. In: Proceedings of the International Conference on Learning Representations (ICLR). OpenReview.net (2022)
3. Brughmans, D., Leyman, P., Martens, D.: Nice: an algorithm for nearest instance counterfactual explanations. arXiv **v2** (2021), `https://arxiv.org/abs/2104.07411`
4. Dominguez-Olmedo, R., Karimi, A.H., Schölkopf, B.: On the adversarial robustness of causal algorithmic recourse. In: Proceedings of the 39th International Conference on Machine Learning (ICML). vol. 162, pp. 5324–5342 (2022)
5. Ferrario, A., Loi, M.: The robustness of counterfactual explanations over time. Access **10**, 82736–82750 (2022)
6. Guidotti, R.: Counterfactual explanations and how to find them: literature review and benchmarking. Data Mining and Knowledge Discovery pp. 1–55 (2022)
7. Guyomard, V., Fessant, F., Guyet, T.: VCNet: A self-explaining model for realistic counterfactual generation. In: Proceedings of the European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML/PKDD). pp. 437–453 (2022)
8. Laugel, T., Lesot, M.J., Marsala, C., Detyniecki, M.: Issues with post-hoc counterfactual explanations: a discussion. arXiv (2019), `https://arxiv.org/abs/1906.04774`
9. Maragno, D., Kurtz, J., Röber, T.E., Goedhart, R., Birbil, S.I., Hertog, D.d.: Finding regions of counterfactual explanations via robust optimization (2023), `https://arxiv.org/abs/2301.11113`
10. Mishra, S., Dutta, S., Long, J., Magazzeni, D.: A survey on the robustness of feature importance and counterfactual explanations. arXiv (v2) (2023), `https://arxiv.org/abs/2111.00358`
11. Mothilal, R.K., Sharma, A., Tan, C.: Explaining machine learning classifiers through diverse counterfactual explanations. In: Proceedings of the conference on Fairness, Accountability, and Transparency (FAccT). pp. 607–617 (2020)
12. de Oliveira, R.M.B., Martens, D.: A framework and benchmarking study for counterfactual generating methods on tabular data. Applied Sciences **11**(16), 7274 (2021)
13. Pawelczyk, M., Bielawski, S., van den Heuvel, J., Richter, T., Kasneci, G.: CARLA: A python library to benchmark algorithmic recourse and counterfactual explanation algorithms. In: Conference on Neural Information Processing Systems (NeurIPS) – Track on Datasets and Benchmarks. p. 17 (2021)
14. Pawelczyk, M., Broelemann, K., Kasneci, G.: Learning model-agnostic counterfactual explanations for tabular data. In: Proceedings of The Web Conference (WWW'20). pp. 3126–3132 (2020)
15. Pawelczyk, M., Datta, T., van-den Heuvel, J., Kasneci, G., Lakkaraju, H.: Probabilistically robust recourse: Navigating the trade-offs between costs and robustness in algorithmic recourse. In: Proceedings of the International Conference on Learning Representations (ICLR). OpenReview.net (2023)

16. Poyiadzi, R., Sokol, K., Santos-Rodriguez, R., De Bie, T., Flach, P.: Face: feasible and actionable counterfactual explanations. In: Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society. pp. 344–350 (2020)
17. Rawal, K., Kamar, E., Lakkaraju, H.: Algorithmic recourse in the wild: Understanding the impact of data and model shifts. arXiv **v3** (2020), `https://arxiv.org/abs/2012.11788`
18. Upadhyay, S., Joshi, S., Lakkaraju, H.: Towards robust and reliable algorithmic recourse. Advances in Neural Information Processing Systems **34**, 16926–16937 (2021)
19. Ustun, B., Spangher, A., Liu, Y.: Actionable recourse in linear classification. In: Proceedings of the conference on Fairness, Accountability, and Transparency (FAccT). pp. 10–19 (2019)
20. Van Looveren, A., Klaise, J.: Interpretable counterfactual explanations guided by prototypes. In: Proceedings of the European Conference on Machine Learning and Knowledge Discovery in Databases (ECML/PKDD). pp. 650–665 (2021)
21. Virgolin, M., Fracaros, S.: On the robustness of sparse counterfactual explanations to adverse perturbations. Artificial Intelligence **316**, 103840 (2023)
22. Wachter, S., Mittelstadt, B.D., Russell, C.: Counterfactual explanations without opening the black box: Automated decisions and the GDPR. Harvard Journal of Law and Technology **31**(2), 841–887 (2018)