

## RESEARCH ARTICLE

# The Robustness of Counterfactual Explanations Over Time

ANDREA FERRARIO<sup>1</sup> AND MICHELE LOI<sup>2</sup><sup>1</sup>Mobililar Lab for Analytics at ETH, 8092 Zurich, Switzerland<sup>2</sup>Department of Mathematics, Politecnico di Milano, 20133 Milano, Italy

Corresponding author: Andrea Ferrario (aferrario@ethz.ch)

**ABSTRACT** Counterfactual explanations are a prominent example of post-hoc interpretability methods in the explainable Artificial Intelligence (AI) research domain. Differently from other explanation methods, they offer the possibility to have recourse against unfavourable outcomes computed by machine learning models. However, in this paper we show that retraining machine learning models over time may invalidate the counterfactual explanations of their outcomes. We provide a formal definition of this phenomenon and we introduce a method, namely counterfactual data augmentation, to help improving the robustness of counterfactual explanations over time. We test our method in an empirical study where we simulate different model retraining scenarios. Our results show that counterfactual data augmentation improves the robustness of counterfactual explanations over time, therefore contributing to their use in real-world machine learning applications.


**INDEX TERMS** Machine learning, explainable artificial intelligence, counterfactual explanations, robustness, algorithmic recourse, counterfactual data augmentation.

## I. INTRODUCTION

The provision of explanations of machine learning model outcomes—also called post-hoc explanations—is key in the domain of explainable Artificial Intelligence (xAI) [1]–[3]. Post-hoc explanations are interfaces between humans and the machine learning model that are “both an accurate proxy of the decision maker [i.e., the model] and comprehensible to humans” [4]. They are invoked in relation to the need to 1) audit and improve machine learning models by supporting their interpretability, 2) enable learning from data by discovering previously unknown patterns, and 3) establish compliance with legislations and legal requirements [5]–[7]. Counterfactual explanations [8] are a class of post-hoc interpretability explanations that provide the person subjected to a machine learning model-generated decision with understandable information on the model outcome, and a strategy to achieve an alternative (future) one. They are an example of “contrastive explanations in xAI” [5], [9]: they explain a given model outcome by sharing a “what-if” alternative

scenario comprising of feature-perturbed versions of the same individual [10]–[12]. Recent literature from the xAI domain has discussed selected desiderata that may support the applicability of counterfactual explanations in real-world machine learning model pipelines [10], [12]–[17]. In particular, the desiderata of feasibility, actionability and sparsity would allow to generate and share cognitively accessible counterfactual explanations that respect causal models between features, and suggest actionable strategies whose alternative scenarios comprise the change of a limited number of features.

In this paper, we investigate an additional desideratum of counterfactual explanations: their **robustness over time**. Our discussion is motivated by the common assumption in the xAI literature that the machine learning model whose outcomes have to be explained remains “stable” or does not change, in a given time frame of interest [10], [18], [19]. This assumption is violated in most real-world applications, where machine learning models are retrained with frequencies that depend on the application under consideration. Although the literature on xAI has highlighted the possibility of this conflict [10], [18], [19], a systematic analysis of the consequences of the interactions between the change of machine

The associate editor coordinating the review of this manuscript and approving it for publication was Guangcun Shan .

learning models over time and the provision of counterfactual explanations is still missing. This analysis would need to formalize the emergence of undesirable interactions between the evolution over time of machine learning models and the counterfactual explanations of their outcomes, and propose quantitative methods to mitigate the risk stemming from these interactions. In particular, these methods should allow researchers to simulate different scenarios characterizing the machine learning model retraining routines in real-world applications.

In this paper, we promote such a two-step analysis. In the first step, we start by discussing the emergence of unfavorable cases—called “unfortunate counterfactual events” (UCEs). UCEs are the result of undesirable interactions between models changing over time and the counterfactual explanations of their outcomes. They happen when the retraining of machine learning models invalidates the investment of resources behind the successful implementation of a scenario originally recommended by a feasible, actionable and possibly sparse counterfactual explanation. For instance, UCEs may emerge when individuals follow the scenario of a counterfactual to try getting a loan, or getting accepted at a university that performs automated screening of student candidates with machine learning (e.g., GPA scoring [20]). From a normative perspective, UCEs are a violation of the principle of algorithmic recourse, i.e., “the systematic process of reversing unfavorable decisions by algorithms and bureaucracies across a range of counterfactual scenarios” [20]. Clearly, algorithmic recourse allows individuals managing their precarity in an algorithmically-assisted society [20]. Therefore, researchers need to investigate UCEs as they may hinder the efficacy of algorithmic recourse and develop strategies to mitigate their impact on the institutions promoting the use of the algorithms and those affected by their outcomes.

As a result, our goal is to improve the robustness of counterfactual explanations over time, that is to reduce the number of UCEs resulting from the retraining of machine learning models, quantitatively. Therefore, in the second step of our analysis, we propose to use counterfactual data augmentation (CDA) [21]–[23]. CDA promotes the use of data points and their counterfactuals while training machine learning models. Recently, researchers have successfully applied CDA to mitigate gendered language, reduce spurious correlations and improve machine learning model performance in natural language processing use cases [21], [22], [24], [25]. Others have used CDA in computer vision to improve performance of deep learning models [23] or proposed CDA as an alternative to over-sampling algorithms, such as SMOTE (Synthetic Minority Over-sampling Technique) [26], [27]. To the best of our knowledge, no study has used CDA in the context of xAI, yet alone to improve the robustness of counterfactual explanations over time. Specifically, we suggest to use CDA by adding to training data all those counterfactual explanations  $c(x)$  that have been shared with the affected individuals, i.e.,  $x$ , until the point of time of model retraining. The machine learning model would then be trained also on the pairs

$(x, c(x))$ . The idea is that, learning the pairs  $(x, c(x))$  at model retraining, the percentage of UCEs would be reduced compared to model retraining without CDA, therefore improving the robustness of the counterfactuals  $c(x)$ 's.

In an empirical study, we analyze the efficacy of CDA against the emergence of UCEs. To do so, we consider the logistic regression and random forest model classes, and we use Diverse Counterfactual Explanations (DiCE) algorithm [10] to generate their counterfactual explanations on two widely-used datasets in the xAI domain: the Adult–Income and German–Credit datasets [28]. We implement four different retraining scenarios to discuss design factors that may affect the efficacy of using CDA to mitigate the emergence of UCEs in real-world machine learning use cases. Our study results show that CDA allows reducing the number of UCEs that result from retraining, although the robustness against their emergence depends, in particular, on the model that is retrained, the number of added counterfactuals and whether they are originally computed by a model of the same class considered at retraining (or a different one), and the amount of “new” data used at retraining. As a result, our two-step analysis is a first attempt to ensure the consistent use of counterfactual explanations in real-world applications over time, fostering trust in institutions and their algorithm-supported decision-making procedures. The contributions of the paper are as follows:

- 1) we provide a formalization of the lack of robustness of counterfactual explanations over time discussing the emergence of the UCEs in machine learning applications;
- 2) we discuss different strategies to manage UCEs and we propose using CDA as a quantitative method to mitigate the emergence of UCEs in machine learning applications;
- 3) in an empirical study, we show that CDA allows reducing the number of UCEs across different scenarios of machine learning model retraining.

The paper is structured as follows. In Section II we present the state of the art in the field of counterfactual explanations, we discuss the role of time in machine learning modeling and the provision of model explanations and we introduce the relevant literature on CDA. In Section III, we introduce a formalization of UCEs and CDA. In Section IV we introduce an empirical study where we apply CDA to mitigate the emergence of UCEs. In Section V we discuss the results of the empirical study and future avenues of research. We then present our conclusions.

## II. RELATED WORK

### A. WHAT ARE COUNTERFACTUAL EXPLANATIONS?

Counterfactual explanations [8] are explanations of machine learning model outcomes that provide people with a scenario describing a state of the world—called “closest world” [8]—in which an individual would have received an alternative machine learning outcome. For example, they

explain to an individual why he or she did not receive a bank loan providing the “what-if” scenario: “you would have received the loan if your income was higher by 10,000\$” [10]. This “what-if” scenario shows that an alternative outcome can be reached by altering the values of a subset of the features describing the instance at hand (i.e., the data point of the individual asking for an explanation of the denied loan, in the above example) [5], [8]. For this reason, counterfactual explanations are an example of model-agnostic “feature-highlighting explanations” [18]. The counterfactual scenario provided by a counterfactual explanation is a “hypothetical point that is classified differently from the point currently in question” [18]. We call it “counterfactual” (data point) for simplicity. Not only counterfactuals provide a human-interpretable [8] explanation of a machine learning outcome, but in many applications, such as in financial services, public administration, the education system, or healthcare<sup>1</sup> they outline a strategy, or “recommendation”, to achieve an alternative, and possibly favorable, one, through the provision of a “what-if” scenario. This aspect differentiates counterfactual explanations from more descriptive machine learning model outcome explanation methods, such as Local Interpretable Model-agnostic Explanations [29] and Shapley values [30]. In this regard, counterfactual explanations are the most prominent example of xAI methods supporting algorithmic recourse, i.e., “the systematic process of reversing unfavorable decisions by algorithms and bureaucracies across a range of counterfactual scenarios” [20]. As Karimi *et al.* observe, underlying algorithmic recourse “is the desire to assist individuals that are negatively affected by automated decision-making systems to improve their circumstance” [31]. Then counterfactuals provide assistance to individuals by supporting their understanding of the reasons behind an algorithmic—typically, unfavourable—outcome and suggesting actions to perform to achieve an alternative—typically, favourable—one [31].<sup>2</sup>

Let us introduce some notation before discussing a formal definition of counterfactual explanations. A machine learning model<sup>3</sup> is a map  $\theta : \mathbb{R}^d \rightarrow \{0, 1\}$ , where  $x \mapsto \theta(x) = \hat{y}$ . The model is trained on samples  $(x, y)$  from a distribution on  $\mathcal{X} \times \mathcal{Y}$ , where  $\mathcal{X} = \mathbb{R}^d$  and  $\mathcal{Y} = \{0, 1\}$  without any loss of generality.

**Definition 1 (Counterfactual Explanation [8]):** Let  $\theta$  be a machine learning model and  $x \in \mathbb{R}^d$  with outcome

<sup>1</sup>Counterfactuals are similar to how physicians sometimes communicate with patients: “If you have had a total cholesterol level below 200 (mg/dl), then you would have not needed statins.”

<sup>2</sup>For these reasons, researchers in xAI are typically interested in the generation of counterfactual explanations for individuals (e.g., customers, convicts, students, and patients in digital health interventions) [10], [20], [32]–[34]. Therefore, we will typically refer to an “individual” as the data point whose machine learning outcome is explained with a counterfactual. In principle, counterfactuals may be generated for use cases where data points are not individuals, such as in the natural language processing (e.g., hate speech detection) or computer vision (e.g., classification of fundus images) domains. However, in these use cases, other types of explanations, such as Shapley values [30], are usually preferred [35], [36].

<sup>3</sup>In this paper, we consider only the case of binary classification problems.

$\theta(x) = \hat{y} \in \{0, 1\}$ . A counterfactual explanation  $c(x)$  of  $x$  (in short: “counterfactual”) is an element of the set

$$\arg \min_{x' \in \mathbb{R}^d} l(\theta(x'), f(\hat{y})) + d(x, x'), \quad (1)$$

where  $l$  is a loss function,  $d$  is a distance on  $\mathbb{R}^d$ , and  $f(\hat{y}) = 1 - \hat{y}$  denotes the outcome alternative to  $\hat{y}$  in  $\{0, 1\}$ .

The first term in (1) encodes the counterfactual condition, i.e., the search for the alternative outcome  $f(\hat{y})$  for a candidate  $x' \in \mathbb{R}^d$ , while the second term keeps the counterfactual “close” to the original instance  $x$ . Therefore, counterfactuals are algorithmically generated by “identifying the features that, if minimally changed, would alter the output [i.e., the current outcome] of the model” [18].

In the original formulation of (1), Wachter *et al.* choose  $l(\theta(x'), f(\hat{y})) := (\theta(x') - f(\hat{y}))^2$  and  $d(x, x')$  is a normalized  $l_1$ -distance [8]. However, since Wachter *et al.*’s work, many other algorithms to compute counterfactuals have been introduced in the xAI literature. We refer to [11], [37]–[39] for more details. By definition, counterfactual explanations are model agnostic. In fact, to compute a counterfactual explanation  $c(x)$  of an input  $x$ , the model  $\theta$  is treated as a “black-box”, or an input-output system. The outcomes  $\theta(x')$  of  $\theta$  are used in (1), instead. Moreover, as counterfactual explanations identify those features that, if changed, would alter the model output of  $x$ , they provide, at the same time, a degree of protection to companies’ intellectual property by the disclosure of these features to third parties [18]. Moreover, they comply with legal requirements on explanations in both Europe and the United States [18]. For these reasons, they have begun to attract the interest of different sectors of society, such as businesses, regulators, and legal scholars [18].

## B. SELECTED DESIDERATA OF COUNTERFACTUAL EXPLANATIONS

Recently, the literature on counterfactual explanations has focused on selected desiderata, i.e., properties that counterfactual explanations should satisfy to appropriately support individuals during algorithmic recourse [10], [13]–[16]. We discuss some prominent examples in what follows.

Counterfactual explanations are said to be **feasible** [8], [16], if they propose a scenario that respects the causal model [15], [40] of the variables of the dataset at hand. For example, a feasible counterfactual explanation may suggest that a loan would have been granted to an individual, if his annual income had been +10,000\$, other things equal (i.e., all other variables unchanged). On the other hand, a counterfactual scenario suggesting to decrease age, or increase the educational level from high school diploma to a master’s degree without increasing age, violates causal constraints among variables. In the first case, it simply suggests an impossible recommendation. In the second case, the recommendation is not compatible with the need to spend years to achieve a master’s degree, starting from a high school diploma. We also note that some features may simply be immutable, such as an individual’s birthplace or the date of marriage. Therefore,

they cannot be part of feasible counterfactual scenarios. We note that the Wachter *et al.*'s original algorithm in (1) to compute counterfactual explanations does not implement feasibility constraints. Recent approaches aim at ensuring the feasibility of counterfactual explanations by implementing post-hoc constraints on a set of generated counterfactuals. These constraints are originally introduced by domain experts to encode known causal relations between features [10], [16].

Let us consider a feasible counterfactual explanation. We say that it is **actionable** [10], [16], if the corresponding scenario can be reasonably implemented by the individual whose outcome is explained by the provision of the counterfactual explanation. Clearly, actionability is context-dependent: in particular, it depends on the capabilities of the individual implementing the counterfactual scenario. Considering Mothilal *et al.*'s example again [10], to increase the annual income of 10,000\$ may be a relatively easy task for affluent individuals. However, it may represent a daunting challenge for low-income ones. Considering actionable counterfactual explanations allows excluding explanations that, although feasible, propose scenarios whose implementation is practically not realizable. This, in turn, represents an impediment to the pursuit of algorithmic recourse [20].

Lastly, **sparsity** is the property of those counterfactual explanations whose scenarios suggest to alter only the values of a few variables [10], [13]. Mothilal *et al.* argue that "intuitively, a counterfactual example will be more feasible if it makes changes to fewer number of features" [10]. In other words, sparse counterfactuals "differ from the original datapoint in a small number of factors, making the change easier to comprehend" [41]. Therefore, they are deemed to be cognitively accessible. Sparsity becomes an important desideratum of counterfactual explanations, especially in big data contexts. Wachter *et al.* aim at ensuring sparsity of counterfactual explanations by using an  $l_1$ -distance measure  $d$  in (1) [8]. More recent studies have discussed sparsity by means of a two steps approach making use of a "growing spheres" algorithm [13] and the use of a "post-hoc operation to restore the value of continuous features back to their values in  $x$  [the input data point] greedily until the predicted class [...] changes" [10].

In summary, feasible, actionable, and sparse counterfactual explanations recommend causality-consistent scenarios that can be reasonably implemented by the individuals impacted by algorithmically-generated outcomes, once they act on the values of a limited number of features. Finally, we note that authors have recently suggested additional desiderata of counterfactual explanations, such as **diversity** and **robustness to local perturbations** [10], [12], [17]. The former refers to the possibility of generating diverse counterfactuals for a given outcome to explain [10]. In fact, the goal of diversity is to provide individuals with different counterfactual scenarios to perform algorithmic recourse [10]. The latter refers to the degree to which counterfactuals are sensitive to (possibly adverse) perturbations of the data point whose

machine learning outcome has to be explained, instead [12], [17]. We refer to [10], [12], [17] for all details.

### C. TIME, MACHINE LEARNING MODELS, AND COUNTERFACTUALS

Machine learning models [42] are inherently dynamic objects. They are designed to perform a task, such as the binary classification of a bank's customer in "creditworthy" or "not creditworthy", by learning on data. This process is referred to as "training", or "learning" [42]. After training, and depending on the application, (trained) machine learning models are deployed in IT architectures where they are fed upon batches of new data to generate predictions (also referred to as outcomes) and support human decision-making. Typically, the training of machine learning models does not occur only once, i.e., just before their deployment. In fact, the process can be periodically repeated, whenever new batches of data are made available, and the performance of the machine learning model degrades. This happens as the model often generates predictions in changing environments, whose evolution is not encoded in the dataset originally used for its training. For example, in e-commerce new products become available and can be recommended on an online marketplace platform. As a result, time affects machine learning models, their predictions, and the explanations promoted by research in xAI. In particular, as noted by Kroll *et al.*, "there is the added risk that the rule disclosed [by an explanation] is obsolete by the time it can be analyzed" [19].

The effects of time dependency of counterfactual explanations on their generation and provision has not yet been structurally investigated. In fact, Barocas *et al.* [18] discuss four key assumptions of feature highlighting explanations, such as counterfactual explanations [18]. One assumption is that "the model is stable over time, monotonic, and limited to binary outcomes" [18]. Stability over time is not further specified, and may be interpreted as the absence of retraining or change of selected model properties. Similarly, Mothilal *et al.* [10] argue that counterfactual explanations provide the information on "what to do to obtain a better outcome in the future", [10], but only "assuming that the algorithm remains relatively static" [10]. However, it is not clear how the stability of a model over time relates to the counterfactual explanations of model outcomes. Similarly, Verma *et al.* mention the dynamics of machine learning systems as a challenge to be tackled by future research on counterfactuals [43]. At the same time, Venkatasubramanian and Alfano warn against the possibility that certain attributes of a deployment model, such as a classifier that accepts students based on a GPA score cutoff value, may change over time and invalidate algorithmic recourse efforts [20]. In addition, Rawal *et al.* investigate bounds on the costs of counterfactuals when a model is retrained [44]. However, they consider only the case where the same model and hyperparameter settings are considered at retraining, focusing only on data distribution shifts over time [44].



Finally, we note that Pawelczyk *et al.* [45] approached the problem of generating counterfactuals under predictive multiplicity [46], or the phenomenon of having multiple machine learning models with similar performance. They computed the expected cost of counterfactuals under predictive multiplicity (i.e., the minimal perturbation that would alter the label of a given data point), generalizing previous results by Ustun *et al.* [47]. The case of predictive multiplicity can be applied in the scenario where an existing model can be replaced by a competing one [45]. However, no elaboration on the possible problems stemming from the interaction between time, machine learning models, and counterfactuals and their solutions is provided.

In summary, although xAI scholars have recently started promoting a discussion on time and explanations of machine learning models, a structured approach to the desideratum of **time robustness** of counterfactual explanations that addresses the specificities of real-world model update scenarios is still missing.

#### D. COUNTERFACTUAL DATA AUGMENTATION

Recently, different authors proposed to use counterfactual explanations to augment datasets used in machine learning problems. In the context of natural language processing, counterfactual data augmentation (CDA) has been applied to mitigate gender bias and for machine translation. In the first case, counterfactual instances are generated by swapping gendered words [48] or applying more general interventions on text instances [24]. Both methods generate matched pairs of textual data that are used in the learning of algorithms and allow retaining accuracy or improve debiasing benchmarks [24], [48]. In the second case, the use of counterfactuals leads to an improvement in performance with respect to traditional methods, such as translation, backtranslation and translation robustness [25]. Moreover, Kaushik *et al.* proposed to tackle the problem of machine learning model reliance on spurious correlations by letting human editors edit sampled documents (e.g., movie reviews) “to render (designated) counterfactual labels applicable” [22]. For example, in a sentiment analysis exercise, human editors were directed to revise negative movie review to make them positive, following editing guidelines [21]. Their results showed that the models trained on the augmented datasets that included counterfactuals are less sensitive to spurious correlations and with a high out-of-sample performance on different datasets [21], [22]. In computer vision, Teney *et al.* tackled the problem of spurious correlations in machine learning modeling using counterfactual instances to improve the training objective of deep models [23]. In fact, the authors noted that deep learning models for image recognition may rely on examples of spurious correlation, such as object co-occurrences, that may not hold on test data [23]. Therefore, they generated counterfactual images from existing annotated images by masking relevant regions and added them to training data [23]. This procedure resulted in

improved model performance on out-of-sample data [23]. Finally, the use of CDA has emerged also in the reinforcement learning domain to improve the performance of reinforcement learning models [49] or as an alternative to more traditional data augmentation algorithms, such as SMOTE [26], [27]. To the best of our knowledge, however, no study has yet investigated the use of CDA in the xAI research domain, yet alone to improve the robustness of machine learning model explanations over time. We explore this approach for the case of counterfactual explanations in the forthcoming sections.

### III. TIME ROBUSTNESS OF COUNTERFACTUAL EXPLANATIONS

#### A. “UNFORTUNATE COUNTERFACTUAL EVENTS”

As previously commented, feasible and actionable counterfactual explanations not only describe a scenario in which an individual could achieve an alternative—typically, favourable—outcome in understandable terms, but they highlight an actionable strategy to achieve it. In the case of sparse counterfactual explanations, in fact, this strategy focuses on altering the values of a limited number of variables. Such a counterfactual scenario may give the possibility to have recourse against an unfavourable outcome, in a given time window. However, the points of time at which 1) the explanation is generated by a model in use at a given institution and it shared with the impacted individual, and 2) its recommended scenario is “successfully implemented”,<sup>4</sup> may differ. This simple observation is at the basis of what we call an “unfortunate counterfactual event” (UCE). To discuss the emergence of UCEs we consider a setting with two agents, namely  $I$  (“institution”) and  $A$  (“agent”).  $I$  may be any institution that is using a machine learning model and its predictions to provide a service, such as a bank. The model is deployed in an IT infrastructure and counterfactual explanations of its outcomes are computed by a fixed algorithm  $C$ .<sup>5</sup>  $A$  is an individual accessing the service provided by  $I$  and that is affected by the outcomes generated by the model deployed by  $I$ . For example,  $A$  may be a customer of a bank or a candidate for an open job position. We arrive at the definition of an UCE:

*Definition 2 (Unfortunate Counterfactual Event):* Let  $I$  and  $A$  be as above. Let us consider the following scenario:

- 1) At time  $t_0$ ,  $I$  deploys a machine learning model  $\theta_{t_0}$  and an algorithm  $C$  to compute counterfactual explanations of  $\theta_{t_0}$ 's outcomes;
- 2) At time  $t_1 \geq t_0$ ,  $A$  receives a counterfactual explanation  $c(x_{t_1})$  computed by  $(\theta_{t_0}, C)$ , where  $\theta_{t_0}(x_{t_1}) = \hat{y}$ ;

<sup>4</sup>That is, the individual follows the counterfactual scenario and this result is reported to the institution.

<sup>5</sup>The algorithm  $C$  may implement the optimization problem in (1), or use DiCE [10], for example. In what follows, we focus on the case where the institution  $I$  deploys an algorithm  $C$  that is not updated over time.

3) At time  $t_2 > t_1$ ,  $I$  trains and deploys the machine learning model  $\theta_{t_2}$  such that  $\theta_{t_2} \neq \theta_{t_0}$ .<sup>6</sup>

Then, if there exists a time  $t^* \geq t_2$  such that

$$x_{t^*} = c(x_{t_1}), \quad (2)$$

$$\theta_{t_2}(x_{t^*}) = \theta_{t_0}(x_{t_1}) = \hat{y}, \quad (3)$$

we say that an “unfortunate counterfactual event” relative to agent  $A$  and the explanation  $c(x_{t_1})$  has occurred.

Let us discuss the definition of an UCE in some detail. Condition (2) states that at time  $t^*$  the counterfactual scenario encoded by  $c(x_{t_1})$  is the data point representing the agent  $A$ . This means that the counterfactual scenario  $c(x_{t_1})$  has been “successfully implemented” by  $A$ , it overrides the original data point  $x_{t_1}$  in the database of the institution  $I$  and it is accessed by  $\theta_{t_2}$  to compute its outcome, that is  $\hat{y}$ . For a “successful implementation” of the counterfactual scenario  $c(x_{t_1})$  to happen, the features in the scenario  $c(x_{t_1})$  have to be communicated to  $I$  or being updated in some way. For example, in the bank example by Mothilal *et al.* [10], if the counterfactual scenario suggests that an increase of liquidity per annum equal to +10,000\$ would have granted the agent  $A$  a loan, then  $A$  has to put the suggested amount of money in an account at the bank  $I$  and this action has to be recorded in the bank databases before  $\theta_{t_2}$  computes the creditworthiness of  $A$ . Moreover, condition (2) implies that all features that are not changed in the scenario are constant in the time interval  $[t_1, t^*]$ . For instance, in the banking example by Mothilal *et al.* [10], this means that  $[t_1, t^*]$  cannot be longer than one year, if the variable **Age** is used by the bank model to score creditworthiness of its customers, but it is not part of the counterfactual scenario  $c(x_{t_1})$  suggested to  $A$ .

Condition (3) states that the retrained model  $\theta_{t_2}$  does not properly encode the counterfactual scenario  $c(x_{t_1})$  originally computed by the model  $\theta_{t_0}$  (and the algorithm  $C$ ), although the scenario has been successfully implemented by  $A$ , by condition (2). In fact, agent  $A$  would have expected the outcome  $\theta_{t_2}(c(x_{t_1})) = f(\hat{y})$  as the result of the implementation of the counterfactual scenario. In summary, if an UCE occurs, all  $A$ 's efforts spent to implement the recommendations of the counterfactual explanation  $c(x_{t_1})$  are frustrated by the implementation of a retrained machine learning model  $\theta_{t_2}$  that did not properly learn the counterfactual  $c(x_{t_1})$  and its alternative (and expected) outcome  $f(\hat{y})$ .

### B. ON THE DIFFERENT STRATEGIES TO ADDRESS UCES

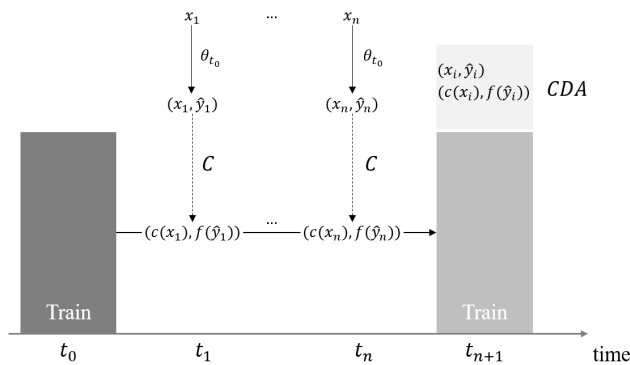
At the time of writing, the literature discusses no actionable solution to the emergence of UCES in machine learning applications. However, institutions promoting the use of counterfactual explanations need to design and implement procedures to address the emergence of UCES in their machine learning-assisted services and products. A key observation is that ethically and legally-informed strategies

<sup>6</sup> $\theta_{t_0}$  and  $\theta_{t_2}$  may belong to the same class of machine learning models (although with different sets of hyperparameters), or not.

that an institution may consider to address UCES—and minimize the risks associated to their emergence<sup>7</sup>—inform the procedures implemented for their resolution. In general, different strategies lead to different data and model-affecting procedures against UCES. One strategy may simply suggest to refrain from using counterfactual explanations in real-world applications due to the possibility of UCES. However, in this case, individuals would be deprived of the main tool that xAI offers to act systematically against unfavourable outcomes and exercise algorithmic recourse [20]. Another strategy may suggest to avoid any mitigation of UCES, promoting the idea that individuals should accept that “successfully implemented” counterfactual scenarios do not guarantee their sought-after outcome due to machine learning retraining routines. Therefore, this strategy states that counterfactuals do not entail any form of commitment between the institution generating them and the individuals affected by the outcomes addressed by the explanations. However, this point of view on counterfactuals is in contrast with the one that is currently promoted in the xAI literature. In fact, Barocas *et al.* [18], when discussing counterfactual explanations of credit loans, highlight that Wachter *et al.* have argued that the law should treat a counterfactual explanations as a promise rather than just an explanation [8]. More precisely, they argue that if a rejected applicant makes the recommended changes, i.e., successfully implements the counterfactual scenario, the promise should be honored and the credit granted, irrespective of the changes to the model that have occurred in the meantime [8].

Finally, an institution may try avoiding UCES by implementing the counterfactuals  $c(x)$ 's as constraints during the retraining of its machine learning models. This procedure stems from a strategy that aims to fulfill *all* counterfactual scenarios generated for individuals, e.g., customers and patients, until model retraining. However, this choice suffers from some limitations. First, the strategy is committed to preserve all counterfactuals, independently on additional constraints, such as economic necessity, that may emerge as a result of events such as a financial crisis or a global pandemic. This approach may represent a risk for the survivability of the institution. In fact, in case of an economic downturn scenario, a credit lending institution fulfilling all extant counterfactuals would likely award loans that are too risky, implying a reduction in overall profit. This, due to the economic downturn, may well cause the financial collapse of the institution. Second, the implementation of counterfactual constraints at model retraining is laborious. In fact, the use of constraints would require the update of all the optimization routines used for learning the models (e.g., a logistic regression, a random forest or a neural network). In addition, to implement counterfactual constraints as regularization terms in the

<sup>7</sup>For example, UCES are a source of reputational risk for the institution. They may lead to customer churn, with an effect on the profitability of the portfolio that depends, among others, on the number and the reserves of the customers impacted by the violation of their implemented counterfactual scenarios.



**FIGURE 1.** CDA at the retraining of machine learning models. At  $t_0$  a model  $\theta_{t_0}$  is trained and then deployed. A sequence  $(c(x_1), \dots, c(x_n))$  of counterfactuals is computed using the model  $\theta_{t_0}$  and the algorithm  $C$  on  $n$  data points  $(x_1, \dots, x_n)$  at times  $t_1 < \dots < t_n$ , with  $t_1 > t_0$ . As in Section II-A, we denote by  $\hat{y}_i = \theta_{t_0}(x_i)$  the prediction computed by  $\theta_{t_0}$  on  $x_i$ , for  $i = 1, \dots, n$ . At time  $t_{n+1}$ , where  $t_{n+1} > t_n$  the elements of the set  $\{(x_i, \hat{y}_i), (c(x_i), f(\hat{y}_i))\}_{i=1, \dots, n}$  are added to the data used to retrain the model  $\theta_{t_0}$ .

optimization algorithm of any given machine learning model one may introduce hyperparameters, similarly Levanon and Rosenfeld's proposal for strategic classification examples [50]. However, in that case, the strength of the regularization may affect both the percentage of UCEs at retraining and the performance of the model and would require tuning [50].

### C. COUNTERFACTUAL DATA AUGMENTATION AND UCEs

We introduce a strategy to manage the emergence of UCEs that is alternative to 1) avoiding the generation of counterfactuals in real-world applications, 2) avoiding any mitigation of UCEs, and 3) implementing counterfactual constraints. In fact, our strategy aims to reduce the percentage of UCEs at model retraining using CDA. This means to perform retraining of the machine learning model on data that include, in particular, all the counterfactual scenarios that have been generated and shared with third parties until that moment, together with their alternative outcomes. We show this proposal in Figure 1: at time  $t_{n+1}$ , the set of data and their counterfactuals

$$CDA := \{(x_i, \hat{y}_i), (c(x_i), f(\hat{y}_i))\}_{i=1, \dots, n} \quad (4)$$

is added to the data set used to retrain  $\theta_{t_0}$ . The prediction  $\hat{y}_i$  is computed using  $\theta_{t_0}$ , i.e.,  $\theta_{t_0}(x_i) = \hat{y}_i$  for all  $i = 1, \dots, n$ . A fixed algorithm  $C$  computes all the counterfactuals  $c(x_i)$ , instead.

As a result, the efficacy of CDA in improving the robustness of counterfactuals over time is measured by comparing the percentage of UCEs emerging after retraining with CDA with the percentage of UCEs in the case of model retraining without CDA.<sup>8</sup> Therefore, if counterfactual explanations of model outcomes are provided, then a “good” retrained model

<sup>8</sup>The focus here is on the retraining methodology (with or without CDA) as a mean to reduce the percentage of UCEs. In fact, the two retraining procedures may result in different models.

would be characterized not only by a satisfactory level of performance, but also by a low percentage of UCEs, considering all counterfactuals generated until the time of retraining. Moreover, the percentage of UCEs can be used also as a measure of robustness of the counterfactual explanations generated by the model over time. For example, if CDA is used  $m$  consecutive times, denoting by  $p_i$  the percentage of UCEs emerging at the  $i$ -th retraining, then the average  $\frac{1}{m} \sum_{i=1}^m p_i$  may be used as a metric to measure the average efficacy of using CDA to mitigate the emergence of UCEs for a given machine learning model application over time.

In summary, the use of CDA at model retraining to mitigate the emergence of UCEs stems from a strategy that provides an alternative to the options of refraining from the use of counterfactuals, violating past counterfactuals by ignoring the history of interactions with individuals asking them, and being trustworthy but risking institutional collapse (e.g., bank bankruptcy). Using CDA, institutions can try limiting the damage to their trustworthiness and deontology by reducing UCEs. Practically, the main benefit using CDA to manage the emergence of UCEs lies in its applicability in real-world machine learning use cases. In fact, as opposed to the use of counterfactual constraints, CDA can be easily implemented as part of a data pipeline: it simply requires the generation and storage of counterfactual explanations without modifying the definitions or implementations of machine learning models. Finally, we note that the use of CDA contributes to alter the distribution of data used by the machine learning models. In fact, depending on the application, the class distribution of training data, e.g., the number of creditworthy vs. not creditworthy customers in the credit lending case, may be imbalanced. Usually, the creditworthy customers are the vast majority in the portfolio. As an effect, considering counterfactual scenarios and their outcomes in the training data at time  $t_1$  may contribute to increase class imbalance as, arguably, counterfactuals may be requested mostly by those individuals who received an unfavourable outcome. In that case, and depending on the degree of class imbalance, machine learning techniques, such as class-weighted learning, sub-sampling or over-sampling [51], [52] may be taken into account.

## IV. EXPERIMENTS

In this section, we discuss an empirical study that aims to evaluate the use of CDA at machine learning model retraining to mitigate the emergence of UCEs.

### A. DATA

In our study, we considered the 1) Adult-Income, and 2) German-Credit datasets. They are among the most used datasets in the xAI research domain, in particular for the study of counterfactual explanations [37]. The Adult-Income dataset is used to classify whether an individual's income is over 50,000\$. We preprocessed the dataset as in [10]. This means that we use the same set of samples and features that Mothilal *et al.* consider by following the

original preprocessing by Zhu in [53]. The resulting dataset comprises 26,048 samples and nine features, that is hours per week, education level, occupation, work class, race, age, marital status, sex, and income. The dataset shows class imbalance, as 24% of all samples shows income “>50,000\$”. The German-Credit dataset [54] contains information about customers of a bank accessing loan opportunities, instead. It comprises 1000 samples and contains 20 features (i.e., sociodemographic and credit-related information) and it is used to classify creditworthiness of customers of the bank. Following Mothilal *et al.*, we consider the whole set of 20 features without further preprocessing [10]. The dataset shows class imbalance, as 70% of all samples are classified as creditworthy.<sup>9</sup>

## B. MACHINE LEARNING MODELING AND COUNTERFACTUALS

### 1) PARTITIONING DATA FOR CDA

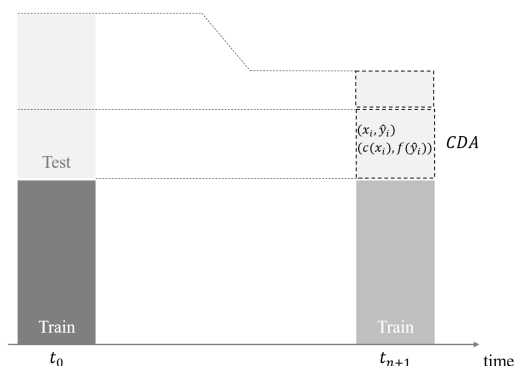
In Figure 2, we show the strategy for the retraining of machine learning models using CDA. The idea behind the proposed strategy is to simulate the CDA setting shown in Figure 1 by 1) performing a stratified split of data into training and test datasets, 2) training the “original” model  $\theta_{t_0}$  on training data, 3) computing counterfactuals from the subset of data points in the test dataset with unfavourable outcomes predicted by the model  $\theta_{t_0}$ , 4) adding different percentages of counterfactuals at retraining, that is at  $t_{n+1}$ , and 5) considering different percentages of the remaining test data to simulate “new” data to be used to train model  $\theta_{t_{n+1}}$ . As a result, the elements of the CDA set defined in (4) are sampled and computed from test data, only. This allows simulating the generation of counterfactuals after the model training at  $t_0$ , as in Figure 1.

We chose a 50:50 stratified split for both the German-Credit and Adult-Income datasets. The split is chosen to collect enough test data to compute counterfactual explanations and implement different retraining scenarios while retaining model performance.

### 2) MACHINE LEARNING MODELS

In the study, we considered two classes of machine learning models: 1) logistic regressions, and 2) random forests. We used the logistic regression and random forest implementations in the Python library `scikit-learn`. We started by training a model  $\theta_{t_0}$  (see Figure 2) using both classes. To do so, we performed a 5-fold cross-validation on training data (see Section IV-B1) to select the combination of hyperparameters leading to best performance. In the case of logistic regression models, we tuned the inverse of regularization strength, while in the case of random forests we tuned the number of trees in the ensemble and their depth. We denote the best models from the logistic regression and random forest class by **LOG** and **RF**. We report their hyperparameters in Appendix VII-A and we report their performance

<sup>9</sup>We used the version of the German Credit dataset available at <https://online.stat.psu.edu/stat857/node/215/>.



**FIGURE 2.** Strategy to implement CDA using the German-Credit and Adult-Income datasets. Counterfactuals are computed using test data that result in an unfavourable outcome as computed by  $\theta_{t_0}$ . Test data are added at retraining to simulate the use of “new” data at retraining.

on test data in Table 1, for both datasets. As the goal of the study is not to optimize for model performance, we did not implement any algorithm to address class imbalance in the German-Credit and Adult-Income datasets.

### 3) GENERATING COUNTERFACTUALS

We computed different percentages of counterfactuals from the test data with unfavourable outcome (cfr. Section IV-B1 and Figure 2) predicted by **LOG** and **RF**, for both datasets. Namely, we generated a number of counterfactuals equal to 5%, 10%, 15%, and 20% of all training data at  $t_0$ . Considering the Adult-Income dataset, the unfavourable outcome is “ $\leq 50,000\$$ ”. For the German-Credit dataset, it corresponds to “not creditworthy”. Therefore, by construction, a counterfactual suggests a strategy to achieve an annual income above the 50,000\$ threshold for the Adult-Income data, and the favourable outcome “creditworthy” for the German-Credit data.

To compute the counterfactuals we used the DiCE algorithm [10]. DiCE improves the original Wachter *et al.*’s algorithm to generate counterfactuals [8] by generating diverse counterfactuals, i.e., explanations showing diverse types of feature changes [10]. Diversity is deemed to be beneficial, as different individuals may want (or need) to take on different counterfactual scenarios [10]. It is implemented as a constraint in the optimization problem to generate counterfactuals, similarly to the sparsity desideratum [10]. In this study, we used the default hyperparameters of the DiCE algorithm and its Python implementation in the library `dice`.<sup>10</sup>

In Table 2, we show some examples of counterfactuals for the models **LOG** and **RF** on German-Credit data. All counterfactuals are feasible and sparse. In fact, they suggest to act upon pairs of features respecting the causality model between features. The **LOG** counterfactual suggests to change the value of **Account balance** from “no balance or debit” to “ $\geq 200$  DM or checking account for at least

<sup>10</sup><https://github.com/interpretml/DiCE>



1 year”<sup>11</sup> and **Purpose** of the requested loan from “new car” to “repair”. The **RF** counterfactual suggests to lower the duration of the requested loan, instead. Finally, in Table 3, we show some examples of counterfactuals for the models **LOG** and **RF** on `Adult-Income` data. Also in this case, all counterfactuals are feasible and sparse. The counterfactuals of both models suggests to change the value of **Age** and **Education**.

### C. PROCEDURE

To study the efficacy of CDA to mitigate the emergence of UCEs, we considered four model retraining scenarios. These scenarios are defined in terms of the following design factors:

- 1) the choice of class of machine learning models at  $t_0$  and  $t_{n+1}$ , i.e., at retraining (see Figure 1);
- 2) whether the counterfactuals  $c(x_i)$  used at retraining are generated by a model  $\theta_{t_0}$  in the same class of the one at retraining, i.e.,  $\theta_{t_{n+1}}$  (or not);
- 3) the percentage of counterfactuals  $c(x_i)$  added to the data used at retraining;
- 4) the percentage of “new” data points added to the data used at retraining.

We above design factors allowed us to introduce retraining scenarios that simulate “real-world” applications. In fact, depending on the machine learning application at hand, different choices of model classes at retraining, types of counterfactuals, their percentages as well as those of “new data” may occur. For each retraining scenario, to train the model  $\theta_{t_{n+1}}$  we performed a 5-fold cross-validation on the logistic regression and random forest classes selecting the combination of hyper-parameters that lead to the best model performance, similarly to the procedure performed at  $t_0$  and described in Section IV-B2. We then analyzed the UCEs considering the best model  $\theta_{t_{n+1}}$ , for both the logistic regression and random forest classes. The proposed scenarios are:

- **Scenario 1** (*same-model class robustness without CDA*): we do not use CDA but we consider different percentages of test data at retraining. We report the mean number of UCEs (using same-model class counterfactuals) as a function of the percentage of test data.
- **Scenario 2** (*same-model class robustness with CDA*): we retrain a logistic regression, respectively a random forest, with CDA considering different percentages of **LOG**, respectively **RF**, counterfactuals and test data. We report the mean number of UCEs (using same-model class counterfactuals) as a function of the percentage of counterfactuals added at retraining.
- **Scenario 3** (*different-model class robustness without CDA*): we do not use CDA but we consider different percentages of test data at retraining. We report the mean number of UCEs (using **RF** counterfactuals for the logistic regression and the **LOG** counterfactuals for

the random forest) as a function of the percentage of test data.

- **Scenario 4** (*different-model class robustness with CDA*): we retrain a logistic regression, respectively a random forest, with CDA considering different percentages of **RF**, respectively **LOG**, counterfactuals and test data. We report the mean number of UCEs (using **RF** counterfactuals for the logistic regression and the **LOG** counterfactuals for the random forest) as a function of the percentage of counterfactuals added at retraining.

We summarize the four scenarios in Table 4 for the sake of readability. In all scenarios, we retrained models adding percentages of test data equal to 0%, 10%, 20%, and 30% of all training data (including counterfactuals for scenario 2 and 4). In scenario 2 and 4, we retrained the machine learning models adding a percentage of counterfactuals equal to 5%, 10%, and 20% of all training data. As a result, in scenario 1 and 3, the mean percentage of UCEs is computed over the distribution of the different percentages of counterfactuals that are computed by the corresponding model at  $t_0$ , for each percentage of test data added at retraining. In scenario 2 and 4, the mean of UCEs is computed over the distribution of the different percentages of test data added at retraining, for each percentage of counterfactuals, instead. We collect the results of different retraining scenarios for the `German-Credit` data in Figure 3 and in Figure 4 for the `Adult-Income` data.

### D. RESULTS

#### 1) SCENARIO 1

For both the `German-Credit` and `Adult-Income` datasets, the retrained models show no UCE if no test data is added at retraining. In fact, by definition of our retraining strategy in scenario 1 (see Figure 2), these models are **LOG** and **RF**. Clearly, the models trained at  $t_0$  correctly classify all their counterfactuals as expected, if no additional data is used at retraining. If test data are added, however, the retrained models show different mean percentages of UCEs, with the logistic regression models being more robust than the random forests. For example, considering the `German-Credit` dataset, if a percentage of test data equal to 10% of training data is used at retraining, the random forest model shows a mean percentage of UCEs equal to 34.083% (SD=3.131%). The number of UCEs stabilizes if additional test data are added at retraining. The logistic regression models show a similar pattern. However, considering the `Adult-Income` dataset and the random forest models, the percentage of UCEs increases monotonically as the number of test data increases.

#### 2) SCENARIO 2

Considering the `German-Credit` dataset, the use of CDA does not reduce the number of UCEs for the logistic regression models, using different percentages of **LOG** counterfactuals at retraining. In fact, on average, the percentage of UCEs

<sup>11</sup>A full description of all variables of the is found at: <https://online.stat.psu.edu/stat857/node/222/>.

**TABLE 1.** Performance on test data of the best models LOG and RF, for both German-Credit and Adult-Income datasets.

	Model	AUC	Precision	Recall	F1	Accuracy
German-Credit	LOG	0.784	0.786	0.894	0.837	0.756
	RF	0.756	0.780	0.880	0.827	0.742
Adult-Income	LOG	0.862	0.677	0.485	0.565	0.821
	RF	0.861	0.694	0.469	0.600	0.824

**TABLE 2.** Counterfactual explanations generated by DiCE for the LOG and RF models (German-Credit data). Both LOG and RF return the unfavourable outcome “not creditworthy” for the original data point (“Original”). For all counterfactuals, we show only the features whose values differ from those of the original data point.

	Original	CF (LOG)	Original	CF (RF)
Account Balance	2	4	2	-
Duration of Credit	39	-	12	9
Payment Status of Previous Credit	3	-	2	-
Purpose	6	1	0	-
Credit Amount	11760	-	836	-
Value Savings/Stocks	2	-	2	-
Length employment	4	-	2	-
Instalment	2	-	4	-
Sex & Marital Status	3	-	2	-
Guarantors	1	-	1	-
Duration in Current address	3	-	2	-
Most valuable available asset	4	-	2	-
Age	32	-	23	-
Concurrent Credits	3	-	1	-
Type of apartment	1	-	2	-
# Credits at this Bank	1	-	1	-
Occupation	3	-	2	-
# dependents	1	-	1	-
Telephone	2	-	1	-
Foreign Worker	1	-	1	-

is equal to 11.146% (SD=1.473%). This is no improvement with respect to scenario 1. However, considering the random forest models, the use of CDA at retraining supports a notable decrease in UCEs with respect to scenario 1. In fact, on average, these models shows a mean percentage of UCEs equal to 3.250% (SD=2.986%). We also note that, for both model class, the addition of different percentages of test data at retraining generally results in a high-volatility in the number of UCEs, for a fixed percentage of counterfactuals added at retraining. Considering Adult-Income data, the use of CDA strongly reduces the percentages of UCEs with respect to scenario 1 for both model classes. In fact, on average, the retrained logistic regression models show a percentage of UCEs equal to 2.024% (SD=0.269%) and the retrained random forest show a percentage of UCEs equal to 0.197% (SD=0.074%). As opposed to the German-Credit dataset case, both model classes show stability with respect to the addition of test data at retraining, for each percentage of added counterfactuals, as shown by the standard deviations of the respective distributions of UCEs.

### 3) SCENARIO 3

For both German-Credit and Adult-Income data, logistic regression and random forest models show poor robustness against different-model class UCEs, for all percentages of test data added at retraining. In particular, the logistic regressions achieve a mean percentage of UCEs on

**TABLE 3.** Counterfactual explanations generated by DiCE for the LOG and RF models (Adult-Income data). Both LOG and RF return the unfavourable outcome “≤50,000\$” for the original data point (“Original”). For all counterfactuals, we show only the features whose values differ from those of the original data point.

	Original	CF (LOG)	CF (RF)
Age	22	27	28
Workclass	Private	-	-
Education	HS-grad	Bachelors	Masters
Marital Status	Married	-	-
Occupation	Sales	-	-
Race	White	-	-
Gender	Male	-	-
Hours per Week	55	-	-

German-Credit data equal to 40.271% (SD=1.673%). Random forests shows a mean percentage of UCEs on Adult-Income data equal to 46.813% (SD=0.545%). Overall, logistic regressions shows a higher percentage of UCEs on German-Credit data, with comparable mean percentages of UCEs for different percentages of test data added at retraining. Random forests show a similar pattern on the Adult-Income dataset, instead. For all models, datasets and each fixed percentage of test data added at retraining, the distribution of UCEs is characterized by high-volatility.

### 4) SCENARIO 4

Considering German-Credit data, the use of CDA at retraining leads to a mean percentage of UCEs

**TABLE 4.** The different retraining scenarios considered in our study. The original models at  $t_0$  are either **LOG** or **RF**. The models to be retrained at  $t_{n+1}$  belong either to the logistic regression or random forest class. The different scenarios allows considering different design factors that may affect the efficacy of CDA in mitigating the emergence of UCEs in real-world applications.

	Model at $t_0$	CDA	Counterfactual type added at retraining	Model class at retraining $t_{n+1}$
Scenario 1	<b>LOG</b>	-	-	logistic regression
	<b>RF</b>	-	-	random forest
Scenario 2	<b>LOG</b>	yes	<b>LOG</b>	logistic regression
	<b>RF</b>	yes	<b>RF</b>	random forest
Scenario 3	<b>RF</b>	-	-	logistic regression
	<b>LOG</b>	-	-	random forest
Scenario 4	<b>RF</b>	yes	<b>RF</b>	logistic regression
	<b>LOG</b>	yes	<b>LOG</b>	random forest

(considering different-model counterfactuals) equal to 35.063% (SD=1.420%) for logistic regressions and 1.458% (SD=1.228%) for random forests. Therefore, the use of CDA allows to reduce the number of UCEs for both models as compared to the case where no different-model counterfactuals are used at retraining, i.e., scenario 3. The improved robustness is most notable in the case of random forest models retrained on **LOG** counterfactuals. Moreover, logistic regressions show a percentage of UCEs that is stable across different percentages of **RF** counterfactuals added at retraining. Finally, the percentages of UCEs shown by both model classes are stable with respect to the addition of test data at retraining. Considering *Adult-Income* data, the use of CDA supports a notable decrease in UCEs with respect to scenario 3 for both model classes. On the one hand, in fact, for logistic regressions the number of UCEs decreases monotonically with the increase of different-model counterfactuals at retraining, i.e., **RF** counterfactuals. The mean percentage of UCEs is equal to 8.474% (SD=2.143%), as opposed to the mean percentage equal to 44.740% (SD=1.939%) resulting from the retraining of logistic regressions in scenario 3. On the other hand, the mean percentage of UCEs for the **RF** model is equal to 0.237% (SD=0.242%), as opposed to the mean percentage equal to 46.813% (SD=0.545%) that is achieved by random forests in scenario 3.

## V. DISCUSSION AND NEXT STEPS

### A. CDA AND MACHINE LEARNING MODELS

The scenarios proposed in the study show that, overall, random forest models are more robust than logistic regressions against the emergence of UCEs when CDA is used in the retraining procedure. In particular, as shown by scenario 4, random forests are robust also against the emergence of UCEs from different-model counterfactuals, i.e., those generated by the **LOG** model. However, although the use of CDA allows reducing the percentage of UCEs also in the presence of logistic regressions, its efficacy seems to be dataset-dependent. In fact, the efficacy of CDA in the case of *German-Credit* data is less than in the case of *Adult-Income* data. In general, the robustness of logistic regressions against UCEs emerging from different-model counterfactuals is less prominent than the one shown by random forests. Overall, we argue that the more extensive hyperparameter tuning performed in

the case of random forests may have supported the search for a more robust model against UCEs.

### B. CDA AND COUNTERFACTUAL TYPES

Interestingly, the reduction of UCEs due to the use of CDA at retraining is stronger in the case of different-model class counterfactuals. This becomes particularly evident considering random forests trained on *Adult-Income* data. In particular, in the case of same-model class counterfactuals, although CDA contributes to reduce the number of UCEs considering logistic regressions on *Adult-Income* data, no reduction happens on *German-Credit* data overall. In general, the percentage of same-model or different-model class counterfactuals added at retraining does not seem to affect the percentage of UCEs that are generated by both model classes at retraining. However, logistic regressions on *Adult-Income* data benefit from the increase of the percentage of counterfactuals at retraining, as this is associated to a reduction of UCEs.

### C. CDA AND “NEW DATA”

Finally, the percentage of “new” data added at retraining affects the percentage of UCEs generated by different models differently. On the one hand, on *German-Credit* data the percentage of UCEs generated by logistic regression models does not vary significantly considering different (non-zero) percentages of “new” data at retraining, as shown by scenario 1 and 3. On the other hand, on *Adult-Income* data the percentage of UCEs generated by logistic regression models reaches its maximum when a percentage of “new” data corresponding to 20% of the training data (including counterfactuals) is considered at retraining. Random forest models generate percentages of UCEs that do not vary significantly considering different (non-zero) percentages of “new” data if no counterfactual is added at retraining on *German-Credit* data, or **LOG** counterfactuals are added considering *Adult-Income* data. On the same dataset, the percentage of UCEs increases monotonically as a function of the percentage of test data.

In summary, our study shows that using CDA at the retraining of machine learning models allows reducing the number of UCEs. Therefore, CDA improves the robustness of the counterfactual explanations that are generated over time by

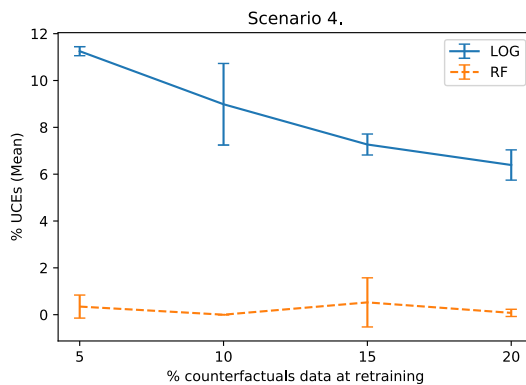
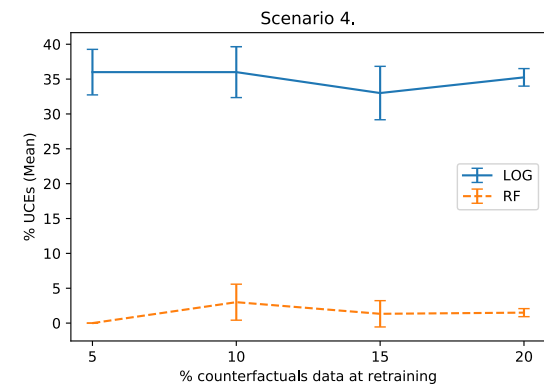
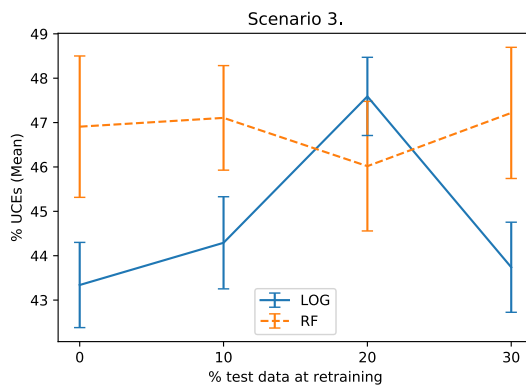
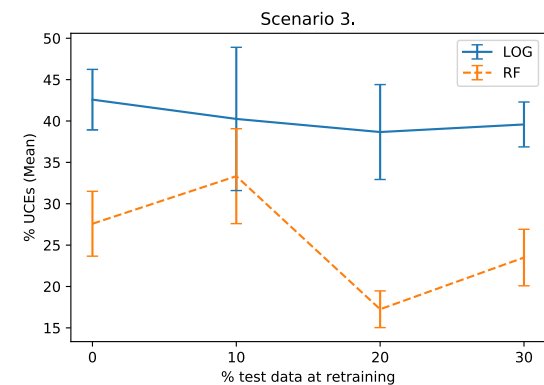
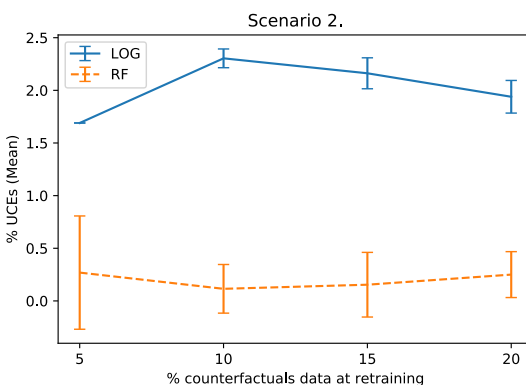
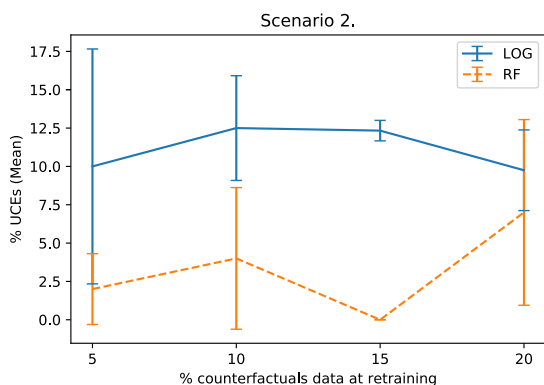
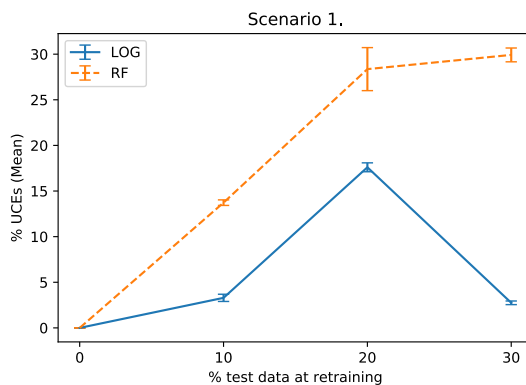
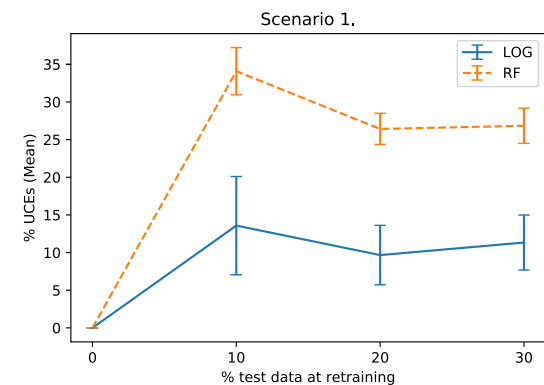


FIGURE 3. German-Credit dataset: retraining scenarios.

FIGURE 4. Adult-Income dataset: retraining scenarios.

machine learning models. However, the efficacy of CDA in addressing the emergence of UCEs at retraining depends on different design factors defining the retraining scenario,

such as the percentage and type of counterfactual added at retraining, as well as the percentage of “new” data. In particular, considering different percentages of counterfactuals



used at retraining allows simulating real-world scenarios where institutions show different levels of maturity in providing explanations of the machine learning outcomes.<sup>12</sup> Therefore, in applications, different retraining procedures could be pursued to identify empirically the most robust one with respect to the preservation of counterfactual explanations. In addition to the design factors proposed in the study, institutions may consider also additional ones, such as performance-complexity trade-offs, fairness considerations or MLOps-specific requirements.<sup>13</sup>

Finally, research is needed to improve the formalization of the concept of counterfactual “commitment” between institutions and individuals. An example is provided by a recent perspective on trust in AI, called “contractual trust” [56]. Following this perspective, one could argue that the provision of counterfactuals to support algorithmic recourse could be considered as a contract of AI systems whose outcomes have to be explained [56]. Contracts are explicit functionalities that the AI is supposed to maintain, such as a given level of accuracy [56]. Holding the counterfactual contract contributes to the trustworthiness of the algorithm-supported service offered by the institution [56], [57]. Therefore, in the contractual trust account, the “commitment” between the institution and the individuals is that the institution holds the counterfactual contract, maintaining the trustworthiness of the AI, according to the strategy in place against the emergence of the UCEs. This strategy needs to publicly declare the conditions under which counterfactual commitments will be respected and the UCEs have to be considered as normatively acceptable [58].

## VI. LIMITATIONS

This study has several limitations. Although the use of two datasets is in line with relevant literature on counterfactual explanations [8], [16], [41], [45], and the *German-Credit* and *Adult-Income* datasets are widely used in the xAI literature, to support the generalizability of our results and further test the efficacy of CDA, additional studies focusing on real-world datasets and different machine learning applications are needed. Moreover, we tested CDA on two classes of machine learning model, only. We also note that the proposed CDA method is applicable to those cases where the set of features used to generate counterfactuals and to retrain models is kept fixed. Therefore, our proposal is not applicable to those use cases where the set of features changes, for example, due to the addition of data sources over time. Additionally, in our study we considered a fixed algorithm  $C$  to compute counterfactuals. However, in real-world applications, the algorithm may change, for example, by choosing

<sup>12</sup>This level of maturity is represented, for example, by the infrastructure deployed for the generation and distribution of counterfactuals.

<sup>13</sup>However, in real-world applications interpretability constraints may pre-select the models to be retrained with CDA. For example, in clinical applications, physicians may prefer to train logistic regressions as opposed to, for example, gradient boosting machines as the former are commonly used to predict clinically-relevant patient outcomes and they are easy to interpret [55].

different sets of hyperparameters. The effect of changing an algorithm to compute counterfactuals on the time robustness of the explanations is left to future investigations. Finally, the proposed CDA method is a peculiar example of performative prediction [59], i.e., a shift of data distribution induced by the use of machine learning models and algorithms to compute explanations.<sup>14</sup> However, a rigorous formulation of a counterfactual-induced performative prediction (and its possible interaction with other types of distribution shifts) is not yet available.

## VII. CONCLUSION

Counterfactual explanations are a class of explanations of machine learning outcomes with interesting properties. In fact, the possibility of suggesting a strategy to have recourse against a machine learning model outcome is a useful tool available to those affected by AI-assisted decisions. However, the change over time, due to retraining, of machine learning models may give rise to the possibility of invalidating the efforts spent to implement the scenario suggested by the counterfactual explanation. This possibility, that we called “unfortunate counterfactual event”, *de facto* impedes the use of counterfactuals in real-world applications and represents a risk to the trustworthiness of the AIs that institutions may use to deliver their services and products. Therefore, our approach is to improve the robustness of counterfactual explanations over time by managing the emergence of UCEs. To do so, we proposed to use counterfactual data augmentation every time machine learning models are retrained. This approach—which is of easy implementation—is a first step towards a systematic analysis of the methods to ensure the consistent use of counterfactual explanations in real-world applications, and support trust in institutions, and their AIs, while striving for the interpretability of machine learning models. However, as different model retraining scenarios may lead to different degrees of protection against the emergence of UCEs, further empirical investigations need to take into account the different rationales and pragmatic choices characterizing the deployment of machine learning models in real-world applications.

## APPENDIX

### A. HYPERPARAMETER TUNING: RESULTS

**TABLE 5.** Hyperparameters of the best models, as resulting from 5-fold cross-validation on train data.  $C$ =inverse of regularization strength,  $n_{estimators}$ = number of trees in the ensemble,  $max\_depth$ = maximal tree depth.

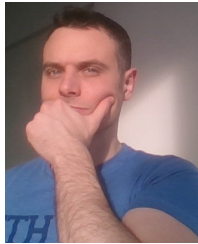
	Model	Hyperparameters
German-Credit	LOG	$C=1$
	RF	$n_{estimators}=10, max\_depth=7$
Adult-Income	LOG	$C=1$
	RF	$n_{estimators}=400, max\_depth=9$

<sup>14</sup>More precisely, performative prediction is an *endogenous* type of data distribution shift. On the contrary, a global financial crisis or a pandemic changing the populations of customers or patients represent *exogenous* types of distribution shifts [59].

## REFERENCES

- [1] F. Doshi-Velez and B. Kim, "Towards a rigorous science of interpretable machine learning," 2017, *arXiv:1702.08608*.
- [2] Z. C. Lipton, "The Mythos of model interpretability: In machine learning, the concept of interpretability is both important and slippery," *Queue*, vol. 16, no. 3, pp. 31–57, Jun. 2018.
- [3] T. Miller, "Explanation in artificial intelligence: Insights from the social sciences," *Artif. Intell.*, vol. 267, pp. 1–38, Feb. 2019.
- [4] G. Riccardo, A. Monreale, S. Ruggieri, F. Turini, F. Giannotti, and D. Pedreschi, "A survey of methods for explaining black box models," *ACM Comput. Surv.*, vol. 51, no. 5, pp. 1–42, 2018.
- [5] B. Mittelstadt, C. Russell, and S. Wachter, "Explaining explanations in AI," in *Proc. Conf. Fairness, Accountability, Transparency*, Jan. 2019, pp. 279–288.
- [6] W. Samek and K.-R. Müller, "Towards explainable artificial intelligence," in *Explainable AI: Interpreting, Explaining and Visualizing Deep Learning*. Cham, Switzerland: Springer, 2019, pp. 5–22.
- [7] D. S. Watson and L. Floridi, "The explanation game: A formal framework for interpretable machine learning," in *Ethics, Governance, and Policies in Artificial Intelligence*. Cham, Switzerland: Springer, 2021, pp. 185–219.
- [8] S. Wachter, B. Mittelstadt, and C. Russell, "Counterfactual explanations without opening the black box: Automated decisions and the GDPR," *Harv. J. Law Technol.*, vol. 31, no. 2, p. 841, 2018.
- [9] D. Martens and F. Provost, "Explaining data-driven document classifications," *MIS Quart.*, vol. 38, no. 1, pp. 73–100, 2014.
- [10] R. K. Mothilal, A. Sharma, and C. Tan, "Explaining machine learning classifiers through diverse counterfactual explanations," in *Proc. Conf. Fairness, Accountability, Transparency*, Jan. 2020, pp. 607–617.
- [11] R. Guidotti, A. Monreale, F. Giannotti, D. Pedreschi, S. Ruggieri, and F. Turini, "Factual and counterfactual explanations for black box decision making," *IEEE Intell. Syst.*, vol. 34, no. 6, pp. 14–23, Nov. 2019.
- [12] A. Artelt, V. Vaquet, R. Velioglu, F. Hinder, J. Brinkrolf, M. Schilling, and B. Hammer, "Evaluating robustness of counterfactual explanations," in *Proc. IEEE Symp. Ser. Comput. Intell. (SSCI)*, Dec. 2021, pp. 1–9.
- [13] T. Laugel, M.-J. Lesot, C. Marsala, X. Renard, and M. Detyniecki, "Inverse classification for comparison-based interpretability in machine learning," 2017, *arXiv:1712.08443*.
- [14] M. Loi, A. Ferrario, and E. Viganò, "Transparency as design publicity: Explaining and justifying inscrutable algorithms," *Ethics Inf. Technol.*, vol. 23, no. 3, pp. 253–263, Sep. 2021.
- [15] D. Mahajan, C. Tan, and A. Sharma, "Preserving causal constraints in counterfactual explanations for machine learning classifiers," 2019, *arXiv:1912.03277*.
- [16] R. Poyiadzi, K. Sokol, R. Santos-Rodriguez, T. De Bie, and P. Flach, "FACE: Feasible and actionable counterfactual explanations," in *Proc. AAAI/ACM Conf. AI, Ethics, Soc.*, Feb. 2020, pp. 344–350.
- [17] M. Virgolin and S. Fracaras, "On the robustness of counterfactual explanations to adverse perturbations," 2022, *arXiv:2201.09051*.
- [18] S. Barocas, A. D. Selbst, and M. Raghavan, "The hidden assumptions behind counterfactual explanations and principal reasons," in *Proc. Conf. Fairness, Accountability, Transparency*, Jan. 2020, pp. 80–89.
- [19] J. A. Kroll, J. Huey, S. Barocas, E. W. Felten, J. R. Reidenberg, D. G. Robinson, and H. Yu, "Accountable algorithms," *Univ. Pennsylvania Law Rev.*, vol. 165, pp. 633–705, Feb. 2017.
- [20] S. Venkatasubramanian and M. Alfano, "The philosophical basis of algorithmic recourse," in *Proc. Conf. Fairness, Accountability, Transparency*, Jan. 2020, pp. 284–293.
- [21] D. Kaushik, E. Hovy, and Z. C. Lipton, "Learning the difference that makes a difference with counterfactually-augmented data," 2019, *arXiv:1909.12434*.
- [22] D. Kaushik, A. Setlur, E. Hovy, and Z. C. Lipton, "Explaining the efficacy of counterfactually augmented data," 2020, *arXiv:2010.02114*.
- [23] D. Teney, E. Abbasnedjad, and A. V. D. Hengel, "Learning what makes a difference from counterfactual examples and gradient supervision," in *Proc. Eur. Conf. Comput. Vis. Glasgow, U.K.: Springer*, Aug. 2020, pp. 580–599.
- [24] K. Lu, P. Mardziel, F. Wu, P. Amancharla, and A. Datta, "Gender bias in neural natural language processing," in *Logic, Language, and Security*. Cham, Switzerland: Springer, 2020, pp. 189–202.
- [25] Q. Liu, M. Kusner, and P. Blunsom, "Counterfactual data augmentation for neural machine translation," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics, Hum. Lang. Technol.*, 2021, pp. 187–197.
- [26] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic minority over-sampling technique," *J. Artif. Intell. Res.*, vol. 16, no. 1, pp. 321–357, 2002.
- [27] M. Temraz and M. T. Keane, "Solving the class imbalance problem using a counterfactual method for data augmentation," 2021, *arXiv:2111.03516*.
- [28] R. Kohavi and B. Becker. (1996). *UCI Machine Learning Repository: Adult Data Set*. [Online]. Available: <https://archive.ics.uci.edu/ml/datasets/adult>
- [29] M. T. Ribeiro, S. Singh, and C. Guestrin, "Why should I trust you? Explaining the predictions of any classifier," in *Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2016, pp. 1135–1144.
- [30] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," in *Proc. 31st Int. Conf. Neural Inf. Process. Syst.*, 2017, pp. 4768–4777.
- [31] A.-H. Karimi, B. Schölkopf, and I. Valera, "Algorithmic recourse: From counterfactual explanations to interventions," in *Proc. ACM Conf. Fairness, Accountability, Transparency*, Mar. 2021, pp. 353–362.
- [32] N. Aggarwal, "The norms of algorithmic credit scoring," *Cambridge Law J.*, vol. 80, no. 1, pp. 42–73, Mar. 2021.
- [33] A. Ferrario, B. Demiray, K. Yordanova, M. Luo, and M. Martin, "Social reminiscence in older adults' everyday conversations: Automated detection using natural language processing and machine learning," *J. Med. Internet Res.*, vol. 22, no. 9, Sep. 2020, Art. no. e19133.
- [34] A. Ferrario, M. Luo, A. J. Polsinelli, S. A. Moseley, M. R. Mehl, K. Yordanova, M. Martin, and B. Demiray, "Predicting working memory in healthy older adults using real-life language and social context information: A machine learning approach," *JMIR Aging*, vol. 5, no. 1, Mar. 2022, Art. no. e28333.
- [35] D. G. Kyrolos and J. R. Green, "MetaHate: A meta-model for hate speech detection," in *Proc. IEEE Int. Conf. Big Data*, Dec. 2021, pp. 2496–2502.
- [36] A. Singh, S. Sengupta, and V. Lakshminarayanan, "Explainable deep learning models in medical image analysis," *J. Imag.*, vol. 6, no. 6, p. 52, Jun. 2020.
- [37] A.-H. Karimi, G. Barthe, B. Schölkopf, and I. Valera, "A survey of algorithmic recourse: Contrastive explanations and consequential recommendations," *ACM Comput. Surv.*, vol. 2022, pp. 1–14, Apr. 2022.
- [38] M. Benk and A. Ferrario. (2020). *Explaining Interpretable Machine Learning: Theory, Methods and Applications*. [Online]. Available: <https://ssrn.com/abstract=3748268>
- [39] I. Stepin, J. M. Alonso, A. Catala, and M. Pereira-Farina, "A survey of contrastive and counterfactual explanation generation methods for explainable artificial intelligence," *IEEE Access*, vol. 9, pp. 11974–12001, 2021.
- [40] J. Pearl, *Causality*. Cambridge, U.K.: Cambridge Univ. Press, 2009.
- [41] C. Russell, "Efficient search for diverse coherent explanations," in *Proc. Conf. Fairness, Accountability, Transparency*, Jan. 2019, pp. 20–28.
- [42] T. Mitchell, *Machine Learning*. New York, NY, USA: McGraw-Hill, 1997.
- [43] S. Verma, J. Dickerson, and K. Hines, "Counterfactual explanations for machine learning: A review," 2020, *arXiv:2010.10596*.
- [44] K. Rawal, E. Kamar, and H. Lakkaraju, "Algorithmic recourse in the wild: Understanding the impact of data and model shifts," 2020, *arXiv:2012.11788*.
- [45] M. Pawelczyk, K. Broelemann, and G. Kasneci, "On counterfactual explanations under predictive multiplicity," in *Proc. Conf. Uncertainty Artif. Intell.*, 2020, pp. 809–818.
- [46] C. Marx, F. Calmon, and B. Ustun, "Predictive multiplicity in classification," in *Proc. Int. Conf. Mach. Learn.*, 2020, pp. 6765–6774.
- [47] B. Ustun, A. Spangher, and Y. Liu, "Actionable recourse in linear classification," in *Proc. Conf. Fairness, Accountability, Transparency*, Jan. 2019, pp. 10–19.
- [48] R. Zmigrod, S. J. Mielke, H. Wallach, and R. Cotterell, "Counterfactual data augmentation for mitigating gender stereotypes in languages with rich morphology," 2019, *arXiv:1906.04571*.
- [49] S. Pitis, E. Creager, and A. Garg, "Counterfactual data augmentation using locally factored dynamics," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 33, 2020, pp. 3976–3990.
- [50] S. Levanon and N. Rosenfeld, "Strategic classification made practical," in *Proc. Int. Conf. Mach. Learn.*, 2021, pp. 6243–6253.
- [51] A. Fernández, S. García, M. Galar, R. C. Prati, B. Krawczyk, and F. Herrera, *Learning From Imbalanced Data Sets*, vol. 10. Cham, Switzerland: Springer, 2018.
- [52] B. Krawczyk, "Learning from imbalanced data: Open challenges and future directions," *Prog. Artif. Intell.*, vol. 5, no. 4, pp. 221–232, 2016.
- [53] H. Zhu. (2016). *Predicting Earning Potential Using the Adult Dataset*. [Online]. Available: [https://rpubs.com/H\\_Zhu/235617](https://rpubs.com/H_Zhu/235617)

- [54] H. Hofmann. (2013). *UCI Machine Learning Repository: Statlog (German Credit Data) Data Set*. [Online]. Available: [https://archive.ics.uci.edu/ml/datasets/statlog+\(german+credit+data\)](https://archive.ics.uci.edu/ml/datasets/statlog+(german+credit+data))
- [55] J. E. Zimmerman, A. A. Kramer, D. S. McNair, and F. M. Malila, "Acute physiology and chronic health evaluation (APACHE) IV: Hospital mortality assessment for today's critically ill patients," *Crit. Care Med.*, vol. 34, no. 5, pp. 1297–1310, May 2006.
- [56] A. Jacovi, A. Marasović, T. Miller, and Y. Goldberg, "Formalizing trust in artificial intelligence: Prerequisites, causes and goals of human trust in AI," in *Proc. ACM Conf. Fairness, Accountability, Transparency*, Mar. 2021, pp. 624–635.
- [57] A. Ferrario and M. Loi, "How explainability contributes to trust in AI," in *Proc. ACM Conf. Fairness, Accountability, Transparency*, New York, NY, USA, Jun. 2022, pp. 1457–1466.
- [58] A. Ferrario and M. Loi, "A series of unfortunate counterfactual events: The role of time in counterfactual explanations," 2020, *arXiv:2010.04687*.
- [59] J. Perdomo, T. Zmic, C. Mendler-Dünner, and M. Hardt, "Performative prediction," in *Proc. Int. Conf. Mach. Learn.*, 2020, pp. 7599–7609.



**ANDREA FERRARIO** received the Ph.D. degree in mathematics from ETH Zurich, in 2012. He has worked as a Consultant in analytics and AI for five years before his return to ETH, in 2018. Since then, he has been a Postdoctoral Researcher at the Chair of Technology Marketing and the Scientific Director of the Mobiliar Lab for Analytics, ETH Zurich. His research interests include the intersection between philosophy and the applications of technology, with a focus on AI and mixed reality.

His interests comprise the ethics and epistemology of AI, the use of natural language processing and machine learning for digital health interventions, and the use of immersive augmented reality to solve problems on the interpretability of machine learning models collaboratively.



**MICHELE LOI** is a Political Philosopher and a Technologist. Since September 2021, he has been a Postdoctoral Researcher on a research project on fairness in prediction for health-related purposes with the Department of Mathematics, Politecnico di Milano, and in the context of the META Group, funded by a Marie Skłodowska Curie Individual Fellowship. He also co-leads a team of computer scientists, empirical social scientists, and philosophers at the University of Zurich and Zurich University of Applied Science in a Swiss National Science Foundation-funded interdisciplinary project on fairness in algorithmic decision making.

...