UNIVERSITY OF TECHNOLOGY SYDNEY
Faculty of Engineering and Information Technology

Causality for Interpretable Machine Learning

by

Tri Dung Duong

A Thesis Submitted
in Fulfillment of the
Requirements for the Degree

Doctor of Philosophy

Sydney, Australia

2023

Certificate of Original Authorship

I, Tri Dung Duong declare that this thesis, is submitted in fulfilment of the requirements for the award of Doctor of Philosophy, in the School of Computer Science, Faculty of Engineering and Information Technology at the University of Technology Sydney.

This thesis is wholly my own work unless otherwise referenced or acknowledged. In addition, I certify that all information sources and literature used are indicated in the thesis.

This document has not been submitted for qualifications at any other academic institution.

This research is supported by the Australian Government Research Training Program.

Signature:

Date: 01 March 2023

Abstract

The past few years have borne witness to a marked surge in the adoption of machine learning (ML) techniques across a broad spectrum of fields, such as image analysis, text categorization, predictive credit scoring, and recommendation systems, among others. These techniques have made significant strides in various sectors, yet there is a growing concern among researchers about the "black-box" nature intrinsic to these methods. As a consequence, the need for interpreting machine learning models has taken center stage in scholarly debates. However, conventional approaches to machine learning interpretability have primarily focused on associative relationships rather than causal ones.

This study seeks to bridge the existing gap in the causal interpretation of machine learning models by developing and enhancing both causal inference and counterfactual methodologies. Initially, it offers a comprehensive review of the causal analysis techniques utilized in machine learning models. Following this, the research proposes an innovative approach to causal inference, one that is anchored in the concept of dynamic propensity scores. In the context of counterfactual explanation, the study brings forward two strategies: one that prioritizes causality to safeguard the causal bonds within counterfactual instances, and another that utilizes a framework based on normalizing flows, designed to yield scalable and robust counterfactual samples. Concerning counterfactual fairness, the study aspires to formulate a min-max strategy designed to achieve counterfactual fairness even within an imperfect structural causal model. Collectively, this research is committed to enhancing the interpretability of machine learning models through the provision of causal explanations and counterfactual analyses.

To my loved ones

Acknowledgments

Completing a Ph.D. is a challenging endeavor that requires commitment, hard work, and a strong support system. Looking back on my journey, I am filled with gratitude for my supervisors, friends, and family, whose support was invaluable to my success. The task of writing this thesis was no small feat, and I wish to express my sincere thanks to those who have accompanied me on this journey.

Firstly, my heartfelt appreciation goes to my primary supervisor, Professor Guandong Xu, for his consistent support, motivation, and inspiration throughout my Ph.D. studies. His guidance has significantly contributed to my development as a researcher, and I am confident his mentorship will continue to impact my career positively. Despite the challenges brought about by the COVID-19 pandemic, Professor Xu's guidance ensured that my research progress was not adversely affected. The experiences and skills I have gained under his supervision are invaluable, and these memories will be cherished.

A large part of this thesis is built upon peer-reviewed publications. I wish to express my gratitude to the anonymous reviewers whose feedback and suggestions greatly enhanced the quality of my research papers. Their efforts were crucial in shaping this thesis, and I am appreciative of their contributions.

I also extend my thanks to my dear friends who supported me throughout my Ph.D. journey: my co-supervisor Qian Li, Hamad Zogan and his family, Rafiq Md, Xiangmeng Wang, Thac Do, Thanh Tung Khuat, Sunny Verma, among others, provided constant support, encouragement, and motivation. The shared conversations, laughter, and camaraderie over the past three years will always hold a special place in my heart.

Lastly, I convey my deep love and gratitude to my parents and sister for their

unwavering support and sacrifices. They have been my pillars of strength, providing me with the necessary financial and emotional support, and allowing me to concentrate on my Ph.D. research. Their selflessness and unconditional love are beyond words.

In conclusion, I owe my success to the countless individuals who've supported me throughout my Ph.D. journey. Without their persistent encouragement and motivation, completing this thesis wouldn't have been feasible. I am eternally thankful for their influence in my life and I eagerly look forward to advancing on my path with their continuous support and guidance.

Tri Dung Duong

Sydney, Australia, 2023.

List of Publications

Published Papers

- **Duong, T. D.**, Li, Q., & Xu, G. (2022) CeFlow: A robust and efficient counterfactual explanation framework with normalizing flows. Advances in Knowledge Discovery and Data Mining PAKDD 2023.

- Li, Q., **Duong, T. D.**, Wang, Z., Liu, S., Wang, D., & Xu, G. (2021, October). Causal-aware generative imputation for automated underwriting. In Proceedings of the 30th ACM International Conference on Information & Knowledge Management (pp. 3916-3924).

- **Duong, T. D.**, Li, Q., & Xu, G. (2022). Stochastic intervention for causal inference via reinforcement learning. Neurocomputing, 482, 40-49.

- **Duong, T. D.**, Li, Q., & Xu, G. (2021, July). Stochastic intervention for causal effect estimation. In 2021 International Joint Conference on Neural Networks (IJCNN) (pp. 1-8). IEEE.

- Xu, G., **Duong, T. D.**, Li, Q.(2021). Causality Learning: A New Perspective for Interpretable Machine Learning. IEEE Intelligent Informatics.

Under Reviewed Papers

- **Duong, T. D.**, Li, Q., & Xu, G. (2022) Achieving Counterfactual Fairness with Imperfect Structural Causal Model.

- **Duong, T. D.**, Li, Q., & Xu, G. (2022) Causality-based counterfactual explanation for classification models.

Contents

List of Figures

List of Tables

List of Abbreviations

CE	Counterfactual explanation
ML model	Machine learning model
CF	Counterfactual fairness
ATE	Average treatment effect
XAI	Explainable artificial intelligence
IML	Interpretable machine learning
SCM	Structural causal model
GBoostR	Gradient boosting regression
GBoostC	Gradient boosting classifier
IML	Interpretable machine learning
RMSE	Root mean squared error
MAE	Mean absolute error
TPR	true positive rate
TNR	true negative rate
NF	Normalizing flows
RF	Reinforcement learning

List of Notations

If there is no specific definition in each section, the following are the default meanings of notations used in this thesis.

Symbol	**Meaning**
τ_{ATE}	Average treament effect
$p_t(\boldsymbol{x})$	Propensity score
X	Random scalars
$\boldsymbol{X}$	Random vectors
$\mathcal{L}(.)$	Loss function
$\mathcal{D} = \{x_i, s_i, y_i\}_{i=1}^n$	A dataset consisting of n instances

Part I

Introduction

Chapter 1

Introduction

In recent years, machine learning has ushered in substantial changes across a myriad of sectors, such as healthcare, finance, and transportation. Nevertheless, the pervasive use of these sophisticated machine learning models prompts crucial questions about their transparency and interpretability. When these models become convoluted and challenging to comprehend, it hampers our ability to trust their outcomes and make informed decisions based on them.

Causality, or the study of cause and effect relationships between events, presents an appealing solution to this predicament. Incorporating causal knowledge into machine learning models may open up avenues to simultaneously achieving high precision and interpretability, while also dodging common obstacles such as deceptive correlations or confounding factors.

This thesis aims to delve into the intersectionality of causal inference and machine learning. Specifically, we want to explore how the concept of causality can improve the understanding and transparency of machine learning systems. Our focus will be on several essential questions, like how counterfactual analysis can help clarify machine learning models and how structural causal models can enhance the fairness of these models. To reach this objective, we will first conduct a thorough review of the prevalent literature on causality and machine learning, and then deploy a variety of methods and techniques—normalizing flows, structural causal models, stochastic propensity score, etc.—to investigate the power of causality for interpretability in machine learning.

The outcomes of this thesis could potentially guide the design of reliable and trustworthy machine learning systems, fostering novel applications in domains such as explainable AI, decision support, and policy-making. Through the lens of causal-

ity, we could develop machine learning models that are not just accurate and efficient but are also transparent and comprehendible to human users.

1.1 Introduction

Machine learning has undoubtedly been a force of transformation across an array of sectors, including healthcare, finance, and transportation. However, as the utilization of these advanced models intensifies, their transparency and interpretability are increasingly under scrutiny. The complex architecture of these models can bring about significant challenges in fully trusting their results and effectively using them for informed decision-making. This issue's criticality has catalyzed the search for viable solutions, making the interpretability of machine learning indispensable in rendering these models more comprehensible to humans.

However, conventional machine learning methods usually center more on identifying patterns or associations, rather than unearthing causal relationships. Causality, the study of cause-effect connections, provides an appealing resolution to this issue. By integrating causal understanding into machine learning models, we could potentially strike a balance between high accuracy and easy interpretability. This approach also helps us avoid common issues such as misleading correlations or variables that can confound the analysis.

Our research strives to unearth how the concept of causality can bolster the clarity and transparency of machine learning systems. Our inquiry will concentrate on vital topics, such as the value of counterfactual analysis in explaining and interpreting machine learning models and how structural causal models can facilitate the fairness of these models. As a first step, we will conduct a thorough review of the existing works on causality and machine learning to establish a solid understanding of the field. Following this, we will employ various methods and techniques that include normalizing flows, structural causal models, and the concept of stochastic propensity scores. This approach will enable us to explore the potential and effectiveness of causality in enhancing the interpretability of machine learning.

The results of our study could guide the development of future machine learning

systems, helping make them more reliable and trustworthy. In addition, our findings could stimulate advances in various areas such as explainable artificial intelligence, decision support, and policy-making. By leveraging the power of causality, we aim to build machine learning models that strike a good balance between accuracy, efficiency, transparency, and ease of use. In the bigger picture, this research is set to substantially contribute to the advancement of machine learning, helping it better address the ongoing issues of transparency and interpretability.

1.2 Research Objectives

To achieve the overarching aims of this research, it is necessary to address the following major objectives in detail:

- **Conducting a comprehensive survey for interpretable machine learning under causal perspective**: Recognizing and evaluating existing methods is a crucial step towards developing novel solutions. Therefore, this objective entails conducting an exhaustive assessment of existing learning algorithms and categorizing them in a meaningful manner. This will involve a thorough review of the literature, including recent advances in the field of interpretable machine learning under a causal perspective. The review will be conducted with a focus on identifying the strengths and weaknesses of existing methods, and evaluating their applicability to various real-world scenarios. This comprehensive survey will not only establish a solid foundation for the development of our proposed algorithms in this thesis, but also provide valuable insights for other objectives.

- **Development of stochastic propensity score for estimating average treatment effect**: Building upon the insights gained from the comprehensive survey, this objective focuses on the construction of a stochastic propensity score for causal inference. The proposed approach aims to deal with stochastic propensity score instead of static ones, and then develop the policy optimization algorithms based on reinforcement learning.

- **Development of counterfactual explanation algorithms using multi-objective optimization and structural causal model**: This objective proposes an improved version of counterfactual explanation algorithms, which will enhance the quality of counterfactual samples. This will be achieved through the use of multi-objective optimization and a structural causal model, which will provide a more accurate representation of the underlying causal mechanisms. The multi-objective optimization will ensure that the counterfactual explanations are both accurate and interpretable, while the structural causal model will provide a more accurate representation of the causal relationships between variables. The completed algorithm will be evaluated against existing counterfactual explanation methods to assess its performance in generating high-quality explanations.

- **Investigation of counterfactual explanation using normalizing flows**: This objective aims to further enhance the robustness and stability of counterfactual samples. By incorporating invertible neural networks with a Gaussian Mixture Model, the objective leverages the promising technique of normalizing flows. This approach directly applies recent advancements in generative modeling to the development of counterfactual explanations. Building on the methodology of previous objectives, this investigation represents a logical progression in refining the quality of counterfactual explanations.

- **Investigation of counterfactual fairness with imperfect structural causal model**: This final objective addresses an important consideration in real-world applications. It seeks to achieve counterfactual fairness even when the structural causal model is imperfect or incomplete. Building on the insights gained from the previous objectives, this objective involves the development of a deep learning model capable of handling situations where the structural causal model is not fully understood. By doing so, it directly applies the knowledge and methodology developed in earlier objectives to address an essential aspect of fairness in machine learning models.

A summary of research motivation, research objectives, and our contributions is presented in Figure 1.1.

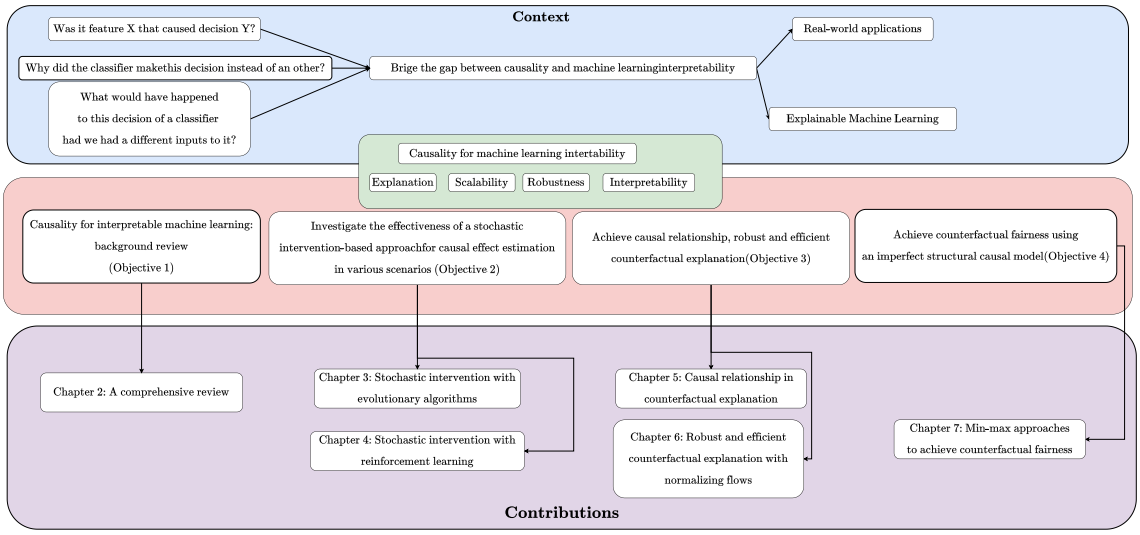


Figure 1.1 : A summary of motivation, objectives, and contributions in this thesis.

1.3 Thesis Organization

The organization of this thesis is presented in Figure 1.2. In this introductory chapter, I have discussed the motivation, objectives, and contributions of my Ph.D. project, which aims to provide a comprehensive understanding of interpretable machine learning from a causality perspective.

Chapter 2: This chapter presents a detailed review of the advancements made in the field of interpretable machine learning, with a special emphasis on causality. Various methods such as causal inference, counterfactual explanation, counterfactual fairness, and structural causal models are discussed extensively. The chapter also emphasizes the importance of infusing causality into machine learning models to significantly enhance their interpretability and improve the decision-making process.

Chapter 3: This chapter delves into the field of causal inference, providing fundamental knowledge about potential outcome frameworks. We illuminate the concepts of causality, counterfactuals, and identification in the context of Structural Causal Models (SCMs). Moreover, we put forward a new approach for estimating Average

Treatment Effects (ATEs) by using stochastic propensity scores. This method is proposed to overcome the limitations of existing methods used for estimating ATEs.

Chapter 4: This chapter extends the approach from *Chapter 3*, utilizing reinforcement learning and stochastic propensity scores for policy optimization.

Chapter 5: This chapter introduces a causality-focused strategy for generating counterfactual explanations by employing multi-objective optimization. The proposed method aims to produce counterfactual samples that not only fulfill the expected outcomes but also retain diverse and causal features. The effectiveness of this method is assessed on multiple real-world datasets, and its performance is compared with existing methods.

Chapter 6: This chapter delves into the use of normalizing flows for counterfactual explanations. Normalizing flows, a kind of model adept at understanding the hidden distributions in a dataset, have great potential. We propose an approach that brings together normalizing flows and counterfactual explanation. This method stands out for its ability to generate robust samples effectively.

Chapter 7: This chapter introduces the concept of counterfactual fairness using an imperfect structural causal model. We discuss the challenges of achieving fairness in machine learning models and propose a method with an imperfect structural causal models (SCMs).

Chapter 8: This chapter concludes the thesis by discussing the key findings and potential research directions for further study.

By following this organized structure, this thesis aims to provide a comprehensive understanding of interpretable machine learning from a causality perspective, as well as introduce novel approaches for causal effect estimations, generating counterfactual explanations and achieving counterfactual fairness.

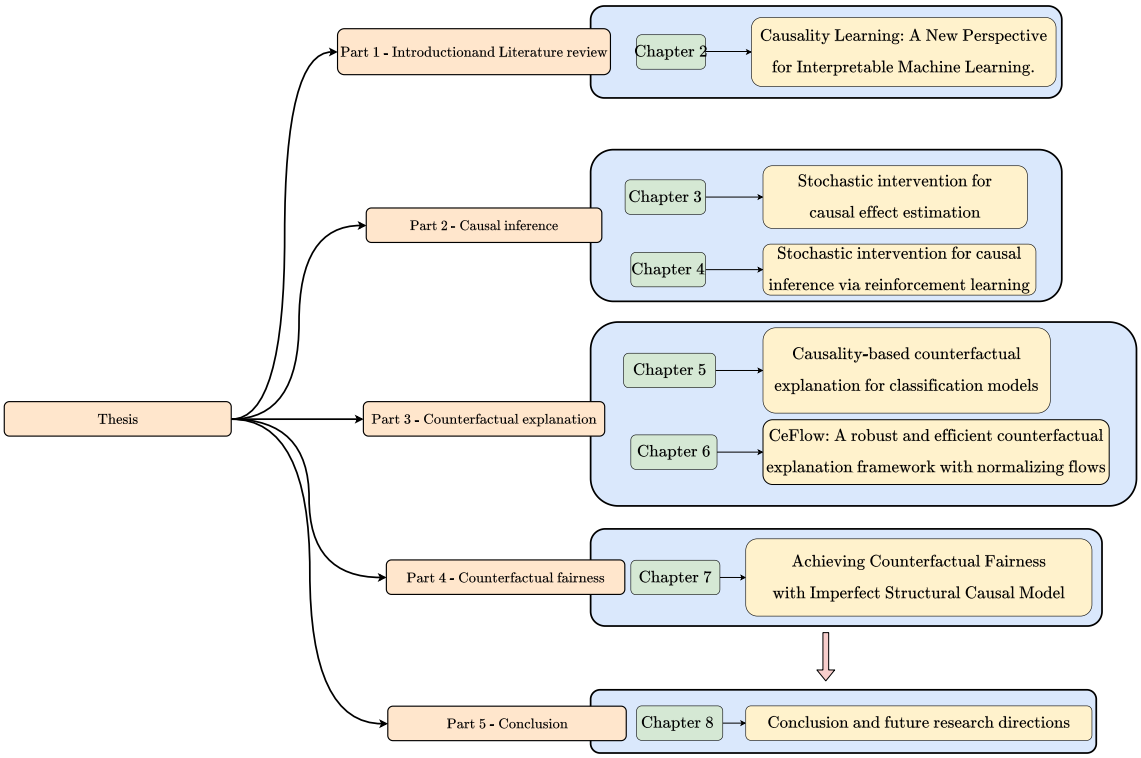


Figure 1.2 : Thesis structure

Chapter 2

Literature Review

This chapter presents a comprehensive overview of causal analysis, beginning with fundamental background information and crucial concepts. It then proceeds to summarize the most recent causal approaches employed in interpretable machine learning. The chapter also discusses various evaluation techniques used for assessing the quality of these methods and highlights open issues in causal interpretability. The content of this chapter is adapted from the following paper:

- Xu, G., **Duong, T. D.**, Li, Q.(2021). Causality Learning: A New Perspective for Interpretable Machine Learning. IEEE Intelligent Informatics.

2.1 Introduction

In the past decades, machine learning has achieved the impressive performance in diverse tasks, and is increasingly applied in science, society and business. However, most of state-of-the-art models remained incomprehensible for both researchers, users and engineers, causing difficulties when deploying in real world. Specifically, there are several high-stake decision-making domains such as self-driving cars, crime prediction or personalized medicine in which the lack of transparency in machine learning prevents themselves from being adopted. Take for instance, in the health-care sector where each decision can affect the people's survival, physicians are frequently concerned about the safety and trust of any deployed models. They do not likely trust the model's prediction if they can not understand the rationales behind it. Consequently, interpretability in machine learning plays a significantly important role in generating trust-worthy models. This furthermore allows researchers, data scientists and engineers to ensure the models following the human understanding, ethnic codes, fairness and security. We as human have an insatiable curious nature;

thus, our goal is not only to understand models' mechanism but also to generate and extract new knowledge of the world.

In view of the time of explainable AI shown in Figure 2.1, interpretable machine learning can be divided into two branches: ad-hoc and post-hoc methods. The evolutionary history of noticeable traditional interpretable machine learning techniques is briefly described in the Figure 2.1. The ad-hoc type focuses on building the model architecture, algorithms or mechanisms that are self-explainable and transparent. Intrinsically interpretable models are the central research in the early years of artificial intelligence with the dominance of symbolism methods, followed by more advanced approaches such as decision sets (Lakkaraju et al. 2016), generalized linear regression, generalized additive model (Zhang et al. 2019a; Caruana et al. 2015; Zhang et al. 2019b), Bayesian probabilistic model (Darwen 2019; Letham et al. 2015b), rule-based model (Wang 2017; Letham et al. 2015a), attention mechanism (Arik and Pfister 2021), fuzzy inference systems (Jang and Sun 1993; Guillaume 2001; Wang and Lee 2002), TabNet (Arik and Pfister 2021), etc. With the rapid growth of deep learning in recent decades, machine learning model is gradually evolved into complicated and incomprehensible forms, which leads to the increasing attention on post-hoc interpretations. Several prominent approaches in this category include Local surrogate models (LIME (Ribeiro et al. 2016), SHAP (Lundberg and Lee 2017), LORE (Guidotti et al. 2018), etc), influence functions (Koh and Liang 2017) and feature importance estimation (Schwab et al. 2019; Schwab and Karlen 2019a) have been introduced.

However, traditional interpretable machine learning focuses on the association instead of the causality. With the emergence of causal inference, an increasing number of causality-oriented methods have been proposed in interpretable machine learning. In comparison with traditional methods, causal approaches can be utilized to identify causes and effects of models architecture or conduct the reasoning over its decisions and behaviors. This article examines the overview of interpretable machine learning, presents the causal analysis in machine learning interpretability and finally discusses the future research directions. More specifically, we first present

the background of causal analysis with key concepts, models and evaluation metrics. We then provide an overview of state-of-the-art works on causal interpretability. We also illustrate the potential evaluation metrics used in interpretable machine learning.

2.2 Causality Analysis

Causality analysis can exploit the causality mechanisms underlying the data-generating process, which is more advanced than the predictive or descriptive capability in machine learning techniques. Causal inference and causal discovery are two main research topics for causality analysis. The goal of causal inference is to estimate the causal effect of treatment (i.e., a decision made or action taken) on the outcome (i.e., the result of treatment). Causal discovery examines whether a set of causal relationships exists among the variables. This paper would primarily focus on causal inference, which is more correlated to machine learning interpretability.

2.2.1 Causal Inference

Causal inference has been widely applied in econometric, social science and medicine fields for evaluating the policy's effect or the drugs' side effect. Effect estimation is tied to the outcome caused by the treatment applied to an instance. An instance is the atomic research object, which can be a physical object or an individual person. Treatment and outcome are terms that denote a decision made or action taken and its result, respectively. We first introduce the essential concepts for learning treatment effect followed by the causal models.

- Covariates X encompass the background variables or features of the instance and are integral to the understanding of treatment effects and outcomes.
- Treatment T refers to the action (manipulation or intervention) that applies to a instance.
- Outcome Y is the result of the treatment applied on a instance.
- Confounder Z is a variable which causally affects both treatment and outcome.

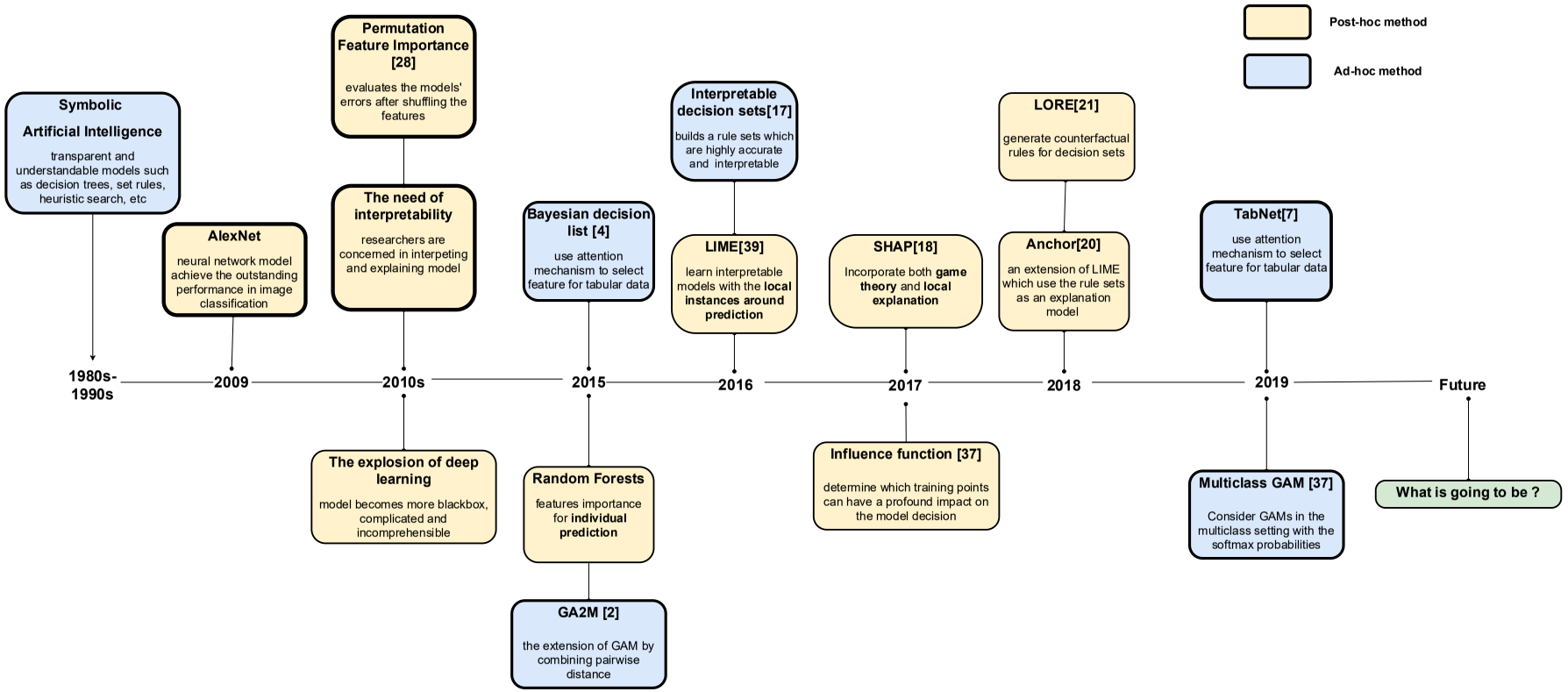


Figure 2.1 : The evolution of interpretable machine learning

To better understand causal inference, we give the following example combined with the notations defined above. To prove the efficiency of the medication on the disease, the scientist needs to assess its positive effect into the patients' recovery rate. Figure 2.2 depicts the corresponding causal relationships among the essential variables. The treatment T is whether the drugs are applied or not, and the observed features X are the patients' condition such as the level of insulin and cholesterol, heart rate, etc. Outcome Y is the recovery rate and age is the confounder Z. This is simply because age firstly determined the need of applying medication into patients, since the young people may not necessarily take the medicine. Age also affects to the recovery rate: the youth has a higher probability to recover than the elderly.

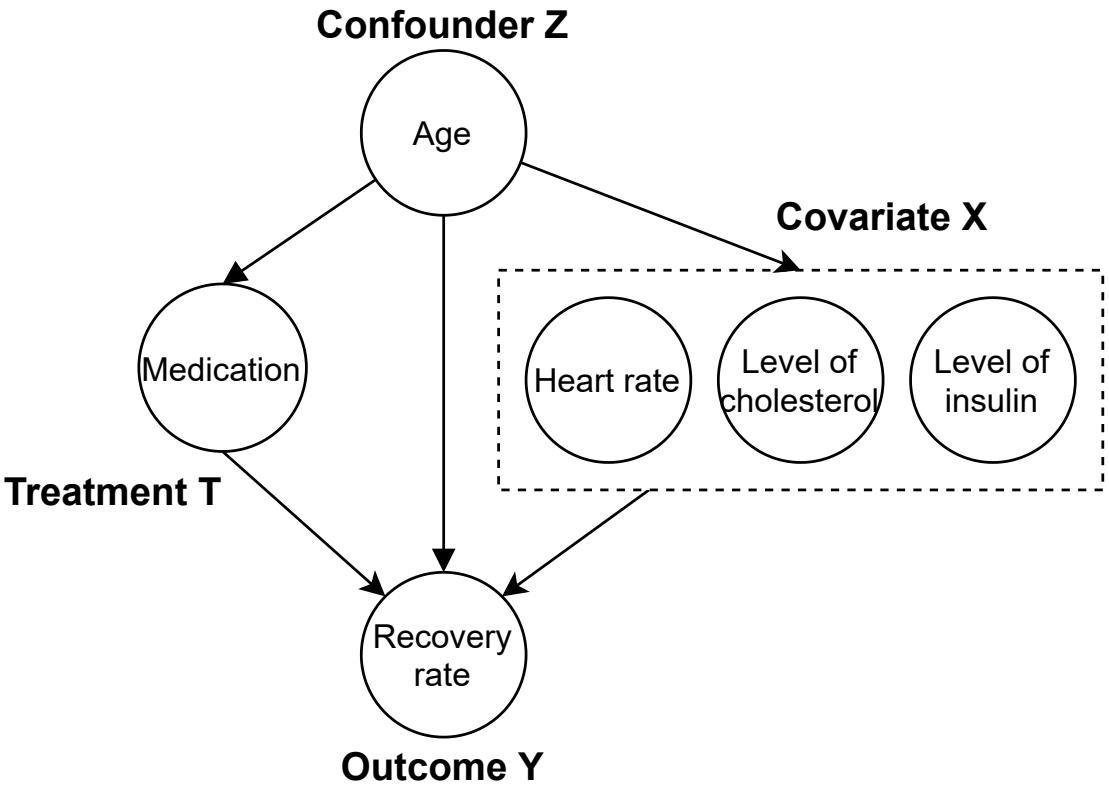


Figure 2.2 : The causal graph for recovery rate problem

2.2.2 Causal Models

We now introduce the two most important formal frameworks used for causal inference, namely the structural causal models and the potential outcome framework.

Structural causal model (Pearl 2009a) consists of two main components: the causal graph and structural equations. Causal graph is the probabilistic graphical model which is used to represent the assumption about prior knowledge and data generating process. A causal graph is defined as $\mathcal{G} = \langle \mathcal{V}, \mathcal{E} \rangle$ where $\mathcal{V}$ is the set of nodes and $\mathcal{E}$ is the set of edges. Structural equation is a set of equations Eq. (2.1) which are used to represent the causal effect illustrated by the edge in the causal graph.

$$\begin{aligned} X &= f_X(E_X), \\ T &= f_T(X, E_T). \\ Y &= f_Y(X, D, E_Y) \end{aligned} \tag{2.1}$$

where E_X, E_T, E_Y are exogenous variables, which are independent from other models' variable, and are determined outside the model.

Structural equation is defined as an order triple $\langle \mathcal{U}, \mathcal{V}, \mathcal{E} \rangle$ where:

- $\mathcal{U}$: is a set of exogenous variables which are independent from other models' variable, and is determined outside the model. Take for instance, the degree of temperature and moisture is exogenous to the causal model including the crop yield and farming.
- $\mathcal{V}$: is a set of endogenous variables which can be changed and determined by other variables within the model. A good case in point is the price in the supply and demand model, since this feature is dependent on both the supplier and consumer demand.
- $\mathcal{F}$: is a set of structural equations such that

  $$v_i = f_i(PA(v_i), u_i) \tag{2.2}$$

  with $v_i \in \mathcal{V}$, $u_i \in \mathcal{U}$ and $PA(v_i)$ illustrating the set of endogenous parent of v_i

Potential outcome framework is proposed by Neyman and Rubin (Rubin 1974). Considering binary treatments for a set of units, there are two possible outcomes for each unit. The unit will be assigned to the control treatment if $T = 0$, or

to the treated treatment if $T = 1$. As a result, we denote two potential outcomes Y_0 and Y_1 as the results caused by $T = 0$ and $T = 1$, respectively. Importantly, only one potential outcome is observed corresponds to the assigned treatment T, and we call this as the observed (factual) outcome Y. The unobserved potential outcome refers to the counterfactual outcome. Given the treatment T_i, the relationship between the observed outcome Y and two potential outcomes are

$$Y_i = T_i Y_1 + (1 - T_i) Y_0 \tag{2.3}$$

2.2.3 Treatment Effect Metric

With the key concepts and causal models, the treatment effect can be measured at the population, treated group, subgroup, and individual level. For simplicity, we discuss the treatment effect under the binary treatment, and it can be easily extended to multiple treatments by considering multiple potential outcomes.

The individual treatment effect (ITE) is defined as the change of Y_0 and Y_1, while keeping the covariates X unchanged (i.e., condition on those covariates). For an instance i with covariates X_i, its corresponding ITE is

$$ITE(\boldsymbol{X}_i) = E[Y_1|X_i] - E[Y_0|X_i] \tag{2.4}$$

As only one potential outcome is observed, it is nearly impossible to estimate the effect at the individual level. A more feasible way is to measure treatment effect at the average level.

The average treatment effect (ATE) measures the treatment effect at the whole population level as

$$ATE = E[Y_1 - Y_0] \tag{2.5}$$

The average treatment effect (ATT) is for the group of instances with the treatment equal to 1, i.e., the treated group.

$$ATT = E[Y_1 - Y_0|T = 1] \tag{2.6}$$

Conditional average treatment effect (CATE) known as heterogeneous treatment effect is defined on the subgroup with the particular covariate $X = x$.

$$CATE = E[Y_1 - Y_0|X = x] \tag{2.7}$$

2.2.4 Tools for Causal Analysis

Several libraries or tools are available for causal inference. Examples including *Double Machine Learning* (Chernozhukov et al. 2016), *Meta-learners* (Künzel et al.), *Orthogonal Learning* (Oprescu et al. 2018; Foster and Syrgkanis 2019) have been supported by EconML, CausalML, DoWhy and CausalNex, whereas causal discovery methods including graph inference and pairwise inference are provided in Causal Discovery Toolbox. Meanwhile, TIGRAMITE is a novel framework for causal discovery in time series.

2.3 Interpretable machine learning with causality

Pearl (Pearl and Mackenzie 2018) argues that causal reasoning is indispensable for machine learning to reach the human-level artificial intelligence, since it is the basic mechanism of human to be aware of the world. As a result, causal methodology is gradually becoming a vitally important component in explainable and interpretable machine learning. However, most of current interpretability techniques pay attention to solving the correlation statistic rather than the causation. Therefore, the causal approaches should be emphasized to achieve a higher degree of interpretability.

2.3.1 Model-Agnostic Causality for Deep Neural Neworks

The traditional way to analyze Deep Neural Network is to build several models with different architectures and make a comparison between their performances. The problem is that re-training DNNs is computationally expensive, and infeasible when it comes to the complicated architecture. Inspired by causal model, several methods have been proposed to interpret neural network model.

Chattopadhyay et al. (Chattopadhyay et al. 2019) define $\text{ACE}^{y}_{do(x_i=\alpha)}$ as the causal attribution of neuron x_i to the output neuron y_i, and $\mathbb{E}[y|do(x_i = \alpha)]$ as

the interventional expectation Eq. (2.8). The polynomial function is selected to estimate this value.

$$\mathbb{E}[y|do(x_i = \alpha)] = \int yp(y|do(x_i = \alpha))dy \tag{2.8}$$

Narendra et al. (Narendra et al. 2018) propose to construct a modified structural causal model as an abstraction of a DNN to make an reasoning over its elements. Thereafter, they rank each component based on their contribution to the final prediction for evaluation.

Based on TCAV (Kim et al. 2017) which generates a high-level concept-based explanation such as gender, race, background, others, the study in (Goyal et al. 2020) evaluates the *causal concept effect* on a neural network prediction. They overcome the problem of do-operator by using Variational AutoEncoder (VAE).

Regarding Generative Adversarial Networks (GANs) interpretability, Bau et al. (Bau et al. 2018) proposes an approach for visualization and understanding at unit-, object-, and scene- level by estimating the causal effect of the models' interpretable components. There are two main steps in their approach: dissection and intervention. In the dissection step, the classes with the explicit representation are firstly identified. Thereafter, they make an intervention by forcing the units to be appeared and disappeared, and calculate its causal effect. Meanwhile, the authors (Besserve et al. 2020) propose a causal framework to explore the intervention effect for proving that the components in images generated by GAN can be modified independently.

In terms of reinforcement learning, action influence model (Madumal et al. 2019) is introduced for explaining the behavior of RL agents. They construct a modified structural causal model, learn the causal equation as the regression model during training the agent, and finally generate the contrastive explanation to answer the counterfactual question "Why does the agent choose action A instead of action B?".

2.3.2 Post-hoc Interpretability

Model-Agnostic explanations are particular challenging when the models' parameters have more complex relationships. To further aid the intepretability, the practitioners propose a variety of post-hoc interpretability methods to exploit what a trained model has learned, without changing the underlying model. Most widely useful post-hoc interpretation methods fall into two main categories: causal feature learning and counterfactual explanations, respectively.

Casual Feature Learning

Recent work on feature learning derives the subset of features that have causal contributions to the models' prediction. Early causal feature learning is to find the Markov Blanket (MB) containing a set of features which makes the target (T) independent from other features given MB(T). In the study (Cawley 2008), the authors firstly use the HITON algorithm (Aliferis et al. 2003) to derive the Markov Blanket, and thereafter deploy Max-Min Hill-Climbing (MMHC) algorithm to identify the causes and effects of the target variable. Given the number of transfer learning tasks D, Peters et al. (Peters et al. 2015) assume that there exists a subset of features X_{S^*} such that the conditional distribution $Y_k|X_{S*}$ is the same for different tasks k, and other settings Eq. (2.9). They propose an algorithm called subset search which samples the subset features, and then adopt the Levene test to assess the assumption.

$$Y_k|X^k_{S*} \approx Y'_k|X^{k'}_{S*} \quad \forall k, k' \in 1, ..., D \tag{2.9}$$

CXPlain (Schwab and Karlen 2019a) is the causal framework that can explain more complex machine learning models by estimating the feature importance. Granger-causal objective is introduced to quantify how much the exclusion of a single feature reduces model performance. Particularly, CXPlain trains a separate explanation model to any predictor f by optimizing a Granger-causal objective. CXPlain can also estimate the uncertainty of features importance by calculating confidence interval (CI).

Counterfactual Explanation

Counterfactual explanation is the example-based model-agnostic method which generates new instances that would change the models' prediction. The prominent example (Grath et al. 2018) in this research is that one person x with the annual income a and the current balance b has been rejected a loan by the financial institution, so how she/he can change her/his income and balance to a' and b' in order to receive the loan. Given the set of points P, in order to generate the set of counterfactual samples F, the objective function of counterfactual explanation (Wachter et al. 2018) is to optimize the following function:

$$\begin{aligned} &\arg\min_{x}\max_{\lambda}(\lambda\cdot(\hat{f}(x')-y')^2+d(x,x')) \\ &d(x_i,x')=\sum_{k\in F}\frac{\mid x_k-x'_k\mid}{MAD_k} \\ &MAD_k=\underset{(j\in \mathrm{P})}{\mathrm{median}}(|X_{j,k}-\underset{(l\in \mathrm{P})}{\mathrm{median}}(X_{l,k})|) \end{aligned} \tag{2.10}$$

where x is an original instance, x' is the counterfactual instance which close to x, y' is the target class label for x', λ is the regularized parameter, $d(x,x')$ denotes the distance between the original instance and the counterfactual samples, MAD_k is the median absolute deviation for feature k.

Grath et al. (Grath et al. 2018) extend $d(x,x')$ in Eq. (2.10) by adding a weight vector Θ. The vector Θ is used to evaluate models' feature importance, and can be obtained by many algorithms such as K-Nearest Neighbors or global feature evaluation. Dhurandhar et al. (Dhurandhar et al. 2018) combine the loss function generated from Convolutional AutoEncoder, while Arnaud (Van Looveren and Klaise 2020) uses the prototypes function to ensure that the generated perturbation falls into the same distribution with the original data as well as increasing the computational speed without tuning too many parameters. Additionally, the counterfactual samples should be as diverse as possible; the study (Mothilal et al. 2020a) proposes to use determinant of kernel matrix to illustrate this property.

To empower the capability of counterfactual explanations, constraints are considered in optimization problem of counterfactual explanation. Take for example, a

person cannot decrease his age, or change his race and skin color. Recent work (Ustun et al. 2019; Russell 2019a) adopt Mixed Integer Programming (MIP) formulation to deal with categorical, numeric and mixed data type. Meanwhile, Artelt et al. (Artelt and Hammer 2020) propose convex density constraints to generate counterfactual located in a region of the data space. Specifically, the density constraint $\hat{p}_y \geq \delta$ denoted by a kernel density estimator or a Gaussian mixture model is added into the distance function $d(x, x')$.

CERTIFAI (Sharma et al. 2019) proposed by Sharma et al. as a novel and flexible approach which can be used in any type of data. CERTIFAI uses the customized genetic algorithm to choose individuals that have the best fitness scores defined as follows.

$$fitness = \frac{1}{d(x, x')}$$
$$d(x, x') = \begin{cases} \frac{n_{x'on}}{n} l_1(\mathbf{x}, \mathbf{x}') + \frac{n_{cat}}{n} simp(\mathbf{x}, \mathbf{x}') & \text{tabular data} \\ \frac{1}{SSIM(x,x')} & \text{image data} \end{cases} \tag{2.11}$$

For tabular data, CERTIFAI chooses l_1 norm for continuous features and a simple matching distance for categorical features (simp). For image data, Structural Similarity Index Measure (SSIM) (Zhou Wang et al. 2004) measures the similarity of what humans consider. n_{con} and n_{cat} are the number of continuous features and categorical features, respectively.

Instead of identifying the minimum changes leading to the desired outcome, a new line of counterfactual explanations provides feasible paths to transform a selected instance into one that meets a certain goal. FACE (Poyiadzi et al. 2020) proposed by Poyiadzi et al. constructs a graph over the data points with the weights illustrating the feasible degree to transit between two vertices. FACE thereafter can be solved by the *Disjstra* algorithm to find the shortest path from the original instance to the counterfactual one.

2.3.3 Visualization of Causal Effect

Visualization-based method is another commonplace approach for quick understanding what the models have learned. Partial dependence plot (PDP) (Goldstein et al. 2015) depicts the marginal effect of features into the predicted outcomes. The partial dependence function is defined as:

$$\hat{f}_{x_S}(x_S) = E_{x_C}\left[\hat{f}(x_S, x_C)\right] = \int \hat{f}(x_S, x_C)p(x_C)dx_C \tag{2.12}$$

Zhao et al.(Zhao and Hastie 2019) use Partial dependence plot (PDP) an its extension called Individual Conditional Expectation (ICE) to extract the causal information from machine learning model. These visualization tools allow to measure the predictions' change after making an intervention, which can help to discover the features' causal relationship.

2.4 Evaluation

Evaluation in causal interpretability is an extremely difficult task, at least in the current stage, since there are nearly no grouthtruth data to evaluate the methods' performance. Evaluation for traditional interpretable machine learning evaluation can be classified into three categories (Doshi-Velez and Kim 2017): application-based, human-based and function-based. We apply the same category and focus on evaluations that can be used in causal interpretability.

2.4.1 Application-based

In real-world scenario where the machine learning model is deployed to assist experts, application-based evaluation illustrates how well the models provide explanations to human experts for improving their performance in specific tasks. Take for example, a randomized experiment (Williams et al. 2016) is conducted among a group of learner to solve the problems. They then rate the explanation generated by the machine learning models. With the assistance of models, the performance of people in different tasks is proved to be improved.

2.4.2 Human-based

Human-based evaluation methods refer to evaluate the performance of interpretable models with the assistance of human. Madumal et al. (Madumal et al. 2019) generate explanation for the reinforcement learning. They implement an RL agent, and conduct an experiment running on StarCraft II, a strategic game, with 120 participants. *Explanation Satisfactory Scale* (Hoffman et al. 2018) is defined as the degree of human understanding of the AI system to measure the quality of generated explanations.

2.4.3 Function-based

Functional-based evaluation methods can be carried out without the assistance of human to evaluate the performance of the explanation model. There are some evaluation procedures for different techniques in Section IV:

Causal Interpretability for DNN

The lack of ground truth for feature effect makes it challenging to evaluate the performance of causal effect estimation. Chattopadhyay et al. (Chattopadhyay et al. 2019) compare the salient map (Kadir and Brady 2001) generated by causal attribution method with Integrated Gradient (Sundararajan et al. 2017). Harradon etc al. (Harradon et al. 2018) identify the components having the significant causal effects into the individual prediction. Specifically, they conduct the experiments in three different architectures VGG 19 in Birds200, VGG 16 and 6-layer cov network applied in Inria dataset. Thereafter, they make a query for individual input, and then visualize top k variables according to their causal effect.

Counterfactual Explanations

A previous research (Mothilal et al. 2020a) suggests that there are three main metrics to evaluate the counterfactual explanation: *proximity*, *diversity* and *sparsity*. The *proximity* is to reflect the similarity between the CF examples and the original one which was calculated as the mean proximity all over the examples. Meanwhile,

the *diversity* measures the mean of the distances between the pairs of samples, ensuring that the generated instances should be as diverse as possible. Finally, the *sparsity* is the average number of changes converting CF examples to the original one.

$$\begin{aligned} \text{proximity} &= \frac{-1}{k}\sum_{i=1}^{k} dist(x_{cf_i}, x) \\ \text{diversity} &= \frac{1}{C_2^k}\sum_{i=1}^{k-1}\sum_{j=i+1}^{k} dist(x_{cf_i}, x_{cf_j}) \\ \text{sparsity} &= 1 - \frac{1}{k \cdot d}\sum_{i=1}^{k}\sum_{l=1}^{d} 1[x_{cf_i}^l \not\equiv x_i^l] \end{aligned} \tag{2.13}$$

with x_{cf} and x are the counterfactual samples and original instance, respectively, $dist(x_{cf_i}, x_{cf_j})$ illustrates the distance between two generated counterfactual instances, d is the number of input features, k is the number of counterfactual samples to be generated.

2.5 Open questions and discussions

The need of explaining and interpreting models becomes highly critical along with the growing popularity of deep learning and automated machine learning. Although, there are currently several studies in this field, several open problems still remain unresolved.

1) Counterfactual explanation in classification tasks. There are a plethora of constraints, especially features' causal relationship, should be taken into consideration when adopting counterfactual explanation. Take for example, the counterfactual explanation cannot recommend the users to change sensitive and discriminative features such as race and gender in order to be accepted by the system. Therefore, its reasonability and feasibility should be discovered and investigated more strictly.

2) Counterfactual explanation in recommendation system and time series data. Although recommendation system gains the immense popularity these days, there are not many studies working on counterfactual explanation for such system. How we can make an intervention into human actions to enable the system to change

their recommended items still remains an open question. Meanwhile, regarding time series data, it is also interesting to discover that what the model would change its prediction if we change something in the past.

3) Causal reasoning in knowledge graph. Knowledge graph is recently utilized as an effective tool in several tasks such as recommendation system, knowledge extraction, classification, etc. Instead of embedding the knowledge graph as the latent features, Xian et al. (Xian et al. 2019) state that the true intelligent recommendation systems have to own the ability to recommend their items based on their causal reasoning.

4) Explanation understandable by non-experts. A number of recent methods frequently provide the explanations to experts and researchers rather than the end-users. Therefore, another challenge is to generate explanation under the form such as rules, natural language, images, etc which can allow nonprofessional people to catch up with machine learning model behaviors.

2.6 Conclusion

Interpretable machine learning is expected to become a mainstream topic in the foreseeable future. This paper provides the desiderata and brief overview of causal inference, followed by the causality based interpretable machine learning. We present two main causal approaches for interpretable machine learning including feature importance estimation, causal effects of model components, and counterfactual explanation. Finally, we has discussed several potentially unresolved problems in this field which open opportunities for researchers to work in.

In machine learning, the more data the better. However, in causal inference, the more data alone is not yet enough. Having more data only helps to get more precise estimates, but it cannot make sure these estimates are correct and unbiased. Machine learning methods enhance the development of causal inference, meanwhile, causal inference also helps machine learning methods. The simple pursuit of predictive accuracy is insufficient for modern machine learning research, and correctness and interpretability are also the targets of machine learning methods. Causal infer-

ence is starting to help to improve machine learning, such as recommender systems or reinforcement learning.

Part II

Causal inference

Part II investigates the realm of causal inference in machine learning. Chapter 3 starts by digging deep into causal inference, explaining the basic ideas of potential outcome frameworks. This chapter discovers the concepts of causality, counterfactuals, and identification within the domain of Structural Causal Models (SCMs). Additionally, we introduce a groundbreaking approach for estimating Average Treatment Effects (ATEs) by harnessing the power of stochastic propensity scores. This method stands poised to address the limitations observed in existing techniques applied to estimate ATEs. On the other hand, Chapter 4 builds upon the solid foundation laid in Chapter 3, extending the methodology to encompass reinforcement learning alongside stochastic propensity scores for policy optimization. This extension marks a significant stride towards a deeper understanding and application of causal inference principles in the field of machine learning.

Chapter 3

Stochastic Intervention for Causal Effect Estimation

Causal inference methods are widely applied in various decision-making domains such as precision medicine, optimal policy and economics. Central to these applications is the treatment effect estimation of intervention strategies. Current estimation methods are mostly restricted to the deterministic treatment, which however, is unable to address the stochastic space treatment policies. Moreover, previous methods can only make binary yes-or-no decisions based on the treatment effect, lacking the capability of providing fine-grained effect estimation degree to explain the process of decision making. In this chapter, we therefore advance the causal inference research to estimate stochastic intervention effect by devising a new stochastic propensity score and stochastic intervention effect estimator (SIE). Meanwhile, we design a customized genetic algorithm specific to stochastic intervention effect (Ge-SIO) with the aim of providing causal evidence for decision making. We provide the theoretical analysis and conduct an empirical study to justify that our proposed measures and algorithms can achieve a significant performance lift in comparison with state-of-the-art baselines. The majority of the content in this chapter is based on the following paper:

- **Duong, T. D.**, Li, Q., & Xu, G. (2021, July). Stochastic intervention for causal effect estimation. In 2021 International Joint Conference on Neural Networks *IJCNN* (pp. 1-8). IEEE (IJCNN 2021, CORE A).

3.1 Introduction

Causal inference increasingly plays a vitally important role in a wide range of fields including online marketing, precision medicine, political science, etc. For ex-

ample, a typical concern in precision medicine is whether an *alternative* medication treatment for a certain illness will lead to a better outcome *. Treatment effect estimation can answer this question by comparing outcomes under different treatments.

Estimating treatment effect is challenging, because only the factual outcome for a specific treatment assignment (say, treatment `A`) is observable, while the counterfactual outcome corresponding to alternative treatment `B` is usually unknown. Aiming at deriving the absent counterfactual outcomes, existing causal inference from observations methods can be categorized into these main branches: re-weighting methods (Gruber and Van der Laan 2011; Austin and Stuart 2015), tree-based methods (Chipman et al. 2007; Hill 2011; Wager and Athey 2018), matching methods (Rosenbaum and Rubin 1983; Dehejia and Wahba 2002; Stuart et al. 2011) and doubly robust learners (Research 2019; Dudík et al. 2011). In general, the matching approaches focus on finding comparable pairs based on distance metrics such as propensity score or Euclidean distance, while re-weighting methods assign each unit in the population a weight to equate groups based on the covariates. Meanwhile, tree-based machine learning models including decision tree or random forest are utilized in the tree-based approach to derive the counterfactual outcomes. Doubly Robust Learner is another recently developed approach that combines the propensity score weighting with the regression outcome to produce an unbiased and robust estimator.

Existing treatment effect estimation from observational data faces two major challenges. First, most of previous studies focus on the deterministic intervention which sets each individual a fixed treatment value, incapable of dealing with dynamic and stochastic intervention (Dudík et al. 2014; Pearl et al. 2000; Tian 2008). They can not address the question like "how the health status changes (the desired outcome) for the patient if the doctor adopts 50% dose reduction in the patient", which might be of practical interest in real world. Second, existing methods fail in exploiting the relationships between the desired response and the intervention on

* *Treatment* and *outcome* are terms in the theory of causal inference, which for example denote a promotion strategy taken and its resulting profit, respectively

the treatment, resulting in black-box effect estimation.

To address these issues, we propose a novel influence function based model to provide sufficient causal evidence to answer decision-making questions about stochastic interventions. Particularly, our model consists of three novel components: *stochastic propensity score*, *stochastic intervention effect estimator* (SIE) and *customized genetic algorithm* for stochastic intervention optimization (Ge-SIO). The main contributions of our model are summarized below:

- We propose a new stochastic propensity score learning the treatment effect trajectory, which tackles the limitation of existing approaches only dealing with deterministic intervention effects.
- Based on the general efficiency theory, we provide theoretical proof that SIE can achieve fast parametric convergence rates when the potential outcome model can not be perfectly estimated.
- Ge-SIO is proposed to find the optimal intervention leading to the desired response, which can be widely applicable in domain-specific decision-making.

3.2 Related Works

Conventionally, causal inference can be trickled by either the randomized experiment (also known as A/B testing in online settings) or observational data. In randomized experiment, units are randomly assigned to a treatment and their responses are recorded. One treatment is selected as the best among the alternatives by comparing the predefined statistical criteria. While randomized experiments have been popular in traditional causal inference, it is prohibitively expensive (Chan et al. 2010; Kohavi and Longbotham 2011) and infeasible (Bottou et al. 2013) in some real-world settings (Li et al. 2017; Wang et al. 2019; Xu et al. 2020). As an alternative method, observational study is becoming increasingly critical and available in many domains such as medicine, public policy and advertising. However, observational study needs to deal with data absence problem, which differs fundamentally from supervised learning (). This is simply because only the factual outcome (symptom)

for a specific treatment assignment (say, treatment `A`) is observable, while the counterfactual outcome corresponding to alternative treatment `B` in the same situation is always unknown.

3.2.1 Treatment Effect Estimation

The simplest way to estimate treatment effect in observational data is the matching method that finds the comparable units in the treated and controlled groups. The prominent matching methods include Propensity Score Matching (PSM) (Rosenbaum and Rubin 1983; Dehejia and Wahba 2002) and Nearest Neightbor Matching (NNM) (Stuart et al. 2011). Particularly, for each treated individual, PSM and NNM select the nearest units in the controlled group based on some distance functions, and then calculate the difference between two paired outcomes. Another popular approach is reweighting method that involves in building a classifier model to estimate the probability of a treatment assigned to a particular unit, and uses the predicted score as the weight for each unit in dataset. TMLE (Gruber and Van der Laan 2011) and IPSW (Austin and Stuart 2015) fall into this category. Ordinary Linear Regression (OLS) (Goldberger et al. 1964) is another commonplace method that fits two linear regression models for the treated and controlled group, with each treatment as the input features, and the outcome as the output. The predicted counterfactual outcomes thereafter are used to calculate the treatment effect. Meanwhile, decision tree is a popular non-parametric machine learning model, attempting to build the decision rules for the regression and classification tasks. Bayesian Additive Regression Trees (BART) (Chipman et al. 2007; Hill 2011) and Causal Forest (Wager and Athey 2018) are the prominent tree-based method in causal inference. While BART (Chipman et al. 2007; Hill 2011) builds the decision tree for the treated and controlled units, Causal Forest (Wager and Athey 2018) constructs the Random Forest model to derive the counterfactual outcomes, and then calculates the difference between the paired potential outcomes to obtain the average treatment effect. They are proven to obtain the more accurate treatment effect than matching methods and reweighting methods in the non-linear outcome setting.

Doubly Robust Learner (Research 2019; Dudík et al. 2011) is the recently pro-

posed approach that constructs a regression estimator predicting the outcome based on the covariates and treatment, and builds a classifier model to fit the treatment. DRL finally combines the both predicted propensity score and predicted outcome to estimate treatment effect.

3.2.2 Stochastic Intervention Optimization

Our work connects to the uplift modelling which optimizes the treatment effect by uplifting the expected response under the treatment policy (Zaniewicz and Jaroszewicz 2013; Alemi et al. 2009; Hansotia and Rukstales 2002; Manahan 2005). Uplift modelling measures the effectiveness of a treatment and then predicts the corresponding expected response. The most popular and widely-used approach is Separate Model Approach (SMA) (Zaniewicz and Jaroszewicz 2013; Alemi et al. 2009) which builds two different regression models. The first one uses treated unit data, whilst another works the controlled unit data. Several state-of-the-art machine learning models such as Random Forest, Gradient Boosting Regression or Adaboost can be used to construct the predictive model (Liaw et al. 2002; Solomatine and Shrestha 2004; Friedman 2001). The predicted responses are then calculated, and the optimal treatments are selected as the result. SMA has been widely applied in marketing (Hansotia and Rukstales 2002) and customer segmentation (Manahan 2005). However, when dealing with the data containing a great deal of noisy and missing information, the model outcomes are prone to be incorrect and biased, which leads to the poor performance. Other commonplace methods include Class Transformation Model (Jaskowski and Jaroszewicz 2012) and Uplift Random Forest (Guelman and Guelman 2014) that build the classification model for each outcome in the dataset. These techniques therefore can only handle the categorical outcomes, instead the continuous ones.

3.3 Preliminaries and Problem Definition

3.3.1 Notation

In this study, we consider the observational dataset $Z = \{\boldsymbol{x}_i, y_i, t_i\}_{i=1}^n$ with n units, where $\boldsymbol{x} \in \mathbb{R}^{n \times d}$ is the d-dimensional covariate, y and $t \in \{0, 1\}$ are the outcome and the treatment for the unit, respectively. The treatment variable is binary in many cases, thus the unit will be assigned to the control treatment if $t = 0$, or the treated treatment if $t = 1$. Accordingly, $y_0(\boldsymbol{x})$ and $y_1(\boldsymbol{x})$ are profit accrued from customer i corresponding to either the controlled or treated group. The central goal of causal inference is to compare the potential outcomes of the same units under two or more treatment conditions, which is implemented by computing the average treatment effect (ATE), i.e.,

$$\tau_{\text{ATE}} = \mathbb{E}[y_0(\boldsymbol{x}) - y_1(\boldsymbol{x})] \tag{3.1}$$

3.3.2 Propensity Score

Rosenbaum and Rubin

In practice, one widely-adopted parametric model for propensity score $p_t(\boldsymbol{x})$ is the logistic regression

$$\hat{p}_t(\boldsymbol{x}) = \frac{1}{1 + \exp\left(\boldsymbol{w}^\top \boldsymbol{x} + \omega_0\right)} \tag{3.2}$$

where $\boldsymbol{w}$ and ω_0 are estimated by minimizing the negative log-likelihood

3.3.3 Assumption

Following the general practice in causal inference literature, the following two assumptions should be taken into consideration to ensure the identifiability of the treatment effect, i.e. *Positivity* and *Ignorability.*

Assumption 3.1 (Positivity)**.** *. Each unit has a positive probability to be assigned by a treatment, i.e.,*

$$p_t(\boldsymbol{x}) > 0, \quad \forall \boldsymbol{x} \text{ and } t \tag{3.3}$$

Assumption 3.2 (Ignorability). *The assignment to the treatment t is independent of the outcomes $\boldsymbol{y}$ given covariates $\boldsymbol{x}$*

$$y_1, y_0 t | \boldsymbol{x} \tag{3.4}$$

3.4 Stochastic Intervention Effect

The stochastic intervention effect can be expressed by the difference between the observed outcome and the counterfactual outcome under the stochastic intervention. Because the observed outcome is fixed, stochastic intervention effect estimation is transformed as the problem of estimating the counterfactual outcome.

3.4.1 Stochastic Counterfactual Outcome

To estimate the counterfactual outcome, we first propose a flexible and task-specific stochastic propensity score to characterize the stochastic intervention.

Definition 3.1 (Stochastic Propensity Score). *The stochastic propensity score with respect to stochastic degree δ is*

$$q_t(\boldsymbol{x}, \delta) = \frac{\delta \cdot \hat{p}_t(\boldsymbol{x})}{\delta \cdot \hat{p}_t(\boldsymbol{x}) + 1 - \hat{p}_t(\boldsymbol{x})} \tag{3.5}$$

where $\hat{p}_t(\boldsymbol{x})$ is denoted by

$$\hat{p}_t(\boldsymbol{x}) = \frac{\exp\left(\sum_{j=1}^{s} \beta_j g_j(\boldsymbol{x})\right)}{1 + \exp\left(\sum_{j=1}^{s} \beta_j g_j(\boldsymbol{x})\right)} \tag{3.6}$$

where $\{g_1, \cdots, g_s\}$ are nonlinear basis functions.

The proposed stochastic propensity score in Definition 4.1 has two promising properties compared with (4.3). On the one hand, propensity score (4.3) fails to quantify the causal effect under stochastic intervention. So we introduce δ in (4.6) to represent the stochastic intervention indicating the extent to which the propensity scores are fluctuated from their actual observational values. For instance, the stochastic intervention that the doctor adopts 50% dose increase in the patient can be expressed by $\delta = 1.5$.

On the other hand, the linear term $\boldsymbol{w}^\top \boldsymbol{x} + \omega_0$ in Eq. (4.3) may lead to misspecification

On the basis of the stochastic propensity score, we propose an influence function specific to estimate counterfactual outcome under stochastic intervention. Meanwhile, we also analyze the asymptotic behavior of the counterfactual outcome with theoretical guarantees. We prove that our influence function can achieve double robustness and fast parametric convergence rates.

Theorem 3.1. *With the stochastic intervention of degree δ on observed data $z = (\boldsymbol{x}, y, t)$, we have*

$$\varphi(z, \delta) = q_t(\boldsymbol{x}, \delta) \cdot m_1(\boldsymbol{x}, y) + (1 - q_t(\boldsymbol{x}, \delta)) \cdot m_0(\boldsymbol{x}, y) \tag{3.7}$$

being the efficient influence function for the resulting counterfactual outcome $\hat{\psi}$, i.e.,

$$\hat{\psi} = \mathbb{P}_n \left[\varphi(z, \delta) \right] \tag{3.8}$$

where $m_1(\boldsymbol{x}, y)$ or $m_0(\boldsymbol{x}, y)$ is given by

$$m_t(\boldsymbol{x}, y) = \frac{\mathbb{I}_t \cdot (y - \hat{\mu}(\boldsymbol{x}, t))}{t \cdot \hat{p}_t(\boldsymbol{x}) + (1 - t)(1 - \hat{p}_t(\boldsymbol{x}))} + \hat{\mu}(\boldsymbol{x}, t) \tag{3.9}$$

and $\mathbb{I}_t$ is an indicator function, $\hat{p}_t$ is the estimated propensity score in Eq. (4.7) and $\hat{\mu}$ is potential outcomes model that can be fitted by machine learning methods.

Proof. Throughout we assume the observed data quantity ψ can be estimated under the positivity assumption from Section 4.3.3. For the unknown ground-truth $\psi(\delta)$, we will prove φ is the influence function of $\psi(\delta)$ in Eq. (4.9) by checking

$$\int \hat{\psi}(y, x, t, \mathbb{P}) d\mathbb{P} = \int \left(\varphi(y, x, t, \delta) - \psi \right) d\mathbb{P} = 0 \tag{3.10}$$

Eq. (4.11) indicates that the uncentered influence function φ is unbiased for ψ. Given $q_t(\boldsymbol{x}, \delta)$ as the stochastic propensity score in Eq. (4.6), we check the property (4.11)

by

$$
\begin{aligned}
&\int (\varphi(y,x,t,\delta)-\psi)\, d\mathbb{P} \\
&= \int \{q_t \cdot m_1(\boldsymbol{x},y) + (1-q_t) m_0(\boldsymbol{x},y) - \psi(\delta)\}\, d\mathbb{P}(y,x,t,\delta) \\
&= \int \{q_t \frac{\mathbb{I}_{t=1}\cdot(y-\hat{\mu}(x,1))}{\hat{p}_t} + (1-q_t)\frac{\mathbb{I}_{t=0}\cdot(y-\hat{\mu}(x,0))}{1-\hat{p}_t} \\
&+ q_t\hat{\mu}(x,1) + (1-q_t)\hat{\mu}(x,0) - \psi(\delta)\} d\mathbb{P}(y,x,t,\delta) \\
&= \int \{q_t \frac{\mathbb{I}_{t=1}\cdot(y-\hat{\mu}(x,1))}{\hat{p}_t} + (1-q_t)\frac{\mathbb{I}_{t=0}\cdot(y-\hat{\mu}(x,0))}{1-\hat{p}_t} \\
&+ q_t\hat{\mu}(x,1) + (1-q_t)\hat{\mu}(x,0) - \mathbb{E}[q_t\hat{\mu}(x,1) \\
&+ (1-q_t)\hat{\mu}(x,0)]\} d\mathbb{P}(y,x,t,\delta) \\
&\overset{(1)}{=} \int \left\{ q_t \frac{\mathbb{I}_{t=1}\cdot(y-\hat{\mu}(x,1))}{\hat{p}_t} \right\} d\mathbb{P}(y,x,t,\delta) \\
&+ \int \left\{ (1-q_t)\frac{\mathbb{I}_{t=0}\cdot(y-\hat{\mu}(x,0))}{1-\hat{p}_t} \right\} d\mathbb{P}(y,x,t,\delta) \\
&= \int \left\{ q_t \frac{\mathbb{I}_{t=1}\cdot y}{\hat{p}_t} + (1-q_t)\frac{\mathbb{I}_{t=0}\cdot y}{1-\hat{p}_t} \right\} d\mathbb{P}(y,x,t,\delta) \\
&- \int \left\{ q_t \frac{\mathbb{I}_{t=1}\cdot \hat{\mu}(x,1)}{\hat{p}_t} - (1-q_t)\frac{\mathbb{I}_{t=0}\cdot \hat{\hat{\mu}}(x,0)}{1-\hat{p}_t} \right\} d\mathbb{P}(x,t,\delta) \\
&\overset{(2)}{=} 0
\end{aligned}
$$

The second equation (1) follows from the iterated expectation, and the second equation (2) follows from the definition of $\hat{\mu}(\boldsymbol{x},t)$ and the usual properties of conditional distribution $d\mathbb{P}(x,y,\delta) = d\mathbb{P}(y|x,\delta)d\mathbb{P}(x,\delta)$. So far we have proved that φ is the influence function of average treatment effect $\psi(\delta)$. We have proved that the uncentered efficient influence function can be used to construct unbiased semiparametric estimator for $\psi(\delta)$, i.e., that $\int \varphi\mathbb{P} = \psi$. □

3.5 Stochastic Intervention Optimization

Estimating the stochastic intervention effect is not enough, we are more interested in "what is the optimal level/degree of treatment for a patient to achieve the most expected outcome?". In this section, we apply influence-based estimator to search for the optimal intervention that achieves the optimal expected response over the whole population. We model the stochastic intervention using the

Algorithm 3.1: SIE: Stochastic Intervention Effect

Input: Observed units $\{z_i : (\boldsymbol{x}_i, t_i, y_i)\}_{i=1}^n$

1: Initialize a stochastic degree δ.
2: Randomly split Z into k disjoint groups
3: **while** each group **do**
4: Fit the propensity score $\hat{p}_t(\boldsymbol{x}_i)$ by Eq. (4.7)
5: Fit the potential outcome model $\hat{\mu}(\boldsymbol{x}_i, t_i)$
6: Compute $\tau_i = \hat{p}_t(\boldsymbol{x}_i)\hat{\mu}(\boldsymbol{x}_i, 1) + (1 - \hat{p}_t(\boldsymbol{x}_i))\,\hat{\mu}(\boldsymbol{x}_i, 0)$
7: Calculate $q_t(\boldsymbol{x}_i; \delta)$ by Eq. (4.6)
8: Calculate $m_1(\boldsymbol{x}_i)$ and $m_0(\boldsymbol{x}_i)$ by Eq. (4.10)
9: Calculate the influence function $\varphi(z_i, \delta)$ by Eq. (4.9).
10: **end while**
11: Compute $\hat{\tau}_{\text{ATE}} = \frac{1}{n}\sum_{i=1}^n \tau_i$
12: Compute $\hat{\tau}_{\text{SIE}} = \frac{1}{n}\sum_{i=1}^n (\varphi(z_i, \delta) - y_i)$

Output: stochastic intervention effect τ_{SIE}, ATE τ_{ATE}

stochastic propensity score $\hat{q}_t(\boldsymbol{x}, \delta)$, and look for a set of stochastic interventions $\Delta = \{\delta_1^*, \cdots, \delta_n^*\}$ where the i-th intervention $\delta_i^* \in \Delta$ maximizes the expected response specific to i-th unit $z_i = (\boldsymbol{x}_i, y_i, t_i)$, denoted by $\varphi(z_i, \delta_i)$:

$$\delta_i^* = \arg\max_{\delta_i} \varphi(z_i, \delta_i) \tag{3.11}$$

Note that the optimization problem in Eq. (4.18) is non-differentiable. To avoid using further assumptions for solving it, we formulate a customised genetic algorithm

For stochastic intervention optimization, each candidate solution is described by the n-dimensional intervention Δ (the "genes") and the objective values of the candidates are evaluated by Eq (4.18). Usually, a random population of solutions is initialized, which undergoes through the process of evolution to obtain the better fitness function until the stopping condition is reached. Specifically, Ge-SIO first selects m solutions as the population of parents based on their fitness values. Among

the selected parent solutions, m solutions are chosen pairwise with the uniform distribution to produce children, which is called crossover process. The n-dimensional Δ are recombined by the simulated binary crossover recombinator. Crossover takes m selected parents and combines them, for the sake of diversity to the solutions. The children, which constitute solutions, are modified by the mutation operator. Mutation has a small chance to change Δ, which may create more fitter solutions. Thus, the Ge-SIO first generates children by crossover and modifies them by mutation thereafter. After the process of evolution is done, the fittest Δ is returned as the optimal solution to the desired expected response $\hat{\psi}$. We run it with the number of generations to repeat the above process so as to find the optimal solution. The full stochastic intervention optimization algorithm is shown in Algorithm 4.2.

Algorithm 3.2: Ge-SIO: Stochastic Intervention Optimization

Input: Observed units $\{z_i : (\boldsymbol{x}_i, t_i, y_i)\}_{i=1}^n$

Initialize a batch of population $\Gamma = \{\boldsymbol{\Delta}_1, \cdots, \boldsymbol{\Delta}_m\}$ with $\boldsymbol{\Delta}_i \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\nu})$
for G generation **do**
  for $k = 1, \cdots, m$ **do**
    for $i = 1, \cdots, n$ **do**
      Compute $q_t(\boldsymbol{x}, \delta_i)$ by Eq. (4.6)
      Calculate $m_1(\boldsymbol{x}_i)$ and $m_0(\boldsymbol{x}_i)$ by Eq. (4.10)
      Calculate $\varphi(z_i, \delta)$ by Eq. (4.8).
    end for
    Compute k-th fitness $\Phi(\Delta_k) = \sum_{i=1}^n \varphi(z_i, \delta)$
  end for
  Select $\Delta_1, \cdots, \Delta_m \in \Gamma$ based on its fitness function
  Randomly pair $\lceil m/2 \rceil$ $\{\Delta_1, \Delta_2\} \in \Gamma$
  for each pair $\{\Delta_1, \Delta_2\}$ **do**
    Perform uniform crossover$(\Delta_1, \Delta_2) \to \Delta_1', \Delta_2'$
    Perform uniform mutation $\Delta_1' \to \tilde{\Delta}_1, \Delta_2' \to \tilde{\Delta}_2$
    Update Γ by replacing $\{\Delta_1, \Delta_2\}$ with $\{\tilde{\Delta}_1, \tilde{\Delta}_2\}$
  end for
end for
Choose $\Delta^* = \arg\max_{\Delta \in \Gamma} \Phi(\boldsymbol{\Delta})$

Output: Δ^*

3.6 Experiments and Results

In this section, we conduct intensive experiments and compare our methods with state-of-the-art methods on two tasks: average treatment effect estimation and stochastic intervention effect optimization. Recall that the influence-based estimator φ depends on the nuisance function of propensity score p_t and outcome μ. We first perform average treatment effect estimation to confirm that $\hat{p}_t$ and $\hat{\mu}$ are unbiased and robust estimators. Moreover, the stochastic intervention optimization task is

carried out to demonstrate the effectiveness of our Ge-SIO, as well as investigate the impact of stochastic parameter δ on the expected response.

3.6.1 Baselines

We briefly describe the comparison methods which are used in two tasks of treatment effect estimation and stochastic intervention optimization.

Treatment effect estimation

We can not able to directly evaluate SIE on the estimation of stochastic intervention effect, because no dataset with ground-truth stochastic counterfactual outcome is available. On the contrary, the benchmark datasets having two potential outcomes are available for ATE estimation. Therefore, we perform ATE estimation to evaluate the robustness of $\hat{p}_t$ and $\hat{\mu}$ thus to indirectly evaluate the performance of SIE. We use Gradient Boosting Regression with 100 regressors for the potential outcome models $\hat{\mu}$. We compare our proposed estimator (SIE) with the following baselines including Doubly Robust Leaner

Stochastic Intervention Optimization

We compare our proposed method (Ge-SIO) with Separate Model Approach (SMA) with different settings. SMA (Zaniewicz and Jaroszewicz 2013; Alemi et al. 2009) aims to build two separate regression models for the outcome prediction in the treated and controlled group, respectively. Under the setting of SMA, we apply four well-known models for predicting outcome including Random Forest (SMA-RF) (Sołtys et al. 2015; Grimmer et al. 2017), Gradient Boosting Regressor (SMA-GBR) (Friedman 2001), Support Vector Regressor (SMA-SVR) (Zaniewicz and Jaroszewicz 2013), and AdaBoost (SMA-AB) (Solomatine and Shrestha 2004). We also compare the performance of these models with the random policy to justify that optimization algorithms can help to target the potential customers to generate greater revenue. For the settings of SMA, we use Gradient Boosting Regressor with 1000 regressors, AdaBoost Regression with 50 regressors, and Random Forest Tree Regressor with 100 trees.

3.6.2 Datasets

IHDP (Hill 2011) is a standard semi-synthetic dataset used in the *Infant Health and Development Program*, which is a popularly used semi-synthetic benchmark containing both the factual and counterfactual outcomes. We conduct the experiment on 100 simulations of IHDP dataset, in which each dataset is divided into training and testing set. The training dataset is highly imbalanced with 139 treated and 608 controlled units out of total 747 units, respectively, whilst the testing dataset has 75 units. Each unit has 25 covariates representing the individuals' characteristics. The outcomes are their IQ scores at age 3 (Dorie 2016).

Online promotion dataset (OP Dataset) provided by EconML project (Research 2019) is chosen to evalute stochastic intervention optimization [†]. This dataset consists of 10k records in online marketing scenario with the treatment of discount price and the outcome of revenue, each represents a customer with 11 covariates. We split the data into two part: 80% for training and 20% for testing set. We run 100 repeated experiments with different random states to ensure the model outcome reliability. With this dataset, we aim to investigate how different price policies applied to different customers will result in the best generated revenue. We directly model the revenue as the expected response for the uplift modelling algorithm.

3.6.3 Evaluation Metrics

In this section, we briefly describe the two evaluation metrics used for treatment effect estimation and optimization. Based on Eq. (4.1), we define the metric for evaluating the task of treatment effect estimation as the mean absolute error between the estimated and true ATE:

$$\epsilon_{ATE} = |\hat{\tau}_{\text{ATE}} - \tau_{\text{ATE}}| \tag{3.12}$$

Moreover, the main performance metric in the task is the expected value of the response under the proposed treatment, followed by the uplifting models study

[†] https://msalicedatapublic.blob.core.windows.net/datasets/Pricing/pricing_sample.csv

3.6.4 Results and Discussions

In this section, we aim to report the experimental results of 1) how our proposed estimator (SIE) can accurately estimate the average treatment effect; 2) how our optimization algorithm (Ge-SIO) can be used for finding optimal stochastic intervention in online promotion application; and 3) how the impacts of data size and stochastic degree are.

Treatment Effect Estimation

The results of ϵ_{ATE} derived from `IHDP` dataset with 100 simulations and `OP` dataset with 100 repeated experiments are presented in the Table 4.1 and Table 3.2, respectively. As seen clearly, amongst all approaches, our proposed method SIE achieves the best performance of the estimated ATE, while the Doubly Robust Learner performs next satisfactorily. Particularly, on `IHDP`, SIE outperforms all other methods in both training and testing set. In order to investigate the impact of data size chosen on estimation, we also run experiments and plot the performance of models in different data sizes in Figure **??**. Notably, SIE consistently produces the more accurate average treatment effect than others as the data size increases. Causal Forest and Doubly Robust Learner also produce the very competitive results, whereas the lowest performance belongs to IPWE. Turning to the experimental results on online promotion dataset in Table 3.2, SIE also has an outstanding performance consistently. Additionally, Doubly Robust Learner methods are ranked second, while the competitive results are recorded with BART. It is also worthy to note that although TMLE performs well in training set, its performance likely degrades when dealing with out-of-sample data in testing set. Overall, these results validate that our proposed SIE estimator proves to be effective and has an outstanding performance in the small and highly imbalanced dataset (`IHDP`) as well as in real-world application dataset (`OP`).

Stochastic Intervention Optimization

For the online promotion scenario, we model the revenue in dataset as the expected response of each customer under proposed treatment. Figure 4.2 presents

Table 3.1 : ϵ_{ATE} on 100 simulations of `IHDP` for training and testing (lower is better).

Method	`IHDP Dataset` ($\epsilon_{\text{ATE}} \pm$ `std`)	
	Train	Test
OLS	0.746 ± 0.140	1.264 ± 0.250
BART	1.087 ± 0.120	2.808 ± 0.100
Causal Forest	0.360 ± 0.050	0.883 ± 0.614
TMLE	0.326 ± 0.060	0.831 ± 1.750
ForestDRLearner	1.044 ± 0.040	1.224 ± 0.080
LinearDRLearner	0.691 ± 0.080	0.797 ± 0.170
IPWE	1.701 ± 0.140	5.897 ± 0.300
SIE	**0.284 ± 0.050**	**0.424 ± 0.090**

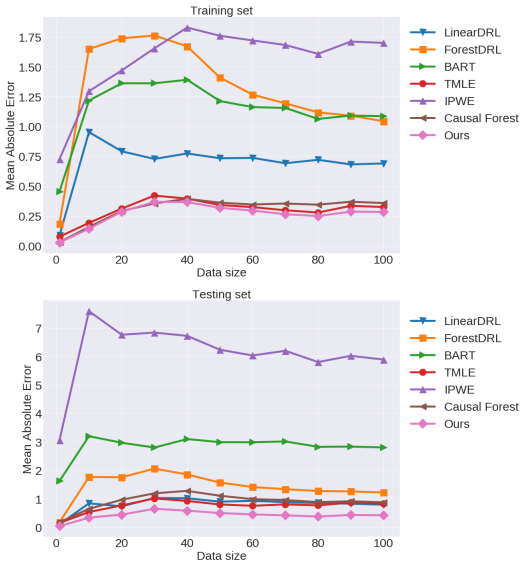


Figure 3.1 : ϵ_{ATE} on `IHDP` under different datasize

the revenue of uplifting modeling methods with different data sizes including 1000, 5000 and 10000 records. We set 100 generations for our Ge-SIO. Apparently, Ge-SIO generally produces the greatest revenue in all three datasizes, while SMA-ABR achieves the second-best performance with a very competitive result. Moreover, there is no significant difference in the performance of SMA with different settings. In contrast, the lowest revenue is generated by the random stochastic intervention

Table 3.2 : ϵ_{ATE} on OP dataset in 100 repeated experiments (lower is better).

Method	OP Dataset ($\epsilon_{\text{ATE}} \pm$ std)	
	Train	Test
OLS	5.906 ± 0.004	5.907 ± 0.000
BART	0.504 ± 0.042	0.505 ± 0.043
Causal Forest	3.520 ± 0.034	3.520 ± 0.034
TMLE	0.660 ± 0.000	3.273 ± 0.000
ForestDRLearner	0.240 ± 0.014	0.241 ± 0.013
LinearDRLearner	0.139 ± 0.009	0.139 ± 0.008
IPWE	5.908 ± 0.004	5.908 ± 0.015
SIE	**0.137 ± 0.000**	**0.119 ± 0.000**

that fails to choose the target customers to provide the promotion. The possible reason behind our proposed method's outstanding performance is that instead of getting the uplift signal like SMA, we directly intervene into the propensity score to produce the best stochastic intervention. From the business view, this emphasizes the crucial importance of the stochastic intervention optimization in online marketing campaign.

On the other hand, Figure 6.3 provides the information on the expected response with the various stochastic degree δ in OP and IHDP dataset with 90% confidence interval. More specifically, when increasing degree δ from 0 to 5, the expected revenue also increases accordingly. The revenue thereafter reaches the highest point and remains nearly stable when δ is greater than 5. Similarly, the expected IQ score per children in the IHDP dataset also witnesses the same trend: the IQ score climbs gradually as stochastic degree δ rises. The plot of the relationship between the expected response and stochastic degree δ provides valuable insights into the degree of intervention we should make to achieve the optimal stochastic intervention, which can greatly facilitate the decision-making process.

Figure 3.2 : Expected revenue per customer from `OP` dataset by different models

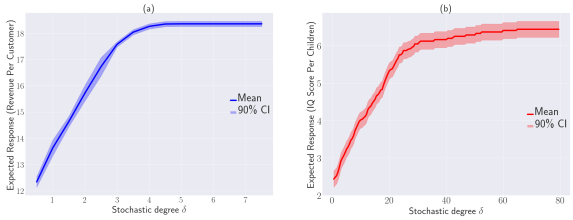


Figure 3.3 : (a) Expected revenue per customer from `OP` dataset with uniform 90% confidence. (b) Expected IQ score per children from `IHDP` dataset with uniform 90% confidence

3.7 Conclusion

Causal inference increasingly gains the attention from both academia and industry as a powerful tool to deal with the scenario where people are not only interested to know the treatment effect but also the optimal intervention for the expected responses (Wang et al. 2020; Yin et al. 2021). To extend causal inference to addressing stochastic interventions, this paper focuses on the dynamic intervention that is not discussed much in the recent study. In general, the contribution of this study is twofold. Based on stochastic propensity score, we propose a novel stochastic intervention effect estimator along with a customised genetic algorithm for stochastic intervention optimization. Our method can learn the trajectory of the stochastic intervention effect, providing causal insights for decision-making applications. Theoretical and numerical results justify that our methods outperform state-of-the-art baselines in both treatment effect estimation and stochastic intervention optimization.

Chapter 4

Stochastic Intervention for Causal Inference via Reinforcement Learning

This chapter builds upon the insights gathered in Chapter 3, demonstrating how the combination of stochastic propensity scores and reinforcement learning can effectively optimize policy decisions. We construct a customised reinforcement learning algorithm based on the random search solver which can effectively find the optimal policy to produce the greatest expected outcomes for the decision-making process. Finally, we conduct extensive empirical experiments to validate that our framework can achieve superior performance in comparison with state-of-the-art baselines. The content of the chapter is mainly from the paper:

- **Duong, T. D.**, Li, Q., & Xu, G. (2022). Stochastic intervention for causal inference via reinforcement learning. Neurocomputing, 482, 40-49 (Q1 Scorpus).

4.1 Introduction

Causal inference aims at estimating the causal effects of an intervention or treatment on an outcome, which increasingly plays a vitally important role in scientific investigations and real-world applications (Li et al. 2021a; Xu et al. 2020; Li et al. 2021b). A widely used example of the causal effect for binary treatment is that the expectation of the outcome in a hypothetical world in which everybody receives treatment is compared with their counterparts in a world where nobody does*. Other examples include "What is the effect of sleep deprivation on health outcomes?" and

* *Treatment* and *outcome* are terms in the theory of causal inference, which for example denote a promotion strategy taken and its resulting profit, respectively

"How would family socio-economic status affect career prospects?". Therefore, it is of great interest to develop models that can correctly predict the optimal treatment based on given subject characteristics. Treatment effect estimation can address this by comparing outcomes under different treatments.

Estimating treatment effect is challenging, because only the factual outcome for a specific treatment assignment (say, treatment `A`) is observable, while the counterfactual outcome corresponding to alternative treatment `B` is usually unknown. Aiming at deriving the absent counterfactual outcomes, existing causal inference from observations methods can be categorized into these main branches: re-weighting methods (Gruber and van der Laan 2012; Austin and Stuart 2015), tree-based methods (Chipman et al. 2007; Hill 2011; Wager and Athey 2018), matching methods (Rosenbaum and Rubin 1983; Dehejia and Wahba 2002; Stuart et al. 2011; Li et al. 2021d) and doubly robust learners (Research 2019; Dudík et al. 2011). In general, re-weighting methods assign each unit in the population a weight to balance groups based on the covariates, while tree-based models employ decision tree or random forest to estimate the counterfactual outcomes. Meanwhile, the matching approaches focus on finding the comparable pairs based on distance metrics such as propensity score (Rubin 1974), Euclidean distance (Yao et al. 2018) or Hilbert norm (Li et al. 2021c). Doubly Robust Learner is another recently developed approach that combines propensity score weighting with the regression outcome to produce an unbiased and robust estimator.

However, recent studies in treatment effect estimation mainly focus on the deterministic intervention which sets each individual a deterministic treatment, incapable of dealing with dynamic and stochastic intervention (Dudík et al. 2014; Pearl et al. 2000; Tian 2008). i.e., the treatment is deterministic. In many real-world applications, however, the effect of a stochastic intervention might be of interest. For example, rather than "if we do not use the medicine treatment `A` for all units, what is resulting in health status (the desired outcome)?", the medical researcher is more eager to know "how all units' health status change if we adopt 50% dose reduction in medicine treatment `A`". In this case, the treatment variable is no longer determin-

istic but a stochastic value, and traditional causal inference methods fail to capture the stochastic intervention on the treatment variable.

To address these issues, we propose a novel influence function based model to provide sufficient causal evidence to answer decision-making questions about stochastic interventions. Stochastic intervention estimation in our method can provide a fine-grained treatment effect estimation to gradually quantify the effect of the stochastic intervention on the outcomes. In addition, we exploit stochastic intervention optimization to customize stochastic intervention assignment, i.e., what is the best degree of intervention on the treatment to achieve the desired outcome. The main contributions of our work are summarized below:

- We propose a causal inference framework to learn the treatment effect under the stochastic intervention, which tackles the limitation of existing approaches only dealing with deterministic intervention effects. Particularly, our framework introduces the concept of stochastic propensity score, and develops a semi-parametric influence function to learn stochastic intervention effect.

- Based on the general efficiency theory, we theoretically analyze the asymptotic behavior of our semi-parametric influence function. We prove that our influence function can achieve double robustness and fast parametric convergence rates. We also empirically demonstrate the effectiveness of the proposed influence function.

- Based on the stochastic treatment effect estimation, our framework is capable of customizing the stochastic intervention, with the goal of uplifting desired outcomes on downstream decision-making applications. We formulate the stochastic intervention optimization as a derivative-free optimization problem and design a random search solver to efficiently achieve the optimal expected outcome.

4.2 Related works

Conventionally, causal inference can be conducted by either the randomized experiment (also known as A/B testing in online settings) or observational data (Duong et al. 2021b,a). In randomized experiment, units are randomly assigned to a treatment and their outcomes are recorded. One treatment is selected as the best among the alternatives by comparing the predefined statistical criteria. While randomized experiments have been popular in traditional causal inference, it is prohibitively expensive (Chan et al. 2010; Kohavi and Longbotham 2011; Xu et al. 2020) and infeasible (Bottou et al. 2013; Li et al. 2021b) in some real-world settings. As an alternative method, observational study is becoming increasingly critical and available in many domains such as medicine, public policy and advertising. However, observational study needs to deal with data absence problem, which differs fundamentally from supervised learning. This is simply because only the factual outcome (symptom) for a specific treatment assignment (say, treatment `A`) is observable, while the counterfactual outcome corresponding to alternative treatment `B` in the same situation is always unknown. In the context of binary treatment, the individuals given the treatment are the treated group, whereas other individuals in the population are the control group.

4.2.1 Treatment Effect Estimation

The simplest way to estimate treatment effect in observational data is the matching method that finds the comparable units in the treated and controlled groups. The prominent matching methods include Propensity Score Matching (PSM) (Rosenbaum and Rubin 1983; Dehejia and Wahba 2002) and Nearest Neightbor Matching (NNM) (Stuart et al. 2011). Particularly, for each treated individual, PSM and NNM select the nearest units in the controlled group based on some distance functions, and then calculate the difference between two paired outcomes. Another popular approach is reweighting method that involves building a classifier model to estimate the probability of a treatment assigned to a particular unit, and uses the predicted score as the weight for each unit in dataset. TMLE (Gruber and van der Laan 2012)

and IPSW (Austin and Stuart 2015) fall into this category. Ordinary Linear Regression (OLS) (Goldberger et al. 1964) is another commonplace method that fits two linear regression models for the treated and controlled group, with each treatment as the input features and the outcome as the output. The predicted counterfactual outcomes thereafter are used to calculate the treatment effect. Meanwhile, decision tree is a popular non-parametric machine learning model, attempting to build the decision rules for the regression and classification tasks. Bayesian Additive Regression Trees (BART) (Chipman et al. 2007; Hill 2011) and Causal Forest (Wager and Athey 2018) are the prominent tree-based method in causal inference. While BART (Chipman et al. 2007; Hill 2011) builds the decision tree for the treated and controlled units, Causal Forest (Wager and Athey 2018) constructs the Random Forest model to derive the counterfactual outcomes, and then calculates the difference between the paired potential outcomes to obtain the average treatment effect. They are proven to obtain the more accurate treatment effect than matching methods and reweighting methods in the non-linear outcome setting. Doubly Robust Learner (Research 2019; Dudík et al. 2011) is the recently proposed approach that constructs a regression estimator predicting the outcome based on the covariates and treatment, and builds a classifier model to fit the treatment. DRL finally combines both predicted propensity score and predicted outcome to estimate treatment effect.

4.2.2 Stochastic Intervention Optimization

Our work focuses on estimating the intervention effect and thus finding the optimal intervention to maximize the expected outcomes in the population. This is closely related to the uplift modelling studies, with the goal of uplifting (or maximizing) the outcome with the treatment as compared to the outcome without the treatment(Zaniewicz and Jaroszewicz 2013; Alemi et al. 2009; Hansotia and Rukstales 2002; Manahan 2005). Among the uplifting models, the most popular and widely-used approach is Separate Model Approach (SMA) (Zaniewicz and Jaroszewicz 2013; Alemi et al. 2009). SMA is applicable to binary treatment and builds two regression models under each treatment, respectively. The treatment with the best predictive

outcome is chosen and defined as the optimal one. The advantage of SMA lies in its easy implementation since SMA does not require a specific machine learning algorithm. Several state-of-the-art machine learning algorithms such as Random Forest, Gradient Boosting (Natekin and Knoll 2013) or Adaboost can be applied to these two regression models (Liaw et al. 2002; Solomatine and Shrestha 2004; Friedman 2001). SMA has been widely applied in marketing (Hansotia and Rukstales 2002) and customer segmentation (Manahan 2005). However, when dealing with the data containing a great deal of noisy and missing information, the model outcomes are prone to be incorrect and biased, which leads to poor performance. Other commonplace methods for uplift modelling include Class Transformation Model (Jaskowski and Jaroszewicz 2012) and Uplift Random Forest (Guelman and Guelman 2014); these techniques however only deal with the binary outcome, so we do not discuss them here.

4.3 Preliminaries and Problem Definition

4.3.1 Notation

In this study, we consider the observational dataset $Z = \{\boldsymbol{x}_i, y_i, t_i\}_{i=1}^{n}$ with n units, where $\boldsymbol{x} \in \mathbb{R}^{n \times d}$ is the d-dimensional covariate, y and $t \in \{0, 1\}$ are the outcome and the treatment for the unit, respectively. The treatment variable is binary in many cases, thus the unit will be assigned to the control treatment if $t = 0$, or the treated treatment if $t = 1$. As a result, y_0 and y_1 are the potential outcomes corresponding to the control and treated units. According to the Rubin-Neyman causal model (Imbens and Rubin 2015), two potential outcomes $y_0(\boldsymbol{x})$ and $y_1(\boldsymbol{x})$ exist for $\boldsymbol{x}$ with the treatment $t = 0$ and $t = 1$, respectively. It is noted that either y_0 or y_1 can be observed for each subject in the population.

After introducing the observational data and the key terminologies, the central goal of causal inference, i.e., treatment effect estimation, can be quantitatively definitions using the above definitions. To make the definition clear, here we define the treatment effect under binary treatment. At the population level, the treatment

effect is named as the Average Treatment Effect (ATE), which is defined as

$$\tau_{\text{ATE}} = \mathbb{E}[y_0(\boldsymbol{x}) - y_1(\boldsymbol{x})] \tag{4.1}$$

For causal inference, our objective is to estimate the treatment effects from the observational data. Based on the estimated ATE, different treatment conditions can be selected and applied to users to achieve preferred outcome.

To further illustrate the treatment effect estimation, we take an online marketing scenario for example. We denote each customer as a high-dimensional vector of features $\boldsymbol{x}$. The customer indexed by i receives a promotion treatment $t_i \in \{0, 1\}$. Accordingly, $y_0(\boldsymbol{x}_i)$ and $y_1(\boldsymbol{x}_i)$ are profit accrued from customer i corresponding to either control treatment or treated treatment. The effectiveness of a promotion campaign can be evaluated by computing average treatment effect of the promotion treatment on the customers.

4.3.2 Propensity Score

Rosenbaum and Rubin (Rosenbaum and Rubin 1983) first proposed propensity score technique to deal with the high-dimensional covariates. The propensity score is widely used in causal inference methods to estimate treatment effects from observational data (Hirano et al. 2003; Pirracchio et al. 2016; Luo et al. 2010; Abdia et al. 2017). This is largely because propensity score can help eliminating the great portion of bias, leading to a more balanced dataset and thus allowing a simple and direct comparison between the treated and untreated individuals. Particularly, propensity score can summarise the mechanism of treatment assignment and thus squeezes covariate space into one dimension to avoid the possible data sparseness issue (Bang and Robins 2005; Dehejia and Wahba 2002; Austin and Stuart 2015; Hirano et al. 2003). The propensity score is defined as the probability that a unit is assigned to a particular treatment $t = 1$ given the covariate $\boldsymbol{x}$, i.e.,

$$p_t(\boldsymbol{x}) = \mathbb{P}(t = 1|\boldsymbol{x}) \tag{4.2}$$

In practice, one widely-adopted parametric model for estimating propensity score

$p_t(\boldsymbol{x})$ is the logistic regression

$$\hat{p}_t(\boldsymbol{x}) = \frac{1}{1 + \exp\left(\boldsymbol{w}^\top \boldsymbol{x} + \omega_0\right)} \tag{4.3}$$

where $\boldsymbol{w}$ and ω_0 are parameters estimated by minimizing the negative log-likelihood (Martens et al. 2008).

4.3.3 Assumption

Following the general practice in causal inference literature (Pearl 2010, 2003; Scheines 1997), we consider the following two assumptions to ensure the identifiability of the treatment effect, i.e. *Positivity* and *Ignorability*.

Assumption 4.1 (Positivity)**.** *Each unit has a positive probability to be assigned by a treatment, i.e.,*

$$p_t(\boldsymbol{x}) > 0, \quad \forall \boldsymbol{x} \text{ and } t \tag{4.4}$$

Assumption 4.2 (Ignorability)**.** *The assignment to the treatment t is independent of the outcomes $\boldsymbol{y}$ given covariates $\boldsymbol{x}$*

$$y_1, y_0 t | \boldsymbol{x} \tag{4.5}$$

4.4 Stochastic Intervention Effect

Recall the goal of causal inference is to compute the treatment effect estimation that can be evaluated by the metric in Eq. (4.1). Namely, treatment effect estimation can be expressed by the difference between the observed outcome and the counterfactual outcome under a intervention on the treatment. Apparently, the observed outcome in the dataset is generated by the observed treatment (e.g., $t = 1$). By contrast, the counterfactual outcome is generated by intervening the treatment, e.g., shifting treatment from observed $t = 1$ to counterfactual $t = 0$, which is however unobserved in practice. Thus, intervention effect estimation is turned into a problem of predicting the counterfactual outcome generated by an intervention on the treatment.

4.4.1 Stochastic Counterfactual Outcome

Before predicting the counterfactual outcome, we first propose stochastic propensity score to characterize the stochastic intervention.

Definition 4.1 (Stochastic Propensity Score)**.** *The stochastic propensity score with respect to stochastic degree δ is*

$$q_t(\boldsymbol{x}, \delta) = \frac{\delta \cdot \hat{p}_t(\boldsymbol{x})}{\delta \cdot \hat{p}_t(\boldsymbol{x}) + 1 - \hat{p}_t(\boldsymbol{x})} \tag{4.6}$$

and $\hat{p}_t(\boldsymbol{x})$ is denoted by

$$\hat{p}_t(\boldsymbol{x}) = \frac{\exp\left(\sum_{j=1}^{s} \beta_j g_j\left(\boldsymbol{x}\right)\right)}{1 + \exp\left(\sum_{j=1}^{s} \beta_j g_j\left(\boldsymbol{x}\right)\right)} \tag{4.7}$$

where $\{g_1, \cdots, g_s\}$ are nonlinear basis functions.

The proposed stochastic propensity score in Definition 4.1 has promising properties compared with classical propensity score in Eq. (4.3). Particularly, traditional propensity score focuses on setting the treatment as fixed values rather than stochastic intervention, which is not desirable to quantify the causal effect of stochastic intervention on treatment. So we introduce δ in Eq. (4.6) to represent the stochastic intervention indicating the extent to which the propensity scores have fluctuated from their actual observational values. For instance, the stochastic intervention that the doctor adopts 50% dose increase in the patient can be expressed by $\delta = 1.5$. On the other hand, if there are higher-order terms or non-linear trends among covariates $\boldsymbol{x}$, classical propensity score using $\boldsymbol{w}^\top \boldsymbol{x} + \omega_0$ in Eq. (4.3) may lead to misspecification (Dalessandro et al. 2012). So we propose to use a sum of nonlinear function $\sum_{j=1}^{s} \beta_j g_j$ in Eq. (4.7) that captures the non-linearity involving covariates to create an unbiased estimator of treatment effect.

On the basis of the stochastic propensity score, we propose an influence function specific to estimate counterfactual outcome under stochastic intervention. Meanwhile, we also analyze the asymptotic behavior of the counterfactual outcome with theoretical guarantees. Theorem 4.1 shows that our influence function can construct an unbiased semiparametric estimator for the counterfactual outcome. In addition,

Theorem 4.2 guarantees that our influence function can achieve double robustness and fast parametric convergence rates. With such an unbiased and efficient counterfactual outcome estimator, we can achieve more accurate stochastic treatment effects.

Theorem 4.1. *With the stochastic intervention of degree δ on observed data $z = (\boldsymbol{x}, y, t)$, we have*

$$\varphi(z, \delta) = q_t(\boldsymbol{x}, \delta) \cdot m_1(\boldsymbol{x}, y) + (1 - q_t(\boldsymbol{x}, \delta)) \cdot m_0(\boldsymbol{x}, y) \tag{4.8}$$

being the efficient influence function for the resulting counterfactual outcome $\hat{\psi}$, i.e.,

$$\hat{\psi} = \mathbb{P}_n \left[\varphi(z, \delta)\right] \tag{4.9}$$

where $m_1(\boldsymbol{x}, y)$ or $m_1(\boldsymbol{x}, y)$ is given by

$$m_t(\boldsymbol{x}, y) = \frac{\mathbb{I}_t \cdot (y - \hat{\mu}(\boldsymbol{x}, t))}{t \cdot \hat{p}_t(\boldsymbol{x}) + (1 - t)(1 - \hat{p}_t(\boldsymbol{x}))} + \hat{\mu}(\boldsymbol{x}, t) \tag{4.10}$$

and $\mathbb{I}_t$ is an indicator function, $\hat{p}_t$ is the estimated propensity score in Eq. (4.7) *and $\hat{\mu}$ is potential outcomes model that can be fitted by machine learning methods.*

Proof. Throughout we assume the observed data quantity ψ can be estimated under the positivity assumption from Section 4.3.3. For the unknown ground-truth $\psi(\delta)$, we will prove φ is the influence function of $\psi(\delta)$ in Eq. (4.9) by checking

$$\int \hat{\psi}(y, x, t, \mathbb{P}) d\mathbb{P} = \int \left(\varphi(y, x, t, \delta) - \psi\right) d\mathbb{P} = 0 \tag{4.11}$$

Eq. (4.11) indicates that the uncentered influence function φ is unbiased for ψ. Given $q_t(\boldsymbol{x}, \delta)$ as the stochastic propensity score in Eq. (4.6), we check the property (4.11)

by

$$
\begin{aligned}
&\int (\varphi(y,x,t,\delta)-\psi)\, d\mathbb{P} \\
&= \int \{q_t \cdot m_1(\boldsymbol{x},y) + (1-q_t) m_0(\boldsymbol{x},y) - \psi(\delta)\}\, d\mathbb{P}(y,x,t,\delta) \\
&= \int \{q_t \frac{\mathbb{I}_{t=1}\cdot(y-\hat{\mu}(x,1))}{\hat{p}_t} + (1-q_t)\frac{\mathbb{I}_{t=0}\cdot(y-\hat{\mu}(x,0))}{1-\hat{p}_t} \\
&+ q_t\hat{\mu}(x,1) + (1-q_t)\hat{\mu}(x,0) - \psi(\delta)\} d\mathbb{P}(y,x,t,\delta) \\
&= \int \{q_t \frac{\mathbb{I}_{t=1}\cdot(y-\hat{\mu}(x,1))}{\hat{p}_t} + (1-q_t)\frac{\mathbb{I}_{t=0}\cdot(y-\hat{\mu}(x,0))}{1-\hat{p}_t} \\
&+ q_t\hat{\mu}(x,1) + (1-q_t)\hat{\mu}(x,0) - \mathbb{E}[q_t\hat{\mu}(x,1) \\
&+ (1-q_t)\hat{\mu}(x,0)]\} d\mathbb{P}(y,x,t,\delta) \\
&\overset{(1)}{=} \int \left\{ q_t \frac{\mathbb{I}_{t=1}\cdot(y-\hat{\mu}(x,1))}{\hat{p}_t} \right\} d\mathbb{P}(y,x,t,\delta) \\
&+ \int \left\{ (1-q_t)\frac{\mathbb{I}_{t=0}\cdot(y-\hat{\mu}(x,0))}{1-\hat{p}_t} \right\} d\mathbb{P}(y,x,t,\delta) \\
&= \int \left\{ q_t \frac{\mathbb{I}_{t=1}\cdot y}{\hat{p}_t} + (1-q_t)\frac{\mathbb{I}_{t=0}\cdot y}{1-\hat{p}_t} \right\} d\mathbb{P}(y,x,t,\delta) \\
&- \int \left\{ q_t \frac{\mathbb{I}_{t=1}\cdot \hat{\mu}(x,1)}{\hat{p}_t} - (1-q_t)\frac{\mathbb{I}_{t=0}\cdot \hat{\hat{\mu}}(x,0)}{1-\hat{p}_t} \right\} d\mathbb{P}(x,t,\delta) \\
&\overset{(2)}{=} 0
\end{aligned}
$$

The second equation (1) follows from the iterated expectation, and the second equation (2) follows from the definition of $\hat{\mu}(\boldsymbol{x},t)$ and the usual properties of conditional distribution $d\mathbb{P}(x,y,\delta) = d\mathbb{P}(y|x,\delta)d\mathbb{P}(x,\delta)$. So far we have proved that φ is the influence function of average treatment effect $\psi(\delta)$. We have proved that the uncentered efficient influence function can be used to construct unbiased semiparametric estimator for $\psi(\delta)$, i.e., that $\int \varphi\mathbb{P} = \psi$. □

4.4.2 Asymptotic Behavior Analysis

Theorem 4.1 ensures that the counterfactual outcome $\hat{\psi}$ can be estimated by its influence function φ that depends on the nuisance function $(\hat{\mu}(\cdot), \hat{p}_t(\cdot))$. We further analyze the asymptotic behavior of the influnence function-based estimator $\hat{\psi}$ to prove that $\hat{\psi}$ attains robustness even if $\hat{\mu}$ is mis-specified. With this theorem, we can claim that our semiparametric estimator is robust to the estimation of $(\hat{\mu}(\cdot), \hat{p}_t(\cdot))$.

Algorithm 4.1: SIE: Stochastic Intervention Effect

Input: Observed units $\{z_i : (\boldsymbol{x}_i, t_i, y_i)\}_{i=1}^n$

1: Initialize a stochastic degree δ.
2: Randomly split Z into k disjoint groups $\{Z_1, \cdots, Z_k\}$
3: **while** each group Z_k **do**
4: Fit the potential outcome model $\hat{\mu}(\boldsymbol{x}, t)$
5: Fit the propensity score $\hat{p}_t(\boldsymbol{x})$ by Eq. (4.7)
6: Calculate $q_t(\boldsymbol{x}; \delta)$ by Eq. (4.6)
7: Calculate $m_1(\boldsymbol{x})$ and $m_0(\boldsymbol{x})$ by Eq. (4.10)
8: Calculate $\varphi(Z_k, \delta)$ by Eq. (4.8)
9: **end while**
10: Calculate the counterfactual outcome $\hat{\psi}(\boldsymbol{x}, t, \delta)$ by Eq. (4.9)
11: Compute $\hat{\tau}_{\text{ATE}} = \frac{1}{n}\sum_{i=1}^n (\hat{\psi}(\boldsymbol{x}_i, t_i, \delta) - y_i)$

Output: stochastic intervention effect $\hat{\tau}_{\text{ATE}}$

This is crucial for incorporating machine learning into our stochastic causal inference framework.

Theorem 4.2. *The stochastic outcome estimator $\hat{\psi}$ in Eq.* (4.9) *is asymptotically linear with influence function ψ, i.e.,*

$$\hat{\psi} - \psi = \mathbb{P}_n\{\varphi(z); \eta\} + o_p(1/\sqrt{n}) \tag{4.12}$$

Proof. For notation simplicity, we use $z = (y, x, t, \delta)$ and $\eta = (\mu(\cdot), p_t(\cdot))$ for the true estimators in the proof. Suppose the estimator $\hat{\eta} = (\hat{\mu}, \hat{p}_t)$ converges to some $\overline{\eta} = (\overline{\mu}, \overline{p}_t)$ in the sense that $\|\hat{\eta} - \overline{\eta}\| = o_p(1)$, where either $\overline{p}_t = p_t$ or $\overline{\mu} = \mu$ correspond to the true nuisance function. Therefor, we conclude that at least one nuisance estimator needs to converge to the correct function, but the other one can be misspecified. We denote the mispecified functions $\tilde{\mu}$ and $\tilde{p}_t$ in the neighborhood of μ and p_t, respectively. From the fact that $\mathbb{P}\{\varphi(z; p_t, \tilde{\mu})\} = \mathbb{P}\{\varphi(z; \tilde{p}_t, \mu)\}$, we have $\mathbb{P}\{\varphi)(z; \overline{\eta}\} = \mathbb{P}\{\varphi)(z; \eta\} = \psi$ for any $\overline{p}_t$ and $\overline{\mu}$. We can write

$$\hat{\psi} - \psi = (\mathbb{P}_n - \mathbb{P})\,\varphi(z; \hat{\eta}) + \mathbb{P}\{\varphi(z; \hat{\eta}) - \varphi(z; \overline{\eta})\} \tag{4.13}$$

If $\hat{\mu}$ and $\hat{p}_t$ are usual parametric functions in Donsker classes (Dudley 2010), then $\varphi(z;\hat{\eta})$ is enabled with Donsker property, i.e.,

$$(\mathbb{P}_n - \mathbb{P})\,\varphi(z;\hat{\eta}) = (\mathbb{P}_n - \mathbb{P})\,\varphi(z;\eta) + o_p(1/\sqrt{n}) \tag{4.14}$$

Substitute Eq. (4.14) to Eq. (4.13), we have

$$\hat{\psi} - \psi = (\mathbb{P}_n - \mathbb{P})\,\varphi(z;\eta) + \mathbb{P}\{\varphi(z;\hat{\eta}) - \varphi(z;\bar{\eta})\} + o_p(1/\sqrt{n}) \tag{4.15}$$

The iterated expectation of term $\mathbb{P}\{\varphi(z;\hat{\eta}) - \varphi(z;\bar{\eta})\}$ in Eq. (4.15) equals

$$\sum_{t\in\{0,1\}} \mathbb{P}\left[\frac{p_t(\boldsymbol{x}) - \hat{p}_t(\boldsymbol{x})}{t\cdot\hat{p}_t(\boldsymbol{x}) + (1-t)\{1-\hat{p}_t(\boldsymbol{x})\}}\{\mu(\boldsymbol{x},t) - \hat{\mu}(\boldsymbol{x},t)\}\right] \tag{4.16}$$

According to the fact that $0 < \hat{p}_t < 1$ and the Cauchy-Schwarz inequality $\mathbb{P}(f\cdot g) \leq \|f\|\|g\|$, then $\mathbb{P}\{\varphi(z;\hat{\eta}) - \varphi(z;\bar{\eta})\} \leq$

$$\sum_{t\in\{0,1\}} \|p_t(\boldsymbol{x}) - \hat{p}_t(\boldsymbol{x})\|\,\|\mu(\boldsymbol{x},t) - \hat{\mu}(\boldsymbol{x},t)\| \tag{4.17}$$

Therefore, if $\hat{p}_t$ a correct parametric model for propensity score, so that $\|\hat{p}_t - p_t\| = o_p(\frac{1}{\sqrt{n}})$, then we only need $\hat{\mu}$ to be consistent, $\|\hat{\mu} - \mu\| = o_p(1)$ to allow $\mathbb{P}\{\varphi(z;\hat{\eta}) - \varphi(z;\bar{\eta})\} = o_p(\frac{1}{\sqrt{n}})$ asymptotically negligible. Then our influence-based estimator satisfied $\hat{\psi} - \psi = (\mathbb{P}_n - \mathbb{P})\,\varphi(z;\eta) + o_p(\frac{1}{\sqrt{n}})$.

□

According to Theorem (4.2), if the propensity score model in Eq. (4.7) is unbiased, the potential outcome model can be estimated by $\hat{\psi}$ in a flexible manner. Because the influence function we defined contains all information about an estimator's asymptotic behavior (up to $o_p(1/\sqrt{n})$ error).

4.5 Stochastic Intervention Optimization

Estimating the stochastic intervention effect is not enough; we are more interested in "what is the optimal level/degree of treatment for a patient to achieve the most expected outcome?". A direct way to find the optimal treatment is through reinforcement learning (Li et al. 2021e; Fang et al. 2019), which focuses on finding policy/intervention for controlling dynamical systems with the goal of maximizing the

desired outcome on downstream decision-making tasks. This is done by the agent repeatedly observing its state, taking action (according to a policy/intervention), and receiving a reward. Over time the agent modifies its policy to maximize its long-term desired outcome. In this paper, we focus particularly on model-free reinforcement learning algorithms, which have become popular in offering off-the-shelf solutions without requiring models of the system dynamics (Feinberg et al. 2018,?). However, the intervention is stochastic rather than deterministic, which tends to result in large training variances in action space. Handling large variance is a significant challenge in model-free reinforcement learning (RL) (Cheng et al. 2019), which would result in the degenerate performance in the intervention optimization.

To alleviate the aforementioned issue, we consider the basic random search method, which explores in the parameter space rather than the action space and thus achieves the optimal expected outcome in a more efficient manner. We model the stochastic intervention using the stochastic propensity score $\hat{q}_t(\boldsymbol{x}, \delta)$, and look for the optimal stochastic interventions parameter $\Delta^* \in \mathbb{R}^{n\text{x}1}$ such that:

$$\Delta^* = \arg\max_{\Delta} \sum_{i=1}^{n} \varphi(z_i, \Delta) \tag{4.18}$$

Note that the optimization problem in Eq. (4.18) is non-differentiable. To avoid using further assumptions for solving it, we formulate a customised reinforcement learning algorithm (Mania et al. 2018) (RS-SIO) to exploit the search space. The main advantage of RS-SIO is model-agnostic which can handle with any black-box functions and flexibly deal with any data type including continuous and categorical features. Therefore, with modifications specific to the intervention effect estimation, RS-SIO solves Eq. (4.5) through the discovery process of trial-and-error search (Qiang and Zhongli 2011; Whitehead and Ballard 1991; Barto and Sutton 1995) which gradually updates the stochastic parameters in every step based on the rewards. Particularly, the algorithm firstly initializes the stochastic intervention parameter $\Delta_0 = \mathbf{0} \in \mathbb{R}^{n\text{x}1}$ and samples a set of $\boldsymbol{\delta}$ having the same size as Δ_0. Thereafter, for each $\boldsymbol{\delta}$, we compute the rewards when the search process moves

toward to the positive ($\Phi(\boldsymbol{\delta_k}, +)$) and negative direction ($\Phi(\boldsymbol{\delta_k}, -)$), and then select the number of b largest awards for these directions as $\max\{\Phi(\boldsymbol{\delta_k}, +), \Phi(\boldsymbol{\delta_k}, -)\}$. In order to update the stochastic parameters Δ, we exploit the update directions $\frac{1}{b}\sum_{k=1}^{b}[\Phi(\boldsymbol{\delta_k}, +) - \Phi(\boldsymbol{\delta_k}, -)]\boldsymbol{\delta_k}$. The full stochastic intervention optimization algorithm is shown in Algorithm 4.2.

Algorithm 4.2: Random Search based Reinforcement Learning for SIO (RS-SIO)

Input: Observed units $\{z_i : (\boldsymbol{x}_i, t_i, y_i)\}_{i=1}^n$, step-size α, standard deviation of the exploration noise , number of steps l, number of top-performing directions b.

Initialize the stochastic intervention parameter $\Delta_0 = \mathbf{0} \in \mathbb{R}^{n\text{x}1}$
Sample $\boldsymbol{\delta}_1$, $\boldsymbol{\delta}_2$,.., $\boldsymbol{\delta}_m$ of the same size as Δ_0.
for $j = 1, \cdots, l$ **do**
  if ending conditions are satisfied **then**
    Break
  end if
  for $k = 1, \cdots, m$ **do**
    for $i = 1, \cdots, n$ **do**
      Compute $q_t(\boldsymbol{x_i}, \boldsymbol{\delta_k})$ by Eq. (4.6)
      Compute $m_1(\boldsymbol{x}_i)$ and $m_0(\boldsymbol{x}_i)$ by Eq. (4.10)
      Compute $\varphi(z_i, \Delta_j + \boldsymbol{\delta_k})$ and $\varphi(z_i, \Delta_j - \boldsymbol{\delta_k})$ by Eq. (4.8).
    end for
  end for
  for $k = 1, \cdots, m$ **do**
    Compute the reward

$$\Phi(\boldsymbol{\delta_k}, +) = \sum_{i=1}^{n} \varphi(z_i, \Delta_j + \boldsymbol{\delta_k}), \quad \Phi(\boldsymbol{\delta_k}, -) = \sum_{i=1}^{n} \varphi(z_i, \Delta_j - \boldsymbol{\delta_k}) \tag{4.19}$$

  end for
  Sort $\boldsymbol{\delta_k}$ by $\max\{\Phi(\boldsymbol{\delta_k}, +), \Phi(\boldsymbol{\delta_k}, -)\}$ and select b top-performing directions.
  Update

$$\Delta_{j+1} = \Delta_j + \frac{\alpha}{b} \sum_{k=1}^{b} [\Phi(\boldsymbol{\delta_k}, +) - \Phi(\boldsymbol{\delta_k}, -)]\boldsymbol{\delta_k} \tag{4.20}$$

end for

Output: Δ_j

4.6 Experiments and Results

In this section, we conduct intensive experiments and compare our framework with state-of-the-art methods on two tasks: treatment effect estimation and stochas-

tic intervention effect optimization. Recall that the influence-based estimator φ depends on the nuisance function of propensity score p_t and outcome μ. We first perform average treatment effect estimation to confirm that $\hat{p}_t$ and $\hat{\mu}$ are unbiased and robust estimators. Moreover, the stochastic intervention optimization task is carried out to demonstrate the effectiveness of our RS-SIO.

4.6.1 Baselines

We briefly describe the comparison methods which are used in two tasks of treatment effect estimation and stochastic intervention optimization.

Evaluating the performance of SIE is not an easy task, because the ground-truth counterfactual outcome is unobserved in practice. On the contrary, the benchmark datasets having two potential outcomes are available for ATE estimation. Therefore, we perform ATE estimation to evaluate the robustness of $\hat{p}_t$ and $\hat{\mu}$ thus to indirectly evaluate the performance of SIE. We use Gradient Boosting algorithm (Natekin and Knoll 2013) with 100 regressors for the potential outcome models $\hat{\mu}$. We compare our proposed estimator (SIE) with the following baselines including Doubly Robust Leaner (Dudík et al. 2011) (LinearDRLearner and ForestDRLearner), IPWE (Austin and Stuart 2015), BART (Hill 2011), Causal Forest (Wager and Athey 2018; Athey et al. 2019), TMLE (Gruber and van der Laan 2012) and OLS (Goldberger et al. 1964). Regarding implementation and parameters setup, we adopt Causal Forest (Wager and Athey 2018; Athey et al. 2019) with 100 trees, BART (Hill 2011) with 50 trees and TMLE (Gruber and van der Laan 2012) from the libraries of cforest, pybart and zepid in Python. For Doubly Robust Learner (DR) (Dudík et al. 2011), we use the two implementations, i.e. LinearDRL and ForestDRL from the package EconML (Research 2019) with Gradient Boosting for regression task with 100 regressors as the outcome model, and Gradient Boosting for classification task with 200 classifiers as the treatment model. Ultimately, we use package DoWhy (Sharma and Kiciman 2020) for IPWE (Austin and Stuart 2015) and OLS.

For stochastic intervention optimization, we compare our proposed method (RS-SIO) with Separate Model Approach (SMA) with different settings. SMA is a uplift

modeling method that estimates the user-level incremental effect of a treatment using machine learning models. SMA (Zaniewicz and Jaroszewicz 2013; Alemi et al. 2009) aims to build two separate regression models for the outcome prediction in the treated and controlled group, respectively. Under the setting of SMA, we apply four well-known models for predicting outcome including Ridge (SMA-Ridge) (Hoerl and Kennard 1970), Gradient Boosting (SMA-GBR) (Friedman 2001), Support Vector Regression (SMA-SVR) (Zaniewicz and Jaroszewicz 2013), and AdaBoost (SMA-AB) (Solomatine and Shrestha 2004). We also compare the performance of these models with the random policy to justify that optimization algorithms can help to target the potential customers to generate greater revenue. For the settings of SMA, we use Support Vector Regression, Gradient Boosting, Ada Boosting Regression and Ridge Regression.

Hyperparameters tuning. For the treatment effect estimation task, we use gird search (Liashchynskyi and Liashchynskyi 2019) for hyperparameters tuning with the optimal values presented in Table 4.2. For the policy optimization task, detailed settings for SMA approaches are shown in Table 4.3, while hyperparameters of our proposed method (RS-SIO) are selected by searching in value ranges presented in Table 4.4.

4.6.2 Datasets

`IHDP` (Hill 2011) is a standard semi-synthetic dataset used in the *Infant Health and Development Program*, which is a popularly used semi-synthetic benchmark containing both the factual and counterfactual outcomes. We conduct the experiments on 100 simulations of `IHDP` dataset, and divide the dataset into training and testing set[†]. The training dataset is highly imbalanced with 139 treated and 608 controlled units out of 747 units, respectively, whilst the testing dataset has 75 units. Each unit has 25 covariates representing the individuals' characteristics. The outcomes are their IQ scores at age three (Dorie 2016).

[†]http://www.fredjo.com/files/ihdp_npci_1-100.train.npz and http://www.fredjo.com/files/ihdp_npci_1-100.test.npz

Online promotion dataset (`OP` Dataset) provided by EconML project (Research 2019) is chosen to evaluate stochastic intervention optimization[‡]. This dataset consists of 10k records in online marketing scenario with the treatment of discount price and the outcome of revenue, each represents a customer with 11 covariates. We split the data into two parts: 80% for training and 20% for testing set. We run 100 repeated experiments with different random states to ensure the model outcome reliability. We aim to investigate how to maximize the revenue by applying different price policies to different customers.

`Lalonde` [§] (Dehejia and Wahba 1999, 2002) is the real-world dataset about the men in the *National Supported Work Demonstration* who were or were not provided the on-job training for more than nine months. Each unit has six features, including age, education, black (1 if black, 0 otherwise), Hispanic (1 if Hispanic, 0 otherwise), married and degree. The outcomes are their earnings in 1975 and 1978 with 297 treated and 425 control observations. The main goal of this dataset is to determine the monetary benefits of the job training on the people. For this dataset, we conduct experiments to find the optimal policy such that their earnings in 1975 and 1978 are maximized. We also repeat experiments 100 times with different random states to ensure model stability.

4.6.3 Evaluation Metrics

In this section, we briefly describe the two evaluation metrics used for stochastic intervention effect estimation and stochastic intervention optimization, respectively.

- **Stochastic Intervention Effect Estimation**. Based on average treatment effect (ATE) in Eq. (4.1), we evaluate the performance of treatment effect estimation by the mean absolute error between the estimated and true ATE:

$$\epsilon_{ATE} = |\hat{\tau}_{\text{ATE}} - \tau_{\text{ATE}}| \tag{4.21}$$

[‡] https://msalicedatapublic.blob.core.windows.net/datasets/Pricing/pricing_sample.csv

[§] https://users.nber.org/~rdehejia/data/.nswdata2.html

- **Stochastic Intervention Optimization**. Followed by the uplifting model studies (Zhao et al. 2017; Hitsch and Misra 2018), we use the expected value of the outcome under the policy proposed by the models as the main metric, which can be measured as:

$$\begin{aligned}\hat{y} &= \mathbb{E}[\boldsymbol{y}|\boldsymbol{t} = \pi(\boldsymbol{x})] \\ &= \frac{1}{n}\sum_{i=0}^{n}\sum_{a=0}^{1}\frac{1}{p_i}y_i\mathbb{I}\{t_i = \pi(x_i)\}\mathbb{I}\{t_i = a\}\end{aligned} \tag{4.22}$$

  where p_i is the propensity score of individual i, $\mathbb{I}\{.\}$ is the indicator function with 1 for true condition and 0 otherwise and $\pi(\boldsymbol{x})$ is the proposed policy. Our method uses $\pi(\boldsymbol{x}) = \Delta(\boldsymbol{x})$ with Δ is the stochastic intervention parameter. Particularly, the expected outcome is computed as if the predicted treatment matches the current treatment, the expected outcome is scaled by the inverse of propensity score y_i/p_i, otherwise, equals zero.

4.6.4 Results and Discussions

In this section, we aim to report the experimental results of 1) how our proposed estimator (SIE) can accurately estimate the average treatment effect; 2) how our optimization algorithm (RS-SIO) can be used for finding optimal stochastic intervention in real-world datasets.

Treatment Effect Estimation

The results of ϵ_{ATE} derived from `IHDP` dataset with 100 simulations and `OP` dataset with 100 repeated experiments are presented in the Table 4.1. As seen clearly, amongst all approaches, our proposed estimator SIE achieves the best performance under ϵ_{ATE} in both two datasets, followed by TMLE for `IHDP` dataset, and LinearDRL and ForestDRL for `OP` dataset. Particularly, on `IHDP`, SIE outperforms all other methods in both training and testing sets. In order to investigate the impact of data size chosen on estimation, we also run experiments and plot the performance of models in different data sizes in Figure **??**. Notably, SIE consistently produces the more accurate average treatment effect than others as the data size increases, while TMLE is ranked second in this dataset. LinearDRL and Causal Forest

also produce very competitive results, whereas IPWE performs the worst. For the experimental results on the online promotion dataset, SIE consistently achieves the outstanding performance under ϵ_{ATE}, followed by the performance of LinearDRL and ForestDRL, while the competitive results are also recorded with BART. It is also worthy to note that although TMLE performs well in the training set, its performance likely degrades when dealing with out-of-sample data in the testing set. Regarding the computational time, IPWE and BART are the best-performing and worst-performing, respectively. Our proposed method is ranked third among the baselines for both two datasets, which is acceptable in consideration of our superior performances on treatment effect estimation compared to IPWE and BART.

Method	IHDP Dataset			OP Dataset		
	Train	Test	Time (ms)	Train	Test	Time (s)
OLS	0.746 ± 0.140	1.264 ± 0.250	242.498 ± 0.000	5.906 ± 0.004	5.906 ± 0.004	8.891 ± 0.000
BART	1.087 ± 0.120	2.808 ± 0.100	2353.843 ± 0.000	0.504 ± 0.042	0.505 ± 0.043	14.180 ± 0.000
Causal Forest	0.360 ± 0.050	0.883 ± 0.614	180.100 ± 0.000	3.520 ± 0.034	3.520 ± 0.034	5.907 ± 0.000
TMLE	0.326 ± 0.060	0.831 ± 0.175	584.659 ± 0.000	0.660 ± 0.000	3.273 ± 0.000	9.723 ± 0.000
ForestDRLearner	1.044 ± 0.040	1.224 ± 0.080	241.148 ± 0.000	0.240 ± 0.014	0.241 ± 0.013	7.807 ± 0.000
LinearDRLearner	0.691 ± 0.080	0.797 ± 0.170	269.193 ± 0.000	0.139 ± 0.009	0.139 ± 0.008	7.107 ± 0.000
IPWE	1.701 ± 0.140	5.897 ± 0.300	84.531 ± 0.000	5.908 ± 0.000	5.908 ± 0.015	2.1725 ± 0.000
SIE	**0.284 ± 0.050**	**0.424 ± 0.090**	200.135 ± 0.000	**0.137 ± 0.000**	**0.119 ± 0.000**	7.002 ± 0.000

Table 4.1 : ϵ_{ATE} and running time of baselines on IHDP(lower is better).

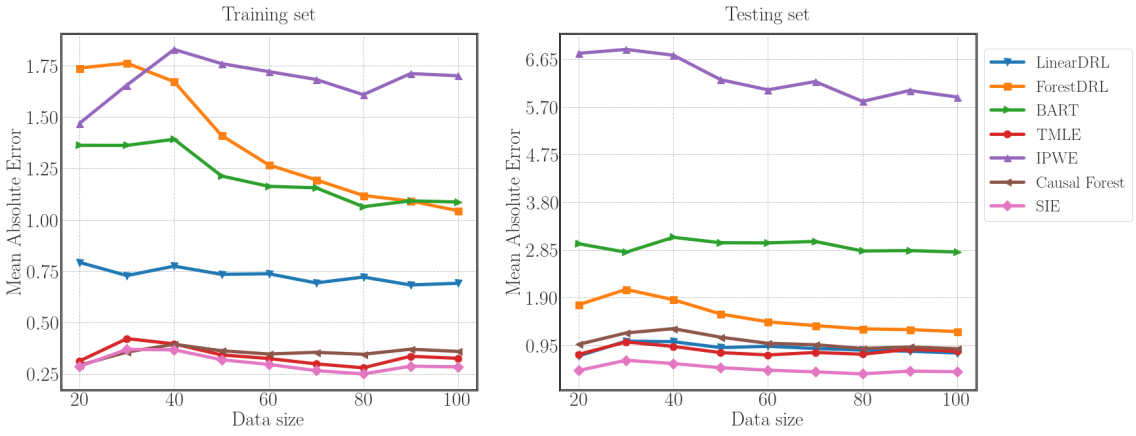


Figure 4.1 : ϵ_{ATE} of baselines on IHDP dataset with different samples.

Algorithm	Parameter name	Value	Package / Language
OLS	Outcome model	Linear regression	
BART	Number of trees	50	bartpy[¶]
Causal Forest	Number of trees	100	cfforest [‖]
	Split ratio	0.7	
	Min leaf	5	
	Max depth	25	
TMLE	Number of estimators	200	TMLE[**]
	Treatment model	Logistic Regression	
	Outcome model	Gradient Boosting	
ForestDRLearner	Treatment model	Gradient Boosting	dowhy[††]
	Outcome model	Gradient Boosting	
	n_estimators for treatment model	200	
	n_estimators for outcome model	100	
LinearDRLearner	Treatment model	Gradient Boosting	dowhy[‡‡]
	Outcome model	Gradient Boosting	
	n_estimators for treatment model	200	
	n_estimators for outcome model	100	
IPWE	Treatment model	Logistic Regression	dowhy
SIE	Treatment model	Gradient Boosting	
	Outcome model	Gradient Boosting	
	n_estimators for treatment model	250	
	n_estimators for outcome model	150	

Table 4.2 : Hyperparameters for treatment effect estimation in `IHDP` dataset. We denote n_estimators for number of predictive models using in Boosting algorithms.

Stochastic Intervention Optimization

For the online promotion dataset, we model the revenue as the expected outcome of each customer under the policy/intervention. Figure 4.2 presents the revenue of uplifting modeling methods with different data sizes including 1000, 5000 and 10000 records. We set step $l = 100$ for our proposed method (RS-SIO). RS-SIO compares favorably to recent uplift modeling techniques that optimize the policy (or intervention) on treatment to maximize the expected outcome. Apparently, RS-

Method	Predictive model	Hyperparameter	Selected values			Package
			Revenue OP	Earning 1975	Earning 1978	
SMA-SVR	Support Vector Regression	kernel	linear	poly	poly	scikit-learn
SMA-GBR	Gradient Boosting	n_estimator	50	100	150	
SMA-ABR	Ada Boosting Regression	n_estimator	50	100	150	
SMA-Ridge	Ridge Regression	solver	auto	auto	auto	

Table 4.3 : Hyperparameters for policy optimization methods. We denote n_estimators for number of predictive models using in Boosting algorithms

Parameters	Search space	Selected values		
		Revenue	Earning 1975	Earning 1978
step_size (α)	[0.01, 0.02, 0.04, 0.06, 0.08, 0.1]	0.08	0.06	0.08
exploration noise ()	[0.01, 0.02, 0.04, 0.06, 0.08, 0.1]	0.02	0.04	0.04
top-performing direction (b)	[10, 20, 25, 35, 40, 45, 50, 55]	35	55	50

Table 4.4 : Value ranges of hyperparameters used during hyperarameter tuning of our proposed method (RS-SIO). Revenue is for `Lalonde` dataset, and earning 1975 and 1978 are for `Lalonde` dataset.

SIO generally produces the highest revenue in datasets with different samples, while SMA-ABR achieves the second-best performance with a very competitive result. Moreover, we find that no significant difference in the performance of SMA with different settings. In contrast, random stochastic intervention produces the lowest revenue, which fails to target the customers for the promotion. On the other hand, Figure 4.3 illustrates the predicted earnings in 1975 and 1978 by different methods. As can be seen, the maximum earning is produced by our proposed method, while random policy/intervention produces the lowest earnings in 1975 and 1978. SMA-Ridge and SMA-GBR achieve competitive performance in this dataset. The possible reason behind our outstanding performance is that instead of focusing on predicting the outcomes like SMA, we directly intervene into the propensity score to produce the best stochastic intervention. Apart from that, we also exploit the sensitivity of parameters (i.e., exploration noise, number of steps and number of top-performing directions) of our policy optimization approach (RS-SIO) in `Lalonde` dataset in two years 1975 and 1978. Figure 4.4 illustrates the curves of performance (i.e., the revenue outcome) produced by varying one parameter (i.e., defined in x-axis)

and fixing the other two parameters. This figure indicates that in most cases our performance is stable under the permutations of every individual parameter.

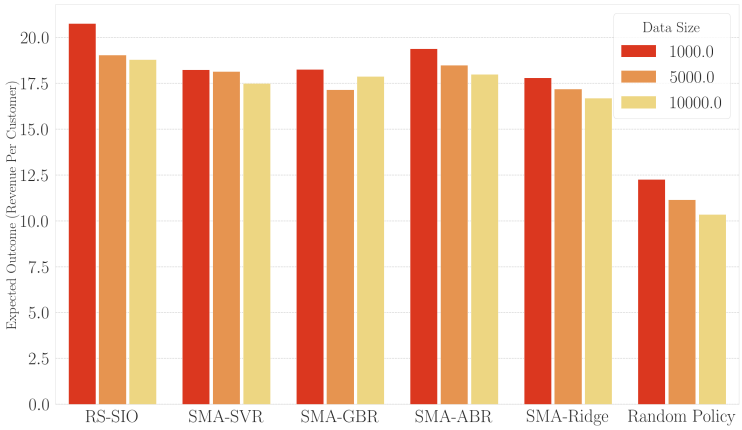


Figure 4.2 : Intervention optimization on `OP` dataset by different baselines.

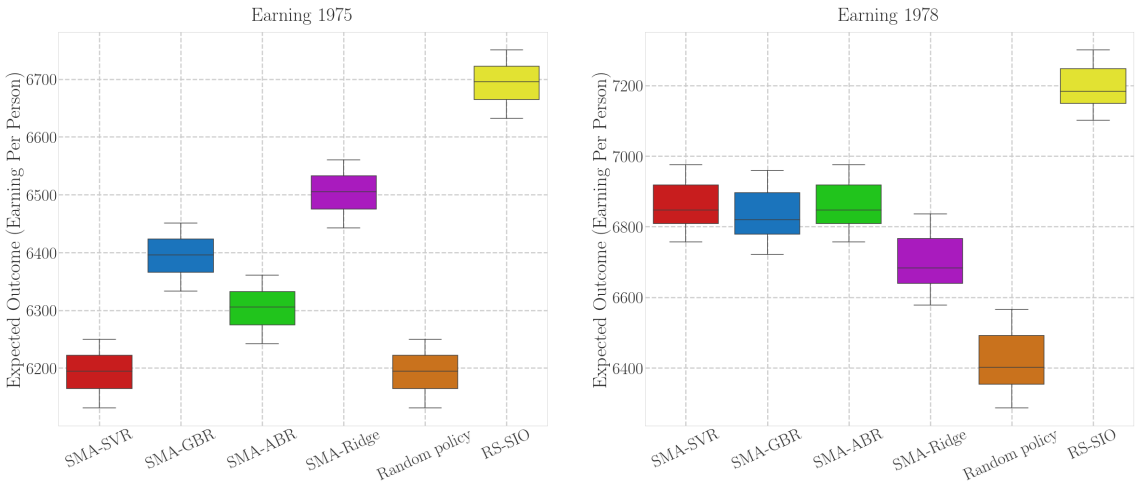


Figure 4.3 : Intervention optimization on `Lalonde` dataset by different baselines.

4.7 Conclusion

We have developed a causal inference framework that admits the stochastic intervention in treatment effect estimation and designs an effective causal solution for the intervention effect optimization. In general, the contribution of this study is twofold. Firstly, we propose a novel treatment effect estimator based on stochastic propensity score so as to learn the dynamic stochastic intervention effect in a

Figure 4.4 : Hyperparameter sensitivity.

flexible manner. Secondly, we design a reinforcement learning algorithm to find the optimal intervention for maximizing the expected outcome, thus providing causal insights for an effective decision-making process. We provide theoretical guarantees for the stochastic intervention effect estimator to achieve double robustness and fast parametric convergence rates. Extensive numerical results justify that our framework outperforms state-of-the-art baselines in both treatment effect estimation and stochastic intervention optimization.

One limitation of our causal framework is that the stochastic intervention is set to static data, i.e., the observational data are time-independent. In many real-world applications, however, events change over time, e.g., each unit may receive a stochastic intervention multiple times, and the timing of these interventions may differ across units (Kennedy et al. 2017; Hill 2011; Galagate 2016). Of practical interest is to perform a more detailed empirical study on the time-dependent stochastic intervention.

Part III

Counterfactual explanation

Part III provides an in-depth exploration of advanced methodologies in counterfactual explanations. Each chapter provides insights and proposed approaches to address complex challenges in the field. In Chapter 5, a causality-focused method is introduced for generating counterfactual explanations through the utilization of multi-objective optimization. This method aims to produce counterfactual samples that not only fulfill the expected outcomes but also retain causal relationship in counterfactual samples. The effectiveness of this approach is rigorously assessed on multiple real-world datasets, with comprehensive comparisons made against existing methods. Regarding Chapter 6, we explore the the combination of normalizing flows and counterfactual explanation. This methodology capitalizes on the strengths of normalizing flows in understanding hidden data distributions, leading to the generation of robust samples. This innovative approach holds significant promise for advancing the field of counterfactual explanation.

Chapter 5

Causality-based counterfactual explanation for classification models

Counterfactual explanation is one branch of interpretable machine learning that produces a perturbation sample to change the model's original decision. The generated samples can act as a recommendation for end-users to achieve their desired outputs. Most of the current counterfactual explanation approaches are the gradient-based method, which can only optimize the differentiable loss functions with continuous variables. Accordingly, the gradient-free methods are proposed to handle the categorical variables, which however have several major limitations: 1) causal relationships among features are typically ignored when generating the counterfactuals, possibly resulting in impractical guidelines for decision-makers; 2) the counterfactual explanation algorithm requires a great deal of effort into parameter tuning for dertermining the optimal weight for each loss functions which must be conducted repeatedly for different datasets and settings. In this work, to address the above limitations, we propose a prototype-based counterfactual explanation framework (ProCE). ProCE is capable of preserving the causal relationship underlying the features of the counterfactual data. In addition, we design a novel gradient-free optimization based on the multi-objective genetic algorithm that generates the counterfactual explanations for the mixed-type of continuous and categorical features. Numerical experiments demonstrate that our method compares favorably with state-of-the-art methods and therefore is applicable to existing prediction models. All the source codes and data are available at https://github.com/tridungduong16/multiobj-scm-cf. The majority of the content in this chapter is derived from the following paper:

- **Duong, T. D.**, Li, Q., & Xu, G. (2022) Causality-based counterfactual ex-

planation for classification models (Under review for Expert System with Application)

5.1 Introduction

Machine learning (ML) is increasingly recognized as an effective approach for large-scale automated decisions in several domains. However, when an ML model is deployed in critical decision-making scenarios such as criminal justice (Završnik 2021; Kaur et al. 2020) or credit assessment (Galindo and Tamayo 2000), many people are skeptical about its accountability and reliability. Hence, interpretablity is vital to make machine learning models transparent and understandable by humans. Recent years witness an increasing number of studies that have explored ML mechanisms under the causal perspective (Schwab and Karlen 2019b; Williams et al. 2016; Zhao and Hastie 2021). Among these studies, counterfactual explanation (CE) is the prominent example-based method that focuses on generating counterfactual samples for interpreting model decisions. For example, consider a customer `A` whose loan application has been rejected by the ML model of a bank. Counterfactual explanations can generate a "what-if" scenario of this person, e.g., "your loan would have been approved if your income was $51,000 more". Namely, the goal of counterfactual explanation is to generate perturbations of an input that leads to a different outcome from the ML model. By allowing users to explore such "what-if" scenarios, counterfactual examples are interpretable and are easily understandable by humans.

Despite recent interests in counterfactual explanations, existing methods suffer three limitations. First, the counterfactual methods neglect the causal relationship among features, leading to the infeasible counterfactual samples for decision makers (Ustun et al. 2019; Poyiadzi et al. 2020). In fact, a counterfactual sample is considered as feasible if the changes satisfy conditions restricted by the causal relations. For example, since education causes the choice of the occupation, changing the occupation without changing the education is infeasible for the loan applicant in the real world. Namely, the generated counterfactuals need to preserve the causal relations between features in order to be realistic and actionable. Second, on the algorithm

level, most counterfactual methods use the gradient-free optimization algorithm to deal with various data and model types (Sharma et al. 2020; Poyiadzi et al. 2020; Dhurandhar et al. 2019; Grath et al. 2018; Lash et al. 2017). These gradient-free optimizations rely on the heuristic search, which however suffers from inefficiency due to the large heuristic search space. In addition, optimizing the trade-off among different loss terms in the objective function is difficult, which often leads to sub-optimal counterfactual samples (Mahajan et al. 2019; Mothilal et al. 2020b; Grath et al. 2018).

To address the above limitations, we propose a prototype-based counterfactual explanation framework (ProCE) in this paper. ProCE is a model-agnostic method and is capable of explaining the classification in the mixed feature space. It should be emphasized that the proposed method focuses on maintaining the causal relationships among the features in dataset instead of the causal relationship between features and target variable (Fernández-Loría et al. 2020). Overall, our contributions are summarized as follows:

- By integrate causal discovery framework and causal loss function, our proposed method can produce the counterfactual samples that satisfy the causal constraints among features.
- We utilize the auto-encoder model and class prototype to guide the search progress and speed up the searching speed of counterfactual samples.
- We design a novel multi-objective optimization that can find the optimal trade-off between the objectives while maintaining diversity in counterfactual explanations' feature space.

5.2 Background

5.2.1 Preliminary

Throughout the paper, lower-cased letters x and $\boldsymbol{x}$ denote the deterministic scalars and vectors, respectively. We consider a dataset $\mathcal{D} = \{\boldsymbol{x}_i, c_i\}_{i=1}^{n}$ consisting

of n instances, where $\boldsymbol{x}_i \in \mathcal{X}$ is a sample, $c_i \in \mathcal{C} = \{0, 1\}$ is the class of individuals $\boldsymbol{x}_i$, and $\boldsymbol{x}_i^j$ is the j-th feature of $\boldsymbol{x}_i$. Also, we consider a classifier $\mathcal{H} : \mathcal{X} \rightarrow \mathcal{Y}$ that has the input of feature space $\mathcal{X}$ and the output as $\mathcal{Y} = \{0, 1\}$. We denote $Q_\phi(.)$ as an encoder model parameterized by ϕ. Finally, $\text{proto}_*(\boldsymbol{x})$ and $\mathcal{K}(\boldsymbol{x})$ are the prototype and the set of K-nearest instances of an instance $\boldsymbol{x}$, respectively.

Definition 5.1 (Counterfactual Explanation)**.** *With the original sample $\boldsymbol{x}_{org} \in \mathcal{X}$, and original prediction $y_{org} \in \mathcal{Y}$, the counterfactual explanation aims to find the nearest counterfactual sample $\boldsymbol{x}_{cf}$ such that the outcome of classifier for $\boldsymbol{x}_{cf}$ changes to desired output class y_{cf}. In general, the counterfactual explanation $\boldsymbol{x}_{cf}$ for the individual $\boldsymbol{x}_{org}$ is the solution of the following optimization problem:*

$$\boldsymbol{x}_{cf}^* = \underset{\boldsymbol{x}_{cf} \in \mathcal{X}}{\text{argmin}}\, f(\boldsymbol{x}_{cf}) \quad \textit{subject to} \quad \mathcal{H}(\boldsymbol{x}_{cf}^*) = y_{cf} \tag{5.1}$$

where $f(\boldsymbol{x}_{\text{cf}})$ is the function measuring the distance between $\boldsymbol{x}_{\text{org}}$ and $\boldsymbol{x}_{\text{cf}}$. Eq (6.1) demonstrates the optimization objective that minimizes the similarity of the counterfactual and original samples, as well as ensures the classifier to change its decision output. For such explanations to be plausible, they should only suggest small changes in a few features.

To make it clear, we consider a simple scenario that a person with a set of features {income: \$50k, CreditScore: "good", education: "bachelor" , age: 52} applies for a loan in a financial organization and receives the reject decision from a predictive model. In this case, the company can utilize the counterfactual explanation (CF) as an advisor that provides constructive advice for this customer. To allow this customer successfully get the loan, CF can give an advice that how to change the customer's profile such as increasing his/her income to \$51k, or enhancing the education degree to "Master". This toy example illustrates that CF is capable of providing interpretable advice that how to makes the least changes for the sample to achieve the desired outcome.

5.2.2 Related Work

Recently, there has been an increasing number of studies in this field. The existing counterfactual explanation methods can be categorized into gradient-based methods(Moore et al. 2019; Wachter et al. 2017; Mothilal et al. 2020b), auto-encoder model (Dhurandhar et al. 2018; Mahajan et al. 2019), heuristic search based methods (Poyiadzi et al. 2020; Sharma et al. 2020) and integer linear optimization (Cui et al. 2015; Kanamori et al. 2020).

Gradient-based methods: Counterfactual explanation is first proposed by the study (Wachter et al. 2017) as the example-based method to interpret machine learning models' decision. In this study, the authors construct the cross-entropy loss between the desired class and counterfactual samples' prediction with the purpose of changing the model output. Thereafter, some gradient-descent optimization algorithms would be used to minimize the constructed loss. This approach draws much attention with a plethora of studies (Grath et al. 2018; Dhurandhar et al. 2018; Mothilal et al. 2020b,b) that aim to customize the loss function to enhance the properties of counterfactual generation. For example, the study (Grath et al. 2018) extends the distance functions in Eq (6.1) by using a weight vector (Θ) to emphasize the importance of each feature. Some algorithms such as k-nearest neighbors or global feature evaluation can be deployed to find this vector (Θ). Another framework called DiCE (Mothilal et al. 2020b) proposes using the diversity score to produce the number of generated samples that allows users to have more options. They thereafter use the weighted sum to combine different loss functions together and also adopt the gradient-descent algorithm to approximately find the optimal solution. The research (Van Looveren and Klaise 2019) utilizes the class prototype to guide the search progress to fall into the distribution of the expected class. This method however does not consider the causal relationship among features. The differentiable methods are the prominent approach in counterfactual explanation that allows to optimize easily and control the loss functions, but are only restricted to the differentiable models, and finds it hard to deal with the non-continuous values in tabular data.

Auto-encoder model: Other recent studies based on the variational auto-encoder (VAE) model utilizes the properties of generative models to generate new counterfactual samples. In the study (Pawelczyk et al. 2020), the authors first construct an encoder-decoder architecture. Thereafter, they generate the latent representation from the encoder, and make some perturbation into the latent representation, and go through the decoder until the prediction models achieve the desired class. Meanwhile, another line of recent work (Mahajan et al. 2019) proposes the conditional auto-encoder model by combining different loss functions including prediction loss and proximity loss. They thereafter generate multiple counterfactual samples for all input data points by conditioning on the target class. These studies heavily rely on gradient-descent optimization which can face difficulties when handling categorical features. In addition, VAE models that maximize the lower bound of the log-likelihood rather than measuring the exact log-likelihood can give unstable and inconsistent results.

Heuristic search methods: There is an increasing number of counterfactual explanation methods for non-differentiable models, which makes the previous gradient-based approach not applicable. They utilizes heuristic search for the optimization problem such as Nelder-Mead (Grath et al. 2018), growing spheres (Laugel et al. 2018), FISTA (Dhurandhar et al. 2019; Van Looveren and Klaise 2019), or genetic algorithms (Dandl et al. 2020; Lash et al. 2017; Sharma et al. 2020). The main idea of these approaches adopts evolutionary algorithms to effectively finds the optimal counterfactual samples based on the defined cost functions. For example, CERTIFAI (Sharma et al. 2020) customizes the genetic algorithm for the counterfactuals search progress. CERTIFAI adopts the indicator functions (1 for different values, else 0) and mean squared error for categorical and continuous features, respectively. Apart from that, the study (Poyiadzi et al. 2020) introduces a method called FACE that adopts Dijsstra's algorithm to generate counterfactual samples by finding the shortest path of the original input and the existing data points. The main advantage of FACE is that the produced path from Dijsstra's algorithm provides an insight into the step-by-step and feasible actions that users can take to

achieve their goals. The generated samples of this method are limited to the input space without generating new data.

Integer linear optimization The studies (Ustun et al. 2019; Cui et al. 2015) propose to adopt integer linear optimization (ILO) solver for linear models utilizing linear costs to generate the actionable changes. Specifically, they formulate the problem of finding counterfactual samples according to the cost function as a mixed-integer linear optimization problem and then utilize some existing solvers (Bliek1ú et al. 2014) to obtain the optimal solution. To speed up the counterfactual samples search process, the study (Artelt and Hammer 2020) introduces convex constraints to bound the solutions in a region of data space locally. Although these approaches seem promising when dealing with non-continuous features and non-differentiable functions, they can be applied to linear models only.

Our method extends the line of studies (Van Looveren and Klaise 2019; Mahajan et al. 2019) by integrating both structural causal model and class prototype. We also formulate the problem as the multi-objective optimization problem and propose an algorithm to find the counterfactual samples effectively.

5.3 Methodology

In this section, we firstly present different objective functions corresponding to different properties of counterfactual samples. The structural causal model and causal distance are also investigated to exploit the underlying causal relationship among features. Then, we formulate the counterfactual sample generation as a multi-objective optimization problem and propose an algorithm based on the non-dominated sorting genetic algorithm (NSGA-II) to obtain the optimal solutions. Figure 5.1 generally describes the overall architecture of our proposed framework containing four main different loss functions: 1) prediction loss that ensures the valid counterfactual samples, 2) proximity loss encourages that only small changes would be performed in the counterfactual samples from the original one, 3) prototype-based loss that guides the search progress, and finally 4) causality-preserving loss that maintains the causal relationships. Moreover, there are three models in the

framework: provided prediction model (h), auto-encoder model (Q_ϕ), and structural causal model ($\mathcal{M}$).

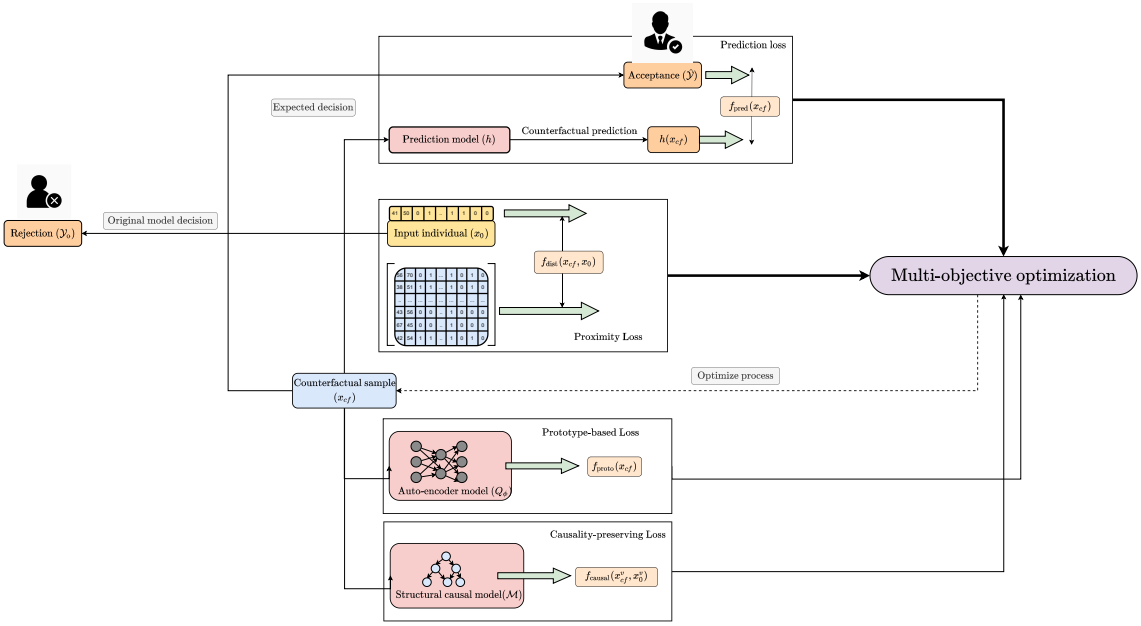


Figure 5.1 : The overall framework for the proposed ProCE. The counterfactual samples are first initialized randomly.

5.3.1 Prototype-based Causal Model

Counterfactuals provide these explanations in the form of "how to assign these features with different values, your credit application would have been accepted". This indicates that counterfactual samples should be constrained under several particular conditions. We first provide definitions of each constraint condition and further tie them together as a multi-objective optimization problem to find an optimal counterfactual explanation. For clarity, we first introduce each constrain condition as loss function as follows.

Prediction Loss

We firstly consider the prediction loss which is the prominent loss function for counterfactual explanation. In order to achieve the desired outcome, prediction loss aims to calculate the distance between the counterfactual and expected/desired

predictions. This loss function encourages the predictive models to change their predictions of counterfactual samples towards the desired outcomes. Particularly, for the classification scenario, we use the cross-entropy loss to minimize the counterfactual and expected outcome. The prediction loss is defined as follows:

$$f_{\text{pred}}(\boldsymbol{x}_{\text{cf}}) = -y_{\text{cf}} \log(\mathcal{H}(\boldsymbol{x}_{\text{cf}})) - (1 - y_{\text{cf}}) \log(1 - \mathcal{H}(\boldsymbol{x}_{\text{cf}})) \tag{5.2}$$

Cross-entropy loss (Russell 2019b) normally measures the performance of a classification model whose output is a probability value between 0 and 1. Cross-entropy loss is considered in this case to increases as the predicted probability of counterfactual samples $\mathcal{H}(\boldsymbol{x}_{\text{cf}})$ diverges from desired outcome y_{cf}.

Prototype-based Loss

In practice, the search space of counterfactuals might be incredibly large which thus results in slow optimization. Inspired by the work (Van Looveren and Klaise 2019), we utilize the class prototype to guide the search progress with the aim of improving the efficiency of finding the counterfactual solutions. Class prototype is first defined as the mean encoding of the instances belonging to the same class (Snell et al. 2017). Therefore, in our work, we construct an auto-encoder model to obtain the latent space which allows us to learn a better representation of these instances.

We resort to an encoder function denoted by $Q_\phi : \mathcal{X} \rightarrow \mathbb{R}^E$ which projects the input feature $\mathcal{X}$ to the E-dimensional latent space. We denote $\mathcal{K}(\boldsymbol{x}_{\text{org}}) = \{\boldsymbol{x}_k, c_k\}_{k=1}^K$ as a set of K-nearest instances of $\boldsymbol{x}_{\text{org}}$ by estimating the latent distance $||Q_\phi(\boldsymbol{x}_k) - Q_\phi(\boldsymbol{x}_{\text{org}})||_2^2$. Moreover, the classes of these K instances, i.e., $\{c_k\}_{k=1}^K$, are different from the original prediction y_{org} meaning that $c_k \neq y_{\text{org}}$. Formally, $\mathcal{K}(\boldsymbol{x}_{\text{org}})$ is defined as:

$$\mathcal{K}(\boldsymbol{x}_{\text{org}}) = \{\boldsymbol{x}_k, c_k\}_{k=1}^K \subset \mathcal{D} \tag{5.3}$$

such that

$$\begin{cases} c_k \neq y_{\text{org}} \\ ||Q_\phi(\boldsymbol{x}_r) - Q_\phi(\boldsymbol{x}_{\text{org}})||_2^2 \geq ||Q_\phi(\boldsymbol{x}_j) - Q_\phi(\boldsymbol{x}_{\text{org}})||_2^2 \quad \forall \boldsymbol{x}_r \in \{\mathcal{D} \backslash \mathcal{K}(\boldsymbol{x}_{\text{org}})\} \end{cases} \tag{5.4}$$

Therefore, a prototype of an original instance $\boldsymbol{x}_{\text{org}}$ is computed by the mean of its nearest neighbors in the latent space:

$$\text{proto}_*(\boldsymbol{x}_{\text{org}}) = \frac{1}{K} \sum_{\boldsymbol{x}_k \in \mathcal{K}(\boldsymbol{x}_{\text{org}})} Q_\phi(\boldsymbol{x}_k) \tag{5.5}$$

The definition of proto_* in Eq. 5.5 indicates that the prototype is in fact the representatives of the samples belonging to counterfactual class. We thus define the prototype loss function as L_2-norm distance between the representation of the counterfactual samples $\boldsymbol{x}_{\text{cf}}$ in the latent space and the obtained prototypes:

$$f_{\text{proto}}(\boldsymbol{x}_{\text{cf}}) = \|Q_\phi(\boldsymbol{x}_{\text{cf}}) - \text{proto}_*\|_2^2 \tag{5.6}$$

Features cost

One of the main obstacles of generating counterfactual samples is to compute the feature cost which captures the effort required for changing from original instance $\boldsymbol{x}_{\text{org}}$ to counterfactual ones $\boldsymbol{x}_{\text{cf}}$. From the fundamental principles of counterfactual explanation, the generated samples should be as close as to the original one. The smallest changes mean that the least efforts are made for decision-makers to take to achieve their desired goals. However, even experts would find it hard to put the precise cost to demonstrate how unactionable the feature is. Moreover, when it comes to the mixed-type tabular data that contains both the categorical and continuous features, it is challenging to define the distance loss function (Jia et al. 2015; Kaufman and Rousseeuw 2009; van de Velden et al. 2019; Foss et al. 2019). The previous studies (Sharma et al. 2020; Mothilal et al. 2020b; Dandl et al. 2020) normally apply the indicator function that returns 1 when two categorical values match and returns 0 otherwise, and adopts L_2-norm distance for comparing continuous features. However, the indicator function which only returns 0 and 1 fails to measure the degree of similarity of two categories. In this study, we use the encoder model Q_ϕ to map the categorical features into the latent space before estimating their distance. The main advantage of this approach is that the encoder model has the capability to capture the underlying relationship and pattern between each categorical value. This means that manual feature engineering such as assigning weight for each category is not

necessary, thus saving a great deal of time and effort. Thus, we come up with the distance between two samples is defined as below:

$$f_{\text{dist}}(\boldsymbol{x}_{\text{cf}}, \boldsymbol{x}_{\text{org}}) = \begin{cases} \|\boldsymbol{x}_{\text{cf}}^j - \boldsymbol{x}_{\text{org}}^j\|_2^2, & \text{if } \boldsymbol{x}^j \text{ is } j\text{-th continuous feature} \\ \|Q_\phi(\boldsymbol{x}_{\text{cf}}^j) - Q_\phi(\boldsymbol{x}_{\text{org}}^j)\|_2^2, & \text{if } \boldsymbol{x}^j \text{ is } j\text{-th categorical feature} \end{cases} \tag{5.7}$$

Causality-preserving Loss

Although the distance function in Eq. (6.4) demonstrates the similarity of two samples, it fails to capture the causal relationship between each feature. To deal with this problem, we integrate the structural causal model, and thus construct the causal loss function to ensure the features' causal relationships in generated samples. We provide some fundamental definitions about causality and thereafter define the corresponding causal loss. In general, a structural causal model $\mathcal{M} = \{\mathbf{U}, \mathbf{V}, \mathbf{F}\}$ (Pearl 2009a) consists of three main components defined as below:

- $\mathbf{U}$ is the set of exogenous nodes which has no parents in the causal graph.
- $\mathbf{V}$ is the set of random variables which are endogenous nodes whose causal mechanisms we are modeling. These variables have parents in the causal graph.
- $\mathbf{F}$ is the set of structural causal functions describing the causal relationships among the unobserved and observed variables. Specifically, for each node $\boldsymbol{X} \in V$, a function $f_X \in F$ such that $X = f_X(\text{Pa}(X), \mathbf{U}_X)$ where $\text{Pa}(X)$ is the parent nodes of X.

A causal graph indicates a probabilistic graphical model that represents the assumptions about data-generating mechanism. A causal graph consists of a set of nodes and edges where each node represents a random variable, and each edge illustrates the causal relationship. The causal effect in causal model is facilitated by *do-operator* or intervention (Pearl et al. 2000) that assigns value $\boldsymbol{x}$ to a random variable X denoted by $do(\boldsymbol{x})$. The symbol $do(x)$ is a model manipulation on a causal graph $\mathcal{M}$, which is defined as substitution of causal equation $X = f_X(\text{Pa}(X)_\mathcal{G}, \mathbf{U}_X)$ with $X = \boldsymbol{x}$.

For each endogenous node $v \in V$, and its parent nodes $(v_{p_1}, v_{p_2}, \ldots, v_{p_k})$, we estimate each node v as $v = g(v_{p_1}, v_{p_2}, \ldots, v_{p_k})$ to represent their causal relationship with $g(*)$ is the structural causal equation constructed by linear regression model. Since having the full causal graph is often impractical in real-world setting, it is quite challenging to estimate structural causal equation $g(*)$. In this work, we utilize LiNGAM (Shimizu 2014) which is a novel estimation technique based on the non-Gaussianity of the data to determine the function $g(*)$. During the counterfactuals generation progress, we firstly produce the predicted value of endogenous node $\boldsymbol{x}^v$ based on their parents before estimating the distance, which is measured as:

$$\begin{aligned} f_{\text{causal}}(\boldsymbol{x}^v_{\text{cf}}, \boldsymbol{x}^v_{\text{org}}) &= \|\boldsymbol{x}^v_{\text{cf}} - \boldsymbol{x}^v_{\text{org}}\|_2^2 \\ &= \|g(\boldsymbol{x}^{v_{p_1}}_{\text{cf}}, \boldsymbol{x}^{v_{p_2}}_{\text{cf}}, \ldots, \boldsymbol{x}^{v_{p_k}}_{\text{cf}}) - \boldsymbol{x}^v_{\text{org}}\|_2^2 \end{aligned} \tag{5.8}$$

With a set of observed variables containing the endogenous and exogenous ones $\boldsymbol{X} = \{\mathbf{U}, \mathbf{V}\}$, we can re-write the general distance between the original and counterfactual sample is the sum of distance of normal distance and causal distance. For the exogenous nodes $\mathbf{U}$ (nodes without any parents in the causal network), we still utilize the Eq. (5.7) which computingthe distance between two instances, while the causal distance in Eq. (5.8) is employed for exogenous variables $\mathbf{V}$ (the remaining features).

$$f_{\text{final\_dist}}(\boldsymbol{x}_{\text{cf}}) = \sum_{u}^{\mathbf{U}} f_{\text{dist}}(\boldsymbol{x}^u_{\text{cf}}, \boldsymbol{x}^u_{\text{org}}) + \sum_{v}^{\mathbf{V}} f_{\text{causal}}(\boldsymbol{x}^v_{\text{cf}}, \boldsymbol{x}^v_{\text{org}}) \tag{5.9}$$

5.3.2 Multi-objective Optimization

In this section, we aim to describe the proposed algorithm which is used for optimization process. With the loss functions presented in Sections 5.3.1 including f_{pred}, f_{proto}, $f_{\text{final\_dist}}$, we come up with the general objective functions Eq (5.10). These loss functions illustrates different properties that counterfactual samples should adhere to. The general loss functions containing three different losses is:

$$\mathcal{L}(\boldsymbol{x}_{\text{cf}}) = \{f_{\text{pred}}(\boldsymbol{x}_{\text{cf}}), f_{\text{proto}}(\boldsymbol{x}_{\text{cf}}), f_{\text{final\_dist}}(\boldsymbol{x}_{\text{cf}})\} \tag{5.10}$$

Therefore, the optimal solutions can be re-written as follows:

$$\boldsymbol{x}_{\text{cf}}^{*} = \underset{\boldsymbol{x}_{\text{cf}} \in \mathcal{X}}{\operatorname{argmin}} \mathcal{L}(\boldsymbol{x}_{\text{cf}}) \tag{5.11}$$

In order to obtain the optimal solutions, the majority of existing studies (Mahajan et al. 2019; Mothilal et al. 2020b; Grath et al. 2018) uses the trade-off parameter sum assigning each loss function a weight, and combines them together. This approach seems to be reasonable; however, it is very challenging to balance the weights for each loss, resulting in a great deal of efforts and time into hyperparameter tuning. To address this issue, we propose to formulate the counterfactual explanation search as the multi-objective problem (MOP). In this study, we modify the elitist non-dominated sorting genetic algorithm (NSGA-II) (Deb et al. 2002a) to deal with this optimization problem. Its main superiority is to optimize each loss function simultaneously as well as provide the solutions presenting the trade-offs among objective functions. To make it clear, we first present some related definitions. Given a set of n candidate solutions $\mathcal{P} = \{\boldsymbol{x}_i\}_{i=1}^{n}$, we have the following ones:

Definition 5.2 (Dominance in the objective space)**.** *In the multi-objective optimization problem, the goodness of a solution is evaluated by the dominance(Deb et al. 2002b). Given two solutions $\boldsymbol{x}$ and $\hat{\boldsymbol{x}}$ along with a number of p objective functions f_i, we have:*

1. *$\boldsymbol{x}$ weakly dominates $\hat{\boldsymbol{x}}$ ($\boldsymbol{x} \succeq \hat{\boldsymbol{x}}$) iff $f_i(\boldsymbol{x}) \geq f_i(\hat{\boldsymbol{x}})\ \forall i \in \{1, \ldots, p\}$.*

2. *$\boldsymbol{x}$ dominates $\hat{\boldsymbol{x}}$ ($\boldsymbol{x} \succ \hat{\boldsymbol{x}}$) iff $\boldsymbol{x} \succeq \hat{\boldsymbol{x}}$ and $\boldsymbol{x} \neq \hat{\boldsymbol{x}}$.*

Definition 5.3 (Pareto front)**.** *Pareto front is a set of m solutions denoted by $F_* = \{\boldsymbol{x}_j\}_{j=1}^{m} \subset \mathcal{P}$ such that $\boldsymbol{x}_j$ dominates all remaining solutions $\boldsymbol{x}_r \in \{\mathcal{P} \backslash F_*\}$ with all objective functions. It means that $f_i(\boldsymbol{x}_j) \geq f_i(\boldsymbol{x}_r)\ \forall i \in \{1, \ldots, p\}$. The main goal of non-dominated solutions is to provide a reasonable compromise between all the objective functions that enhance one function's performance but not degrade others.*

Definition 5.4 (Non-dominated sorting procedure)**.** *Non-dominated sorting step is mainly used to sort the solutions in population according to the Pareto dominance principle, which plays a central role in the selection procedure. In fact, the set of*

candidate solutions $\mathcal{P}$ can be divided into a set of H disjoint Pareto front as $\mathcal{F} = \{F_1, F_2, \ldots, F_H\}$ where H is the maximum number of fronts. Non-dominated sorting is a procedure for finding them. Particularly, in the non-dominated sorting step, all the non-dominated solutions from Definition 5.3 are selected from the population and are constructed as the Pareto front F_1. After that, the non-dominated solutions are chosen from the remaining population. The process is repeated until all the solutions are assigned to a front F_H.

Definition 5.5 (Crowding distance)**.** *One of the vital characteristics of a population solution is diversity. In order to encourage the diversity of candidate solutions, the simplest approach is to choose the individuals having a low density. Particularly, to measure this characteristic, the crowding distance (Fortin and Parizeau 2013) is used to rank each candidate solution. Specficially, the crowding distance of an instance $\boldsymbol{x}$ is calculated as follows:*

$$d(\boldsymbol{x}) = \sqrt{\sum_{i=1}^{p} \left(\frac{f_i(\boldsymbol{x}_a) - f_i(\boldsymbol{x}_b)}{f_i^{min} - f_i^{max}} \right)^2} \tag{5.12}$$

where p is the number of objective functions, $\boldsymbol{x}_a$ and $\boldsymbol{x}_b$ are two nearest instances of $\boldsymbol{x}$ by calculating the Euclidean distance, f_i is the i-th objective function, f_i^{min} and f_i^{max} are its minimum or maximum value, respectively. The fundamental concept behind crowding distance is to compute the Euclidean distance between each candidate solution $\{\boldsymbol{x}_j\}_{j=1}^{m}$ in a front $F_*$by using p objective functions corresponding to p-dimensional hyper space.

The optimization process for objective function (7.7) is given by Algorithm 6.1. The main idea behinds our approach is that for each generation, the algorithm chooses the Pareto Front for each objective function and evolves to the better ones. We firstly find the nearest class prototype of the original sample $\boldsymbol{x}_{\text{org}}$, which is used to measure the prototype loss function later. For the optimal counterfactual $\boldsymbol{x}_{\text{cf}}^*$ finding progress, each candidate solution is represented by the D-dimensional feature as the genes. A random candidate population is initialized with the Gaussian distribution. Thereafter, the objective functions including f_{pred}, f_{proto}, $f_{\text{final\_dist}}$ are calculated for

each candidate. Non-dominated sorting procedure illustrated in Definition 5.4 is then performed to obtain a set of Pareto fronts $\mathcal{F} = \{F_i\}_{h=1}^{H}$.

The crowding distance function illustrated in Definition 5.5 and Eq. (5.12) then is adopted as the score to assign to each individual in the current population. The algorithm only keeps the candidate solutions having the greatest ranking score, which illustrates that these solutions have low density. The cross-over and mutation procedures (Whitley 1994) are finally performed to generate the next population. Particularly, the cross-over of two parents generates the new candidate solutions by randomly swapping parts of genes. Meanwhile, the mutation procedure randomly alters some genes in the candidate solutions to encourage diversity and avoid local minimums. We repeat this process through many generations to find the optimal counterfactual solution.

Algorithm 5.1: Multi-objective Optimization for Prototype-based Counterfactual Explanation (ProCE)

Input: An original sample $\boldsymbol{x}_{\text{org}}$ with its prediction y_{org}, desired class y_{cf}, a provided machine learning classifier $\mathcal{H}$ and encoder model Q_ϕ.

Compute prototype proto_* by Eq. (5.5).
Initialize a batch of initial population with n candidate solutions $\mathcal{P} = \{\boldsymbol{\Delta}_i\}_{i=1}^n$ with $\boldsymbol{\Delta}_i \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\nu})$.
$\mathcal{Q} = \emptyset$
for $g = 1$ to G generation **do**
    $\mathcal{P} = \mathcal{P} \cup \mathcal{Q}$
    for each candidate solution $\boldsymbol{\Delta}_i$ in $\mathcal{P}$ **do**
        Compute $f_{\text{pred}}(\boldsymbol{\Delta}_i)$ based on Eq. (5.2).
        Use proto_* to compute $f_{\text{proto}}(\boldsymbol{\Delta}_i)$ based on Eq. (5.6).
        Compute $f_{\text{final\_dist}}(\boldsymbol{\Delta}_i)$ based on Eq. (5.9).
    end for
    Obtain $\mathcal{F} = \{F_h\}_{h=1}^H$ by using non-dominated sorting procedure in Definition 5.4.
    $\mathcal{P} = \emptyset$
    $h = 0$
    while $|\mathcal{P}| + |F_h| < n$ **do**
        $\mathcal{P} = \mathcal{P} \cup F_h$
        $h = h + 1$
    end while
    Compute the crowding distance as the ranking score for each solution in $\mathcal{P}$ based on Eq. (5.12).
    Keep n individuals in $\mathcal{P}$ based on ranking score.
    Randomly pair $\lceil n/2 \rceil$ $\{\boldsymbol{\Delta}_1, \boldsymbol{\Delta}_2\} \in \mathcal{P}$
    for each pair $\{\boldsymbol{\Delta}_1, \boldsymbol{\Delta}_2\}$ **do**
        Perform crossover$(\boldsymbol{\Delta}_1, \boldsymbol{\Delta}_2) \rightarrow \boldsymbol{\Delta}_1', \boldsymbol{\Delta}_2'$
        Perform mutation $\boldsymbol{\Delta}_1' \rightarrow \tilde{\boldsymbol{\Delta}}_1, \boldsymbol{\Delta}_2' \rightarrow \tilde{\boldsymbol{\Delta}}_2$
        $\mathcal{Q} = \mathcal{Q} \cup \{\tilde{\boldsymbol{\Delta}}_1, \tilde{\boldsymbol{\Delta}}_2\}$
    end for
end for
$\boldsymbol{\Delta}^* \leftarrow \mathcal{P}[0]$

Output: $\boldsymbol{x}_{\text{cf}} = \boldsymbol{\Delta}^*$

5.4 Experiments

We conduct experiments on four datasets to demonstrate the superior performance of our method when compared with state-of-the-art methods. All implementations are conducted in Python 3.7.7 with 64-bit Red Hat, Intel(R) Xeon(R) Gold 6150 CPU @ 2.70GHz. For our method, we construct the multi-objective optimization algorithm with the support of library Pymoo* (Blank and Deb 2020). More details of implementation settings can be found in our code repository.

5.4.1 Datasets

This section provides information about the datasets, on which we perform the comparison experiments. Our method is capable of generating counterfactual samples while maintaining the causal relationship. To validate this claim, we consider some feature conditions that restrict the generated counterfactual samples for each dataset. For simplicity, we denote $a \propto b$ for the condition that (a increase $\Rightarrow$ b increase) AND (a decrease $\Rightarrow$ b decrease). We use four datasets including `Simple-BN`, `Sangiovese`, `Adult` and `Law`.

`Simple-BN` (Mahajan et al. 2019) is a synthetic dataset containing 10,000 records with three features (a_1,a_2,a_3) and a binary output (y). The data is generated based on the followed causal mechanism:

$$\begin{aligned} a_1 &\sim \mathcal{N}(\mu_1, \sigma_1) \\ a_2 &\sim \mathcal{N}(\mu_2, \sigma_2) \\ a_3|a_1, a_2 &\sim \mathcal{N}(k_3 * (a_1 + a_2)^2 + b_3, \sigma_3) \\ y|a_1, a_2, a_3 &\sim \text{Ber}(\sigma(k_y * (a_1 * a_2) + b_y - a_3)) \end{aligned} \tag{5.13}$$

As illustrated by structural causal equations in Eq (5.13), two random variables a_1 and a_2 follow the corresponding normal distribution $\mathcal{N}(\mu_1, \sigma_1)$ and $\mathcal{N}(\mu_2, \sigma_2)$, while a_3 follows the normal distribution with mean value determined by the function of a_1 and a_2. Additionally, target variable y follows the Bernoulli distribution with the

*https://pymoo.org/algorithms/nsga2.html

function of a_1, a_2 and a_3. Based on these generating mechanism, we consider the following causal relationship between a_1, a_2 and a_3:

$$(a_1, a_2) \propto a_3 \tag{5.14}$$

The condition in Eq (5.13) means that a_3 monotonically increase and decrease by a function of two random variables a_1 and a_2.

`Sangiovese`[†](Magrini et al. 2017) dataset evaluates the impact of several agronomic settings on the quality of the Tuscan grapes. This dataset provides information about 14 continuous features along with the binary output. We consider the task of determining whether the grapes' quality is good or not. Based on the conditional linear Bayesian network provided with the dataset, we consider a causal relationship between two features including mean number of sprouts (SproutN) and mean number of bunches (BunchN) that is:

$$\text{BunchN} \propto \text{SproutN} \tag{5.15}$$

`Adult`[‡](Dua and Graff 2017) is the real-world dataset providing information of loan applicants in the financial organization. It is a mixed-type dataset that consists of instances having both continuous features and categorical features. For this dataset, we consider the task of determining whether the annual income of a person exceeds \$50k dollars. Similar to the study (Mahajan et al. 2019), with $\boldsymbol{x}_*^{\text{age}}$ and $\boldsymbol{x}_*^{\text{education}}$ referring to the feature age and education of an individual, we consider two conditions as below:

$$\boldsymbol{x}_{\text{cf}}^{\text{age}} \geq \boldsymbol{x}_{\text{org}}^{\text{age}} \tag{5.16}$$

$$\boldsymbol{x}_{\text{cf}}^{\text{education}} \propto \boldsymbol{x}_{\text{org}}^{\text{age}} \tag{5.17}$$

Regarding the first condition ($\boldsymbol{x}_{\text{cf}}^{\text{age}} \geq \boldsymbol{x}_{\text{org}}^{\text{age}}$), counterfactual algorithms should not suggest decreasing individuals' ages since it violates the natural constraint that

[†]https://www.bnlearn.com/bnrepository/clgaussian-small.html

[‡]https://archive.ics.uci.edu/ml/datasets/adult

human age increases over time. Meanwhile, the second condition ($\boldsymbol{x}_{\text{cf}}^{\text{education}} \propto \boldsymbol{x}_{\text{org}}^{\text{age}}$) demonstrates the education-age causal relationship that obtaining a higher degree of education such as from "Bachelor" to "PhD" requires years to complete, thus causing age to increase. As a result, any counterfactual sample increasing education-level without increasing age is infeasible.

`Law`[§](Wightman 1998) dataset provides information of students with their features: sex, race and their entrance exam scores (LSAT), grade-point average (GPA) and first year average grade (FYA). The main task is to determine which applicants will be accepted to the law program. We consider a causal relationship:

$$(\text{LSAT}, \text{GPA}) \propto \text{FYA} \tag{5.18}$$

In order to evaluate the models' effectiveness, we randomly split each dataset into 80% training and 20% test set. We conduct 100 repeated experiments, then evaluate performance on the test set and finally report the average statistics.

5.4.2 Evaluation Metrics

In this section, we briefly describe six quantitative metrics that are used to evaluate the performance of our proposed method and baselines. We sample a number of n factual samples and generate the counterfactual samples for them. Meanwhile n_{cat} and n_{con} are the corresponding number of categorical and continuous features. 1(.) is the indicator function that returns 1 when the conditions are satisfied, otherwise returns 0.

Target-class validity (%Tcv) (Mahajan et al. 2019; Poyiadzi et al. 2020) evaluates how well the algorithm can produce valid samples. Particularly, %Tcv is calculated as the ratio of the number of samples belonging to the desired class and the number of factual samples. Higher target-class validity is favorable, demonstrating that the algorithm can generate greater numbers of counterfactual samples towards the desirable target variable.

[§] http://www.seaphe.org/databases.php

$$\%\text{Tcv} = \sum_{i=0}^{n} \frac{1(h(\boldsymbol{x}_{\text{cf}}) = y_{\text{cf}})}{n} \tag{5.19}$$

Causal-constraint validity (%Ccv) measures the percentage of counterfactual samples satisfying the pre-defined causal conditions. With this metric, the main aim is to evaluate how well our algorithm can generate feasible counterfactual samples that do not violate the causal relationship among features (Mahajan et al. 2019). With the causal conditions defined in the Section 5.4.1, using n_s as the number samples satisfying causal conditions, the causal-constraint validity is defined in Eq (5.20). Higher causal-constraint validity is preferable, illustrating the greater number of satisfied counterfactual samples.

$$\%\text{Ccv} = \frac{n_s}{n} \tag{5.20}$$

Categorical proximity measures the proximity for categorical features representing the total number of matches on the values of each category between $\boldsymbol{x}_{\text{cf}}$ and $\boldsymbol{x}_{\text{org}}$. Higher categorical proximity is better, implying that the counterfactual sample preserves the minimal changes from the original (Mothilal et al. 2020b).

$$\text{Cat\_proximity} = 1 - \sum_{i=0}^{n} \sum_{j=0}^{n_{cat}} 1(\boldsymbol{x}_{\text{cf}}^{j} \neq \boldsymbol{x}_{\text{org}}^{j}) \tag{5.21}$$

Continuous proximity illustrates the proximity of the continuous features, which is calculated as the negative of L_2-norm distance between the continuous features in $\boldsymbol{x}_{\text{cf}}$ and $\boldsymbol{x}_{\text{org}}$. Higher continuous proximity is preferable, implying that the distance between the continuous features of $\boldsymbol{x}_{\text{org}}$ and $\boldsymbol{x}_{\text{cf}}$ should be as small as possible (Mothilal et al. 2020b).

$$\text{Con\_proximity} = -\sum_{i=0}^{n} \sum_{j=0}^{n_{con}} \|\boldsymbol{x}_{cf}^{j} - \boldsymbol{x}_{0}^{j}\|_{2}^{2} \tag{5.22}$$

IM1 and IM2 are two interpretability metrics (IM) proposed in (Van Looveren and Klaise 2019). Let Q_{ϕ}^{org}, Q_{ϕ}^{cf} and Q_{ϕ}^{full} be the auto-encoder models trained specifically on samples of class y_{org}, samples of class y_{cf} and the full dataset, respectively, we first provide the general idea behind these two metrics. On the one

hand, **IM1** measures the ratio of reconstruction errors of counterfactual sample $\boldsymbol{x}_{\text{cf}}$ using Q_ϕ^{cf} and Q_ϕ^{org}. A smaller value for **IM1** indicates that $\boldsymbol{x}_{\text{cf}}$ can be reconstructed more accurately by the autoencoder trained only on instances of the counterfactual class y_{cf} than by the autoencoder trained on the original class y_{org}. This therefore demonstrate that the counterfactual sample $\boldsymbol{x}_{\text{cf}}$ lies closer to the data manifold of counterfactual class y_{cf}, which is considered to be more interpretable. On the other hand, **IM2** evaluates the similarity of counterfactual sample $\boldsymbol{x}_{\text{cf}}$ produced by Q_ϕ^{cf} and Q_ϕ. A low value of IM2 means that the reconstructed instances of $\boldsymbol{x}_{\text{cf}}$ are very similar when using either Q_ϕ^{cf} or Q_ϕ^{full}. Therefore, the data distribution of the counterfactual class y_{cf} describes x_{cf} as close as the distribution of all classes. Particularly, **IM1** and **IM2** are defined as follows:

$$\text{IM1}(Q_\phi^{\text{cf}}, Q_\phi^{\text{org}}, \boldsymbol{x}_{\text{cf}}) = \sum_{i=0}^{\text{n}} \frac{\|\boldsymbol{x}_{\text{cf}} - Q_\phi^{\text{cf}}(\boldsymbol{x}_{\text{cf}})\|_2^2}{\|\boldsymbol{x}_{\text{cf}} - Q_\phi^{\text{org}}(\boldsymbol{x}_{\text{cf}})\|_2^2 + \epsilon} \tag{5.23}$$

$$\text{IM2}(Q_\phi^{\text{cf}}, Q_\phi^{\text{full}}, \boldsymbol{x}_{\text{cf}}) = \sum_{i=0}^{\text{n}} \frac{\|Q_\phi^{\text{cf}}(\boldsymbol{x}_{\text{cf}}) - Q_\phi^{\text{full}}(\boldsymbol{x}_{\text{cf}})\| r_2}{\|\boldsymbol{x}_{\text{cf}}\|_2^2 + \epsilon} \tag{5.24}$$

5.4.3 Baseline Methods

We compare our proposed method (ProCE) with several baselines including Wachter (AR), Growing Sphere (GS), CERTIFAI, CCHVAE and FACE. All of them are the recent approaches in the counterfactual explanation with available source codes and framework. The brief description of these baselines are illustrated as follows:

1. **Wachter (Wach)** (Wachter et al. 2017) which is a fundamental approach that generates counterfactual explanations by minimizing L_1-norm by using gradient descent to find counterfactuals x_{cf} as close as to original instance x_{org}.

2. **Growing Sphere (GS)** (Laugel et al. 2017) is a random search algorithm, which generates samples around the factual input point until a point with a corresponding counterfactual class label was found. Growing hyperspheres are utilized to create the random samples around the original instance. This

approach deals with immutable features by excluding them from the search procedure.

3. **CERTIFAI** (Sharma et al. 2019) CERTIFAI is an approach that utilizes genetic algorithm to finds the counterfactual samples more effectively. The source code for this method is not avaibale; therefore, we implement the CERTIFAI with the support from Python library PyGAD[¶].

4. **DiCE** (Mothilal et al. 2020b). DiCE is one of the most prominent counterfactual explanation framework. This construct the weighted sum of different loss functions including proximity, diversity and sparsity together, and optimize the combined loss via the gradient-descent algorithm. For implementation, we utilize the source code[‖] with default settings.

5. **FACE** (Poyiadzi et al. 2020) produces a feasible and actionable set of counterfactual actions based on the shortest path lengths as determined by density-weighted metrics. The generated counterfactuals by this method that are plausible and coherent with the underlying data distribution.

For all the experiments, we build two predictions model namely **1st classifier** and **2nd classifier**. The first classifier is a neural network with three hidden layers, while the second one has five hidden layers with the following architecture:

1st classifier

- hidden Layer 1(Number of features, 64), batch normalization layer, dropout(0.1), activation function ReLU
- hidden Layer 2(64, 32), batch normalization layer, Dropout(0.1), activation function ReLU
- hidden Layer 3(32, 16), batch normalization layer, Dropout(0.1), activation function ReLU

[¶] https://github.com/ahmedfgad/GeneticAlgorithmPython

[‖] https://github.com/divyat09/cf-feasibility

- last Layer (16, Data size), activation function sigmoid

2nd classifier

- hidden layer 1(Number of features, 256), batch normalization layer, Dropout(0.1), activation function ReLU
- hidden layer 2(Number of features, 128), batch normalization layer, Dropout(0.1), activation function ReLU
- hidden layer 3(Number of features, 64), batch normalization layer, Dropout(0.1), activation function ReLU
- hidden layer 4(64, 32), batch normalization layer, Dropout(0.1), activation function ReLU
- hidden layer 5(32, 16), batch normalization layer, Dropout(0.1), activation function ReLU
- last hidden layer (16, Data size), activation function sigmoid

The continuous features in datasets are in different value ranges; therefore, following the common practice in feature engineering (Zheng and Casari 2018; **?**; **?**), we normalize the continuous feature to range (0,1). Moreover, regarding the categorical features, we transform them into numeric forms by using a label encoder.

5.4.4 Results and Discussions

The performance of different metrics on 1st and 2nd classifier are illustrated in Table 5.1 and 5.2, respectively. Regarding to the 1st classifier from Table 5.1, all three methods achieve the competitive target-class validity, except the Watch performance in all datasets with around 90% of samples belonging to the target class. Regarding the percentage of samples satisfying the causal constraints, by far the greatest performance is achieved by ProCE with 85.91%, 91.84%, 95.64% and 90.43% for `Simple-BN`, `Sangiovese`, `Adult` and `Law` datasets, respectively. FACE

also produces a competitive performance across four datasets in terms of this metric, standing at 81.49%, 88.65%, 92.49% and 86.71% while the majority of generated samples from Watch violate the causal constraints (63.61%, 58.1%, 70.40% and 76.71%). The performance of %Ccv cannot be achieved to 100% for all the methods which demonstrates that it is quite challenging to maintain the causal constraints in counterfactual samples. Moreover, these results indicate that by integrating the structural causal model, our proposed method can effectively produce the counterfactual samples preserving the features' causal relationships. Regarding interpretability scores, our proposed method achieved the best IM1 and IM2 on four datasets. DiCE is ranked second recorded with competitive result in `Adult` dataset (0.0809 for IM1 and 0.2679 for IM2) and `Law` dataset (0.0423 for IM1 and 0.0427 for IM2). The performance of all metrics on the 2^{nd} classifier in Table 5.2 also demonstrates the competitive performance of our proposed method across all metrics. We also notice that although the 2^{nd} has a more complicated architecture than the 1^{st} classifier, there is a small variation on the performance of counterfactual explanation algorithm. Finally, as expected, by using prototype as a guideline of the counterfactual search process, ProCE produces more interpretable counterfactual instances recorded with good performance in IM1 and IM2. By contrast, it is challenging for other approaches to reconstruct the counterfactual samples, leading to high interpretability scores (IM1 and IM2).

On the other hand, to better comprehend the effectiveness of our proposed method in producing counterfactual samples compared with other approaches, we also perform a statistical significance test (paired t-test) between our approach (ProCE) and other methods on each dataset and each metric with the obtained results on 100 randomly repeated experiments and report the result of p-value in Table 5.1 and 5.2. We find that our model is statistically significant with $p < 0.05$, thus demonstrating the effectiveness of ProCE in counterfactual samples generation task.

Figure 6.2 provides information about the categorical proximity in the Adult dataset and continuous proximity in four datasets. For the categorical proximity

on both 1st and 2nd classifier, ProCE consistently achieves an average of 5 out of the total 6 categories in the dataset meaning that the counterfactual sample generated from ProCE preserves an average of 5 categorical features from the original instances. CERTIFAI and FACE also yield competitive results for categorical proximity, whereas the lowest result is recorded in the GS algorithm (1.7 to 3.5 categories). These results illustrate that the gradient-free based approach including ProCE, CERTIFAI and FACE can achieve better performance in handling the non-continuous features in tabular data. When it comes to the continuous proximity, ProCE produces the counterfactual sample with the greatest similarity around over -0.02, -0.078, -0.1875 and -0.17 corresponding to `Simple-BN`, `Sangiovese`, `Adult` and `Law dataset`. Our proposed method produces the least fluctuation in continuous proximity for `Sangiovese`, `Simple-BN`, `Adult`, while the biggest variation is witnessed in `Law`.

We also report the running time of different methods in Table 6.2. Overall, the shortest time is recorded with Watch method on `Simple-BN`, `Sangiovese`, and `Law` datasets. The possible reason is that Watch is the naive approach which optimizes the basic proximity loss functions using gradient descent. This therefore allows producing the counterfactual sample in a prominent time but demonstrates a poor performance in several metrics. Our approach (ProCE) also demonstrates competitive time performance on these three datasets. Regarding `Adult` dataset which contains both categorical and continuous features, our approach performs counterfactual sample generation in the outstanding time and also surpasses other non-gradient descent methods such as FACE, CERTIFAI and GS.

Figure 5.3a and 5.3b show the variation of our method's performance with the different numbers of K-nearest neighbors for class prototype and E-dimensional embedding sizes of the auto-encoder model, respectively. It is clear from Figure 5.3a that the performance of continuous proximity for `Simple-BN`, `Sangiovese` and `Adult` datasets is nearly stable with different embedding sizes, while `Law` witnesses a quite significant variation, increasing from around -0.336 to -0.224 corresponding to embedding sizes of 32 to 256, followed by a slight decrease to -0.33 (embedding size

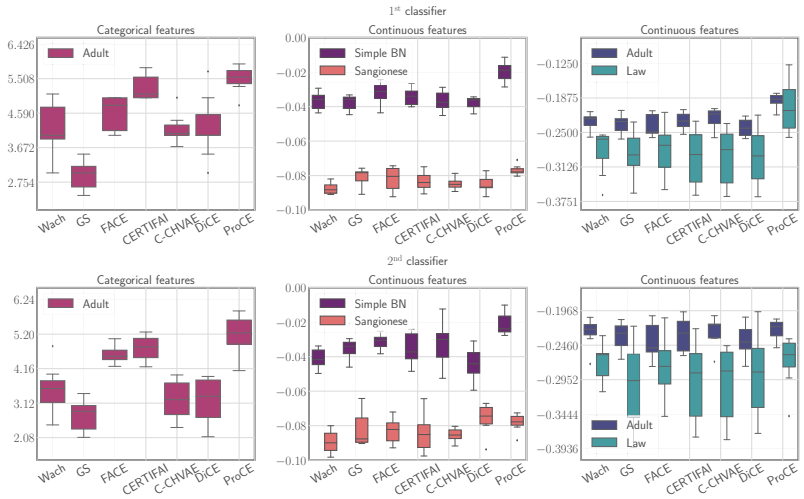
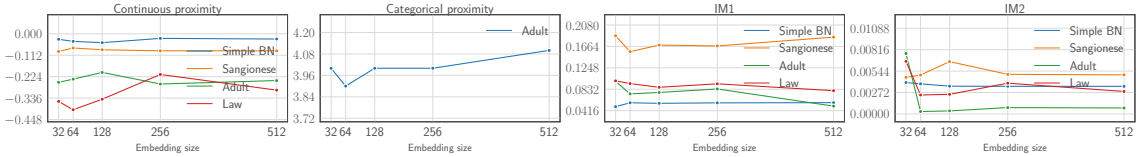
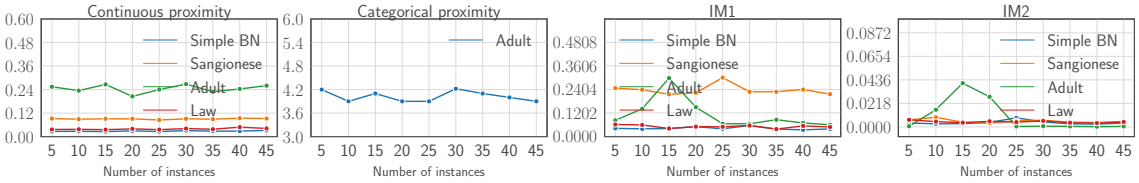


Figure 5.2 : Baseline results in terms of **Continuous proximity** and **Categorical proximity**. Higher continuous and categorical proximity are better.

512). A similar pattern also is recorded for the remaining metrics including categorical proximity, IM1, and IM2 with the good and stable performance at an embedding size of 256. The slight small fluctuations possibly illustrate that the impact of embedding size on the model performance is not very significant. Moreover, 256 is the preferable embedding size, while the sizes of 32 and 512 seem to be relatively small and large to sufficiently capture latent information for embedding vectors. Regarding categorical proximity, the performance declines slightly by 0.1 from 32 to 64, and thereafter varies slightly around 4.0 - 4.09 with embedding sizes of 128, 256, and 512. On the other hand, as can be seen from Figure 5.3b, IM1 and IM2 demonstrate a similar pattern illustrated by the worst performance when the number of instances of 15, followed by a stagnant performance from 25 to 45 instances. It is believed that the similar trend occurring in IM1 and IM2 is reasonable due to their similar properties illustrated in Section 5.4.2. Meanwhile, there is no significant variation in the performance of continuous and categorical proximity across four datasets. These results suggest that the performance of our proposed method witnesses a small variation in all evaluation metrics regarding two hyperparameters (embedding sizes and numbers of nearest neighbors), implying our model's stability and robustness.



(a) Our performance under different sizes of E-dimensional embedding for encoder function Q_ϕ.



(b) Our performance under different numbers of K-nearest neighbors for class prototype

Figure 5.3 : Sensitivity of hyperparameters.

5.5 Conclusion

This paper introduces a novel counterfactual explanation algorithm by integrating the structural causal model and the class prototype. We also proposed formulating the counterfactual generation as a multi-objective problem and construct an optimization algorithm to find the optimal counterfactual explanation in an effective manner. Our experiments validate that our method outperforms the state-of-the-art methods on many evaluation metrics. For future work, we plan to extend our framework to the imperfect structural causal model that is very commonplace in real-world scenarios. Meanwhile, other multi-objective optimization algorithms such as reinforcement learning and multi-task learning are also worthy of investigation.

Method	Dataset	Performance				p-value			
		%Tcv	%Ccv	IM1	IM2 (x10)	%Tcv	%Ccv	IM1	IM2
Wach	Simple-BN	91.00	63.61	0.0379 ± 0.0741	0.0769 ± 0.1385	0.0129	0.0289	0.0393	0.0446
GS	Simple-BN	100.00	79.72	0.0453 ± 0.0835	0.0792 ± 0.0202	0.0340	0.0480	0.0223	0.0483
CERTIFAI	Simple-BN	100.00	77.44	0.0489 ± 0.1353	0.0271 ± 0.0711	0.0098	0.0226	0.0365	0.0218
DiCE	Simple-BN	100.00	73.61	0.0376 ± 0.1345	0.0815 ± 0.1762	0.0227	0.031	0.0135	0.0427
FACE	Simple-BN	100.00	81.49	0.0365 ± 0.0583	0.0429 ± 0.1614	0.0256	0.0197	0.0444	0.0468
ProCE	Simple-BN	**100.00**	**85.91**	**0.0322 ± 0.1014**	**0.0211 ± 0.0845**	-	-	-	-
Wach	Sangiovese	92.03	58.10	0.2513 ± 0.1452	0.0533 ± 0.0132	0.0260	0.0365	0.0447	0.0358
GS	Sangiovese	100.00	89.60	0.2295 ± 0.0584	0.0425 ± 0.1502	0.0131	0.0469	0.014	0.0162
CERTIFAI	Sangiovese	100.00	74.29	0.2915 ± 0.1920	0.0721 ± 0.1366	0.0410	0.0389	0.0215	0.0212
DiCE	Sangiovese	100.00	78.10	0.2447 ± 0.0759	0.0374 ± 0.1657	0.0297	0.0306	0.0388	0.0102
FACE	Sangiovese	100.00	88.65	0.2424 ± 0.0962	0.0873 ± 0.0495	0.0471	0.0148	0.0140	0.0119
ProCE	Sangiovese	**100.00**	**91.84**	**0.2152 ± 0.1686**	**0.0370 ± 0.0574**	-	-	-	-
Wach	Adult	93.95	70.40	0.0709 ± 0.1582	0.3063 ± 0.1382	0.048	0.0285	0.0242	0.0407
GS	Adult	100.00	70.13	0.2241 ± 0.0396	0.3343 ± 0.0564	0.0144	0.0274	0.0114	0.0468
CERTIFAI	Adult	100.00	91.99	0.0939 ± 0.0834	0.3735 ± 0.1150	0.0320	0.0348	0.0310	0.0222
DiCE	Adult	100.00	80.40	0.0809 ± 0.1538	0.2679 ± 0.1661	0.0318	0.0169	0.0275	0.0415
FACE	Adult	100.00	92.49	0.1283 ± 0.0336	0.3245 ± 0.1881	0.0215	0.0346	0.019	0.0242
ProCE	Adult	**100.00**	**95.64**	**0.0675 ± 0.1908**	**0.2171 ± 0.0546**	-	-	-	-
Wach	Law	92.45	76.71	0.0536 ± 0.1312	0.0470 ± 0.0800	0.0159	0.026	0.0115	0.0378
GS	Law	100.00	86.23	0.0484 ± 0.1173	0.0487 ± 0.0858	0.0481	0.0392	0.0314	0.0315
CERTIFAI	Law	100.00	82.72	0.0567 ± 0.1427	0.0461 ± 0.1797	0.0102	0.0425	0.0191	0.0340
DiCE	Law	100.00	85.75	0.0423 ± 0.1902	0.0427 ± 0.0801	0.0138	0.0206	0.0122	0.0487
FACE	Law	100.00	86.71	0.0418 ± 0.0125	0.0435 ± 0.1160	0.0125	0.0315	0.0333	0.0450
ProCE	Law	**100.00**	**90.43**	**0.0410 ± 0.1268**	**0.0421 ± 0.1907**	-	-	-	-

Table 5.1 : Performance of all methods on 1st classifier.We compute p-value by conducting a paired t-test between our approach (ProCE) and baselines with 100 repeated experiments for each metric.

Method	Dataset	Performance				p-value			
		%Tcv	%Ccv	IM1	IM2 (x10)	%Tcv	%Ccv	IM1	IM2
Wach	Simple-BN	93.33	70.96	0.0512 ± 0.0466	0.0262 ± 0.0507	0.0320	0.0096	0.0372	0.0487
GS	Simple-BN	100.00	79.46	0.0401 ± 0.1888	0.0354 ± 0.0352	0.0242	0.038	0.0274	0.0308
CERTIFAI	Simple-BN	100.00	83.68	0.0465 ± 0.0389	0.0824 ± 0.1345	0.0378	0.0138	0.031	0.0255
DiCE	Simple-BN	100.00	82.93	0.0342 ± 0.0790	0.0448 ± 0.0260	0.0376	0.0324	0.0497	0.0277
FACE	Simple-BN	100.00	82.03	0.0458 ± 0.1209	0.0435 ± 0.0123	0.0215	0.0086	0.0275	0.0437
ProCE	Simple-BN	**100.00**	**89.09**	**0.0318 ± 0.0104**	**0.0202 ± 0.0167**	-	-	-	-
Wach	Sangiovese	93.92	74.49	0.2731 ± 0.1090	0.0445 ± 0.0919	0.0255	0.0291	0.0474	0.0363
GS	Sangiovese	100.00	71.44	0.2654 ± 0.0394	0.0407 ± 0.0770	0.0319	0.0378	0.0294	0.0447
CERTIFAI	Sangiovese	100.00	80.95	0.2583 ± 0.1369	0.0798 ± 0.1898	0.0281	0.0304	0.0389	0.0297
DiCE	Sangiovese	100.00	92.25	0.2603 ± 0.1383	0.0880 ± 0.1144	0.0436	0.0323	0.0478	0.0381
FACE	Sangiovese	100.00	77.95	0.2302 ± 0.0029	0.0522 ± 0.0169	0.0464	0.0152	0.0351	0.0184
ProCE	Sangiovese	**100.00**	**86.25**	**0.2127 ± 0.0973**	**0.0360 ± 0.0388**	-	-	-	-
Wach	Adult	91.45	75.23	0.1731 ± 0.1270	0.3520 ± 0.1592	0.0127	0.0454	0.0407	0.0378
GS	Adult	100.00	75.82	0.1719 ± 0.1673	0.1565 ± 0.1634	0.0308	0.0099	0.0224	0.0447
CERTIFAI	Adult	100.00	80.56	0.1512 ± 0.0920	0.2326 ± 0.0686	0.0265	0.0351	0.0309	0.0341
DiCE	Adult	100.00	76.43	0.2371 ± 0.1801	0.3823 ± 0.0016	0.0154	0.0396	0.0427	0.0343
FACE	Adult	100.00	76.02	0.1649 ± 0.1448	0.3393 ± 0.0083	0.0254	0.0144	0.0105	0.0285
ProCE	Adult	**100.00**	**92.85**	**0.1467 ± 0.1096**	**0.1324 ± 0.1027**	-	-	-	-
Wach	Law	90.55	73.36	0.0437 ± 0.0913	0.0594 ± 0.1896	0.0375	0.0474	0.0462	0.0349
GS	Law	100.00	84.09	0.0532 ± 0.0988	0.0643 ± 0.0244	0.0269	0.0267	0.0402	0.0334
CERTIFAI	Law	100.00	80.88	0.0382 ± 0.0915	0.0592 ± 0.0566	0.0495	0.0172	0.0428	0.0286
DiCE	Law	100.00	87.54	0.0382 ± 0.0530	0.0461 ± 0.1928	0.0421	0.0489	0.0342	0.0373
FACE	Law	100.00	75.51	0.0422 ± 0.1875	0.0383 ± 0.0029	0.0476	0.0374	0.015	0.0304
ProCE	Law	**100.00**	**79.48**	**0.0317 ± 0.1073**	**0.0313 ± 0.1648**	-	-	-	-

Table 5.2 : Performance of all methods on 2nd classifier. We compute p-value by conducting a paired t-test between our approach (ProCE) and baselines with 100 repeated experiments for each metric.

Method	Dataset	Running time (s)	
		1st classifier	2nd classifier
Wach	Simple-BN	**3.030 ± 0.105**	**5.111 ± 0.135**
GS	Simple-BN	7.126 ± 0.153	6.541 ± 0.053
CERTIFAI	Simple-BN	6.213 ± 0.007	6.237 ± 0.088
DiCE	Simple-BN	6.522 ± 0.088	6.455 ± 0.016
FACE	Simple-BN	8.022 ± 0.014	6.599 ± 0.173
ProCE	Simple-BN	4.085 ± 0.055	6.017 ± 0.160
Wach	Sangiovese	**5.125 ± 0.097**	**5.768 ± 0.113**
GS	Sangiovese	8.048 ± 0.176	12.549 ± 0.086
CERTIFAI	Sangiovese	7.688 ± 0.131	8.906 ± 0.105
DiCE	Sangiovese	13.426 ± 0.158	11.775 ± 0.086
FACE	Sangiovese	7.810 ± 0.076	11.348 ± 0.200
ProCE	Sangiovese	6.809 ± 0.162	7.304 ± 0.101
Wach	Adult	7.046 ± 0.151	7.260 ± 0.058
GS	Adult	6.472 ± 0.021	6.464 ± 0.145
CERTIFAI	Adult	13.851 ± 0.001	9.457 ± 0.120
DiCE	Adult	7.943 ± 0.046	10.326 ± 0.016
FACE	Adult	10.821 ± 0.162	9.140 ± 0.149
ProCE	Adult	**4.837 ± 0.026**	**5.733 ± 0.019**
Wach	Law	**4.821 ± 0.068**	**4.957 ± 0.131**
GS	Law	12.126 ± 0.093	13.480 ± 0.152
CERTIFAI	Law	5.516 ± 0.009	6.337 ± 0.027
DiCE	Law	6.150 ± 0.038	8.103 ± 0.0410
FACE	Law	5.450 ± 0.184	6.661 ± 0.025
ProCE	Law	4.830 ± 0.130	5.001 ± 0.152

Table 5.3 : We report running time of different methods on four datasets.

Chapter 6

CeFlow: A Robust and Efficient Counterfactual Explanation Framework with Normalizing Flows.

In this chapter, we shift our focus towards two additional aspects of the counterfactual explanations which have already discussed in Chapter 5: robustness and efficiency. Although state-of-the-art counterfactual explanation methods are proposed to use variational autoencoder (VAE) to achieve promising improvements, they suffer from two major limitations: 1) the counterfactuals generation is prohibitively slow, which prevents algorithms from being deployed in interactive environments; 2) the counterfactual explanation algorithms produce unstable results due to the randomness in the sampling procedure of variational autoencoder. In this work, to address the above limitations, we design a robust and efficient counterfactual explanation framework, namely CeFlow, which utilizes normalizing flows for the mixed-type of continuous and categorical features. Numerical experiments demonstrate that our technique compares favorably to state-of-the-art methods. We release our source code* for reproducing the results. The content of this chapter is from:

1. **Duong, T. D.**, Li, Q., & Xu, G. (2023) CeFlow: A robust and efficient counterfactual explanation framework with normalizing flows. Advances in Knowledge Discovery and Data Mining (PAKDD 2023, CORE A).

6.1 Introduction

Machine learning (ML) has resulted in advancements in a variety of scientific and technical fields, including computer vision, natural language processing, and conversational assistants. Interpretable machine learning is a machine learning sub-field

*https://github.com/tridungduong16/fairCE.git

that aims to provide a collection of tools, methodologies, and algorithms capable of producing high-quality explanations for machine learning model judgments. A great deal of methods in interpretable ML methods has been proposed in recent years. Among these approaches, counterfactual explanation (CE) is the prominent example-based method involved in how to alter features to change the model predictions and thus generates counterfactual samples for explaining and interpreting models (Mahajan et al. 2019; Artelt and Hammer 2020; Grath et al. 2018; Wachter et al. 2017; Xu et al. 2020). An example is that for a customer `A` rejected by a loan application, counterfactual explanation algorithms aim to generate counterfactual samples such as "your loan would have been approved if your income was $51,000 more" which can act as a recommendation for a person to achieve the desired outcome. Providing counterfactual samples for black-box models has the capability to facilitate human-machine interaction, thus promoting the application of ML models in several fields.

The recent studies in counterfactual explanation utilize variational autoencoder (VAE) as a generative model to generate counterfactual sample (Pawelczyk et al. 2020; Mahajan et al. 2019). Specifically, the authors first build an encoder and decoder model from the training data. Thereafter, the original input would go through the encoder model to obtain the latent representation. They make the perturbation into this representation and pass the perturbed vector to the encoder until getting the desired output. However, these approaches present some limitations. First, the latent representation which is sampled from the encoder model would be changed corresponding to different sampling times, leading to unstable counterfactual samples. Thus, the counterfactual explanation algorithm is not robust when deployed in real applications. Second, the process of making perturbation into latent representation is so prohibitively slow (Mahajan et al. 2019) since they need to add random vectors to the latent vector repeatedly; accordingly, the running time of algorithms grows significantly. Finally, the generated counterfactual samples are not closely connected to the density region, making generated explanations infeasible and non-actionable. To address all of these limitations, we propose a Flow-based coun-

terfactual explanation framework (CeFlow) that integrates normalizing flow which is an invertible neural network as the generative model to generate counterfactual samples. Our contributions can be summarized as follows:

- We introduce CeFlow, an efficient and robust counterfactual explanation framework that leverages the power of normalizing flows in modeling data distributions to generate counterfactual samples. The usage of flow-based models enables to produce more robust counterfactual samples and reduce the algorithm running time.
- We construct a conditional normalizing flow model that can deal with tabular data consisting of continuous and categorical features by utilizing variational dequantization and Gaussian mixture models.
- The generated samples from CeFlow are close to and related to high-density regions of other data points with the desired class. This makes counterfactual samples likely reachable and therefore naturally follow the distribution of the dataset.

6.2 Related works

An increasing number of methods have been proposed for the counterfactual explanation. The existing methods can be categorized into gradient-based methods (Wachter et al. 2017; Mothilal et al. 2020b), auto-encoder model (Mahajan et al. 2019), heuristic search methods (Poyiadzi et al. 2020; Sharma et al. 2020) and integer linear optimization (Kanamori et al. 2020). Regarding gradient-based methods, The authors in the study construct the cross-entropy loss between the desired class and counterfactual samples' prediction with the purpose of changing the model output. The created loss would then be minimized using gradient-descent optimization methods. In terms of auto-encoder model, generative models such as variational auto-encoder (VAE) is used to generate new samples in another line of research. The authors (Pawelczyk et al. 2020) first construct an encoder-decoder architecture. They then utilize the encoder to generate the latent representation,

make some changes to it, and run it through the decoder until the prediction models achieve the goal class. However, VAE models which maximize the lower bound of the log-likelihood instead of measuring exact log-likelihood can produce unstable and unreliable results. On the other hand, there is an increasing number of counterfactual explanation methods based on heuristic search to select the best counterfactual samples such as Nelder-Mead (Grath et al. 2018), growing spheres (Laugel et al. 2018), FISTA (Dhurandhar et al. 2019; Van Looveren and Klaise 2019), or genetic algorithms (Dandl et al. 2020; Lash et al. 2017). Finally, the studies (Ustun et al. 2019) propose to formulate the problem of finding counterfactual samples as a mixed-integer linear optimization problem and utilize some existing solvers (Bliek1ú et al. 2014; Artelt and Hammer 2020) to obtain the optimal solution.

6.3 Preliminaries

Throughout the paper, lower-cased letters x and $\boldsymbol{x}$ denote the deterministic scalars and vectors, respectively. We consider a classifier $\mathcal{H} : \mathcal{X} \to \mathcal{Y}$ that has the input of feature space $\mathcal{X}$ and the output as $\mathcal{Y} = \{1...\mathcal{C}\}$ with $\mathcal{C}$ classes. Meanwhile, we denote a dataset $\mathcal{D} = \{\boldsymbol{x}_n, y_n\}_{n=1}^{N}$ consisting of N instances where $\boldsymbol{x}_n \in \mathcal{X}$ is a sample, $y_n \in \mathcal{Y}$ is the predicted label of individuals $\boldsymbol{x}_n$ from the classifier $\mathcal{H}$. Moreover, f_θ is denoted for a normalizing flow model parameterized by θ. Finally, we split the feature space into two disjoint feature subspaces of categorical features and continuous features represented by $\mathcal{X}^{\text{cat}}$ and $\mathcal{X}^{\text{con}}$ respectively such that $\mathcal{X} = \mathcal{X}_{\text{cat}} \times \mathcal{X}_{\text{con}}$ and $\boldsymbol{x} = (\boldsymbol{x}^{\text{cat}}, \boldsymbol{x}^{\text{con}})$, and $\boldsymbol{x}^{\text{cat}_j}$ and $\boldsymbol{x}^{\text{con}_j}$ is the corresponding j-th feature of $\boldsymbol{x}^{\text{cat}}$ and $\boldsymbol{x}^{\text{con}}$.

6.3.1 Counterfactual Explanation

With the original sample $\boldsymbol{x}_{\text{org}} \in \mathcal{X}$ and its predicted output $y_{\text{org}} \in \mathcal{Y}$, the counterfactual explanation aims to find the nearest counterfactual sample $\boldsymbol{x}_{\text{cf}}$ such that the outcome of classifier for $\boldsymbol{x}_{\text{cf}}$ is changed to desired output class y_{cf}. We aim to identify the perturbation $\boldsymbol{\delta}$ such that counterfactual instance $\boldsymbol{x}_{\text{cf}} = \boldsymbol{x}_{\text{org}} + \boldsymbol{\delta}$ is the solution of the following optimization problem:

$$\boldsymbol{x}_{\text{cf}} = \underset{\boldsymbol{x}_{\text{cf}} \in \mathcal{X}}{\operatorname{argmin}}\, d(\boldsymbol{x}_{\text{cf}}, \boldsymbol{x}_{\text{org}}) \quad \text{subject to} \quad \mathcal{H}(\boldsymbol{x}_{\text{cf}}) = y_{\text{cf}} \tag{6.1}$$

where $d(\boldsymbol{x}_{\text{cf}}, \boldsymbol{x}_{\text{org}})$ is the function measuring the distance between $\boldsymbol{x}_{\text{org}}$ and $\boldsymbol{x}_{\text{cf}}$. Eq (6.1) demonstrates the optimization objective that minimizes the similarity of the counterfactual and original samples, as well as ensures to change the classifier to the desirable outputs. To make the counterfactual explanations plausible, they should only suggest minimal changes in features of the original sample. (Mothilal et al. 2020b).

6.3.2 Normalizing Flow

Normalizing flows (NF) (Dinh et al. 2014) is the active research direction in generative models that aims at modeling the probability distribution of a given dataset. The study (Dinh et al. 2016) first proposes a normalizing flow, which is an unsupervised density estimation model described as an invertible mapping $f_\theta : \mathcal{X} \to \mathcal{Z}$ from the data space $\mathcal{X}$ to the latent space $\mathcal{Z}$. Function f_θ can be designed as a neural network parametrized by θ with architecture that has to ensure invertibility and efficient computation of log-determinants. The data distribution is modeled as a transformation $f_\theta^{-1} : \mathcal{Z} \to \mathcal{X}$ applied to a random variable from the latent distribution $\boldsymbol{z} \sim p_{\mathcal{Z}}$, for which Gaussian distribution is chosen. The change of variables formula gives the density of the converted random variable $\boldsymbol{x} = f_\theta^{-1}(\boldsymbol{z})$ as follows:

$$\begin{aligned} p_{\mathcal{X}}(\boldsymbol{x}) &= p_{\mathcal{Z}}(f_\theta(\boldsymbol{x})) \cdot \left| \det \left(\frac{\partial f_\theta}{\partial \boldsymbol{x}} \right) \right| \\ &\propto \log \left(p_{\mathcal{Z}}(f_\theta(\boldsymbol{x})) \right) + \log \left(\left| \det \left(\frac{\partial f_\theta}{\partial \boldsymbol{x}} \right) \right| \right) \end{aligned} \tag{6.2}$$

With N training data points $\mathcal{D} = \{\boldsymbol{x}_n\}_{n=1}^N$, the model with respects to parameters θ can be trained by maximizing the likelihood in Equation (6.3):

$$\theta =_\theta \left(\prod_{n=1}^{N} \left(\log(p_{\mathcal{Z}}(f_\theta(\boldsymbol{x}_n))) + \log \left(\left| \det \left(\frac{\partial f_\theta(\boldsymbol{x}_n)}{\partial \boldsymbol{x}_n} \right) \right| \right) \right) \right) \tag{6.3}$$

6.4 Methodology

In this section, we illustrate our approach (CeFlow) which leverages the power of normalizing flow in generating counterfactuals. First, we define the general architecture of our framework in section 6.4.1. Thereafter, section 6.4.2 and 6.4.3 illustrate

how to train and build the architecture of the invertible function f for tabular data, while section 6.4.4 describes how to produce the counterfactual samples by adding the perturbed vector into the latent representation.

6.4.1 General architecture of CeFlow

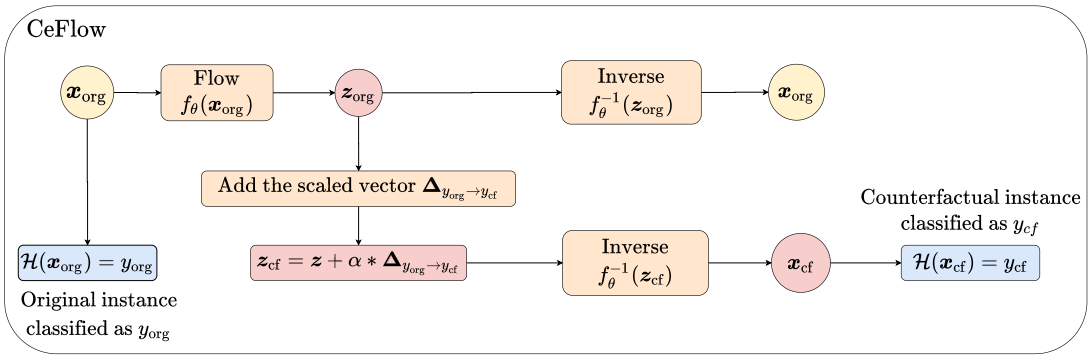


Figure 6.1 : Counterfactual explanation with normalizing flows (CeFlow).

Figure 6.1 generally illustrates our framework. Let $\boldsymbol{x}_{\text{org}}$ be an original instance, and f_θ denote a pre-trained, invertible and differentiable normalizing flow model on the training data. In general, we first construct an invertible and differentiable function f_θ that converts the original instance $\boldsymbol{x}_{\text{org}}$ to the latent representation $\boldsymbol{z}_{\text{org}} = f(\boldsymbol{x}_{\text{org}})$. After that, we would find the scaled vector $\boldsymbol{\delta}_z$ as the perturbation and add to the latent representation $\boldsymbol{z}_{\text{org}}$ to get the perturbed representation $\boldsymbol{z}_{\text{cf}}$ which goes through the inverse function f_θ^{-1} to produce the counterfactual instance $\boldsymbol{x}_{\text{cf}}$. With the counterfactual instance $\boldsymbol{x}_{\text{cf}} = f_\theta^{-1}(\boldsymbol{z}_{\text{org}} + \boldsymbol{\delta}_z)$, we can re-write the objective function Eq. (6.1) into the following form:

$$\begin{cases} \boldsymbol{\delta}_z = \text{argmin}_{\boldsymbol{\delta}_z \in \mathcal{Z}}\, d(\boldsymbol{x}_{\text{org}}, \boldsymbol{\delta}_z) \\ \mathcal{H}(\boldsymbol{x}_{\text{cf}}) = y_{\text{cf}} \end{cases} \tag{6.4}$$

One of the biggest problems of deploying normalizing flow is how to handle mixed-type data which contains both continuous and categorical features. Categorical features are in discrete forms, which is challenging to model by the continuous distribution only (Ho et al. 2019). Another challenge is to construct the objective

function to learn the conditional distribution on the predicted labels (Winkler et al. 2019; Izmailov et al. 2020). In the next section, we will discuss how to construct the conditional normalizing flow f_θ for tabular data.

6.4.2 Normalizing flows for categorical features

This section would discuss how to handle the categorical features. Let $\{\boldsymbol{z}^{\text{cat}_m}\}_{m=1}^{M}$ be the continuous representation of M categorical features $\{\boldsymbol{x}^{\text{cat}_m}\}_{m=1}^{M}$ for each $\boldsymbol{x}^{\text{cat}_m} \in \{0, 1, ..., K-1\}$ with $K > 1$. Follow by several studies in the literature (Ho et al. 2019; Hoogeboom et al. 2020), we utilize variational dequantization to model the categorical features. The key idea of variational dequantization is to add noise $\boldsymbol{u}$ to the discrete values $\boldsymbol{x}^{\text{cat}}$ to convert the discrete distribution $p_{\mathcal{X}^{\text{cat}}}$ into a continuous distribution $p_{\phi_{\text{cat}}}$. With $\boldsymbol{z}^{\text{cat}} = \boldsymbol{x}^{\text{cat}} + \boldsymbol{u}_k$, ϕ_{cat} and θ_{cat} be models' parameters, we have following objective functions:

$$\begin{aligned} \log p_{\mathcal{X}^{\text{cat}}}(\boldsymbol{x}^{\text{cat}}) &\geq \int_{\boldsymbol{u}} \log \frac{p_{\phi_{\text{cat}}}(\boldsymbol{z}^{\text{cat}})}{q_{\theta_{\text{cat}}}(\boldsymbol{u}|\boldsymbol{x}^{\text{cat}})} du \\ &\approx \frac{1}{K} \sum_{k=1}^{K} \log \prod_{m=1}^{M} \frac{p_{\phi_{\text{cat}}}(\boldsymbol{x}^{\text{cat}_m} + \boldsymbol{u}_k)}{q_{\theta_{\text{cat}}}(\boldsymbol{u}_k|\boldsymbol{x}^{\text{cat}})} \end{aligned} \tag{6.5}$$

Followed the study (Hoogeboom et al. 2020), we choose Gaussian dequantization which is more powerful than the uniform dequantization as $q_{\theta_{\text{cat}}}(\boldsymbol{u}_k|\boldsymbol{x}^{\text{cat}}) = \text{sig}\left(\mathcal{N}\left(\boldsymbol{\mu}_{\theta_{\text{cat}}}, \boldsymbol{\Sigma}_{\theta_{\text{cat}}}\right)\right)$ with mean $\boldsymbol{\mu}_{\theta_{\text{cat}}}$, covariance $\boldsymbol{\Sigma}_{\theta_{\text{cat}}}$ and sigmoid function $\text{sig}(\cdot)$.

6.4.3 Conditional Flow Gaussian Mixture Model for tabular data

The categorical features $\boldsymbol{x}^{\text{cat}}$ going through the variational dequantization would convert into continuous representation $\boldsymbol{z}^{\text{cat}}$. We then perform merge operation on continuous representation $\boldsymbol{z}^{\text{cat}}$ and continuous feature $\boldsymbol{x}^{\text{con}}$ to obtain values $(\boldsymbol{z}^{\text{cat}}, \boldsymbol{x}^{\text{con}}) \mapsto \boldsymbol{x}^{\text{full}}$. Thereafter, we apply flow Gaussian mixture model (Izmailov et al. 2020) which is a probabilistic generative model for training the invertible function f_θ. For each predicted class label $y \in \{1...\mathcal{C}\}$, the latent space distribution $p_{\mathcal{Z}}$ conditioned on a label k is the Gaussian distribution $\mathcal{N}\left(\boldsymbol{z}^{\text{full}} \mid \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\right)$ with mean $\boldsymbol{\mu}_k$ and covariance $\boldsymbol{\Sigma}_k$:

$$p_{\mathcal{Z}}(\boldsymbol{z}^{\text{full}} \mid y = k) = \mathcal{N}\left(\boldsymbol{z}^{\text{full}} \mid \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\right) \tag{6.6}$$

As a result, we can have the marginal distribution of $\boldsymbol{z}^{\text{full}}$:

$$p_{\mathcal{Z}}(\boldsymbol{z}^{\text{full}}) = \frac{1}{\mathcal{C}} \sum_{k=1}^{\mathcal{C}} \mathcal{N}\left(\boldsymbol{z}^{\text{full}} \mid \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\right) \tag{6.7}$$

The density of the transformed random variable $\boldsymbol{x}^{\text{full}} = f_\theta^{-1}(\boldsymbol{z}^{\text{full}})$ is given by:

$$p_{\mathcal{X}}(\boldsymbol{x}^{\text{full}}) = \log\left(p_{\mathcal{Z}}(f_\theta(\boldsymbol{x}^{\text{full}}))\right) + \log\left(\left|\det\left(\frac{\partial f_\theta}{\partial \boldsymbol{x}^{\text{full}}}\right)\right|\right) \tag{6.8}$$

Eq. (6.7) and Eq. (6.8) together lead to the likelihood for data as follows:

$$p_{\mathcal{X}}(\boldsymbol{x}^{\text{full}} \mid y = k) = \log\left(\mathcal{N}\left(f_\theta(\boldsymbol{x}^{\text{full}}) \mid \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\right)\right) + \log\left(\left|\det\left(\frac{\partial f_\theta}{\partial \boldsymbol{x}^{\text{full}}}\right)\right|\right) \tag{6.9}$$

We can train the model by maximizing the joint likelihood of the categorical and continuous features on N training data points $\mathcal{D} = \{(\boldsymbol{x}_n^{\text{con}}, \boldsymbol{x}_n^{\text{cat}})\}_{n=1}^N$ by combining Eq. (6.5) and Eq. (6.9):

$$\begin{aligned}
\theta^*, \phi_{\text{cat}}^*, \theta_{\text{cat}}^* &= _{\theta,\phi_{\text{cat}},\theta_{\text{cat}}} \prod_{n=1}^{N} \left(\prod_{\boldsymbol{x}_n^{\text{con}} \in \mathcal{X}^{\text{con}}} p_{\mathcal{X}}\left(\boldsymbol{x}_n^{\text{con}}\right) \prod_{\boldsymbol{x}_n^{\text{cat}} \in \mathcal{X}^{\text{cat}}} p_{\mathcal{X}}\left(\boldsymbol{x}_n^{\text{cat}}\right) \right) \\
&= _{\theta,\phi_{\text{cat}},\theta_{\text{cat}}} \prod_{n=1}^{N} \left(\log\left(\mathcal{N}\left(f_\theta(\boldsymbol{x}_n^{\text{full}}) \mid \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\right)\right) + \log\left(\left|\det\left(\frac{\partial f_\theta}{\partial \boldsymbol{x}_n^{\text{full}}}\right)\right|\right) \right)
\end{aligned} \tag{6.10}$$

6.4.4 Counterfactual generation step

In order to find counterfactual samples, the recent approaches (Mothilal et al. 2020b; Wachter et al. 2017) normally define the loss function and deploy some optimization algorithm such as gradient descent or heuristic search to find the perturbation. These approaches however demonstrates the prohibitively slow running time, which prevents from deploying in interactive environment(Höltgen et al. 2021). Therefore, inspired by the study (Hvilshøj et al. 2021), we add the scaled vector as the perturbation from the original instance $\boldsymbol{x}_{\text{org}}$ to counterfactual one $\boldsymbol{x}_{\text{cf}}$. By Bayes' rule, we notice that under a uniform prior distribution over labels $p(y = k) = \frac{1}{\mathcal{C}}$ for $\mathcal{C}$ classes, the log posterior probability becomes:

$$\log p_{\mathcal{X}}(y = k|\boldsymbol{x}) = \log \frac{p_{\mathcal{X}}(\boldsymbol{x}|y = k)}{\sum_{k=1}^{\mathcal{C}} p_{\mathcal{X}}(\boldsymbol{x}|y = k)} \propto ||f_\theta(\boldsymbol{x}) - \boldsymbol{\mu}_k||^2 \tag{6.11}$$

We observed from Eq. (6.11) that latent vector $\boldsymbol{z} = f_\theta(\boldsymbol{x})$ will be predicted from the class y with the closest model mean $\boldsymbol{\mu}_k$. For each predicted class $k \in \{1...\mathcal{C}\}$,

we denote $\mathcal{G}_k = \{\boldsymbol{x}_m, y_m\}_{m=1}^{M}$ as a set of M instances with the same predicted class as $y_m = k$. We define the mean latent vector $\boldsymbol{\mu}_k$ corresponding to each class k such that:

$$\boldsymbol{\mu}_k = \frac{1}{M} \sum_{\boldsymbol{x}_m \in \mathcal{G}_k} f_\theta(\boldsymbol{x}_m) \tag{6.12}$$

Therefore, the scaled vector that moves the latent vector $\boldsymbol{z}_{\text{org}}$ to the decision boundary from the original class y_{org} to counterfactual class y_{cf} is defined as:

$$\boldsymbol{\Delta}_{y_{\text{org}} \to y_{\text{cf}}} = \left| \boldsymbol{\mu}_{y_{\text{org}}} - \boldsymbol{\mu}_{y_{\text{cf}}} \right| \tag{6.13}$$

The scaled vector $\boldsymbol{\Delta}_{y_{\text{org}} \to y_{\text{cf}}}$ is added to the original latent representation $\boldsymbol{z}_{\text{cf}} = f_\theta(\boldsymbol{x}_{\text{org}})$ to obtained the perturbed vector. The perturbed vector then goes through inverted function f_θ^{-1} to re-produce the counterfactual sample:

$$\boldsymbol{x}_{\text{cf}} = f_\theta^{-1}(f_\theta(\boldsymbol{x}_{\text{org}}) + \alpha \boldsymbol{\Delta}_{y_{\text{org}} \to y_{\text{cf}}}) \tag{6.14}$$

We note that the hyperparameter α needs to be optimized by searching in a range of values. The full algorithm is illustrated in Algorithm 6.1.

Algorithm 6.1: Counterfactual explanation flow (CeFlow)

Input: An original sample $\boldsymbol{x}_{\text{org}}$ with its prediction y_{org}, desired class y_{cf}, a provided machine learning classifier $\mathcal{H}$ and encoder model Q_ϕ.

1: Train the invertible function f_θ by maximizing the log-likelihood:

$$\begin{aligned}
\theta^*, \phi^*_{\text{cat}}, \theta^*_{\text{cat}} &= _{\theta, \phi_{\text{cat}}, \theta_{\text{cat}}} \prod_{n=1}^{N} \left(\prod_{\boldsymbol{x}_n^{\text{con}} \in \mathcal{X}^{\text{con}}} p_{\mathcal{X}} \left(\boldsymbol{x}_n^{\text{con}} \right) \prod_{\boldsymbol{x}_n^{\text{cat}} \in \mathcal{X}^{\text{cat}}} p_{\mathcal{X}} \left(\boldsymbol{x}_n^{\text{cat}} \right) \right) \\
&= _{\theta, \phi_{\text{cat}}, \theta_{\text{cat}}} \prod_{n=1}^{N} \left(\log \left(\mathcal{N} \left(f_\theta(\boldsymbol{x}_n^{\text{full}}) \mid \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k \right) \right) + \log \left(\left| \det \left(\frac{\partial f_\theta}{\partial \boldsymbol{x}_n^{\text{full}}} \right) \right| \right) \right)
\end{aligned}$$

2: Compute mean latent vector $\boldsymbol{\mu}_k$ for each class k by $\boldsymbol{\mu}_k = \frac{1}{M} \sum_{\boldsymbol{x}_m \in \mathcal{G}_k} f(\boldsymbol{x}_m)$.

3: Compute the scaled vector $\boldsymbol{\Delta}_{y_{\text{org}} \to y_{\text{cf}}} = \left| \boldsymbol{\mu}_{y_{\text{org}}} - \boldsymbol{\mu}_{y_{\text{cf}}} \right|$.

4: Find the optimal hyperparameter α by searching a range of values.

5: Compute $\boldsymbol{x}_{\text{cf}} = f_\theta^{-1}(f_\theta(\boldsymbol{x}_{\text{org}}) + \alpha \boldsymbol{\Delta}_{y_{\text{org}} \to y_{\text{cf}}})$.

Output: $\boldsymbol{x}_{\text{cf}}$.

6.5 Experiments

We run experiments on three datasets to show that our method outperforms state-of-the-art approaches. The specification of hardware for the experiment is Python 3.8.5 with 64-bit Red Hat, Intel(R) Xeon(R) Gold 6238R CPU @ 2.20GHz. We implement our algorithm by using Pytorch library and adopt the RealNVP architecture (Dinh et al. 2016). During training progress, Gaussian mixture parameters are fixed: the means are initialized randomly from the standard normal distribution and the covariances are set to I. More details of implementation settings can be found in our code repository[†].

We evaluate our approach via three datasets: `Law` (Wightman 1998), `Compas` (Larson et al. 2016) and `Adult` (Dua and Graff 2017). `Law`[‡](Wightman 1998) dataset provides information of students with their features: their entrance exam scores (LSAT), grade-point average (GPA) and first-year average grade (FYA). `Compas`[§](Larson et al. 2016) dataset contains information about 6,167 prisoners who have features including gender, race and other attributes related to prior conviction and age. `Adult`[¶](Dua and Graff 2017) dataset is a real-world dataset consisting of both continuous and categorical features of a group of consumers who apply for a loan at a financial institution.

We compare our proposed method (CeFlow) with several state-to-the-art methods including Actionable Recourse (AR) (Ustun et al. 2019), Growing Sphere (GS) (Laugel et al. 2017), FACE (Poyiadzi et al. 2020), CERTIFAI (Sharma et al. 2020), DiCE (Mothilal et al. 2020b) and C-CHVAE (Pawelczyk et al. 2020). Particularly, we implement the CERTIFAI with library PyGAD[‖] and utilize the available source code[**] for implementation of DiCE, while other approaches are implemented with

[†] https://anonymous.4open.science/r/fairCE-538B

[‡] http://www.seaphe.org/databases.php

[§] https://www.propublica.org

[¶] https://archive.ics.uci.edu/ml/datasets/adult

[‖] https://github.com/ahmedfgad/GeneticAlgorithmPython

[**] https://github.com/divyat09/cf-feasibility

Carla library (Pawelczyk et al. 2021). Finally, we report the results of our proposed model on a variety of metrics including success rate (success), l_1-norm (l_1), categorical proximity (Mothilal et al. 2020b), continuous proximity (Mothilal et al. 2020b) and mean log-density (Artelt and Hammer 2020). Note that for l_1-norm, we report mean and variance of l_1-norm corresponding to l_1-mean and l_1-variance. Lower l_1-variance aims to illustrate the algorithm's robustness.

Table 6.1 : Performance of all methods on the classifier. We compute p-value by conducting a paired t-test between our approach (CeFlow) and baselines with 100 repeated experiments for each metric.

Dataset	Method	Performance				p-value		
		success	l_1-mean	l_1-var	log-density	success	l_1	log-density
Law	AR	98.00	3.518	2.0e-03	-0.730	0.041	0.020	0.022
	GS	100.00	3.600	2.6e-03	-0.716	0.025	0.048	0.016
	FACE	100.00	3.435	2.0e-03	-0.701	0.029	0.010	0.017
	CERTIFAI	100.00	3.541	2.0e-03	-0.689	0.029	0.017	0.036
	DiCE	94.00	**3.111**	2.0e-03	-0.721	0.018	0.035	0.048
	C-CHVAE	100.00	3.461	1.0e-03	-0.730	0.040	0.037	0.016
	CeFlow	**100.00**	3.228	**1.0e-05**	**-0.679**	-	-	-
Compas	AR	97.50	1.799	2.4e-03	-14.92	0.038	0.034	0.046
	GS	100.00	1.914	3.2e-03	-14.87	0.019	0.043	0.040
	FACE	98.50	1.800	4.8e-03	-15.59	0.036	0.024	0.035
	CERTIFAI	100.00	1.811	2.4e-03	-15.65	0.040	0.048	0.038
	DiCE	95.50	1.853	2.9e-03	-14.68	0.030	0.029	0.018
	C-CHVAE	100.00	1.878	1.1e-03	-13.97	0.026	0.015	0.027
	CeFlow	**100.00**	**1.787**	**1.8e-05**	**-13.62**	-	-	-
Adult	AR	100.00	3.101	7.8e-03	-25.68	0.044	0.037	0.018
	GS	100.00	3.021	2.4e-03	-26.55	0.026	0.049	0.028
	FACE	100.00	2.991	6.6e-03	-23.57	0.027	0.015	0.028
	CERTIFAI	93.00	3.001	4.1e-03	-25.55	0.028	0.022	0.016
	DiCE	96.00	2.999	9.1e-03	-24.33	0.046	0.045	0.045
	C-CHVAE	100.00	3.001	8.7e-03	-24.45	0.026	0.043	0.019
	CeFlow	**100.00**	**2.964**	**1.5e-05**	**-23.46**	-	-	-

Table 6.2 : We report running time of different methods on three datasets.

Dataset	AR	GS	FACE	CERTIFAI	DiCE	C-CHVAE	CeFlow
Law	3.030 ± 0.105	7.126 ± 0.153	6.213 ± 0.007	6.522 ± 0.088	8.022 ± 0.014	9.022 ± 0.066	**0.850 ± 0.055**
Compas	5.125 ± 0.097	8.048 ± 0.176	7.688 ± 0.131	13.426 ± 0.158	7.810 ± 0.076	6.879 ± 0.044	**0.809 ± 0.162**
Adult	7.046 ± 0.151	6.472 ± 0.021	13.851 ± 0.001	7.943 ± 0.046	11.821 ± 0.162	12.132 ± 0.024	**0.837 ± 0.026**

The performance of different approaches regarding three metrics: l_1, success metrics and log-density are illustrated in Table 6.1. Regarding success rate, all three

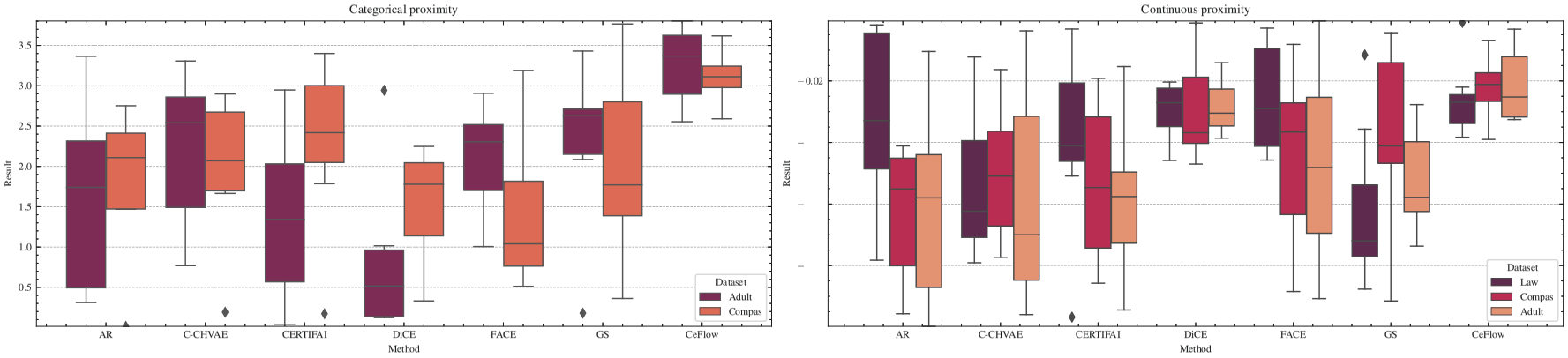


Figure 6.2 : Baseline results in terms of **Categorical proximity** and **Continuous proximity**. Higher continuous and categorical proximity are better.

methods achieve competitive results, except the AR, DiCE and CERTIFAI performance in all datasets with around 90% of samples belonging to the target class. These results indicate that by integrating normalizing flows into counterfactuals generation, our proposed method can achieve the target of counterfactual explanation task for changing the models' decision. Apart from that, for l_1-mean, CeFlow is ranked second with 3.228 for `Law`, and is ranked first for `Compas` and `Adult` (1.787 and 2.964). Moreover, our proposed method generally achieves the best performance regarding l_1-variance on three datasets. CeFlow also demonstrates the lowest log-density metric in comparison with other approaches achieving at -0.679, -13.62 and -23.46 corresponding to `Law`, `Compas` and `Adult` dataset. This illustrates that the generated samples are more closely followed the distribution of data than other approaches. We furthermore perform a statistical significance test to gain more insights into the effectiveness of our proposed method in producing counterfactual samples compared with other approaches. Particularly, we conduct the paired t-test between our approach (CeFlow) and other methods on each dataset and each metric with the obtained results on 100 randomly repeated experiments and report the result of p-value in Table 6.1. We discover that our model is statistically significant with $p < 0.05$, proving CeFlow's effectiveness in counterfactual samples generation tasks. Meanwhile, Table 6.2 shows the running time of different approaches. Our approach achieves outstanding performance with the running time demonstrating around 90% reduction compared with other approaches. Finally, as expected, by using normalizing flows, CeFlow produces more robust counterfactual samples with the lowest l_1-variance and demonstrates an effective running time in comparison with other approaches.

Figure 6.2 illustrates the categorical and continuous proximity. In terms of categorical proximity, our approach achieves the second-best performance with lowest variation in comparison with other approaches. The heuristic search based algorithm such as FACE and GS demonstrate the best performance in terms of this metric. Meanwhile, DiCE produces the best performance for continuous proximity, whereas CeFlow is ranked second. In general, our approach (CeFlow) achieves competitive

performance in terms of proximity metric and demonstrates the least variation in comparison with others. On the other hand, Figure 6.3 shows the variation of our method's performance with the different values of α. We observed that the optimal values are achieved at 0.8, 0.9 and 0.3 for `Law`, `Compas` and `Adult` dataset, respectively.

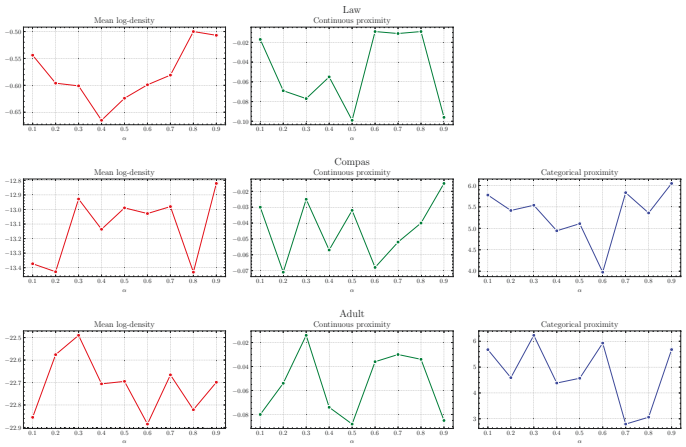


Figure 6.3 : Our performance under different values of hyperparameter α. Note that there are no categorical features in `Law` dataset.

6.6 Conclusion

In this paper, we introduced a robust and efficient counterfactual explanation framework called CeFlow that utilizes the capacity of normalizing flows in generating counterfactual samples. We observed that our approach produces more stable counterfactual samples and reduces counterfactual generation time significantly. The better performance witnessed is likely because that normalizing flows can get the exact representation of the input instance and also produce the counterfactual samples by using the inverse function. Numerous extensions to the current work can be investigated upon successful expansion of normalizing flow models in interpretable machine learning in general and counterfactual explanation in specific. One potential direction is to design a normalizing flow architecture to achieve counterfactual fairness in machine learning models.

Part IV

Counterfactual fairness

Part IV embarks on a comprehensive exploration of the pivotal concept of counterfactual fairness for algorithmic decision-making. Chapter 7 introduces and thoroughly examine the concept of counterfactual fairness, and proposed an approach to achieve counterfactual fairness with imperfect structural causal model. This chapter not only identifies these challenges but also proposes a novel and innovative methodology that strategically harnesses the untapped potential of imperfect structural causal models (SCMs). This pioneering approach signifies a significant leap forward in our collective endeavor to achieve fairness in machine learning, offering a promising pathway for future exploration and widespread implementation.

Chapter 7

Achieving Counterfactual Fairness with Imperfect Structural Causal Model

Counterfactual fairness alleviates the discrimination between the model prediction toward an individual in the actual world (observational data) and that in counterfactual world (i.e., what if the individual belongs to other sensitive groups). The existing studies need to pre-define the structural causal model that captures the correlations among variables for counterfactual inference; however, the underlying causal model is usually unknown and difficult to be validated in real-world scenarios. Moreover, the misspecification of the causal model potentially leads to poor performance in model prediction and thus makes unfair decisions. In this chapter, we propose a novel minimax game-theoretic model for counterfactual fairness that can produce accurate results meanwhile achieve a counterfactually fair decision with the relaxation of strong assumptions of structural causal models. In addition, we also theoretically prove the error bound of the proposed minimax model. Empirical experiments on multiple real-world datasets illustrate our superior performance in both accuracy and fairness. Source code is available at https://github.com/tridungduong16/counterfactual_fairness_game_theoretic. The main material of this chapter is derived from the following reference:

1. **Duong, T. D.**, Li, Q., & Xu, G. (2022) Achieving Counterfactual Fairness with Imperfect Structural Causal Model (Under review for Knowledge-based System).

7.1 Introduction

As machine learning (ML) is increasingly leveraged in high-stake domains such as criminal justice (Berk et al. 2021) or credit assessment (Zhang and Zhou 2019), the

concerns regarding ethical issues in designing ML algorithms have arisen recently. Fairness is one of the most important concerns to avoid discrimination in the model prediction towards an individual or a population. Recent years witness an increasing number of studies that have explored fairness-aware machine learning under the causal perspective (Nabi and Shpitser 2018; Kusner et al. 2017; Zhang and Bareinboim 2018; Chiappa 2019). Causal models specifically (Pearl et al. 2009) provide an intuitive and powerful way of reasoning the causal effect of sensitive attribution on the final decision. Among these studies in this line of work, counterfactual fairness is a causal and individual-level fairness notion first proposed by (Kusner et al. 2017), which considers a model counterfactually fair if its predictions are identical in both a) the original world and b) the counterfactual world where an individual belongs to another demographic group.

As the first practice of counterfactual fairness, (Kusner et al. 2017) first constructs a structural causal model using prior domain knowledge. Unobserved variables are then inferred which are independent of and have no causal relationship to the sensitive attributes. The inferred latent variables are thereafter used as the input for the predictive models. The main limitation of the study is that the strong assumption of the causal model is required which is however hard to achieve in a real-world setting, especially when it comes to a large-scale dataset with a great number of features (VanderWeele 2009; Peters et al. 2016). Additionally, even if prior knowledge of causal structure is available, counterfactual fairness algorithms involves computing counterfactuals in the true underlying structural causal model (SCM) (Pearl 2009b), and thus relies on strong impractical assumptions. Specifically, the algorithm requires complete knowledge of the true structural equations (Fong 2013; Bollen and Pearl 2013; Pearl 2012). Another obstacle is that the tabular data contains both continuous and categorical data, making them difficult to be represented by the probabilistic equations. Moreover, when removing all other features and only using non-descendants of sensitive ones, there are possibly insufficient features used for model training which can degrade the model capability and significantly deteriorate the accuracy performance.

To tackle the above limitations, we propose a novel counterfactual fairness approach with the knowledge about structural causal models is limited. In particular, we aim to minimize the sensitive information impact on model decisions, while maintaining satisfactory model accuracy. To achieve the optimal solutions that maximize the fairness-accuracy trade-offs, we propose a minimax game-theoretic approach that consists of three main components. As shown in Figure 7.1, the invariant-encoder model p_θ learns the invariant representation that is unchangeable from sensitive attributes. After that, the fair-learning predictive model utilizes the invariant representation as the input with the purpose of not only guaranteeing the main learning tasks but also assuring the fairness aspect, while sensitive-awareness model used both the invariant representation and sensitive information that can produce the good learning performance. For theoretical proof, we provide a theoretical analysis for the generalization bound of the minimax objective functions. To illustrate the effectiveness of our proposed method, we compare our method with state-of-the-art methods on three benchmark datasets including `Law`, `Compas` and `Adult` datasets. The experimental results indicate that our proposed method can achieve outstanding fairness performance in comparison with other baselines. Specifically, our contributions can be summarized as follows:

- We introduce a minimax game-theoretic approach to obtain the invariant-encoder model and fair-learning predictive model that can jointly produce the counterfactually fair prediction and obtain the competitive performance on both classification and regression tasks.
- We prove the theoretical generalization bounds for the adversarial algorithm of the proposed minimax model.
- We perform the extensive experiments on three datasets and demonstrate the effectiveness of the proposed method to achieve satisfactory fairness and accuracy.

7.2 Preliminaries

In this section, we provide notations and problem statements and then review individual and counterfactual fairness notions.

Throughout the paper, upper-cased letters X and $\boldsymbol{X}$ represent the random scalars and vectors respectively, while lower-cased letters x and $\boldsymbol{x}$ denote the deterministic scalars and vectors, respectively. We consider a dataset $\mathcal{D} = \{x_i, s_i, y_i\}_{i=1}^n$ consisting of n instances, where $x_i \in \boldsymbol{X}$ is the normal features (e.g. age, working hours,..), $s_i \in \boldsymbol{S}$ is the sensitive feature (e.g. race and gender), and $y_i \in Y$ is the target variable regarding individuals i. Sensitive features specify an individual's belongings to socially salient groups (e.g. women and Asian). $H(.)$ and $I(.)$ are the corresponding Shannon entropy and mutual information (Cover et al. 1991), and $\mathcal{L}(.)$ is the loss function (e.g. cross-entropy for classification tasks, mean square error for regression tasks). Finally, f_θ represents a neural network model parameterized by θ.

Figure 7.1 generally illustrates our proposed approach that consists of an invariant-encoder model (q_θ) generating the invariant features, a fair-learning predictor (f_{ϕ_1}) trained by invariant features and a sensitive-aware predictor (f_{ϕ_2}) trained on invariant and sensitive representation.

Definition 7.1 (Counterfactual fairness (Kusner et al. 2017)). *A classifier is considered as counterfactual fair given the sensitive attribute $S = s$ if:*

$$P(\hat{Y}_{S\leftarrow s} = y|X = x, S = s) = P(\hat{Y}_{S\leftarrow \hat{s}} = y|X = x, S = \hat{s}) \tag{7.1}$$

where $\hat{Y}$ denotes the model prediction depends on X and S, while model prediction for intervention $S \leftarrow \hat{s}$ is denoted as $\hat{Y}_{S\leftarrow s}$. Meanwhile, $P(\hat{Y}_{S\leftarrow \hat{s}} = y|X = x, S = \hat{s})$ is the counterfactual prediction where we change the value $S = s$ to $S = \hat{s}$.

The Eq. (7.1) ensures that the distribution over possible predictions is the same in both the actual world and a counterfactual world where the sensitive attribute(s) were modified while all other conditions remain unchanged.

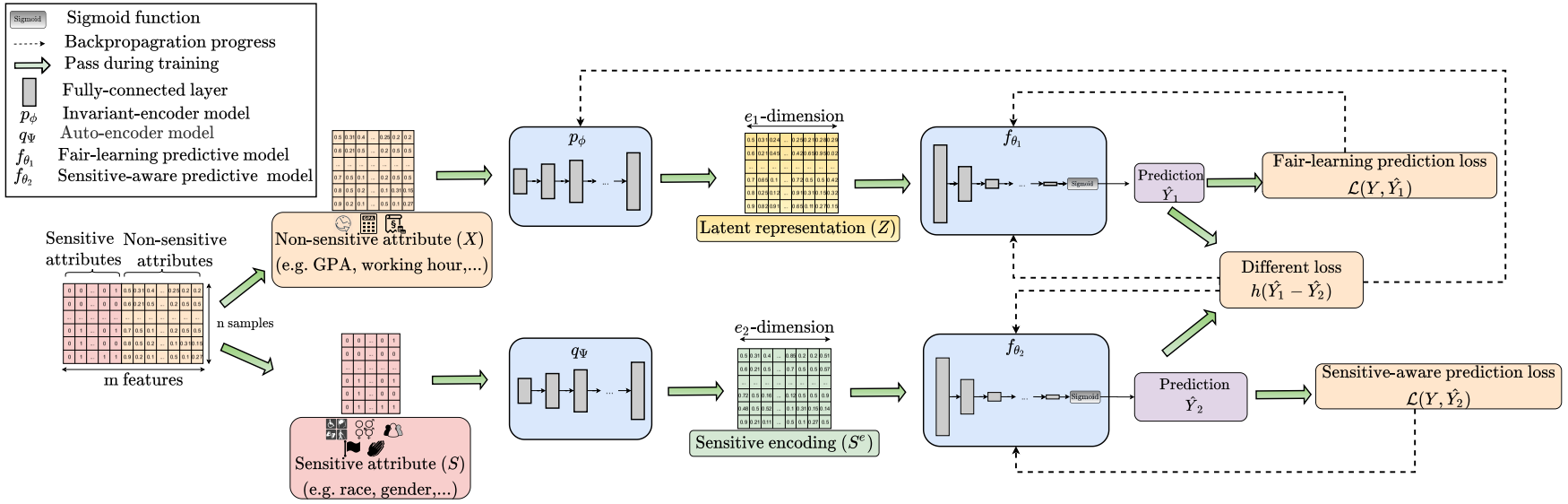


Figure 7.1 : The framework consists of three trainable components: the invariant-encoder model p_ϕ, fair-learning model f_{ϕ_1} and sensitive-awareness f_{ϕ_2} model.

7.3 Related work

This section focuses on the research related to our work and then highlights the main limitation of these studies.

Individual fairness. Since counterfactual fairness analyses fairness at the individual level, our work is closely related to individual fairness works. (Dwork et al. 2012) first captures the main idea of individual fairness that two individuals having the same particular task should be treated similarly. This principle draws much attention with a plethora of studies (Miconi 2017; Biega et al. 2018; Mukherjee et al. 2020; Sharifi-Malvajerdi et al. 2019). However, this concept is hard to apply in practice due to the barrier of defining the similarity regarding the individual tasks. This leads to the shift from achieving fairness decisions to defining similar tasks. Another recent study (Speicher et al. 2018) provides a unified approach to evaluate the performance of individual fairness algorithms by using a generalized entropy index that has been previously used widely in economics as a measure of income inequality in the population. Our work utilizes the generalized entropy index as the primary metric for evaluation purposes.

Counterfactual fairness. In order to achieve fairness in the model decision, the traditional approach is the unawareness model (Grgic-Hlaca et al. 2016) that only uses the non-sensitive attributes as the input for predictive models. This approach seems to be reasonable but neglects the biased effect of sensitive attributes on normal features. Thus, the study (Kusner et al. 2017) first proposed the approach of counterfactual fairness by only using the non-descendants of the sensitive attribute for prediction tasks. They first assume the causal graph structure with the latent variables independent from the sensitive attributes. The study thereafter fits the data into the causal model and produces the posterior distribution for unobserved variables. The inferred variables are finally utilized as inputs for the predictive model. Apart from that, multi-world counterfactual fairness (Russell et al. 2017) introduces another alternative method to deal with the uncertainty of the ground-truth causal model. They first have an assumption that there are several

possible causal diagrams that represent different counterfactual worlds. Thereafter, the authors build a neural network and then use the gradient descent algorithm to minimize the difference in the predictions between the different worlds. Although this approach seems to be promising, it also needs a list of causal models to be taken into consideration. To sum up, all of the above methods require strong assumptions about causal graphs to infer the latent variables. Moreover, sensitive attributes such as race, gender, and nationality are personally intrinsic attributes and immensely influential that normally have a causal relationship to other features.

7.4 Methodology

This section illustrates our proposed method, which can achieve counterfactual fairness without the assumption of structural causal models. In summary, our proposed method aims to learn a representation along with a predictive model which together can make a counterfactual fair prediction and maintain the prediction accuracy. In summary, our proposed approach contains three main components: 1) invariant-encoder model learning the invariant representation that is unchangeable from sensitive attributes; 2) fair-learning predictive model which not only guarantees the main learning tasks but also assures the fairness aspect; 3) sensitive-awareness model that contains the sensitive information which can produce the good learning performance. Each component would be discussed in detail in Section 7.4.2.

7.4.1 Motivation

We first consider an example of probabilistic graphical model in Figure 7.2, we have two sensitive features: $\boldsymbol{S_1}$ and $\boldsymbol{S}_2$, non-sensitive features $\boldsymbol{X}_1$, $\boldsymbol{X}_2$, and $\boldsymbol{X}_3$, and the target variable Y. We want to find a latent representation $\boldsymbol{Z}$ that is independent of the sensitive attributes. Producing the latent representation that is invariant across different sensitive attributes is a challenging task. However, we can handle this challenge if we have information about sensitive attributes. We make an assumption that the probability $p(Y|\boldsymbol{Z})$ remains the same across different sensitive attributes $\boldsymbol{S}_1$ and $\boldsymbol{S}_2$ because $\boldsymbol{Z}$ has the direct causal relationship to target variable Y and also does not rely on $\boldsymbol{S}_1$ and $\boldsymbol{S}_2$. Meanwhile, other remaining features $\boldsymbol{X}_1$,

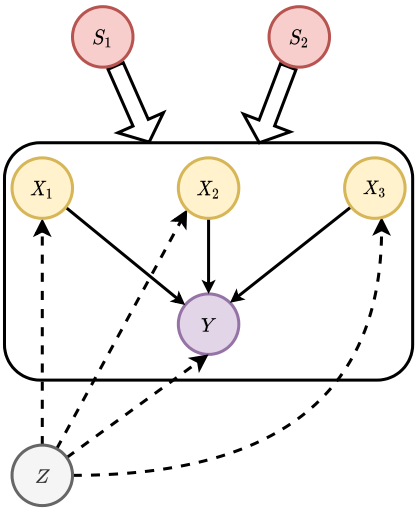


Figure 7.2 : A structural causal model illustrates the causal relationships between different features. $\boldsymbol{S}_1$ and $\boldsymbol{S}_2$ are sensitive features (e.g., gender or race), $\boldsymbol{X}_1$, $\boldsymbol{X}_2$ and $\boldsymbol{X}_3$ are the non-sensitive features (e.g., education or working hours), $\boldsymbol{Z}$ is a latent representation that is independent of sensitive attributes and Y is the target variable. The large white arrows from $\boldsymbol{S}_1$ and $\boldsymbol{S}_2$ represent that $\boldsymbol{S}_1$ and $\boldsymbol{S}_2$ have the causal effects to every variables ($\boldsymbol{X}_1$, $\boldsymbol{X}_2$, $\boldsymbol{X}_3$) and target variable (Y) contained in the box.

$\boldsymbol{X}_2$ and $\boldsymbol{X}_3$ are causally influenced by $\boldsymbol{S}_1$ and $\boldsymbol{S}_2$; thus, the probability of $p(Y|\boldsymbol{X}_1)$, $p(Y|\boldsymbol{X}_2)$ and $p(Y|\boldsymbol{X}_3)$ will change if we change the value of $\boldsymbol{S}_1$ and $\boldsymbol{S}_2$.

In general, our main purpose is to find a representation ($\boldsymbol{Z}$) that is invariant across different sensitive attributes. We want to design an invariant-encoder model $p_\theta : \boldsymbol{X} \rightarrow \boldsymbol{Z}$ that learns the representation ($\boldsymbol{Z}$) from the input ($\boldsymbol{X}$). An ideal invariant representation should satisfy the following conditions.

$$Y \perp do(\boldsymbol{S})|\boldsymbol{Z} \iff H(Y|\boldsymbol{Z}, do(\boldsymbol{S})) = H(Y|\boldsymbol{Z}) \tag{7.2}$$

where $\perp$ denotes probabilistic independence. Y is independent of $do(\boldsymbol{S})$ only when conditioned on latent representation $\boldsymbol{Z}$ and we call this variance property. Moreover, the Eq. (7.2) means that the representation ($\boldsymbol{Z}$) is unchangeable, and sensitive attributes ($\boldsymbol{S}$) do not provide extra information to predict target variables

(Y). This means that making an intervention on sensitive attribute $(\boldsymbol{S})$ does not lead to changes in the model prediction (Y).

7.4.2 Three-player model for invariant fairness

Our proposed framework is illustrated in Figure 7.1 which describes the training and backpropagation process as well as inputs and different components. In general, the proposed framework has three main trainable models including an invariant-encoder model (q_θ) that generates the invariant features, a fair-learning predictor (f_{ϕ_1}) that predicts outcomes based on invariant features and a sensitive-aware predictor (f_{ϕ_2}) that predicts outcomes based on both invariant and sensitive representation. The framework also includes a pre-trained auto-encoder model (q_ψ) that produces latent representation from sensitive features. For clarity, we will briefly describe the auto-encoder model and then present the two predictors followed by the invariant-encoder model.

Auto-encoder model. Sensitive attributes $(\boldsymbol{S})$ are in the categorical form and discrete values, which is hard to utilize in neural networks. Therefore, we construct the auto-encoder model (q_ψ) with the purpose of 1) converting discrete values to continuous form which is more suitable to the complicated models, 2) capturing the intrinsic relationship between categorical groups, and flexibly control the dimensional number of embedding vector. The auto-encoder model (q_ψ)(Ng et al. 2011) is trained beforehand by using all of the features as the input. The encoder-decoder architecture with an embedding layer is used that aims to project sensitive attributes $\boldsymbol{S}$ onto an e-dimensional latent space $\mathcal{R}^e$ $(q_\psi : \boldsymbol{S} \rightarrow \boldsymbol{S}^e)$. The latent representation of sensitive attributes $(\boldsymbol{S}^e)$ would be thereafter utilized to be injected into the sensitive-aware predictor later.

Two predictors. The fair-learning predictor $f_{\phi_1} : \boldsymbol{Z} \rightarrow Y$ that predicts target variable (Y) from the latent representation $(\boldsymbol{Z})$. Meanwhile, the sensitive-aware predictive model $f_{\phi_2} : (\boldsymbol{Z}, \boldsymbol{S}^e) \rightarrow Y$ makes a prediction (Y) from latent representation $(\boldsymbol{Z})$ and sensitive attributes information $(\boldsymbol{S}^e)$. The only difference between them is that the sensitive-aware predictor can access to sensitive information, while the fair-

learning one only uses the invariant representation. The loss functions for the fair-learning and sensitive-aware predictive model are $\mathcal{L}(Y, f_{\phi_1}(\boldsymbol{Z}))$ and $\mathcal{L}(Y, f_{\phi_2}(\boldsymbol{Z}, \boldsymbol{S}^e))$, respectively. Thus, the optimal solutions for both of them can be definedas follows:

$$\phi_i^* = \underset{\phi_1}{\operatorname{argmin}}\, \mathbb{E}[\mathcal{L}(Y, f_{\phi_1}(\boldsymbol{Z}))] \tag{7.3}$$

$$\phi_2^* = \underset{\phi_2}{\operatorname{argmin}}\, \mathbb{E}[\mathcal{L}(Y, f_{\phi_2}(\boldsymbol{Z}, \boldsymbol{S}^e))] \tag{7.4}$$

Invariant-encoder model. The invariant-encoder model $q_\theta : \boldsymbol{X} \to \boldsymbol{Z}$ learns the representation ($\boldsymbol{Z}$) from the input ($\boldsymbol{X}$). The aims of the invariant-encoder model are first to optimize the fair-learning predictive model, and then to minimize the gap between the predictions of two predictive models. This allows to ensure the model accuracy in the prediction task and excludes sensitive information in model decisions. Therefore, the learning objective for the invariant-encoder model is:

$$\theta^* = \underset{\theta}{\operatorname{argmin}}\, \mathcal{L}(Y, f_{\phi_1^*}(\boldsymbol{Z})) + \lambda h(f_{\phi_1^*}(\boldsymbol{Z}) - f_{\phi_2^*}(\boldsymbol{Z}, \boldsymbol{S}^e)) \tag{7.5}$$

where $h(t)$ is a strictly monotonic function that increases when $t > 0$, and decreases when $t < 0$.

Objective function and training. By combining learning objectives from Eq. (7.3), (7.4) and (7.5), we can produce the final objective function in the minimax form Eq. (7.6) with the latent representation $\boldsymbol{Z} = q_\theta(\boldsymbol{X})$. Overall, the loss function represents the minimax game where the invariant-encoder model plays cooperative games with the fair-learning predictor and adversarial games with the sensitive-aware predictor. The objective functions first aims to minimize the prediction of function f_{ϕ_1} and make the gap between f_{ϕ_2} and f_{ϕ_1} as small as possible.

$$\underset{\theta,\phi_1}{\operatorname{argmin}}\, _{\phi_2} \mathcal{L}(Y, f_{\phi_1}(\boldsymbol{Z})) + \lambda h(f_{\phi_1}(\boldsymbol{Z}) - f_{\phi_2}(\boldsymbol{Z}, \boldsymbol{S}^e)) \tag{7.6}$$

Regarding the training process for three models, the loss functions corresponding to each model are first calculated. We thereafter update each model by descending stochastic gradients regarding invariant-encoder model, fair-learning predictor and ascending stochastic gradient of sensitive-aware predictor. We perform updating procedure with a number of steps, and only one model is updated for each step. In

our experiments, we used Adam optimization algorithm (Kingma and Ba 2014) to optimize (7.6).

7.5 Theoretical analysis for Three-player model

This section provides the generalization bound for our proposed method under the minimax setting. Remember we consider the local minimax empirical risk minimization problem

$$\min_{\theta,\phi_1}\max_{\phi_2}\mathbb{E}[\mathcal{L}\left(Y, f_{\phi_1}(\boldsymbol{Z})\right)] \tag{7.7}$$

By applying a duality argument, we reformulate the dual problem via the probability of sensitive attributes. Let P^* be the ideal fair sample distribution corresponding to P/Q_0, according to the underlying exposure mechanism Q_0 and data distribution P. We choose the Wasserstein distance to investigate how to transport from the observed data distribution to an ideal data distribution that is independent of the sensitive attributes. The reason is that unlike the Kullback-Leibler divergence, the Wasserstein metric is a true probability metric and considers both the probability of and the distance between various outcome events. Wasserstein distance provides a meaningful and smooth representation of the distance between distributions.The Wasserstein Distance is furthermore to measure distances between probability distributions on a given metric space. The use of the Wasserstein distance is motivated because this distance is defined and computable even between distributions with disjoint supports.

Definition 7.2 (Wasserstein Distance)**.** *The Wasserstein distance for our problem is defined as:*

$$W_c\left(\hat{P}, P^*\right) = \inf_{\gamma\in\Pi\left(\hat{P},P^*\right)}\mathbb{E}_{((\mathbf{x},\mathbf{z},y),(\mathbf{x}',\mathbf{z}',y'))\sim\gamma}\left[c\left((\mathbf{x},\mathbf{z},y),\left(\mathbf{x}',\mathbf{z}',y'\right)\right)\right] \tag{7.8}$$

where $c : \mathcal{X}\times\mathcal{X}\to[0,+\infty)$ is the convex, lower semicontinuous transport cost function with $c(\mathbf{t},\mathbf{t}) = 0$, and $\Pi\left(\hat{P}, P^*\right)$ is the set of all distributions whose marginals are given by $\hat{P}$ and P^*.

The Wasserstein distance intuitively refers to the minimum cost associated with transporting mass between probability measures. Suppose that the transportation

cost c in (7.8) is continuous and the probability of fair representation is bounded away from zero, i.e., $f_{\phi_1}(\boldsymbol{Z})$, then the minimax objective (7.7) has a desirable formulation as

$$\min_{f_\theta \in \mathcal{F}} \sup_{\hat{q}} \mathbb{E}_P \left[\frac{\delta\left(Y, f_{\phi_1}(\boldsymbol{Z})\right)}{\hat{q}(\mathbf{Z}|\mathbf{X})} \right] - \lambda W_c\left(\hat{q}(\boldsymbol{Z} \mid \boldsymbol{X}), q_*\right) \tag{7.9}$$

To make sense of (7.9), we see that while $\hat{q}(\boldsymbol{Z} \mid \boldsymbol{X})$ is acting adversarially against f_{ϕ_1} as the inverse weights in the first term, it cannot arbitrarily increase the objective function, since the second terms act as a regularizer that keeps $\hat{q}(\boldsymbol{Z} \mid \boldsymbol{X})$ close to the fair representation $\boldsymbol{Z}$. The objective loss in (7.9) can be converted to a two-model adversarial game:

$$\min_{f_{\phi_1} \in \mathcal{F}} \sup_{f_{\phi_2} \in \mathcal{G}} \mathbb{E}_P \left[\frac{\mathcal{L}\left(Y, f_{\phi_1}(\boldsymbol{Z})\right)}{G\left(f_{\phi_1}(\boldsymbol{Z})\right)} \right] - \lambda W_c\left(G\left(f_{\phi_1}(\mathbf{Z})\right), G\left(f^*_{\phi_1}\right)\right) \tag{7.10}$$

Theorem 7.1 (McDiarmid Inequality)**.** *(McDiarmid et al. 1989) Let $\Omega_1, \ldots, \Omega_m$ be probability spaces. Let $\Omega = \prod_{k=1}^{m} \Omega_k$ and let X be a random variable on Ω which is uniformly difference-bounded by $\frac{\lambda}{m}\cdot$ Let $\mu = \mathrm{E}(X)$. Then, for any $\tau > 0$*

$$P(X - \mu \geq \tau) \leq \exp\left(-\frac{2\tau^2 m}{\lambda^2}\right) \tag{7.11}$$

Suppose that the transportation cost c is continuous and the probability of fair representation is bounded away from zero, i.e., $\hat{q}(\mathbf{Z} \mid \mathbf{X})$, then the minimax objective has a desirable formulation as

$$\min_{f_{\phi_1} \in \mathcal{F}} \sup_{\hat{q}} \mathbb{E}_P \left[\frac{\delta\left(Y, f_{\phi_1}(\mathbf{Z})\right)}{\hat{q}(\mathbf{Z} \mid \mathbf{X})} \right] - \lambda W_c\left(\hat{q}(\mathbf{Z} \mid \mathbf{X}), q_*\right) \tag{7.12}$$

The following theorem discusses the theoretical guarantees for the generalization error of Eq. (7.10).

Theorem 7.2. *Suppose the mapping G from f_{ϕ_1} to $\hat{q}(\boldsymbol{Z} \mid \boldsymbol{X})$ is one-to-one and surjective with $g_\psi \in \mathcal{G}$. Let $\tilde{\mathcal{G}}(\rho) = \left\{g_\psi \in \mathcal{G} \mid W_c\left(G\left(g_\psi\right), G\left(g^*\right)\right) \leq \rho\right\}$. Then under the conditions specified in Proposition 7.5. for all $\gamma \geq 0$ and $\rho > 0$, the following inequality holds with probability at least $1 - \epsilon$:*

$$\sup_{g_\psi \in \tilde{\mathcal{G}}(\rho)} \mathbb{E}_P \left[\frac{\mathcal{L}\left(Y, f_{\phi_1}(\boldsymbol{Z})\right)}{G\left(f_{\phi_1}(\boldsymbol{Z})\right)} \right] \leq c_1 \gamma \rho + \mathbb{E}_{P_n}\left[\Delta_\gamma\left(f_{\phi_1}; (\boldsymbol{Z}, Y)\right)\right] + \frac{24\mathcal{J}(\tilde{\mathcal{F}}) + c_2\left(M, \sqrt{\log\frac{2}{\epsilon}}, \gamma\right)}{\sqrt{n}} \tag{7.13}$$

where $\mathbb{E}_{P_n}\left[\Delta_\gamma\left(f_{\phi_1};(\boldsymbol{Z},Y)\right)\right]$ *is a cost-regulated loss given in Proof part below,* c_1 *is a positive constants and* c_2 *is a simple linear function with positive weights.*

The above theorem states our main theoretical result on the worst-case generalization bound under the minimax setting.

Proof. We introduce a cost-regulated loss which is defined as

$$\Delta_\gamma\left(f_{\phi_1};(\mathbf{z},y)\right)=\sup_{(\mathbf{z}',y')\in\mathcal{X}}\left\{\frac{\delta\left(y',f_{\phi_1}(\mathbf{Z}')\right)}{q\left(o=1\mid\mathbf{z}'\right)}-\gamma c\left((\mathbf{z},y),(\mathbf{z}',y')\right)\right\} \tag{7.14}$$

Based on definition of Δ_γ, we have

$$\begin{aligned}
&\sup_{f_{\phi_1}\in\tilde{\mathcal{G}}(\rho)}\mathbb{E}_P\left[\frac{\delta\left(Y,f_{\phi_1}(\mathbf{Z})\right)}{G\left(f_{\phi_1}(\mathbf{X},\mathbf{Z})\right)}\right]\\
&\leq\inf_{\gamma\geq0}\left\{\gamma\rho+\int\sup_{\mathbf{h}\in\mathcal{X}}\left(\frac{\delta_{f_{\phi_1}}(\mathbf{h})}{\hat{q}(\mathbf{h})}-\gamma c\left(\mathbf{h},\mathbf{h}'\right)\right)dP(\mathbf{h})\right\}\\
&=\inf_{\gamma\geq0}\left\{\gamma\rho+\mathbb{E}_P\left[\Delta_\gamma\left(f_{\phi_1};\mathbf{H}\right)\right]\right\}\quad\text{(by the definition of }\Delta_\gamma)\\
&\leq\inf_{\gamma\geq0}\left\{\gamma\rho+\mathbb{E}_{P_n}\left[\Delta_\gamma\left(f_{\phi_1};\mathbf{H}\right)\right]+\sup_{f_{\phi_1}\in\mathcal{F}}\left(\mathbb{E}_P\left[\Delta_\gamma\left(f_{\phi_1};\mathbf{H}\right)\right]-\mathbb{E}_{P_n}\left[\Delta_\gamma\left(f_{\phi_1};\mathbf{H}\right)\right]\right)\right\}
\end{aligned} \tag{7.15}$$

Let $W_\gamma=\sup_{f_{\phi_1}\in\mathcal{F}}\left(\mathbb{E}_P\left[\Delta_\gamma\left(f_{\phi_1};\mathbf{H}\right)\right]-\mathbb{E}_{P_n}\left[\Delta_\gamma\left(f_{\phi_1};\mathbf{H}\right)\right]\right)$, then we have

$$W_\gamma=\frac{1}{n}\sup_{f_{\phi_1}\in\mathcal{F}}\left[\sum_{i=1}^{N}\mathbb{E}_P\left[\Delta_\gamma\left(f_{\phi_1};\mathbf{H}\right)\right]-\Delta_\gamma\left(f_{\phi_1};\mathbf{H}_i\right)\right]\quad\gamma\geq0 \tag{7.16}$$

According to Theorem 7.1 and the fact that $\left|\delta_{f_{\phi_1}}(\mathbf{h})\right|\leq\mu M$ holds uniformly, we have

$$p\left(W_\gamma-\mathbb{E}W_\gamma\geq\mu M\sqrt{\frac{\log 1/\epsilon}{2N}}\right)\leq\epsilon \tag{7.17}$$

where $\epsilon_1,\ldots,\epsilon_N$ is denoted as the i.i.d Rademacher random variables independent of $\mathbf{H}$, and $\mathbf{H}_i'$ is the i.i.d copy of $\mathbf{H}_i$ for $i=1,\ldots,N$.

Considering Eq. (7.16), we use the symmetrization argument to reformulate $\mathbb{E}W_\gamma$ in Eq. (7.17) as

$$
\begin{aligned}
\mathbb{E}W_\gamma &= \mathbb{E}\left[\sup_{f_{\phi_1}\in\mathcal{F}}\left|\sum_{i=1}^{N}\Delta_\gamma\left(f_{\phi_1};\mathbf{H}_i'\right)-\sum_{i=1}^{N}\Delta_\gamma\left(f_{\phi_1};\mathbf{H}_i\right)\right|\right] \\
&= \mathbb{E}\left[\sup_{f_{\phi_1}\in\mathcal{F}}\left|\frac{1}{N}\sum_{i=1}^{N}\epsilon_i\Delta_\gamma\left(f_{\phi_1};\mathbf{H}_i'\right)-\frac{1}{N}\sum_{i=1}^{N}\Delta_\gamma\left(f_{\phi_1};\mathbf{H}_i\right)\right|\right] \\
&\leq 2\mathbb{E}\left[\sup_{f_{\phi_1}\in\mathcal{F}}\left|\frac{1}{N}\sum_{i=1}^{N}\epsilon_i\Delta_\gamma\left(f_{\phi_1};\mathbf{H}_i\right)\right|\right]
\end{aligned} \tag{7.18}
$$

Apparently, each $\epsilon_i\Delta_\gamma\left(f_{\phi_1};\mathbf{H}_i\right)$ is zero-mean, and now we show that it is sub-Gaussian as well. The bounded difference between two f_{ϕ_1}, f'_{ϕ_1} is

$$
\begin{aligned}
&\mathbb{E}\left[\exp\left(\lambda\left(\frac{1}{\sqrt{N}}\epsilon_i\Delta_\gamma\left(f_{\phi_1};\mathbf{H}_i\right)-\frac{1}{\sqrt{N}}\epsilon_i\Delta_\gamma\left(f'_{\phi_1};\mathbf{H}_i\right)\right)\right)\right] \\
&= \left(\mathbb{E}\left[\exp\left(\frac{\lambda}{\sqrt{N}}\epsilon_1\left(\Delta_\gamma\left(f_{\phi_1};\mathbf{H}_1\right)-\Delta_\gamma\left(f'_{\phi_1};\mathbf{H}_1\right)\right)\right)\right]\right)^N \\
&= \left(\mathbb{E}\left[\exp\left(\frac{\lambda}{\sqrt{N}}\epsilon_1\left(\sup_{\mathbf{h}'}\inf_{\mathbf{h}''}\left\{\frac{\delta_{f_{\phi_1}}(\mathbf{h}')}{q\left(\mathbf{h}'\right)}-\gamma c\left(\mathbf{H}_1,\mathbf{h}'\right)-\frac{\delta_{f'_{\phi_1}}(\mathbf{h}'')}{q\left(\mathbf{h}''\right)}\right\}+\gamma c\left(\mathbf{H}_1,\mathbf{h}''\right)\right)\right)\right]\right)^N \\
&\leq \left(\mathbb{E}\left[\exp\left(\frac{\lambda}{\sqrt{N}}\epsilon_1\left(\sup_{\mathbf{h}'}\left\{\frac{\delta_{f_{\phi_1}}(\mathbf{h}')}{q\left(\mathbf{h}'\right)}-\frac{\delta_{f'_{\phi_1}}(\mathbf{h}')}{q\left(\mathbf{h}'\right)}\right\}\right)\right)\right]\right)^N \\
&\leq \exp\left(\lambda^2\left\|\frac{\delta_{f_{\phi_1}}}{q}-\frac{\delta_{f'_{\phi_1}}}{q}\right\|_\infty^2/2\right) \quad \text{(by Hoeffding's inequality)}
\end{aligned} \tag{7.19}
$$

Hence we see that $\frac{1}{\sqrt{N}}\epsilon_i\Delta_\gamma\left(f_{\phi_1};\mathbf{H}_i\right)$ is sub-Gaussian with respect to $\left\|\frac{\delta_{f_{\phi_1}}}{q}-\frac{\delta_{f'_{\phi_1}}}{q}\right\|_\infty^2$. Therefore, $\mathbb{E}W_\gamma$ can be bounded— by using the standard technique for Rademacher complexity and Dudley's entropy integral

$$
\mathbb{E}W_\gamma \leq \frac{24}{N}\mathcal{J}(\tilde{\mathcal{F}}) \tag{7.20}
$$

Based on all above bounds in (7.15), (7.17) and (7.20) we obtain the desired result. □

7.6 Experiments

Compared to other fairness criteria, evaluating the performance of counterfactual fairness is frustratingly difficult due to the absence of ground truth samples. In fact,

from the observational data, we are unable to observe the characteristic of individuals in the counterfactual world where we make an intervention into their sensitive attributes. In fact, we cannot simply change the values of sensitive attributes since the intervention on the sensitive features can lead to changes in some non-sensitive features due to the causal effects. For example, we have an observational individual x, but do not have its counterfactual version $\hat{x}$; therefore, it is not feasible to evaluate the performance of predictive model $f(.)$ by measuring the similarity of $f(x)$ and $f(\hat{x})$. In the previous studies (Kusner et al. 2017; Russell et al. 2017; Wu et al. 2019), they generate both the original samples and counterfactual samples from the structural causal model. However, it is hard to verify the trustworthiness of the samples due to the unidentifiability of the causal model. In this research, by getting a pair of similar individuals sharing the same properties, we thus can approximately evaluate the model performance. This means that instead of evaluating in the counterfactual space, we can approximately evaluate the performance of counterfactual fairness via the individual fairness criteria. We conducted extensive experiments on three real-world datasets with different evaluation metrics for two tasks including regression and classification tasks.

7.6.1 Datasets

We evaluate our approach via regression datasets including `LSAC` (Wightman 1998) and classification datasets including `Compas` (Larson et al. 2016) and `Adult`.

- `LSAC`* (Wightman 1998). `LSAC` dataset provides information about law students including their gender, race, entrance exam scores (LSAT), grade-point average (GPA) and first-year average grade (FYA). The main task is to determine which applicants would have a high possibility to obtain high FYA. The school also ensures that model decisions are not biased by sensitive attribtues including race and gender. We pay attention to predict the FYA of a student.
- `Compas`† (Larson et al. 2016). `Compas` dataset has been released by ProPub-

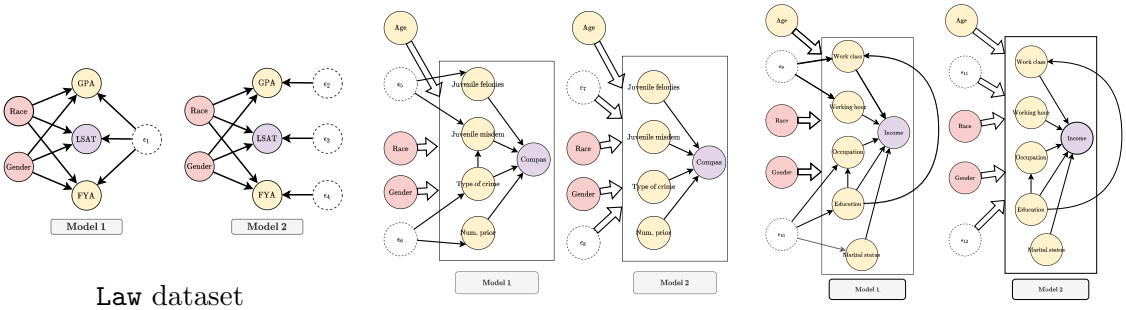
*Download at: http://www.seaphe.org/databases.php

†Download at: https://www.propublica.org

lica about prisoners in Florida (US) and also has been previously explored for fairness studies in criminal justice (Berk et al. 2021). The dataset contains information about 6,167 prisoners, and each individual has two sensitive attributes including gender, race and other attributes related to prior conviction and age. The main task is to predict whether or not a prisoner will re-offend within two years after being released from prison.

- `Adult`[‡](Dua and Graff 2017). `Adult` dataset is the real-world dataset providing information about loan applicants in the financial organization. The dataset consists of both continuous features and categorical features. The main task is to determine whether a person has an annual income exceeding $50k dollars. The sensitive attributes are gender and race.

To evaluate the generalization capability of models, we randomly split each dataset into 80% training and 20% test set. We conduct 100 repeated experiments, then evaluate performance on the test set and finally report the average statistics.



`Law` dataset

Compas dataset

`Adult` dataset

Figure 7.3 : Causal diagrams for Law, Compas and `Adult` dataset. $\{\epsilon_1 \cdots \epsilon_{12}\}$ are the unobserved variables. The large white arrows represent that each variable has a causal effect on every variables contained in the box.

[‡]Download at: https://archive.ics.uci.edu/ml/datasets/adult

7.6.2 Baselines

We make a comparison with several state-to-the-art methods as below.

- **Full features (Full)** (Kusner et al. 2017) is the standard technique that uses all the features including both the sensitive and non-sensitive ones.
- **Unaware features (Unaware)** (Chen et al. 2019) does not consider sensitive features such as race or gender in the input and only utilizes non-sensitive features.
- **Counterfactual fairness model (CF)** (Kusner et al. 2017) uses the causal graph and infers the latent variables which are not the child nodes of the sensitive features. Since there is no ground truth causal model, we consider two causal diagrams illustrated in Figure 7.3 for each dataset as $\mathbf{CF}_1$ and $\mathbf{CF}_2$. $\mathbf{CF}_1$ and $\mathbf{CF}_2$ correspond to two different structural causal models
- **Multi-world models (Multi-wolrd)** (Russell et al. 2017) minimizes the model predictions when considering different structural causal models. Specifically, with two causal diagrams in Figure 7.3 for each dataset, we build a neural network and use the gradient descent algorithm to minimize the model output from two causal models.
- **Auto-encoder model (AE)** (Ng et al. 2011) uses the encoder-decoder architecture to learn the latent representation. This model utilizes all the features including the sensitive and non-sensitive ones as the input.

Note that **Full**, **Unaware**, $\mathbf{CF}_1$, $\mathbf{CF}_2$ and **Autoencoder** are the representation methods that only produce features, so we use these features as an input and construct predictive models including Linear Regression **(LR)** and Gradient Boosting Regression **(GBboostR)** for regression task, and Logistic Regression **(Log)** and Gradient Boosting Classifier **(GBboostC)** for the classification task. As our method aims to output a fair and informative representation, we use two models: Invariant-encoder model and Fair-learning predictive model. In order to gain more

Method	Regression metric			Fairness metric	
	RMSE	**MAE**	**R2score**	**Wasserstein**	**Gaussian**
Full-LR	**0.870 ± 3.2e-03**	**0.705 ± 7.4e-03**	0.120 ± 5.0e-03	0.522 ± 2.2e-03	0.719 ± 3.2e-03
Full-GBoostR	0.935 ± 2.8e-03	0.751 ± 3.2e-03	**-0.014 ± 4.9e-03**	0.010 ± 6.5e-03	0.037 ± 5.5e-03
Unaware-LR	0.889 ± 7.5e-03	0.718 ± 3.2e-03	0.083 ± 2.4e-03	0.097 ± 4.5e-03	0.194 ± 2.1e-03
Unaware-GBoostR	1.034 ± 7.8e-03	0.829 ± 5.1e-03	**-0.242 ± 3.2e-03**	**0.009 ± 4.8e-03**	0.030 ± 5.1e-03
CF_1-LR	0.906 ± 4.1e-03	0.730 ± 5.1e-03	0.048 ± 4.8e-03	0.019 ± 3.1e-03	0.045 ± 3.0e-03
CF_1-GBoostR	0.909 ± 2.1e-03	0.732 ± 4.6e-03	0.0410 ± 5.1e-03	0.013 ± 7.6e-03	0.037 ± 4.3e-03
CF_2-LR	0.914 ± 7.3e-03	0.736 ± 5.1e-03	0.030 ± 6.4e-03	0.070 ± 7.5e-03	0.030 ± 8.1e-03
CF_2-GBoostR	0.913 ± 4.9e-03	0.734 ± 3.6e-03	0.034 ± 7.5e-03	0.070 ± 7.3e-03	0.032 ± 8.5e-03
Multi-world	0.917 ± 3.9e-03	0.736 ± 7.1e-03	0.025 ± 7.1e-03	0.030 ± 5.8e-03	0.036 ± 4.7e-03
AE-LR	**0.870 ± 7.1e-03**	**0.705 ± 4.1e-03**	0.121 ± 6.0e-03	0.532 ± 2.1e-03	0.705 ± 8.1e-03
AE-GBoostR	0.889 ± 3.6e-03	0.715 ± 8.1e-03	0.221 ± 5.1e-03	0.425 ± 8.1e-03	0.815 ± 4.8e-03
InvEnc-LR	0.905 ± 3.4e-03	0.727 ± 7.1e-03	0.040 ± 2.1e-03	0.131 ± 8.1e-03	0.160 ± 5.4e-03
InvEnc-GBoostR	0.904 ± 7.1e-03	0.773 ± 3.1e-03	0.131 ± 3.2e-03	0.183 ± 1.9e-03	0.179 ± 7.1e-03
InvFair (Ours)	0.900 ± 2.2e-03	0.739 ± 2.5e-03	0.087 ± 4.1e-03	**0.009 ± 8.6e-03**	**0.029 ± 2.1e-03**

Table 7.1 : Performance comparisons on `Law` dataset. The mean and variance for each method are obtained via 100 repeated runs. For **R2score**, results in bold font show the corresponding models are unreliable. For the **remaining metrics**, the best results are bold. For each method, we use (baseline)-LR/GBoostR to show the baseline combined with Logistic regression or Gradient boosting.

insights into model behaviors, we first utilize Invariant-encoder model **(InvEnc)** to generate the latent representation, and also combine with LR, GBboostR, and GBboostC. Finally, we adopt the fair-learning predictive model and invariant-encoder together, referred as **InvFair**.

7.6.3 Evaluation metrics

Our method aims to learn the fair and informative representation that can be used for downstream classification or regression. We use two metrics for prediction and fairness performance, and consider both regression and classification tasks.

For the prediction performance, we use root mean squared error (RMSE) and mean absolute error (MAE) for the regression task. For the classification task, we

Method	Regression metric			Fairness metric	
	RMSE	**MAE**	**R2score**	**Wasserstein**	**Gaussian**
Full-LR	0.0356	0.0178	0.0227	0.0388	0.019
Full-GBoostR	0.0262	0.0475	0.031	0.0433	0.034
Unaware-LR	0.0106	0.0085	0.0234	0.0161	0.0432
Unaware-GBoostR	0.0371	0.0324	0.0322	0.028	0.0392
CF_1-LR	0.0416	0.0302	0.0251	0.0262	0.0495
CF_1-GBoostR	0.0115	0.0214	0.0089	0.0161	0.0215
CF_2-LR	0.0449	0.0325	0.041	0.0253	0.0376
CF_2-GBoostR	0.0444	0.0308	0.0314	0.0464	0.0479
Multi-world	0.0253	0.0377	0.0440	0.0402	0.0136
AE-LR	0.0426	0.0124	0.0254	0.0229	0.0284
AE-GBoostR	0.0438	0.0238	0.0264	0.0251	0.0289
InvEnc-LR	0.0162	0.0321	0.0215	0.0154	0.0278
InvEnc-GBoostR	0.0285	0.0474	0.0259	0.0489	0.0349

Table 7.2 : We compute p-value by conducting a paired t-test between our approach and baselines with 100 repeated experiments for each metric on Law dataset.

use Precision, Recall, F_1 score, and Balanced Accuracy (Brodersen et al. 2010) for evaluation purpose. We emphasize that since the `Adult` and `Compas` datasets are highly imbalanced, we use the Balanced Accuracy instead of the traditional accuracy, which is defined as Balanced Acc $= \frac{\text{TPR}+\text{TNR}}{2}$ where TPR and TNR are true positive rate, and true negative rate, respectively.

For the fairness performance, we use Wasserstein distance (Wasserstein) (Rüschendorf 1985) and maximum mean discrepancy (MMD) with Gaussian kernel (Gretton et al. 2012; Oh et al. 2019) (Gaussian) in the regression task. On the other hand, we utilize generalized entropy index (Speicher et al. 2018) to evaluate the performance in the classification task. Generalized entropy index that has been previously used widely in economics is explored by (Speicher et al. 2018) as the unified approach to evaluate the performance of individual fairness algorithms defined as follows:

$$\mathcal{E}(\alpha) = \begin{cases} \frac{1}{n\alpha(\alpha-1)} \sum_{i=1}^{n} \left[\left(\frac{b_i}{\mu} \right)^{\alpha} - 1 \right], & \alpha \neq 0, 1, \\ \frac{1}{n} \sum_{i=1}^{n} \frac{b_i}{\mu} \ln \frac{b_i}{\mu}, & \alpha = 1, \\ -\frac{1}{n} \sum_{i=1}^{n} \ln \frac{b_i}{\mu}, & \alpha = 0. \end{cases} \tag{7.21}$$

where $b_i = \hat{y}_i - y_i + 1$, $\mu = \frac{1}{n} \sum_i^n b_i$. In this study, we use $\mathcal{E}(1)$ and $\mathcal{E}(2)$ which are called Theil index (TI) and coefficient of variation (CV), respectively. For all metrics except Precision, Recall, F_1 score, and Balanced Accuracy, lower values are better.

Method	Classification metric				Fairness metric	
	Balanced Acc	**F_1**	**Precision**	**Recall**	**CV**	**TI**
Full-Log	0.660 ± 5.3e-03	0.664 ± 7.5e-03	0.672 ± 3.1e-03	0.670 ± 3.0e-03	0.891 ± 4.5e-03	0.285 ± 2.8e-03
Full-GBoostC	0.665 ± 1.8e-03	0.670 ± 7.5e-03	0.675 ± 7.8e-03	0.674 ± 7.6e-03	0.872 ± 6.5e-03	0.271 ± 8.2e-03
Unaware-Log	0.662 ± 4.5e-03	0.666 ± 7.9e-03	0.674 ± 7.9e-03	0.672 ± 6.0e-03	0.887 ± 1.8e-03	0.282 ± 4.6e-03
Unaware-GBoostC	0.662 ± 7.9e-03	0.666 ± 7.7e-03	0.672 ± 4.2e-03	0.672 ± 4.9e-03	0.880 ± 3.1e-03	0.276 ± 5.0e-03
CF_1-Log	0.500 ± 8.0e-03	0.381 ± 2.3e-03	0.522 ± 5.8e-03	0.540 ± 4.6e-03	1.306 ± 6.5e-03	0.615 ± 1.4e-03
CF_1-GBoost	0.534 ± 3.7e-03	0.517 ± 2.7e-03	0.551 ± 5.4e-03	0.556 ± 1.4e-03	1.159 ± 3.2e-03	0.463 ± 6.0e-03
CF_2-Log	0.623 ± 2.5e-03	0.627 ± 5.7e-03	0.628 ± 7.5e-03	0.629 ± 7.2e-03	0.904 ± 3.7e-03	0.286 ± 6.8e-03
CF_2-GBoost	0.573 ± 8.5e-03	0.572 ± 5.6e-03	0.576 ± 5.5e-03	0.571 ± 3.4e-03	0.871 ± 4.7e-03	0.262 ± 6.9e-03
Multi-world	0.500 ± 7.5e-03	0.381 ± 8.6e-03	0.522 ± 2.1e-03	0.540 ± 8.5e-03	1.306 ± 4.5e-03	0.615 ± 3.6e-03
AE-Log	0.659 ± 6.2e-03	0.663 ± 4.8e-03	0.667 ± 4.3e-03	0.667 ± 5.4e-03	0.876 ± 6.9e-03	0.272 ± 4.0e-03
AE-GBoostC	0.666 ± 5.3e-03	0.670 ± 6.4e-03	0.676 ± 6.8e-03	0.675 ± 8.6e-03	0.874 ± 5.8e-03	0.273 ± 1.6e-03
InvEnc-Log	**0.670 ± 1.5e-03**	**0.675 ± 2.9e-03**	**0.681 ± 6.8e-03**	**0.680 ± 8.5e-03**	0.869 ± 4.1e-03	0.270 ± 1.6e-03
InvEnc-GBoostC	0.666 ± 5.8e-03	0.670 ± 1.4e-03	0.676 ± 8.7e-03	0.675 ± 5.7e-03	0.874 ± 6.3e-03	0.273 ± 2.1e-03
InvFair (Ours)	0.668 ± 2.7e-03	0.672 ± 2.5e-03	0.672 ± 2.6e-03	0.673 ± 5.9e-03	**0.836 ± 5.1e-03**	**0.211 ± 2.2e-03**

Table 7.3 : Performance comparison on `Compas` dataset. The mean and variance for each method are obtained via 100 repeated experiments. The best results are bold. For each method, we name (*)-Log/GBoostC with (*) representing the baseline method.

7.6.4 Implementation details

All implementations are conducted in Python 3.7.7 with 64-bit Red Hat, Intel(R) Xeon(R) Gold 6150 CPU @ 2.70GHz. The models for all datasets were trained with the following settings: 200 epochs, batch size of 64, Adam optimizer with the learning rate of 10^{-3}, smooth loss function (Girshick 2015) for `Law` dataset and cross-entropy loss function for `Adult` and `Compas` dataset. We used LeakyReLu (Maas

Method	Classification metric				Fairness metric	
	Balanced Acc	F_1	Precision	Recall	CV	TI
Full-Log	0.0419	0.0498	0.040	0.0489	0.0214	0.031
Full-GBoostC	0.0112	0.0482	0.0101	0.0261	0.014	0.027
Unaware-Log	0.0106	0.0450	0.0458	0.0342	0.047	0.0184
Unaware-GBoostC	0.0476	0.0164	0.0347	0.0364	0.0391	0.0159
CF_1-Log	0.0364	0.0367	0.0323	0.017	0.0305	0.0173
CF_1-GBoost	0.0386	0.0302	0.0331	0.0144	0.0105	0.049
CF_2-Log	0.0343	0.0475	0.0459	0.0122	0.0384	0.0148
CF_2-GBoost	0.0184	0.0259	0.0104	0.0173	0.0475	0.0302
Multi-world	0.0258	0.0129	0.0473	0.0186	0.0316	0.0344
AE-Log	0.0164	0.0363	0.0166	0.0468	0.0454	0.0151
AE-GBoostC	0.0401	0.0387	0.0183	0.0207	0.0335	0.0171
InvEnc-Log	0.0208	0.036	0.0272	0.0147	0.0481	0.0398
InvEnc-GBoostC	0.0377	0.0364	0.0157	0.030	0.0117	0.0333

Table 7.4 : We compute p-value by conducting a paired t-test between our approach and baselines with 100 repeated experiments for each metric on Compas dataset.

et al. 2013) as the $h(.)$ function. We implemented the baseline methods by using Pyro library (Bingham et al. 2019), while our method was implemented by Pytorch. As regards the evaluation metric, we utilized the available functions from library AI360 (Bellamy et al. 2018) and GeomLoss (Feydy et al. 2019). More details of implementation settings can be found in the provided source code.

7.6.5 Comparison results

In this section, we report the empirical performance of different methods across three datasets on both the regression and classification tasks. In general, we aim to investigate the following research questions 1) how our approach achieves better fairness and accuracy tradeoff compared to other baselines; 2) how the model performance fluctuates with different hyperparameter λ (the values of λ for competitive and stable performance).

Regression task. Table 7.1 indicates the performance comparison for `Law` dataset. In particular, Full-LR and AE-LR models result in the best accuracy

Method	Classification metric				Fairness metric	
	Balanced Acc	**F_1**	**Precision**	**Recall**	**CV**	**TI**
Full-Log	0.606 ± 7.6e-03	0.741 ± 6.4e-03	0.743 ± 6.5e-03	0.771 ± 3.4e-03	0.745 ± 8.7e-03	0.215 ± 7.7e-03
Full-GBoostC	0.730 ± 6.5e-03	**0.820 ± 1.8e-03**	0.818 ± 6.2e-03	**0.827 ± 3.8e-03**	0.616 ± 3.4e-03	0.142 ± 3.4e-03
Unaware-Log	0.551 ± 1.9e-03	0.700 ± 7.8e-03	0.697 ± 7.2e-03	0.747 ± 2.5e-03	0.801 ± 8.3e-03	0.249 ± 8.6e-03
Unaware-GBoostC	0.725 ± 7.8e-03	0.816 ± 8.4e-03	0.815 ± 7.8e-03	0.824 ± 7.2e-03	0.622 ± 3.7e-03	0.145 ± 6.7e-03
CF_1-Log	0.515 ± 8.6e-03	0.670 ± 5.8e-03	0.675 ± 2.3e-03	0.749 ± 8.2e-03	0.814 ± 3.2e-03	0.272 ± 8.4e-03
CF_1-GBR	0.513 ± 2.1e-03	0.666 ± 3.3e-03	0.712 ± 6.8e-03	0.757 ± 2.4e-03	0.801 ± 7.0e-03	0.273 ± 6.0e-03
CF_2-Log	0.515 ± 4.1e-03	0.669 ± 1.3e-03	0.671 ± 7.7e-03	0.747 ± 6.4e-03	0.817 ± 8.0e-03	0.272 ± 3.7e-03
CF_2-GBoostC	0.520 ± 5.2e-03	0.674 ± 1.9e-03	0.700 ± 5.5e-03	0.756 ± 8.1e-03	0.802 ± 6.4e-03	0.269 ± 3.6e-03
Multi-world	0.510 ± 3.9e-03	0.664 ± 5.1e-03	0.664 ± 7.0e-03	0.747 ± 2.5e-03	0.819 ± 3.6e-03	0.275 ± 7.8e-03
AE-Log	0.730 ± 2.1e-03	0.817 ± 6.0e-03	0.815 ± 4.2e-03	0.823 ± 8.2e-03	0.819 ± 8.2e-03	0.242 ± 2.3e-03
AE-GBoostC	0.724 ± 5.5e-03	0.815 ± 1.9e-03	0.814 ± 7.6e-03	0.823 ± 2.7e-03	0.823 ± 3.3e-03	0.245 ± 5.4e-03
InvEnc-Log	0.723 ± 7.9e-03	0.812 ± 6.9e-03	0.810 ± 4.9e-03	0.819 ± 6.2e-03	0.627 ± 8.1e-03	0.146 ± 2.8e-03
InvEnc-GBoostC	0.724 ± 3.6e-03	0.815 ± 2.3e-03	0.814 ± 8.2e-03	0.823 ± 1.4e-03	0.623 ± 2.6e-03	0.145 ± 7.4e-03
InvFair (Ours)	**0.778 ± 8.6e-03**	0.728 ± 5.4e-03	**0.835 ± 3.5e-03**	0.707 ± 7.2e-03	**0.556 ± 4.7e-03**	**0.090 ± 7.4e-03**

Table 7.5 : Performance comparison on `Adult` dataset. The mean and variance for each method are obtained via 100 repeated experiments. The best results are bold. For each method, we name (*)-Log/GBoostC with (*) representing features generated by baseline method.

outcome with the lowest RMSE and MAE; however, this model fails to produce the fair prediction demonstrated by the highest fairness metrics. The possible reason is that both Full-LR and AE-LR use all features including the sensitive features, which is beneficial for the accuracy aspect but contains bias. The counterfactual fairness (CF_1- and CF_2-) and Multi-world methods in contrast witness a good performance when they come to fairness with a significantly low Wasserstein and Gaussian distance, but have quite high regression metrics. Meanwhile, our proposed method (InvFair) consistently produces the lowest results in Wasserstein and Gaussian distance and achieves quite competitive results in RMSE and MAE. We also observe that Linear Regression (-LR) performs better than Gradient Boosting Regression (-GBoostR). Finally, we notice that although the outstanding results are also recorded with Unaware-GBoostR in fairness aspects (0.009 for Wasserstein and 0.03 for Gaussian), its R2score is a negative number which implies the poor performance in the regression task.

Method	Classification metric				Fairness metric	
	Balanced Acc	**F_1**	**Precision**	**Recall**	**CV**	**TI**
Full-Log	0.0288	0.0341	0.0443	0.0145	0.014	0.0257
Full-GBoostC	0.0275	0.0149	0.0458	0.0193	0.0111	0.0091
Unaware-Log	0.0122	0.0173	0.0151	0.0398	0.038	0.0478
Unaware-GBoostC	0.0369	0.0476	0.0431	0.0385	0.0177	0.0352
CF_1-Log	0.0473	0.0366	0.0144	0.0222	0.0446	0.0337
CF_1-GBoost	0.0473	0.0338	0.0193	0.0492	0.0109	0.0298
CF_2-Log	0.0405	0.0269	0.0393	0.0441	0.0254	0.0203
CF_2-GBoost	0.0361	0.0348	0.021	0.0151	0.0107	0.0233
Multi-world	0.0378	0.0264	0.011	0.0137	0.0326	0.0242
AE-Log	0.0369	0.0307	0.0325	0.0092	0.0143	0.013
AE-GBoostC	0.0285	0.0294	0.0155	0.0409	0.0175	0.025
InvEnc-Log	0.0372	0.0321	0.048	0.0258	0.0311	0.0164
InvEnc-GBoostC	0.0239	0.0466	0.0218	0.0456	0.0218	0.0253

Table 7.6 : We compute p-value by conducting a paired t-test between our approach and baselines with 100 repeated experiments for each metric on Adult dataset.

Classification task. We analyze the task of classification on `Compas` and `Adult` datasets on Table 7.3 and Table 7.5, respectively. It is illustrated from Table 7.3 that the poor performance is recorded with counterfactual fairness (CF_1- and CF_2-) and Multi-world approach with low results for classification metrics. In contrast, the latent representation produced from InvEnc combined with Logistic Regression and GBooost achieves the greatest results in terms of the classification metric including Balanced Accuracy, Precision, Recall and f-measure. Meanwhile, our proposed method surpasses all of the other methods regarding fairness metrics (CV and TI). It is moreover ranked second regarding Balanced Accuracy and F_1 score and ranked third regarding Precision and Recall. On the other hand, Table 7.5 shows the results of different methods in `Adult` dataset. This dataset is highly imbalanced with the ratio of positive and negative classes being 70% and 30%. Our proposed approach produces the best Balanced Accuracy and Precision, while Full-GBoostC has the greatest F_1 and Recall score. Regarding fairness metrics, our method consistently surpasses all of the remaining methods. Moreover, gradient boosting classification (-GBoostC) performs better Logistic Regression model (-Log). As seen from the

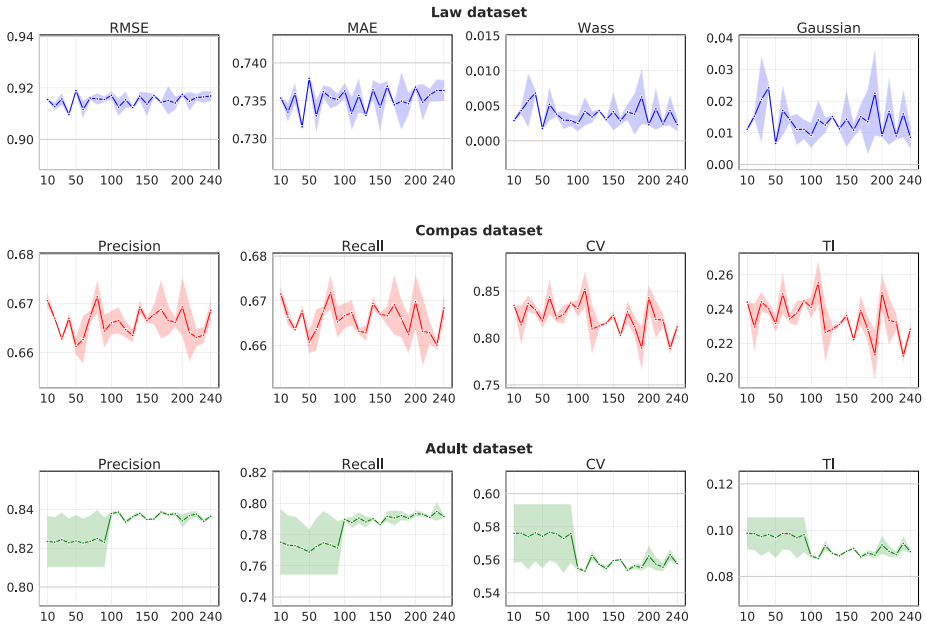


Figure 7.4 : We report the performance of our approach with different hyperparameter λ on `Law`, `Compas` and `Adult` datasets. For each λ, we repeat the experiment 100 times to get the mean and variance.

classification task, counterfactual fairness (CF_1- and CF_2-) and Multi-world model perform poorly in the classification task, possibly due to the misspecification of structural causal models. The invariant-encoder model (InvEnc) that minimizes the prediction of sensitive-awareness and fair-learning models allows the latent representation to achieve favorable outcomes in terms of accuracy aspects. Furthermore, when combined with the fair-learning models in our final approach (InvFair), it produces competitive results in both the prediction and fairness performance.

Statistical significance. To better comprehend the effectiveness of our proposed method in producing counterfactual samples compared with other approaches, we also perform a statistical significance test (paired t-test) between our approach and other methods on each dataset and each metric with the obtained results on 100 randomly repeated experiments and report the result of p-value in Table 7.2,

Table 7.4 and Table 7.6. We find that our model is statistically significant with $p < 0.05$, thus demonstrating the effectiveness of our proposed method in achieving counterfactual fairness.

Sensitivity of hyperparameter. Figure 7.4 shows the variation of our proposed method performance with different settings of hyperparameter λ. For `Law` dataset, Gaussian distance fluctuates slightly from 0.01 to 0.02, while RMSE, MAE and Wasserstein are recorded at steady results. In terms of our proposed method performance on `Compas` and `Adult` datasets. In general, Precision and Recall share the same patterns, while CV and TI demonstrate similar trends. For `Compas` dataset, the performance of Precision and Recall have slight fluctuations of 0.66 and 0.68, while CV and TI vary marginally around 0.8-0.85 and 0.2-0.25, respectively. For `Adult` dataset, the performance witnesses a quite big variation before λ reaches 100, and thereafter achieves the outstanding and stable performance when λ is greater than 100.

7.7 Conclusion

This paper proposes a minimax game-theoretic approach that can maintain competitive performance in predictive tasks and make counterfactually fair decisions at the individual level. We believe that training minimax objective functions for invariant-encoder model and fair-learning predictive model allow us to exclude the sensitive information in models' decisions, and also maintain high accuracy performance. Empirical results on three real-world datasets demonstrated that our proposed approach (InvFair) performs best regarding fairness metrics and also achieves a favorable fairness-accuracy tradeoff. Most importantly, our approach does not require prior knowledge about the structural causal model, making it attractive in real-world applications. In future work, we plan to investigate how to estimate fair causal effects.

Part V

Conclusions

Chapter 8

Conclusion and Future Work

8.1 Thesis Summary

Machine learning has made impressive strides in recent years, with algorithms delivering human-level performance across a broad range of tasks. However, one significant challenge with machine learning is the lack of interpretability, which hampers our understanding of the underlying factors driving algorithmic decisions.

In response to this challenge, this thesis introduces several strategies for interpretable machine learning, incorporating insights from the field of causal inference. Specifically, we present the stochastic propensity score approach for estimating average treatment effects and for policy optimization. Regarding counterfactual explanations, we suggest two frameworks, ProCE and CE-Flows. These methods help in establishing causal relationships in counterfactual samples, generating robust samples, and enhancing the efficiency of sample generation. Lastly, we turn our attention to counterfactual fairness, proposing a min-max game theoretic approach capable of achieving this property, even with imperfect structural causal models.

Overall, this thesis contributes to the growing field of interpretable machine learning by integrating principles from causal inference. It offers a promising path for the development of more transparent and trustworthy machine learning systems.

8.2 Key Findings and Conclusions

The objectives of this study have been proficiently accomplished through the following endeavors:

In Chapter 2, we conducted a comprehensive survey on interpretable machine learning and its relation to causality. This chapter provides a comprehensive overview

of various facets of interpretable machine learning and its associations with causality. While our survey offers a broad understanding of interpretable machine learning, it's important to acknowledge that the field is in a state of constant evolution, which may lead to the emergence of new algorithms after our study.

Secondly, in Chapters 3 and 4, we delve into the exploration of a stochastic propensity score for estimating average treatment effects. This methodology aids in the process of determining treatment effects in observational studies and streamlines policy optimization. However, one of limitations of our causal framework is that it cannot deal with time-independent stochastic interventions.

Subsequently, in Chapters 5 and 6, we introduced algorithms for counterfactual explanation, specifically designed to establish causal relationships in counterfactual samples. Additionally, we proposed another algorithm to generate robust samples efficiently. Although our proposed approach achieves robustness and causal relationships, an important aspect we have yet to consider is fairness properties, which hold significant role in counterfactual explanations.

Finally, in Chapter 7, we introduced a model for counterfactual fairness founded on an imperfect structural causal model. This model presents a solution to address the vital ethical concern of fairness in machine learning models, thereby augmenting their acceptance and trustworthiness. Our work lays the groundwork for exploring counterfactual fairness with the relaxation of assumptions of SCMs.

In conclusion, this study has successfully attained its objectives and has made a substantial contribution to the swiftly evolving field of interpretable machine learning and causality.

8.3 Future Work

Causality is an essential aspect of understanding the relationship between variables and their impact on outcomes in many fields, including medicine, economics, and social sciences. Interpretable machine learning (IML) models aim to provide transparency and explainability to the decision-making process of black-box models. However, without considering the causal structure of the data, IML models may suf-

fer from spurious correlations and misleading explanations. Therefore, incorporating causal reasoning into IML models can enhance their interpretability, reliability, and decision-making accuracy.

Here are some potential research directions for causal IML:

- **Developing causal inference methods for IML**: Current causal inference methods, such as structural equation modeling and counterfactual analysis, focus on the analysis of small-scale data and do not scale well for high-dimensional, complex datasets. Therefore, there is a need for developing scalable and efficient causal inference methods that can be integrated with IML models.

- **Combining causal modeling and IML**: Causal modeling can provide a framework for designing and evaluating IML models by incorporating prior knowledge about the causal relationships between variables. By combining causal modeling and IML, we can develop more robust and accurate models that can explain the underlying mechanisms of the data.

- **Evaluating the causal impact of IML models**: IML models may have unintended consequences or generate unfair outcomes due to the causal structure of the data. Therefore, it is essential to evaluate the causal impact of IML models and ensure that they do not violate ethical and legal principles.

- **Applying causal IML to real-world applications**: The integration of causal reasoning into IML models has the potential to improve decision-making in many domains, such as healthcare, finance, and social policy. Therefore, future studies should focus on developing and evaluating causal IML models in real-world applications and assessing their effectiveness and practicality.

In summary, causal reasoning is an important aspect of interpretability in machine learning, and integrating causal inference methods into IML models can enhance their interpretability and decision-making accuracy.

Bibliography

Abdia, Y., Kulasekera, K., Datta, S., Boakye, M. & Kong, M., 2017, 'Propensity scores based methods for estimating average treatment effect and average treatment effect among treated: A comparative study', *Biometrical Journal*, vol. 59, no. 5, pp. 967–985.

Alemi, F., Erdman, H., Griva, I. & Evans, C., 2009, 'Improved statistical methods are needed to advance personalized medicine', *The open translational medicine journal*, vol. 1, pp. 16–20.

Aliferis, C. F., Tsamardinos, I. & Statnikov, A., 2003, 'Hiton: a novel markov blanket algorithm for optimal variable selection', *AMIA annual symposium proceedings*, , vol. 2003American Medical Informatics Association, p. 21.

Arik, S. Ö. & Pfister, T., 2021, 'Tabnet: Attentive interpretable tabular learning', *Proceedings of the AAAI conference on artificial intelligence*, , vol. 35pp. 6679–6687.

Artelt, A. & Hammer, B., 2020, 'Convex density constraints for computing plausible counterfactual explanations', *Artificial Neural Networks and Machine Learning–ICANN 2020: 29th International Conference on Artificial Neural Networks, Bratislava, Slovakia, September 15–18, 2020, Proceedings, Part I 29*, Springer, pp. 353–365.

Athey, S., Tibshirani, J., Wager, S. et al., 2019, 'Generalized random forests', *The Annals of Statistics*, vol. 47, no. 2, pp. 1148–1178.

Austin, P. C. & Stuart, E. A., 2015, 'Moving towards best practice when using inverse probability of treatment weighting (iptw) using the propensity score to

estimate causal treatment effects in observational studies', *Statistics in medicine*, vol. 34, no. 28, pp. 3661–3679.

Bang, H. & Robins, J. M., 2005, 'Doubly robust estimation in missing data and causal inference models', *Biometrics*, vol. 61, no. 4, pp. 962–973.

Barto, A. G. & Sutton, R. S., 1995, 'Reinforcement learning', *Handbook of brain theory and neural networks*, pp. 804–809.

Bau, D., Zhu, J.-Y., Strobelt, H., Zhou, B., Tenenbaum, J. B., Freeman, W. T. & Torralba, A., 2018, 'Gan dissection: Visualizing and understanding generative adversarial networks', *arXiv preprint arXiv:1811.10597*.

Bellamy, R. K. E., Dey, K., Hind, M., Hoffman, S. C., Houde, S., Kannan, K., Lohia, P., Martino, J., Mehta, S., Mojsilovic, A., Nagar, S., Ramamurthy, K. N., Richards, J., Saha, D., Sattigeri, P., Singh, M., Varshney, K. R. & Zhang, Y., 2018, 'AI Fairness 360: An extensible toolkit for detecting, understanding, and mitigating unwanted algorithmic bias', <https://arxiv.org/abs/1810.01943>.

Berk, R., Heidari, H., Jabbari, S., Kearns, M. & Roth, A., 2021, 'Fairness in criminal justice risk assessments: The state of the art', *Sociological Methods & Research*, vol. 50, no. 1, pp. 3–44.

Besserve, M., Mehrjou, A., Sun, R. & Schölkopf, B., 2020, 'Counterfactuals uncover the modular structure of deep generative models', *8th International Conference on Learning Representations (ICLR 2020)(virtual)*, International Conference on Learning Representations.

Biega, A. J., Gummadi, K. P. & Weikum, G., 2018, 'Equity of attention: Amortizing individual fairness in rankings', *The 41st international acm sigir conference on research & development in information retrieval*, pp. 405–414.

Bingham, E., Chen, J. P., Jankowiak, M., Obermeyer, F., Pradhan, N., Karaletsos, T., Singh, R., Szerlip, P., Horsfall, P. & Goodman, N. D., 2019, 'Pyro: Deep universal probabilistic programming', *The Journal of Machine Learning Research*, vol. 20, no. 1, pp. 973–978.

Blank, J. & Deb, K., 2020, 'Pymoo: Multi-objective optimization in python', *Ieee access*, vol. 8, pp. 89497–89509.

Bliek1ú, C., Bonami, P. & Lodi, A., 2014, 'Solving mixed-integer quadratic programming problems with ibm-cplex: a progress report', *Proceedings of the twenty-sixth RAMP symposium*, pp. 16–17.

Bollen, K. A. & Pearl, J., 2013, 'Eight myths about causality and structural equation models', *Handbook of causal analysis for social research*, Springer, pp. 301–328.

Bottou, L., Peters, J., Quiñonero-Candela, J., Charles, D. X., Chickering, D. M., Portugaly, E., Ray, D., Simard, P. & Snelson, E., 2013, 'Counterfactual reasoning and learning systems: The example of computational advertising', *The Journal of Machine Learning Research*, vol. 14, no. 1, pp. 3207–3260.

Brodersen, K. H., Ong, C. S., Stephan, K. E. & Buhmann, J. M., 2010, 'The balanced accuracy and its posterior distribution', *2010 20th international conference on pattern recognition*, IEEE, pp. 3121–3124.

Caruana, R., Lou, Y., Gehrke, J., Koch, P., Sturm, M. & Elhadad, N., 2015, 'Intelligible Models for HealthCare: Predicting Pneumonia Risk and Hospital 30-day Readmission', *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD '15*, ACM Press, Sydney, NSW, Australia, pp. 1721–1730, viewed 23rd March 2020.

Cawley, G. C., 2008, 'Causal & non-causal feature selection for ridge regression', Guyon, I., Aliferis, C., Cooper, G., Elisseeff, A., Pellet, J.-P., Spirtes, P. & Statnikov, A. (eds.) *Proceedings of the Workshop on the Causation and Prediction Challenge at WCCI 2008*, , vol. 3 of *Proceedings of Machine Learning Research*PMLR, Hong Kong, pp. 107–128.

Chan, D., Ge, R., Gershony, O., Hesterberg, T. & Lambert, D., 2010, 'Evaluating online ad campaigns in a pipeline: causal models at scale', *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 7–16.

Chattopadhyay, A., Manupriya, P., Sarkar, A. & Balasubramanian, V. N., 2019, 'Neural network attributions: A causal perspective', *arXiv preprint arXiv:1902.02302*.

Chen, J., Kallus, N., Mao, X., Svacha, G. & Udell, M., 2019, 'Fairness under unawareness: Assessing disparity when protected class is unobserved', *Proceedings of the conference on fairness, accountability, and transparency*, pp. 339–348.

Cheng, R., Verma, A., Orosz, G., Chaudhuri, S., Yue, Y. & Burdick, J., 2019, 'Control regularization for reduced variance reinforcement learning', *International Conference on Machine Learning*, PMLR, pp. 1141–1150.

Chernozhukov, V., Chetverikov, D., Demirer, M., Duflo, E., Hansen, C., Newey, W. & Robins, J., 2016, 'Double/debiased machine learning for treatment and causal parameters', *arXiv preprint arXiv:1608.00060*.

Chiappa, S., 2019, 'Path-specific counterfactual fairness', *Proceedings of the AAAI Conference on Artificial Intelligence*, , vol. 33pp. 7801–7808.

Chipman, H. A., George, E. I. & McCulloch, R. E., 2007, 'Bayesian ensemble learning', *Advances in neural information processing systems*, pp. 265–272.

Cover, T. M., Thomas, J. A. et al., 1991, 'Entropy, relative entropy and mutual information', *Elements of information theory*, vol. 2, no. 1, pp. 12–13.

Cui, Z., Chen, W., He, Y. & Chen, Y., 2015, 'Optimal action extraction for random forests and boosted trees', *Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*, pp. 179–188.

Dalessandro, B., Perlich, C., Stitelman, O. & Provost, F., 2012, 'Causally motivated attribution for online advertising', *Proceedings of the sixth international workshop on data mining for online advertising and internet economy*, pp. 1–9.

Dandl, S., Molnar, C., Binder, M. & Bischl, B., 2020, 'Multi-objective counterfactual explanations', *arXiv preprint arXiv:2004.11165*.

Darwen, P. J., 2019, 'Bayesian model averaging for river flow prediction', *Applied Intelligence*, vol. 49, no. 1, pp. 103–111.

Deb, K., Pratap, A., Agarwal, S. & Meyarivan, T., 2002a, 'A fast and elitist multiobjective genetic algorithm: Nsga-ii', *IEEE transactions on evolutionary computation*, vol. 6, no. 2, pp. 182–197.

Deb, K., Pratap, A., Agarwal, S. & Meyarivan, T., 2002b, 'A fast and elitist multiobjective genetic algorithm: Nsga-ii', *IEEE transactions on evolutionary computation*, vol. 6, no. 2, pp. 182–197.

Dehejia, R. H. & Wahba, S., 1999, 'Causal effects in nonexperimental studies: Reevaluating the evaluation of training programs', *Journal of the American statistical Association*, vol. 94, no. 448, pp. 1053–1062.

Dehejia, R. H. & Wahba, S., 2002, 'Propensity score-matching methods for nonexperimental causal studies', *Review of Economics and statistics*, vol. 84, no. 1, pp. 151–161.

Dhurandhar, A., Chen, P.-Y., Luss, R., Tu, C.-C., Ting, P., Shanmugam, K. & Das, P., 2018, 'Explanations based on the missing: Towards contrastive explanations with pertinent negatives', *Advances in neural information processing systems*, pp. 592–603.

Dhurandhar, A., Pedapati, T., Balakrishnan, A., Chen, P.-Y., Shanmugam, K. & Puri, R., 2019, 'Model agnostic contrastive explanations for structured data', *arXiv preprint arXiv:1906.00117*.

Dinh, L., Krueger, D. & Bengio, Y., 2014, 'Nice: Non-linear independent components estimation', *arXiv preprint arXiv:1410.8516*.

Dinh, L., Sohl-Dickstein, J. & Bengio, S., 2016, 'Density estimation using real nvp', *International Conference on Learning Representations*, .

Dorie, V., 2016, 'Npci: Non-parametrics for causal inference', *URL: https://github.com/vdorie/npci*.

Doshi-Velez, F. & Kim, B., 2017, 'Towards a rigorous science of interpretable machine learning', *arXiv preprint arXiv:1702.08608*.

Dua, D. & Graff, C., 2017, 'UCI machine learning repository', <http://archive.ics.uci.edu/ml>.

Dudík, M., Erhan, D., Langford, J., Li, L. et al., 2014, 'Doubly robust policy evaluation and optimization', *Statistical Science*, vol. 29, no. 4, pp. 485–511.

Dudík, M., Langford, J. & Li, L., 2011, 'Doubly robust policy evaluation and learning', *Proceedings of the 28th International Conference on International Conference on Machine Learning*, ICML'11, Omnipress, Madison, WI, USA, p. 1097–1104.

Dudley, R. M., 2010, 'Universal donsker classes and metric entropy', *Selected Works of RM Dudley*, Springer, pp. 345–365.

Duong, T. D., Li, Q. & Xu, G., 2021a, 'Prototype-based counterfactual explanation for causal classification', *arXiv preprint arXiv:2105.00703*.

Duong, T. D., Li, Q. & Xu, G., 2021b, 'Stochastic intervention for causal effect estimation', *2021 International Joint Conference on Neural Networks (IJCNN)*, IEEE, pp. 1–8.

Dwork, C., Hardt, M., Pitassi, T., Reingold, O. & Zemel, R., 2012, 'Fairness through awareness', *Proceedings of the 3rd innovations in theoretical computer science conference*, pp. 214–226.

Fang, Z., Cao, Y., Li, Q., Zhang, D., Zhang, Z. & Liu, Y., 2019, 'Joint entity linking with deep reinforcement learning', *The World Wide Web Conference*, pp. 438–447.

Feinberg, V., Wan, A., Stoica, I., Jordan, M. I., Gonzalez, J. E. & Levine, S., 2018, 'Model-based value estimation for efficient model-free reinforcement learning', *arXiv preprint arXiv:1803.00101*.

Fernández-Loría, C., Provost, F. & Han, X., 2020, 'Explaining data-driven decisions made by ai systems: The counterfactual approach', *arXiv preprint arXiv:2001.07417*.

Feydy, J., Séjourné, T., Vialard, F.-X., Amari, S.-i., Trouve, A. & Peyré, G., 2019, 'Interpolating between optimal transport and mmd using sinkhorn divergences', *The 22nd International Conference on Artificial Intelligence and Statistics*, pp. 2681–2690.

Fong, B., 2013, 'Causal theories: A categorical perspective on bayesian networks', *arXiv preprint arXiv:1301.6201*.

Fortin, F.-A. & Parizeau, M., 2013, 'Revisiting the nsga-ii crowding-distance computation', *Proceedings of the 15th annual conference on Genetic and evolutionary computation*, pp. 623–630.

Foss, A. H., Markatou, M. & Ray, B., 2019, 'Distance metrics and clustering methods for mixed-type data', *International Statistical Review*, vol. 87, no. 1, pp. 80–109.

Foster, D. J. & Syrgkanis, V., 2019, 'Orthogonal statistical learning', *arXiv preprint arXiv:1901.09036*.

Friedman, J. H., 2001, 'Greedy function approximation: a gradient boosting machine', *Annals of statistics*, pp. 1189–1232.

Galagate, D., 2016, *Causal inference with a continuous treatment and outcome: alternative estimators for parametric dose-response functions with applications.*, Ph.D. thesis.

Galindo, J. & Tamayo, P., 2000, 'Credit risk assessment using statistical and machine learning: basic methodology and risk modeling applications', *Computational Economics*, vol. 15, no. 1, pp. 107–143.

Girshick, R., 2015, 'Fast r-cnn', *Proceedings of the IEEE international conference on computer vision*, pp. 1440–1448.

Goldberger, A. S. et al., 1964, 'Econometric theory.', *Econometric theory*.

Goldstein, A., Kapelner, A., Bleich, J. & Pitkin, E., 2015, 'Peeking inside the black box: Visualizing statistical learning with plots of individual conditional expec-

tation', *Journal of Computational and Graphical Statistics*, vol. 24, no. 1, pp. 44–65.

Goyal, Y., Feder, A., Shalit, U. & Kim, B., 2020, 'Explaining Classifiers with Causal Concept Effect (CaCE)', *arXiv:1907.07165 [cs, stat]*, arXiv: 1907.07165.

Grath, R. M., Costabello, L., Van, C. L., Sweeney, P., Kamiab, F., Shen, Z. & Lecue, F., 2018, 'Interpretable credit application predictions with counterfactual explanations', *arXiv preprint arXiv:1811.05245*.

Gretton, A., Borgwardt, K. M., Rasch, M. J., Schölkopf, B. & Smola, A., 2012, 'A kernel two-sample test', *The Journal of Machine Learning Research*, vol. 13, no. 1, pp. 723–773.

Grgic-Hlaca, N., Zafar, M. B., Gummadi, K. P. & Weller, A., 2016, 'The case for process fairness in learning: Feature selection for fair decision making', *NIPS Symposium on Machine Learning and the Law*, , vol. 1p. 2.

Grimmer, J., Messing, S. & Westwood, S. J., 2017, 'Estimating heterogeneous treatment effects and the effects of heterogeneous treatments with ensemble methods', *Political Analysis*, vol. 25, no. 4, pp. 413–434.

Gruber, S. & van der Laan, M., 2012, 'tmle: An r package for targeted maximum likelihood estimation', *Journal of Statistical Software*, vol. 51, pp. 1–35.

Gruber, S. & Van der Laan, M. J., 2011, 'tmle: An r package for targeted maximum likelihood estimation', .

Guelman, L. & Guelman, M. L., 2014, 'Package 'uplift"', *CRAN*.

Guidotti, R., Monreale, A., Ruggieri, S., Pedreschi, D., Turini, F. & Giannotti, F., 2018, 'Local rule-based explanations of black box decision systems', *arXiv preprint arXiv:1805.10820*.

Guillaume, S., 2001, 'Designing fuzzy inference systems from data: An interpretability-oriented review', *IEEE Transactions on fuzzy systems*, vol. 9, no. 3, pp. 426–443.

Hansotia, B. & Rukstales, B., 2002, 'Incremental value modeling', *Journal of Interactive Marketing*, vol. 16, no. 3, p. 35.

Harradon, M., Druce, J. & Ruttenberg, B., 2018, 'Causal learning and explanation of deep neural networks via autoencoded activations', *arXiv preprint arXiv:1802.00541*.

Hill, J. L., 2011, 'Bayesian nonparametric modeling for causal inference', *Journal of Computational and Graphical Statistics*, vol. 20, no. 1, pp. 217–240.

Hirano, K., Imbens, G. W. & Ridder, G., 2003, 'Efficient estimation of average treatment effects using the estimated propensity score', *Econometrica*, vol. 71, no. 4, pp. 1161–1189.

Hitsch, G. J. & Misra, S., 2018, 'Heterogeneous treatment effects and optimal targeting policy evaluation', *Available at SSRN 3111957*.

Ho, J., Chen, X., Srinivas, A., Duan, Y. & Abbeel, P., 2019, 'Flow++: Improving flow-based generative models with variational dequantization and architecture design', *International Conference on Machine Learning*, PMLR, pp. 2722–2730.

Hoerl, A. E. & Kennard, R. W., 1970, 'Ridge regression: Biased estimation for nonorthogonal problems', *Technometrics*, vol. 12, no. 1, pp. 55–67.

Hoffman, R. R., Mueller, S. T., Klein, G. & Litman, J., 2018, 'Metrics for explainable ai: Challenges and prospects', *arXiv preprint arXiv:1812.04608*.

Höltgen, B., Schut, L., Brauner, J. M. & Gal, Y., 2021, 'Deduce: Generating counterfactual explanations at scale', *eXplainable AI approaches for debugging and diagnosis.*, .

Hoogeboom, E., Cohen, T. S. & Tomczak, J. M., 2020, 'Learning discrete distributions by dequantization', *arXiv preprint arXiv:2001.11235*.

Hvilshøj, F., Iosifidis, A. & Assent, I., 2021, 'Ecinn: efficient counterfactuals from invertible neural networks', *arXiv preprint arXiv:2103.13701*.

Imbens, G. W. & Rubin, D. B., 2015, *Causal inference in statistics, social, and biomedical sciences*, Cambridge University Press.

Izmailov, P., Kirichenko, P., Finzi, M. & Wilson, A. G., 2020, 'Semi-supervised learning with normalizing flows', *International Conference on Machine Learning*, PMLR, pp. 4615–4630.

Jang, J.-S. & Sun, C.-T., 1993, 'Functional equivalence between radial basis function networks and fuzzy inference systems', *IEEE transactions on Neural Networks*, vol. 4, no. 1, pp. 156–159.

Jaskowski, M. & Jaroszewicz, S., 2012, 'Uplift modeling for clinical trial data', *ICML Workshop on Clinical Data Analysis*, .

Jia, H., Cheung, Y.-m. & Liu, J., 2015, 'A new distance metric for unsupervised learning of categorical data', *IEEE transactions on neural networks and learning systems*, vol. 27, no. 5, pp. 1065–1079.

Kadir, T. & Brady, M., 2001, 'Saliency, scale and image description', *Int. J. Comput. Vision*, vol. 45, no. 2, p. 83–105.

Kanamori, K., Takagi, T., Kobayashi, K. & Arimura, H., 2020, 'Dace: Distribution-aware counterfactual explanation by mixed-integer linear optimization', *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20, Christian Bessiere (Ed.). International Joint Conferences on Artificial Intelligence Organization*, pp. 2855–2862.

Kaufman, L. & Rousseeuw, P. J., 2009, *Finding groups in data: an introduction to cluster analysis*, vol. 344, John Wiley & Sons.

Kaur, H., Nori, H., Jenkins, S., Caruana, R., Wallach, H. & Wortman Vaughan, J., 2020, 'Interpreting interpretability: Understanding data scientists' use of interpretability tools for machine learning', *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, pp. 1–14.

Kennedy, E. H., Ma, Z., McHugh, M. D. & Small, D. S., 2017, 'Nonparametric methods for doubly robust estimation of continuous treatment effects', *Journal of the Royal Statistical Society. Series B, Statistical Methodology*, vol. 79, no. 4, p. 1229.

Kim, B., Wattenberg, M., Gilmer, J., Cai, C. J., Wexler, J., Viégas, F. B. & Sayres, R., 2017, 'Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (tcav)', *ICML*, .

Kingma, D. P. & Ba, J., 2014, 'Adam: A method for stochastic optimization', *arXiv preprint arXiv:1412.6980*.

Koh, P. W. & Liang, P., 2017, 'Understanding black-box predictions via influence functions', *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, JMLR. org, pp. 1885–1894.

Kohavi, R. & Longbotham, R., 2011, 'Unexpected results in online controlled experiments', *ACM SIGKDD Explorations Newsletter*, vol. 12, no. 2, pp. 31–35.

Kusner, M. J., Loftus, J., Russell, C. & Silva, R., 2017, 'Counterfactual fairness', *Advances in neural information processing systems*, vol. 30.

Künzel, S. R., Sekhon, J. S., Bickel, P. J. & Yu, B., n.d., 'Metalearners for estimating heterogeneous treatment effects using machine learning', *Proceedings of the National Academy of Sciences*, vol. 116.

Lakkaraju, H., Bach, S. H. & Leskovec, J., 2016, 'Interpretable decision sets: A joint framework for description and prediction', *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pp. 1675–1684.

Larson, J., Mattu, S., Kirchner, L. & Angwin, J., 2016, 'How we analyzed the compas recidivism algorithm', *ProPublica (5 2016)*, vol. 9, no. 1.

Lash, M. T., Lin, Q., Street, N., Robinson, J. G. & Ohlmann, J., 2017, 'Generalized inverse classification', *Proceedings of the 2017 SIAM International Conference on Data Mining*, SIAM, pp. 162–170.

Laugel, T., Lesot, M.-J., Marsala, C., Renard, X. & Detyniecki, M., 2017, 'Inverse classification for comparison-based interpretability in machine learning', *arXiv preprint arXiv:1712.08443*.

Laugel, T., Lesot, M.-J., Marsala, C., Renard, X. & Detyniecki, M., 2018, 'Comparison-based inverse classification for interpretability in machine learning', *International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems*, Springer, pp. 100–111.

Letham, B., Rudin, C., McCormick, T. H. & Madigan, D., 2015a, 'Interpretable classifiers using rules and Bayesian analysis: Building a better stroke prediction model', *The Annals of Applied Statistics*, vol. 9, no. 3, pp. 1350–1371, arXiv: 1511.01644.

Letham, B., Rudin, C., McCormick, T. H., Madigan, D. et al., 2015b, 'Interpretable classifiers using rules and bayesian analysis: Building a better stroke prediction model', *The Annals of Applied Statistics*, vol. 9, no. 3, pp. 1350–1371.

Li, Q., Duong, T. D., Wang, Z., Liu, S., Wang, D. & Xu, G., 2021a, 'Causal-aware generative imputation for automated underwriting', *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, pp. 3916–3924.

Li, Q., Wang, X. & Xu, G., 2021b, 'Be causal: De-biasing social network confounding in recommendation', *arXiv preprint arXiv:2105.07775*.

Li, Q., Wang, Z., Li, G., Cao, Y., Xiong, G. & Guo, L., 2017, 'Learning robust low-rank approximation for crowdsourcing on riemannian manifold', *Procedia Computer Science*, vol. 108, pp. 285–294.

Li, Q., Wang, Z., Li, G., Pang, J. & Xu, G., 2021c, 'Hilbert sinkhorn divergence for optimal transport', *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3835–3844.

Li, Q., Wang, Z., Liu, S., Li, G. & Xu, G., 2021d, 'Causal optimal transport for treatment effect estimation', *IEEE Transactions on Neural Networks and Learning Systems*.

Li, X., Cao, Y., Li, Q., Shang, Y., Li, Y., Liu, Y. & Xu, G., 2021e, 'Rlink: Deep reinforcement learning for user identity linkage', *World Wide Web*, vol. 24, no. 1, pp. 85–103.

Liashchynskyi, P. & Liashchynskyi, P., 2019, 'Grid search, random search, genetic algorithm: a big comparison for nas', *arXiv preprint arXiv:1912.06059*.

Liaw, A., Wiener, M. et al., 2002, 'Classification and regression by randomforest', *R news*, vol. 2, no. 3, pp. 18–22.

Lundberg, S. M. & Lee, S.-I., 2017, 'A unified approach to interpreting model predictions', *Advances in neural information processing systems*, pp. 4765–4774.

Luo, Z., Gardiner, J. C. & Bradley, C. J., 2010, 'Applying propensity score methods in medical research: pitfalls and prospects', *Medical Care Research and Review*, vol. 67, no. 5, pp. 528–554.

Maas, A. L., Hannun, A. Y., Ng, A. Y. et al., 2013, 'Rectifier nonlinearities improve neural network acoustic models', *Proc. icml*, , vol. 30Citeseer, p. 3.

Madumal, P., Miller, T., Sonenberg, L. & Vetere, F., 2019, 'Explainable reinforcement learning through a causal lens', *arXiv preprint arXiv:1905.10958*.

Magrini, A., Di Blasi, S. & Stefanini, F. M., 2017, 'A conditional linear gaussian network to assess the impact of several agronomic settings on the quality of tuscan sangiovese grapes', *Biometrical Letters*, vol. 54, no. 1, pp. 25–42.

Mahajan, D., Tan, C. & Sharma, A., 2019, 'Preserving causal constraints in counterfactual explanations for machine learning classifiers', *arXiv preprint arXiv:1912.03277*.

Manahan, C., 2005, 'A proportional hazards approach to campaign list selection', *SAS User Group International (SUGI) 30 Proceedings*.

Mania, H., Guy, A. & Recht, B., 2018, 'Simple random search of static linear policies is competitive for reinforcement learning', Bengio, S., Wallach, H., Larochelle, H., Grauman, K., Cesa-Bianchi, N. & Garnett, R. (eds.) *Advances in Neural Information Processing Systems*, , vol. 31.

Martens, E. P., Pestman, W. R., de Boer, A., Belitser, S. V. & Klungel, O. H., 2008, 'Systematic differences in treatment effect estimates between propensity score methods and logistic regression', *International journal of epidemiology*, vol. 37, no. 5, pp. 1142–1147.

McDiarmid, C. et al., 1989, 'On the method of bounded differences', *Surveys in combinatorics*, vol. 141, no. 1, pp. 148–188.

Miconi, T., 2017, 'The impossibility of" fairness": a generalized impossibility result for decisions', *arXiv preprint arXiv:1707.01195*.

Moore, J., Hammerla, N. & Watkins, C., 2019, 'Explaining deep learning models with constrained adversarial examples', *Pacific Rim International Conference on Artificial Intelligence*, Springer, pp. 43–56.

Mothilal, R. K., Sharma, A. & Tan, C., 2020a, 'Explaining machine learning classifiers through diverse counterfactual explanations', *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, ACM, Barcelona Spain, pp. 607–617, viewed 11th March 2020.

Mothilal, R. K., Sharma, A. & Tan, C., 2020b, 'Explaining machine learning classifiers through diverse counterfactual explanations', *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, pp. 607–617.

Mukherjee, D., Yurochkin, M., Banerjee, M. & Sun, Y., 2020, 'Two simple ways to learn individual fairness metrics from data', *International Conference on Machine Learning*, PMLR, pp. 7097–7107.

Nabi, R. & Shpitser, I., 2018, 'Fair inference on outcomes', *Proceedings of the AAAI Conference on Artificial Intelligence*, , vol. 32.

Narendra, T., Sankaran, A., Vijaykeerthy, D. & Mani, S., 2018, 'Explaining deep learning models using causal inference', *arXiv preprint arXiv:1811.04376*.

Natekin, A. & Knoll, A., 2013, 'Gradient boosting machines, a tutorial', *Frontiers in neurorobotics*, vol. 7, p. 21.

Ng, A. et al., 2011, 'Sparse autoencoder', *CS294A Lecture notes*, vol. 72, no. 2011, pp. 1–19.

Oh, J. H., Pouryahya, M., Iyer, A., Apte, A. P., Tannenbaum, A. & Deasy, J. O., 2019, 'Kernel wasserstein distance', *arXiv preprint arXiv:1905.09314*.

Oprescu, M., Syrgkanis, V. & Wu, Z. S., 2018, 'Orthogonal random forest for causal inference', *arXiv preprint arXiv:1806.03467*.

Pawelczyk, M., Bielawski, S., Heuvel, J. v. d., Richter, T. & Kasneci, G., 2021, 'Carla: a python library to benchmark algorithmic recourse and counterfactual explanation algorithms', *arXiv preprint arXiv:2108.00783*.

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

Pearl, J., 2003, 'Statistics and causal inference: A review', *Test*, vol. 12, no. 2, pp. 281–345.

Pearl, J., 2009a, 'Causal inference in statistics: An overview', *Statistics Surveys*, vol. 3, pp. 96–146.

Pearl, J., 2009b, *Causality*, Cambridge university press.

Pearl, J., 2010, 'Causal inference', *Causality: Objectives and Assessment*, pp. 39–58.

Pearl, J., 2012, 'The causal foundations of structural equation modeling', Tech. rep., CALIFORNIA UNIV LOS ANGELES DEPT OF COMPUTER SCIENCE.

Pearl, J. & Mackenzie, D., 2018, *The Book of Why: The New Science of Cause and Effect*, 1st edn., Basic Books, Inc., USA.

Pearl, J. et al., 2000, 'Models, reasoning and inference', *Cambridge, UK: Cambridge-UniversityPress*.

Pearl, J. et al., 2009, 'Causal inference in statistics: An overview', *Statistics surveys*, vol. 3, pp. 96–146.

Peters, J., Bühlmann, P. & Meinshausen, N., 2015, 'Causal inference using invariant prediction: identification and confidence intervals', *arXiv e-prints*, arXiv:1501.01332.

Peters, J., Bühlmann, P. & Meinshausen, N., 2016, 'Causal inference by using invariant prediction: identification and confidence intervals', *Journal of the Royal Statistical Society. Series B (Statistical Methodology)*, pp. 947–1012.

Pirracchio, R., Carone, M., Rigon, M. R., Caruana, E., Mebazaa, A. & Chevret, S., 2016, 'Propensity score estimators for the average treatment effect and the average treatment effect on the treated may yield very different estimates', *Statistical methods in medical research*, vol. 25, no. 5, pp. 1938–1954.

Poyiadzi, R., Sokol, K., Santos-Rodriguez, R., De Bie, T. & Flach, P., 2020, 'Face: Feasible and actionable counterfactual explanations', *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*, pp. 344–350.

Qiang, W. & Zhongli, Z., 2011, 'Reinforcement learning model, algorithms and its application', *2011 International Conference on Mechatronic Science, Electric Engineering and Computer (MEC)*, IEEE, pp. 1143–1146.

Research, M., 2019, 'EconML: A Python Package for ML-Based Heterogeneous Treatment Effects Estimation', https://github.com/microsoft/EconML, version 0.x.

Ribeiro, M. T., Singh, S. & Guestrin, C., 2016, '" Why should i trust you?" Explaining the predictions of any classifier', *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pp. 1135–1144.

Rosenbaum, P. R. & Rubin, D. B., 1983, 'The central role of the propensity score in observational studies for causal effects', *Biometrika*, vol. 70, no. 1, pp. 41–55.

Rubin, D. B., 1974, 'Estimating causal effects of treatments in randomized and nonrandomized studies.', *Journal of educational Psychology*, vol. 66, no. 5, p. 688.

Rüschendorf, L., 1985, 'The wasserstein distance and approximation theorems', *Probability Theory and Related Fields*, vol. 70, no. 1, pp. 117–129.

Russell, C., 2019a, 'Efficient Search for Diverse Coherent Explanations', *Proceedings of the Conference on Fairness, Accountability, and Transparency - FAT* '19*, ACM Press, Atlanta, GA, USA, pp. 20–28, viewed 23rd March 2020.

Russell, C., 2019b, 'Efficient search for diverse coherent explanations', *Proceedings of the Conference on Fairness, Accountability, and Transparency*, pp. 20–28.

Russell, C., Kusner, M. J., Loftus, J. R. & Silva, R., 2017, 'When worlds collide: integrating different counterfactual assumptions in fairness', *Advances in Neural Information Processing Systems 30. Pre-proceedings*, vol. 30.

Scheines, R., 1997, 'An introduction to causal inference', .

Schwab, P. & Karlen, W., 2019a, 'CXPlain: Causal Explanations for Model Interpretation under Uncertainty', *arXiv:1910.12336 [cs, stat]*, arXiv: 1910.12336.

Schwab, P. & Karlen, W., 2019b, 'Cxplain: Causal explanations for model interpretation under uncertainty', *arXiv preprint arXiv:1910.12336*.

Schwab, P., Miladinovic, D. & Karlen, W., 2019, 'Granger-Causal Attentive Mixtures of Experts: Learning Important Features with Neural Networks', *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, pp. 4846–4853.

Sharifi-Malvajerdi, S., Kearns, M. & Roth, A., 2019, 'Average individual fairness: Algorithms, generalization and experiments', *Advances in Neural Information Processing Systems*, vol. 32, pp. 8242–8251.

Sharma, A. & Kiciman, E., 2020, 'Dowhy: An end-to-end library for causal inference', *arXiv preprint arXiv:2011.04216*.

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

Sharma, S., Henderson, J. & Ghosh, J., 2020, 'Certifai: A common framework to provide explanations and analyse the fairness and robustness of black-box models', *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*, pp. 166–172.

Shimizu, S., 2014, 'Lingam: Non-gaussian methods for estimating causal structures', *Behaviormetrika*, vol. 41, pp. 65–98.

Snell, J., Swersky, K. & Zemel, R., 2017, 'Prototypical networks for few-shot learning', *Advances in neural information processing systems*, vol. 30.

Solomatine, D. P. & Shrestha, D. L., 2004, 'Adaboost. rt: a boosting algorithm for regression problems', *2004 IEEE International Joint Conference on Neural Networks (IEEE Cat. No. 04CH37541)*, , vol. 2IEEE, pp. 1163–1168.

Sołtys, M., Jaroszewicz, S. & Rzepakowski, P., 2015, 'Ensemble methods for uplift modeling', *Data mining and knowledge discovery*, vol. 29, no. 6, pp. 1531–1559.

Speicher, T., Heidari, H., Grgic-Hlaca, N., Gummadi, K. P., Singla, A., Weller, A. & Zafar, M. B., 2018, 'A unified approach to quantifying algorithmic unfairness: Measuring individual &group unfairness via inequality indices', *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 2239–2248.

Stuart, E. A., King, G., Imai, K. & Ho, D., 2011, 'Matchit: nonparametric preprocessing for parametric causal inference', *Journal of statistical software*.

Sundararajan, M., Taly, A. & Yan, Q., 2017, 'Axiomatic attribution for deep networks', *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, JMLR. org, pp. 3319–3328.

Tian, J., 2008, 'Identifying dynamic sequential plans', *Proceedings of the Twenty-Fourth Conference on Uncertainty in Artificial Intelligence*, pp. 554–561.

Ustun, B., Spangher, A. & Liu, Y., 2019, 'Actionable recourse in linear classification', *Proceedings of the Conference on Fairness, Accountability, and Transparency*, pp. 10–19.

van de Velden, M., Iodice D'Enza, A. & Markos, A., 2019, 'Distance-based clustering of mixed data', *Wiley Interdisciplinary Reviews: Computational Statistics*, vol. 11, no. 3, p. e1456.

Van Looveren, A. & Klaise, J., 2019, 'Interpretable counterfactual explanations guided by prototypes', *arXiv preprint arXiv:1907.02584*.

Van Looveren, A. & Klaise, J., 2020, 'Interpretable Counterfactual Explanations Guided by Prototypes', *arXiv:1907.02584 [cs, stat]*, arXiv: 1907.02584.

VanderWeele, T. J., 2009, 'Concerning the consistency assumption in causal inference', *Epidemiology*, vol. 20, no. 6, pp. 880–883.

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

Wachter, S., Mittelstadt, B. & Russell, C., 2018, 'Counterfactual Explanations without Opening the Black Box: Automated Decisions and the GDPR', *arXiv:1711.00399 [cs]*, arXiv: 1711.00399.

Wager, S. & Athey, S., 2018, 'Estimation and inference of heterogeneous treatment effects using random forests', *Journal of the American Statistical Association*, vol. 113, no. 523, pp. 1228–1242.

Wang, J.-S. & Lee, C. G., 2002, 'Self-adaptive neuro-fuzzy inference systems for classification applications', *IEEE Transactions on Fuzzy Systems*, vol. 10, no. 6, pp. 790–802.

Wang, T., 2017, 'Multi-Value Rule Sets', *arXiv:1710.05257 [cs], NIPS*, arXiv: 1710.05257.

Wang, X., Li, Q., Zhang, W., Xu, G., Liu, S. & Zhu, W., 2020, 'Joint relational dependency learning for sequential recommendation', *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, Springer, pp. 168–180.

Wang, Z., Li, Q., Li, G. & Xu, G., 2019, 'Polynomial representation for persistence diagram', *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6123–6132.

Whitehead, S. D. & Ballard, D. H., 1991, 'Learning to perceive and act by trial and error', *Machine Learning*, vol. 7, no. 1, pp. 45–83.

Whitley, D., 1994, 'A genetic algorithm tutorial', *Statistics and computing*, vol. 4, no. 2, pp. 65–85.

Wightman, L. F., 1998, 'Lsac national longitudinal bar passage study. lsac research report series.', .

Williams, J. J., Kim, J., Rafferty, A., Maldonado, S., Gajos, K. Z., Lasecki, W. S. & Heffernan, N., 2016, 'Axis: Generating explanations at scale with learnersourcing and machine learning', *Proceedings of the Third (2016) ACM Conference on Learning@ Scale*, pp. 379–388.

Winkler, C., Worrall, D., Hoogeboom, E. & Welling, M., 2019, 'Learning likelihoods with conditional normalizing flows', *arXiv preprint arXiv:1912.00042*.

Wu, Y., Zhang, L. & Wu, X., 2019, 'Counterfactual fairness: Unidentification, bound and algorithm', *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence*, .

Xian, Y., Fu, Z., Muthukrishnan, S., De Melo, G. & Zhang, Y., 2019, 'Reinforcement knowledge graph reasoning for explainable recommendation', *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 285–294.

Xu, G., Duong, T., Li, Q., Liu, S. & Wang, X., 2020, 'Causality learning: A new perspective for interpretable machine learning', *IEEE Intelligent Informatics Bulletin*.

Yao, L., Li, S., Li, Y., Huai, M., Gao, J. & Zhang, A., 2018, 'Representation learning for treatment effect estimation from observational data', *Advances in Neural Information Processing Systems*, vol. 31.

Yin, J., Li, Q., Liu, S., Wu, Z. & Xu, G., 2021, 'Leveraging multi-level dependency of relational sequences for social spammer detection', *Neurocomputing*, vol. 428, pp. 130–141.

Zaniewicz, Ł. & Jaroszewicz, S., 2013, 'Support vector machines for uplift modeling', *2013 IEEE 13th International Conference on Data Mining Workshops*, IEEE, pp. 131–138.

Završnik, A., 2021, 'Algorithmic justice: Algorithms and big data in criminal justice settings', *European Journal of criminology*, vol. 18, no. 5, pp. 623–642.

Zhang, J. & Bareinboim, E., 2018, 'Fairness in decision-making—the causal explanation formula', *Proceedings of the AAAI Conference on Artificial Intelligence*, , vol. 32.

Zhang, X., Tan, S., Koch, P., Lou, Y., Chajewska, U. & Caruana, R., 2019a, 'Axiomatic interpretability for multiclass additive models', *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 226–234.

Zhang, X., Tan, S., Koch, P., Lou, Y., Chajewska, U. & Caruana, R., 2019b, 'Axiomatic Interpretability for Multiclass Additive Models', *Proceedings of the 25th*

ACM SIGKDD International Conference on Knowledge Discovery & Data Mining - KDD '19, ACM Press, Anchorage, AK, USA, pp. 226–234, viewed 11th March 2020.

Zhang, Y. & Zhou, L., 2019, 'Fairness assessment for artificial intelligence in financial industry', *arXiv preprint arXiv:1912.07211*.

Zhao, Q. & Hastie, T., 2019, 'Causal interpretations of black-box models', *Journal of Business & Economic Statistics*, vol. 0, no. 0, pp. 1–10.

Zhao, Q. & Hastie, T., 2021, 'Causal interpretations of black-box models', *Journal of Business & Economic Statistics*, vol. 39, no. 1, pp. 272–281.

Zhao, Y., Fang, X. & Simchi-Levi, D., 2017, 'Uplift modeling with multiple treatments and general response types', *Proceedings of the 2017 SIAM International Conference on Data Mining*, SIAM, pp. 588–596.

Zheng, A. & Casari, A., 2018, *Feature engineering for machine learning: principles and techniques for data scientists*, " O'Reilly Media, Inc.".

Zhou Wang, Bovik, A. C., Sheikh, H. R. & Simoncelli, E. P., 2004, 'Image quality assessment: from error visibility to structural similarity', *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612.