FairPFN: Transformers Can do Counterfactual Fairness

Jake Robertson [1 2] **Noah Hollmann** [3] **Noor Awad** [1] **Frank Hutter** [1 4]

Abstract

Machine Learning systems are increasingly prevalent across healthcare, law enforcement, and finance but often operate on historical data, which may carry biases against certain demographic groups. Causal and counterfactual fairness provides an intuitive way to define fairness that closely aligns with legal standards. Despite its theoretical benefits, counterfactual fairness comes with several practical limitations, largely related to the reliance on domain knowledge and approximate causal discovery techniques in constructing a causal model. In this study, we take a fresh perspective on counterfactually fair prediction, building upon recent work in in-context-learning (ICL) and prior-fitted networks (PFNs) to learn a transformer called FairPFN. This model is pretrained using synthetic fairness data to eliminate the causal effects of protected attributes directly from observational data, removing the requirement of access to the correct causal model in practice. In our experiments, we thoroughly assess the effectiveness of FairPFN in eliminating the causal impact of protected attributes on a series of synthetic case studies and real-world datasets. Our findings pave the way for a new and promising research area: transformers for causal and counterfactual fairness.

1. Introduction

Algorithmic bias is one of the most pressing AI-related risks, arising when ML-assisted decisions produce discriminatory outcomes towards historically underprivileged demographic groups (Angwin et al., 2016). Despite the topic of fairness receiving significant attention in the ML community, various critics from outside the fairness community argue that statistical measures of fairness and current methods to optimize them are largely misguided in terms of their context-dependence and transferability to effective legislation. Recent work in causal fairness has proposed the popular notion of counterfactual fairness, which provides the intuition that outcomes are the same in the real world as in the counterfactual world where *protected attributes* - such as gender, ethnicity, or sexual orientation - take on a different value. According to a recent review contrasting observational and causal fairness metrics (Castelnovo et al., 2022), the non-identifiability of causal models from observational data (Peters et al., 2012) presents a significant challenge in applying causal fairness in practice, as causal mechanisms are often complex due to the intricate nature of bias in real-world datasets. If causal model assumptions are incorrect - for example, when a covariate is assumed not to be influenced by a protected attribute when in fact it is - proposing the wrong causal graph can provide a false sense of security and trust (Ma et al., 2023).

In this study, we introduce a novel approach to counterfactual fairness based on the recently proposed TabPFN. Our transformer-based approach coined FairPFN, is pre-trained on a synthetic benchmark of causally generated data and learns to identify and remove the causal effect of protected attributes. In our experimental results across a series of synthetic case-studies and real-world datasets, we demonstrate the effectiveness, flexibility, and extensibility of transformers for causal and counterfactual fairness.

2. Background

Algorithmic Fairness Algorithmic bias occurs when past discrimination against a demographic group such as ethnicity or sex is reflected in the training data of an ML algorithm. In such cases, ML algorithms are well known to reproduce and even amplify this bias in their predictions (Barocas et al., 2023). Fairness as a topic of research concerns the measurement of algorithmic bias and the development of principled methods that produce non-discriminatory predicted outcomes.

Causal Fairness Analysis Causal ML is a new and emerging research field that aims to represent data-generating processes and prediction problems in the language of causality, offering support for causal modeling, mediation analysis, and counterfactual explanations. The Causal Fairness Anal-

[1] University of Freiburg, Freiburg, Germany [2] Zuse School ELIZA, Darmstadt, Germany [3] Charité, Berlin, Germany [4] ELLIS Institute Tübingen, Tübingen, Germany. Correspondence to: Jake Robertson <robertsj@cs.uni-freiburg.de>.

Next Gen AI Safety at the International Conference on Machine Learning 2024, Vienna, Austria. Copyright 2024 by the author(s).

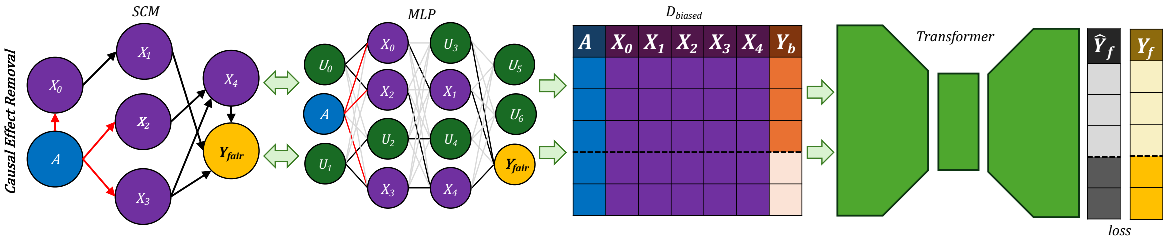

Figure 1. **FairPFN Pre-training:** FairPFN is pre-trained on a synthetic prior of datasets generated from sparse SCMs with exogenous protected attributes. A biased dataset is generated and passed as context to the transformer, and the loss is calculated with respect to the fair outcomes calculated by removing the causal influence of the protected attribute.

ysis (CFA) framework (Plecko & Bareinboim, 2022) draws parallels between causal modeling and legal doctrines of direct and indirect discrimination. By categorizing variables into protected attributes A, mediators X_{med}, confounders X_{conf}, and outcomes Y, the CFA defines the Fairness Cookbook of causal fairness metrics: the Direct Effect (DE), Indirect Effect (IE), and Spurious Effect (SE). These metrics facilitate mediation analysis to assess the remaining causal effects of various bias-mitigation approaches.

Counterfactual Fairness A related causal concept of fairness is counterfactual fairness (Kusner et al., 2017), which requires that outcomes remain the same in both the real world and a counterfactual world where a protected attribute assumes a different value. Given a causal graph, counterfactual fairness can be obtained either by fitting to observable non-descendants (Level-One), the inferred values of an exogenous unobserved variable (Level-Two) or the noise terms of an Additive Noise Model for observable variables (Level-Three). Counterfactual fairness has gained significant popularity in the fairness community, inspiring recent work on path-specific extensions (Peters et al., 2014) and the application of Variational Autoencoders (VAEs) to achieve counterfactually fair latent representations[1] (Ma et al., 2023).

A key challenge in the CFA, counterfactual fairness, and causal ML, in general, is the assumption regarding the prior knowledge of causal graphs and models, which relies heavily on domain knowledge and approximate causal discovery techniques. In the context of fairness, (Castelnovo et al., 2022) argue that it is challenging to obtain causal graphs representing complex systemic inequalities. Additionally, (Ma et al., 2023) demonstrate that proposing an incorrect causal graph or model can deteriorate counterfactual fairness and potentially lead to adverse impacts (e.g. fairwashing) if the causal relationships between protected attributes and other variables are incorrectly assumed.

Prior-Fitted Networks Prior-Fitted Networks (PFNs) are a recent approach to incorporating prior knowledge into neural networks via pre-training on datasets sampled from a prior distribution (Müller et al., 2021). This allows PFNs to perform well on downstream tasks with limited data.

TabPFN (Hollmann et al., 2022), a recent application of PFNs to small, tabular classification problems, trains a transformer on a hypothesis of synthetic datasets generated from sparse SCMs, achieving state-of-the-art results by integrating over the simplest causal explanations for the data in a single forward pass of the network.

3. Methodology

In this section, we introduce FairPFN, a novel bias mitigation technique that synergizes concepts from prior-fitted networks (PFNs) with principles of causal and counterfactual fairness. FairPFN aims to eliminate the causal and counterfactual effects of protected attributes using only observational data.

Synthetic Prior Data Generation The main methodological contribution of FairPFN is its fairness prior, designed to represent the causal mechanisms of bias in real-world data. FairPFN's fairness prior includes a key addition to the TabPFN hypothesis space, namely the inclusion and specification of protected attributes in the randomly generated SCMs as *exogenous* variables[2].

The first step of FairPFN is the generation of *biased* synthetic datasets that realistically represent the causal mechanisms of bias in real-world datasets. We provide a visual overview of this process in (Figure 1). Taking inspiration from TabPFN, we represent SCMs as Multi-Layer-Perceptrons (MLPs) with linear layers serving to represent the structural equation $f = P \cdot W^T x + \epsilon$ where W are the weights of the activations, ϵ is Gaussian Noise, and P is a dropout mask sampled from a log-scale to encourage

[1] CLAIRE is not included as a baseline as their training code or model is not publicly available.

[2] The simplifying assumption of exogenous protected attributions is commonly made in the causal fairness literature as protected attributes are typically unchangeable by definition and hold ancestral closure (Plecko & Bareinboim, 2022)

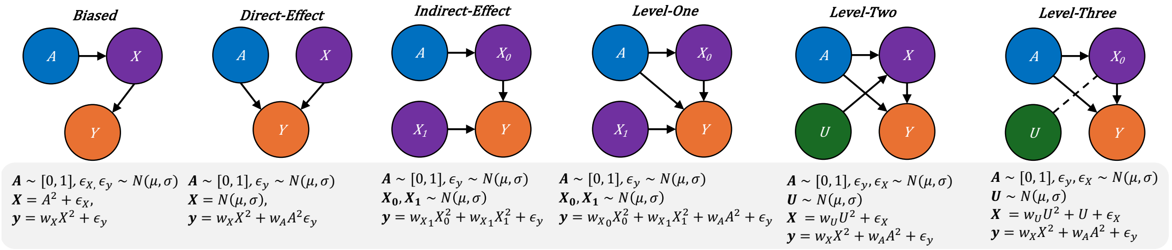
Biased
A
X
Y
$A \sim [0,1], \epsilon_{X,} \epsilon_y \sim N(\mu,\sigma)$
$X = A^2 + \epsilon_X,$
$y = w_X X^2 + \epsilon_y$
Direct-Effect
A
X
Y
$A \sim [0,1], \epsilon_y \sim N(\mu,\sigma)$
$X = N(\mu,\sigma),$
$y = w_X X^2 + w_A A^2 \epsilon_y$
Indirect-Effect
A
X_0
X_1
Y
$A \sim [0,1], \epsilon_y \sim N(\mu,\sigma)$
$X_0, X_1 \sim N(\mu,\sigma)$
$y = w_{X_1} X_0^2 + w_{X_1} X_1^2 + \epsilon_y$
Level-One
A
X_0
X_1
Y
$A \sim [0,1], \epsilon_y \sim N(\mu,\sigma)$
$X_0, X_1 \sim N(\mu,\sigma)$
$y = w_{X_0} X_0^2 + w_{X_1} X_1^2 + w_A A^2 + \epsilon_y$
Level-Two
A
X
U
Y
$A \sim [0,1], \epsilon_y, \epsilon_X \sim N(\mu,\sigma)$
$U \sim N(\mu,\sigma)$
$X = w_U U^2 + \epsilon_X$
$y = w_X X^2 + w_A A^2 + \epsilon_y$
Level-Three
A
X_0
U
Y
$A \sim [0,1], \epsilon_y, \epsilon_X \sim N(\mu,\sigma)$
$U \sim N(\mu,\sigma)$
$X = w_U U^2 + U + \epsilon_X$
$y = w_X X^2 + w_A A^2 + \epsilon_y$

Figure 2. **Causal Case Studies:** Visualization and data generating processes of synthetic causal case studies, a handcrafted set of benchmarks designed to evaluate FairPFN's ability to remove various sources of bias in causally generated data.

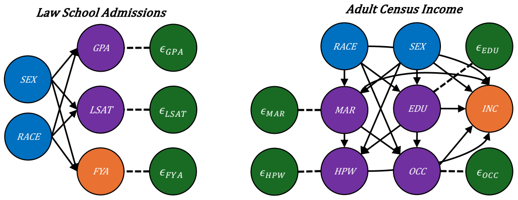

Figure 3. **Real-World Datasets**: Causal graphs of real-world datasets Law School Admissions and Adult Census Income.

sparsity of the represented SCM.

The exogenous protected attribute is sampled from the input to the MLP as a binary variable $A \in \{a_0, a_1\}$ where a_i is sampled from the same range as non-protected exogenous variables U_{fair} to prevent numeric overflow. We uniformly sample m features X from the second hidden layer on to ensure that they contain rich representations of the causes. Finally, we select the target Y from the output layer. Because Y is a continuous variable, we binarize over a random threshold. We note that without binarizing, future versions of FairPFN are extensible to regression tasks and handling multiple protected attributes.

Via a forward pass of the MLP, we generate a dataset $D_{bias} = (A, X_{bias}, Y_{bias})$ of n samples and repeat this process throughout training on randomly sampled SCMs, number of features, and number of samples to generate a rich synthetic representation of real-world, biased data.

FairPFN Pre-training The strategy by which we pre-train the transformer to perform counterfactual fairness is by generating two datasets, D_{bias} and D_{fair}. The fair dataset is generated by performing dropout on the outgoing edges of the protected attribute in the sampled MLP. This has the effect of setting the weight to 0 in the represented equation $f = 0 \cdot wx + \epsilon$, meaning that the effect of the protected attribute is reduced to Gaussian noise ϵ as visualized in Figure 11.[3]. Having generated two datasets, we pass in D_{bias} as context to the transformer, and calculate the loss with respect to the transformer's predictions and the fair outcomes Y_{fair} (Figure 4). It's worth noting that we simply discard X_{fair} in this strategy, but discuss how it could be applied to train FairPFN to be a fairness pre-processsing technique in Section 5.

Fairness Prior-Fitting We train the transformer for approximately 3 days on an `RTX-2080` GPU. Throughout training, we vary several hyperparameters, including the size and connectivity of the MLPs, the number of features sampled, and the number of dataset samples generated. To calculate the loss between the predicted and ground truth values of Y_{fair} classification setting, we apply Binary-Cross-Entropy (BCE) loss and a decaying learning rate schedule.

Causal Case Studies First, we introduce our synthetic benchmark, a hand-crafted set of causal case studies with increasing difficulty, designed to evaluate FairPFN's ability to remove various sources of bias in causally generated data.

Our simplest case study is the `Biased` scenario, where the protected attribute A has an indirect causal effect on the outcome. This case study aims to simulate what happens when FairPFN encounters a scenario where the outcome is only causally influenced by a protected attribute. Next, we implement `Direct` and `Indirect Effect` scenarios to evaluate FairPFN's ability in isolating the direct and indirect effects of bias. Finally, we implement three scenarios, `Level-One`, `Level-Two`, and `Level-Three` with inspiration drawn from the three levels of counterfactual fairness. We provide an overview of our causal case studies with their corresponding data-generating processes in Figure 2.

To provide a diverse synthetic benchmark, we independently generate 100 datasets per case study varying the causal weights of simulated protected attributes w_A, the number of samples $m \in (100, 1000)$ (log-scale), and the standard deviation of Gaussian noise terms $\sigma \in (0, 1)$ (log-scale). We also create counterfactual versions of each dataset which we

[3] We note that this bias removal strategy motivates our sampling of A from an arbitrary distribution $A \in \{a_0, a_1\}$ and not $A \in \{0, 1\}$ because $f = 0 \cdot wx + \epsilon$ would have the same result as $f = p \cdot 0x + \epsilon$

use to evaluate FairPFN for counterfactual fairness, which we measure as the Mean-Absolute Error (MAE) between predictions on the real and counterfactual datasets.

Real-World Datasets We also apply FairPFN to two real-world datasets whose causal graphs are widely agreed upon in the causal fairness community. The first problem we focus on is the Law School Admissions problem, which comes from the 1998 LSAC National Longitudinal Bar Passage Study (Wightman, 1998). The LSAC study recorded law school admissions data from approximately 30,000 applicants to top US law schools and reports a significant disparity of bar passage and first-year average (FYA) outcomes with respect to applicant race.

We use the causal graph visualized in Figure 3 and observational data as input to the `dowhy.gcm` module (Sharma & Kiciman, 2020), which fits a causal model using Random Forest Regressors to estimate non-linear causal relationships. We use these causal models to measure the TE and create counterfactual data. We also apply the `compute_noise` functionality to infer the values of noise terms ϵ_{GPA} and ϵ_{LSAT} to use later as training data for our `Level-Three` baseline (Appendix Section A).

The next problem we focus on is the Adult Census Income problem (Asuncion & Newman, 2007), a dataset drawn from the 1994 US Census that records the demographic information and income outcomes ($INC \geq 50K$) for nearly 50,000 individuals. Again, we fit a causal model in order to measure the TE of protected attribute $RACE$, create a counterfactual dataset (Figure 7), and infer values of noise terms ϵ (Appendix Figure 12).

4. Results

In this section, we evaluate the performance of FairPFN on our benchmark of synthetic and real-world scenarios, with the key message that FairPFN removes the causal and counterfactual effect of protected attributes without any knowledge of the causal model.

Synthetic Data First, we evaluate FairPFN on our synthetic causal case studies, by visualizing the change in causal effect (DE, IE, or TE) before and after bias-mitigation with FairPFN (Figure 5), with a color gradient of blue to green to represent the increasing amount of noise in each dataset. We observe across all case studies that FairPFN learns to remove the causal effect of the protected attribute with a small variance and highlight two interesting effects.

First, we observe on 5 out of 6 case studies that datasets with higher noise levels can generally be solved while maintaining a lower level of error. This could be due to 1) the lower `Unfair` TCE in these datasets or 2) the increased identifiability of SCMs with noise and non-linearity (Peters et al., 2014). Additionally, we find that on the `Biased` case study, FairPFN often achieves an error (1-AUC) less than 0.5. This suggests that FairPFN does not revert to a random classifier when data is only causally influenced by protected attributes as there is still fair information (namely ϵ_X and ϵ_y) in the data. Instead, FairPFN removes only the causal effect $w_A A^2$ in the corresponding structural equation, allowing the noise terms ϵ_X and ϵ_y to influence its predictions.

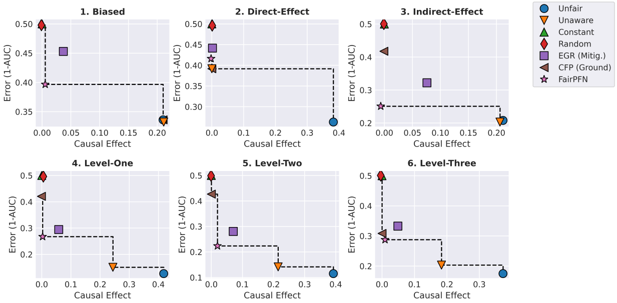

Figure 4. **Causal Effect Removal (Synthetic):** Average causal effect (IE, DE, or TEE) and error (1-AUC) of FairPFN compared to our baselines. FairPFN is on the Pareto Front across all synthetic case studies, dominates `EGR` on 5 out of 6, and always improves upon `CFP` in terms of error.

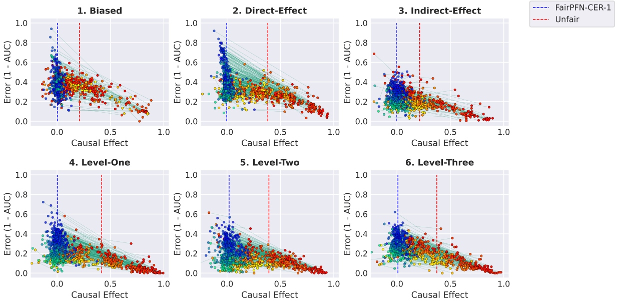

Figure 5. **Effect of Noise Terms (Synthetic)**: Causal Effect (TCE) and erorr (1-AUC) of FairPFN compared to the `Unfair` baseline on each individual dataset from our causal case studies. We provide a color gradient for both baselines (blue to green and red to yellow) to depict increasing amount of noise in the data. FairPFN consistently reduces the TCE on all benchmark groups, achieving lower error on datasets with larger amounts of noise.

We also observe in Figure 4 that FairPFN dominates `EGR` in 5 out of 6 case studies, is on the Pareto Front in all 6, and always improves in terms of predictive performance compared to `CFP`. This is likely attributed to the effect observed in Figure 4 on the `Biased` case study, where FairPFN learns to remove only the causal effect of the protected attribute, still allowing all remaining information to influence its predictions.

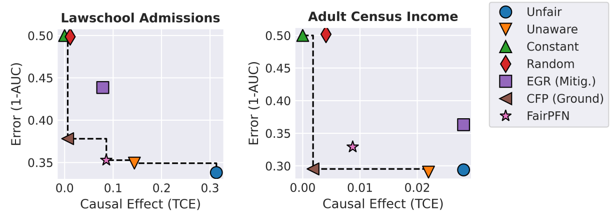

Figure 6. **Causal Effect Removal (Real-World):** Causal effect (TCE) and error (1-AUC) of FairPFN on our real-world datasets compared to other baselines. FairPFN is Pareto Optimal in both cases and provides a strong balance of causal fairness and accuracy.

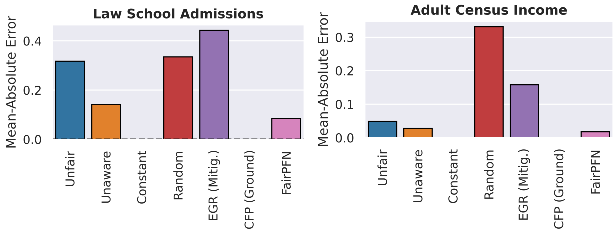

Figure 7. **Counterfactual Fairness (Real-World):** Mean Absolute Error (MAE) between predictive distributions on the original and counterfactual versions of our real-world datasets. FairPFN achieves competitive MAE with `CFP` and `Constant` baselines without having prior knowledge of the causal graph.

Real-World Data We also evaluate FairPFN on the Law School Admissions and Adult Census Income datasets, using causal models fit to the structures posed in Figure to measure the TeE and MAE 3. We note again that in evaluation FairPFN receives no information about the causal graphs or models. In Figure 6, we measure the causal effect across different baselines, observing that FairPFN shows significant improvement in terms of TCE compared to the `Unfair` and `Unaware` baselines. It also demonstrates competative TCE and improved error on the Law School dataset compared to the `CFP` baselines On the Adult dataset, FairPFN is outperformed by the `CFP` baseline, which achieves dominating TCE and error. This outcome is likely explained by the fact that the `Unfair` TCE on the Adult dataset is already quite small (0.03), and thus the four fair noise terms in Figure 3 have a relatively higher representative capacity than in the Law School problem. However, FairPFN still reduces the TCE to less than 0.01, a very acceptable outcome in the broader scope of the problem.

In Figure 7, we also measure the MAE between the predictive distributions on the real and counterfactual datasets, $\hat{Y}_{real}$ and $\hat{Y}_{a\to a'}$. We observe that FairPFN achieves competitive MAE with `CFP` in both scenarios, learning to make counterfactually fair predictions without having access to the causal model or graph. We note that interestingly, `EGR` performs similarly poorly to `Random` in both scenarios, aligning with the intuition that randomization is not a counterfactually fair strategy as individuals do not receive consistent outcomes in either the real or counterfactual worlds.

5. Future Work & Discussion

In this study, we introduce FairPFN, a novel bias-mitigation technique that learns a pre-trained transformer to remove the causal effect of protected attributes in fairness-aware binary classification problems from observational data alone. FairPFN addresses a key limitation in the causal fairness literature by eliminating the need for prior knowledge of the true causal graph in fairness datasets, making it easier for practitioners to apply counterfactual fairness to complex problems where the underlying causal model is unknown. This expands the scope and applicability of causal fairness techniques, enabling their use in a broader range of scenarios. Looking ahead, we believe that FairPFN opens the door to several promising avenues of research.

Real-World Evaluation A crucial next step in FairPFN would be to train a module to predict the effect of interventions on the protected attribute, producing counterfactual datasets to evaluate on. Doing so with FairPFN could hold advantages in robustness as compared to using causal discovery techniques such as (Lorch et al., 2022), since our pre-trained transformer integrates over the possible causal explanations for the data.

Transparency and Interpretability In cases where a causal graph or a subset of causal relationships are known, incorporating this domain knowledge as additional input to the transformer could enhance both FairPFN's human-centricity and performance. Additionally, a future direction could involve predicting the causal graphs that explain the data, adding an extra layer of interpretability.

Fairness Preprocessing By modifying FairPFN's output to predict not only fair outcomes but also fair versions of observational variables, we can improve interpretability and transparency while allowing practitioners to use their preferred ML model during deployment. FairPFN could also be repurposed as a generative model to create fair training data, increasing the performance of the selected model.

Business Necessity Incorporating these business-necessity from (Plecko & Bareinboim, 2022) variables into our fairness prior could enable specifying variables through which to allow the causal effect of the protected attribute. This extension is similar to path-specific counterfactual (Peters et al., 2014), which would also open up many more application areas, such as medical diagnosis, where the social effects of protected attributes like sex should be removed, yet their biological effects must be preserved to provide individualized treatment.

Acknowledgements

The authors of this work would like to thank the reviewers and the the organization of the ICML Next Generation AI Safety Workshop for the opportunity to share our work and recieve feedback. We would like to additionally thank the Zuse School ELIZA Master's Scholarship Program for their financial and professional support of our main author.

References

Agarwal, A., Beygelzimer, A., Dudík, M., Langford, J., and Wallach, H. A reductions approach to fair classification. In *International conference on machine learning*, pp. 60–69. PMLR, 2018.

Angwin, J., Larson, J., Mattu, S., and Kirchner, L. Machine bias. *ProPublica, May*, 23(2016):139–159, 2016.

Asuncion, A. and Newman, D. Uci machine learning repository, 2007.

Barocas, S., Hardt, M., and Narayanan, A. *Fairness and machine learning: Limitations and opportunities*. MIT Press, 2023.

Castelnovo, A., Crupi, R., Greco, G., Regoli, D., Penco, I. G., and Cosentini, A. C. A clarification of the nuances in the fairness metrics landscape. *Scientific Reports*, 12 (1):4209, 2022.

Hollmann, N., Müller, S., Eggensperger, K., and Hutter, F. Tabpfn: A transformer that solves small tabular classification problems in a second. *arXiv preprint arXiv:2207.01848*, 2022.

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

Lorch, L., Sussex, S., Rothfuss, J., Krause, A., and Schölkopf, B. Amortized inference for causal structure learning, 2022. URL https://arxiv.org/abs/2205.12934.

Ma, J., Guo, R., Zhang, A., and Li, J. Learning for counterfactual fairness from observational data. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 1620–1630, 2023.

Müller, S., Hollmann, N., Arango, S. P., Grabocka, J., and Hutter, F. Transformers can do bayesian inference. *arXiv preprint arXiv:2112.10510*, 2021.

Peters, J., Mooij, J., Janzing, D., and Schölkopf, B. Identifiability of causal graphs using functional models. *arXiv preprint arXiv:1202.3757*, 2012.

Peters, J., Mooij, J. M., Janzing, D., and Schölkopf, B. Causal discovery with continuous additive noise models. 2014.

Plecko, D. and Bareinboim, E. Causal fairness analysis. *arXiv preprint arXiv:2207.11385*, 2022.

Sharma, A. and Kiciman, E. Dowhy: An end-to-end library for causal inference. *arXiv preprint arXiv:2011.04216*, 2020.

Wightman, L. F. Lsac national longitudinal bar passage study. lsac research report series. 1998.

A. Baseline Models

To compare FairPFN to a diverse set of traditional, causal-fairness, and fairness-aware ML algorithms, we also implement several baselines which we summarize below:

- `Unfair`: A `TabPFNClassifier` is fit the entire dataset (X, A, y)
- `Unaware`: A `TabPFNClassifier` is fit to non-protected attributes (X, y)
- `Constant`: A ”classifier” that always predicts the majority class
- `Random`: A ”classifier” that randomly predicts the target
- `Level-One`: A `TabPFNClassifier` is fit to non-descendant observables of the protected attribute (X_{fair}, y) if any exist
- `Level-Two`: A `TabPFNClassifier` is fit to non-descendant unobservables of the protected attribute (U_{fair}, y) if any exist
- `Level-Three`: A `TabPFNClassifier` is fit to noise terms of observables (ϵ, y) if any exist
- `EGR`: Exponentiated Gradient Reduction (EGR) for fairness metric DP as proposed by (Agarwal et al., 2018)

We note that these baselines are specifically designed to provide ground truths of the best and worst that can be done in terms of fairness metrics and that certain baselines are only applicable to certain datasets. For example `Unfair`, `Unaware`, `Random`, `Constant`, and `EGR` are applicable on all synthetic and real-world datasets. `Level-One` is only applicable to `Direct Effect`, `Indirect Effect` synthetic causal case studies. `Level-Two` is additionally applicable to the `Level-Two` synthetic case study, and `Level-Three` is additionally applicable to the `Level-Three` synthetic case study as well as the real-world datasets where the causal model is known and noise terms ϵ can be estimated.

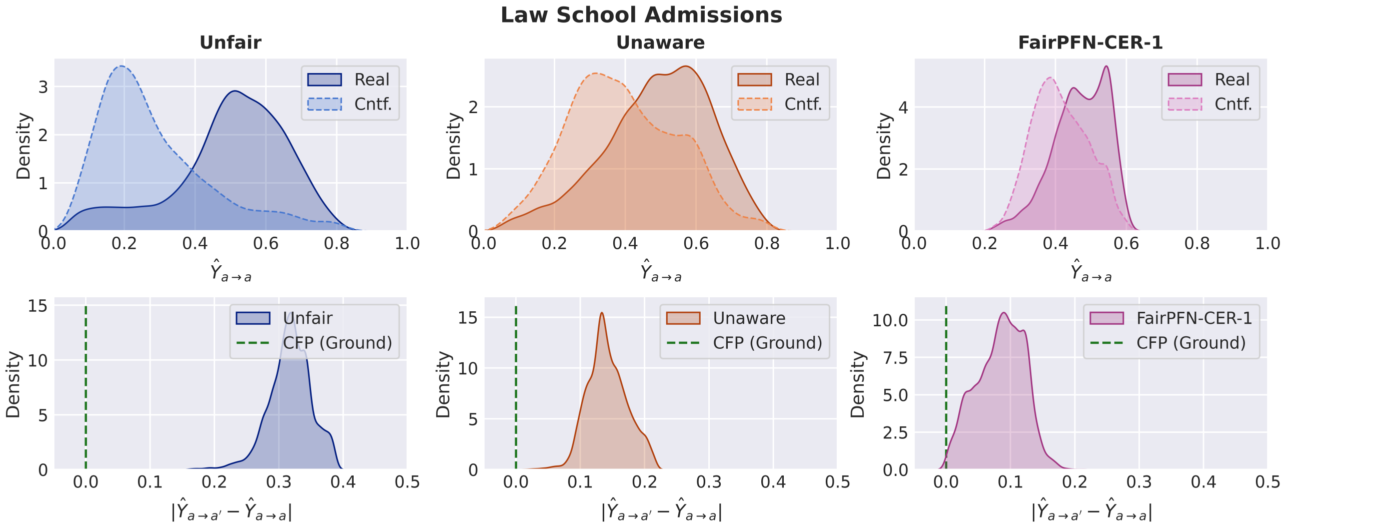

Figure 8. **Aligning Counterfactual Distributions (Law School):** Alignment of real and counterfactual predictive distributions $\hat{Y}$ and $\hat{Y}_{a\rightarrow a'}$ on the Law School Admissions problem. FairPFN best aligns the predictive distributions (top) and achieves the lowest mean (0.1) and maximum (0.2) absolute error.

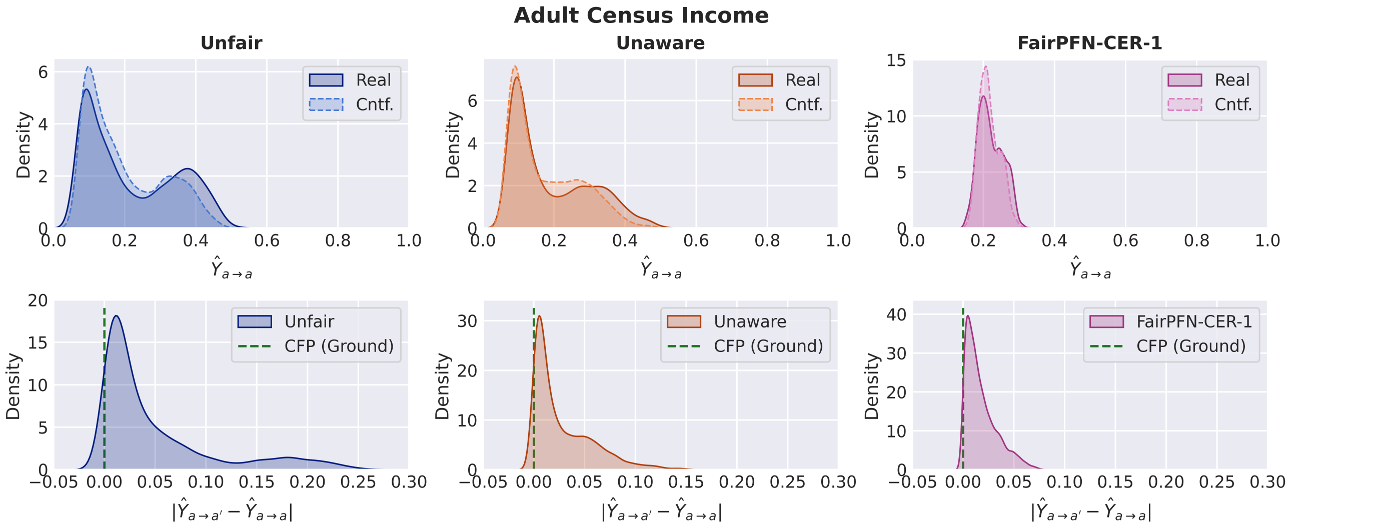

Figure 9. **Aligning Counterfactual Distributions (Adult):** Alignment of real and counterfactual predictive distributions $\hat{Y}$ and $\hat{Y}_{a\rightarrow a'}$ on the Adult Census Income problem. FairPFN best aligns the predictive distributions (top) and achieves the lowest mean (0.01) and maximum (0.75) absolute error.

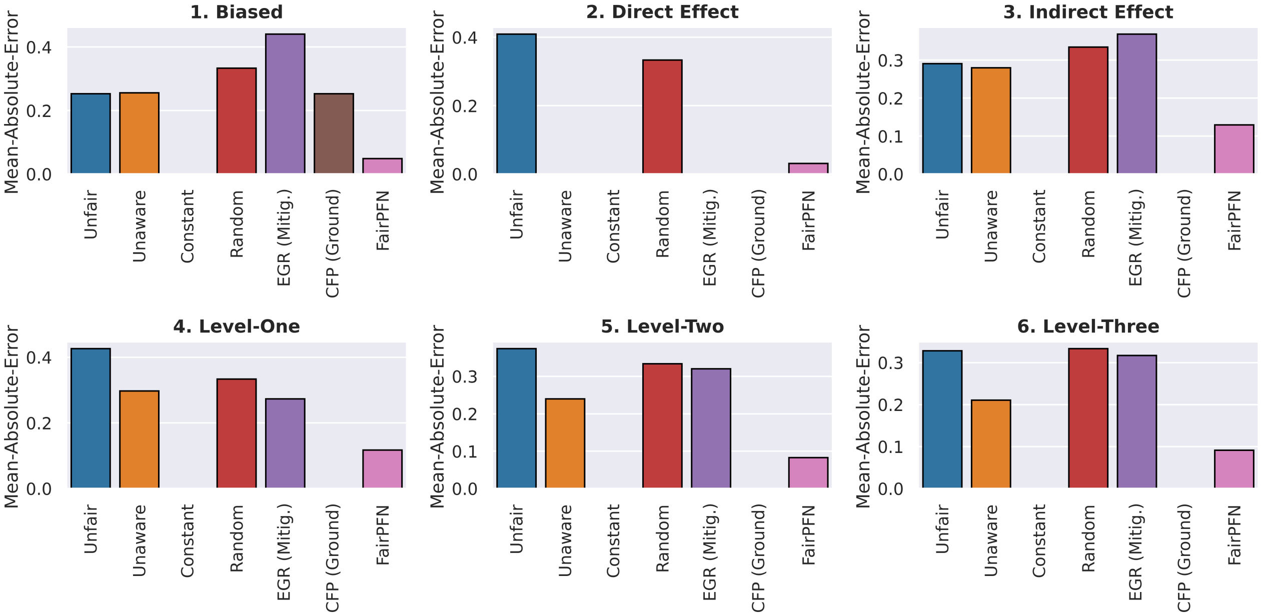

Figure 10. **Counterfactual Fairness (Synthetic):** Mean Absolute Error (MAE) between predictive distributions on the original and counterfactual versions of our causal case studies. FairPFN achieves competitive MAE with `CFP` and `Constant` baselines without having prior knowledge of the causal graph.

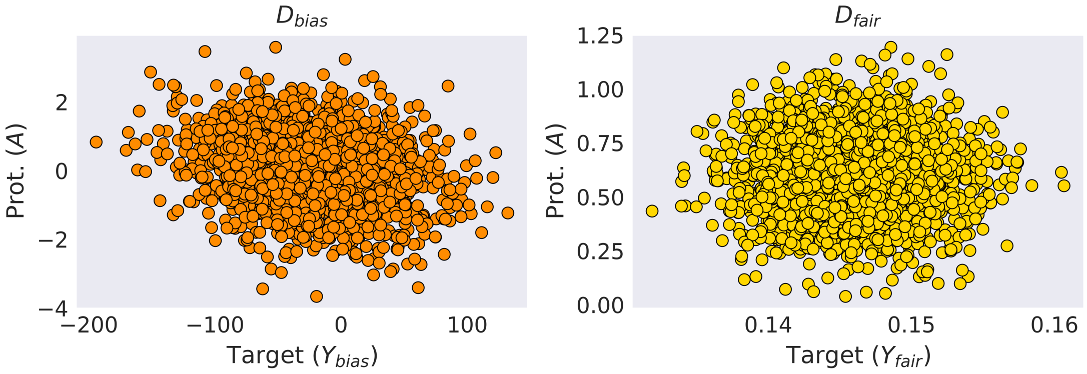

Figure 11. **Effect of Dropout:** Visualization of the effect of dropout on the outgoing edges of a protected attribute in a sampled MLP. In the biased dataset (left), the protected attribute has a slight negative correlation with the target, while in the fair dataset this effect is reduced to Gaussian Noise.

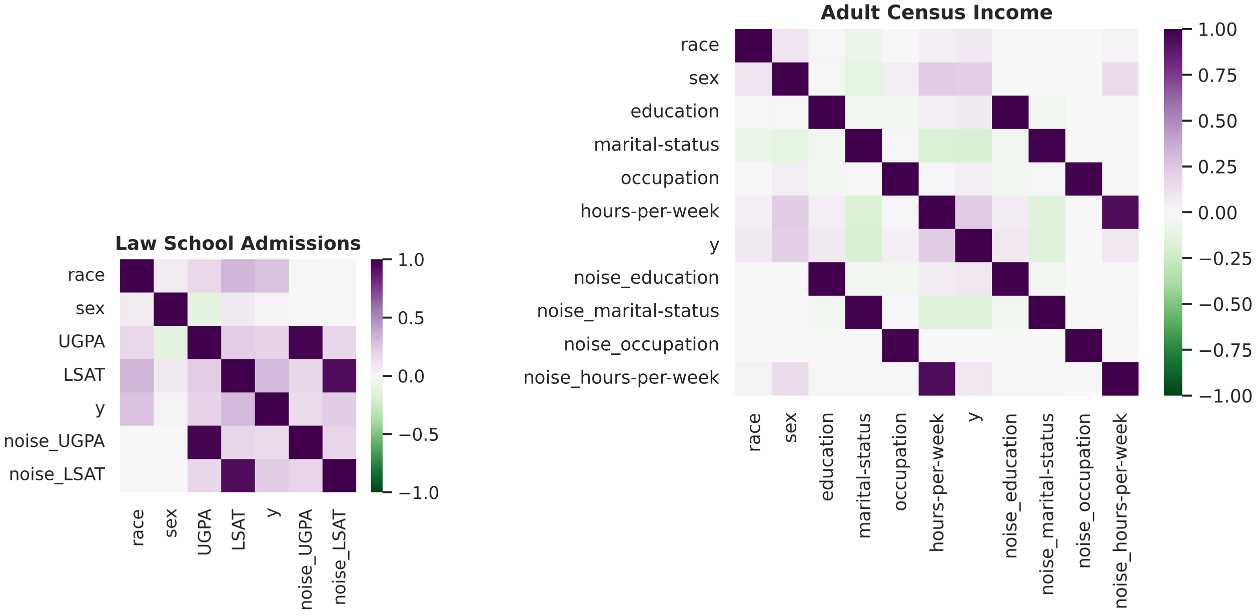

Figure 12. **Derivation of Noise Variables**: Pearson correlation of features including noise terms calculated using inverse probabilistic programming in `dowhy`'s `compute_noise` functionality. Noise terms are uncorrelated with protected attributes and highly correlated with their corresponding observable.