FairPFN: Transformers Can Do Counterfactual Fairness

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

[1]University of Frieburg
[2]ELLIS Institute Tübingen
[3]Zuse School ELIZA
[4]Charité Universitätsmedizin Berlin

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 aligns closely with legal standards, capturing the intuition that a decision is fair to an individual if it remains unchanged in a hypothetical scenario where the individual is part of another demographic group. Despite the theoretical benefits of counterfactual fairness, it comes with several practical limitations, largely related to the over-reliance on domain knowledge and approximate causal discovery techniques in constructing a causal model. In this study, we take a fresh perspective on achieving counterfactual fairness, building upon recent work in in-context-learning (ICL) and prior-fitted networks (PFNs) to learn a transformer called FairPFN. This model is trained using synthetic fairness data to eliminate the causal effects of protected attributes directly from observational data. In our experiments, we thoroughly assess the effectiveness of FairPFN in eliminating the causal impact of protected attributes. 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 [2]. 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*, (i.e. attributes which should not influence decision-making such as ethnicity or sex) take on a different value. According to a recent review contrasting observational and causal fairness metrics [5], causal fairness holds advantages in human-centricity and its analogy to the legal notions of direct and indirect discrimination. However, the non-identifiability of causal models [11] presents a significant challenge in proposing a causal model, as the causal effects of protected attributes 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 have adverse impacts [9].

In this study, we introduce a novel approach to counterfactual fairness based on the recently proposed TabPFN [7]. Our approach, coined FairPFN, learns to identify and remove the causal effect of protected attributes by training on a synthetic benchmark of many causally generated data sets. FairPFN makes predictions that satisfy counterfactual fairness by removing the direct and indirect effects of discrimination. In our experimental results, we demonstrate the effectiveness, flexibility, and extensibility of PFNs for algorithmic fairness.

AutoML 2024 Workshop Track

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 [4]. Fairness research aims to define and measure algorithmic bias as well as develop principled methods that mitigate it.

Causal Fairness Causal ML is a new and emerging research field that aims to represent observational data and prediction problems in the language of causality, providing formalizations for causal modeling, mediation analysis, and counterfactual explanations [6]. The Causal Fairness Analysis (CFA) framework [13] 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 causal effects of protected attributes on outcomes both in fairness problems and datasets as well as after applying bias-mitigation techniques.

Counterfactual Fairness. A related causal concept of fairness is counterfactual fairness [8], 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) (Appendix Figure 4). Counterfactual fairness has gained significant popularity in the fairness community, inspiring recent work on path-specific extensions [12] and CLAIRE, which applies Variational Autoencoders (VAEs) to achieve counterfactually fair latent representations [9]. CLAIRE is not included as a baseline as their training code or model is not publicly available. 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. [5] argue that it is challenging to obtain causal graphs representing complex systemic inequalities. Additionally, Ma et al. [9] demonstrate that proposing an incorrect causal graph or model can invalidate counterfactual fairness and potentially lead to adverse impacts 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 [10]. This allows PFNs to perform well on downstream tasks with limited data by leveraging the learned prior knowledge. TabPFN [7], a recent application of PFNs to small, tabular classification problems, trains a transformer on a hypothesis of synthetic datasets generated from sparse Structural Causal Models (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 integrates concepts from prior-fitted networks (PFNs) with principles of causal and counterfactual fairness. FairPFN aims to eliminate the causal effects of protected attributes from predictions on 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.[1] The first step

[1] 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 [13].

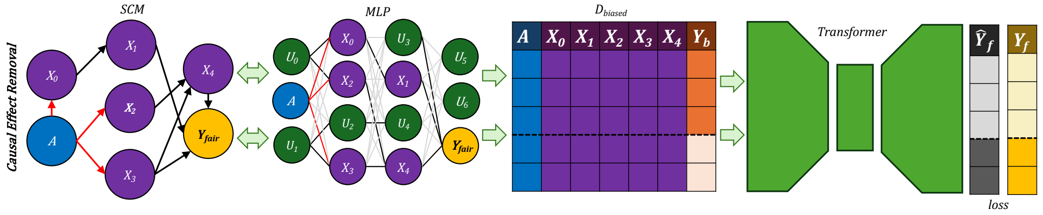

Figure 1: **FairPFN Pre-training**: FairPFN is pre-trained on a synthetic prior of datasets generated from sparse SCMs with exogenous protected attributes. A biased dataset is generated and passed as context to the transformer, and the loss is calculated with respect to the fair outcomes calculated by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

of FairPFN is the generation of *biased* synthetic datasets that realistically represents the causal mechanisms of bias in real-world datasets. We provide a visual overview of this process in Figure 1. Taking inspiration from [7], 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 sparsity of the represented SCM. The exogenous protected attribute is sampled from the inputs to the MLP as a binary variable $A \in \{a_0, a_1\}$ where a_i is sampled from the same distribution of non-protected exogenous variables U_{fair} to prevent numeric overflow. We uniformly sample m features X from the second hidden layer onwards to ensure that they contain rich representations of the causes in the input layer. Finally, we select the target Y from the output layer. Because Y is output as a continuous variable, we binarize it over a random defined threshold.[2]

Causal Effect Removal. The strategy by which we 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 $f = 0 \cdot wx + \epsilon$. This means that the effect of the protected attribute is transformed to Gaussian noise ϵ as visualized in Appendix Figure 7.[3]. Having generated two datasets, we pass in D_{bias} as context to the transformer and calculate the loss between the transformer's predictions and the fair outcomes Y_{fair}. This strategy relates to aligning real and counterfactual predictions as required by counterfactual fairness [8], because if the protected attribute has no causal effect on outcomes, intervening on it will not produce different counterfactual outcomes.

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.[4] 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 scenarios with increasing difficulty (Figure 2 top). We generate 100 independent datasets per benchmark class varying the causal weights of simulated protected attributes w_A, the number of samples $m \in (100, 10,000)$ (log-scale), and the standard deviation of noise terms $\sigma \in (0, 1)$ (log-scale).

[2] Without binarizing, future versions of FairPFN are extensible to fairness regression tasks and to handle multiple protected attributes.

[3] 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 in the latter case $f = 0 \cdot wx + \epsilon$ would have the same result as $f = p \cdot 0x + \epsilon$, potentially causing the transformer to learn to treat all samples as if they belonged to the underprivileged group.

[4] In practice, the utility of FairPFN is restricted to datasets which fit the assumptions in our prior that we make about the causal effect of protected attributes. The likelihood that FairPFN be applicable to any fairness dataset can be increased by either 1) increasing the flexibility of the prior or 2) pre-training on more datasets.

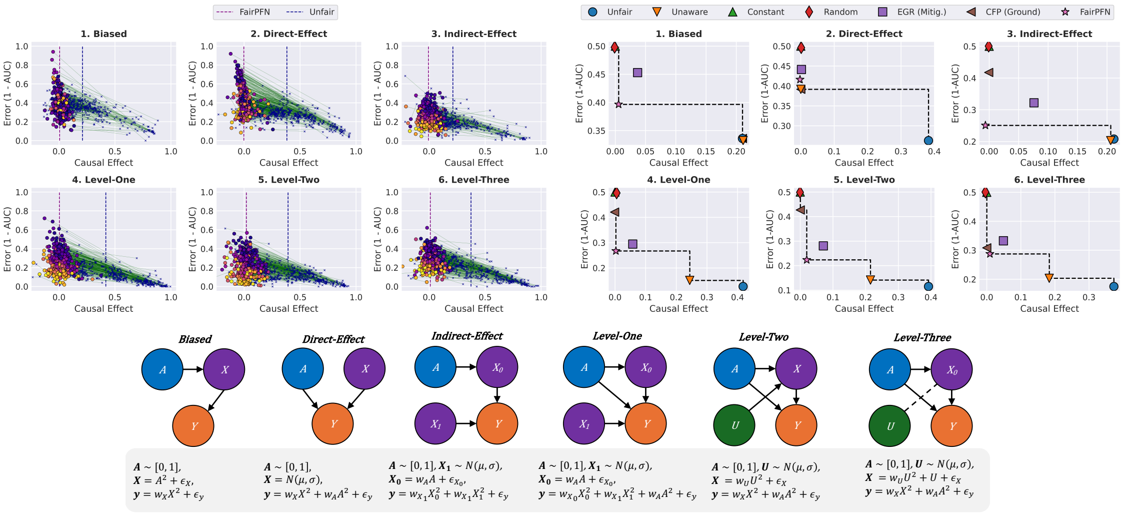

Figure 2: **Synthetic Data**: Causal effect (IE, DE, or TCE) and error (1-AUC) of FairPFN vs. the `Unfair` baseline (top left) and compared to all baselines (top right) on our synthetic causal case studies (top). FairPFN reliably removes the causal effect of protected attributes (left), achieving Pareto optimality in 5/6 case studies.

Real-World Datasets. We also apply FairPFN to two real-world datasets with widely agreed-upon causal graphs. The first problem we focus on is the Law School Admissions problem [15], which records law school admissions data from approximately 30,000 applicants to top US law schools, reporting 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 [14], which fits a causal model to measure the TCE and create counterfactual data. We also apply inverse probabilistic programming using 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 [3], which 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 and infer values of noise terms ϵ (Appendix Figure 8).

4 Results

Synthetic Data. First, we evaluate FairPFN on our synthetic causal case studies by visualizing the change in causal effect (NDE, NIE, or TCE) before and after bias-mitigation with FairPFN (Figure 2 bottom left). We observe across all case studies that FairPFN learns to remove the causal effect of the protected attribute with two interesting effects. First, we observe that datasets with higher noise levels (green and yellow) can generally be solved while maintaining a higher level of accuracy. This could be due to 1) the lower initial causal effect in these datasets or 2) the increased identifiability of SCMs with noise and non-linearity [12]. In either case, this outcome shows that FairPFN is able to identify and remove the causal effect of protected attributes even in noisy datasets, an encouraging outcome for the use of FairPFN in real-world scenarios where measurement error is often present. Additionally, we find that the `Biased` case study often achieves an AUC greater than 0.5, which provides an interesting insight into the inner workings of FairPFN. By maintaining a degree of accuracy in its predictions, FairPFN effectively removes only the causal effect $w_A A^2$ of the protected attribute in the corresponding structural equation, while allowing the noise terms ϵ_X and ϵ_y (or the fair parts of these variables) to influence its predictions. This result shows that FairPFN performs Level-Three counterfactual fairness (Section 2), automatically inferring the values of fair noise terms, while having no explicit knowledge of the causal model.

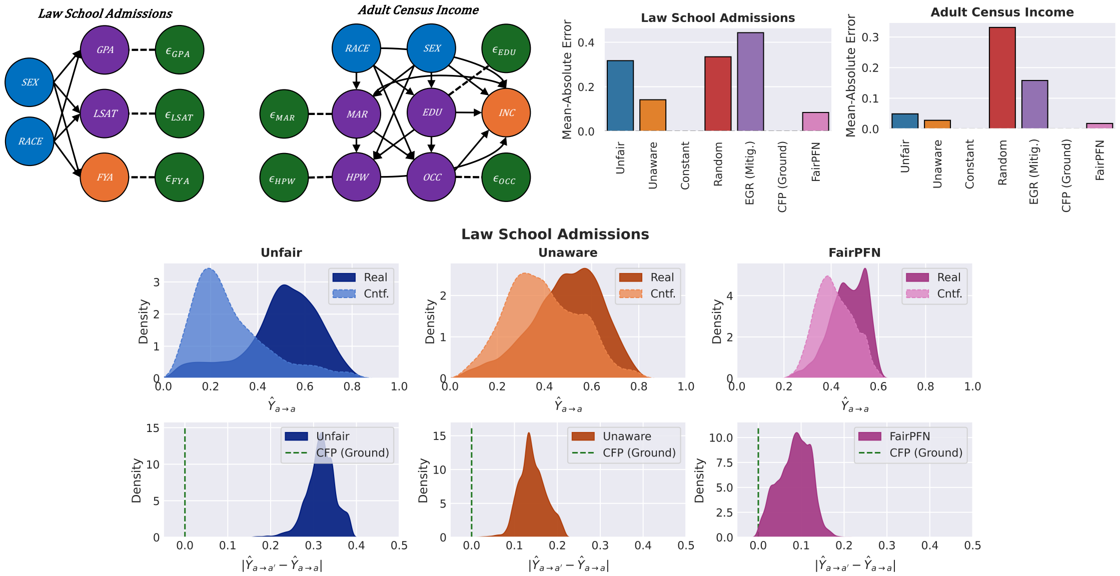

Figure 3: **Real-World Data**: Alignment of real and counterfactual predictive distributions $\hat{Y}$ and $\hat{Y}_{a\rightarrow a'}$ on the Law School Admissions problem (bottom left) and Mean-Absolute Error (MAE) between predictions on original and counterfactual real-world datasets (right). FairPFN aligns real and counterfactual predictions (bottom), achieving competitive MAE with the `CFP` ground truth (top right).

Real-World Data. In Figure 3 (bottom-left), we visualize the alignment of real and counterfactual predictive distributions $\hat{Y}$ and $\hat{Y}_{a\rightarrow a'}$ obtained by intervening on the protected attribute *RACE*. We observe that FairPFN better aligns predictive distributions compared to the `Unfair` and `Unaware` baselines, reducing the maximum difference between real and counterfactual predictions to less than 0.2. We observe the same effect in Figure 3 (right), where we measure elementwise Mean-Absolute Error (MAE) between real and counterfactual predictive distributions. Again, we observe that FairPFN minimizes the difference between these distributions, achieving competitive MAE with the ground truth `CFP` baseline which has full knowledge of the assumed causal model.

5 Conclusion

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, achieving counterfactual fairness with no knowledge of the underlying causal model. Through experiments on a benchmark of synthetic and real-world datasets, we demonstrate the inner workings and potential of FairPFN.

Future Work. Looking ahead, we believe that FairPFN opens the door to several promising avenues of research, including PFNs for fairness pre-processing and fair data generation, path-specific variations of causal and counterfactual fairness, and multi-objective optimization. While we believe that proposing a causal model is cumbersome, we also acknowledge its benefits in terms of human-centricity and interpretability. As such, future versions of FairPFN could incorporate domain knowledge regarding known causal relationships or output a (set of) predictions regarding possible causal explanations for the data.

Limitations. A key limitation of FairPFN is that although it removes the requirement of knowing the causal model, real-world evaluation of causal effects and counterfactual alignment still require this assumption. Therefore, a crucial follow-up would include predicting the effect of interventions on protected attributes, allowing FairPFN to be evaluated and reliably deployed in real-world fairness applications.

Acknowledgements. The authors of this work would like to thank the reviewers and the the organization of the AutoML Conference Workshop track 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

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

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

[3] Asuncion, A. and Newman, D. (2007). Uci machine learning repository.

[4] Barocas, S., Hardt, M., and Narayanan, A. (2023). *Fairness and machine learning: Limitations and opportunities.* MIT Press.

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

[6] Cho, K. (2024). A brief introduction to causal inference in machine learning.

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

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

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

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

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

[12] Peters, J., Mooij, J. M., Janzing, D., and Schölkopf, B. (2014). Causal discovery with continuous additive noise models.

[13] Plecko, D. and Bareinboim, E. (2022). Causal fairness analysis. *arXiv preprint arXiv:2207.11385.*

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

[15] Wightman, L. F. (1998). Lsac national longitudinal bar passage study. lsac research report series.

Broader Impact Statement

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.

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 [1]

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. To synthesize the naming of the `Level-X` baseleines we refer to this strategy as `CFP`.

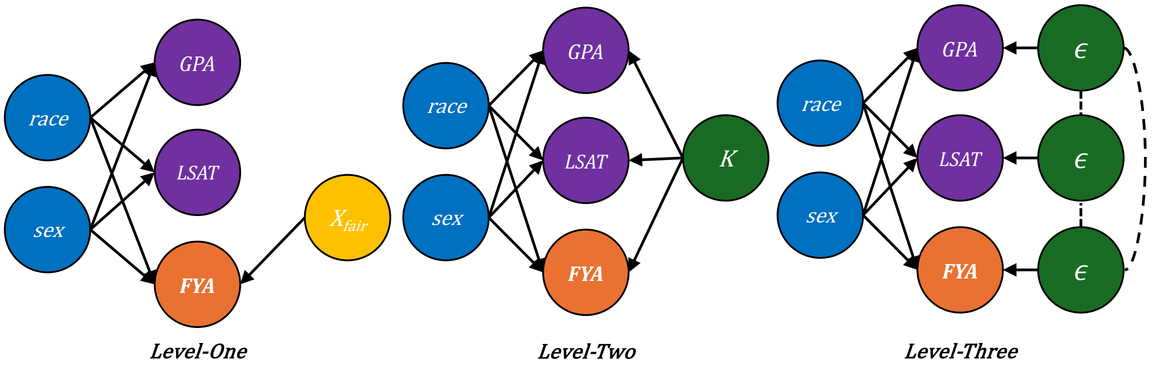

Figure 4: **Counterfactually Fair Prediction (CPF)**: Visualization of the three levels of Counterfactually Fair Prediction (CFP), obtained either by fitting to observable non-descendants (Level-One), the inferred values of an exogenous unobserved variable (Level-Two) or the noise terms of an Additive Noise Model for observable variables (Level-Three).

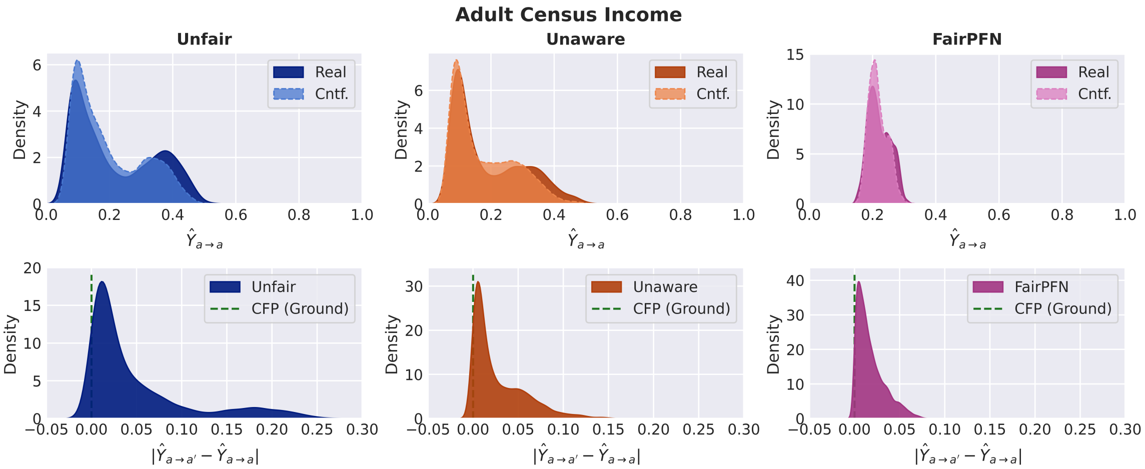

Figure 5: **Aligning Counterfactual Distributions (Adult)**: Alignment of real and counterfactual predictive distributions $\hat{Y}$ and $\hat{Y}_{a \to a'}$ on the Adult Census Income problem. FairPFN best aligns the predictive distributions (top) and achieves the lowest mean (0.01) and maximum (0.75) absolute error.

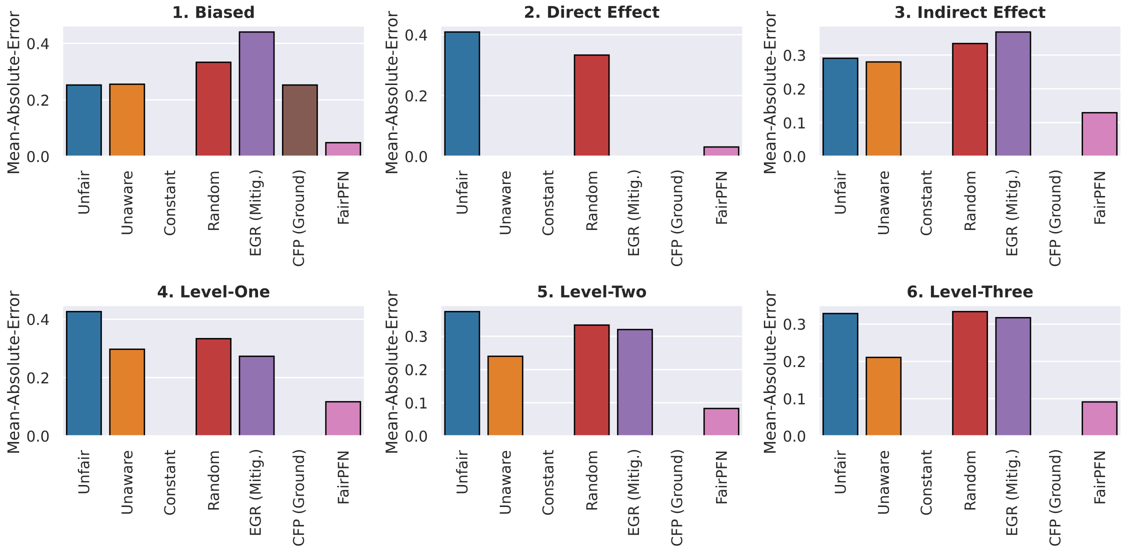

Figure 6: **Counterfactual Fairness (Synthetic)**: Mean Absolute Error (MAE) between predictive distributions on the original and counterfactual versions of our causal case studies. FairPFN achieves competitive MAE with `CFP` and `Constant` baselines without having prior knowledge of the causal graph.

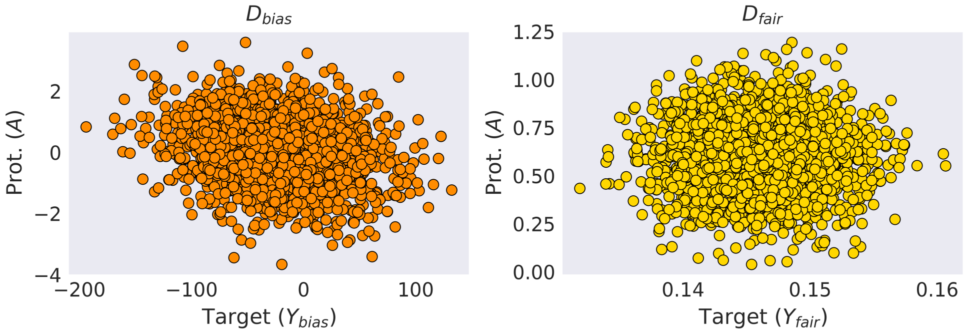

Figure 7: **Effect of Dropout**: Visualization of the effect of dropout on the outgoing edges of a protected attribute in a sampled MLP. In the biased dataset (left), the protected attribute has a slight negative correlation with the target, while in the fair dataset this effect is reduced to Gaussian Noise.

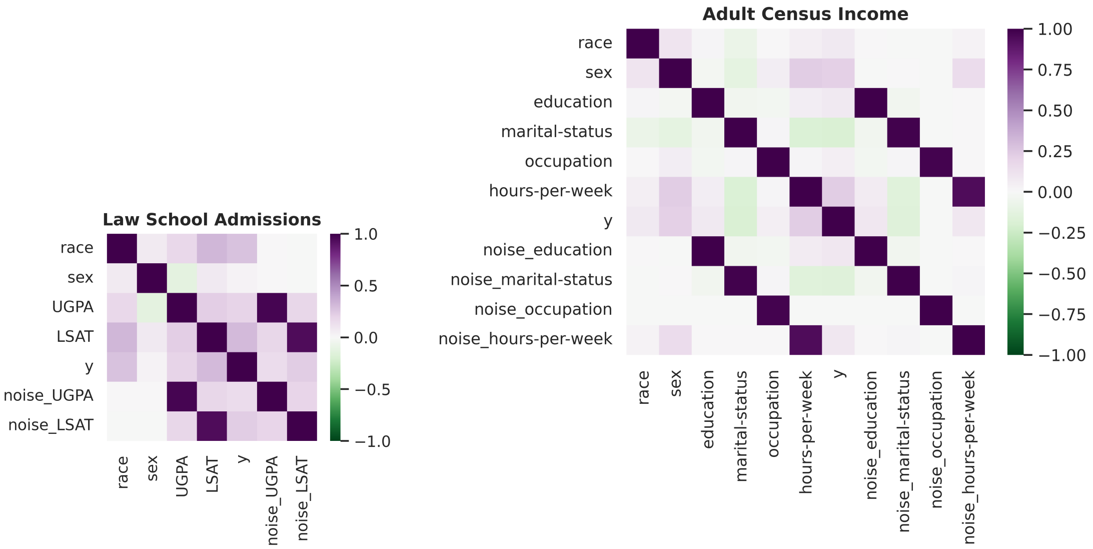

Figure 8: **Derivation of Noise Variables**: Pearson correlation of features including noise terms calculated using inverse probabilistic programming in dowhy's `compute_noise` functionality. Noise terms are uncorrelated with protected attributes and highly correlated with their corresponding observable.