Applications of Fairness: Recovery of Ground-truth from Biased Labels

Lina Battikha
lbattikha@ucsd.edu

Sai Poornasree Balamurugan
s2balamurugan@ucsd.edu

Abstract

Our work this quarter focused on the discovery of latent ground truth labels from biased labels. This fairness algorithm and its accuracy will be evaluated on a variety of datasets, including common fairness datasets from UCI Adults, German Credit Scores, and Compas (recidivism data). The objective of this algorithm is to reduce unfair biases that are affected by a sensitive feature within the data. These biases can significantly impact predictive models, leading to unfair outcomes that disproportionately affect certain groups. Since these models often biased as they use sensitive attributes as a determining factor for outcomes, ensuring fairness in predictions remains a critical challenge.

Website: https://s2balamurugan.github.io/agency-jekyll-theme/
Code: https://github.com/linabat/fairness-application

1 Introduction

Bias in artificial intelligence is an extremely important problem to be explored and addressed as the results of such models can greatly impact an individual's life, especially those who are of minority groups. One common reason that bias arises is due to sensitive features that are included in the dataset, like gender or race. The inclusion of such sensitive features can cause the observed outcome label to pick up on unfair patterns and dependencies that will lead such labels to be biased. To address such problems, our project focuses on the discovery of fair latent variables from the biased labels. Latent variable discovery involves the identification and characterization of hidden variables that influence the distribution and outcomes of the observed data. These variables are not explicitly provided in the dataset but can be inferred by analyzing patterns and relationships among the observed features. Our work has focused on building a model for latent variable discovery for fairness applicability. The latent variable that we are looking to discover is the true unbiased outcome label that is not influenced by the identified sensitive feature within the dataset.

Latent variable discovery is very applicable in the domain of fairness as addressing bias in decision-making systems requires understanding of how latent variables interact with observable features. By accounting for unobserved confounders, these models can enhance fairness, ensuring equitable outcomes for all subpopulations. This is the domain our project falls under. Specifically, we use tabular datasets where we identify non-confounding features of the data, the sensitive feature, and the observed biased label.

With our model, our objective is to receive fair results for all groups that are a part of sensitive features. It is important to note that with the implementation and exploration of efficacy of our model, we assume that the sensitive feature and the feature with the observed labels are completely independent of each other.

1.1 Literature Review

Our work is inspired by the methodology outlined in Scalable Out-of-distribution Robustness in the Presence of Unobserved Confounders by Parjanya Prashant, Seyedeh Baharan Khatami, Bruno Ribeiro, and Babak Salimi Prashant et al. (2025) as well as Group Fairness by Probabilistic Modeling with Latent Fair Decisions by YooJung Choi, Meihua Dang, and Guy Van den Broeck Choi, Dang and Van den Broeck (2021).

The Scalable Out-of-distribution Robustness in the Presence of Unobserved Confounders paper addresses the challenge of training machine learning models when there are unobserved variables, known as latent confounders, which can affect the relationship between the covariates (input features) X and the target variable Y. These unobserved confounding variable Z can be a cause for distribution shift of $P(Y|X)$. To handle this, the paper proposes using a proxy variable S, which is a known variable that helps approximate the latent confounder Z. The proxy variable is crucial in managing sub-population shifts, where the distribution of subgroups in the data changes between training and testing. By using the proxy variable S, the model can infer the latent confounder Z, which is unobserved during

both training and testing. In this setup, the model has access to all covariate input data X and the target variable Y during training, but the latent confounder Z is not directly observed. One of the stages proposed in the methods of this paper is the Encoder-Decoder model. In the first stage, the Encoder-Decoder model is used to estimate the latent confounder distribution by factoring the joint distribution of the input features X and the proxy variable S. The encoder portion of the model learns how the input features X are related to the latent confounder Z, while the decoder models the relationship between S and Z. This factorization allows the model to infer the distribution of the latent confounder, $P(Z|X)$, given the input data Prashant et al. (2025).

The Group Fairness by Probabilistic Modeling with Latent Fair Decisions paper also focuses on latent variable discovery, specifically addressing group fairness. Given a sensitive feature and an observed variable, the authors suggest a methodology where demographic parity can be established with a latent fair label. Demographic parity is where each group within the sensitive feature is experiencing similar label classification distribution; $P(D_f|S=0) = P(D_f|S=1)$.The authors introduce features X, sensitive feature S, observed biased label D and a latent fair variable, D_f , representing the true, unbiased labels. Knowing such variables, it is possible to achieve demographic parity by enforcing the independence between latent variable D_f and S. This is so because of the following reason; D is generated from D_f. In other words, D's true underlying distribution is that of D_f. However, the probabilities of D are affected by the sensitive feature S, causing label bias. To recover the true underlying distribution, D must become independent of S. In the process of doing so, D_f is recovered. With that, the fair label should not have dependencies on the sensitive feature, which allows for demographic parity. Choi, Dang and Van den Broeck (2021)

1.2 Datasets

Our datasets consist of two different scenarios. The first scenario is where the observed biased label is binary, while the second scenario is where the biased label is multi-class. In both cases, the sensitive feature is binary.

1.3 Binary Datasets Implemented

- **Binary Synthetic Dataset**: This dataset was synthetically created to evaluate model performance under ideal conditions, with 5000 rows and 30 features. It consists of a sensitive binary feature S sampled from a binomial distribution, a feature matrix X influenced by S and multinomial noise, and a target variable Y derived from a linear function of X with added noise. With this we have binary fair labels Y. To get unfair labels, we inject Y with bias by systematically modifying labels assigned based on S. Y labels will be modified so that Y = 1 is more likely for samples where S=1, while Y=0 are for samples where S=0. We modify 40% of the Y labels to introduce bias.
- **UCI Adults Dataset**: This dataset a popular tool for determining if a person makes more than $50,000 annually and is obtained from the U.S. Census Bureau. It in-

cludes information about age, education, occupation, work class, marital status, and weekly hours worked, among other demographic and employment variables. This is a larger dataset with 32,561 rows and 5 non-sensitive features. The dataset, which represents actual income distribution inequalities, is often utilized in fairness and prejudice research. Due to its structured nature, it serves as a benchmark for classification models in machine learning. In this dataset, the sensitive feature that we will be using is gender, 1 if male and 0 if female. The observed unfair label is income; 1 if >50K and 0 if <50K UCI Machine Learning Repository (1996).

- **UCI German Credit Dataset**: The UCI Machine Learning Repository provides the German Credit dataset, which is used to evaluate credit risk by categorizing people as "good" or "bad" credit risks. It contains information on an applicant's home situation, work status, financial history, and personal characteristics such as age and sex. This dataset is small with 1000 rows and 3 non-confounding features. The data set is frequently used to examine bias and fairness in financial decision making because of its importance in lending choices, which helps to assess moral issues with credit scoring models. The sensitive feature is age; 1 if age > 33 , 0 if age < 33. Note: 33 is the median age in this dataset and is used as the threshold UCI Machine Learning Repository (1994).
- **Compas Dataset**: Criminal justice data used to determine the probability of recidivism (reoffending) based on demographic and criminal history criteria may be found in the COMPAS dataset, which is gathered by ProPublica. Features including age, sex, race, past crimes, and COMPAS risk scores are among them. This dataset has 7214 rows and 5 non-confounding features. Studies have revealed that the COMPAS risk assessment tool includes racial discrepancies, making it a crucial resource for assessing ethical difficulties in judicial decision-making and predictive policing. As a result, this dataset has been at the center of arguments on algorithmic fairness and bias. For this dataset, the sensitive feature is race - 1 if African-American, 0 if not African-American. The observed label is prediction of recidivism - 1 if they will reoffend within a 2 year frame and 0 if they won't ProPublica (2016).

1.4 Multi-Class Datasets Implemented

- **Multi-Class Synthetic Dataset**: This dataset was synthetically created to evaluate model performance under ideal conditions, with 5000 rows and 30 features. It consists of a sensitive binary feature S sampled from a binomial distribution, a feature matrix X influenced by S and multinomial noise, and a target variable Y derived from a linear function of X with added noise. We split these fair labels Y into 4 groups so that they're multiclass. To get unfair labels, we inject Y with bias by systematically modifying labels assigned based on S. Y labels will be modified so that samples with S=1 have higher Y values, while samples with S = 0 have lower Y values. We modify 40% of the Y labels to introduce bias.
- **UCI Drug (Cannabis) Consumption Dataset**: This dataset is taken from UCI Machine Learning Repository. This dataset contains 1884 rows and 10 non-confounding features. It includes information about how frequently an individual uses cannabis,

along with personal information like gender, ethnicity, and country. The use of this dataset is to predict how likely an individual is to consume cannabis. We pre-process the dataset so there are 4 groups of likeliness: (1) never used, (2) not used in the last year, (3) used in the last year, (4) used in the last week. For this dataset, the sensitive feature is education level - 1 if received higher education, 0 if didn't receive higher education. The observed label is likeliness of cannabis use Khadija (2022).

2 Methods

2.1 Preprocessing

To ensure consistency and comparability across the real-world datasets we used in our evaluation, we applied a preprocessing pipeline to transform that data into a standardized numerical format. For the set of non-sensitive features X, we converted categorical variables into numerical values where necessary then standardized all features in X.

For the sensitive feature S in each dataset, we binarized the values based on two distinct groups identified within the feature. Similarly, when the observed label Y was multi-class, we identified the relevant categories and applied the label encoding to ensure a numerical representation.

Through this preprocessing procedure, all datasets were transformed such that X, S, and Y were fully numerical, facilitating fair and consistent analysis across different models an evaluation metrics.

2.2 Model

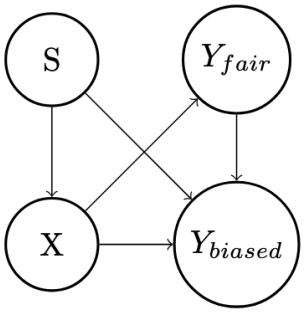


Figure 1: Causal graph showing underlying independencies established

Our approach uses a similar encoder-decoder methodology as used in the Scalable Out-of-distribution Robustness in the Presence of Unobserved Confounders Prashant et al. (2025) while establishing independence between the sensitive feature and fair observed label as mentioned in the Group Fairness by Probabilistic Modeling with Latent Fair Decisions Choi, Dang and Van den Broeck (2021) paper. Given a dataset that has an identifiable sensitive

feature S, an observed biased label Y, and non-confounding features X, we treat Y as the proxy variable that will be used in the model to recover the true fair label Y_f. Figure 1 shows the underlying causal graph that makes up our model. Since we know that Y_{biased} is of the same distribution of Y_{fair}, however, with its probabilities affected by S; we are able to recover Y_{fair} by establishing independence between S and Y_{biased}.

We will use an encoder-decoder model with adversarial biasing used as a regularizer, so that the independence condition between S and Y_f is established. Adversarial biasing is a machine learning concept used to reduce bias in predictive models by implementing an adversarial network. The main idea is to train two competing networks: a predictor that learns to classify data while the adversary tries to predict the sensitive feature from the Y label. The predictor is trained to minimize classification loss while simultaneously manipulating the model so that the sensitive feature cannot be predicted from the Y label. This makes sure that predictions are fair, preventing models from making biased decisions based on the sensitive attribute.

To do this, the model can be broken down into 3 main sections: Encoder, Adversarial, and Decoder.

1. **Encoder:** Forward pass of retrieving $pseudo\_Y$, given X. $pseudo\_Y$ will be the intermediate label that will be used to establish independence with S. It works to avoid direct dependence on S.
2. **Adversarial Branch:** The adversarial branch attempts to exploit any dependence between $pseudo\_Y$ and S, revealing whether $pseudo\_Y$ still contains information about S. The adversary uses $pseudo\_Y$ to predict S. If the adversary can predict S from $pseudo\_Y$, then this is indication that independence has not yet been established yet.

   To prevent the adversary from successfully predicting S, the encoder learns to remove S related information from $psuedo\_Y$. This is done through the Gradient Reversal Layer (GRL), which act as a regularizer to flip the gradient so that the encoder does not encode information about S in pseudo Y.

   The adversary will work to minimize the loss between S_pred (the predicted S) and S (true S) to extract as much information as possible from $psuedo\_Y$. However, when GRL flips the gradient of the loss, the encoder will be forced to maximize this loss. As a result, the encoder removes dependencies between $psuedo\_Y$ and S, leading $psuedo\_Y$ to be independent of S.

   This adversarial training process happens during the back-propagation between the encoder and adversary. GRL acts as a regularizer by flipping the gradient, which forces the encoder to make $psuedo\_Y$ less informative about S, thereby reducing the adversary's ability to predict S. The encoder would be working against the adversary, and as a result, the adversary will eventually fail if $psuedo\_Y$ is independent of S, which is the desired outcome.

   The degree of independence between Y and S is controlled by the lambda parameter,

which determines how strongly to penalize the dependence. A larger lambda value enforces greater independence between Y and S. However, this comes at the cost of potentially reducing overall model accuracy.

3. **Decoder:** The decoder will try to predict Y given S and $pseudo\_Y$. While we are trying to establish independence between Y and S by removing unfair dependencies of S on Y, some fair dependencies can still exist. For this reason, $psuedo\_Y$ and S are being used to predict Y so that the output includes accurate and fair results.

The loss functions that are being used are binary cross-entropy and categorical cross-entropy. Binary cross-entropy measures the difference between two probability distributions: the predicted probability distribution and the true distribution. It quantifies how well the predicted probabilities align with the actual class labels by computing the negative log-likelihood of the true class. It also penalizes incorrect predictions more harshly when the predicted probability is far from the true label. In binary classification, cross-entropy is used with the sigmoid activation function, while in multiclass classification, it is often combined with softmax. Lower cross-entropy values indicate better model predictions.

A loss function used in multi-class classification problems, where each sample is a member of multiple classes, is categorical cross-entropy. By calculating the negative log-likelihood of the proper class, it assesses how closely the projected probability distribution matches the actual class labels. The loss for a single sample is computed given a predicted probability distribution y-hat and a one-hot encoded true label y. Since the model gives the right class a higher probability, lower numbers correspond to better predictions.

Using the pseudo-label Y, the adversarial branch aims to predict the sensitive characteristic S. Because the gradient reversal layer makes sure that this loss function optimizes the error in predicting S, bias is reduced because Y must be independent of S.

There are two scenarios which we applied this model to. The first scenario is where both the sensitive feature and the observed label are binary, meaning the values in each column are either 1 or 0. The second scenario is where the sensitive feature is binary (1 or 0), while the observed label is multi-class (not limited to 1 or 0).

2.3 Evaluation

Once we had results for the fair latent variable retrieved from the encoder-decoder model, we wanted to evaluate our results. To do so, we used a logistic regression model to evaluate performance. We had a baseline logistic regression model that was trained on X and the observed Y label. We also had a logistic regression trained on preprocessed X and the fair Y labels ($psuedo\_Y$) that were retrieved from the encoder-decoder model. Using these classifiers, we were able to evaluate different metrics. The metrics we used for evaluation of our results were demographic parity, as used in Group Fairness by Probabilistic Modeling with Latent Fair Decisions Choi, Dang and Van den Broeck (2021), Area Under the Curve, Accuracy, and Equalized Odds Difference.

- **Demographic Parity Difference**: For each sub group within the sensitive feature S, we calculate the proportion of each class label in $pseudo\_Y$. The demographic par-

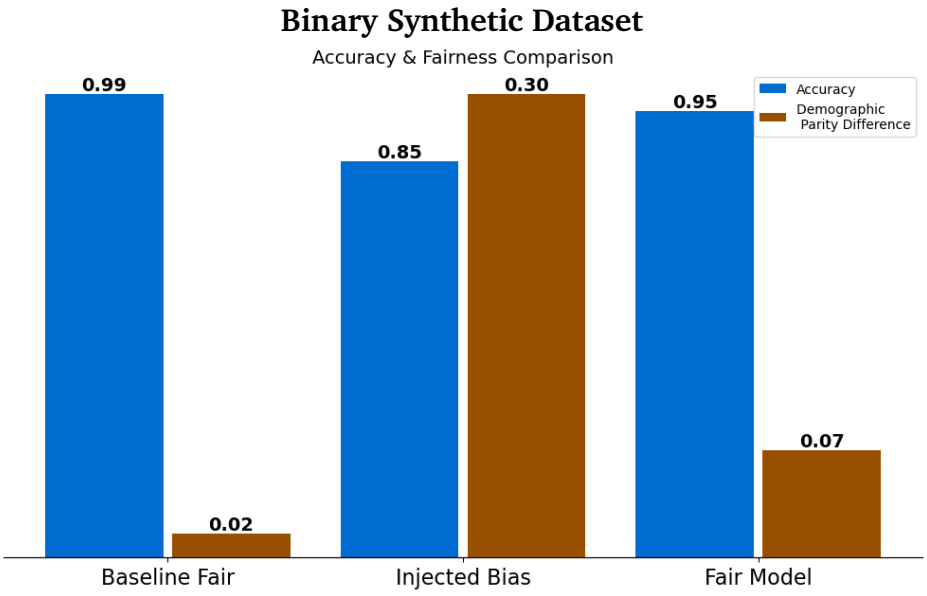
ity difference is the absolute difference between the proportions across the sensitive groups. Ideally, this value should be close to 0 as that would indicate that the groups are achieving similar proportions for each label. Example: $P(Yes\_Recidivism|African\_American) = P(Yes\_Recidivism|Not\_African\_American.)$. For multi-class we take the additional step of averaging the difference across all the labels.

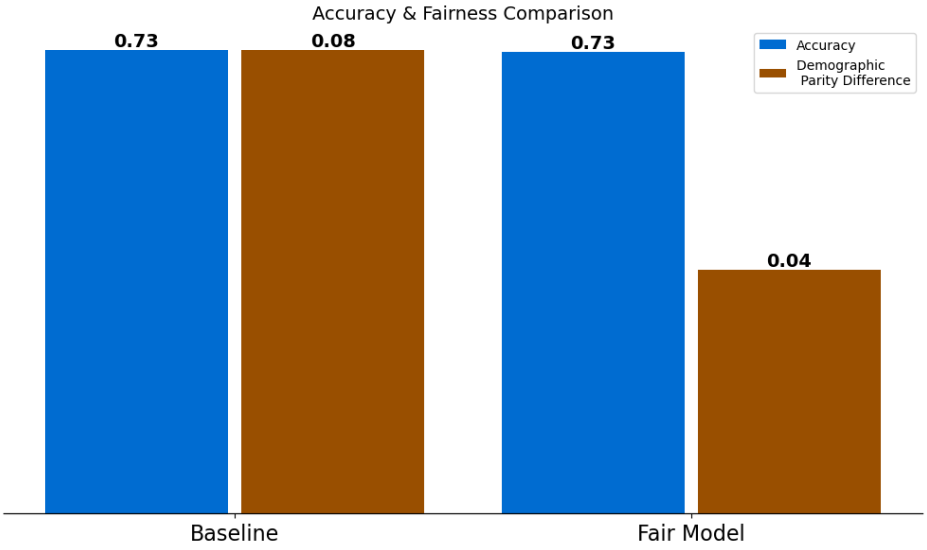
- **Area Under the Curve (AUC)**: This metric is used to measure the model's ability to evaluate how well the models can discriminate between the classes in the observed labels. In the case where the observed label is binary, AUC measures the model's ability to distinguish between the positive and negative classes. It calculates the area under the ROC curve, a curve the reflects the model's performance across all possible decision thresholds, providing discriminative ability. For multiclass cases, AUC is computing used a one-vs-rest approach, where the model is evaluate on its ability to distinguish each class from all the other. The AUC values for each class are average to provide an overall performance score.
- **Accuracy**: Accuracy is calculated as the proportion of correctly predicted labels over the total number of predictions. It serves as a general measure of model performance.
- **Equalized Odds Difference**: This metric measures the disparity in model performance between different sensitive groups, specifically comparing True Positive Rates (TPR) and False Positive Rates (FPR) for each group. For each group in the sensitive group S, we calculate the TPR and FPR, where TPR is the proportion that are correctly predicted by the model while the FPR is the proportion that are incorrectly predicted by the model. We the take the absolute difference and then the sum of the differences. $|TPR_1 - TPR_2| + |FPR_1 - FPR_2|$. For multi-class, the TPR and FPR differences are calculated fore each class individually. These differences are then averages across all the classes. The sum of the averaged differences for FPR and TPR is computed.The Equalized Odds Difference should be close to 0 as this would indicate that the model is not favoring one group over the another in terms of misclassifications rates.

For each dataset that we ran our model on, through a mixture of cross validation and hand-checking, we found the lambda value, batch size, and number of epochs that would give us values for fairness metrics and high values for performance metrics. The metrics are being evaluated on fair Y label $pseudo\_Y$ that is returned as we would like to evaluate how well the model works to establish independence between the sensitive feature and the observed label.

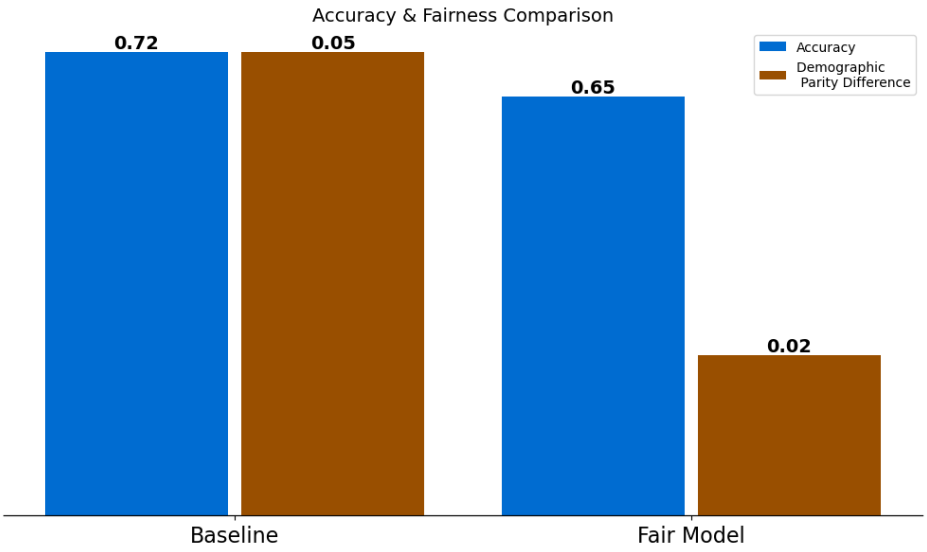
3 Results

3.1 Binary Label

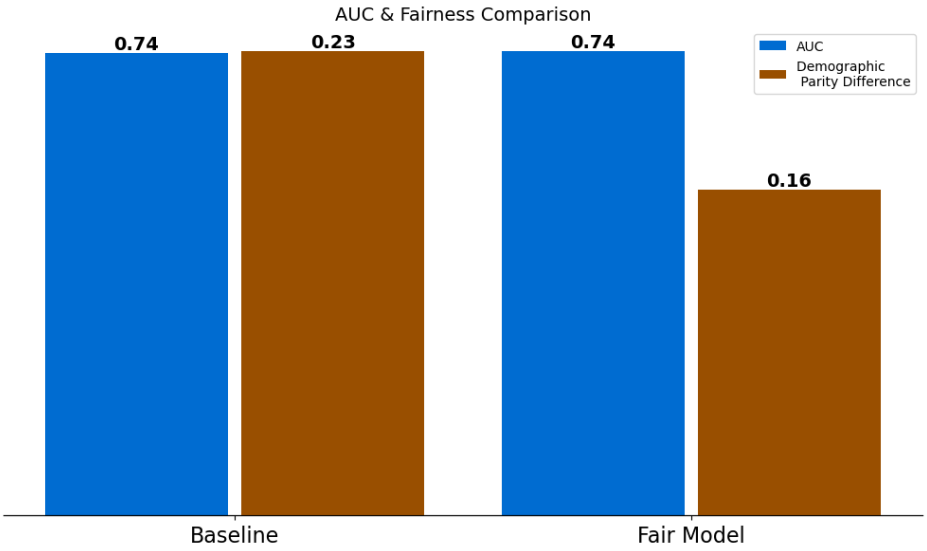




German Credit Scores

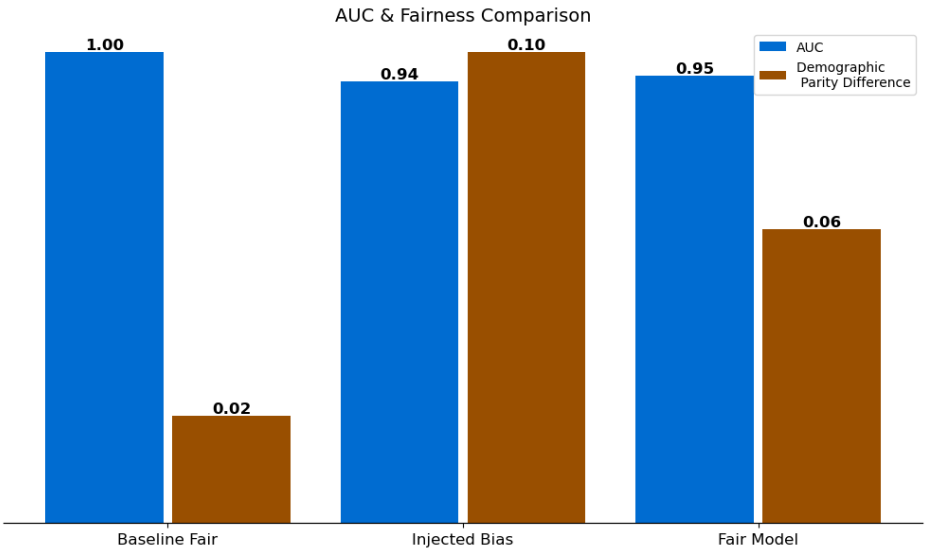


COMPAS

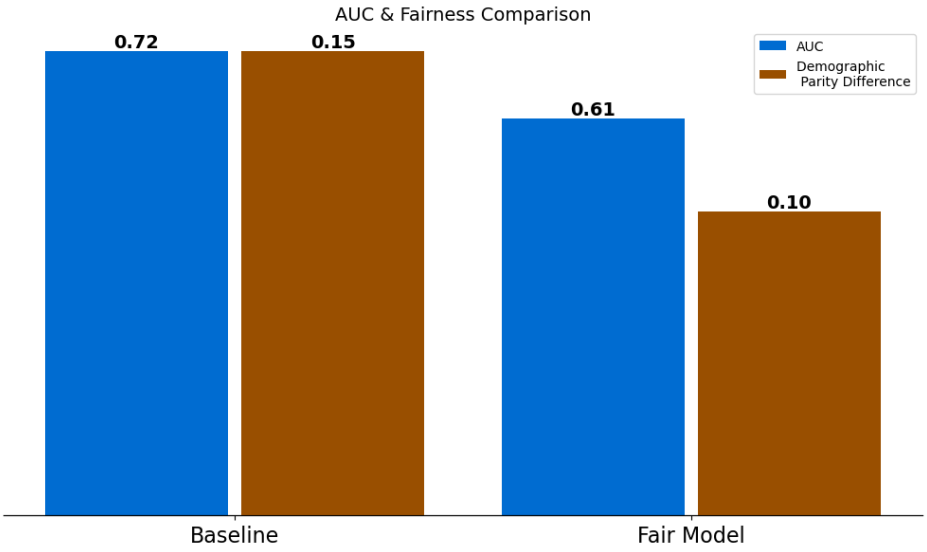


3.2 Multi-Class

Synthetic Dataset



Cannabis Consumption



4 Discussion

We tested our model on both synthetic and real world datasets for cases where the observed label is either binary and where it is multi-class. This approach allowed us to assess the model's efficacy and robustness across different scenarios. Here we provide demographic parity difference as the fairness metric we evaluate on and either AUC or accuracy as the performance metric to evaluate. To see all performance of all 4 metrics we evaluated on, see Section C in the Appendix.

In the binary data results, we found that model was effective in decreasing demographic parity difference while maintaining high performance scores, whether that be area under the curve (AUC) or accuracy. Our synthetic dataset shows that the model successfully re-

moves injected error, bringing accuracy and demographic parity difference of the fair model close to accuracy and demographic parity difference of the original fair data. With this, we show that the underlying model works in ideal scenarios where the sensitive feature and the observed label are completely independent of each other, enabling the recovery of the fair ground-truth label.

For all the real-world datasets with binary observed labels, the demographic parity decreased. Looking at the results for the UCI Adult dataset, we can see that the model effectively worked by maintaining the accuracy for both the baseline and fair model class, but decreasing the demographic parity difference by 50%. Similarly, the COMPAS dataset maintained the same AUC score while lowering the demographic parity difference by 30%. However, maintenance of such high performance is not always the case, as seen through UCI's German Credit Score results data. For this dataset, although demographic parity difference of results from the fair model was reduced by more than 50% comparing to the baseline, performance also decreased by nearly 10%. This highlights the common trade-off between fairness and performance, where applying fairness constraints can lead to a reduction in certain performance metrics.

Looking at our multi-class data results, we observed that while the model remained effective, the trade-off between demographic parity and performance (AUC/accuracy) was more pronounced compared to binary data. In the case of the synthetic dataset, the demographic parity difference of the fair model decreased relative to dataset with the injected error. However, the difference in demographic parity between the biased data and the fair model was similar to the difference between the fair data and the fair model. Moreover, the fair model's AUC score was closer to that of the biased data than the baseline fair data, indicating that the model does not perform as well when the observed label has more than 2 classes.

When evaluating the model on the Cannabis Consumption dataset, we observed a similar pattern where demographic parity difference decreased, but at the cost of decreasing the performance metric, in this case AUC. For this dataset, the demographic parity difference decreased by 33%, while the AUC score decreased by 13%. These results suggest that further work is needed to improve the model for multi-class datasets.

The results for both binary and multi-class scenarios show that given the sensitive feature and the observed label of the dataset, the recovery of ground truth variables is possible. We are able to recover the underlying distribution of the true fair label from the observed label by establishing the independence. However, this comes at the cost of performance decreasing.

4.1 Limitations

A limitation we encountered was the limited availability of datasets for evaluation fairness algorithms. While the commonly used datasets provide a useful benchmark, their limited scope and outdated nature make it challenging to assess fairness across different domains and real-world applications. Additionally, the identification of sensitive features is not al-

ways straightforward, and handling multiple sensitive features within a dataset presents further challenge.

4.2 Future Works

While our work this quarter showed efficacy of model used on both synthetic and real-world datasets, future work can be done to evaluate the robustness and strength of the model. The first would be to improve the model for cases where the sensitive feature is binary and the observed label is multi-class. More testing will need to be done a variety of datasets with a different number of classes to truly ensure that the model works in an accurate and fair model, as both the datasets that we evaluate on for multi-class had 4 classes. Another note for future works is exploring efficacy of the model where the sensitive feature is not a binary attribute. This will need to be done in cases where the observed label is binary and multi-class. Additionally, we focused on the used of $pseudo\_Y$ for evaluation, to see how well the model establishes independence between the sensitive feature and the recovered independent fair label. However, in reality, there are cases where the sensitive feature can in fact have fair dependencies with the Y. Once we are confident with the model's ability to establish independence in a variety of scenarios, the next step would be evaluating Y_{pred}, which is an output of the model. Y_{pred} will hold some fair dependencies with S, which can lead to better accuracy of the model. An additional thing to be explored is performance of model on datasets of different sizes. The largest dataset we explored was roughly 32,000 rows. However, often times dataset are much larger, so it's necessary to further explore a variety of different dataset sizes. Finally, with the use this encoder-decoder model, it is important to identify and create documentation where independence should be prioritized over accuracy and vice-versa and how to adjust the lambda value accordingly.

5 Appendix

5.1 A. Quarter 2 Report Proposal

Link: https://www.overleaf.com/read/ftkrvdmkgdng#b5f66b

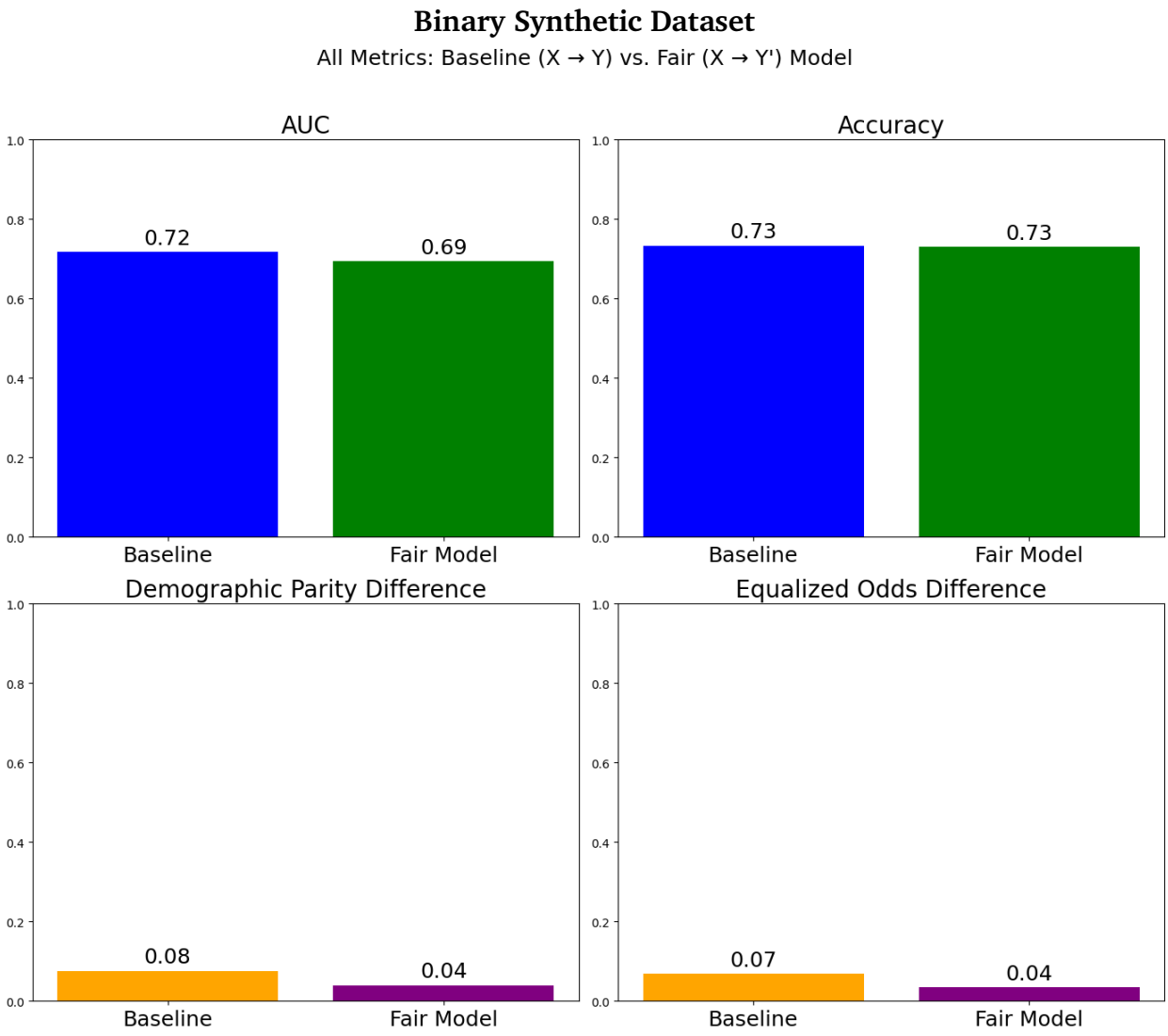
5.2 B. Preprocessing With Added Noise

When preprocessing the UCI Adults dataset and the UCI Drug Consumption dataset, we added Gaussian noise to X. This is so that the model learns more robust and generalizable patterns by preventing overfitting to specific feature values and making it more resistant to small variation in the data. For the UCI Adults dataset, we also added noise to Y so that the model becomes more robust to label uncertainty, allowing for real-world inconsistencies in classification. This can help improve generalization. For these two dataset, added noise improved results.

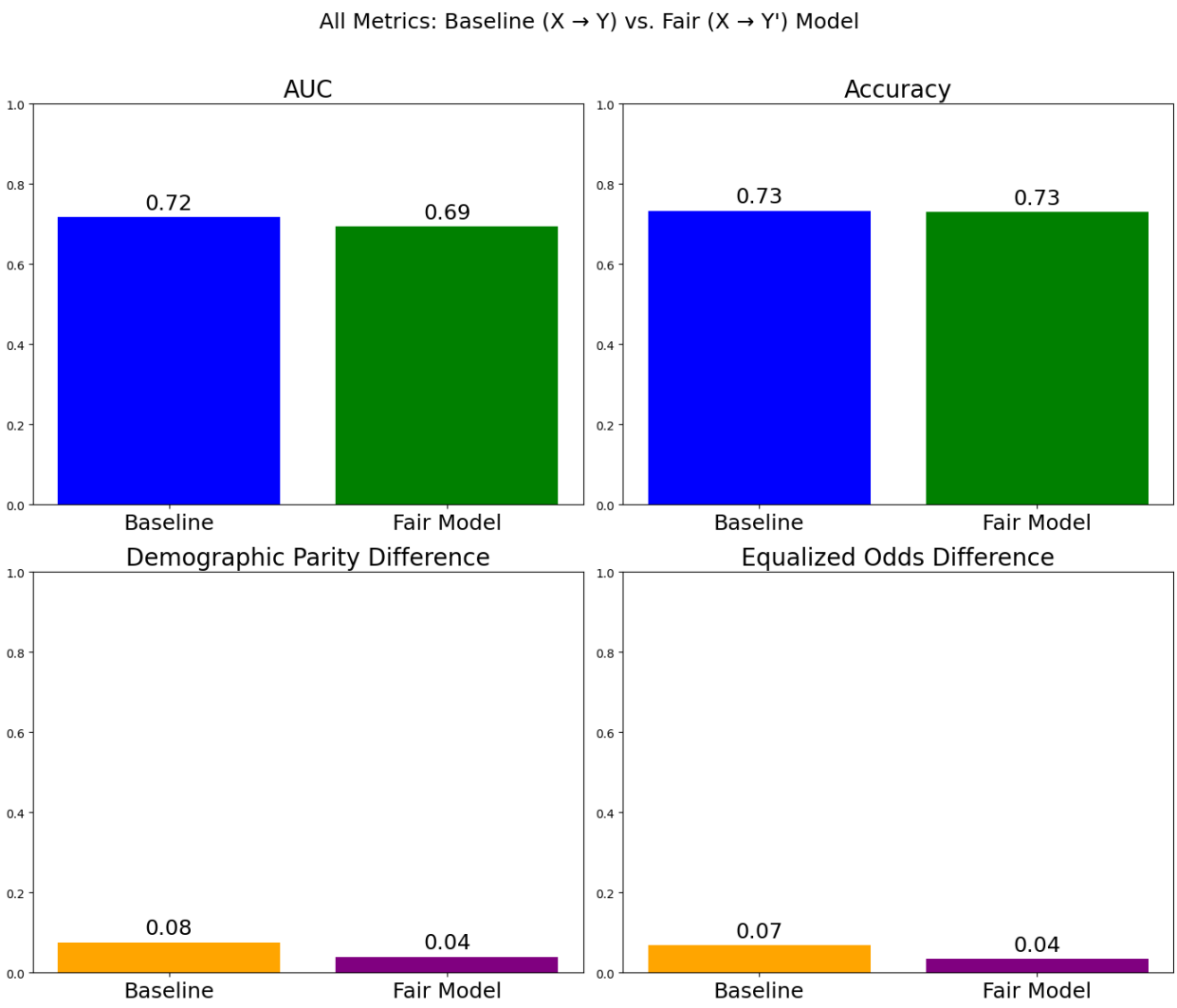
5.3 C. Metrics for All Datasets Run

The graphs below include all 4 metrics we evaluated for each datasets. These are different fairness and performance metrics that were evaluated.

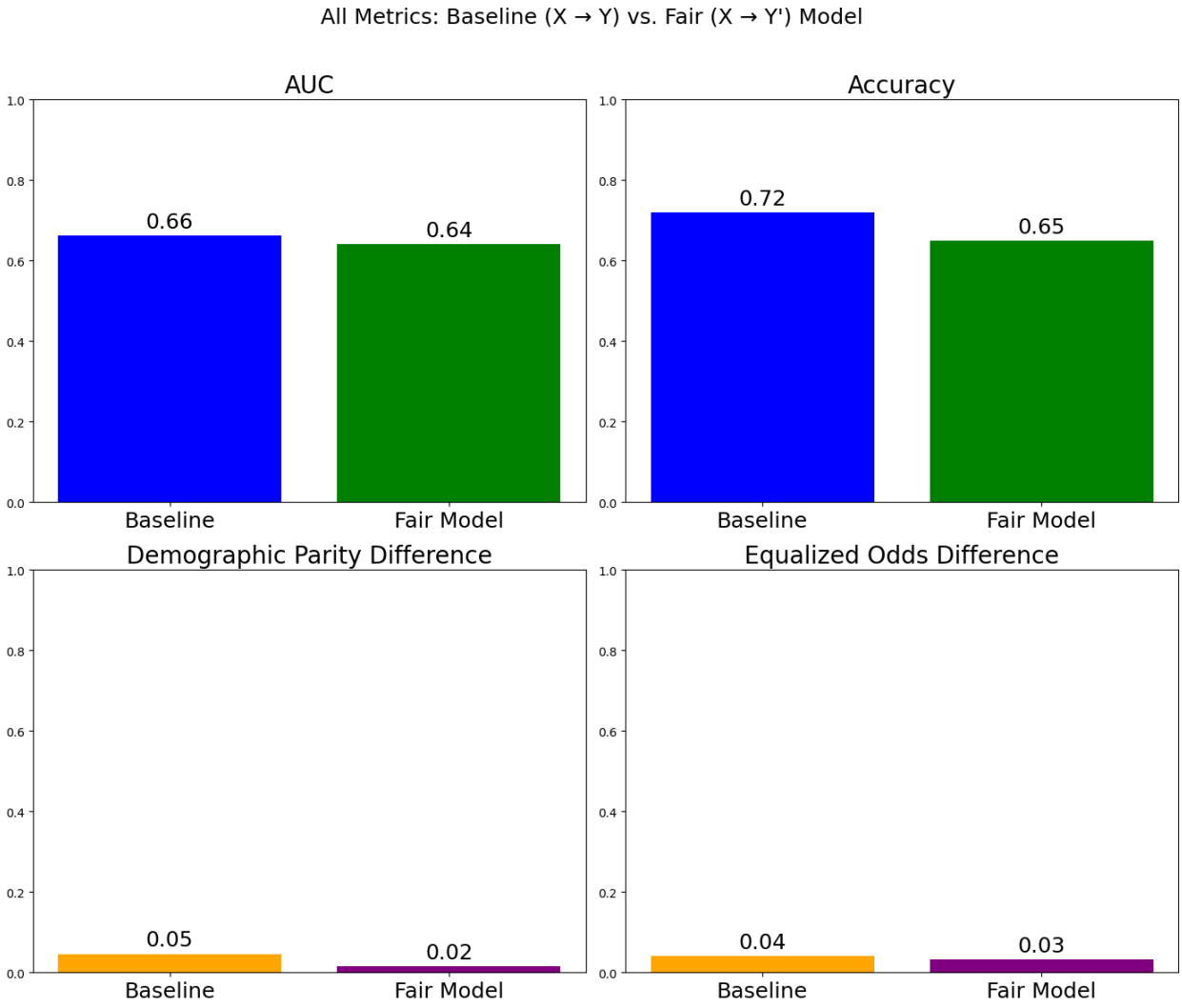
Binary Label



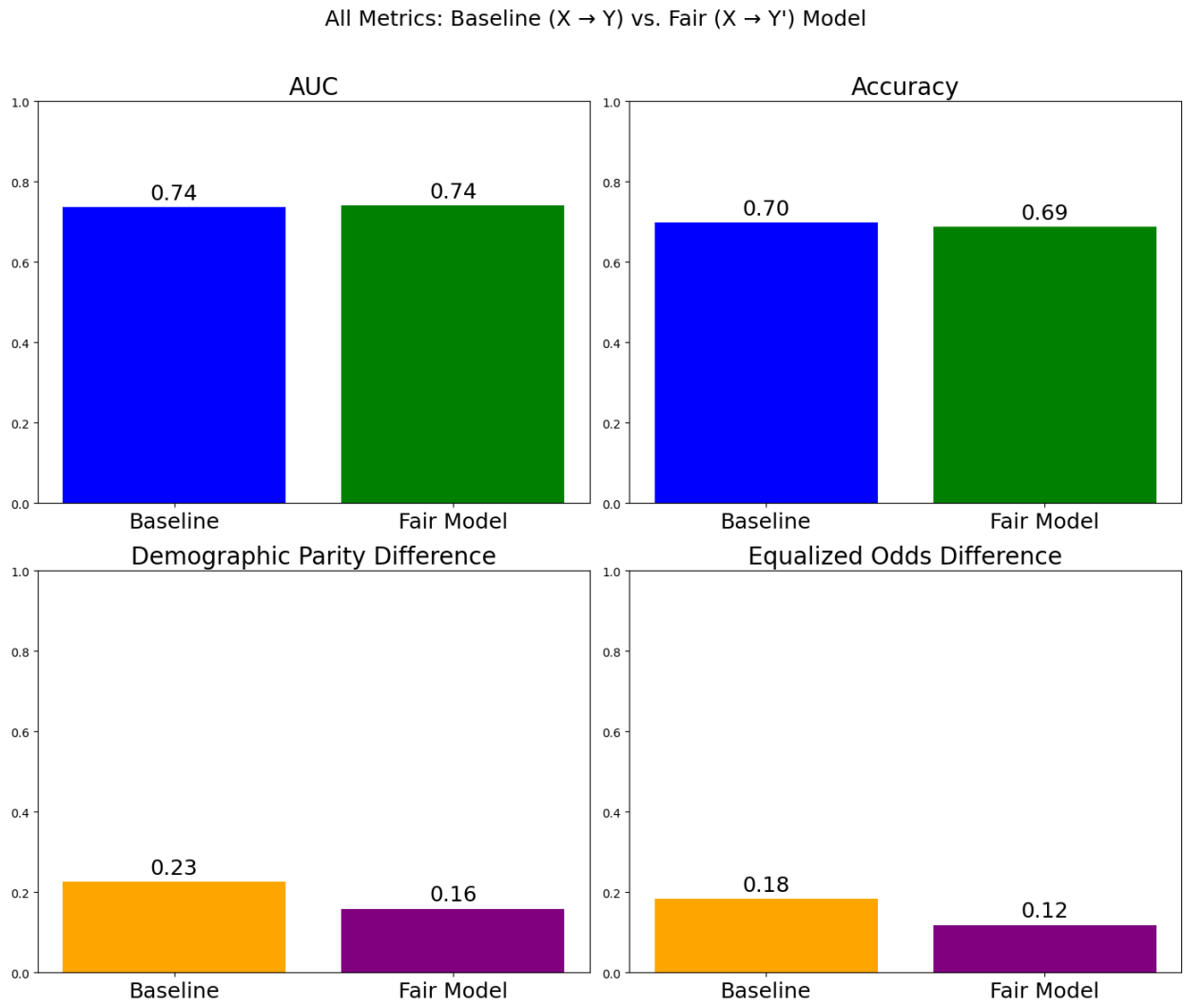
UCI Adults



German Credit Scores

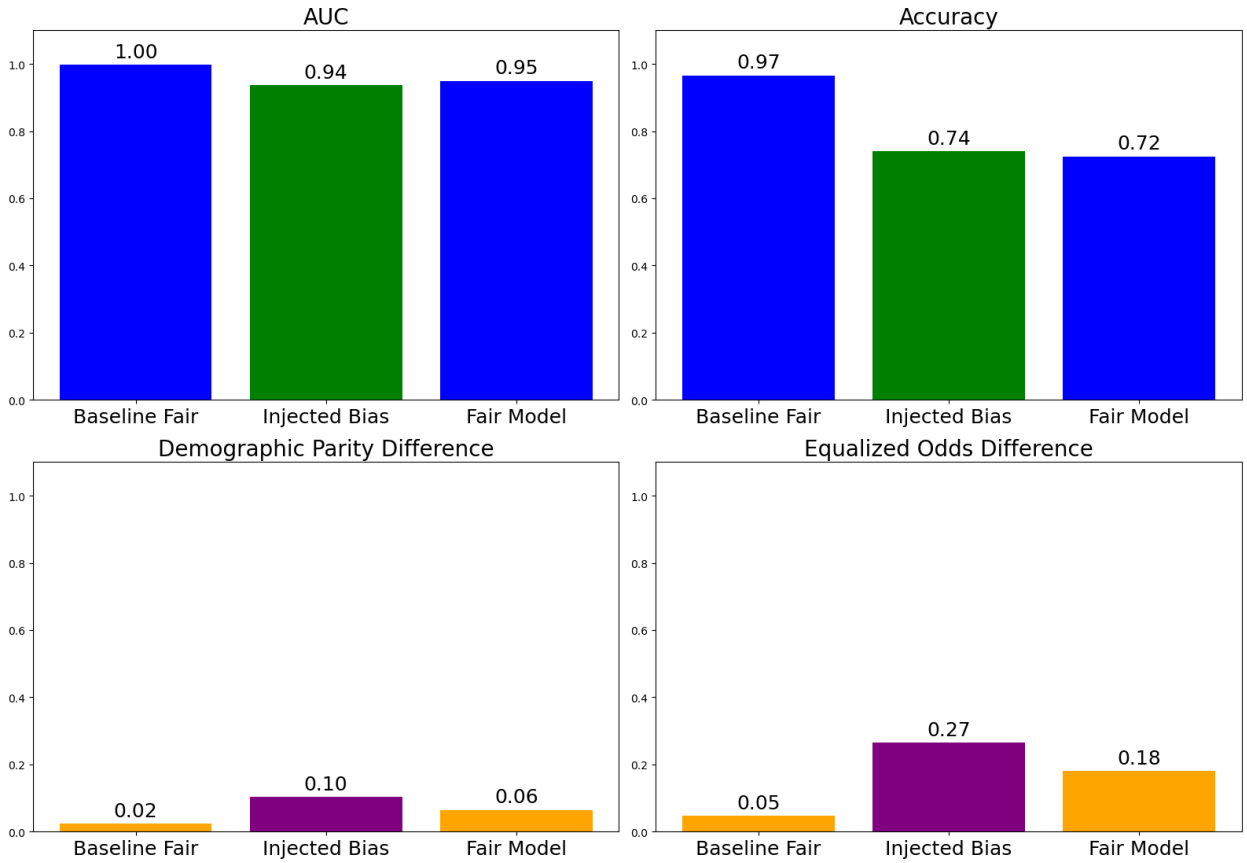

All Metrics: Baseline (X → Y) vs. Fair (X → Y') Model
AUC
0.66
0.64
Baseline
Fair Model
Accuracy
0.72
0.65
Baseline
Fair Model
Demographic Parity Difference
0.05
0.02
Baseline
Fair Model
Equalized Odds Difference
0.04
0.03
Baseline
Fair Model

COMPAS

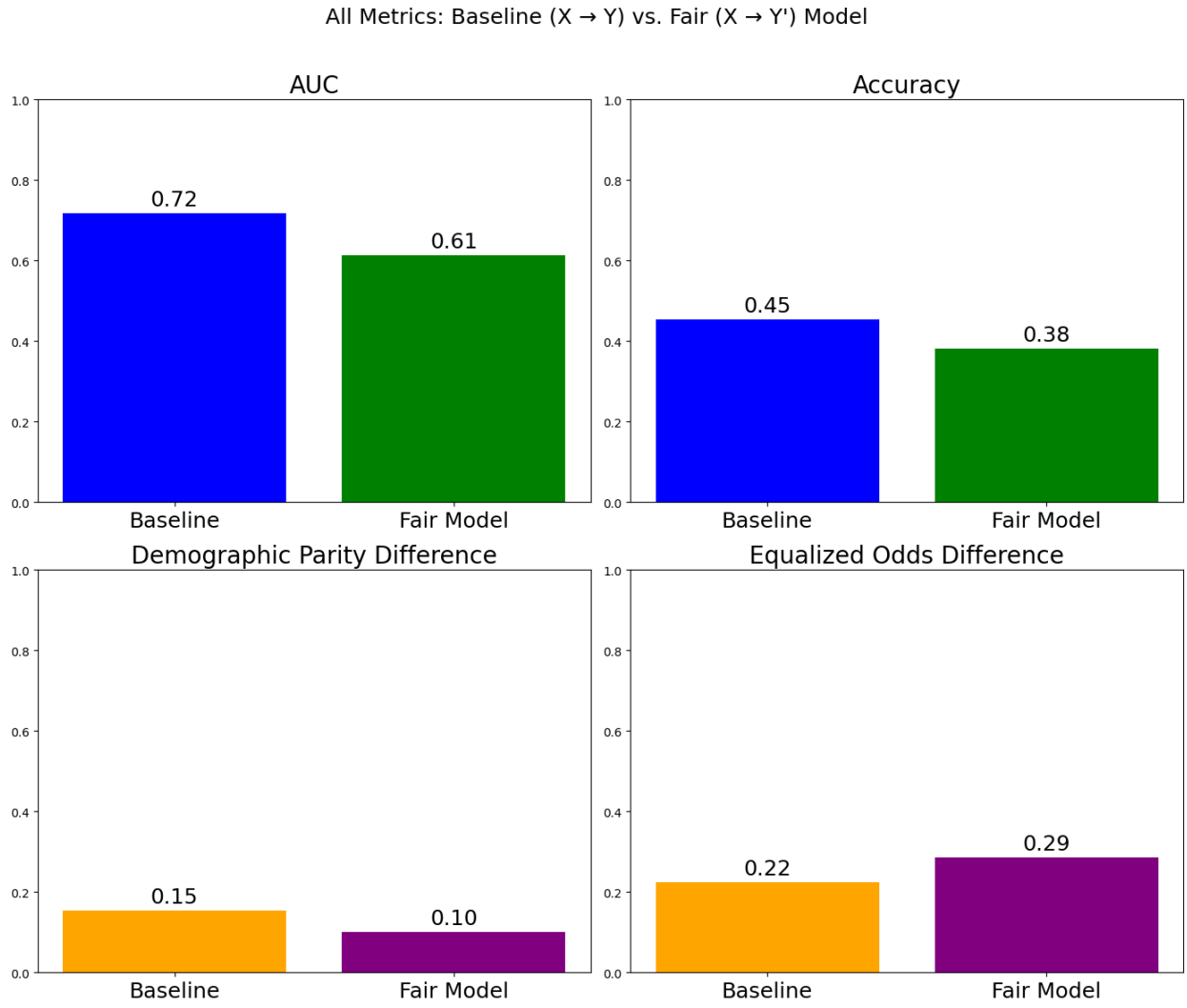


Multi-Class Label

Synthetic Dataset



Cannabis Consumption



6 Contributions

Lina Battikha:

- Abstract, Introduction, Literature Review, Datasets, Methods, Results, Discussion
- Cleaned up code + README.md for reproducibility
- Completed all code and retrieved results from datasets
- Edit and Completed Poster
- Edited Website

Sai Poornasree Balamurugan:

- Focused work on website
- Creating and started initial process of cleaning report
- Introduction, Methods, Datasets implemented
- Working on implementing multi-classification for Health Readmission datatset

- Set up poster outline and content

References

Choi, YooJung, Meihua Dang, and Guy Van den Broeck. 2021. "Group Fairness by Probabilistic Modeling with Latent Fair Decisions." 35 : 12051–12059. [Link]

Khadija, Obey. 2022. "Drug Consumptions UCI Dataset." [Link]

Prashant, Parjanya Prajakta, Seyedeh Baharan Khatami, Bruno Ribeiro, and Babak Salimi. 2025. "Scalable Out-of-Distribution Robustness in the Presence of Unobserved Confounders." In *Proceedings of the 28th International Conference on Artificial Intelligence and Statistics (AISTATS)*. [Link]

ProPublica. 2016. "COMPAS Recidivism Analysis." [Link]

UCI Machine Learning Repository. 1994. "Statlog (German Credit Data) Data Set." [Link]

UCI Machine Learning Repository. 1996. "Adult Data Set." [Link]