Bias Mitigation in AI Models for Cardiovascular Diseases Prediction

Giorgia Castelli, Alice Fratini, Madalina Ionela Mone

Abstract

Bias in artificial intelligence algorithms used in healthcare is a growing concern due to its potential to negatively impact patient outcomes. AI tools, despite their power, can perpetuate systemic unfairness towards certain groups. This is particularly critical in healthcare, where existing biases can exacerbate inequalities. Historical examples, such as the Framingham Risk Score—part of the 2018 CV Risk Calculator developed by the American Heart Association and the American College of Cardiology, which estimates the 10-year and lifetime risk of atherosclerotic cardiovascular disease based on factors like age, sex, race, cholesterol, blood pressure, and lifestyle—demonstrate how biases in traditional medical tools have led to variable accuracy across different ethnic groups and genders, due to the underrepresentation of diverse cohorts in their development[1]. Women, for instance, experience higher rates of missed and delayed cardiovascular disease diagnoses, partly because of their underrepresentation in clinical trials. While some level of bias in data and tools is inevitable, the validation of AI tools to minimize bias offers a promising opportunity to reduce disparities in healthcare outcomes.

1 Introduction

In our project, we set out to explore the presence and mitigation of bias in artificial intelligence models applied to healthcare, focusing on cardiovascular disease diagnosis. Utilizing the Heart Attack Prediction Dataset available on Kaggle [2], which includes real patient data such as age, sex, blood pressure, cholesterol levels, and lifestyle indicators, we trained well-known machine learning algorithms to evaluate predictive performance and detect disparities in model behavior. The core of our investigation lies in understanding how bias—often hidden within data distributions or model assumptions—can affect fairness and accuracy, particularly in a medical context with potentially serious consequences for patients.

To achieve this, we initially used the PyCaret library to train and evaluate baseline models. We then integrated Aequitas-FairLib toolkit to assess fairness metrics and uncover patterns of discrimination within the dataset. Building on these insights, we implemented and tested a set of established bias mitigation techniques. In particular, we adopted a comprehensive strategy by applying methods across all three main categories of bias mitigation: **pre-processing**, **in-processing**, and **post-processing**. This taxonomy, widely adopted in fairness-aware machine learning research, is also reflected in the comprehensive survey by Hort et al. (2023) [5], which analyzed over 300 studies on bias mitigation strategies for classification models. This approach allowed us to compare the effectiveness of different techniques and to observe how model outcomes changed depending on the type of intervention.

This report presents a detailed overview of our approach, including the methodology, evaluation metrics, and tools adopted. It is accompanied by Jupyter notebooks containing the full experimental pipeline. Through this work, we aim to contribute to the broader discourse on fairness in AI, providing a concrete case study of how bias can be measured and managed in real-world health data.

2 Importance of Bias in Machine Learning Algorithms for Healthcare Applications

Machine learning (ML) is playing an increasingly central role in healthcare, offering powerful tools for diagnosis, prognosis, and clinical decision support. However, this potential comes with a critical caveat: ML models often learn from biased data and risk reinforcing structural inequalities already present in the healthcare system. When biases are left unchecked, the consequences may include misdiagnoses, inequitable treatment recommendations, and loss of trust among underrepresented patient groups. As highlighted by Ganapathi et al. [3], many AI health datasets fail to adequately represent diverse populations, and current regulatory frameworks often lack robust requirements for dataset documentation and demographic transparency.

In this report, we address the importance of bias awareness in ML applications for cardiovascular risk prediction. Our approach includes evaluating the fairness of trained models using PyCaret and the Aequitas-FairLib library, as well as applying bias mitigation strategies. Building on prior work such as Paladino et al.[4], who emphasize the variability of AutoML performance depending on dataset composition, we stress that fairness must be considered alongside accuracy to ensure clinical utility across diverse populations.

2.1 How ML Models Develop Bias & The Potential Consequences in Healthcare

ML models generate predictions by learning patterns from historical data, but when these data are incomplete, unbalanced, or improperly labeled, models can reproduce or amplify unfair treatment. For example, a risk scoring tool may be trained predominantly on data from a specific demographic (e.g. middle-aged white males), leading to poor generalization for others. Underrepresented groups may then receive inaccurate risk assessments, suboptimal interventions or delayed care. Ganapathi et al.[3] note that even when models are validated, insufficient diversity in test sets can hide performance issues affecting minorities.

Additionally, bias can stem from how datasets are built—decisions on what to measure, how to label, and who to include often reflect broader social and institutional inequities. Paladino et al. [4] further show that even advanced ML tools like AutoML are sensitive to these biases, suggesting that automation alone does not guarantee fairness. Without proactive strategies, such as demographic-aware preprocessing, subgroup evaluation, and transparent documentation, ML systems may ultimately reinforce the very disparities they aim to reduce.

3 Bias Mitigation Methods

Bias mitigation in healthcare ML algorithms is crucial to ensuring fairness, accuracy, and trustworthiness in patient care. Numerous bias mitigation strategies have been proposed in the literature, and they are commonly categorized into three main groups: **pre-processing**, **in-processing**, and **post-processing** methods. This classification refers to the stage at which the intervention is applied in the machine learning pipeline - before, during, or after model training.

In our work, we adopted representative techniques from each category. This organization follows the taxonomy presented by Hort et al. (2023) [5], who conducted an extensive review of 341 publications on bias mitigation for ML classifiers. Their survey provides a detailed overview of methods, metrics, and evaluation practices, offering a useful framework for researchers and practitioners tackling algorithmic bias.

Below, we summarize the most widely used methods in the literature for each category.

3.1 Pre-processing

Pre-processing methods aim to address bias before the training phase of the ML model. This can involve techniques to modify or restructure the training data to remove discriminatory information or ensure equitable representation of different groups. Common pre-processing techniques include:

- *Disparate Impact Remover*: Perturbation of the DataFrame in order to get the distributions of privileged and unprivileged groups closer; the preservation of rank-ordering within groups is also taken into account.

- *Learning Fair Representations (LFR)*: This technique aims to learn data representations that are either independent of sensitive attributes or that reduce their discriminatory impact. This may involve using dimensionality reduction techniques or generative adversarial networks (GANs). For example, one study explored using variational autoencoders, normalizing flows, and adversarial learning to generate fair representations by applying nonlinear transformations and dropping biased features.

- *Reweighing*: Assigning different weights to data instances based on their sensitive attributes can help mitigate the influence of biased attributes.
- *Perturbation*: perturbation of each non-protected attribute, through the modifications of its label, ensuring that its distribution is brought closer together while preserving the ranks within each group.
- *Downsampling*: reducetion of the number of instances in the privileged group.
- *SMOTE Algorithm*: generating synthetic elements by interpolating between a random row and one of its k-nearest neighbours.

3.2 In-processing

In-processing methods intervene directly during the training phase of a machine learning model. Unlike pre-processing techniques, which modify the data, or post-processing methods, which adjust the model's output, in-processing algorithms incorporate fairness constraints into the training process itself. These methods aim to optimize the model simultaneously for both predictive performance and fairness, ensuring that the final classifier treats different demographic groups more equitably. Common in-processing techniques include:

- *Adversarial Learning*: This approach involves training a predictor alongside an adversary. The predictor learns to make accurate predictions, while the adversary tries to infer the sensitive attribute from the learned representations. The predictor is penalized if the adversary succeeds, thus encouraging representations that are fair and insensitive to group membership.
- *Fairness Regularization*: These techniques add a penalty term to the model's loss function to penalize unfair predictions. This encourages the learning process to minimize bias-related disparities (e.g., disparate impact, false discovery rate) while preserving predictive power.
- *Exponentiated Gradient Reduction*: This method reduces fair classification to a sequence of cost-sensitive classification problems. It outputs a randomized classifier that balances accuracy and fairness under a given constraint.

3.3 Post-processing

Post-processing methods aim to mitigate bias after the ML model has been trained. These methods typically involve adjusting the model's outputs or decision rules to ensure fairness. Post-processing techniques include:

- *Calibration*: Calibrating the model's outputs separately for different groups can help ensure that the predictions are equally accurate across groups.

- *Classification Thresholds*: Adjusting classification thresholds for different groups can help improve fairness by minimizing false positive or false negative rates for disadvantaged groups. For example, one study showed that by splitting a classifier into two, one for the privileged group and one for the unprivileged group, and then adjusting the classifier for the unprivileged group, it was possible to balance the false positive and false negative rates of the two classifiers.
- *Bias Explanation and Correction*: Using explainability techniques to understand and correct for biases in the model's predictions can help improve fairness. This can involve using Shapley values or other feature attribution techniques to identify and mitigate biased features. For example, one method involves modifying predictions by creating counterfactual predictions. By determining the difference between a prediction in the real world and a counterfactual world where an individual belongs to a different population group, it is possible to correct the impact of the protected attribute on the prediction outcome, ensuring that it aligns with counterfactual predictions.

4 What we did

PyCaret is an automated machine learning tool that simplifies the process of building and evaluating ML models. Aequitas-FairLib is a library that provides a suite of fairness metrics and algorithms for assessing and mitigating bias. To evaluate the effectiveness of different bias mitigation techniques, one can use PyCaret to train models on a healthcare dataset and utilize Aequitas-FairLib to assess and compare their performance in terms of precision, recall and fairness.

4.1 Baselines for evaluation

Establishing a robust and reliable baseline model was a crucial initial step in the bias mitigation process. A baseline provides a clear reference point for assessing initial predictive performance and for conducting a preliminary evaluation of bias, enabling more reliable comparisons with subsequent mitigation strategies. For this purpose, we implemented a baseline prediction model using the automated machine learning (AutoML) tool PyCaret.

Initially, we planned to establish two separate baseline models using different AutoML frameworks, namely PyCaret and AutoGluon. However, this two-baseline strategy was revised during implementation due to technical compatibility issues encountered with AutoGluon, particularly in its integration with the existing project environment and data preprocessing pipeline. These limitations hindered its reliable deployment within the project timeframe. As a result, AutoGluon was excluded, and PyCaret was retained as the sole AutoML tool for baseline model construction. PyCaret was selected due to its ease of use, broad compatibility, and streamlined workflow for training, tuning, and evaluating machine learning models. Its capability to rapidly compare multiple algorithms allowed us to identify a strong and reliable baseline model without extensive manual hyperparameter optimization. This baseline was primarily employed in the evaluation of pre-processing bias mitigation techniques, where a standard, architecture-agnostic reference model is sufficient to assess the impact of data-level interventions on both performance and fairness.

For in-processing bias mitigation, a different baseline strategy was adopted. In-processing methods, such as Adversarial Debiasing, directly modify the learning process by incorporating fairness constraints into the model optimization. Consequently, meaningful comparisons require a consistent model architecture. For this reason, an Adversarial Debiasing model trained without an active debiasing component (i.e., with the adversarial debiasing weight set to zero) was used as the in-processing baseline. This approach enabled a controlled comparison with subsequent trainings of the same architecture using non-zero debiasing weights, allowing us to isolate and evaluate the effect of in-processing fairness constraints on the bias–performance trade-off.

4.2 Metrics

Metrics for Model Performance and Fairness Evaluation In evaluating the models developed during this project, we employed two sets of metrics: one aimed at measuring the predictive performance of the models, and the other focused on assessing fairness, particularly in relation to potential biases between minority and majority groups.

To assess the predictive capacity of the models, we utilized the following metrics:

- *Precision*: Precision is the proportion of true positives out of all instances that the model predicted as positive. It highlights the model's ability to avoid false positives. A high precision indicates that when the model predicts a positive outcome, it is usually correct.

- *Recall*: Also known as sensitivity or true positive rate, recall measures the proportion of actual positives that the model correctly identified. It is particularly useful in cases where missing positive instances (false negatives) is costly.

- *F1 Score*: The F1 score is the harmonic mean of precision and recall. It provides a single metric that balances both concerns, making it especially useful when the class distribution is imbalanced or when both false positives and false negatives are important to consider.

In addition to predictive performance, evaluating the fairness of our models was a central objective. To measure fairness, we employed several bias detection metrics, which helped us assess how equally the models treated different demographic groups, particularly the minority and majority classes:

- *Disparate Impact*: This measures the ratio of positive outcomes between two groups (typically, minority and majority). A disparate impact ratio close to 1 indicates fairness, while values far from 1 suggest the model may be biased against one group.

- *Equality of Opportunity*: This fairness metric focuses on whether individuals who truly belong to the positive class are equally likely to be identified as such across groups. It ensures that true positive rates are comparable between minority and majority groups.

- *Statistical Parity*: This metric assesses whether the probability of receiving a positive prediction is the same across different groups. If statistical parity holds, the model does not favor one group over another in its predictions.

- *Equalized Odds Ratio*: This metric evaluates whether both true positive rates and false positive rates are similar across different groups. A model satisfies equalized odds if it treats groups equally not only in correctly identifying positive cases but also in avoiding incorrect positive predictions. It provides a more comprehensive view of fairness by accounting for multiple types of errors.

By using these metrics, we were able to not only assess how well the models performed in terms of predictive accuracy but also to ensure that fairness considerations were taken into account, mitigating potential biases that could disproportionately affect certain groups.

4.3 Pre-processing techniques

To address the biases detected in our initial dataset, we implemented a series of six pre-processing techniques, whose primary aim is to mitigate the bias originating from the imbalance in the number of instances between the privileged and unprivileged groups.

The following six techniques were selected for their diverse approaches in handling biases:

- *Disparate Impact Remover*: This technique adjusts the features in the dataset to reduce the differences between the distributions of privileged and unprivileged groups, while ensuring that the rank-ordering within each group is preserved. By doing so, it minimizes the disparate impact while maintaining the internal structure of the data, reducing potential biases linked to sensitive attributes.

- *Learning Fair Representations (LFR)*: LFR aims to learn a latent representation of the data that encodes information effectively while mitigating bias related to protected attributes. The technique generates a new representation that balances the predictive power and fairness. By covering information about sensitive attributes in the learned representation, it reduces the influence of these attributes on the model's predictions, resulting in fairer outcomes across different demographic groups.

- *Reweighing*: This method ensures fairness concerning the protected attribute by assigning different weights to instances based on their group membership (privileged or unprivileged). However, since PyCaret does not natively support instance weighting, we utilized the assigned weights to oversample instances from underrepresented groups. This adjustment helps balance the training process, allowing the model to learn equitably from both privileged and unprivileged groups.
- *Perturbation*: The perturbation technique involves modifying each non-protected attribute in the dataset to bring its distribution closer across privileged and unprivileged groups. This is achieved by slightly altering the attribute labels, aiming to equalize the attribute distributions while preserving the rank order within each group. This approach ensures that the characteristics of the groups become more similar without significantly distorting the data.
- *Downsampling*: Downsampling is a straightforward technique that reduces the number of instances in the privileged group to match the size of the unprivileged group. By decreasing the overrepresentation of the privileged group, downsampling helps mitigate biases arising from the imbalance in the dataset, fostering fairer learning conditions during model training.
- *Upsampling with SMOTE Algorithm*: The Synthetic Minority Over-sampling Technique (SMOTE) generates synthetic data points for the underrepresented (unprivileged) group by interpolating between a random instance and one of its k-nearest neighbors. This technique increases the number of instances in the unprivileged group, creating a more balanced dataset. SMOTE helps the model learn from a richer, more diverse set of examples, reducing the risk of bias favoring the privileged group.

For the first three pre-processing experiments we utilized functions provided by the Aequitas-FairLib library, while for the last one we employed a function from the Imblearn package. To support intersectionality, i.e. the mitigation of bias across multiple sensitive attributes leading to several disadvantaged subgroups, we introduced a custom approach where needed. Specifically, in methods that do not natively support intersectionality, we defined a single intersectional variable representing the Cartesian product of the protected attributes, and applied the mitigation techniques on this combined variable. In contrast, the Reweighing method inherently supports intersectionality and thus required no such adaptation.

After applying each of the six pre-processing techniques, we trained a new machine learning model using the automated PyCaret tool. PyCaret was chosen for its efficient pipeline, allowing rapid experimentation with various algorithms and hyperparameter tuning. The models were trained on the transformed datasets resulting from each pre-processing technique.

To evaluate the impact of these techniques, we conducted a second bias evaluation step using the Aequitas-FairLib toolkit.

4.4 In-processing techniques

To mitigate the biases identified during our initial evaluations, we applied one in-processing technique that operates during the model training phase. These methods aim to embed fairness constraints directly into the learning process, encouraging the model to learn unbiased representations and reduce disparities across sensitive groups.

We selected the following in-processing technique:

- *Adversarial Debiasing*: Adversarial learning involves training a model to make accurate predictions while simultaneously reducing its ability to infer sensitive attributes, such as gender or race. This is achieved through a two-network architecture known as *Adversarial Debiasing* (Zhang et al., 2018), which is an in-processing fairness algorithm. The architecture consists of a **predictor network**, which learns to predict the target variable, and an **adversary network**, which attempts to predict the sensitive attribute from the predictor's internal representations or outputs.

  The networks are trained in an adversarial manner: the predictor aims to maximize prediction accuracy while minimizing the adversary's ability to infer the sensitive attribute, whereas the adversary strives to extract that attribute from the predictor's outputs. A *gradient reversal layer* is typically used to ensure that the predictor learns features that are both predictive and fair. The strength of the fairness constraint is controlled by a parameter λ (the adversary weight):

  - $\lambda = 0$: no fairness constraint (standard model)
  - $\lambda > 0$: increasing values enforce stronger fairness constraints

To implement the *Adversarial Debiasing* in-processing technique, we trained multiple models sharing the same architecture but differing in the strength of the adversarial debiasing component. Specifically, a baseline configuration with no active debiasing and three additional variants with increasing adversary weights ($\lambda = 0.1, 0.5, 1.0$) were considered. For each configuration, fairness metrics were evaluated to analyze how the strength of the fairness constraint influences the trade-off between predictive performance and bias mitigation.

4.5 Post-processing Techniques

Post-processing methods mitigate bias by adjusting the model's predictions after training, without altering the underlying algorithm or the training data. In this study, we utilized the IBM AIF360 library to implement three distinct approaches:

- **Equalized Odds Postprocessing (EqOdds)**: This method solves a linear program to optimize probabilities, aiming to equalize the True Positive Rate (TPR) and False Positive Rate (FPR) across privileged and unprivileged groups. It focuses on equalizing error rates.

- **Calibrated Equalized Odds**: This technique optimizes the output probabilities to ensure the model is calibrated (i.e., the predicted probability reflects the true likelihood of the outcome) while minimizing the error disparity between groups.
- **Reject Option Classification (ROC)**: This method addresses uncertainty in the decision boundary. It identifies samples with low confidence (near the decision threshold) and assigns them a favorable outcome if they belong to an unprivileged group (or unfavorable for the privileged group), effectively shifting the classification threshold to improve fairness.

5 Experimental Results

In this section, we present the results obtained from our bias mitigation experiments, comparing the baseline model and the models trained on datasets transformed through different pre-processing techniques. The experiments aimed to evaluate the impact of these techniques on both classification performance and fairness metrics.

5.1 Baseline Model

The automated baseline model, trained using PyCaret without any bias mitigation pre-processing, yielded the following performance metrics:

Class	**Precision**	**Recall**	**F1-Score**
0 (Negative)	0.64	0.87	0.74
1 (Positive)	0.37	0.13	0.20
Macro Avg	0.51	0.50	0.47
Weighted Avg	0.55	0.61	0.55

Table 1: Baseline model performance: precision, recall, and F1-score by class.

These scores indicate that the baseline model's predictive capabilities are limited, as both metrics are relatively low. While there is room for optimization, improving the model's predictive power was not the primary goal of this project. Instead, our focus was on assessing and mitigating potential biases.

As shown in Table 2, the baseline model demonstrates relatively small disparities across most fairness metrics, particularly for Disparate Impact and Statistical Parity Difference, where values are close to the ideal thresholds. However, the Equality of Opportunity values reveal a notable gap between groups, and the Equalized Odds Ratio is substantially below 1, indicating imbalance in the model's error distribution. These findings suggest that, despite seemingly balanced predictions, the model may still propagate unfair treatment in more nuanced ways. Consequently, we explored several bias mitigation techniques to address these residual disparities and improve fairness across sensitive attributes.

Fairness Metric	**Group Comparison**	**Value**
Disparate Impact	Sex (0 vs 1)	[0.996 – 1.007]
	Hemisphere (0 vs 1)	[0.994 – 1.005]
Equality of Opportunity	Sex (0 vs 1)	±0.058
	Hemisphere (0 vs 1)	±0.088
Statistical Parity Difference	Sex (0 vs 1)	±0.0028
	Hemisphere (0 vs 1)	±0.0019
Equalized Odds Ratio	All groups	0.527

Table 2: Fairness metrics for the baseline model (no bias mitigation).

5.2 Pre-processing Experiments

Below, we summarize the results of the six experiments conducted using different pre-processing techniques on the initial dataset.

- *Experiment 1*: The experiment with *Disparate Impact Remover* maintained predictive performance close to the baseline, with a similar F1-score and low recall for the positive class. However, there is a noticeable improvement in *Equality of Opportunity*, particularly with respect to sex, and a slight increase in the *Equalized Odds Ratio*, indicating reduced disparity in error distribution. Other fairness metrics remained substantially unchanged.

- *Experiment 2*: The *Learning Fair Representations (LFR)* experiment preserved fairness metrics close to the baseline, but worsened *Equality of Opportunity* and drastically reduced model performance for the positive class, with an *F1-score* of just 0.06.

- *Experiment 3*: The *Reweighing* experiment reproduced both the predictive performance and fairness metrics of the baseline, showing no meaningful improvement in either fairness or classification outcomes.

- *Experiment 4*: The *Perturbation* experiment led to a sharp drop in recall and F1-score for the positive class, significantly degrading predictive utility. While *Equality of Opportunity* improved slightly, the *Equalized Odds Ratio* dropped to zero. Disparate Impact and Statistical Parity metrics showed minor deviations but remained within acceptable ranges.

- *Experiment 5*: The *Downsampling* experiment caused a complete collapse in performance for the positive class, with zero recall and F1-score, severely limiting model usefulness. Fairness results are mixed: while *Equality of Opportunity* remains balanced, *Disparate Impact* and *Statistical Parity Difference* show larger deviations than in the baseline. The *Equalized Odds Ratio* could not be computed, due to absence of true positives.

- *Experiment 6*: The *SMOTE* experiment improved overall accuracy but failed to enhance performance for the positive class, with very low recall and F1-score. Despite balanced *Equality of Opportunity*, other fairness metrics deteriorated significantly: *Disparate Impact* and *Statistical Parity Difference* showed high imbalance, and the *Equalized Odds Ratio* dropped sharply.

Overall, the pre-processing experiments highlight the challenges of applying fairness techniques in a setting with multiple sensitive attributes and automated model pipelines: the presence of two protected variables increases complexity by introducing intersectional subgroups, which many standard mitigation methods are not designed to handle effectively. Additionally, the use of an AutoML framework that does not natively support instance weighting—limited the applicability of reweighing-based strategies and required custom adaptations. These constraints affected both the flexibility and the effectiveness of some bias mitigation approaches, resulting in trade-offs between fairness and predictive performance.

5.3 In-processing Experiments

In this section, we summarize the results of the experiments conducted using **Adversarial Debiasing** to mitigate bias in heart attack risk prediction across two sensitive attributes: Sex and Hemisphere. The approach leverages an adversarial network to prevent the classifier from encoding information about protected attributes while maintaining predictive accuracy. Four configurations were tested for each attribute: a baseline with no debiasing (λ=0), light debiasing (λ=0.1), moderate debiasing (λ=0.5), and strong debiasing (λ=1.0).

- *Sex as Sensitive Attribute*: The baseline model exhibits substantial bias, with a low Disparate Impact (DI = 0.5896) and noticeable disparities in Equality of Opportunity (0.0446). Positive predictions are slightly higher for females (5.48%) compared to males (3.23%). Light debiasing (λ=0.1) greatly improves fairness metrics, reducing Statistical Parity Difference by 74% and Equality of Opportunity to near zero, with minimal impact on accuracy. Moderate debiasing (λ=0.5) maintains fairness improvements and slightly increases accuracy. Strong debiasing (λ=1.0) achieves the best overall balance, bringing DI within the fair range (0.8315), minimizing disparities across all metrics, and preserving predictive performance.

- *Hemisphere as Sensitive Attribute*: For Hemisphere, the baseline model is already relatively fair (DI = 0.8530, Equality of Opportunity = 0.0050) with minimal differences in positive prediction rates between groups. Light debiasing (λ=0.1) further improves DI and reduces Statistical Parity Difference, while maintaining accuracy. Moderate debiasing (λ=0.5) tends to overcorrect, reducing DI below acceptable thresholds and introducing new bias, and is therefore not recommended. Strong debiasing (λ=1.0) achieves near-perfect fairness (DI = 0.9554, minimal SPD) with minimal accuracy loss, making it the preferred configuration for this attribute.

Overall, the in-processing experiments demonstrate that adversarial debiasing can effectively improve fairness with only minimal impact on predictive performance. Improvements are particularly pronounced for Sex, where baseline bias is more severe, while Hemisphere requires lighter adjustments due to inherently better baseline fairness. The experiments also highlight the importance of carefully selecting the λ parameter: smaller values provide moderate fairness improvements with low risk of overcorrection, whereas higher values (λ=1.0) can achieve optimal fairness, especially when baseline disparity is high.
Preliminary conclusions suggest that for Sex, λ=1.0 provides the best trade-off between fairness and accuracy, while for Hemisphere, both λ=0.1 and λ=1.0 offer effective debiasing. Future work may extend this analysis by exploring debiasing approaches that account for multiple sensitive attributes simultaneously, which was not addressed in this study due to current limitations of the adopted bias mitigation framework. Additionally, further validation could be conducted by incorporating a broader set of fairness metrics to assess model robustness across different subgroups and evaluation perspectives.

5.4 Post-processing Experiments

We evaluated the performance of these techniques on the Heart Attack Prediction dataset, extending the analysis to four protected attributes: *Sex*, *Diabetes*, *Previous Heart Problems*, and *Hemisphere*. This allowed us to assess bias across biological, clinical, and geographic dimensions. The evaluation focused on the trade-off between predictive performance (Recall, Accuracy) and fairness (Demographic Parity Disparity).

- **EqOdds Results**: This method successfully maximized general *Accuracy* and *Recall*. However, it exhibited the poorest performance in terms of fairness, particularly for the attribute *Sex*, where it yielded a high Demographic Parity disparity (0.047). This suggests that strictly enforcing equal error rates without calibration can inadvertently increase demographic inequality.

- **Calibrated EqOdds Results**: While this approach achieved the lowest disparity scores (high fairness) across all attributes—including geographic location (*Hemisphere*)—and maintained good accuracy, it severely penalized *Recall*. In a medical screening context, a low Recall leads to an increase in false negatives, which is clinically unacceptable.

- **Reject Option Classification Results**: This technique emerged as the most balanced solution. It consistently improved *Recall* and F1-scores compared to the other methods, while keeping fairness disparity within acceptable limits (DP $<$ 0.016) for all attributes, including *Hemisphere*. By resolving uncertain cases in favor of the unprivileged groups (e.g., patients from the Southern Hemisphere or with prior conditions), it reduced the risk of missed diagnoses.

Table 3: Comparison of performance (Recall) and fairness (Demographic Parity Disparity) across bias mitigation methods. The **Reject Option** achieves the best trade-off, significantly boosting Recall compared to the baseline while maintaining low disparity.

Method	**Sex (Biological)**		**Hemisphere (Geographic)**	
	Recall ↑	**Disparity (DP) ↓**	**Recall ↑**	**Disparity (DP) ↓**
Original Model	0.0042	0.0033	0.0042	0.0092
EqOdds Postprocessing	1.0000	0.0470	1.0000	0.0246
Calibrated EqOdds	0.0042	0.0033	0.0042	0.0092
Reject Option	**0.9448**	**0.0088**	**0.4713**	**0.0120**

The experimental analysis with post-processing tecniques demonstrates that there is no ”one-size-fits-all” solution for bias mitigation. While *Calibrated Equalized Odds* offers the strictest mathematical fairness, it is unsuitable for this medical application due to the reduction in sensitivity. Conversely, **Reject Option Classification** proves to be the optimal strategy for this case study: it aligns with the clinical priority of minimizing false negatives (high Recall) while ensuring that the diagnostic process remains equitable across different demographic and geographic groups.

6 Conclusion

Addressing bias in healthcare ML algorithms is essential to ensure fairness and improve healthcare outcomes. By employing a combination of pre-processing, in-processing, and post-processing techniques, developers can work towards mitigating the impact of bias and promoting fairness in ML-driven healthcare applications. Tools like PyCaret and AIF360 provide valuable resources to facilitate this process and empower healthcare practitioners to build more equitable and trustworthy ML systems. As ML continues to play an increasingly significant role in healthcare, prioritizing bias mitigation is not only an ethical imperative but also crucial to ensure that these technological advancements are beneficial for all patients.

References

[1] Ko, D.T.; Sivaswamy, A.; Sud, M.; Kotrri, G.; Azizi, P.; Koh, M.; Austin, P.C.; Lee, D.S.; Roifman, I.; Thanassoulis, G.; Tu, K.; Udell, J.A.; Wijeysundera, H.C.; Anderson, T.J. Calibration and discrimination of the Framingham Risk Score and the Pooled Cohort Equations. *CMAJ* **2020**, *192*, E442–E449.

[2] Banerjee, S. Heart Attack Prediction Dataset. Kaggle. *Available online*: `https://www.kaggle.com/datasets/iamsouravbanerjee/heart-attack-prediction-dataset`.

[3] Ganapathi, S.; Palmer, J.; Alderman, J.E. et al. Tackling bias in AI health datasets through the STANDING Together initiative. *Nat Med* **2022**, *28*, 2232–2233.

[4] Paladino, L.M.; Hughes, A.; Perera, A.; Topsakal, O.; Akinci, T.C. Evaluating the Performance of Automated Machine Learning (AutoML) Tools for Heart Disease Diagnosis and Prediction. *AI* **2023**, *4*, 1036–1058.

[5] Hort, M.; Chen, Z.; Zhang, J.M.; Harman, M.; Sarro, F. Bias Mitigation for Machine Learning Classifiers: A Comprehensive Survey. arXiv 2022, abs/2207.07068.