

# BIAS DETECTION TOOLS FOR CLINICAL DECISION MAKING



## Supporting Documentation (Team [Super2021](#))



<b>Team Name</b>	Super2021
<b>Solution Name</b>	BeFair (A Multi-Level-Reweighting Method to Mitigate Bias)
<b>Link to Video</b>	<a href="https://youtu.be/xWZ22PICMbc">https://youtu.be/xWZ22PICMbc</a>
<b>Link to Code</b>	<a href="https://github.com/Anjinkun/NIH-Bias-Detection">https://github.com/Anjinkun/NIH-Bias-Detection</a>
<b>Link to GitHub Page</b>	<a href="https://yhzhu99.github.io/BeFair">https://yhzhu99.github.io/BeFair</a>
<b>List of team members</b>	Yinghao Zhu, Jingkun An, Enshen Zhou, Hao Li, Haoran Feng
<b>Contact information</b>	Email: <a href="mailto:zhuyinghao@buaa.edu.cn">zhuyinghao@buaa.edu.cn</a> Phone: +86 15026559349







### Abstract

We introduce an algorithm for detecting and mitigating bias in machine learning models. We propose an adaptive threshold algorithm to find the point of highest accuracy balance for the model's predicted labels, while also calculating three bias detection metrics: Statistical Parity Difference, Average Odds Difference, and Equal Opportunity Difference. We also improved the existing Reweighting method to support the superposition of multiple protected attributes, and our Multi-Level-Reweighting method significantly improves fairness in the case of multiple protected attributes. We evaluated our methods and found that they perform as well as, or better than, existing methods. We also tested our bias mitigation method by using it to train logistic regression and MLP models on bias-mitigated clinical datasets, finding that it successfully reduces bias while maintaining accuracy. Overall, our methods provide a practical and effective way for clinicians to detect and mitigate bias in their clinical decision decision scenarios.

### GitHub Code

Link: <https://github.com/Anjinkun/NIH-Bias-Detection>

The repository contains required `measure_disparity.py`, `mitigate_disparity.py` and Jupyter notebooks `example_{dataset}.ipynb` that tells how to use our proposed methods (correspond to above two Python file) on two datasets.

 <code>example_adult.ipynb</code>	Add files via upload	4 hours ago
 <code>example_meps.ipynb</code>	Add files via upload	4 hours ago
 <code>measure_disparity.py</code>	chore: format python scripts	7 hours ago
 <code>mitigate_disparity.py</code>	chore: format python scripts	7 hours ago
 <code>readme.txt</code>	Update readme.txt	4 hours ago
 <code>requirements.txt</code>	Update requirements.txt	7 hours ago

#### readme.txt



```
# NIH-Bias-Detection

- Team: Super2021
- Members: Yinghao Zhu, Jingkun An, Enshen Zhou, Hao Li, Haoran Feng

## Usage

- `measure_disparity.py`: detect and evaluate the bias (prediction logits dependent)
- `mitigate_disparity.py`: mitigate bias
- `example_adult.ipynb`: an example tutorial that measures and mitigates disparity on adult census income dataset (AdultDataset)
- `example_meps.ipynb`: an example tutorial that measures and mitigates disparity on clinical dataset (MEPSDataset19)

## Environment Setup

Linux/Windows/MacOS with Python version >= 3.8

(We've tested on Ubuntu 18 and Debian 11)

(Optional) Create a virtual environment with conda

- Install with pip
...
conda create -n befair python=3.9
conda activate befair
pip install -r requirements.txt
...

Please make sure you have correctly installed aif360 package. If not, please install it manually. (AIF360 GitHub repository reference: https://github.com/Trusted-AI/AIF360)
```

Figure 1. GitHub code repository overview.

```
In [5]: print("The test of measure_disparity ends now!")
```

The test of measure\_disparity ends now!

```
In [6]: from aif360.datasets import AdultDataset, MEPSDataset19
from mitigate_disparity import MultiLevelReweighing as Reweighing, BiasRemoverModel
from aif360.metrics import BinaryLabelDatasetMetric
print("The test of mitigate_disparity starts now!")

dataset = MEPSDataset19()
multi_privileged_groups = [
    {"feature_name": "RACE", "privileged_value": 1, "level": 1},
]
multi_unprivileged_groups = [
    {"feature_name": "RACE", "unprivileged_value": 0, "level": 1},
]

privileged_groups2 = [{"RACE": 1}]
unprivileged_groups2 = [{"RACE": 0}]

rw = Reweighing(multi_unprivileged_groups, multi_privileged_groups)
trans_adult_dataset = rw.fit_transform(dataset)

metric_orig_adult = BinaryLabelDatasetMetric(
    dataset,
    unprivileged_groups=unprivileged_groups2,
    privileged_groups=privileged_groups2,
)
print('before reweighing ,race disparate impact is '+str(metric_orig_adult.disparate_impact()))
metric_trans_adult = BinaryLabelDatasetMetric(
    trans_adult_dataset,
    unprivileged_groups=unprivileged_groups2,
    privileged_groups=privileged_groups2,
)
print('after reweighing ,race disparate impact is '+str(metric_trans_adult.disparate_impact()))
brm_model = BiasRemoverModel()
brm_model.fit(dataset)
predic_prob = brm_model.predic_prob(dataset.features)
print('the probability of prediction is')
print(predic_prob)
print("The test of mitigate_disparity ends now!")
```

```
The test of mitigate_disparity starts now!
before reweighing ,race disparate impact is 0.49826823461176517
after reweighing ,race disparate impact is 0.9999999999999999
the probability of prediction is
[[0.38951022 0.61048978]
 [0.11901319 0.88098681]
 [0.93920721 0.06079279]
 ...
 [0.96191023 0.03808977]
 [0.70094193 0.29905807]
 [0.66722456 0.33277544]]
The test of mitigate_disparity ends now!
```

Figure 2. The tutorial Jupyter notebook run on clinical dataset (MEPS).

## Methodology Overview

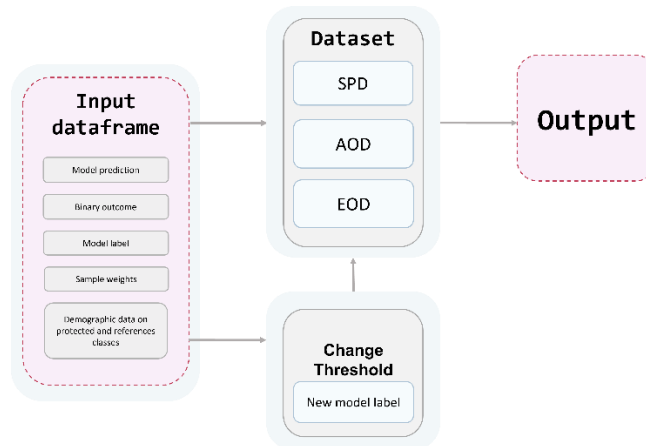


Figure 3. Bias detection evaluation metrics and model's inputs & outputs data flow.

We used three metrics for bias detection: Statistical Parity Difference, Average Odds Difference, and Equal Opportunity Difference. We chose these metrics because our research indicated that they are widely used in previous studies and have a significant impact on detecting bias in data. These three metrics provide guidance for detecting bias in data.

- $SPD = \text{Prob}\{\hat{Y} = 1|A = 0\} - \text{Prob}\{\hat{Y} = 1|A = 1\}$
- $AOD = \frac{1}{2} (|\text{Prob}\{\hat{Y} = 1|A = 0, Y = 0\} - \text{Prob}\{\hat{Y} = 1|A = 1, Y = 0\}| + |\text{Prob}\{\hat{Y} = 1|A = 0, Y = 1\} - \text{Prob}\{\hat{Y} = 1|A = 1, Y = 1\}|)$
- $EOD = \text{Prob}\{\hat{Y} = 1|A = 0, Y = 1\} - \text{Prob}\{\hat{Y} = 1|A = 1, Y = 1\}$

Where  $A$  represents the protected attribute  $\hat{Y}$  represents the predicted label, and  $Y$  represents the true label.

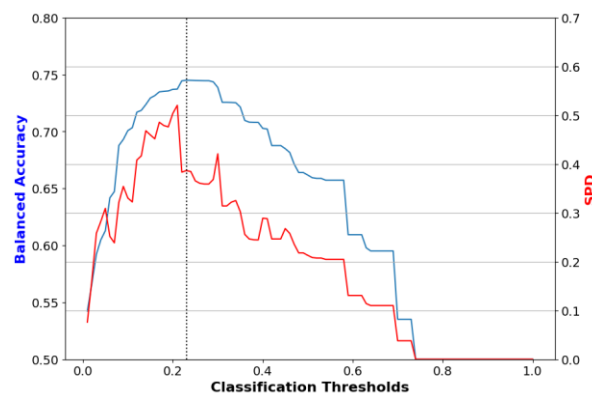


Figure 4. Adaptive thresholding strategy based on balanced accuracy metric.

In the bias detection process, we proposed an algorithm for adaptive thresholding because we had the model's predicted label probability for each sample. When the threshold is different, the model's predicted label may also change. Our adaptive threshold can find the point with the highest accuracy balance and calculate the three bias detection metrics at this point. Additionally, we can plot a graph showing how the balance of accuracy and bias detection metrics changes with the threshold.

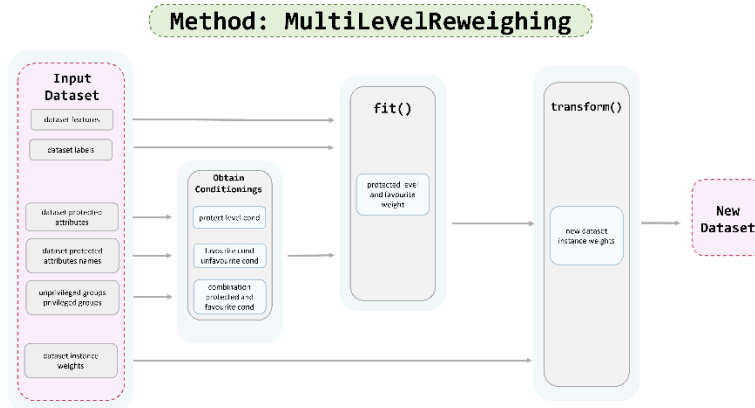


Figure 5. Our proposed optimized Multi-Level-Reweighing bias mitigation method.

In the bias mitigation process, we improved the existing Reweighing method to support the superposition of multiple protected attributes. Samples with multiple protected attributes now have a protected level. Users can also set weights for different protected attributes based on their own experience, and these weights should be integers. Based on the different protected levels, we will reset the weights for the samples.

Here's an introduction to the performance of our Multi-Level-Reweighing method: First, on a single attribute, our performance is exactly the same as the original Reweighing method. When multiple protected attributes are set in the settings, our method can significantly improve fairness, as these metrics are all closer to 1 than before.

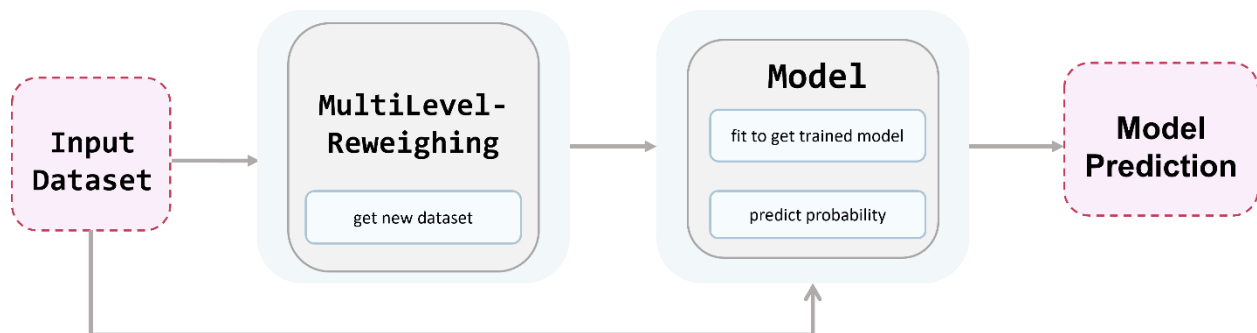


Figure 6. Make bias mitigated predictions with transformed dataset pipeline.

Our bias mitigation method contains 3 steps. First, we perform Multi-Level-Reweighing on the input dataset to obtain a bias-mitigated dataset. We then use this dataset to train our model, which can be either a logistic regression model or an MLP model. Once we have a trained model, we can use it to make bias mitigated predictions.

## Value Proposition

During the bias detection process, my bias detection method identified statistical parity difference (SPD), average odds difference (AOD), and equal opportunity difference (EOD). We used statistical methods to identify these biases, which allowed us to sensitively detect biases in the dataset.

We use the modified reweighing method to mitigate bias in the dataset based on multiple protected attributes. The method calculates the protected class of each sample based on its protected attributes and assigns weights to each sample accordingly. We believe that this approach is closer to real-world scenarios.

However, since we are performing preprocessing, we can not identify bias in the model itself. To identify bias in the algorithmic process, we need to perform in-processing.

Healthcare Scenario

## Healthcare Scenario

For potential biases, I believe we can detect them beyond the protected attributes, because the values of protected attributes may affect other attributes of the sample, which is a potential source of bias. However, performing this task is not easy. First, we cannot determine which common attributes are affected by the protected attributes. To identify these attributes, we may need to perform permutations and combinations of the common attributes. Then, we need to determine how bias is measured when two attributes are combined.

Over time, our bias detection tools can help healthcare professionals make fairer decisions, and these decisions can be used as data to train new models again. This cycle can continue to improve healthcare for all patient groups.

## Operational Requirements

Our tool can be deployed on a large scale and is delivered as a code file. There are no dependencies on vendors or proprietary information exchange standards that would limit the tool's usage.

There are not any dependencies on vendors or proprietary information exchange standards that would limit the tool's impact.

We first apply our improved Reweighting method to mitigate bias in the dataset. Then, we train a logistic regression model on the debiased data, which has some ability to reduce bias. The trained model can be used to make predictions on similar data features as the dataset, and the resulting bias metrics will also have some reduction.

We acknowledge that our developed tool has been distributed under the BSD 3 license.

## Sustainability Plan

For the operation of our tool, we need a technician to use the information provided by the information personnel to retrain the models in the system for better results; we need an administrator to manage and maintain the system, prevent possible information leaks, and handle unexpected issues that may arise due to device failure during system operation; we need information personnel to convert the case data provided by doctors into the data format required by the system and provide feedback to the technician for model retraining; finally, we need a large number of doctors to accurately record patient case information and provide feedback to the information personnel at regular intervals.

For our tool, the level of resources required to keep it running continuously is relatively low. First of all, it is essentially a Python file that only needs to be updated with the latest case data periodically to update the model. According to theoretical analysis, if we continue to practice using our tool and update the model with the latest case data, then the accuracy of our tool will gradually improve and tend towards a balance. Therefore, retrospective analysis is needed after each update to determine if progress is being made in the right direction.

In summary, our tool involves and considers a multidisciplinary team and specialized knowledge required for implementation in the real world. It can accurately calculate retrospective and prospective aspects and can be used in the future with almost zero maintenance costs.

## **Generalizability Plan**

The tool we are studying has significant positive implications for addressing unfairness in the medical field's concentrated datasets. It can be used by medical professional organizations, research institutions, or academic institutions. Since our improvement method is actually a Python file, our target user group members are required to have some Python knowledge.

Our tool is actually a Python file, so it can support large-scale and wide deployment. It does not have dependent vendors or proprietary information exchange standards to restrict the use of our tool. As long as we provide the corresponding data set with protected attributes in the appropriate data format, we can predict data from other clinical disciplines such as cardiology, oncology, obstetrics, etc.

The data set currently used by the tool only has two protected attributes. In the next step, we can provide data sets related to more protected attributes to enhance the predictive ability of our model. The tool provides three popular evaluation metrics for bias, and machine learning experts in the team can add customized evaluation metrics according to the paradigm to make the tool more suitable for the target needs. At the same time, our support for time-related predictions is limited and may require additional tool support.

This tool can only predict bias under the specified protected attributes, and its ability to detect the root cause of bias is relatively weak. The tool welcomes the addition of extensions to determine the potential root cause of the detected bias, such as adding a script to traverse dataset attributes to obtain possible sensitive attributes and determine the potential root cause of the bias.

The tool has made innovations based on the mature theoretical foundation in the industry while maintaining accuracy and greatly eliminating unfairness. It has a significant positive impact on improving medical fairness. Since we do not sacrifice accuracy too much in the prediction process, accurate prediction results can ensure people's trust in ML.

The tool requires information personnel to convert case data given by doctors into the data format required by the system and provide feedback to technical personnel for model retraining. As long as a data set that meets the data format is provided, the tool can be applied to prediction, diagnosis, and treatment recommendation environments and widely used in fields such as cardiology, oncology, obstetrics, etc.

## Implementation Requirements

To promote the implementation of the method we have studied in various application scenarios, we need to develop an implementation strategy and conduct a quantitative evaluation. When developing the implementation strategy, we need to meet the following three requirements: 1. Establish a reasonable interdisciplinary team and clarify the professional knowledge that different team members need to master. 2. Clarify the deployment conditions of the model. 3. Identify the human resources required for the system.

### Implementation Strategy

Our research method can effectively solve the unfairness problem in medical data sets and has research value, application value, and universal applicability. It can be used by medical professional organizations, research institutions, or academic institutions. In order to better promote our research methods, we have developed the following implementation strategy and standard operating procedures for organizations that may apply this method.

1. Establish a multidisciplinary team: We need to establish a multidisciplinary team consisting of data scientists, medical professionals, information personnel, administrators, and project managers or coordinators. This team will work collaboratively to ensure that the implementation and application of the tool in the practical environment can fully consider the professional knowledge of different fields. Team members should have the following skills and expertise:

- Data scientists or machine learning experts: with experience in implementing and optimizing deep learning algorithms. Responsible for configuring the code running environment, executing the code, and maintaining the code.
- Medical professionals: familiar with medical field knowledge and experience to evaluate the effectiveness and safety of the tool.
- Project manager or coordinator: with project management experience and understanding of the medical field and data science.
- Information personnel: converting case data provided by doctors into the data format required by the system and providing feedback to technical personnel for model retraining.
- Administrator: responsible for system management and maintenance, preventing possible information leakage and problems that may occur during system operation due to equipment failure.

The above analysis of the multidisciplinary team resolves the first requirement.

2. Determine human resource arrangements: Since the method we improved is essentially a Python file, the scale of the interdisciplinary team we established can be small, consisting only of a data scientist or machine learning expert, a medical professional, an information specialist, an administrator, and a project manager or coordinator. Therefore, for the promotion of this method, we only need five researchers from different fields, and only one researcher is needed for each field. So, we have a significant advantage in terms of human resource consumption. The above analysis of human resource arrangements solves the third requirement.



3. Develop project plans and schedules: We need the project manager to develop a project plan and schedule to ensure that the implementation and application of the project can be completed on time. This will include the following steps:

- Define project objectives and scope.
- Develop a detailed project plan and schedule.
- Identify project risks and challenges and take appropriate measures to address these issues.
- Develop a project communication plan to ensure communication and collaboration among team members.

4. Implementation and application of the method: We need a data scientist or machine learning expert to successfully implement the method we studied. This will include the following steps:

- Install the required software and libraries.
- Run the Python file and evaluate the tool's results and performance. Since the method we studied is essentially a Python file, it can support large-scale and widely deployed use, and it does not have dependency suppliers and proprietary information exchange standards to limit the use of our tools.
- Make necessary adjustments and improvements and rerun the tool.

5. Evaluate the results and performance of the tool: We need medical professionals to evaluate the results and performance of the tool to ensure that its effectiveness and safety are fully considered. This will include the following steps:

- Define evaluation metrics and standards.
- Perform model evaluation and validation.
- Analyze the results and make necessary improvements and adjustments.

6. Update and maintain the model based on information: Information personnel will convert the increasing amount of hospital medical data into the data required by our model to continuously update it, and management personnel will manage and maintain the system in daily use.

The above is the specific implementation strategy for promoting the method we have studied, which only requires a simple team structure, minimal human resources consumption, simple time planning, and evaluation metrics to complete the application of the method we have studied.

The second point in the implementation and application of the method above solves the third requirement.

### **Evaluation criteria**

In order to evaluate the success of the dissemination of our method, we have developed the following evaluation criteria:

- Degree of project goal achievement: The most important indicator of the success of implementation is whether the project goals have been achieved. Therefore, it is

necessary to clearly define the project goals when the project manager develops the project plan and schedule, ensuring that the entire project is completed within the scheduled time and achieves the expected results.

- **Time and cost control:** Another important indicator is the time and cost of project implementation. It is necessary to monitor the progress and budget of the project, and take timely measures to ensure that the project is completed within the specified time and budget range. If research organizations or professional institutions can successfully use our method to solve the unfairness of medical data sets within the specified time and budget, it indicates that we have well-controlled the time and cost of method promotion and implementation.
- **Project quality:** Project quality is one of the important indicators to measure the success of implementation. It is necessary to ensure that professional organizations, research institutions, and medical institutions using the method meet the expected quality standards, and ensure their reliability, accuracy and stability through testing and validation.
- **Customer satisfaction:** Customer satisfaction is another important indicator to measure the success of implementation. It is necessary to understand the degree of customer satisfaction through feedback, surveys and evaluations of the use of the method we provide, in order to continuously improve and optimize.

## **Lessons Learned**

During this competition, our biggest mistake was focusing too much on the study of the code during the preliminary research phase, and not paying enough attention to reading the papers of our predecessors in detail. As a result, our understanding of many issues was not profound enough, and was more superficial. After studying the papers of our predecessors, I also understood that the performance of bias detection metrics and machine learning models is negatively correlated in most cases, and how to combine these two metrics for evaluation. This competition was a valuable experience that gave us a deeper understanding of scientific research.