Highlight Note

Mitigating Unfairness

A common approach to mitigation is to use one of the algorithms and constraints to train multiple models, and then compare their performance, selection rate, and disparity metrics to find the optimal model for your needs. Often, the choice of model involves a trade-off between raw predictive performance and fairness - based on your definition of fairness for a given scenario. Generally, fairness is measured by reduction in disparity of feature selection (for example, ensuring that an equal proportion of members from each gender group is approved for a bank loan) or by a reduction in disparity of performance metric (for example, ensuring that a model is equally accurate at identifying repayers and defaulters in each age group).

To train the models for comparison, you use mitigation algorithms to create alternative models that apply *parity constraints* to produce comparable metrics across sensitive feature groups. The following table describes some common algorithms used to optimize models for fairness.

Technique	Description	Model type support
Exponentiated Gradient	A *reduction* technique that applies a cost-minimization approach to learning the optimal trade-off of overall predictive performance and fairness disparity	Binary classification and regression
Grid Search	A simplified version of the Exponentiated Gradient algorithm that works efficiently with small numbers of constraints	Binary classification and regression
Threshold Optimizer	A *post-processing* technique that applies a constraint to an existing classifier, transforming the prediction as appropriate	Binary classification

The choice of parity constraint depends on the technique being used and the specific fairness criteria you want to apply. Constraints include:

- **Demographic parity**: Use this constraint with any of the mitigation algorithms to minimize disparity in the selection rate across sensitive feature groups. For example, in a binary classification scenario, this constraint tries to ensure that an equal number of positive predictions are made in each group.
- **True positive rate parity**: Use this constraint with any of the mitigation algorithms to minimize disparity in *true positive rate* across sensitive feature groups. For example, in a binary classification scenario, this constraint tries to ensure that each group contains a comparable ratio of true positive predictions.
- **False positive rate parity**: Use this constraint with any of the mitigation algorithms to minimize disparity in *false positive rate* across sensitive feature groups. For example, in a binary classification scenario, this constraint tries to ensure that each group contains a comparable ratio of false positive predictions.
- **Equalized odds**: Use this constraint with any of the mitigation algorithms to minimize disparity in combined *true positive rate* and *false positive rate* across sensitive feature groups. For example, in a binary classification scenario, this constraint tries to ensure that each group contains a comparable ratio of true positive and false positive predictions.

- **Error rate parity**: Use this constraint with any of the reduction-based mitigation algorithms (**Exponentiated Gradient** and **Grid Search**) to ensure that the error for each sensitive feature group does not deviate from the overall error rate by more than a specified amount.
- **Bounded group loss**: Use this constraint with any of the reduction-based mitigation algorithms to restrict the loss for each sensitive feature group in a *regression* model.