Fairness of decision algorithm in machine learning

Alexis Janin, Benoît Müller, Florence Osmont

1 Introduction

Machine learning (ML) algorithms are used in a wide range of applications, including domains where decisions directly impact individuals. These algorithms are often designed to optimize some cost, but when it comes to human beings, the question of *fairness* must also be considered. One such example is bank loans where banks want to maximize their profit when making loans, which can have major impact on people lives and are therefore shouldn't be discriminated on some attributes. ML models are prone to unfairness because they can be complex objects, and it is not always clear how they decide.

For a *protected attribute*, a naive method would be to ignore it at all time. However, ML algorithm might recover it due to some correlations with other attributes. It must then be known and a notion of fairness can be defined according to it. Different methods have been taken into account for this notion. *Demographic parity* for instance requires independence between the protected attribute and the decision. However, this is very restrictive, the true output variable could actually depend on the attribute and hence would not even be fair.

During this report, we first present a definition of fairness for decision algorithm in ML, in which cases it is applicable and what it means in real case applications. Secondly, we present how to achieve such a fairness definition in the different cases described. Lastly, we present the results of this method on a real data set about credit card default prediction in order to have an example.

2 The fairness framework

We first introduce several notations and definitions. Let X be the features vector, Y_{true} the binary outcome that we would like to predict from X, and a predictor function $f(X) = Y_{\text{pred}}$. In some cases, depending on the ML algorithm used, we can also have access to a score R computed from X, from which a prediction function is derived in combination with a threshold t: $f(X) = Y_{\text{pred}} = \mathbb{1}_{R>t}$. We then call Y_{pred} the prediction and omit the function f. Included in the regular features X, we introduce A, the protected attribute, which is a **discrete** feature we don't want to discriminate. We present below the fairness definitions with respect to A, taken from [2], that are used in this project: equalized odds (2) and equal opportunity (1).

Definitions (fairness)

A prediction Y_{pred} is considered as fair in the **equal opportunity** sense if it satisfies:

$$P(Y_{\text{pred}} = 1|A = i, Y_{\text{true}} = 1) = P(Y_{\text{pred}} = 1|A = j, Y_{\text{true}} = 1) \ \forall \ i, j \tag{1}$$

A prediction Y_{pred} is considered as fair in the **equalized odds** sense if it satisfies:

$$P(Y_{\text{pred}} = 1|A = i, Y_{\text{true}} = y) = P(Y_{\text{pred}} = 1|A = j, Y_{\text{true}} = y) \ \forall \ i, j \text{ and } \forall \ y \in \{0, 1\} \tag{2}$$

Where i, j can take all the values of the discrete feature A. For simplicity, we only consider $A \in \{0, 1\}$, but the generalization is straightforward.

For concrete cases, equal opportunity means that 2 individuals with a different protected attribute A but the same true value 1, have the same probability of being predicted 1.

Definition (derived prediction): We say that a prediction Y_{pred} is derived from a score R and a protected attribute A if it is the image of (R, A) by a certain function, that can possibly be randomized independently of X conditionally on (R, A).

3 Methodology

This method is done as a post-processing step on a prediction. We will first present the methodology in the simplified case, where we are first given a binary prediction. In a second time, we consider the more general case of an initial score prediction.

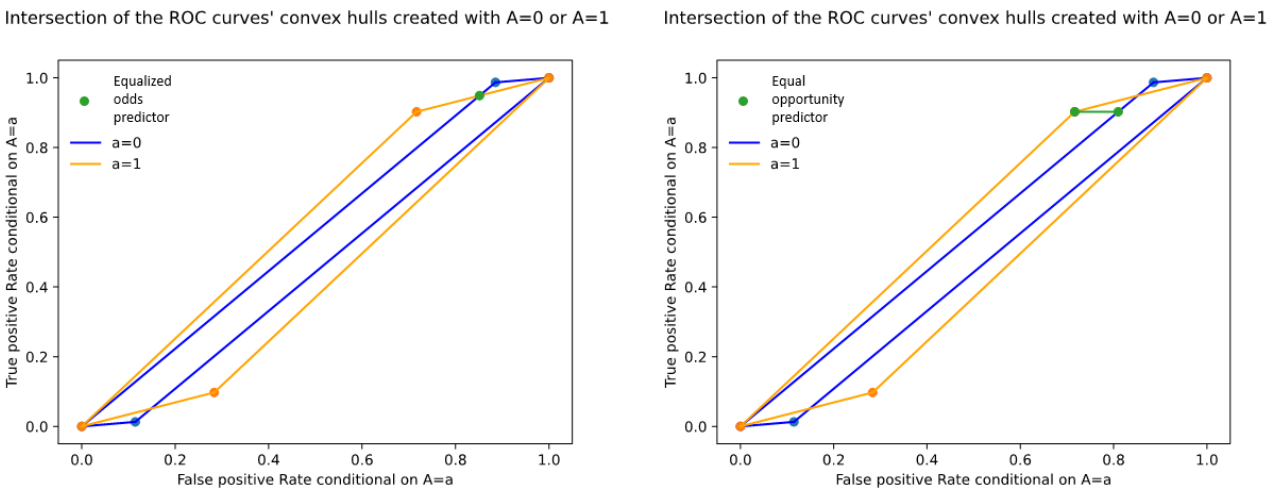


Figure 1: Convex hulls corresponding to a prediction conditioned on the protected attribute $A = a$ and points corresponding to a derived prediction using the two definitions of fairness given

3.1 With a binary prediction

Suppose that we are given a binary prediction Y_{pred} that can only take value in $\{0, 1\}$ and a binary protected feature A also taking values in $\{0, 1\}$. The definitions introduced in 2 have a direct geometric interpretation in the ROC space.

Lemma 3.1 *(Corresponding to Lemma 4.2 and 4.3 in [2])*
A prediction $\hat{Y}$ satisfies :

1. ***Equal opportunity*** *if and only if $\hat{Y}$ has the same vertical coordinate in the two ROC spaces conditioned on $A = 0, 1$.*

2. ***Equalized odds*** *if and only if $\hat{Y}$ has the same vertical and horizontal coordinates in the two ROC spaces, conditioned on $A = 0, 1$.*

Moreover, a prediction is derived from $\hat{Y}$ if and only if it is inside the convex hull formed by the coordinates of $\hat{Y}$ and $1 - \hat{Y}$ in the ROC space with $(0, 0)$ and $(1, 1)$.

This geometric interpretation of the definitions can be seen in the figures 1.

From this geometric definition, a new formulation of the problem as an optimization problem is derived. This optimization problem was proven to be a linear program in [2] thus it can be solved by any regular method like the simplex method. Solving it gives one point in the ROC space if we are looking for an equalized odd prediction, and two if we are just looking for equal opportunity.

The corresponding prediction can be found using a mixture of known ones. As we optimize a linear function, the points will always be the convex combination of only two predictions.

3.2 With a score prediction

We have developed the theory for the binary prediction and deduced how to compute a randomized fair prediction. We now adapt the method to do it for a continuous score R. In this case, instead of having only one prediction, for each choice of threshold t, there is an associated prediction $Y_t = \mathbb{1}\{R > t\}$.

Again, we consider the ROC curves conditioned on the value of A, in which we define the coordinates:

$$\gamma_{ay}(t) = \mathbb{P}(Y_t = 1 | A = a, Y_{\text{true}} = y),$$

and the curves are then given by $\gamma_a = (\gamma_{a0}, \gamma_{a1})$ for $a \in \{0, 1\}$. In practice, we only have access to a finite set $\{t_n\}_n$ of thresholds, but we can consider all the convex hull of the positions $\gamma_a(t_n)$, since we can reach any point in it by mixture randomization. Staying in the convex hull assures that the prediction will be derived from R and A. Now we look for the optimal target positions in those feasible spaces. We have two possible values for A, so two convex hulls, and we need to find a point in each one.

For equal opportunity, we need to have the same vertical coordinate for both points. We optimize according to a cost. This gives us two positions on the ROC space, associated with two thresholds, and hence two Y_a for $a \in \{0, 1\}$. We can construct the fair prediction as Y_A. In the equalized odds case, the two points must have the same position in the ROC space, so we have to look at the intersection of the two convex hulls. We optimize with respect to the chosen cost, we get one optimal point, and two associated Y_a for $a \in \{0, 1\}$ and construct the prediction the same way as equal opportunity. The Y_a used are derived from the target optimal values on the convex hulls. This can be done by randomization, as explained in the next section.

3.3 Algorithms used for estimation and optimization

The values of the coordinates of R and Y in the conditional ROC space are estimated empirically by counting the cases in the data.

For the binary case, we compute the optimal positions in the ROC space by solving a linear program with the CVXPY library ([1]). We look for two positions in the plane, with a reasonable number of conditions, so the algorithm runs very easily in a small amount of time.

For the score case, we evaluate $\gamma_a(t)$ at judicious values: only one time between two consecutive scores, since the value changes only when t passes the value of a score. We compute the optimal positions in the ROC space by transforming them into a one-dimensional optimization problem. Since decreasing the false positive rate is always better, we look only for points that are on the left limit of the convex hull, on a curve.

For equal opportunity, we look for points on the curves with the same height (vertical coordinate). We have to go through both curves, keeping the same height, and choose the optimal position. For equalized odds, we have only one point to look for, but it must be at the intersection of the two convex hulls. So for a fixed height, we have to look at the point on a curve that is furthest to the right.

In both cases, we see that we have to make a matching between the two parameterized curves, to link the same height positions. We decide to first find this reparametrization and then do the optimization. To do it, we go through the curves by iteratively incrementing the position of the one that has the smallest height. If the thresholds are well enough distributed, the heights are always similar. Using this reparametrization, we have a one-dimensional optimization problem with a fast objective function evaluation, so we use a grid search to find the minimum.

For equalized odds, there is a value of a for which we are not on the curve but only on the convex curve. We have to do a randomized mixture from predictions that are on the curve, using a convex combination. We use three points: the prediction that is the nearest, and the two deterministic predictions 1 and 0 (they always the positions (0,0) and (1,1)). Like so, the prediction that is used more often is the nearest, which is more likely to have good accuracy.

4 Experiment on default credit card prediction

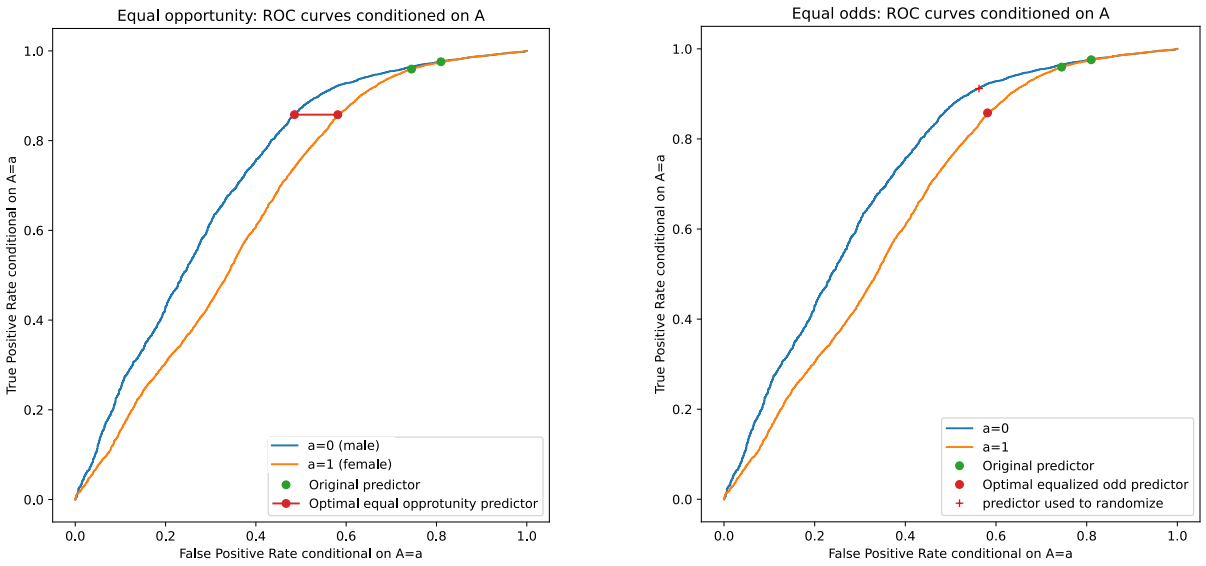
4.1 Data set and initial prediction

To show the effect of the method, we need an initial biased prediction and a protected attribute that can be discriminatory for this task. Taking this into consideration, we chose to work on a dataset concerning customer default payments that can be found at [4]. The dataset contains 24 attributes and 30000 instances. We modify it so that our label Y is 1 if a customer pay and 0 otherwise. We chose to take the gender as a protected attribute A taking value 0 for male and 1 for female. The dataset contains 60% of females. However, the classes are not balanced since there is only 22% of label 0. In order to have a more significant result between the original prediction and the "fair" one, we amplify the bias already present in the data. Taking the previous observation into consideration, the dataset was modified so that a woman who would have paid (original $Y = 1$) has her label flipped with 10% probability.

We create the first prediction by normalizing the feature vector and doing a simple logistic regression. This is done using the function implemented by scikit-learn ([3]). Notice that the threshold used by the function is 0.5. The ROC curve conditioned on A and the point corresponding to this prediction can be seen in Figure 2. We notice that males are more often predicted to pay than females.

4.2 Results

We present here the results of our experiment. In Figure 2a, we plot the conditioned curves of the score and show the positions of the original prediction and the new fair prediction satisfying equal opportunity.



(a) Conditional ROC curves for equal opportunity

(b) Conditional ROC curves for equalized odds

Figure 2: Comparison of predictions for the customer default payment dataset on the conditional ROC space for the score with protected attribute $A = a$

We see that we indeed have the same height for the two points and that they seem to have chosen a pair with a low cost, according to the gradient of the cost that points to the upper left of the graphic. The evolution of the true and false positive rates is displayed in Table 1. The values satisfy indeed the definitions.

We had an accuracy of 78%, which goes down to 75% for equal opportunity, and 74% for equalized odds. This is coherent with the fact that equalized odds is more restrictive than equal opportunity.

4.3 Interpretation and discussion

First, we notice that our results are coherent with the definition. Concretely, while the original predictor tends to predict that males will pay more often than females and thus be accorded more loans, the new predictor is fairer. Indeed, in the equal opportunity case, a person who should be accorded a loan will have the same chance to have it regardless of gender. In the equalized odd case, additionally, a person that shouldn't have one will have the same chance to have one regardless of gender.

However, we notice that our original score is not very good as the ROC curves are near the center of the space. Moreover, the threshold for it was chosen arbitrarily when it could have been optimized. Part of this can be explained by the dataset not being balanced and the label 1 being predicted very often.

5 Conclusion

The fairness methods presented in this report are reasonable concrete definitions that can be applied in many contexts such as for bank loan or credit card payment defaults without discriminating, for example, the sex or the race of the individuals. We presented experimental results for one of these examples.

This fairness definition has the computational advantage of being a pure post-processing method, which allows it to be used in various contexts. It also takes into account the optimization of the cost and encourages making a better prediction for the discriminated group. Finally, it has a clear interpretation in terms of ethics.

However, this method has some strong assumptions: the protected attribute A must be discrete and known for every sample. This limit the domain of application and clearly poses some privacy issues. Since the computation of the fair prediction sometimes requires randomness, the choice itself is sometimes arbitrary and it introduces unfairness within a group. Also, what should happen if we try to predict a single sample for a second time at another moment, should the result still be the same? The last drawback is that the theory of this framework doesn't ensure that our final loss will be close to the original one.

Appendix

Table 1: Coordinates on the gender conditioned ROC spaces, for various methods.

		True positive rate		False positive rate	
		Male	**Female**	**Male**	**Female**
	Unfair	0.8100	0.7448	0.8100	0.7448
Binary	**Equal opportunity**	0.7964	0.7448	0.7964	0.7448
	Equalized odds	0.8042	0.8042	0.8042	0.8042
Score	**Equal opportunity**	0.4856	0.5813	0.4856	0.5813
	Equalized odds	0.5813	0.5813	0.5813	0.5813

References

[1] Steven Diamond and Stephen Boyd. "CVXPY: A Python-embedded modeling language for convex optimization". In: *Journal of Machine Learning Research* 17.83 (2016), pp. 1–5.

[2] Moritz Hardt, Eric Price, and Nathan Srebro. "Equality of Opportunity in Supervised Learning". In: (2016). DOI: 10.48550/ARXIV.1610.02413. URL: https://arxiv.org/abs/1610.02413.

[3] F. Pedregosa et al. "Scikit-learn: Machine Learning in Python". In: *Journal of Machine Learning Research* 12 (2011), pp. 2825–2830.

[4] I-Cheng Yeh. *default of credit card clients Data Set*. 2016. URL: https://archive.ics.uci.edu/ml/datasets/default+of+credit+card+clients.