STAT 542: Multivariate Analysis **Spring 2021**

Lecture 13: Algorithmic fairness

Instructor: Yen-Chi Chen

Reference[1]:

[HPS2016] Hardt, M., Price, E., & Srebro, N. (2016). Equality of opportunity in supervised learning. In Advances in neural information processing systems (pp. 3315-3323).

13.1 Introduction

There is a growing popularity on the study of algorithmic fairness. While there are many ways to achieve/define algorithmic fairness, we focus on one particular approach via post-processing a (potentially unfair) predictor from an algorithm. In this scenario, the algorithmic fairness is an attempt to make a trained algorithm to be *fair* toward some *protected variables* such as ethnicity and gender.

We start with a simple example on binary classification. Suppose in our training data, we observe a label $Y \in \{0,1\}$, a covariate $X \in \mathbb{R}^p$, and a variable $A \in \{0,1\}$ that we wish to protect. So our training data can be viewed as IID random variables

$$(Y_1, X_1, A_1), \cdots, (Y_n, X_n, A_n).$$

We use the training data to construct a classifier $c = c(X, A) \in \{0,1\}$ that minimizes some loss function. Such training data is often obtained from observational studies.

The problem of this the usual classification method is that the training data may be biased toward some values of variable A. So if we consider the predicted value from the classifier $W_i = c(X_i, A_i)$, we may find that the protected variable A turns out to be an influential variable in W. While this is good for prediction, it may not be good if the variable A is a sensitive variable such as racial indicator and gender.

To see why this could be a problem, consider a hypothetical data from the US Customs & Border Protection. Each observation is a traveler's information on a particular trip. Let $Y = 1$ be an indicator of violating US immigration law and A be a foreigner indicator ($A = 1$ means a foreigner) and X is other features of this traveler. After training, our classifier c may place a lot of emphasis on the variable A since in the data, foreign traveler may have a higher chance of violating the immigration law. So we will observe a positive correlation between A, W even after we adjust for Y. However, the fact that being a foreigner itself will not encourage something to violate the immigration law. So such a classifier, although good for prediction, will cause discrimination toward foreigners. In this case, the classifier is said to be *unfair* to the foreigner indicator variable A.

The algorithmic fairness tries to resolve this issue. While there are may ways of defining algorithmic fairness, here we consider a simple one: a classifier c^* is said to be algorithmically fair to variable A if

$$A \perp c^*(X, A)|Y.$$

Namely, given the outcome variable Y, the classifier is independent of A, the variable we wish to protect.

[1] Most contents are following [HPS2016] but I use an alternative way to describe the generating process of a fair predictor Q. The predictor W corresponds to $\widehat{Y}$ of [HPS2016]; the predictor Q corresponds to $\tilde{Y}$ of [HPS2016].

Since any classifier c maps a feature X and the protected variable A into a binary value, we can use a binary random variable $W = c(X, A) \in \{0, 1\}$ as the predictor of Y. In fact, any classifier has a corresponding binary random variable that corresponds to it. So we say that a predictor W is *algorithmically fair* (also known as *equalized odds* in [HPS2016]) to A if

$$A \perp W|Y.$$

From now on, we will consider the case that the predictor W is given (but not necessarily fair) so that we do not need to use the information from X. So the training data we are using consists of IID binary triplets $(Y, A, W) \in \{0, 1\}^3$. This would greatly simplify the problem.

Such a predictor W is often a good one to predicting A but often is not fair toward variable A. So our goal is to create another predictor Q such that it is fair to A, i.e.,

$$A \perp Q|Y, \tag{13.1}$$

while maintaining a good predictive power of Y.

13.2 Constructing a fair predictor

To ensure that Q is a predictor, we need to be able to generate Q with A and W and the information from the training sample. Note that Q is often a random quantity. Namely, the distribution of Q should only depends on A and W but not Y. This is because if it is truly a predictor, it has to be computable for a new case with $(X_{\mathsf{new}}, A_{\mathsf{new}})$ or $(W_{\mathsf{new}}, A_{\mathsf{new}})$. Note that this construction implies that

$$Q \perp Y|A, W. \tag{13.2}$$

Since $Q, A, W \in \{0, 1\}$, the distribution of Q is determined by the following 4 parameters:

$$q_{a,w} = P(Q = 1|A = a, W = w), a, w \in \{0, 1\}.$$

To ensure fairness in equation (13.1), we need to impose some constraint of $q = (q_{00}, q_{01}, q_{10}, q_{11})$.

In the training sample, we observe (Y, A, W) so the joint distribution $p(y, a, w)$ is identifiable/oblivious (i.e., can be computed/estimated from the training data). Thus, we will treat $p(y, a, w)$ as a known quantity to simplify the problem.

The fairness constraint (algorithmically fair) in equation (13.1) is equivalent to

$$P(Q = 1|A = 0, Y = y) = P(Q = 1|A = 1, Y = y), y \in \{0, 1\}. \tag{13.3}$$

The above equation is also known as *equalized odds* in [HPS2016].

Now we derive the constraint over q from equation (13.3). Note that

$$\begin{aligned}
P(Q = 1|A = 0, Y = y) &= \sum_w P(Q = 1, W = w|A = 0, Y = y) \\
&= \sum_w P(Q = 1|W = w, A = 0, Y = y)P(W = w|A = 0, Y = y) \\
&\overset{(13.2)}{=} \sum_w P(Q = 1|W = w, A = 0)P(W = w|A = 0, Y = y) \\
&= q_{0,0}P(w|A = 0, y) + q_{1,0}P(1|A = 0, y).
\end{aligned}$$

As a result, equation (13.3) is equivalent to

$$q_{0,0}P(w|A=0,y)+q_{1,0}P(W=1|A=0,y)=q_{0,1}P(w|A=1,y)+q_{1,1}P(W=1|A=1,y). \tag{13.4}$$

The probability $P(W=w|A=a,Y=y)$ is identifiable from the training data so equation (13.4) is an identifiable constraint on q. We need this constraint for both $y=0$ and $y=1$.

As long as we choose q that satisfies equation (13.4), the resulting predictor Q is algorithmically fair to Y and is computable (derived) with A and W.

The parameter vector $q \in [0,1]^4$ and there are only 2 constraints in equation (13.4). To choose the optimal q, we will try to find the best parameter $q^* \in [0,1]$ such that

$$R(q)=\mathbb{E}[L(Q,Y);Q\sim q] \tag{13.5}$$

is minimized; the function $L(a,b)$ is the loss function of predicting b using a. Namely,

$$q^*=\underset{q\in[0,1]^4}{\mathsf{argmin}}\ R(q) \quad \text{subject to equation (13.4).} \tag{13.6}$$

Lemma 13.1 *The risk function in equation (13.5) is identifiable/estimatible from the training data.*

Proof: We will show that equation (13.5) is identifiable with any given parameter $q=(q_{0,0},q_{0,1},q_{1,0},q_{1,1})$.

Let q be fixed. Then

$$\begin{aligned}
R(q) &= \mathbb{E}[L(Q,Y);Q\sim q]\\
&= \sum_{s,y} L(s,y)P(Q=s,Y=y)\\
&= \sum_{s,y} L(s,y)P(Q=s,Y=y)\\
&= \sum_{s,y,a,w} L(s,y)P(Q=s,Y=y,A=a,W=w)\\
&= \sum_{s,y,a,w} L(s,y)P(Q=s|Y=y,A=a,W=w)P(Y=y,A=a,W=w)\\
&\overset{(13.2)}{=} \sum_{s,y,a,w} L(s,y)P(Q=s|A=a,W=w)P(Y=y,A=a,W=w)\\
&= \sum_{s,y,a,w} L(s,y)q_{a,w}^s(1-q_{a,w})^{1-s}P(Y=y,A=a,W=w).
\end{aligned}$$

Because $P(Y=y,A=a,W=w)$ is identifiable from the training sample and $q_{a,w}$ is specified, the above risk function is identifiable.

■

With Lemma 13.1, we know that the risk $R(q)$ is identifiable for any q from the training sample. So we can easily perform a minimization to find the optimal q^*. Also, the fairness constraint in equation (13.4) is a linear constraint so this minimization problem is easy to implement.

Parameters satisfying the constraint in equation (13.4) have an interesting property. For any binary random variable Z, we define the follow vector:

$$\gamma_a(Z)=(P(Z=1|A=a,Y=0),P(Z=1|A=a,Y=1)). \tag{13.7}$$

It turns out that $\gamma_a(W)$ and $\gamma_a(Q)$ are implicitly associated.

Lemma 13.2 *Consider a subset of* $[0,1]^2$ *as follows:*

$$P_a(W) = \text{Convex Hull}[(0,0), \gamma_a(W), \gamma_a(1-W), (1,1)].$$

Then we have

$$\gamma_a(Q) \in P_a(W)$$

for any Q *that is generated with parameter* q*. Moreover, when* $P(Z=1|A=a,Y=0) \neq P(Z=1|A=a,Y=1)$ *for* $a=0,1$*, every point* $\eta \in P_a(W)$ *has a unique* $q_\eta \in [0,1]^4$ *such that the corresponding random variable* $Q_\eta \sim q_\eta$*.*

Note: in Lemma 13.2, we do not require the parameter q to satisfy the fairness condition (13.4).

Proof:

Part 1: $\gamma_a(Q) \in P_a(W)$**.** Let $q \in [0,1]^4$ be any arbitrary parameter vector. Recall that $\gamma_a(Q) = (P(Q=1|A=a,Y=0), P(Q=1|A=a,Y=1))$. The first component

$$\begin{aligned}
\gamma_a(Q)_1 &= P(Q=1|A=a,Y=0)\\
&= P(Q=1,W=0|A=a,Y=0) + P(Q=1,W=1|A=a,Y=0)\\
&= P(Q=1|W=0,A=a)P(W=0|A=a,Y=0) + P(Q=1|W=1,A=a)P(W=1|A=a,Y=0)\\
&= q_{0,a}P(W=0|A=a,Y=0) + q_{1,a}P(W=1|A=a,Y=0)\\
&= q_{0,a}\cdot(1-\gamma_a(W)_1) + q_{1,a}\cdot\gamma_a(W)_1.
\end{aligned}$$

Similarly, the second component

$$\begin{aligned}
\gamma_a(Q)_2 &= P(Q=1|A=a,Y=1)\\
&= P(Q=1|W=0,A=a)P(W=0|A=a,Y=1) + P(Q=1|W=1,A=a)P(W=1|A=a,Y=1)\\
&= q_{0,a}P(W=0|A=a,Y=1) + q_{1,a}P(W=1|A=a,Y=1)\\
&= q_{0,a}\cdot(1-\gamma_a(W)_2) + q_{1,a}\cdot\gamma_a(W)_2.
\end{aligned}$$

Thus, one can easily see that when we vary $q_{0,a}, q_{1,a} \in [0,1]$, the resulting $\gamma_a(Q)$ will belong to the convex hull formed by the four points $(0,0), \gamma_a(W), \gamma_a(1-W), (1,1)$, which completes the first part of the assertion.

Part 2: uniqueness of q_η**.** To see the uniqueness, note that $\gamma_a(W)$ is identified from the training sample. So it can be viewed as a fixed quantity. Suppose $\gamma_a(Q)$ is given, then the expression of the two components become

$$\begin{aligned}
\gamma_a(Q)_1 &= q_{0,a}\cdot(1-\gamma_a(W)_1) + q_{1,a}\cdot\gamma_a(W)_1,\\
\gamma_a(Q)_2 &= q_{0,a}\cdot(1-\gamma_a(W)_2) + q_{1,a}\cdot\gamma_a(W)_2.
\end{aligned}$$

For each a, we have two parameters $q_{0,a}, q_{1,a}$ and two equations, which leads to a unique solution as long as $\gamma_a(W)_1 \neq \gamma_a(W)_2$ ($\Leftrightarrow P(Z=1|A=a,Y=0) \neq P(Z=1|A=a,Y=1)$), which completes the proof.

■

13.2.1 A summary of the generating process of a fair predictor Q

Here is a summary of how we generate Q with a given A, W and the training data.

1. Use the training sample to identify $p(y,a,w) = P(Y=y, A=a, W=w)$.

2. For any parameter $q \in [0,1]^4$, derive the constraint in equation (13.4).

3. Minimize $R(q) = \sum_{s,y,a,w} L(s,y) q_{a,w}^s (1-q_{a,w})^{1-s} P(Y=y, A=a, W=w)$ with the constraint (13.4) to obtain q^*.

4. Generate $Q|A, W \sim q^*(A, W)$ for each observation in the training sample or for a new observation with $A_{\mathsf{new}}, W_{\mathsf{new}}$.

The quantity Q generated from the above procedure satisfies the following properties:

- Q can be generated with only A, W (i.e., Q is a predictor).
- Q satisfies the condition $Q \perp A|Y$ (i.e., Q is algorithmically fair to A).
- Q minimizes the predictive risk with the above constraints.

13.2.2 An alternative optimization formulation

Thee procedure described in Section 13.2.1 is a direct generating procedure. In some literature such as [HPS2016], the procedure is often written in the following optimization form:

$$\min_Q \mathbb{E}[L(Q, Y)] \tag{13.8}$$

$$\text{s.t. } \gamma_a(Q) \in P_a(W), \tag{13.9}$$

$$\gamma_0(Q) = \gamma_1(Q). \tag{13.10}$$

The minimization refers to finding the random variable Q that minimizes the risk (and subject to the two constraints). The second constraint is associated with Lemma 13.2 and the third constraint is equivalent to the fairness constraint in equation (13.4).

13.3 Applying to other fairness principles

As mentioned at the beginning, there are multiple ways to define fairness. We are just using a simple one. A nice feature of the above procedure is that with a gentle modification, it can be applied to other fairness principles.

For instance, the *equal opportunity* principle requires a predictor Q to satisfy

$$P(Q=1|A=1, Y=1) = P(Q=1|A=0, Y=1). \tag{13.11}$$

Namely, we only require that $Q \perp A|Y = 1$ and does not constraint on the case of $Y = 0$.

Generating Q with constraint (13.11) is simple. We just replace equation (13.4) in step 2 and 3 of Section 13.2.1 with equation (13.11) and work out the implied constraints on the parameter q. Then the generated Q will satisfy the constraint of (13.11).

If we are using the procedure in Section 13.2.2, then we only need to replace equation (13.10) with $\gamma_0(Q)_2 = \gamma_1(Q)_2$ since this would correspond to the constraint in equation (13.11).

13.3.1 Example: test fairness

To illustrate how this idea can be applied to other fairness principles, consider the *test fairness* in the following paper

> [C2017] Chouldechova, A. (2017). Fair prediction with disparate impact: A study of bias in recidivism prediction instruments. Big data, 5(2), 153-163.

In our case, a predictor Q has test fairness if

$$P(Y=1|Q=s,A=0)=P(Y=1|Q=s,A=1) \tag{13.12}$$

for any $s=0,1$.

To see how equation (13.12) constraint the parameter vector q, we expand the first term and consider $s=1$:

$$\begin{aligned}
P(Y=1|Q=1,A=0) &= \frac{P(Y=1,Q=1|A=0)}{P(Q=1|A=0)}\\
&= \frac{\sum_w P(Y=1,Q=1,W=w|A=0)}{\sum_{w'} P(Q=1,W=w'|A=0)}\\
&\overset{(13.2)}{=} \frac{\sum_w P(Q=1|W=w,A=0)P(W=w,Y=1|A=0)}{\sum_{w'} P(Q=1|W=w',A=0)P(W=w'|A=0)}\\
&= \frac{\sum_w q_{w,0}P(W=w,Y=1|A=0)}{\sum_{w'} q_{w',0}P(W=w'|A=0)}.
\end{aligned}$$

The two probabilities $P(W=w,Y=1|A=0)$ and $P(W=w'|A=0)$ are identifiable from the data. A similar calculation shows that

$$P(Y=1|Q=1,A=1)=\frac{\sum_w q_{w,1}P(W=w,Y=1|A=1)}{\sum_{w'} q_{w',1}P(W=w'|A=1)}.$$

So the test fairness constraint in equation (13.12) requires

$$\frac{\sum_w q_{w,0}P(W=w,Y=1|A=0)}{\sum_{w'} q_{w',0}P(W=w'|A=0)}=\frac{\sum_w q_{w,1}P(W=w,Y=1|A=1)}{\sum_{w'} q_{w',1}P(W=w'|A=1)}. \tag{13.13}$$

Also, for the case of $s=0$, the above constraint becomes

$$\frac{\sum_w (1-q_{w,0})P(W=w,Y=1|A=0)}{\sum_{w'} (1-q_{w',0})P(W=w'|A=0)}=\frac{\sum_w (1-q_{w,1})P(W=w,Y=1|A=1)}{\sum_{w'} (1-q_{w',1})P(W=w'|A=1)}. \tag{13.14}$$

Thus, to generate Q that satisfies the test fairness, we need to choose the parameter vector q that satisfies equation (13.13) and (13.14). Namely, we use these two constraints to replace equation (13.4) in step 2 and 3 of Section 13.2.1.

13.4 Continuous predictor

Here we describe the case where instead of having a predictor $W\in\{0,1\}$, we have a continuous predictor $R\in[0,1]$ but we still have $Y,A\in\{0,1\}$. This is a special case that [HPS2016] studied. A common scenario that this would occur is the case of a generative classifier; the quantity R may refer to as the estimated probability of $Y = 1$ given the covariates X and A. But it can be more general that R is just some score that we use for our final decision.

For a continuous predictor R, it equalizes the odds (algorithmically fair) if

$$R \perp A|Y.$$

However, since R is a quantity that is obtained from the training sample, it generally does not equalizes the odds.

Since R is not binary, often our final decision is based on placing a threshold on R such that as $Q_t = I(R \geq t)$. Then this maps the problem into the problem we have faced before. Note that $P(Q_t = 1|A = a, Y = y) = P(R \geq t|A = a, Y = y)$.

One possible way to achieve algorithmic fairness, i.e., $Q_t \perp A|Y$, is to search for t^* such that $P(Q_{t^*} = 1|A = 0, Y = y) = P(Q_{t^*} = 1|A = 1, Y = y)$, namely,

$$P(R \geq t^*|A = 0, Y = y) = P(R \geq t^*|A = 0, Y = y).$$

If we can find such t^*, then Q_{t^*} is a fair predictor of Y.

Consider the ROC (receiver operating characteristic) curve:

$$C_a(t) = (P(R \geq t|A = a, y = 0), P(R \geq t|A = a, y = 1)) \in [0, 1]^2$$

and

$$C_a = \{C_a(t) : t \in [0, 1]\} \subset [0, 1]^2.$$

One can easily see that such t^* is the point where the two curves C_0 and C_1 intersects.

Although this is an attractive idea, it has two drawbacks. First, such t^* may not exist except for the two trivial cases: $t^* = 0, 1$. Second, even if t^* exists, the resulting fair predictor may not have a good prediction power (in terms of loss function) since it is often a single point (excluding $t^* = 0, 1$).

In [HPS2016], the authors proposed a very clever idea using the convex combination of any two points on an ROC curve. To see this idea, suppose we use a *random threshold* T_a such that

$$T_a = \begin{cases} \tau_{a,1}, & \text{with a probability of } \zeta_a \\ \tau_{a,2}, & \text{with a probability of } 1 - \zeta_a, \end{cases}$$

where T_a depends on the parameters $\tau = (\tau_{0,1}, \tau_{0,2}, \tau_{1,1}, \tau_{1,2})$ and $\zeta = (\zeta_0, \zeta_1)$. Let $Q_{\tau,\zeta} = I(R \geq T_A)$. Then one can easily see that

$$\begin{aligned} P(W_\tau = 1|A = a, Y = y) &= P(R \geq T_a|A = a, Y = y) \\ &= \zeta_a \cdot P(R \geq \tau_{a,1}|A = a, Y = y) + (1 - \zeta_a) \cdot P(R \geq \tau_{a,2}|A = a, Y = y), \end{aligned}$$

which is a convex combination of $C_a(\tau_{a,1})$ and $C_a(\tau_{a,2})$. Thus, by varying τ and ζ, we can change the vector

$$V_a(\tau, \zeta) = (P(Q_{\tau,\zeta} = 1|A = a, Y = 0), P(Q_{\tau,\zeta} = 1|A = a, Y = 1)) \in [0, 1]^2$$

to be anywhere within $D_a = \mathsf{Convex\ Hull}(C_a)$. Namely, $V_a(\tau, \zeta) \in D_a$.

Recall that the fairness constraint of $Q_{\tau,\zeta}$ is

$$P(Q_{\tau,\zeta} = 1|A = 0, Y = y) = P(Q_{\tau,\zeta} = 1|A = 1, Y = y), \quad y = 0, 1,$$

which require that the two vectors $V_0(\tau, \zeta) = V_1(\tau, \zeta)$. Since $V_0(\tau, \zeta) \in D_0$ and $V_1(\tau, \zeta) \in D_1$, the feasible region that the two vector agree is $D^* = D_0 \cap D_1$. Note that for a fixed point $v_0 \in D^*$, there might be multiple (τ, ζ) such that $V_0(\tau, \zeta) = V_1(\tau, \zeta) = v_0$, i.e., the choice of (τ, ζ) is not unique.

A nice property of such procedure is that any point in D^* is a feasible solution. So we can optimize the prediction power (minimizing the loss) within the region D^*. Formally, when a loss function L is given, we search for (τ^*, ζ^*) that solves the following minimization problem:

$$\begin{aligned} &\min_{\tau,\zeta} \mathbb{E}[L(Q_{\tau,\zeta}, Y)] \\ \text{s.t.} \quad &V_0(\tau, \zeta) = V_1(\tau, \zeta). \end{aligned}$$

The resulting predictor will have the following three properties:

- Q_{τ^*,ζ^*} is a predictor, i.e., it can be generated with the training sample and R, A.
- Q_{τ^*,ζ^*} equalizes the odds, i.e., $Q_{\tau^*,\zeta^*} \perp A|Y$.
- Q_{τ^*,ζ^*} minimizes the predictive risk with the above constraints.