

# Learning to predict vs. learning to decide

Floyd Kretschmar

December 12, 2019

## 1 Decision making as prediction

Many decision making task are being supplemented or fully automated by data-driven predictive models. The most prolific approach to integrate machine learning into such decision making tasks is to split decision making into two separate tasks: prediction and decision making based on the prediction. Within this framework data can then be utilized to train models that make prediction. Based on these predictions decisions are made according to a decision policy. There are many different decision problems that can be viewed under this lense, but this summary will focus on the case of binary classification with binary sensitive attributes as described in [?].

Therefore let  $\mathcal{X} \subseteq \mathbb{R}^d$  be the feature domain,  $\mathcal{S} = 0, 1$  the range of sensitive attributes and  $\mathcal{Y} = 0, 1$  the set of ground truth labels. It is assumed that for each individual  $\mathbf{x} \in \mathcal{X}$ ,  $s \in \mathcal{S}$  and  $y \in \mathcal{Y}$  are given by the joint probability distribution  $P(\mathbf{x}, s, y) = P(y | \mathbf{x}, s)P(\mathbf{x}, s)$  as defined by [?]. Within this framework a **prediction task** is defined as training a model  $Q(y | x, s; \theta)$  in a supervised mannor, which means solving the optimization problem

$$\begin{aligned} \underset{\theta}{\operatorname{argmin}} \quad & \mathcal{L}(\theta) \\ \text{s.t.} \quad & \mathcal{F}(\theta) \end{aligned}$$

where  $\mathcal{L}$  is the loss function and  $\mathcal{F}$  is a function measuring the fairness. This fairness function can be chosen in a multitude of ways and the core of this work will be about exploring different formulations of  $\mathcal{F}$ , their mathematical properties with regards to optimization as well as their comparative performance. The resulting model  $Q(y | x, s; \theta)$  is a probability distribution defining the probability of an individual being a member of specific class given both their features  $\mathbf{x}$  as well as their sensitive attribute  $s$ .

The second part of a decision making task is actually **making a decision based on a decision policy** and the predictions of  $Q(y | x, s; \theta)$ . As seen

in [?] such a policy can be defined as a mapping  $\pi : \mathcal{X} \times \mathcal{S} \rightarrow \mathcal{P}(0, 1)$  that maps an individual's feature vector and sensitive attribute to a probability distribution over binary decisions  $d \in \{0, 1\}$ . That means  $\pi(d | \mathbf{x}, s)$  is the probability distribution over the possible decisions, which in this case are binary, given the features and sensitive attributes of an individual. The simplest choice for a decision policy is a deterministic threshold policy of the form

$$\pi_Q(d = 1 | \mathbf{x}, s) = \mathbf{1}[Q(y = 1 | \mathbf{x}, s) \geq c]$$

where the policy  $\pi_Q$  makes a positive decision  $d = 1$  for all individuals for which the trained model  $Q$  exceeds a certain confidence threshold  $c$  that the individual is part of class  $y = 1$ . In this scenario we assume that  $Q(y = 1 | \mathbf{x}, s) \approx P(y = 1 | \mathbf{x}, s) - \delta_s$ , meaning that  $Q$  is approximately equal to the true ground truth probability distribution  $P(y | \mathbf{x}, s)$ . This approach directly incorporates fairness constraints by allowing group specific  $\delta_s$  and therefore indirectly allowing for different thresholds  $c$  w.r.t. group affiliation.

As discussed by [?] this approach has been shown by [?] to often lead to better performance than post-processing a potentially unfair predictor. But the same paper argues that due to the fact that  $Q(y = 1 | \mathbf{x}, s)$  and  $P(y = 1 | \mathbf{x}, s) - \delta_s$  are not exactly equal the resulting policy  $\pi_Q$  will usually be suboptimal.

## 2 Decision making under imperfect data

The methods described so far in section 1 were built assuming i.i.d. samples to optimize the parameters  $\theta$  of a given model  $Q$ . The authors of [?] explain that in many real-world applications this assumption is incorrect. In these applications the original data has not been sampled from the true ground truth distribution  $P(\mathbf{x}, s, y)$  but instead from a weighted distribution

$$P_{\pi_{Q_0}} = P(y | \mathbf{x}, s) \pi_0(d = 1 | \mathbf{x}, s) P(\mathbf{x}, s)$$

where  $\pi_0$  is some initial decision policy that is employed while collecting the data. The authors call this kind of ground truth distribution *induced* by  $\pi_0$ . In this scenario the decision whether or not  $y$   $P(y | \mathbf{x}, s)$  comes into existence is based on the decision generated by  $\pi_0$ . An example of this is a loan decision scenario. If a bank wants to train a machine learning model to make predictions on whether or not a potential client will default on their loan, they can only use historic data of past loans that they have given out. But this data is weighted by the historical decision rules based on which

---

the bank has made their decisions so far, since they only have data for the scenario where they have granted a loan in the first place.

The authors of [?] argue that in such a scenario “for error based learning algorithms under no fairness constraints, learning within deterministic threshold policies is guaranteed to fail.”. They instead propose to directly learn a probabilistic decision policy  $\pi_\theta$  that maximizes the utility, which can be formulated as the following optimization problem:

$$\begin{aligned} \underset{\theta}{\operatorname{argmax}} \quad & u(\pi_\theta) \\ \text{s.t.} \quad & \mathcal{F}(\pi_\theta) \end{aligned}$$

Where  $\mathcal{F}$  is again the same fairness measurement function described in 1 and  $u$  is the function that measures the utility of the learned decision policy that is parameterized by  $\theta$ . The authors propose a utility function of the form

$$\begin{aligned} u_P(\pi) &= \mathbb{E}_{\mathbf{x}, y, s \sim P(\mathbf{x}, s, y), d \sim \pi(d | \mathbf{x}, s)} [yd - cd] \\ &= \mathbb{E}_{\mathbf{x}, y, s \sim P(\mathbf{x}, s, y), d \sim \pi(d | \mathbf{x}, s)} [d(y - c)] \\ &= \mathbb{E}_{\mathbf{x}, s \sim P(\mathbf{x}, s)} [\pi(d = 1 | \mathbf{x}, s)(P(y = 1 | \mathbf{x}, s) - c)] \\ &= \int \pi(d = 1 | \mathbf{x}, s)(P(y = 1 | \mathbf{x}, s) - c)P(\mathbf{x}, s) d\mathbf{x} ds \end{aligned}$$

where  $c \in (0, 1)$  represents the cost considerations of the decision maker. The authors prove that the optimal decision policy  $\pi^*$  can be learned only from data generated by a ground truth distribution  $P_{\pi_0}$  that is induced by  $\pi_0$  if this initial policy is an *exploring policy*. This means, “the data collection distribution must not ignore regions where the true distribution puts mass” or more mathematically speaking: “ $\pi_0(d = 1 | \mathbf{x}, s) > 0$  must be true for any measurable subset of  $\mathcal{X} \times \mathcal{S}$  with positive probability under  $P$ ”. For an exploring policy  $\pi_0$  and any arbitrary policy  $\pi$  the utility is then calculated as

---

$$\begin{aligned}
u_P(\pi) &= \mathbb{E}_{\substack{x, y, s \sim P(\mathbf{x}, s, y) \\ d \sim \pi(d | \mathbf{x}, s)}} [d(y - c)] \\
&= \int d(y - c) \pi(d | \mathbf{x}, s) P(\mathbf{x}, s, y) dx ds \\
&= \int d(y - c) \pi(d | \mathbf{x}, s) P(y | \mathbf{x}, s) P(\mathbf{x}, s) dx ds \\
&= \int d(y - c) \pi(d | \mathbf{x}, s) P(y | \mathbf{x}, s) P(\mathbf{x}, s) \frac{\pi_0(d | x, s)}{\pi_0(d | x, s)} dx ds \\
&= \int d(y - c) \pi(d | \mathbf{x}, s) P(y | \mathbf{x}, s) \pi_0(d | x, s) P(\mathbf{x}, s) \frac{1}{\pi_0(d | x, s)} dx ds \\
&= \int d(y - c) \pi(d | \mathbf{x}, s) P_{\pi_0}(\mathbf{x}, s, y) \frac{1}{\pi_0(d | x, s)} dx ds \\
&= \int \frac{d(y - c)}{\pi_0(d | x, s)} \pi(d | \mathbf{x}, s) P_{\pi_0}(\mathbf{x}, s, y) dx ds \\
&= \mathbb{E}_{\substack{x, y, s \sim P_{\pi_0}(\mathbf{x}, s, y) \\ d \sim \pi(d | \mathbf{x}, s)}} \left[ \frac{d(y - c)}{\pi_0(d | x, s)} \right] = u_{P_{\pi_0}}(\pi, \pi_0)
\end{aligned}$$

which is the same as inverse propensity scoring. To formulate the optimization problem in an unconstrained way, the authors propose a combined utility function  $v(\pi)$  of the form

$$v(\pi) = u(\pi) - \frac{\lambda}{2} (\mathcal{F}(\pi))^2$$

which contains a penalty term that is dependent on  $\mathcal{F}$ . The gradient of the new unconstrained optimization problem then turns out to be the following:

$$\nabla_{\theta} v(\pi) = u(\pi) - \frac{\lambda}{2} (\mathcal{F}(y, \mathbf{x}, s))^2$$

### 3 Definition of fairness and the fairness function $\mathcal{F}$

Fairness can be defined in multiple ways, one of which is group fairness. Group fairness encodes the idea of treating different groups equally by imposing equality constraints over different conditional probabilities given the group membership of individuals. Group fairness definitions can be subdivided into three main concepts:

- **No disparate treatment:**
  - **No disparate impact:**
  - **No disparate mistreatment:**
-

### 3.1 Difference in expectation

As talked about in section 1 the choice of the fairness function controls which kind of group fairness the optimization problem is constrained by. In the case of the binary classification scenario demographic parity can be defined as

$$\begin{aligned}
& \pi(d = 1 \mid \mathbf{x}, s = 0)P(\mathbf{x}, y \mid s = 0) = \\
& \pi(d = 1 \mid \mathbf{x}, s = 1)P(\mathbf{x}, y \mid s = 1) \\
& \Leftrightarrow \\
& (0\pi(d = 0 \mid \mathbf{x}, s = 0) + 1\pi(d = 1 \mid \mathbf{x}, s = 0))P(\mathbf{x}, y \mid s = 0) = \\
& (0\pi(d = 0 \mid \mathbf{x}, s = 1) + 1\pi(d = 1 \mid \mathbf{x}, s = 1))P(\mathbf{x}, y \mid s = 1) \\
& \Leftrightarrow \\
& \mathbb{E}_{\substack{\mathbf{x}, y \sim P(\mathbf{x}, y \mid s=0) \\ d \sim \pi(d \mid \mathbf{x}, s)}}[d] = \mathbb{E}_{\substack{\mathbf{x}, y \sim P(\mathbf{x}, y \mid s=1) \\ d \sim \pi(d \mid \mathbf{x}, s)}}[d] \\
& \Leftrightarrow \\
& b_P^0(\pi) = b_P^1(\pi) \\
& \Leftrightarrow \\
& b_P^0(\pi) - b_P^1(\pi) = 0
\end{aligned}$$

which gives a natural choice for the fairness function  $\mathcal{F}(y, \mathbf{x}, s) = b_P^0(\pi) - b_P^1(\pi)$  with the utility function of the unrestrained optimization problem  $v(\pi) = u(\pi) - \frac{\lambda}{2}(b_P^0(\pi) - b_P^1(\pi))^2$ .

### 3.2 Covariance as approximation

The intuitive fairness function that was talked about in section 3.1 has one mayor disadvantage: It is non-convex(?). For that reasons the same authors in a different paper [1] proposed an approximation of this