Learning to predict vs. learning to decide

Floyd Kretschmar

December 13, 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 seperate 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 [1].

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 $\boldsymbol{x} \in \mathcal{X}$, $s \in \mathcal{S}$ and $y \in \mathcal{Y}$ are given by the joint probability distribution $P(\boldsymbol{x}, s, y) = P(y \mid \boldsymbol{x}, s)P(\boldsymbol{x}, s)$ as defined by [1]. Within this framework a **prediction task** is defined as training a model $Q(y \mid x, s; \theta)$ in a supervised mannor, which means solving the optimization problem

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

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 $Q(y \mid x, s; \theta)$ is a probability distribution defining the probability of an individual being a member of specific class given both their features $\boldsymbol{x}$ 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 $Q(y \mid x, s; \theta)$. As seen

in [1] such a policy can be defined as a mapping $\pi : \mathcal{X} \times \mathcal{S} \to \mathcal{P}(0, 1)$ that maps an individuals feature vector and sensitive attribute to a probability distribution over binary decisions $e \in 0, 1$. That means $\pi(e \mid x, s)$ 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 policiy of the form

$$\pi_Q(e = 1 \mid \boldsymbol{x}, s) = \mathbf{1}[Q(y = 1 \mid \boldsymbol{x}, s) \geq c]$$

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

As discussed by [1] this approach has been shown by [2] to often lead to better performance than post-processing a potentially unfair predictor. But the same paper argues that due to the fact that $Q(y = 1 \mid \boldsymbol{x}, s)$ and $P(y = 1 \mid \boldsymbol{x}, s) - \delta_s$ are not exactly equal the resulting policy π_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 θ of a given model Q. The authors of [1] explain that in many real-world applications this assumption is incorrect. In these applications the original data has not been sampeled from the true ground truth distribution $P(\boldsymbol{x}, s, y)$ but instead from a weighted distribution

$$P_{\pi_{Q_0}} = P(y \mid \boldsymbol{x}, s)\pi_0(e = 1 \mid \boldsymbol{x}, s)P(\boldsymbol{x}, s)$$

where π_0 is some intial decision policy that is employed while collecting the data. The authors call this kind of ground truth distribution *induced* by π_0. In this scenario the decision whether or not y $P(y \mid \boldsymbol{x}, s)$ comes into existence is based on the decision generated by π_0. An example of this is a loan decision scenario. If a bank wants to train a machine learning model to make predicitions 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 [1] argue that in such a scenario "for error based learning algorithms under no fairness constraints, learning within detmerninistic threshold policies is guaranteed to fail.". They instead propose to directly learn a probabilistic decision policy π_θ that maximizes the utility, which can be formuated as the following optimization problem:

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

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 θ. The authors propse a utility function of the form

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

where $c \in (0, 1)$ represents the cost considerations of the decision maker. The authors proof that the optimal decision policy π^* can be learned only from data generated by a ground truth distribution P_{π_0} that is induced by π_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 mathematecally speaking: "$\pi_0(e = 1 \mid \boldsymbol{x}, s) > 0$ must be true for any measurable subset of $\mathcal{X} \times \mathcal{S}$ with positive probability under P". For an exploring policy π_0 and any arbitrary policy π the utility is then calculated as

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

which is the same as inverse propensity scoring. To formulate the optimization problem in an unconstraint 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 dependend on $\mathcal{F}$.

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 contraints 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

$$
\begin{gathered}
\pi(e=1 \mid \boldsymbol{x}, s=0) P(\boldsymbol{x}, y \mid s=0) = \\
\pi(e=1 \mid \boldsymbol{x}, s=1) P(\boldsymbol{x}, y \mid s=1) \\
\Leftrightarrow \\
(0\pi(e=0 \mid \boldsymbol{x}, s=0) + 1\pi(e=1 \mid \boldsymbol{x}, s=0)) P(\boldsymbol{x}, y \mid s=0) = \\
(0\pi(e=0 \mid \boldsymbol{x}, s=1) + 1\pi(e=1 \mid \boldsymbol{x}, s=1)) P(\boldsymbol{x}, y \mid s=1) \\
\Leftrightarrow \\
\mathbb{E}_{\substack{\boldsymbol{x}, y \sim P(\boldsymbol{x}, y \mid s=0) \\ e \sim \pi(e \mid \boldsymbol{x}, s)}}[e] = \mathbb{E}_{\substack{\boldsymbol{x}, y \sim P(\boldsymbol{x}, y \mid s=1) \\ e \sim \pi(e \mid \boldsymbol{x}, s)}}[e] \\
\Leftrightarrow \\
\mathbb{E}_{\substack{\boldsymbol{x}, y \sim P(\boldsymbol{x}, y \mid s=0) \\ e \sim \pi(e \mid \boldsymbol{x}, s)}}[f(e, y)] = \mathbb{E}_{\substack{\boldsymbol{x}, y \sim P(\boldsymbol{x}, y \mid s=1) \\ e \sim \pi(e \mid \boldsymbol{x}, s)}}[f(e, y)] \\
\Leftrightarrow \\
b_P^0(\pi) = b_P^1(\pi) \\
\Leftrightarrow \\
b_P^0(\pi) - b_P^1(\pi) = 0
\end{gathered}
$$

which gives a natural choice for the fairness function $\mathcal{F}_{diff}(\pi) = b_P^0(\pi) - b_P^1(\pi)$ with the utility function of the unrestrained optimization problem $v(\pi) = u(\pi) - \frac{\lambda}{2}(b_P^0(\pi) - b_P^1(\pi))^2$. Following an equivalent to the one regarding the utility function in section 2 the inverse propensity scoring has to be applied to the fairness function as well.

$$
\begin{aligned}
b_P^s(\pi) &= \mathbb{E}_{\substack{x, y, s \sim P(\boldsymbol{x}, s, y) \\ e \sim \pi(e \mid \boldsymbol{x}, s)}}[f(e, y)] \\
&= \mathbb{E}_{\substack{\boldsymbol{x}, y, s \sim P_{\pi_0}(\boldsymbol{x}, s, y) \\ e \sim \pi(e \mid \boldsymbol{x}, s)}} \left[\frac{f(e, y)}{\pi_0(e=1 \mid x, s)} \right] = b_{P_{\pi_0}}^s(\pi, \pi_0)
\end{aligned}
$$

This intuitive fairness function has one mayor disadvantage: It is non-convex as can be seen in the following reformulation

$$
\begin{aligned}
&b_P^0(\pi) - b_P^1(\pi) = \mathbb{E}_{\substack{\boldsymbol{x},y\sim P(\boldsymbol{x},y|s=0)\\ e\sim\pi(e|\boldsymbol{x},s=1)}}[f(e,y)] - \mathbb{E}_{\substack{\boldsymbol{x},y\sim P(\boldsymbol{x},y|s=1)\\ e\sim\pi(e|\boldsymbol{x},s=1)}}[f(e,y)] \\
&= \int_x \int_y \underbrace{\left(\int_d f(e,y)\pi(e \mid \boldsymbol{x}, s=0)\, de\right)}_{\text{convex function } g_0(\boldsymbol{x},y)^1} P(\boldsymbol{x}, y \mid s=0)\, dx\, dy - \\
&\int_x \int_y \underbrace{\left(\int_d f(e,y)\pi(e \mid \boldsymbol{x}, s=1)\, de\right)}_{\text{convex function } g_1(\boldsymbol{x},y)^1} P(\boldsymbol{x}, y \mid s=1)\, dx\, dy \\
&= \int_x \underbrace{\int_y g_0(\boldsymbol{x},y) P(\boldsymbol{x}, y \mid s=0)\, dy}_{\text{convex function } h_0(x)^1} dx - \int_x \underbrace{\int_y g_1(\boldsymbol{x},y) P(\boldsymbol{x}, y \mid s=1)\, dy}_{\text{convex function } h_1(x)^1} dx \\
&= \int_x h_0(x)\, dx - \int_x h_1(x)\, dx \\
&= \int_x \underbrace{h_0(x) - h_1(x)}_{convex-concave} dx
\end{aligned}
$$

where $f(e,y) = e$ for the case of demographic parity that has been discussed before.

3.2 Covariance as approximation

As seen in the last part of section 3.1 is the difference in expectations $b_P^0(\pi) - b_P^1(\pi)$ not (necessarily) a convex optimization constraint, therefore making the overall optimization problem hard to optimize. For that reason the authors of [4] introduce a different notion of fairness, that aims to approximate the difference in expectations. For the case of disparate impact propose to formulate the fairness constraint as the covariance between the users sensitive attribute s and the signed distance from the users feature vectors to the decision boundary $d_\theta(\boldsymbol{x})$.

[1] According to [3] $g(x) = \int_A w(y) f(x,y) dy$ is convex if $f(x,y)$ is convex in x for each $y \in A$ and $w(y) \geq 0$ for each $y \in A$

$$
\begin{aligned}
Cov_{DI}(s, d_\theta(\boldsymbol{x})) &= \mathbb{E}_{\substack{\boldsymbol{x},y,s\sim P(\boldsymbol{x},y,s)\\ e\sim\pi(e|\boldsymbol{x},s)}}\left[(s-\mathbb{E}_{s\sim P(s)}[s])(d_\theta(\boldsymbol{x})-\mathbb{E}_{\boldsymbol{x}\sim P(\boldsymbol{x})}[d_\theta(\boldsymbol{x})])\right] \\
=&\mathbb{E}_{\substack{\boldsymbol{x},y,s\sim P(\boldsymbol{x},y,s)\\ e\sim\pi(e|\boldsymbol{x},s)}}\left[(s-\mathbb{E}_{s\sim P(s)}[s])(e-\mathbb{E}_{\substack{\boldsymbol{x},y,s\sim P(\boldsymbol{x},y,s)\\ e\sim\pi(e|\boldsymbol{x},s)}}[e])\right] \\
=&\mathbb{E}_{\substack{\boldsymbol{x},y,s\sim P(\boldsymbol{x},y,s)\\ e\sim\pi(e|\boldsymbol{x},s)}}\left[(s-\mathbb{E}_{s\sim P(s)}[s])e-(s-\mathbb{E}_{s\sim P(s)}[s])\mathbb{E}_{\substack{\boldsymbol{x},y,s\sim P(\boldsymbol{x},y,s)\\ e\sim\pi(e|\boldsymbol{x},s)}}[e]\right] \\
=&\mathbb{E}_{\substack{\boldsymbol{x},y,s\sim P(\boldsymbol{x},y,s)\\ e\sim\pi(e|\boldsymbol{x},s)}}\left[(s-\mathbb{E}_{s\sim P(s)}[s])e\right] \\
&-\mathbb{E}_{\substack{\boldsymbol{x},y,s\sim P(\boldsymbol{x},y,s)\\ e\sim\pi(e|\boldsymbol{x},s)}}\left[(s-\mathbb{E}_{s\sim P(s)}[s])\mathbb{E}_{\substack{\boldsymbol{x},y,s\sim P(\boldsymbol{x},y,s)\\ e\sim\pi(e|\boldsymbol{x},s)}}[e]\right] \\
=&\mathbb{E}_{\substack{\boldsymbol{x},y,s\sim P(\boldsymbol{x},y,s)\\ e\sim\pi(e|\boldsymbol{x},s)}}\left[(s-\mathbb{E}_{s\sim P(s)}[s])e\right] \\
&-\mathbb{E}_{\substack{\boldsymbol{x},y,s\sim P(\boldsymbol{x},y,s)\\ e\sim\pi(e|\boldsymbol{x},s)}}\left[s-\mathbb{E}_{s\sim P(s)}[s]\right]\mathbb{E}_{\substack{\boldsymbol{x},y,s\sim P(\boldsymbol{x},y,s)\\ e\sim\pi(e|\boldsymbol{x},s)}}\left[\mathbb{E}_{\substack{\boldsymbol{x},y,s\sim P(\boldsymbol{x},y,s)\\ e\sim\pi(e|\boldsymbol{x},s)}}[e]\right] \\
=&\mathbb{E}_{\substack{\boldsymbol{x},y,s\sim P(\boldsymbol{x},y,s)\\ e\sim\pi(e|\boldsymbol{x},s)}}\left[(s-\mathbb{E}_{s\sim P(s)}[s])e\right] \\
&-\underbrace{\mathbb{E}_{s\sim P(s)}\left[s-\mathbb{E}_{s\sim P(s)}[s]\right]}_{=0}\mathbb{E}_{\substack{\boldsymbol{x},y,s\sim P(\boldsymbol{x},y,s)\\ e\sim\pi(e|\boldsymbol{x},s)}}[e] \\
=&\mathbb{E}_{\substack{\boldsymbol{x},y,s\sim P(\boldsymbol{x},y,s)\\ e\sim\pi(e|\boldsymbol{x},s)}}\left[(s-\mathbb{E}_{s\sim P(s)}[s])e\right]=\mathcal{F}_{Cov}(\pi)
\end{aligned}
$$

where $d_\theta(x) = e$ is chosen to be the distance function and $\mathcal{D}$ a given set of sample data which is a convex problem. As before we use inverse propensity scoring to control for our data being drawn from the weighted ground truth distribution P_{π_0} instead of the true ground truth distribution P

$$
\begin{aligned}
\mathbb{E}_{s\sim P(s)}[s] &= \int_s sP(s)\,ds \\
&= \int_s s\left(\int_{\boldsymbol{x}}\int_y P(s \mid \boldsymbol{x}, y)P(y \mid \boldsymbol{x})P(\boldsymbol{x})\,d\boldsymbol{x}\,dy\right)ds \\
&= \int_s s\left(\int_{\boldsymbol{x}}\int_y \frac{P(s, \boldsymbol{x}, y)}{P(\boldsymbol{x}, y)}P(y \mid \boldsymbol{x})P(\boldsymbol{x})\,d\boldsymbol{x}\,dy\right)ds \\
&= \int_s s\left(\int_{\boldsymbol{x}}\int_y \frac{P(y \mid s, \boldsymbol{x})P(s \mid \boldsymbol{x})P(\boldsymbol{x})}{P(\boldsymbol{x}, y)}P(y \mid \boldsymbol{x})P(\boldsymbol{x})\,d\boldsymbol{x}\,dy\right)ds \\
&= \int_s s\left(\int_{\boldsymbol{x}}\int_y \frac{P(y \mid s, \boldsymbol{x})P(s \mid \boldsymbol{x})P(\boldsymbol{x})}{P(y \mid \boldsymbol{x})P(\boldsymbol{x})}P(y \mid \boldsymbol{x})P(\boldsymbol{x})\,d\boldsymbol{x}\,dy\right)ds \\
&= \int_s s\left(\int_{\boldsymbol{x}}\int_y \frac{P(y \mid s, \boldsymbol{x})P(s \mid \boldsymbol{x})}{P(y \mid \boldsymbol{x})}P(y \mid \boldsymbol{x})P(\boldsymbol{x})\,d\boldsymbol{x}\,dy\right)ds \\
&= \int_s s\left(\int_{\boldsymbol{x}}\int_y P(y \mid \boldsymbol{x}, s)P(s \mid \boldsymbol{x})P(\boldsymbol{x})\,d\boldsymbol{x}\,dy\right)ds \\
&= \int_s s\left(\int_{\boldsymbol{x}}\int_y P(y \mid \boldsymbol{x}, s)\frac{P(\boldsymbol{x}, s)}{P(\boldsymbol{x})}P(\boldsymbol{x})\,d\boldsymbol{x}\,dy\right)ds \\
&= \int_s s\left(\int_{\boldsymbol{x}}\int_y P(y \mid \boldsymbol{x}, s)P(\boldsymbol{x}, s)\,d\boldsymbol{x}\,dy\right)ds \\
&= \int_s\int_{\boldsymbol{x}}\int_y sP(y \mid \boldsymbol{x}, s)P(\boldsymbol{x}, s)\,d\boldsymbol{x}\,dy\,ds \\
&= \int_s\int_{\boldsymbol{x}}\int_y sP(y \mid \boldsymbol{x}, s)P(\boldsymbol{x}, s)\frac{\pi_0(e=1 \mid x, s)}{\pi_0(e=1 \mid x, s)}\,d\boldsymbol{x}\,dy\,ds \\
&= \int_s\int_{\boldsymbol{x}}\int_y \frac{s}{\pi_0(e=1 \mid x, s)}P(y \mid \boldsymbol{x}, s)\pi_0(e=1 \mid x, s)P(\boldsymbol{x}, s)\,d\boldsymbol{x}\,dy\,ds \\
&= \int_s\int_{\boldsymbol{x}}\int_y \frac{s}{\pi_0(e=1 \mid x, s)}P_{\pi_0}(\boldsymbol{x}, s, y)\,d\boldsymbol{x}\,dy\,ds \\
&= \mathbb{E}_{s\sim P_{\pi_0}(\boldsymbol{x},s,y)}\left[\frac{s}{\pi_0(e=1 \mid x, s)}\right] = \mu_{s_{\pi_0}}
\end{aligned}
$$

for the expectation of the sensitive attribute s and similarly for the entire fairness function

$$
\begin{aligned}
\mathcal{F}_{Cov_P(\pi)} &= \mathbb{E}_{\substack{x,y,s\sim P(\boldsymbol{x},s,y)\\ e\sim\pi(e|\boldsymbol{x},s)}}\left[(s-\mu_{s_{\pi_0}})e\right] \\
&= \mathbb{E}_{\substack{\boldsymbol{x},y,s\sim P_{\pi_0}(\boldsymbol{x},s,y)\\ e\sim\pi(e|\boldsymbol{x},s)}}\left[\frac{(s-\mu_{s_{\pi_0}})e}{\pi_0(e=1 \mid x, s)}\right] = \mathcal{F}_{Cov_{P_{\pi_0}}}(\pi, \pi_0)
\end{aligned}
$$

4 Learning exploring policies

In [1] the authors demonstrate the concept of learning exploring policies using the specific policy class of logistic policies, meaning they choose

$$\begin{aligned}\pi_\theta(d = 1 \mid \boldsymbol{x}, s) &= \sigma(\phi(\boldsymbol{x}, s)^T\theta) \\ &= \frac{1}{1 + e^{-\phi(\boldsymbol{x},s)^T\theta}}\end{aligned}$$

where $\theta \in \Theta \subset \mathbb{R}^m$ are tbe model parameters and $\phi : \mathbb{R}^d \times 0, 1 \to \mathbb{R}^m$ a fixed feature map that maps the features and sensitive attribute of an individual into the parameter space. The authors chose stochastic gradient descent as their optimization problem, meaning the update rule for θ_{t+1} is given by $\theta_{t+1} = \theta_t + \alpha_t \nabla_{\theta_t} v_P(\pi_{\theta_t})$. The gradient of the unconstraint optimization problem $\nabla_\theta v_P(\pi_\theta)$ is given as

$$\begin{aligned}\nabla_\theta v_P(\pi_\theta) &= \nabla_\theta \left(u(\pi_\theta) - \frac{\lambda}{2}(\mathcal{F}(\pi_\theta))^2 \right) \\ &= \nabla_\theta u(\pi_\theta) - \nabla_\theta \frac{\lambda}{2}(\mathcal{F}(\pi_\theta))^2 \\ &= \nabla_\theta u(\pi_\theta) - \lambda(\mathcal{F}(\pi_\theta))\nabla_\theta \mathcal{F}(\pi_\theta)\end{aligned}$$

Then the authors use the log-derivative trick to formulate the gradient of the utility and the fairness functions with regards to the expectation as follows

$$\begin{aligned}\nabla_\theta u(\pi_\theta) &= \nabla_\theta \mathbb{E}_{\substack{\boldsymbol{x},y,s\sim P_{\pi_0}(\boldsymbol{x},s,y) \\ e\sim\pi_\theta(e|\boldsymbol{x},s)}} \left[\frac{e(y-c)}{\pi_0(e = 1 \mid x, s)} \right] \\ &= \nabla_\theta \int \frac{e(y-c)}{\pi_0(e = 1 \mid x, s)} \pi_\theta(e \mid \boldsymbol{x}, s) P_{\pi_0}(\boldsymbol{x}, s, y)\, dx\, dy\, ds\, de \\ &= \int \frac{e(y-c)}{\pi_0(e = 1 \mid x, s)} \nabla_\theta \pi_\theta(e \mid \boldsymbol{x}, s) \frac{\pi_\theta(e \mid \boldsymbol{x}, s)}{\pi_\theta(e \mid \boldsymbol{x}, s)} P_{\pi_0}(\boldsymbol{x}, s, y)\, dx\, dy\, ds\, de \\ &= \int \frac{e(y-c)}{\pi_0(e = 1 \mid x, s)} \pi_\theta(e \mid \boldsymbol{x}, s) \frac{\nabla_\theta \pi_\theta(e \mid \boldsymbol{x}, s)}{\pi_\theta(e \mid \boldsymbol{x}, s)} P_{\pi_0}(\boldsymbol{x}, s, y)\, dx\, dy\, ds\, de \\ &= \int \frac{e(y-c)}{\pi_0(e = 1 \mid x, s)} \pi_\theta(e \mid \boldsymbol{x}, s) \nabla_\theta \log \pi_\theta(e \mid \boldsymbol{x}, s) P_{\pi_0}(\boldsymbol{x}, s, y)\, dx\, dy\, ds\, de \\ &= \int \frac{e(y-c)}{\pi_0(e = 1 \mid x, s)} \nabla_\theta \log \pi_\theta(e \mid \boldsymbol{x}, s) \pi_\theta(e \mid \boldsymbol{x}, s) P_{\pi_0}(\boldsymbol{x}, s, y)\, dx\, dy\, ds\, de \\ &= \mathbb{E}_{\substack{\boldsymbol{x},y,s\sim P_{\pi_0}(\boldsymbol{x},s,y) \\ e\sim\pi_\theta(e|\boldsymbol{x},s)}} \left[\frac{e(y-c)}{\pi_0(e = 1 \mid x, s)} \nabla_\theta \log \pi_\theta(e \mid \boldsymbol{x}, s) \right]\end{aligned}$$

and equivalently for the fairness functions $\mathcal{F}_{Diff}$ and $\mathcal{F}_{Cov}$

$$\begin{aligned}
\nabla_\theta \mathcal{F}_{Diff}(\pi_\theta) &= \nabla_\theta b_P^0(\pi_\theta) - \nabla_\theta b_P^1(\pi_\theta) \\
\nabla_\theta b_P^s(\pi_\theta) &= \mathbb{E}_{\substack{\boldsymbol{x},y,s\sim P_{\pi_0}(\boldsymbol{x},s,y) \\ e\sim\pi_\theta(e|\boldsymbol{x},s)}} \left[\frac{f(e,y)}{\pi_0(e=1 \mid x,s)} \nabla_\theta \log \pi_\theta(e \mid \boldsymbol{x}, s) \right] \\
\nabla_\theta \mathcal{F}_{Cov}(\pi) &= \mathbb{E}_{\substack{\boldsymbol{x},y,s\sim P_{\pi_0}(\boldsymbol{x},s,y) \\ e\sim\pi_\theta(e|\boldsymbol{x},s)}} \left[\frac{(s-\mu_{s_{\pi_0}})e}{\pi_0(e=1 \mid x,s)} \nabla_\theta \log \pi_\theta(e \mid \boldsymbol{x}, s) \right]
\end{aligned}$$

where the gradient of the logarithm can be calculated as follows

$$\begin{aligned}
\nabla_\theta \log \pi_\theta(e \mid \boldsymbol{x}, s) &= \frac{\partial}{\partial \theta_t} \log\left(\sigma(\phi_i^T \theta_t)\right) \\
&= \frac{\partial}{\partial \theta_t} \log\left(\frac{1}{1+e^{-\phi_i^T\theta_t}}\right) \\
&= \frac{\partial}{\partial \theta_t} \log\left(\frac{1}{1+e^{-\alpha}}\right) \\
&= \frac{\partial}{\partial \theta_t} \log\left(\frac{1}{\beta}\right) \\
&= \frac{\partial}{\partial \theta_t} \log\left(\gamma\right) \\
&= \frac{\partial \log(\gamma)}{\partial \gamma} \frac{\partial \gamma}{\partial \beta} \frac{\partial \beta}{\partial \alpha} \frac{\partial \alpha}{\partial \theta_t} \\
&= \frac{\partial \log(\gamma)}{\partial \gamma} \frac{\partial \beta^{-1}}{\partial \beta} \frac{\partial e^{-\alpha}}{\partial \alpha} \frac{\partial \phi_i^T \theta_t}{\partial \theta_t} \\
&= \frac{1}{\gamma} \left(-\frac{1}{\beta^2}\right) \left(-e^{-\alpha}\right) \phi_i \\
&= \frac{1}{\frac{1}{1+e^{-\phi_i^T\theta_t}}} \left(-\frac{1}{(1+e^{-\phi_i^T\theta_t})^2}\right) \left(-e^{-\phi_i^T\theta_t}\right) \phi_i \\
&= \left(-\frac{\left(1+e^{-\phi_i^T\theta_t}\right)}{(1+e^{-\phi_i^T\theta_t})^2}\right) \left(-e^{-\phi_i^T\theta_t}\right) \phi_i \\
&= \frac{e^{-\phi_i^T\theta_t}}{1+e^{-\phi_i^T\theta_t}} \phi_i \\
&= \frac{1}{1+e^{\phi_i^T\theta_t}} \phi_i \\
&= \frac{\phi_i}{1+e^{\phi_i^T\theta_t}}
\end{aligned}$$

where $\phi_i := \phi(\boldsymbol{x}_i, s_i)$ is the fixed feature map evaluated for a given sample i. Using Monte Carlo approximation for the estimation of the expectation we are given the following final form for the gradient of the utility function u_P

$$\begin{aligned}
\nabla_{\theta_t} u(\pi_{\theta_i}) &\approx \frac{1}{n_{t-1}} \sum_{i=1}^{n_{t-1}} \frac{e_i(y_i - c)}{\pi_{\theta_{t-1}}(e = 1 \mid \boldsymbol{x}_i, s_i)} \nabla_\theta \log \pi_\theta(e \mid \boldsymbol{x}_i, s_i) \\
&= \frac{1}{n_{t-1}} \sum_{i=1}^{n_{t-1}} \frac{e_i(y_i - c)}{\sigma(\phi_i^T \theta_{t-1})} \frac{\phi_i}{1 + e^{\phi_i^T \theta_t}} \\
&= \frac{1}{n_{t-1}} \sum_{i=1}^{n_{t-1}} \frac{1}{\frac{1}{1+e^{-\phi_i^T \theta_{t-1}}}} e_i(y_i - c) \frac{\phi_i}{1 + e^{\phi_i^T \theta_t}} \\
&= \frac{1}{n_{t-1}} \sum_{i=1}^{n_{t-1}} \frac{1 + e^{-\phi_i^T \theta_{t-1}}}{1 + e^{\phi_i^T \theta_t}} e_i(y_i - c) \phi_i
\end{aligned}$$

and with an equivalent argument the fairness functions $\mathcal{F}_{Diff}$ and $\mathcal{F}_{Cov}$ take the following form:

$$\begin{aligned}
\nabla_\theta \mathcal{F}_{Diff}(\pi_{\theta_i}) &= \nabla_{\theta_i} b_P^0(\pi_{\theta_i}) - \nabla_{\theta_i} b_P^1(\pi_{\theta_i}) \\
\nabla_{\theta_i} b_P^s(\pi_{\theta_i}) &\approx \frac{1}{n_{t-1}} \sum_{i=1}^{n_{t-1}} \frac{1 + e^{-\phi_i^T \theta_{t-1}}}{1 + e^{\phi_i^T \theta_t}} f(e_i, y_i) \phi_i \\
\nabla_{\theta_t} \mathcal{F}_{Cov}(\pi_{\theta_i}) &\approx \frac{1}{n_{t-1}} \sum_{i=1}^{n_{t-1}} \frac{1 + e^{-\phi_i^T \theta_{t-1}}}{1 + e^{\phi_i^T \theta_t}} e_i(y_i - c) \phi_i
\end{aligned}$$

References

[1] N. Kilbertus, M. Gomez-Rodriguez, B. Schölkopf, K. Muandet, and I. Valera, "Improving consequential decision making under imperfect predictions," *CoRR*, vol. abs/1902.02979, 2019.

[2] B. E. Woodworth, S. Gunasekar, M. I. Ohannessian, and N. Srebro, "Learning non-discriminatory predictors," *CoRR*, vol. abs/1702.06081, 2017.

[3] S. Boyd and L. Vandenberghe, *Convex Optimization*. New York, NY, USA: Cambridge University Press, 2004.

[4] M. B. Zafar, I. Valera, M. Gomez-Rodriguez, and K. P. Gummadi, "Fairness constraints: A flexible approach for fair classification," *Journal of Machine Learning Research*, vol. 20, no. 75, pp. 1–42, 2019.