Fair Policy Targeting Note

JCY

2023 年 7 月 30 日

1 Challenges

(1)Define unfairness measures.

(2)Output the deterministic allocation with different methods.

(3)Compare the unfairness among the original treatment, the Fair Policy Targeting method and the Welfare Maximization method.

2 Two methods to estimate policy

2.1 Welfare Maximization

The population equivalent of the EWM problem belongs to the Pareto frontier. Namely, $\arg\max_{\pi\in\Pi}\left\{p_1 W_1(\pi)+\left(1-p_1\right) W_0(\pi)\right\} \subseteq \Pi_o$. An alternative approach consists in maximizing weighted combinations of the welfare with the weights for each group as given. For instance the allocation

$$\check{\pi}_\omega \in \arg\max_{\pi\in\Pi}\{\omega W_1(\pi)+(1-\omega) W_0(\pi)\} \subseteq \Pi_o, \tag{1}$$

for some specific weight ω belongs to the Pareto frontier.

2.2 Fair Policy Targeting

Under Assumption 2.2, $\pi^\star \in \mathcal{C}(\Pi)$ if and only if

$$\pi^\star \in \arg\inf_{\pi\in\Pi} \operatorname{UnFairness}(\pi) \tag{2}$$

$$\text{subject to } \alpha W_1(\pi)+(1-\alpha) W_0(\pi) \geq \bar{W}_\alpha, \text{for some } \alpha \in (0,1). \tag{3}$$

Formally characterizes the policymakers decision problem, which consists of minimizing the policy’s unfairness criterion, under the condition that the policy is Pareto optimal. Each group’s importance (i.e., α) is implicitly chosen within the optimization problem to maximize fairness. This approach allows for a transparent choice of the policy based on the policy-makers definition of fairness.

3 UnFairness Definition

We consider three notions of UnFairness:

3.1 Counterfactual Envy

Let the conditional welfare, for the policy function being assigned to the opposite attribute, i.e., the effect of $\pi(x, s_1)$, on the group s_2 , conditional on covariates, be

$$V_{\pi(x,s_1)}(x, s_2) = \mathbb{E}\left[\pi(x, s_1) Y_i(1, s_2) + (1 - \pi(x, s_1)) Y_i(0, s_2) \mid X_i(s_2) = x\right] \tag{4}$$

We say that the agent with attribute s_2 envies the agent with attribute s_1, if her welfare (on the right-hand side of Equation (5)) exceeds the welfare she would have received had her covariate and policy been assigned the opposite attribute (left-hand side of Equation (5)), namely

$$\mathbb{E}_{X(s_1)}\left[V_{\pi(X(s_1),s_1)}(X(s_1), s_2)\right] > \mathbb{E}_{X(s_2)}\left[V_{\pi(X(s_2),s_2)}(X(s_2), s_2)\right] \tag{5}$$

We then measure the unfairness towards an individual with attribute s_2 as

$$\mathcal{A}(s_1, s_2; \pi) = \mathbb{E}_{X(s_1)}\left[V_{\pi(X(s_1),s_1)}(X(s_1), s_2)\right] - \mathbb{E}_{X(s_2)}\left[V_{\pi(X(s_2),s_2)}(X(s_2), s_2)\right]. \tag{6}$$

Whenever we aim not to discriminate in either direction, we take the sum of the effects $\mathcal{A}(s_1, s_2; \pi)$ and $\mathcal{A}(s_2, s_1; \pi)$ it connects to previous notions of counterfactual fairness (Kilbertus et al., 2017).

3.2 Predictive Disparity

Prediction disparity and its empirical counterpart take the following form

$$C(\pi) = \mathbb{E}[\pi(X, S) \mid S = 0] - \mathbb{E}[\pi(X, S) \mid S = 1], \quad \hat{C}(\pi) = \frac{\sum_{i=1}^{n} \pi(X_i)(1 - S_i)}{n(1 - \hat{p}_1)} - \frac{\sum_{i=1}^{n} \pi(X_i) S_i}{n\hat{p}_1}, \tag{7}$$

Prediction disparity captures disparity in the treatment probability between groups.

(Welfare disparity). Define the welfare disparity and its empirical counterpart as

$$D(\pi) = W_0(\pi) - W_1(\pi), \quad \widehat{D}(\pi) = \widehat{W}_0(\pi) - \widehat{W}_1(\pi). \tag{8}$$

3.3 Predictive disparity with absolute value

Predictive disparity with absolute value.

The policymaker may also consider $|D(\pi)|$ or $|C(\pi)|$ as measures of UnFairness, in which case the policymaker treats the two groups symmetrically.

4 Simulation

4.1 Background

The paper studies the effect of an entrepreneur ship training and incubation program for undergraduate students in North America on subsequent entrepreneurial activity. We have in total 335 observations, of which 53% treated and the remaining under control, and 26% of applicants are women.15 The population of interest is the pool of final applicants. We construct a targeting rule that assigns the award to the finalist based on the applicant’ s observable characteristics. We maximize subsequent entrepreneurial activity, which is captured using a dummy variable, indicating whether the participant worked in the startup once the program ended.

4.2 Setting

Consider linear decision rules, given their large use in economics

$$\Pi=\left\{\pi(x,\text{ fem })=1\left\{\beta_0+\beta_1\text{ fem }+x^{\top}\phi\geq 0\right\},\quad(\beta_0,\beta_1,\phi)\in\mathcal{B}\right\}. \tag{9}$$

We allow covariates x to be the years to graduation, years of entrepreneurship, the region of the start-up, the major, the school rank.

We consider in-sample capacity constraints imposed on the function class with at most 150 individuals selected for the treatment.

Consider three nested function classes for the welfare maximization method.

The first does not impose any restriction except for the functional form.

The second, imposes that $\beta_1=0$.

The third class imposes that $\beta_1=0$ and that the average effect of the policy on females is at least as large as the one on males. The function classes are

$$\begin{aligned}
&\Pi_1=\left\{\pi(x,\text{ fem })=1\left\{\beta_0+\beta_1\text{ fem }+x^{\top}\phi\geq 0\right\},\quad(\beta_0,\beta_1,\phi)\in\mathcal{B}\right\},\\
&\Pi_2=\left\{\pi(x)=1\left\{\beta_0+x^{\top}\phi\geq 0\right\}\right\},\\
&\Pi_3=\left\{\pi(x)=1\left\{\beta_0+x^{\top}\phi\geq 0\right\},\quad\mathbb{E}_n\left[\left(Y_i(1)-Y_i(0)\right)\pi\left(X_i\right)\mid S=1\right]\geq\mathbb{E}_n\left[\left(Y_i(1)-Y_i(0)\right)\pi\left(X_i\right)\mid S=0\right]\right\},
\end{aligned} \tag{10}$$

where $\mathbb{E}_n[\cdot]$ denote the empirical expectation, estimated using the doubly-robust method.

4.3 Result

4.3.1 The welfare improvement and the importance weights assigned by differen methods

表 1: FTP Envy refers to the Fair Targeting rule that minimizes envy-freeness unfairness; FTP Predictive Disp refers to the Pareto allocation that minimizes the difference in probability of treatment (Abs indicate in absolute value); Welfare Max.1 denotes the method that maximizes the empirical welfare considering Π_1, and similarly Welfare Max.2,3 for the function classes, respectively Π_2, Π_3.

	Welf Fem	Welf Mal	Weight
Fair Envy	0.376	0.272	0.384
FTP Pred	0.432	0.224	0.847
FTP Pred Abs	0.433	0.208	0.924
Welfare Max. 1	0.376	0.272	0.266
Welfare Max. 2	0.288	0.285	0.266
Welfare Max. 3	0.331	0.265	0.266

We collect results of the welfare on female and male students, as well as the relative importance weight assigned to each group for methods that maximize different UnFairness measures in Table 1.

4.3.2 Deterministic allocations in different methods

表 2: Policy Distribution

	Gender	Original treatment	Fair Envy	FTP Pred	FTP Pred Abs	Welfare Max. 1	Welfare Max. 2	Welfare Max. 3
1	0	1	0	0	0	0	1	1
2	1	0	1	1	1	1	0	0
3	0	1	1	0	0	1	0	0
4	1	1	1	1	1	1	1	1
5	0	0	1	0	0	1	0	0
6	0	0	1	0	0	1	0	0
...								
350	0	0	1	0	1	1	0	1
351	1	0	0	1	1	0	0	0
353	0	0	1	1	1	1	1	1
354	0	0	0	0	1	0	1	1
355	1	0	0	0	1	0	1	1
356	1	0	0	0	0	0	0	0

Table 2 shows the different policy distribution under different methods that maximize different UnFairness measures.

4.3.3 Unfairness levels with different methods

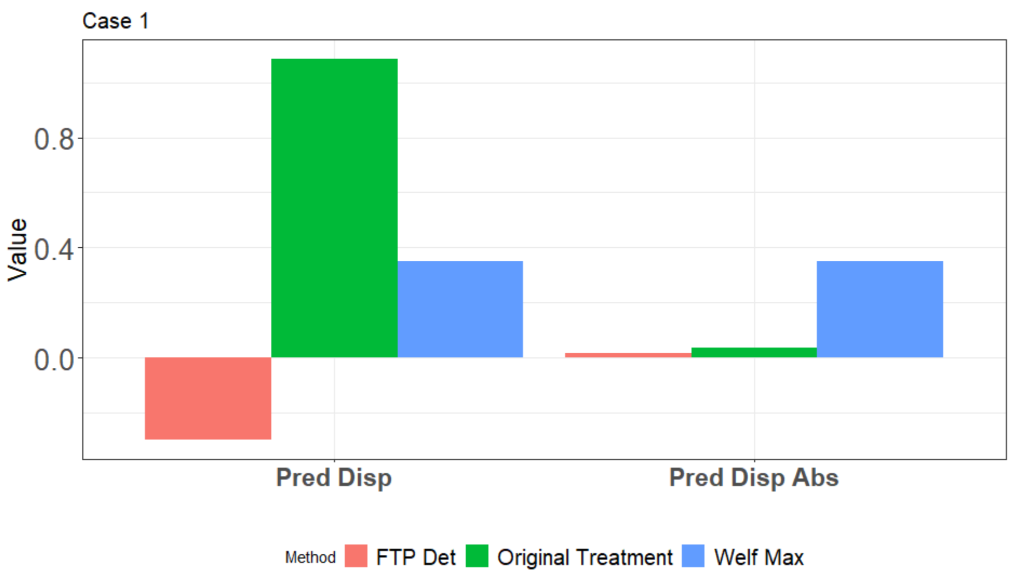

图 1: Unfairness level of the Fair Policy Targeting method with a deterministic allocation rule (in red), Original treatment (in green), and of the welfare maximization method (in blue).

F1 reports the unfairness level with unfairness measured as the difference in the probability of treatments between the two groups. Overall, F1 shows that the level of the unfairness of the proposed method is uniformly smaller than the unfairness achieved by maximizing welfare and also smaller than the original treatment.

5 Code

5.1 Table1

```
#可放在code_new_submission运行
load('./results/elements.RData')
load('./results/alg_envy.RData')
load('./results/alg_parity_abs.RData')
res_ms1_parity_abs <- res_ms1_parity
res_ms3_parity_abs <- res_ms3_parity
load('./results/alg_parity.RData')
source('./library/library.R')
load('./results/solution_EWM.RData')
```

```
## Case 1: 3:8 as additional covariates
#不同定义下的各个人的01分配
C1.pi=cbind(S,D,res_ms3_envy$result$policies,res_ms3_parity$result$
   policies,res_ms3_parity_abs$result$policies,res0_EWM_C1$pi,
   res02_EWM_C1$pi,res03_EWM_C1$pi)
C1.pi=as.data.frame(C1.pi)
colnames(C1.pi)=c('S','D','envy_pi','Predictive_Disp_pi','
   Predictive_Disp_abs_pi','class1_pi','class2_pi','class3_pi')

#不同定义下的分配与原始分配的不同
dif=c(sum(abs(C1.pi$D-C1.pi$envy_pi)),sum(abs(C1.pi$D-C1.pi$
   Predictive_Disp_pi)),sum(abs(C1.pi$D-C1.pi$Predictive_Disp_abs_pi)),
   sum(abs(C1.pi$D-C1.pi$class1_pi)),sum(abs(C1.pi$D-C1.pi$class2_pi)),
   sum(abs(C1.pi$D-C1.pi$class3_pi)))
dif=dif/(length(D))

C1.pi.table= rbind(head(C1.pi),tail(C1.pi))
colnames(C1.pi.table) <- c('Gender','Original treatment','Fair Envy', '
   FTP Pred', 'FTP Pred Abs', 'Welfare Max. 1 ',
'Welfare Max. 2', 'Welfare Max. 3')
C1.pi.table=as.matrix(C1.pi.table)
library(stargazer)
stargazer(C1.pi.table)
write.table(C1.pi.table, file = './tables/c1piest_main_text.txt')
```

5.2 Table2

```
beta_envy_C2 <- res_ms1_envy$result$beta
weight_envy_C2 <- res_ms1_envy$result$alpha
## Compute welfare
g_i <- (Y  - m1)* D/propensity1 - (1 - D) * (Y - m0)/(1 - propensity1) +
    m1 - m0
baseline_effect <- (1 - D) * (Y - m0)/(1 - propensity1) + m0

#计算不同性别下的比例
############################################
beta_envy_C1 <- res_ms3_envy$result$beta
```

```
weight_envy_C1 <- res_ms3_envy$result$alpha

welf1_envy_C1 <- mean(sapply(cbind(1, 1, as.matrix(X[,c(3:8)]))%*%
    beta_envy_C1, function(y) ifelse(y< 0,0,1))*g_i*S/propensity2 +
    baseline_effect*S/propensity2 )
welf0_envy_C1 <- mean(sapply(cbind(1, 0, as.matrix(X[,c(3:8)]))%*%
    beta_envy_C1, function(y) ifelse(y< 0,0,1))*g_i*(1 - S)/(1 -
    propensity2) + baseline_effect*(1 - S)/(1 - propensity2 ))

beta_parity_C1 <- res_ms3_parity$result$beta
weight_parity_C1 <- res_ms3_parity$result$alpha
## Compute welfare
welf1_parity_C1 <- mean(sapply(cbind(1, 1, as.matrix(X[,c(3:8)]))%*%
    beta_parity_C1, function(y) ifelse(y< 0,0,1))*g_i*S/propensity2 +
    baseline_effect*S/propensity2 )
welf0_parity_C1 <- mean(sapply(cbind(1, 0, as.matrix(X[,c(3:8)]))%*%
    beta_parity_C1, function(y) ifelse(y< 0,0,1))*g_i*(1 - S)/(1 -
    propensity2) + baseline_effect*(1 - S)/(1 - propensity2 ))



beta_parity_C1_abs <- res_ms3_parity_abs$result$beta
weight_parity_C1_abs <- res_ms3_parity_abs$result$alpha
## Compute welfare
welf1_parity_C1_abs <- mean(sapply(cbind(1, 1, as.matrix(X[,c(3:8)]))%*%
    beta_parity_C1_abs, function(y) ifelse(y< 0,0,1))*g_i*S/propensity2 +
     baseline_effect*S/propensity2 )
welf0_parity_C1_abs <- mean(sapply(cbind(1, 0, as.matrix(X[,c(3:8)]))%*%
    beta_parity_C1_abs, function(y) ifelse(y< 0,0,1))*g_i*(1 - S)/(1 -
    propensity2) + baseline_effect*(1 - S)/(1 - propensity2 ))

#EWM
## Consider Case 1
beta1_C1_0 <- res0_EWM_C1$beta
beta1_C1_2 <- res02_EWM_C1$beta
beta1_C1_3 <- res03_EWM_C1$beta

beta1_C2_0 <- res0_EWM_C2$beta
```

```
beta1_C2_2 <- res02_EWM_C2$beta
beta1_C2_3 <- res03_EWM_C2$beta
## Compute the corresponding welfares

## Case 1
welf1_EWM_C1_0 <- mean(sapply(cbind(1, 1, as.matrix(X[,c(3:8)]))%*%
    beta1_C1_0, function(y) ifelse(y< 0,0,1))*g_i*S/propensity2 +
    baseline_effect*S/propensity2 )
welf0_EWM_C1_0 <- mean(sapply(cbind(1, 0, as.matrix(X[,c(3:8)]))%*%
    beta1_C1_0, function(y) ifelse(y< 0,0,1))*g_i*(1 - S)/(1 -
    propensity2) + baseline_effect*(1 - S)/(1 - propensity2) )

welf1_EWM_C1_2 <- mean(sapply(cbind(1, as.matrix(X[,c(3:8)]))%*%
    beta1_C1_2, function(y) ifelse(y< 0,0,1))*g_i*S/propensity2 +
    baseline_effect*S/propensity2 )
welf0_EWM_C1_2 <- mean(sapply(cbind(1, as.matrix(X[,c(3:8)]))%*%
    beta1_C1_2, function(y) ifelse(y< 0,0,1))*g_i*(1 - S)/(1 -
    propensity2) + baseline_effect*(1 - S)/(1 - propensity2) )

welf1_EWM_C1_3 <- mean(sapply(cbind(1, as.matrix(X[,c(3:8)]))%*%
    beta1_C1_3, function(y) ifelse(y< 0,0,1))*g_i*S/propensity2 +
    baseline_effect*S/propensity2 )
welf0_EWM_C1_3 <- mean(sapply(cbind(1, as.matrix(X[,c(3:8)]))%*%
    beta1_C1_3, function(y) ifelse(y< 0,0,1))*g_i*(1 - S)/(1 -
    propensity2) + baseline_effect*(1 - S)/(1 - propensity2) )

## Construct the table
welfares_fem_C1 <- c(welf1_envy_C1, welf1_parity_C1,welf1_parity_C1_abs,
     welf1_EWM_C1_0, welf1_EWM_C1_2, welf1_EWM_C1_3)
welfares_mal_C1 <- c(welf0_envy_C1, welf0_parity_C1,welf0_parity_C1_abs,
     welf0_EWM_C1_0, welf0_EWM_C1_2, welf0_EWM_C1_3)

## alpha for EWM is mean(S) by definition
alphas_C1 <- c(weight_envy_C1,  weight_parity_C1, weight_parity_C1_abs,
    rep(mean(S), 3))

## Construct table

```

```
C1_table <- cbind(welfares_fem_C1, welfares_mal_C1,
alphas_C1)
colnames(C1_table) <- c('C1 Welf Fem',  'C1 Welf Mal',
'C1 Weight')
rownames(C1_table) <- c('Fair Envy', 'FTP Pred', 'FTP Pred Abs', 'Welfare Max. 1 ',
'Welfare Max. 2', 'Welfare Max. 3')

library(stargazer)
stargazer(C1_table)
write.table(C1_table, file = './tables/tablec1_main_text.txt')
```

5.3 Figure 1

```

load('./results/elements.RData')
load('./results/alg_parity.RData')
deterministic_ms1 <- res_ms1_parity
deterministic_ms3 <- res_ms3_parity
source('./library/library.R')
library(ggplot2)
library(ggpubr)
original_C1 <-sum(S * D )/sum(S) + sum((1 - S) * D )/sum(1 - S)

parity_C1_det <- -sum(S * deterministic_ms3$result$policies )/sum(S) + sum((1 - S) * deterministic_ms3$result$policies )/sum(1 - S)

#####################################
########## Absolute value #   ########
####################################
load('./results/elements.RData')
load('./results/alg_parity_abs.RData')
deterministic_ms3 <- res_ms3_parity
source('./library/library.R')
original_C1_abs <- abs(-sum(S * D )/sum(S) + sum((1 - S) * D )/sum(1 - S) )
parity_C1_det_abs <- abs(-sum(S * deterministic_ms3$result$policies )/sum(S) + sum((1 - S) * deterministic_ms3$result$policies )/sum(1 - S)
```

```
    )
##########################################
### Compare with EWM
##########################################

load('./results/solution_EWM.RData')
parity_EWM_C1 <- - sum(S * res0_EWM_C1$pi)/sum(S) + sum((1 - S) *
   res0_EWM_C1$pi)/sum(1 - S)
parity_EWM_C1_abs <- abs(- sum(S * res0_EWM_C1$pi)/sum(S) + sum((1 - S)
   * res0_EWM_C1$pi)/sum(1 - S))


library(ggplot2)

data_frame1 <- c(parity_C1_det, original_C1, parity_EWM_C1,
parity_C1_det_abs, original_C1_abs, parity_EWM_C1_abs)


types <- c('FTP Det', 'Original Treatment', 'Welf Max',
'FTP Det', 'Original Treatment', 'Welf Max')

unfairness <- c('Pred Disp', 'Pred Disp', 'Pred Disp',
'Pred Disp Abs', 'Pred Disp Abs', 'Pred Disp Abs')

dd1 <- cbind(data_frame1, types, unfairness)
dd1 <- as.data.frame(dd1)
names(dd1) <- c('UnFairness', 'Method', 'Type')
dd1[,1] <- as.numeric(as.character(dd1[,1]))

bar_chart1 <- ggplot(dd1, aes(y=UnFairness, x=Type, fill = Method)) +
geom_bar(position="dodge", stat="identity") +
ggtitle("Case 1") +
ylab("Value") +
xlab('') +
theme_bw() +
theme(legend.position="bottom",
axis.text.x = element_text(face="bold",
size=17),
```

```
axis.title.x = element_text(size=17),
axis.title.y = element_text(size=17),
legend.text=element_text(size = 15),
plot.title = element_text(size=15),
axis.text.y = element_text(size = 20))


plot(bar_chart1)
```