Average Individual Fairness

Aaron Roth

Based on Joint Work with:
Michael Kearns and Saeed Sharifimalvajerdi
Why was the classifier “unfair”?  

**Question:** Who was harmed?  
**Possible Answer:** The qualified applicants mistakenly rejected.  
**False Negative Rate:** The rate at which harm is done.

**Fairness:** Equal false negative rates across groups?  
[Chouldechova], [Hardt, Price, Srebro], [Kleinberg, Mullainathan, Raghavan]

Statistical Fairness Definitions:
1. Partition the world into groups (often according to a “protected attribute”)
2. Pick your favorite statistic of a classifier.
3. Ask that the statistic be (approximately) equalized across groups.
But...

- A classifier equalizes false negative rates. What does it promise you?
  - The *rate* in false negative rate assumes you are a uniformly random member of your population.
  - If you have reason to believe otherwise, it promises you nothing...
For example

- The following allocation equalizes false negative rates across all four groups.
Sometimes individuals are subject to more than one classification task...
The Idea

• Postulate a distribution over *problems* and *individuals*.
• Ask for a *mapping between problems and classifiers* that equalizes false negative rates across every pair of individuals.
• Redefine *rate*:
  
  *Averaged over the problem distribution.*

  An *individual* definition of fairness.
A Formalization

• An unknown distribution $P$ over individuals $x_i \in X$
• An unknown distribution $Q$ over problems $f_j: X \rightarrow \{0,1\}$, $f_j \in F$
• A hypothesis class $H \subseteq \{0,1\}^X$ (Note $f_j$’s not necessarily in $H$)
• Task: Find a mapping from problems to hypotheses $\psi \in (\Delta H)^F$
  • A new “problem” will be represented as a new labelling of the training set.
  • Finding the hypothesis corresponding to a new problem shouldn’t require resolving old problems. (Allows online decision making)
What to Hope For (Computationally)

• Machine learning learning is already computationally hard [KSS92,KS08,FGKP09,FGPW14,...] even for simple classes like halfspaces.

• So we shouldn’t hope for an algorithm with worst-case guarantees...
  • But we might hope for an efficient reduction to unconstrained (weighted) learning problems.

• “Oracle Efficient Algorithms”
  • This design methodology often results in practical algorithms.
Computing the Optimal Empirical Solution.

Initialize $\lambda_{i}^{1} = 1/n$ for each $i \in \{1, ..., n\}$

For $t = 1$ to $T = O\left(\frac{\log n}{\epsilon^2}\right)$

- **Learner Best Responds:**
  - For each problem $j$, solve the learning problem $h_{j}^{t} = A(S_{j}^{t})$ for $S_{j}^{t} = \left\{\left(\lambda_{i}^{t} + \frac{1}{n}, x_{i}, f_{j}(x_{i})\right)\right\}_{i=1}^{n}$
  - Set $\gamma^{t} = 1[\sum_{i}^{n} \lambda_{i}^{t} \geq 0]$

- **Auditor Updates Weights:**
  - Multiply $\lambda_{i}^{t}$ by $(\text{err}(x_{i}, h^{t}, \hat{Q}) - \gamma)$ for each expert $i$ and renormalize to get updated weights $\lambda_{i}^{t+1}$.

Output the weights $\lambda_{i}^{t}$ for each person $i$ and step $t$. 
Defining $\psi$

- Parameterized by the sequence of dual variables $\lambda^T = \{\lambda^t\}_{t=1}^T$

$$\psi_{\lambda^T}(f):$$

For $t = 1$ to $T$

- Solve the learning problem $h^t = A(S^t)$ for $S^t = \left\{ \left( \lambda^t_i + \frac{1}{n}, x_i, f(x_i) \right) \right\}_{i=1}^n$

Output $p_f \in \Delta H$ where $p_f$ is uniform over $\{h^t\}_{t=1}^T$

(Consistent with ERM solution)
Computing the Optimal Empirical Solution.

**Theorem:** After \( \mathcal{O} \left( m \cdot \frac{\log n}{\epsilon^2} \right) \) calls to the learning oracle, the algorithm returns a solution \( p \in (\Delta H)^m \) that achieves empirical error at most:

\[
\text{OPT}(\alpha, \hat{P}, \hat{Q}) + \epsilon
\]

and satisfies for every \( i, i' \in \{1, \ldots, n\} \):

\[
|\text{FN}(x_i, p, \hat{Q}) - \text{FN}(x_{i'}, p, \hat{Q})| \leq \alpha + \epsilon
\]
Generalization: Two Directions

\[
P \xrightarrow{\hat{P}} S \xrightarrow{f_1 \ldots f_m} S' \xrightarrow{Q}
\]
Generalization

**Theorem:** Assuming

1) $m \geq \text{poly}\left(\log n, \frac{1}{\epsilon}, \log \frac{1}{\delta}\right)$,

2) $n \geq \text{poly}\left(m, VCDIM(H), \frac{1}{\epsilon}, \frac{1}{\beta}, \log \frac{1}{\delta}\right)$

the algorithm returns a solution $\psi$ that with probability $1 - \delta$ achieves error at most:

$$OPT(\alpha, P, Q) + \epsilon$$

and is such that with probability $1 - \beta$ over $x, x' \sim P$:

$$|FN(x, \psi, Q) - FN(x', \psi, Q)| \leq \alpha + \epsilon$$
Does it work?

• It is important to experimentally verify “oracle efficient” algorithms, since it is possible to abuse the model.
  • E.g. use learning oracle as an arbitrary NP oracle.

• A brief “Sanity Check” experiment:
  • Dataset: Communities and Crime
  • First 50 features are designated as “problems” (i.e. labels to predict)
  • Remaining features treated as features for learning.
Takeaways

• We should think carefully about what definitions of “fairness” really promise to individuals.
• Making promises to individuals is sometimes possible, even without making heroic assumptions.
• Once we fix a definition, there is often an interesting algorithm design problem.
• Once we have an algorithm, we can have the tools to explore inevitable tradeoffs.
Thanks!

Average Individual Fairness: Algorithms, Generalization and Experiments
Michael Kearns, Aaron Roth, Saeed Sharifimalvajerdi

Shameless book plug:
The Ethical Algorithm
Michael Kearns and Aaron Roth