# Exercise 10 - Solution

**Introduction to Fairness & Combining Logic and Deep Learning**

Reliable and Trustworthy Artificial Intelligence
ETH Zurich

## 1 Introduction to Fairness

**Problem 1** (Fairness in automated recruiting)**.** Consider a setting in which a company wants to use an automated component in their hiring pipeline. To this end, the recruiting team would like to use a classifier $h : \mathcal{X} \to \{0, 1\}$, which takes in an input $x$ containing data about a certain applicant (e.g. their CV, number of years of experience, grades in relevant university courses, etc.). The output is a binary label $y = h(x)$ indicating the recommendation of the classifier about the application outcome, with $y = 0$ meaning that no offer should be made to this applicant and $y = 1$ meaning that the applicant should receive an offer. The goal is to obtain a classifier that is accurate at predicting whether an applicant is qualified for the job or not. However, by law the company should not discriminate applicants based on whether they come in with a M.Sc. or PhD degree.

To train a model, the ML engineering team gathers data about 40 applicants that applied to the company previously. For each of the applicants, there is information about their application profiles $\{x_i\}_{i=1}^{40}$, their protected attributes $\{g_i\}_{i=1}^{40}$ (indicating an M.Sc. or PhD degree) and their "true labels" $\{y_i\}_{i=1}^{40}$ indicating they are actually qualified for the job or not (e.g. labeled by an expert recruiter). We assume that these triplets $\{(x_i, g_i, y_i)\}_{i=1}^{40}$ are i.i.d. samples from a distribution $\mathcal{D}$ on $\mathcal{X} \times \{0, 1\} \times \{0, 1\}$.

A summary of the training data is presented in the table below, where the total number of qualified ($y_i = 1$) and unqualified ($y_i = 0$) applicants with M.Sc. and PhD respectively are listed.

| Degree \ Qualified | Yes | No | Total |
|---|---|---|---|
| M.Sc. | 16 | 12 | 28 |
| PhD | 8 | 4 | 12 |

**Note:** For the group fairness part of the problem, the classifiers we will consider here do NOT take $g$ as an input. This notation is more general than considering classifiers $h(x, g)$, since if we want to model classifiers that do use $g$, we can just assume that $g$ is also included inside $x$ (e.g. as one additional feature)[1]. In addition, in many cases it is explicitly forbidden by law to directly use a protected attribute when making a decision. Finally, information about $g$ may not be available at test/prediction time.

Even when $g$ is not used inside the classifier, one can still use information about the protected attribute at training time, to guide the training process towards discovering classifiers that will be more fair at test time. This is also one of the points of this exercise. Note that the ML literature is inconsistent on whether the protected attribute can be used as an input to the classifier or not and some papers do consider classifiers that use $g$ also directly. In a lot of applications this is acceptable.

a) The ML team now applies classic ML techniques to learn several classifiers $h_j : \mathcal{X} \to \mathcal{Y}$. For each classifier, we give the number of applicants *from the training data* of each of 4 types ((un-)qualified with a PhD/M.Sc.) that were recommended for acceptance by the learned classifier.

- $h_1$ makes an offer to 7 qualified M.Sc. and 3 qualified PhD applicants
- $h_2$ makes an offer to 4 qualified M.Sc., 2 qualified PhD, 12 unqualified M.Sc. applicants
- $h_3$ makes an offer to 4 qualified M.Sc., 2 qualified PhD, 9 unqualified M.Sc., 3 unqualified PhD applicants
- $h_4$ makes an offer to 7 unqualified M.Sc. and 3 qualified PhD applicants
- $h_5$ makes an offer to all qualified applicants and to none of the unqualified ones
- $h_6$ makes an offer to all unqualified applicants and to none of the qualified ones
- $h_7$ makes an offer to nobody
- $h_8$ makes an offer to exactly half of the applicants from each of the 4 types (accepts 8 qualified M.Sc. etc.), therefore essentially acting as a random $50 - 50$ classifier

For each classifier, determine which of the properties demographic parity, equalized odds and equality of opportunity are satisfied.

[1] We will actually do that in part e), where individual fairness will impose that we do not use $g$ "much" in a classifier.

**Note:** Here you are meant to evaluate the fairness properties on the empirical, training data distribution (i.e. you should use empirical counts instead of probabilities under $\mathcal{D}$).

b) Describe all classifiers that achieve both demographic parity and equalized odds (on the training data) by specifying the number of applicants of each of the 4 types ((un)-qualified M.Sc./PhD) that such classifiers recommend giving offers to.

c) Propose a quantity $\widehat{\Gamma}(h)$ that measures the *amount of demographic parity unfairness* that a classifier $h$ possesses on the training data (that is, a metric measuring how far a classifier is from achieving demographic parity). Evaluate the unfairness of classifiers $h_1, h_2, h_3, h_4$ under this metric.

d) Define the corresponding measure $\Gamma(h)$ on the population level (under the distribution $\mathcal{D}$ of the data, as in the lecture). For a fixed classifier $h$, do you expect the value of the empirical unfairness measure $\widehat{\Gamma}(h)$ to approach the value of the population measure $\Gamma(h)$ as we evaluate $\widehat{\Gamma}(h)$ on more and more applicants? Justify your answer informally.

e) Now consider a situation where the company is instead interested in ensuring a notion of individual fairness. Assume that the input variable $x$ consists of:

- CV (text)
- Age (a positive integer)
- A binary indicator of whether the applicant has a PhD or an M.Sc.

Suppose that the CV does not contain any information about the age and the degree of the applicant (e.g. because this information was removed during pre-processing) and that a metric $d$ is given, which measures a distance between any two CVs (e.g. by using a large language model). Assume also that the fairness requirement that the company is trying to fulfil is "Do not discriminate based on the degree of an applicant. Also you should treat applicants whose age differs by no more that 10 years similarly". Define a distance measure between any two inputs $x$ and $x'$ (any pair of applicants) that can be used together with the concept of individual fairness to ensure that the stated fairness notion is addressed by the company.

**Solution 1.** The empirical equivalent of the demographic parity notion is

$$\frac{\texttt{\# of accepted applicants with M.Sc.}}{\texttt{\# of applicants with M.Sc.}} = \frac{\texttt{\# of accepted applicants with PhD}}{\texttt{\# of applicants with PhD}}.$$

Similarly equalized odds requires that

$$\frac{\texttt{\# of accepted qualified applicants with M.Sc.}}{\texttt{\# of qualified applicants with M.Sc.}}$$

equals

$$\frac{\texttt{\# of accepted qualified applicants with PhD}}{\texttt{\# of qualified applicants with PhD}}$$

and

$$\frac{\texttt{\# of accepted unqualified applicants with M.Sc.}}{\texttt{\# of unqualified applicants with M.Sc.}}$$

equals

$$\frac{\texttt{\# of accepted unqualified applicants with PhD}}{\texttt{\# of unqualified applicants with PhD}}.$$

Finally, equality of opportunity asks for the first condition of equalized odds.

For a fixed classifier $h$ denote by:

- $x$ the number of accepted qualified applicants with a M.Sc.
- $y$ the number of accepted unqualified applicants with a M.Sc.
- $z$ the number of accepted qualified applicants with a PhD.
- $t$ the number of accepted unqualified applicants with a PhD.

Then demographic parity requires that $\frac{x+y}{28} = \frac{z+t}{12}$. Equalized odds requires that $\frac{x}{16} = \frac{z}{8}$ and $\frac{y}{12} = \frac{t}{4}$. Finally, equality of opportunity requires that $\frac{x}{16} = \frac{z}{8}$ only.

a) Checking which constraints are satisfied by which classifier shows that:

- Demographic parity is satisfied by $h_1, h_4, h_7, h_8$
- Equalized odds is satisfied by $h_3, h_5, h_6, h_7, h_8$
- Equality of opportunity is satisfied by all classifiers that satisfy equalized odds and also $h_2$.

b) If a classifier satisfied both equalized odds and demographic parity, this will mean that $\frac{x}{16} = \frac{z}{8}$, $\frac{y}{12} = \frac{t}{4}$ and $\frac{x+y}{28} = \frac{z+t}{12}$. Therefore, $x = 2z$, $y = 3t$ and $3(x+y) = 7(z+t)$, so that $6z + 9t = 7z + 7t$, implying that $z = 2t$. Thus it must be that $(x, y, z, t) = (4t, 3t, 2t, t)$, with possible values of $t$ being $0, 1, 2, 3, 4$.

(c) One can use

$$\widehat{\Gamma}(h) = \left| \frac{\texttt{\# of accepted applicants with M.Sc.}}{\texttt{\# of applicants with M.Sc.}} - \frac{\texttt{\# of accepted applicants with PhD}}{\texttt{\# of applicants with PhD}} \right|.$$

Then $\widehat{\Gamma}(h_1) = \widehat{\Gamma}(h_4) = 0$. We also have $\widehat{\Gamma}(h_2) = |\frac{16}{28} - \frac{2}{12}| = \frac{17}{42}$ and $\widehat{\Gamma}(h_3) = |\frac{13}{28} - \frac{5}{12}| = \frac{1}{21}$.

(d) The corresponding population equivalent is

$$\Gamma(h) = \left| \mathbb{P}_{(X,A,Y)\sim\mathcal{D}}(h(X) = 1 | A = \texttt{M.Sc.}) - \mathbb{P}_{(X,A,Y)\sim\mathcal{D}}(h(X) = 1 | A = \texttt{PhD}) \right|.$$

Note that $\mathbb{P}(h(X) = 1 | A = \texttt{M.Sc.}) = \frac{\mathbb{P}(h(X)=1, A=\texttt{M.Sc.})}{\mathbb{P}(A=\texttt{M.Sc.})}$. As we get more and more samples from $\mathcal{D}$, by the law of large numbers we will have that

$$\frac{\texttt{\# of applicants with M.Sc.}}{\texttt{\# of all applicants}} \approx \mathbb{P}(A = \texttt{M.Sc.}).$$

Similarly,

$$\frac{\texttt{\# of accepted applicants with M.Sc.}}{\texttt{\# of all applicants}} \approx \mathbb{P}(h(X) = 1, A = \texttt{M.Sc.}).$$

Therefore, we expect that

$$\frac{\texttt{\# of accepted applicants with M.Sc.}}{\texttt{\# of applicants with M.Sc.}} \approx \frac{\mathbb{P}(h(X) = 1, A = \texttt{M.Sc.})}{\mathbb{P}(A = \texttt{M.Sc.})} = \mathbb{P}(h(X) = 1 | A = \texttt{M.Sc.}).$$

Similarly,

$$\frac{\texttt{\# of accepted applicants with PhD}}{\texttt{\# of applicants with PhD}} \approx \frac{\mathbb{P}(h(X) = 1, A = \texttt{PhD})}{\mathbb{P}(A = \texttt{PhD})} = \mathbb{P}(h(X) = 1 | A = \texttt{PhD}).$$

Therefore, we do expect $\widehat{\Gamma}(h) \approx \Gamma(h)$.

(e) One example distance is $D(x, x') = d(x.CV, (x').CV) + \lambda \mathbb{1}\{|x.age - (x').age| > 10\}$, for some weight $\lambda$. This ensures that similar applicants are exactly those that have similar CVs and whose ages differ by no more than 10 years, regardless of their gender.

## 2 Combining Logic and Deep Learning

**Problem 2** (Interpreting Queries)**.** Describe (in words) the objective of the following queries. Here, `ref` is an image from the test set, and `N`, $\texttt{N}_1$, $\texttt{N}_2$ are neural networks with two output classes 1 and 2 such that `class(N(ref)) = 1`.

1.
```
find i[10,10]
where i in [0,1],
      ‖i - ref‖∞ < 0.1,
      ‖i - ref‖∞ > 0.05,
      class(N(i)) = 2
```

2.
```
find i[10,10]
where i in [0,1],
      ‖i‖∞ < 0.2,
      class(N(i)) = 1
```

3.
```
find i[10,10]
where i in [0,1],
      ‖i - ref‖2 < 2,
      class(N1(i)) = 1,
      class(N2(i)) = 2
```

**Solution 2.**

1. This query looks for an *adversarial example* grayscale image of 10 by 10 pixels, which has bounded $\ell_\infty$-distance (upper and lower) from the reference image `ref` and is classified differently than `ref`.

2. This query looks for a dark (due to the $\ell_\infty$-norm constraint) grayscale image of 10 by 10 pixels which is classified as 1.

3. Here, we perform *differencing* of the networks $\mathtt{N}_1$ and $\mathtt{N}_2$: the query looks for a grayscale 10 by 10 pixels image that is similar to `ref` (in terms of $\ell_2$-norm) and classified differently by the two networks.

**Problem 3** (Translating Negations)**.** In this question, we will inspect how to support negation ($\neg$) within constraints.

1. Translate the following constraint $\varphi$ to a loss function using the rules discussed in the lecture. Here, `i` is a 2 by 2 pixel query image and `ref` is a given 2 by 2 pixel image from the test set.

   $\varphi$ := `(i[0,0] = ref[0,0] ∧ i[1,0] ≠ ref[1,0]) ∨`
   `(i[0,0] ≤ ref[0,0] ∧ i[1,0] < ref[1,0])`

2. Describe a way to transform a constraint involving negation ($\neg$) to a constraint that lies in the fragment discussed in the lecture (e.g., a constraint that only uses the operations described on lecture slide 20).

3. Transform the constraint $\neg\varphi$ to a constraint not involving negation.

**Solution 3.**

1. It is

$$\begin{aligned}
T(\varphi) &\overset{(\vee)}{=} T(\texttt{i[0,0]} = \texttt{ref[0,0]} \ \wedge \ \texttt{i[1,0]} \neq \texttt{ref[1,0]}) \cdot \\
&\qquad T(\texttt{i[0,0]} \leq \texttt{ref[0,0]} \ \wedge \ \texttt{i[1,0]} < \texttt{ref[1,0]}) \\
&\overset{(\wedge)}{=} (\underbrace{T(\texttt{i[0,0]} = \texttt{ref[0,0]})}_{t_1} + \underbrace{T(\texttt{i[1,0]} \neq \texttt{ref[1,0]})}_{t_2}) \cdot \\
&\qquad (\underbrace{T(\texttt{i[0,0]} \leq \texttt{ref[0,0]})}_{t_3} + \underbrace{T(\texttt{i[1,0]} < \texttt{ref[1,0]})}_{t_4}) \\
&= (t_1 + t_2) \cdot (t_3 + t_4),
\end{aligned}$$

   where

$$\begin{aligned}
t_1 &\overset{(=)}{=} T(\texttt{i[0,0]} \leq \texttt{ref[0,0]} \ \wedge \ \texttt{ref[0,0]} \leq \texttt{i[0,0]}) \\
&\overset{(\wedge)}{=} T(\texttt{i[0,0]} \leq \texttt{ref[0,0]}) \ + \ T(\texttt{ref[0,0]} \leq \texttt{i[0,0]}) \\
&\overset{(\leq)}{=} \max(0, \ \texttt{i[0,0]} - \texttt{ref[0,0]}) \ + \ \max(0, \ \texttt{ref[0,0]} - \texttt{i[0,0]})
\end{aligned}$$

$$t_2 \overset{(\neq)}{=} [\texttt{i[1,0]} = \texttt{ref[1,0]}]$$

$$t_3 \overset{(\leq)}{=} \max(0, \ \texttt{i[0,0]} - \texttt{ref[0,0]})$$

$$\begin{aligned}
t_4 &\overset{(\leq)}{=} T(\texttt{i[1,0]} \leq \texttt{ref[1,0]} \ \wedge \ \texttt{i[1,0]} \neq \texttt{ref[1,0]}) \\
&\overset{(\wedge)}{=} T(\texttt{i[1,0]} \leq \texttt{ref[1,0]}) \ + \ T(\texttt{i[1,0]} \neq \texttt{ref[1,0]}) \\
&\overset{(\leq,\neq)}{=} \max(0, \ \texttt{i[1,0]} - \texttt{ref[1,0]}) \ + \ [\texttt{i[1,0]} = \texttt{ref[1,0]}].
\end{aligned}$$

2. Constraints involving negations can be transformed to the desired fragment by "pushing" negation $\neg$ down to the leafs such that the resulting constraint does not involve any negations (note that $\neq$ is not a negation).

   In particular, one can recursively re-write conjunctions and disjunctions using De Morgan's laws: $\neg(\varphi \wedge \psi)$ is equivalent to $\neg\varphi \vee \neg\psi$, and $\neg(\varphi \vee \psi)$ is equivalent to $\neg\varphi \wedge \neg\psi$. The negation of atomic constraints can be re-written to equivalent constraints not involving negation: for example, $\neg(x \leq y)$ is equivalent to $y < x$.

3. We can re-write the constraint to get rid of the negation as follows:

$$\begin{aligned}
\neg\phi \;&= \neg\big((\texttt{i[0,0]} = \texttt{ref[0,0]} \;\wedge\; \texttt{i[1,0]} \neq \texttt{ref[1,0]}) \vee \\
&\qquad (\texttt{i[0,0]} \leq \texttt{ref[0,0]} \;\wedge\; \texttt{i[1,0]} < \texttt{ref[1,0]})\big) \\
&= \neg(\texttt{i[0,0]} = \texttt{ref[0,0]} \;\wedge\; \texttt{i[1,0]} \neq \texttt{ref[1,0]}) \wedge \\
&\qquad \neg(\texttt{i[0,0]} \leq \texttt{ref[0,0]}) \;\wedge\; \texttt{i[1,0]} < \texttt{ref[1,0]}) \\
&= \big(\neg(\texttt{i[0,0]} = \texttt{ref[0,0]}) \;\vee\; \neg(\texttt{i[1,0]} \neq \texttt{ref[1,0]})\big) \wedge \\
&\qquad \big(\neg(\texttt{i[0,0]} \leq \texttt{ref[0,0]}) \;\vee\; \neg(\texttt{i[1,0]} < \texttt{ref[1,0]})\big) \\
&= (\texttt{i[0,0]} \neq \texttt{ref[0,0]} \;\vee\; \texttt{i[1,0]} = \texttt{ref[1,0]}) \wedge \\
&\qquad (\texttt{i[0,0]} > \texttt{ref[0,0]} \;\vee\; \texttt{i[1,0]} \geq \texttt{ref[1,0]})
\end{aligned}$$

**Problem 4** (Alternative Translation)**.** In the lecture, we studied one particular way to translate constraints to nonnegative loss functions. Consider the following alternative translation $T$, which also produces nonnegative loss functions:

| $\omega$ | $T(\omega)$ |
|---|---|
| $t_1 = t_2$ | $(t_1 - t_2)^2$ |
| $t_1 \leq t_2$ | $\max(\mathrm{sgn}(t_1 - t_2) \cdot (t_1 - t_2)^2,\ 0)$ |
| $\phi \vee \psi$ | $T(\phi) \cdot T(\psi)$ |
| $\phi \wedge \psi$ | $T(\phi) + T(\psi)$ |

Further, consider the formula

$$\psi := (\mathrm{ReLU}(x_1 + 2x_2) = x_3 \;\wedge\; x_3 \leq 4) \;\vee\; (x_3 \leq 0 \;\wedge\; x_1 + x_2 \geq 0),$$

which has free variables $x_1, x_2, x_3$. We denote the set of free variables as $\mathbf{x}$, and the assignment to these variables $x_1 \leftarrow y_1, \ldots, x_3 \leftarrow y_3$ as $\mathbf{y}$. The translation of $\psi$ according to $T$ is denoted $T(\psi)$ and the numerical value of the translation evaluated for assignment $\mathbf{y}$ is indicated by $T(\psi)(\mathbf{x} \leftarrow \mathbf{y})$.

1. Derive the translation $T(\psi)$ of formula $\psi$.

2. Prove that for any assignment $\mathbf{y}$, $T(\psi)(\mathbf{x} \leftarrow \mathbf{y}) = 0$ implies that $\mathbf{y}$ is a satisfying assignment of $\psi$.

**Solution 4.**

1. The formula $\psi$ is transformed as follows:

$$(\underbrace{\text{ReLU}(x_1 + 2 \cdot x_2) = x_3}_{\varphi_1} \wedge \underbrace{x_3 \leq 4}_{\varphi_2}) \vee (\underbrace{x_3 \leq 0}_{\varphi_3} \wedge \underbrace{x_1 + x_2 \geq 0}_{\varphi_4})$$

$$t_1 := T(\varphi_1) = (\text{ReLU}(x_1 + 2 \cdot x_2) - x_3)^2$$

$$t_2 := T(\varphi_2) = \max(\text{sgn}(x_3 - 4) \cdot (x_3 - 4)^2,\ 0)$$

$$t_3 := T(\varphi_3) = \max(\text{sgn}(x_3) \cdot (x_3)^2,\ 0)$$

$$t_4 := T(\varphi_4) = \max(\text{sgn}(-x_1 - x_2) \cdot (x_1 + x_2)^2,\ 0)$$

$$T(\varphi) = (T(\varphi_1) + T(\varphi_2)) \cdot (T(\varphi_3) + T(\varphi_4)) = (t_1 + t_2) \cdot (t_3 + t_4)$$

2. We prove the claim by structural induction over any formula $\omega$ involving only equality ($=$), inequality ($\leq$), disjunction ($\vee$) and conjunction ($\wedge$). Because $\psi$ is an instance of such a formula, the claim also holds for $\psi$.

   **Base case**
   *Equality ($\omega$ has the form $t_1 = t_2$).* It is $T(\omega) = 0 \iff (t_1 - t_2)^2 = 0 \implies t_1 - t_2 = 0 \implies t_1 = t_2$, meaning that $\omega$ is satisfied.

   *Inequality ($\omega$ has the form $t_1 \leq t_2$).* Assume $T(\omega) = 0$, which is equivalent to $\max(\text{sgn}(t_1 - t_2) \cdot (t_1 - t_2)^2, 0) = 0$. This means that $\text{sgn}(t_1 - t_2) \cdot (t_1 - t_2)^2$ is either zero or less than zero. In the first case, $\text{sgn}(t_1 - t_2) \cdot (t_1 - t_2)^2 = 0 \iff t_1 - t_2 = 0 \iff t_1 = t_2$. In the second case, we know that since $(t_1 - t_2)^2$ is always non-negative, $\text{sgn}(t_1 - t_2)$ must be negative and thus $t_1 < t_2$. Therefore, it is guaranteed that $t_1 \leq t_2$, meaning that $\omega$ is satisfied.

   **Step case**
   As induction hypothesis, assume that the claim holds for two arbitrary formulae $\phi$ and $\psi$ (i.e., $T(\phi) = 0$ implies that $\phi$ is satisfied, and similarly for $\psi$).

   *Disjunction ($\omega$ has the form $\phi \vee \psi$).* If $T(\omega) = T(\phi) \cdot T(\psi) = 0$, then either $T(\phi) = 0$ or $T(\psi) = 0$. By the induction hypothesis, either $\phi$ is satisfied or $\psi$ is satisfied, implying that $\phi \vee \psi$ is satisfied.

   *Conjunction ($\omega$ has the form $\phi \wedge \psi$).* If $T(\omega) = T(\phi) + T(\psi) = 0$, then it must be both $T(\phi) = 0$ and $T(\psi) = 0$ (for any $\omega'$, $T(\omega')$ is nonnegative). By the induction hypothesis, $\phi$ and $\psi$ are satisfied, implying that $\phi \wedge \psi$ is satisfied. □