CSAI - Tutorial 2 (07 March 2024)

This week we will be looking at AI Fairness 360 toolkit developed by IBM Research Trusted AI. There are many resources that you can have a look to learn more about the toolkit, we provide some further tutorials in the last section.

We will specifically be looking at one example, which is Credit scoring example. This task is looking at detecting and mitigating age bias on decisions to offer credit using the German Credit dataset. This is a practical session, you can bring your laptops and try the examples yourself; or your tutor will guide you through each of the steps during the session.

Dataset

Some general information about the dataset can be found here. The raw dataset can be downloaded from German Credit dataset. You can open the file (GermanData.xlsx) in Microsoft Excel to check how it looks like. There are 1000 records in total, and each record is represented with 20 attributes (7 numerical or 13 categorical). All these attributes are used to judge a loan applicant. The goal is to classify the applicant into one of two categories, favorable (e.g. 1) or unfavorable (e.g. 2), which is the last attribute.

Part 1: AIF360 Tutorial

The original documentation for the *Credit Scoring* tutorial is provided in this repository. There are two ways to run this tutorial: 1. You can clone the repository and try to run locally, 2. you can use Watson Studio to experiment with the Jupyter notebook (we did not try this option). During the tutorial, we will try to work with the first option and see how things work in practice. Some commands were not working when we did give this a try so we also prepared a CSAI version of the Jupyter notebook, you can download it from here in case you want to experiment yourself.

The basic idea is to define 'age' attribute as a protected attribute, and then to make sure that no bias exist in the training data. A fairness metric (statistical parity) is used to do this check, which shows that some bias exist. A **statistical parity** entails that individuals from the protected and unprotected groups should have the same probability of being assigned the favorable label. Then, a mitigation strategy (Reweighing method) is applied to overcome this issue.

Part 2: Compute Statistical Parity yourself

Before we start this step, we will ensure that *shuffle* option is set to *False* in the Jupyter notebook. If you run the same analysis, you should find -0.118448637 as the difference in mean outcomes between privileged and unprivileged groups. Next, do the following steps:

1. If you are using Excel for this tutorial, download the German Credit dataset. If you are using Jupyter notebook provided, we will handle the dataset part in the code.
2. Use the first 700 rows of the data as your training data.
3. 13th column in the data (i.e. column M in Excel) is the Age column. Replace all age values above or equal to 25 with 1, and the rest with 0.
4. Compute statistical parity.
5. You should compute the value of -0.118448637. Remember that favorable (1)/unfavorable (2) outcome is defined in the last column of the data.

If you use Excel, some useful hints

- You can create one column for the privileged group in case of favorable label. For example, $= IF(AND(M1 = 1, U1 = 1), 1, 0)$ formula assigns 1 if M1 is equal to 1, and U1 is equal to 1. You can apply this to all rows.
- You can create one column for the unprivileged group in case of favorable label. For example, $= IF(AND(M1 = 0, U1 = 1), 1, 0)$ formula assigns 1 if M1 is equal to 0, and U1 is equal to 1. You can apply this to all rows.
- SUM is another function you can use to identify the size of the privileged group (by using column M). Then, you can subtract it from 700 to compute the size of the unprivileged group. You will also use this function to compute the number of positive cases in each group.
- The rest is a simple probability computation!

Optional: Tutorials on Youtube

There are two tutorials on Youtube about this toolkit that you can have a look, if you are interested.

- A tutorial by Prasanna Sattigeri,
- A tutorial by Rachel K. E. Bellamy, Michael Hind, Karthikeyan Natesan Ramamurthy, Kush R. Varshney, presented as part of FAT*2019.