Technologies for Information Systems project on Data Ethics - detailed Report

Measure fairness in healtcare-related database using the **Aequitas** toolkit

Simone Callegarin

April 2023

POLITECNICO
MILANO 1863

Reference professors: Letizia Tanca, Chiara Criscuolo and Tommaso Dolci

The practical project on Data Ethics consisted in the creation and presentation of a python notebook.
The assignment required the use of the **Aequitas toolkit** to measure fairness in an healthcare-related dataset that in my case concerns **diabetes**.
The content of the **notebook** can be divided into eight main **sections**:

1. Setup of the Development Environment
2. Audit Configuration
3. Dataset Description and Analysis
4. Predictive Algorithm (Random Forest Classifier)
5. Data Preprocessing (+ Upload)
6. Predicted Data Exploration
7. Auditing Fairness of the Model using Aequitas
8. Bibliography

1 Setup of the Development Environment

In the first part are indeed present all the various imports of the packages and modules that are used further on in the notebook. Here it is also imported the `SCdataframeFormatter`, a package I developed and published through `PyPi` that contains useful methods to print dataframes in particular formats in order to facilitate their comprehension.

2 Audit Configuration

This section is responsible for the download of the dataset that has to be audited and for the choice of the sensitive attribute (through a dropdown menu of the `ipywidgets` pack). The Audit Configuration section includes also a cell dedicated to store the API token of the GitHub repository (passed through a `json` file) that is later used to upload the processed Data file encoded in `base64` directly into the GitHub repository through the use of the `requests` library.

3 Dataset

The Dataset section includes the references of the dataset, the possibility to explore it, a comprehensive description of the dataset and a preliminary analysis of data correlation. The analysis and description phases involves libraries such as `Seaborn` and `MatPlotLib` in particular for graphs plotting.
The dataset counts 763 tuples with 9 attributes related to the entities' physical parameters and the *Outcome* value that labels individuals as diabetic.

4 Predictive Algorithm

The predictive algorithm used is the `Random Forest Classifier` that is imported from `sklearn` library that also offers tools to handle Data training and test splits.

5 Data Preprocessing (+ Upload)

Data obtained from the predictive algorithm undergoes a preprocessing phase that will lead to the creation of a dataframe containing entities from the test set and the related algorithm's predictions.
The section is also in charge of uploading the `csv` file with the new preprocessed data into the github repository expoliting the `requests` module to send a PUT request to the repository.

6 Predicted Data Exploration

A brief preliminary analisys on the distribution of the *label_value* and the *score* on the sensitive attribute combined with graphs.

7 Auditing Fairness of the Model using Aequitas

The section begins by defining:

- the attributes to audit
- the reference group for each attribute (majority groups)
- the fairness metric(s) that we care about (`FNR, TPR`)
- the disparity tolerance ($\tau = 1.25$)

In our analysis the sensitive attribute was the *AgeCategory* and as presented by the authors of Aequitas[5], the interested fairness metrics were the `FNR` and the `TPR` due to the fact that the interventions made by our predictive model are assistive and intervening with a big part of the population[**?**].
The *false negative rate* (`FNR`) is the fraction of false negatives of a group within the labeled positives of the group:

$$FNR_g = \frac{FN_g}{LP_g} = Pr(\hat{Y} = 0 | Y = 1, A = a_i)$$

The *true positive rate* (`TPR`) is the fraction of true positive of a group within the labeled positives of the group:

$$TPR_g = \frac{TP_g}{LP_g} = Pr(\hat{Y} = 1 | Y = 1, A = a_i)$$

Speaking in terms of parity, the *true positive rate parity* is also called equality of opportunity and assert that the protected and unprotected groups should have equal true positive rates (i.e. same rate of positive outcome, assuming the people in the groups qualify for the outcome).
The *disparity tolerance treshold* controls the range of disparity values that can be considered fair.
Our formulation involves the *"80% rule"* represented by $\tau = 0.8$.
This notion of parity requires all biases to be within the range defined by τ:

$$\tau \leq DisparityMeasure_{group_i} \leq \frac{1}{\tau}$$

$$0.8 \leq DisparityMeasure_{group_i} \leq \frac{1}{0.8} = 1.25$$

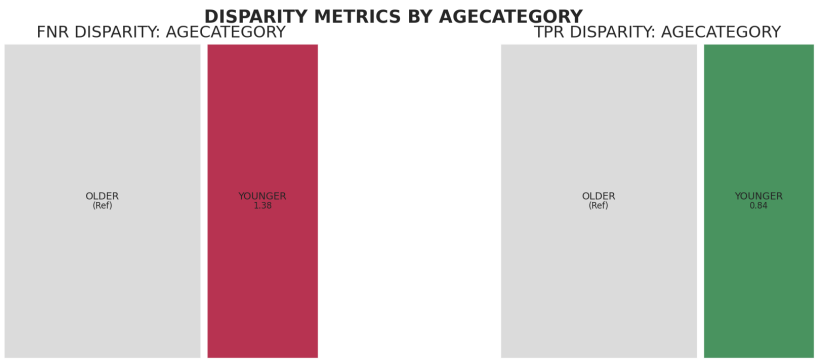
The main tool used to audit the fairness of the dataset is `Aequitas`, that offers 4 main classes:

1. `Group`
2. `Bias`
3. `Fairness`
4. `Plotting/Plot`

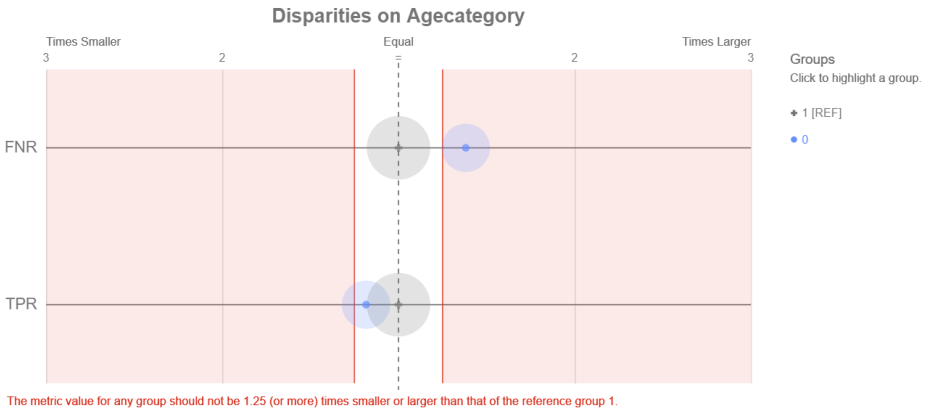
Applying Aequitas programmatically is a three step process represented by the first three python classes.
Here infact `Aequitas` is used to first of all define groups from the attributes contained in the dataset, to calculate disparities on these groups and to assert the fairness of some famous metrics over these groups.
As stated in the paper *"Aequitas: A Bias and Fairness Audit Toolkit"*[5]:

> A traditional binary classification task using supervised learning consists of learning a predictor $\hat{Y} \in \{0, 1\}$, that aims to predict the true outcome $Y \in \{0, 1\}$ of a given data point from the set of features X, based on labeled training data.

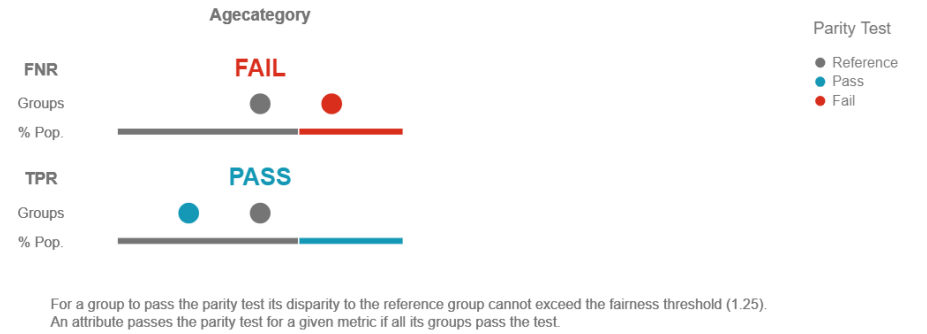
The analysis consist in measuring bias and fairness between the true *outcome* that is the true binary label of a given entity and the *score* assigned to each entity by the predictor.
The initial audit is performed on the whole dataframe that will be sliced for further analyses on the sensitive attribute.
At the end the `Plotting` library has been used to support the disparities evaluation through the use of different graphs focused on each metric of interest.



The analisys on results highlighted that our model is biased for the *AgeCategory* attribute for the *False Negative Rate* (`FNR`), whilst it isn't for the *True Positive Rate* (`TPR`).



This is determined by the fact that the False Negative Rate (FNR) of the YOUNGER group exceed the disparity tolerance treshold, highlighting an unfairness in our model.
On the contrary the True Positive Rate (TPR) shows that the YOUNGER group does not exceed the disparity tolerance treshold, revealing that the predictive model is unbiased over this measure.



Basically *the model tends to indicate that people actually with diabetes are more likely to be labeled negative at a younger age.*

8 Bibliography

A short bibliography has been provided with all the resources that helped the writing of the notebook (those are listed in the References of this report).

References

[1] Kaggle - diabetes dataset

[2] OpenML - Diabetes-130-Hospitals dataset

[3] ScikitLearn - Random Forest Classifier

[4] Aequitas

[5] Aequitas - paper

[6] Aequitas - textbook

[7] Aequitas - tools guide

[8] Aequitas - tutorial python notebook

[9] Aequitas - tutorial

[10] PyPi

[11] GitHub

[12] StackOverflow

[13] Pandas

[14] Fairness in ML

[15] Confusion metrics

[16] Python projects packaging

[17] Hex calculator

[18] Tkinter colors

[19] SCdataframeFormatter