Algorithmic Fairness, Accountability, and Ethics, Spring 2023, IT University of Copenhagen

Mandatory Assignment 2

Germans Savcisens , Camilla Jung Westermann

- **Hand-out**: March 20, 2023
- **Hand-in**: April 11, 2023 at 23:59
- **What to hand-in:** A report as pdf summarizing the main findings, **max. 3 pages**, including plots. A jupyter notebook detailing the process. Upload the two files as a single zip file on learnIT.
- **Where to start:** You can find a template to get started in the assignment on learnIT.
- **Dataset:** US Census data from `https://github.com/zykls/folktables`. We use data of individuals from the state California in 2018, as detailed in the template. The template also details which attributes we use as feature vector. More details on the dataset can be found in the accompanying paper at https://arxiv.org/pdf/2108.04884.pdf or (Folktables Git)[https://github.com/socialfoundations/folktables].

Context:

1. We are going to work with two protected attributes: `SEX` and `RAC1P` (we are going to limit the datasets to `Whites` and `African-Americans`).
2. We have a binary target variable (Total Person's Income, aka `PINCP_TRG`), where the positive label stands for the income above 25 000 USD.
3. We are going to use only **500 samples**.

Task 1 (Bias Analysis)

Task 1.1.: Data Collection and Representation

Let's look at the following attributes: `AGEP`, `RAC1P`, `SEX`, `SCHL`, `CIT`, `COW`. Do *not* use one hot encoded variables for **Task 1**.

1. Discuss sources of bias in the dataset and in the selected features.
2. Cover the following aspect: `Training Data` (refer to the **Lecture 5 Slide #56: How to handle bias?**).

Task 1.2.: Proxies

Look at the relationships between `SEX` vs `AGEP`, `RAC1P`, `PINCP` (or `PINCP_TRG`), `CIT`, `COW`, `MAR` and `WKHP`.

1. Look at feature distributions if you stratify them by `SEX` groups. Do you see any potential sources of bias? Provide arguments.
2. Look at the correlations between `SEX` and other variables.
3. Cover the following aspect: `Proxies` (refer to the *Lecture 5 Slide #56: How to handle bias?*).
4. Supplement your answer with several visualisations. e.g distributions of variables per protected group (*you do not have to provide all of them, but the ones you find interesting*).
5. When discussing correlations do not forget to use the correct metric (e.g. continuous-categorical features etc.). You can use `dython.nominal.associations` and `seaborn.heatmap`.

Note: While discussing bias use definitions/terms described in `A Survey on Bias and Fairness in Machine Learning` and `Social Data: Biases, Methodological Pitfalls, and Ethical Boundaries` (see Lecture 5: Reading Materials).

Task 2 (Model & Data Debiasing)

Now we are going to train a model to predict the income of a person based on the attributes we have at hand. We want to have a model with the high predictive performance, but we also want to make sure that our model does not discriminate against any protected groups.

Task 2.1.: Data

1. Convert categorical to one-hot encoding.
2. Remove redundant categorical columns (as you have done in Lecture 6).
3. Remove protected attributes from the data (keep it aside).
4. Split data into `Training` and `Test` sets.

Task 2.2.: Baseline Model

1. Build your own implementation of the Logistic Regression with L2-penalty (aka Ridge Regression).
    - Do not forget to add the column of ones for the intercept, β_0 (when calculating cost and/or gradient).
    - Do not penalise β_0 (when calculating L2-penalty).
    - You can use `approx_grad=True` or your implementation of `compute_grad`.

2. Use Cross-Validation to find the most optimal value for L2-penalty (you should implement it yourself).
3. Evaluate the overall performance of the final model on the Test Set (use an appropriate metrics) + report uncertainty.
4. Look at the fairness metric associated with each `SEX` and `RAC1P` groups. Are there any discrepancies?

Task 2.3.: Model with the Fair Penalty

1. Add **Individual Fairness Penalty** to your baseline model (refer to the *Lecture 5 Exercises*).
   - Use L2-penalty coefficient from **Task 2.2.**.
   - Do not forget to add the column of ones for the intercept, β_0.
   - Do not penalise β_0 (when calculating L2-penalty).
   - Use `approx_grad=True` to approximate the gradient.
   - **Remember** that you have two protected features! Thus, you need to add one fairness constraint per feature.
2. Plot *Pareto Curve* by varying $\lambda = [1e-3, 5e-3, 1e-2, 5e-2, 0.1, 1]$, evaluate the performance of the model (using your favourite metric). Plot a curve for each group of protected attributes (i.e. 4 curves). What happens as we increase the penalty? Is there a point where all groups get similar performance metric values?
3. Set $\lambda = 0.1$, evaluate the overall performance of the final model on the Test Set (report uncertainty). Use same metric as you used in **Task 2.2**.
4. Set $\lambda = 0.1$, look at the fairness metric associated with each `SEX` and `RAC1P` groups. What do you see (compare results to the baseline model)?

Task 2.4.: Fair PCA

We are going to implement the method from Efficient fair PCA for fair representation learning. Here, we will use dimensionality reduction to remove any existing proxies associated with the protected features (refer to the *Lecture 6 Exercises*).

1. Use `Standard PCA` on non-protected features.
   - Do not forget to normalise data before applying `Fair PCA`.
   - Use $N_{components} = N_{features} - N_{protected\,groups}$, you have 4 protected groups.
   - Look at the correlations between the new dimensions and original protected features (use either *Pearson's* or *Spearman's* coefficient). What do you see?

2. Project your test data with `Standard PCA`, and then project it back into the original space.
    - Calculate the reconstruction error for each sample.
    - Look at the reconstruction error per each protected group. What do you see?
3. Implement `Fair PCA` (refer to Lecture 7 Exercises).
    - Do not forget to normalise data before applying `Fair PCA`.
    - Use $N_{components} = N_{features} - N_{protected\,groups}$, you have 4 protected groups).
    - Look at the correlations between the new dimensions and original protected features. How does results compare to the `Standard PCA`?
4. Project your test data with `Fair PCA`, and then project it back into the original space.
    - Calculate the reconstruction error for each sample.
    - Look at the reconstruction error per each protected group. What do you see? Are there any differences compared to the `Standard PCA`?

Task 2.5: Logistic Regression and Fair PCA

1. Fit a Logistic Regression (your implementation) to the debiased data (via `Fair PCA`).
    - Do not use L2-penalisation
    - Do not forget to add the column of ones for the intercept, β_0.
2. Evaluate the overall performance of the final model on the Test Set (report uncertainty).
3. Look at the fairness metric associated with each `SEX` and `RAC1P` groups. Are there any discrepancies?

Task 3 (Robustness and Evaluation)

Task 3.1.: Model Robustness

Let's assume you decided to sell your model to the Bank. The bank told you it wants to infer how much money people earn before they become clients (so that they can recommend personalised services to clients). Using your knowledge from Lecture 8, discuss the following:

1. Mention 2 things you would do to ensure that your model is *reliable*.
2. Mention 2 things you would do to ensure that you model is *robust*.
3. Mention 2 things you would do to report your results

Note: To answer these questions refer to `How to avoid machine learning pitfalls` (see *Lecture 8: Reading Materials*).

Task 3.2.: Evaluation of the models

- Given the outcome of your study, which classifier is most suited for the prediction task under predictive performance, and fairness considerations (for this particular dataset)?
- Name two advantages and disadvantages of each method you used (i.e. simply dropping the protected attributes, fair regression and fair PCA).

Checklist

We will add the checklist by the end of the week.