Ehsan Toreini
COMP2671
ehsan.toreini@durham.ac.uk

Coursework Assignment
Bias in Artificial Intelligence - 2022

deadline: 05, May, 2022

1 Summary of the coursework

In this coursework, You will be analysing a human-centric [1] dataset and develop a fair machine learning ecosystem to detect, reduce and eventually mitigate different types of bias that exist in the final outcome of the algorithm in various ways.
This coursework involves two parts: (1) a 3–page project report (2) code implementation, discussed in the following two sections accordingly. The deadline for submission is on May 5th, 2022. You will submit all the deliverables in a single compressed file (preferably .zip format). If there are any queries regarding this coursework, please do not hesitate to contact me: ehsan.toreini@durham.ac.uk

2 Project Outline

In this project, you are responsible to design a automated system to help the HR department of an ACME organisation. This system shortlists applicants for interview and proposes a decision on whether or not an specific candidate should be given an offer. Typically, this system will be a classifier that is trained based on the historic data available through previous experiences of the hiring committees in the organisation. As the ML engineer, you are given the dataset, now you are responsible to design a non-biased system (unlike what Amazon designed in 2016 which led to a publicity scandal and eventually, Amazon decided the decease the system [2]).
Please note that the implementation and outcome of each task should be clearly separated in your submission (in both project report and source code). You will submit your implementation in a Jupyter notebook (ipynb format) so clearly segment your notebook and make sure it is in a presentable format. Also, the code should be sufficiently commented and *obviously, should be your own implementation*. *Note: the implementation of the method should be yours. You cannot use any "fair AI" python package in your implementation, i.e. IBM AIF360 or similar packages introduced in the class, you can compare your results with such systems offline to make sure your implementation works correctly though.*
You should submit the answers to the questions proposed here in a separate PDF file in your submission. The style of the analysis should be technical, rather than verbose. This should be understandable by someone with a good knowledge of the bias mitigation techniques. Be concise and straight to the point. Make sure your answer to these questions do not exceed 3 A4 pages, including the citations.

2.1 Task 1: Dataset Analysis [40 Marks]

Download the dataset from the Dataset folder on blackboard. The description of the dataset can be also found in the same folder. Read the relevant documentations and answers the following tasks in your project report document (in the document, clearly specify the answers for each task). Include the implementation tasks in your Jupyter notebook. The data set that we use is *recruitment.xls* [3]. The applicant data set includes the following information within nine variables:

- ApplicantCode (applicant code).
- Gender (1 = male or 2 = female).
- BAMEyn (Black, Asian or Minority Ethnic: 1 = yes or 2 = no)
- ShortlistedNY (0 = rejected or 1 = shortlisted).
- Interviewed (0 = not interviewed or 1 = interviewed).
- FemaleONpanel(1=male only panel or 2=female member on panel).
- OfferNY (1= made an offer or 0 = not offered).

[1] A dataset such that each entry represents one measurement of one person.
[2] https://www.reuters.com/article/us-amazon-com-jobs-automation-insight-idUSKCN1MK08G
[3] available in the coursework folder on Blackboard

- AcceptNY (1 = Accepted or 0 = declined).
- JoinYN (1 = joined or 0 = not joined).

As the variable and value labels indicate, the data set indicates the gender ('Gender' - variable 2) of each person that sent in an application for the graduate job as well as whether or not they were Black, Asian or Minority Ethnic ('BAMEyn' - variable 3). Importantly the 'ShortlistedNY' variable indicates whether, after an initial review of their application, they were considered to be an appropriate candidate for interview (in other words, considered potentially employable). The 'Interviewed' variable indicates whether they were interviewed or not, the 'FemaleONPanel' variable indicates whether there was a female interviewer included on the interview panel. Then a key variable here is whether the applicant was offered a job or not ('OfferNY') and the 'AcceptNY' variable indicates whether they accepted the offer. Finally the 'JoinYN' variable indicates whether the applicant joined the organization.

1. What is the sensitive attribute in the dataset? Clarify what group is privileged and what group is unprivileged? How did you determine it? Describe your answers briefly. (5 Marks)
2. What are the statistics relevant to the privileged and unprivileged groups? (e.g. average, standard deviation, variance, and more). Clearly demonstrate the above numbers in a table in your project report. (5 Marks)
3. Demonstrate the statistical disparity of privileged and unprivileged groups in two scenarios: (1) being invited to the interview (2) being offered a job. Show the disparity analysis results in tables for each of a privileged and unprivileged combinations (i.e. the privileged vs. only one of the unprivileged groups in each table) (10 Marks)
4. Statistically prove the dataset is biased towards the privileged group in *shortlisting process* (i.e. determine the hypothesis for the scenario, calculate the p–value and conclude if you accept the hypothesis or not). Explain the procedure in your project document. (10 Marks)
5. Implement the above proof in your source code (i.e. compute the p-value and determine the hypothesis conclusion accordingly). *Note: It is fine to use an extra statistical package in your source code, but clearly explain its usage in your project report*. (10 Marks)

2.2 Task 2 - Adversarial Bias Mitigation [60 Marks]

Following the conclusion Task 1, you now must know if your dataset is biased or not. Now, you will be responsible to design an adversarial bias mitigation algorithm to mitigate any potential bias in the decision *making a job offer*. You should carefully design the adversarial part of your algorithm based on specific fairness definition. Discuss your findings in the project report. Thus, this task has two part, (1) the regular classifier implementation and (2) adversarial implementation. Clearly specify the parts in your source code. Answer the requested questions in your project report document.
Please follow these steps:

1. Make sure the "regular classification algorithm" works correctly (without the adversarial part). Randomly divide the dataset into 70% training set and 30% test set. Train your model and see how it generalises to the testing dataset (no over-fitting or under–fitting). Explain what you did in the project report. (15 Marks)
2. Demonstrate the results of your trained regular classifier model in output (explicitly report the accuracy rate, you can demonstrate more performance–related metrics). Clearly mention this results in your project report. Do you see any bias in the results? explain your answer. (5 Marks)
3. Implement the adversarial learning component. Clearly explain the fairness notion you chose and your reasons in the project report. (20 Marks)
4. Demonstrate the results (i.e. accuracy and other metrics you chose to include in step 2) after you implemented the adversarial component. Clearly explain how the results changed and whether your chosen notion of fairness is satisfied or not in your project report. (15 Marks)
5. In summary, explain the results of your bias mitigation strategy in your project report (Report a fairness metric and demonstrate how it changes before and after your implementation). (5 Marks)

3 Timeline and Project Marking Guidelines

3.1 Project Timeline

The deadline for submission is on May 5th, 2022. You will submit the two projects in one single compressed file (preferably in zip format). Use your student ID as the name of the zip file. The project report should be in pdf file format. Your final project submission should have the following items:

- Project report, in PDF file format (page limit, 3 A4 pages including the citations)
- Implementation coursework in ipynb file format

3.2 Marking Guideline

Each task will be marked based on the following criteria.
Reported results

- Correctness of the statistical analysis: your reported numbers should be correct and clear.
- Presentation. the reported results should be as instructed. For instance, if it is mentioned to use tables, you will lose mark if you don‘t.
- Clarity of the analysis: The style of the analysis should be technical, rather than verbose. This should be understandable by someone with a good knowledge of the bias mitigation techniques.
- Quality of the writing and citations will be based on the following criteria:
  - overall quality of writing (as below):
    * evidence of adequate and appropriate background reading
    * a clear statement of aims and relevant selection of content
    * sensible planning and organization
    * evidence of systematic thought and argument
    * clarity of expression
    * careful presentation (e.g. accurate typing and proof-reading, helpful diagrams, etc.)
    * observation of conventions of academic discourse, including bibliographic information
    * observation of length requirements
  - clarity of the writing
  - Does the discussions for each task match the source code results?
  - Justifications of implementation choices and fairness criteria.

Implementation If the task includes implementation, the code will be marked as follows:

- Implementation
  - Does your project work?
  - Correct implementation of fair ML solutions.
  - How effective and considerate of various biases is your dataset sampling strategy?
  - How well does your model generalise? Is it over–fitting or under–fitting?
  - The final project should mitigate existing bias in the dataset while maintaining an acceptable level of accuracy
- Sophistication and appropriateness of the solution
  - How well have you applied the relevant theory to the problem?
  - How hackish is your implementation, or is it robust and well-designed?
  - Have you just cited and pasted code, or is their evidence of comprehension with further study and novel design extending beyond the lecture materials?

4 Frequently Asked Questions

It is strongly recommended to read these common questions and answers carefully.

I found code online which looks similar to what I need. Can I use it?

Yes, but you must cite the code in both the written report and in the comments at the top of the code. As a common practice in any software development, you first try to search and make sure you are not the first one who is trying to make it work. However, it is one of my tasks to make sure you are doing something original. So, please adapt the code and make sure you have cited it. Otherwise, it is very likely that you get caught (see the sub-mission Plagiarism and Collusion section on DUO to read about the tools used to detect this). This incurs a very severe departmental penalty.

Isn't the best strategy to just copy the state-of-the-art? Yes, if you notice from the list of suggested projects, you are already working on the state–of–the–art solutions in fair AI. If you know the literature, you will find the most reputable research in the field of fairness and algorithmic bias belongs to a conference called (ACM FAT, which recently got rebranded as ACM FAccT). So, keep looking in their accepted paper list to find the most recent developments in the field.

I'm struggling and feeling overwhelmed by all of this. The maths is too complicated and I don't know where to begin.

Try to read the paper for your suggested project carefully. It must contain a lot of implementation detail that you might have neglected. There might be a sentence somewhere in the paper that inspires you to have another brilliant idea! If you get errors, read them slowly, Google them. When you're confident enough, try to implement something a little bit more complicated. Do a slow, step–by–step implementation approach.

My writing is not as good, will it make my essay mark automatically low?

Not really, I understand for the majority of students (including myself), English is not their first language. Therefore, instead of focusing on using complicated words, try to stay as simple as you can in your writing. My first advice to everyone is to write simple. Avoid complicated sentences, words or grammatical combinations. Focus on the quality of discussions rather than making your essay look fancy. I would also recommend using online grammar check tools (such as Grammarly) to polish any mistakes. Also, you can always borrow technical words or some discussions from the papers you want to cite (just remember to cite them, then rephrase them in your wording to avoid plagiarism).

Can I use deep neural networks or just classical machine learning models?

I recommend you to follow the footsteps of the research paper you chose to implement.

Does the page limit include references? Yes