AI, Ethics, and Society

Homework Project #5

Readings:

- "Data preprocessing techniques for classification without discrimination" by Kamiran and Calders (2012) [https://link.springer.com/content/pdf/10.1007%2Fs10115-011-0463-8.pdf]
- "AI Fairness 360: An Extensible Toolkit for Detecting, Understanding, and Mitigating Unwanted Algorithmic Bias,"
    - https://arxiv.org/abs/1810.01943
    - https://aif360.mybluemix.net/
- The What-If Tool: Code-Free Probing of Machine Learning Models
    - https://ai.googleblog.com/2018/09/the-what-if-tool-code-free-probing-of.html
    - https://pair-code.github.io/what-if-tool/

A predictive learning algorithm predicts an outcome based on learning from previous instances of data. For example: Given an instance of a loan application, predict if the applicant will repay the loan. The learning algorithm makes these predictions based on a training dataset, where many other instances (other loan applications) and actual outcomes (whether they repaid) are provided.

Unfortunately, as you have discovered throughout this class, sometimes the patterns that are found by these learning algorithms may amplify historical biases. For example, a loan repayment algorithm may discover that age plays a significant role in the prediction of repayment because the training dataset happened to have better repayment for one age group than for another. This raises a major problem - even if this outcome is representative of the data, there are legal precedence/laws that make it illegal to base any decision on an applicant's age, regardless of whether this is a good prediction based on historical data.

To enable the mitigation of bias, a number of pre-determined "fairness metrics" have been proposed to help identify the bias since, unless you know what's broke, you can't fix it.

In this assignment, we will look at the impact of computing and applying fairness metrics to "fix" data that could be used to train algorithms associated with learning from credit-based data sets. Remember to answer ALL questions irrespective of the outcome from each step.

Step 1 - *Select one of the datasets for completion of this assignment*:

- German Credit Data Set – https://archive.ics.uci.edu/ml/datasets/statlog+(german+credit+data)
- Taiwan Credit Data Set – https://archive.ics.uci.edu/ml/datasets/default+of+credit+card+clients
- Portuguese Bank Marketing Data Set - https://archive.ics.uci.edu/ml/datasets/Bank+Marketing

Step 2 - *Explore the data by answering the following questions:*

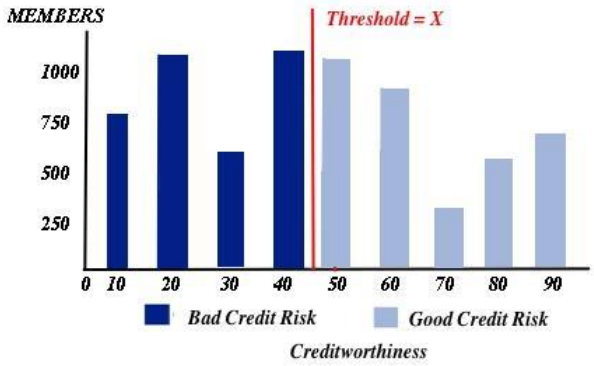
- Which dataset did you select?
- How many observations are in the dataset?
- How many variables in the dataset?
- Which variables did you select as your dependent variables?
- How many and which variables in the dataset are associated with a legally recognized protected class? Of those variables associated with a protected class, what is the associated legal precedence/law it falls under as discussed in the lectures?

Step 3 - *Based on your selected dataset, specify an outcome variable, protected attributes, and split the dataset into training and testing sets*

- Select outcome variable(s) that relates to the creditworthiness of a customer and derive a formula to score each customer based on whether they are an *Excellent Credit Risk (i.e. highly likely to pay back a loan)* versus *Bad Credit Risk (i.e. highly likely to default on loan)*. Select a range of scores from 0-100 where 100 is the maximum value for *Excellent Credit Risk*. To compute creditworthiness, you can apply any algorithm or set of calculations on the variables that makes sense to you - you can implement your own ML algorithm (which is perfectly fine), create a mathematical formula (which is the basis of all things AI/ML) or even just close your eyes, throw a dart, and pick a single variable from the dataset.
- Select a protected class attribute – i.e. choose an attribute on which the bias can occur, basically the attribute you want to test bias for.
- Define an unprivileged group and privileged group– i.e. choose a subset of protected attribute values which are considered unprivileged versus privileged from a fairness perspective (i.e. your unprivileged group would be your historically disadvantaged group of interest).
    - For example, we might select age as our protected class attribute. In this case, I may decide to choose Older (age >= 40) as the unprivileged group and Young (age < 40) as the privileged group.
    - This allows us to transform our data based on binary membership in a protected group (age).
- Randomly split your original dataset into equally-size training and testing sets. How many of each (privileged versus unprivileged) members are in each set?
- Provide your results indicating your selected outcome variable/conversion formula, protected class attribute, privileged group, and privileged group.
- Example Output:
    - Outcome variable: Creditworthiness derived from History of past payments and Y
    - Formula used to score members creditworthiness from 0 to 100 is [Some Formula]
    - Protected Class Attribute: Age
    - Privileged group: Young (age < 40); Number of Members in Training Set: J; Number of Members in Testing Set: K
    - Unprivileged group: Older (age >=40); Number of Members in Training Set: X; Number of Member in Testing Set: Y

Step 4 - *Graph and compute a default threshold that maximizes profit*

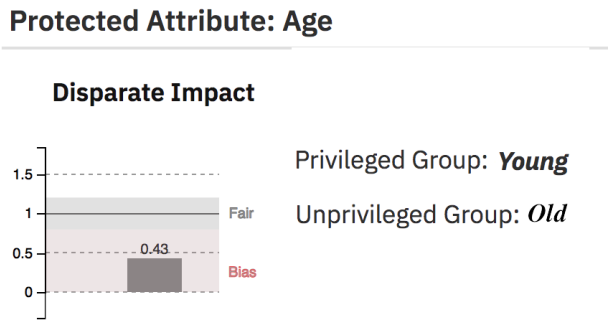
- ~~Using a histogram, graph the data associated with *Creditworthiness* (where creditworthiness is on the X-axis and the number of associated customers with that creditworthiness is on the Y-axis)~~
- ~~Compute a threshold for approving a loan (based on credit risk) that *tries* to maximize profit. Assume that a good credit risk is associated with a creditworthiness score >=X. Highlight the threshold information on the graph.~~
- To compute profits, assume, in this case:
    - Approved Loan/Good Credit Risk = +10 Profit
    - Approved Loan/Bad Credit Risk = -5 Profit
    - Declined Loan/Good Credit Risk = -3 Profit
    - Decline Loan/Bad Credit Risk = 0 Profit
- ~~What is your threshold value? What is the profit based on your threshold value?~~ Compute how many in each group (privileged and unprivileged) received Favorable (i.e. Approved) versus Unfavorable (i.e. Declined) outcomes based on your threshold value. Create a table documenting your results. Note: A Favorable outcome is associated with an Approved Loan. An Unfavorable outcome is associated with a Declined Loan.



Note: Provided for illustrative purposes only.

Step 5 - *Compute fairness metric on your training set*

- Based on your protected class attribute and defined privileged and unprivileged groups, ~~select ***two*** different fairness metrics (as defined in either the AI Fairness 360 Toolkit or What-If Tool) and compute the differences between privileged and unprivileged groups in your training set data~~ [*Note: You can code up your own mathematical formulations, modify open-source code that wasn't developed for this course, or use the python functions provided by the Toolkit(s) directly*]
- For example, if we use Disparate Impact as our fairness metric, we would compute the ratio of the rate of favorable outcome for the unprivileged group to that of the privileged group. The ideal value of this metric is 1.0. A value < 1 implies higher benefit for the privileged group and a value >1 implies a higher benefit for the unprivileged group.
- Identify which fairness metrics were selected and provide your quantitative results from applying the two different fairness metrics on your training set data using your default threshold (Computed in Step 4). ~~Do any of the differences indicate bias either for or against the unprivileged or privileged group? Graph the result for both fairness metrics (indicating the fair/bias thresholds) and document the results in a tabular format.~~



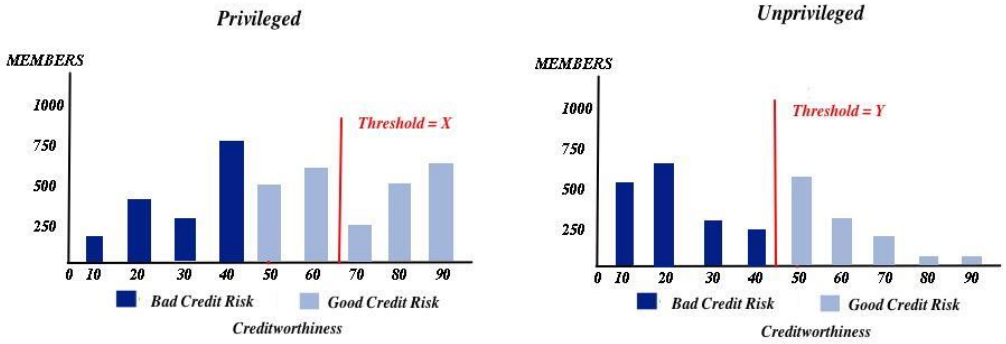
Note: Provided for illustrative purposes only

Step 6 – Mitigate bias in the training dataset

- We are going ~~to try to mitigate any biases in the training dataset. For this step, either 1) define two different creditworthiness formulas for the unprivileged versus privileged groups (relating to Step 3) and/or 2) define different threshold values for approving a loan even if they are considered a bad credit risk (Recall, in Step 4, we assumed that a good credit risk is associated with a creditworthiness score >=X).~~
- Select from one of the two fairness metrics identified in Step 5. Find threshold values for each group that removes differences between the privileged and unprivileged group based primarily on

the fairness metric selected while secondly maximizing profit as much as possible based on the profit formula in Step 4. If you found in Step 4 that there were no (or minimal) differences already, is there a different set of threshold values that provides more profits while still maintaining that minimal difference?

- Graph the histograms associated with *Good Credit Risk* versus *Bad Credit Risk* as a function of your protected class attribute and highlight your threshold information on the graph.
- What is your new creditworthiness formula (if defined differently)? What are your threshold values? What is the profit based on your threshold values? Create a table documenting how many in each group (privileged and unprivileged) received Favorable (i.e. Approved) versus Unfavorable (i.e. Declined) outcomes based on your new threshold values.



Note: Provided for illustrative purposes only.

Step 7 – For each of the fairness metrics selected in Step 5, discuss if there were any differences in the outcomes for the privileged versus unprivileged group? Discuss if the mitigation step in Step 6 was effective and for whom? Did any group receive a positive advantage? Was any group disadvantaged by the mitigation step? What issues would arise if you used this method to mitigate bias?

Step 8 - Turn in a report (in PDF format) documenting your outputs in each Step. The report should follow the JDF format. Jupyter notebook (ipynb files) submission is optional, but a final PDF document per JDF format is required. The file name for submission is GTuserName_Assignment_5, for example, Joyner03_Assignment_5. Reports that are not neat and well organized will receive up to a 10-point deduction. All charts, graphs, and tables should be generated in Python or Excel, or any other suitable software application, else appropriate points will be deducted, which could be the maximum.