Bias in AI Report- Assessing the Repair Process

Joshua Fitch
Department of Computer Science
University of Durham
Durham, United Kingdom
mccb22@durham.ac.uk

I. Project Proposal

For my project I will investigate the effectiveness of the repair process in mitigating bias of a machine learning model. The repair process is a pre-processing technique described by Michael Feldman in a 2015 paper [1]. It is designed to reduce bias of datasets such that a model won't be able to determine protected columns values by looking at unprotected columns. I choose this process not only because it compares favourably to other techniques that mitigate bias [2], but is also very flexible, allowing partial repairs to balance performance and fairness not available with many other methods. Importantly, the repair process is also not model specific, meaning it can be applied to reduce bias of a machine learning model in almost any context. This opens the possibility of it becoming a standard tool to apply to any human-centric dataset where bias might be an issue.

My investigation will involve comparing performance (accuracy and utility) and fairness (zemel fairness and disparate impact) of a logistic regression model trained on data with and without the repair process applied. The dataset I plan to use is the adult income dataset [3] (note I use a version of the dataset not already split into training and testing sets- the csv file is in the folder submitted with my python files) which holds data from a census on whether someone earns above or below 50,000 dollars a year. This can be predicted using the 14 other attributes in the dataset, which include two protected attributes: race and sex. I will make two python programs in jupyter notebook, a conventional and fair implementation of a logistic regression model to predict income for the adult dataset. To allow for consistent comparisons I will use identical pre-processing of the dataset in both files, except I will also implement the repair process in the fair version of my file. I will also use matching model hyperparameters and the same metrics to assess both models after they have been trained. This should allow me to make a conclusion as the whether the repair process has reduced bias of my model.

I will be roughly following the experimental setup in Feldman's paper[1], carrying out identical pre-processing and using the same metrics to assess model performance and fairness. Due to the considerably quicker training time I am using a logistic regression model as opposed to a support vector machine, meaning I will not be able to compare my results directly to those in Feldman's paper[1]. However, I should still see similar patterns in my results, and a considerable reduction in bias after repairing data.

Implementation files will be coded in python 3.9.4, using pandas dataframes and numpy arrays for data structures, scikit-learn for the models and hyperparameter tuning and seaborn/matplotlib for any visualisations.

II. Project Progress

A. Data Analysis

I pre-process the adult income dataset so it is suitable for the repair process and training of machine learning algorithms. Due to the nature of the repair process, it only works on numeric and orderable columns, so any categorical variables that have no clear order are dropped from the dataset. Conveniently, after dropping of these variables there were no null values left in the dataset. Next, categorical variables sex, race and income were encoded into 0's and 1's. As race is a non-orderable categorical feature, it is converted to either 1- white or 0- non-white. As race and sex are dropped after the repair process I also drop them before my conventional implementation, to allow fair comparison. The final dataset used to train, validate and test machine learning models contains 48,842 rows and 7 columns: age, fnlwgt (final weight), education-num, capital-gain, capital-loss, hours-per-week and income.

Analysis of features shows interesting differences in statistics between subgroups. Education-num, which is an integer corresponding to the highest level of education obtained is on average around 0.4 higher for white compared to none white entries, with only 14% of non-whites achieving a bachelors degree compared to 17% of white entries. The proportion of white entries that are male, married and from the US is also considerably higher than for non-white entries. 25% of white entries earn over 50K compared to only 15% of non-white entries. On average female entries are 2.5 years younger than male entries in the dataset, and work 6 hours a week less. There are also less white female entries, and less married female entries compared to males. In the dataset only 11% of females earn

```
Female non-white numeric features :
                age  education-num  capital-gain  capital-loss  hours-per-week
count  3165.000000    3165.000000   3165.000000   3165.000000     3165.000000
mean     37.113428       9.718167    522.729858     49.786414       37.079305
std      12.751543       2.395384   5523.258763    301.970065       10.292822
Male white numeric features :
                age  education-num  capital-gain  capital-loss  hours-per-week
count  28735.000000   28735.000000  28735.000000  28735.000000    28735.000000
mean      39.704507      10.133461   1364.602192    102.975152       42.665947
std       13.475250       2.650503   8495.541557    435.672999       12.209256

Female non-white incomes:
mode is 92.82780410742497 % of total
<=50K    2938
>50K      227
Name: income, dtype: int64

Male white incomes:
mode is 68.4531059683313 % of total
<=50K    19670
>50K      9065
Name: income, dtype: int64
```

over 50K compared to 30% of males, despite very similar levels of education. Finally, it is worth noting that being in two sub-groups often has a cumulative effect on features, non-white females have on average the lowest levels of education, and startlingly only 7% of non-white females in the dataset earn over 50K, compared to 32% of white males.

Just in initial analysis it is clear sampling bias is present in the dataset, with far more values for white people and men compared to non-white people and woman. Especially in the case of men and woman this does not reflect the underlying population, although it is not clear if this bias originated from certain groups being more likely to answer the census, or bias in extracting data from the census. Prejudice bias also appears to be present, with data reflecting common stereotypes, and many attributes being predictive of race and sex. This will most likely mean the machine learning model will learn to associate certain races and sexes with lower or higher income, even if race and sex attributes themselves are dropped from the dataset. This is what I will aim to resolve in my fair algorithm implementation.

B. Conventional Implementation

For my algorithm I decide to use logistic regression. Logistic regression uses log odds and the sigmoid function to make classifications, and is very widely used for binary classification [4]. This is different from Feldman's paper [1], in which a support vector machine is used. I choose to use logistic regression as support vector machines have long training times, making hyperparameter tuning and multiple run throughs of code very difficult.

Before actually training my model I first split the dataset into training (≈70%), testing (≈15%) and validation (≈ 15%) sets. I use the validation set to tune logistic regression hyperparameters. I grid search the most important hyperparameters for logistic regression within reasonable ranges using scikit-learn, allowing me to boost performance.

I then train my tuned model on the training set, and assess its accuracy on the test set. My conventional implementation logistic regression model achieves an 79.6% accuracy on the test set. I also test model accuracy on the training set, this turns out to be very similar to model accuracy on the testing set- an indicator the model is not overfitting.

I then plot a confusion matrix to help calculate utility, defined as 1 – BER (balanced error rate) [1], where:

$$BER = ½(FN/FN+TP) + ½(FP/FP+TN) \quad (1)$$

Utility is a useful measure as it places equal weight between protected attributes values, and is more stable than accuracy [1]. My conventional model has a 0.611 utility compared to 0.796 accuracy, indicating the model does not equally weigh all attributes.

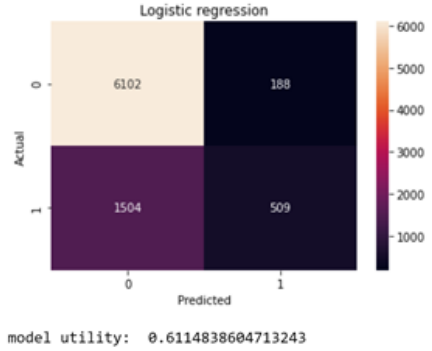

Figure 2: Confusion matrix and utility of conventional logistic regression model

I also calculate 1 - zemel fairness, where zemel fairness is the difference in probability a random privileged sample is predicted to earn over 50K, compared to a random non-privileged sample. It is 0.97 for race and 0.96 for sex. Finally, I calculate disparate impact, very similar to zemel fairness, except it is the probability a random non-priveleged sample earns over 50K divided by the corresponding probability for a privileged sample. It is 0.64 for race and 0.57 for sex. When race and sex are combined, zemel fairness and disparate impact drop to 0.94 and 0.42 respectively. These results show a clear example of bias in my model, most likely stemming from biased data.

After running my conventional model once I experiment with training a model on a subsample of the dataset containing equal numbers of entries from each protected group. This model shows increased accuracy at 84.8%, probably due to a reduction in the number of entries with income over 50K, as many white male entries have been removed. This makes it easier for the model to achieve high accuracy as it will be predicting mainly 0's. However, results for both utility and fairness metrics are very similar to those of the model trained on the original dataset. With utility at 0.616, zemel fairness for race and sex combined at 0.94, and disparate impact at 0.35. This indicates bias of the model is not just due to different sizes of groups trained on, but that there is also strong race and sex bias present within the actual dataset.

C. Fair Implementation

To make my model fairer I implement Feldman's repair tool[1]. The repair tool aims to make marginal distributions of attributes for sensitive features equal, and has a parameter λ that controls to what extent this happens. First all sensitive stratified groups are made (e.g. non-white female), and then each groups distribution on each attribute is made equal. Values of each sensitive group are split into n quantiles, where n is the number of observations in the smallest sensitive group. For each quantile, the median of each group is found, then the median of these medians is taken as the "target value". In a full repair (λ=1) every observation in a quantile is made equal to the target value, in a partial repair λ and the original value are used to find a new value.

In combinatorial repair a sorted list of unique values for that attribute is required (SL) and a new value SL[ri] is found by calculating an index for this list:

$$ri = SL.index(original) + \lambda * (SL.index(target)-SL.index(original)) \quad (2)$$

Geometric repair calculates the new value directly:

$$new\_value = ((1 - \lambda) \cdot original) + (\lambda \cdot target) \quad (3)$$

I implement both and have a parameter to switch between them.

After the repair process all sensitive attribute columns are dropped, and I use this data to train a logistic regression model with identical hyperparameters to the conventional implementation. Using combinatorial repair at λ=0.8 (a value that provides a good balance of performance and fairness) accuracy of the model is 0.802 and utility is 0.606- a decrease of 0.05. However, Zemel fairness of race and sex has increased by 0.04 to 0.98 and disparate impact has

increased by 0.37 to 0.79 compared to the conventional model. Increasing λ further increases scores for fairness metrics, but at the expense of accuracy and utility. To visualize this I plot a graph of model utility (as it is more stable than accuracy) against λ and a graph of fairness (average of all previously described fairness metrics) against λ.

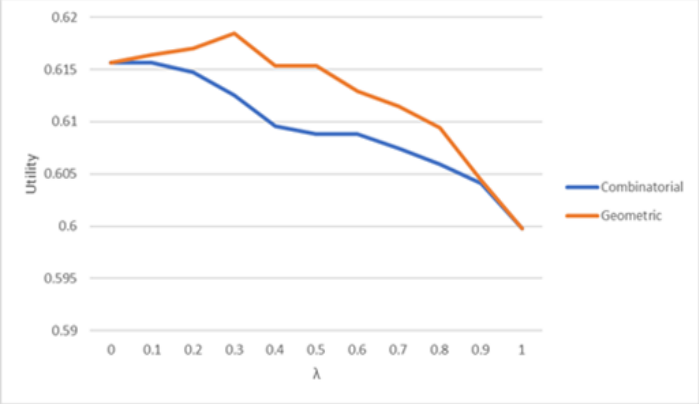

Figure 3. Graph showing change in utility as λ increases

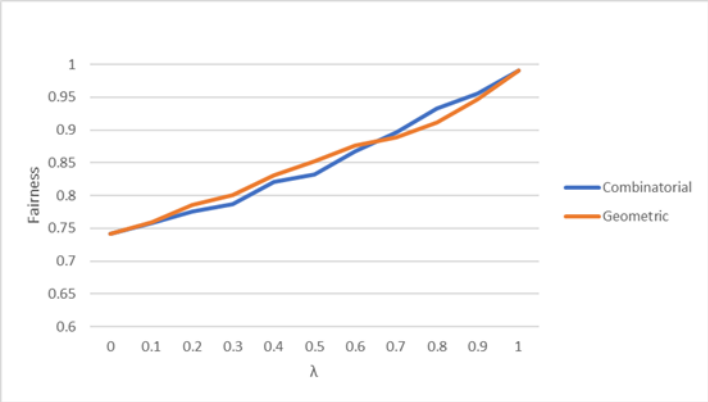

Figure 4. Graph showing change in fairness as λ increases

Due to the use of a different model my results are not exactly the same as those in paper [1], however they do follow the same pattern. As λ increases the marginal distributions of attributes between protected groups become more equal, and this causes performance of the model to decrease. However it also increases the number of entries in protected groups receiving positive outcomes, and therefore fairness measures increase towards 1 as λ does. Interestingly, utility is higher for most points with geometric repair compared to combinatorial, not observed in [1]. This may be due to the difference in algorithm, but as the disparity in numbers is very small, it is also very possible that this is just random chance.

D. Conclusions

Overall my results show that the repair process is very effective for mitigating bias when used with a logistic regression model. When λ=1 my fairness score ≈ 1, indicating the repair process can almost completely eliminate bias from a dataset, and therefore any algorithm trained on that dataset. Moreover, at values of λ between 0 and 1 the repair process is still very effective at mitigating bias. This adjustable parameter could be tuned to meet the constraints of an organization or law that a machine learning model needs to abide by, whilst still preserving as much performance as possible, making it very useful.

From my point of view, the main drawback of the repair process is its inability to deal with non-ordered categoric variables, not only does this result in a loss of information and accuracy, but it may be especially harmful with protected attributes like race. In Feldman's [1] implementation of the repair process all races not in the privileged class are grouped into non-white. This does not take into account the different types and amounts of bias experienced by different groups within that minority class, and could lead to a kind of hidden bias where minority groups as a whole are treated fairly, but sub-groups within this continue to be unfairly treated. As suggested in paper [1], one way to deal with this could be to use the repair process on numeric and ordered categorical variables, and another pre-processing technique on non-ordered categorical variables. Making sure marginal distributions are equalized on sub-groups within a main minority group could also help remedy this.

Nevertheless, the flexibility and effectiveness of the repair process makes it a valuable and useful pre-processing technique to reduce bias, and should be one of the first considerations of researchers if bias of a human centric dataset is an issue.

REFERENCES

[1] Feldman, M. (2015) 'Computational Fairness: Preventing Machine-Learned Discrimination', p. 26.

[2] Friedler, S. A. et al. (2019) 'A comparative study of fairness-enhancing interventions in machine learning', in Proceedings of the Conference on Fairness, Accountability, and Transparency. FAT* '19: Conference on Fairness, Accountability, and Transparency, Atlanta GA USA: ACM, pp. 329–338. doi: 10.1145/3287560.3287589.

[3] UCI Machine Learning Repository: Adult Data Set (1996). Available at: https://archive.ics.uci.edu/ml/datasets/adult (Accessed: 8 May 2021).

[4] Wright, R. E. (1995) 'Logistic regression', in Reading and understanding multivariate statistics. Washington, DC, US: American Psychological Association, pp. 217–244.