Removing Biased Data to Improve Fairness and Accuracy

Qualifying Project Presentation

Sahil Verma

Qualifying Committee: Rene Just, Su-In Lee

Co-authors: Michael Ernst and Rene Just

Automated Decision Making: Uses

- Automated decision making is cheaper, faster, and objective.
- Machine learning (ML) is a popular technique used for automation.
- A wide range of tasks have been automated including critical applications like loan sanctioning, hiring decisions, predictive policing, and parole decisions.

Why is Fairness a Concern?

- For training a ML model, the organization uses historical data.
- If the historical data is demographically biased, that would be reflected in the models.
- Kinds of biases in data that affect fairness:
  - Label bias
  - Selection bias

Past Incidents of Discriminatory Behavior

Several past applications using ML models have been criticized for being unfair to a particular demographic group:

- Amazon hiring model and Apple credit card limits were found to be biased against women.
- Parole recommendation and predictive policing models were found to be discriminating against Black people.

Research Statement

In this work, we aim to develop an approach that debiases the training data, such that the model trained using this data would be fair to all demographic groups.

We focus on a binary classification task.

Individual Fairness

Among several fairness metrics, a popular one is individual fairness.

It states that similar individuals should receive similar outcomes.

In context of ML classification, similar outcome would mean the same class label.

And a distance function can be used to determine if two individuals are sufficiently similar.

Similar individuals not receiving the same outcome is termed as a ***discriminatory pair***.

Intuition for Our Approach

Given a trained model and its training data, we aim to ***identify and remove the training datapoints that cause discrimination***.

To do so, we ***generate a synthetic dataset of similar individuals*** and identify the discriminatory pairs out of them.

The result of this intervention is a ***debiased dataset***. When the model is retrained with this debiased dataset, it should have ***lower individual discrimination***.

The Algorithm (Overview)

- Train the model on the training dataset.
- Generate synthetic dataset of similar individuals, and identify the discriminatory pairs out of them.
- Find the most influential datapoints in the training data:
  - In each discriminatory pair, one individual faced discrimination.
  - Heuristic: individual with lower confidence prediction faced discrimination.
  - Identify training datapoints with the most influence on discriminated individuals.
- Remove the biased points to reach a local minima of individual discrimination.

The Algorithm (Visually)

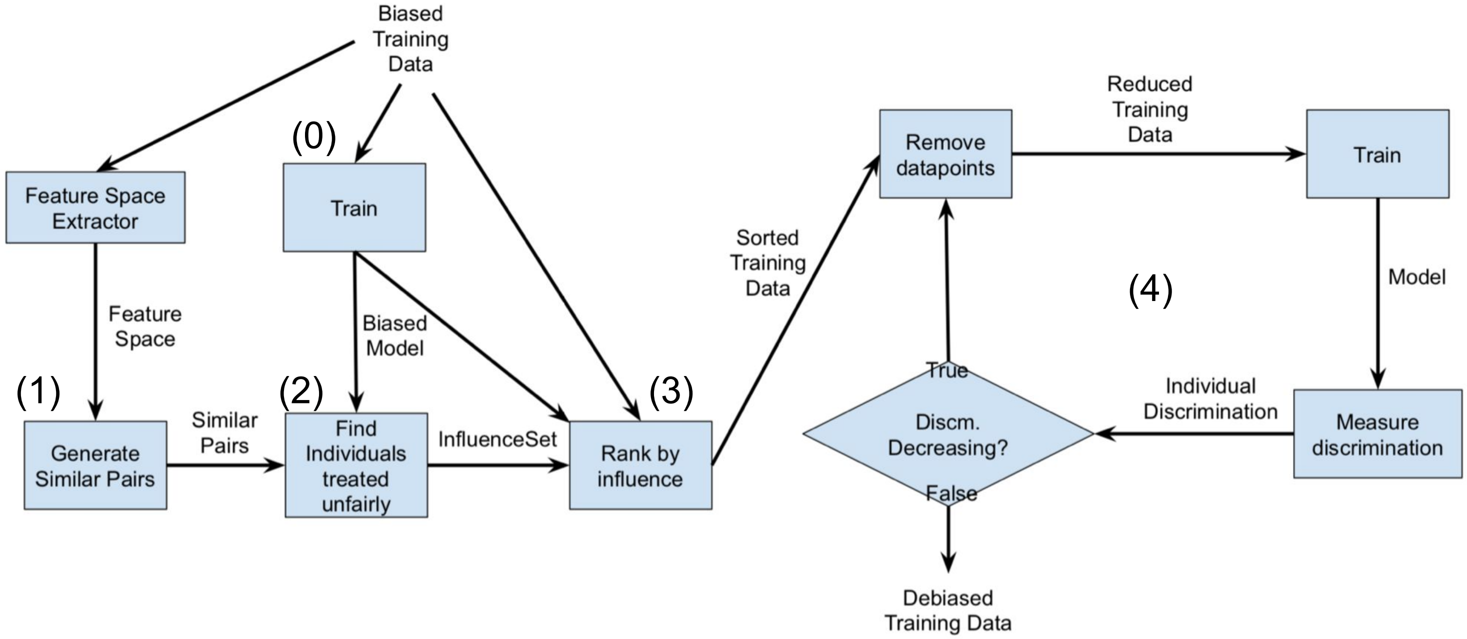


Figure 1: Flowchart with the steps in our approach. The left portion shows the steps in Algorithm 1 and the right portion shows the steps in Algorithm 2. The output of the algorithm is a debiased training data which can be used to train a debiased model.

Running Example

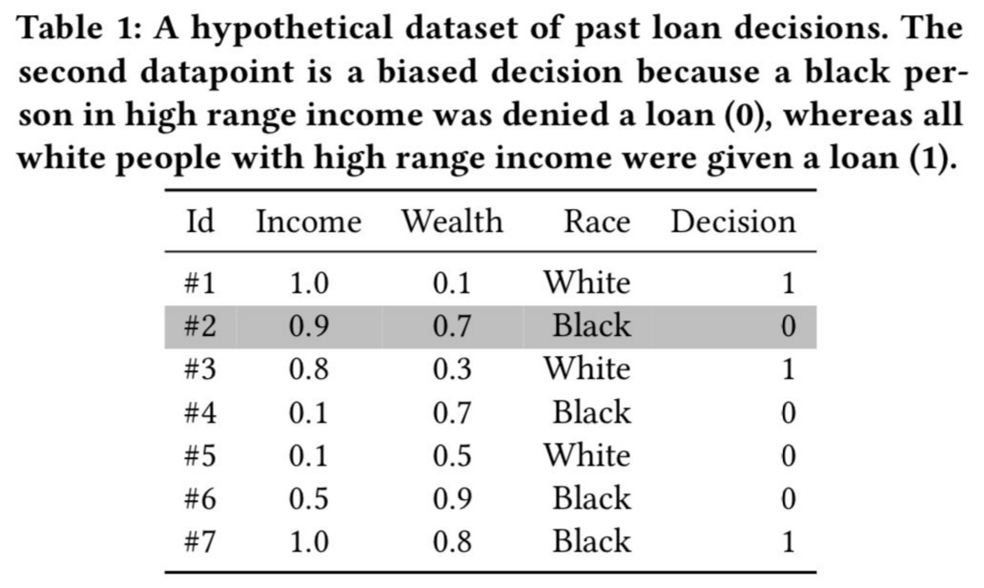


Table 1: A hypothetical dataset of past loan decisions. The second datapoint is a biased decision because a black person in high range income was denied a loan (0), whereas all white people with high range income were given a loan (1).

Id	Income	Wealth	Race	Decision
#1	1.0	0.1	White	1
#2	0.9	0.7	Black	0
#3	0.8	0.3	White	1
#4	0.1	0.7	Black	0
#5	0.1	0.5	White	0
#6	0.5	0.9	Black	0
#7	1.0	0.8	Black	1

The Algorithm (Details)

Step1: Generate a synthetic dataset of similar individuals

Input space
Similarity Condition

Income	Wealth	Race
0.7	0.2	White

0.7	0.2	Black

We generated 700 pairs of similar individuals for the example dataset.

The Algorithm (Details)

Step2: Use the trained model to identify discriminatory pairs within the synthetic dataset

Income	Wealth	Race	Prediction
0.7	0.2	White	1

0.7	0.2	Black	**?**

? = 1 => Non-discriminatory

? = 0 => Discriminatory pair!

Out of the 700 pairs, 26% (181 pairs) were found to be discriminatory pairs.

The Algorithm (Details)

For each pair, identify the individual that was discriminated against.

- In each discriminatory pair, one of the individuals faced discrimination (assuming binary classification).
- Heuristic: Under the assumption that biased data is a minority in the training dataset, the prediction of the individual with lower confidence is considered an unfair prediction.

The Algorithm (Details)

For the discriminatory pair we looked at:

Income	Wealth	Race	Prediction
0.7	0.2	White	1

Prediction confidence: 80%

0.7	0.2	Black	0

Prediction confidence: 60%

181 discriminated individuals were identified, one for each discriminatory pair.

The Algorithm (Details)

Step3: Find the training datapoints with the most influence on discriminated individuals

- Influence functions [1] to rank the training datapoints in order of most to least influential for the unfair predictions (Step 3).
- We hypothesize that this is the order of most to least biased datapoints.

[1] Koh, P.W. & Liang, P.. (2017). Understanding Black-box Predictions via Influence Functions. Proceedings of the 34th International Conference on Machine Learning.

The Algorithm (Details)

For the example dataset, when influence functions were used to sort the training data, the datapoints were ranked as follow:

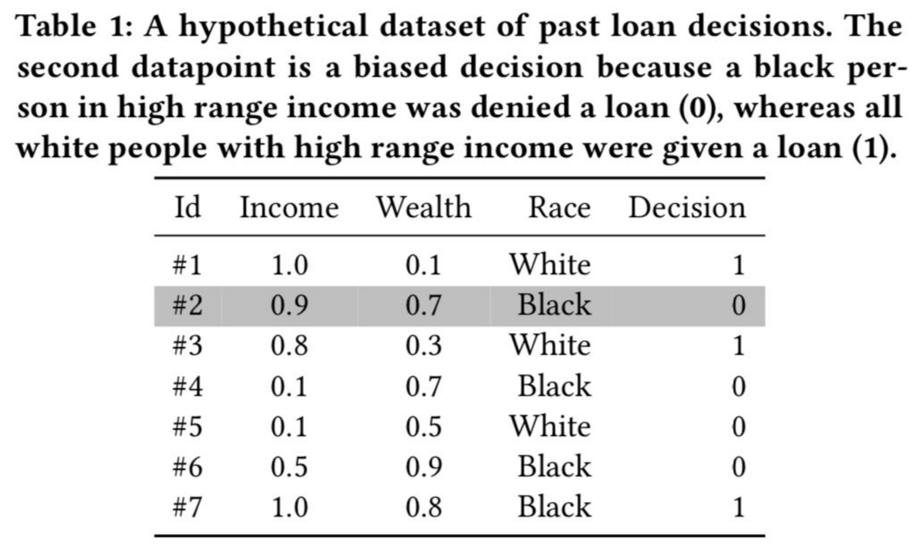
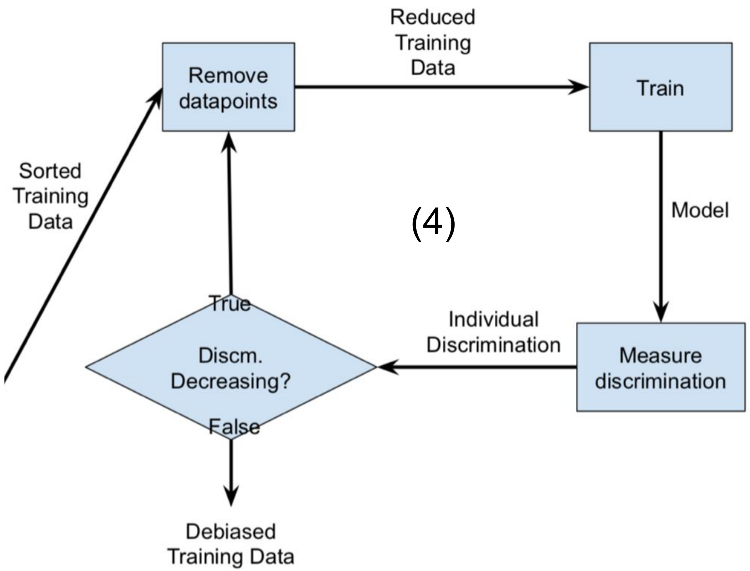


Table 1: A hypothetical dataset of past loan decisions. The second datapoint is a biased decision because a black person in high range income was denied a loan (0), whereas all white people with high range income were given a loan (1).

Id	Income	Wealth	Race	Decision
#1	1.0	0.1	White	1
#2	0.9	0.7	Black	0
#3	0.8	0.3	White	1
#4	0.1	0.7	Black	0
#5	0.1	0.5	White	0
#6	0.5	0.9	Black	0
#7	1.0	0.8	Black	1

#2
#7
#3
#1
#4
#6
#5

The Algorithm (Step 4)



The Algorithm (Details)

Step4: Obtain debiased dataset (in a loop)

- Remove chunks of most biased datapoints.
- Retrain the model with remaining data
- Check for the remaining individual discrimination of the model
- When the remaining individual discrimination reaches a local minima, stop removing datapoints. This is the debiased dataset.

The Algorithm (Details)

For the example dataset, we removed one datapoint at a time:

- After removing #2, the retrained model had only 1 discriminatory pair (0.14% remaining discrimination).
- When both #2 and #7 were removed, the retrained model had 16% remaining discrimination.

Therefore, removing #2 was the local minima. The remaining dataset is the “debiased dataset”.

Evaluation Metrics

We performed experiments to evaluate our technique against baselines, on the following metrics:

- Individual discrimination
- Test accuracy
- Statistical disparity
- Sensitivity to hyperparameter choices (for individual discrimination and test accuracy).

Baselines

We compared our approach to seven baselines:

- No technique used (model trained on *full* dataset).
- Simple exclusion of sensitive attribute (*SR*).
- Disparate Impact Removal (*DIR*).
- Preferential Sampling (*PS*).
- Massaging (*MA*).
- Learning Fair Representations (*LFR*).
- Adversarial Debiasing (*AD*).

IBM AIF360:

Research Questions

- How do the techniques compare in terms of reducing *individual discrimination*?
- How do the techniques compare in terms of *test accuracy*?
- How do the techniques compare in terms of *statistical disparity*?
- How sensitive are the techniques to *hyperparameter choices*?

Datasets

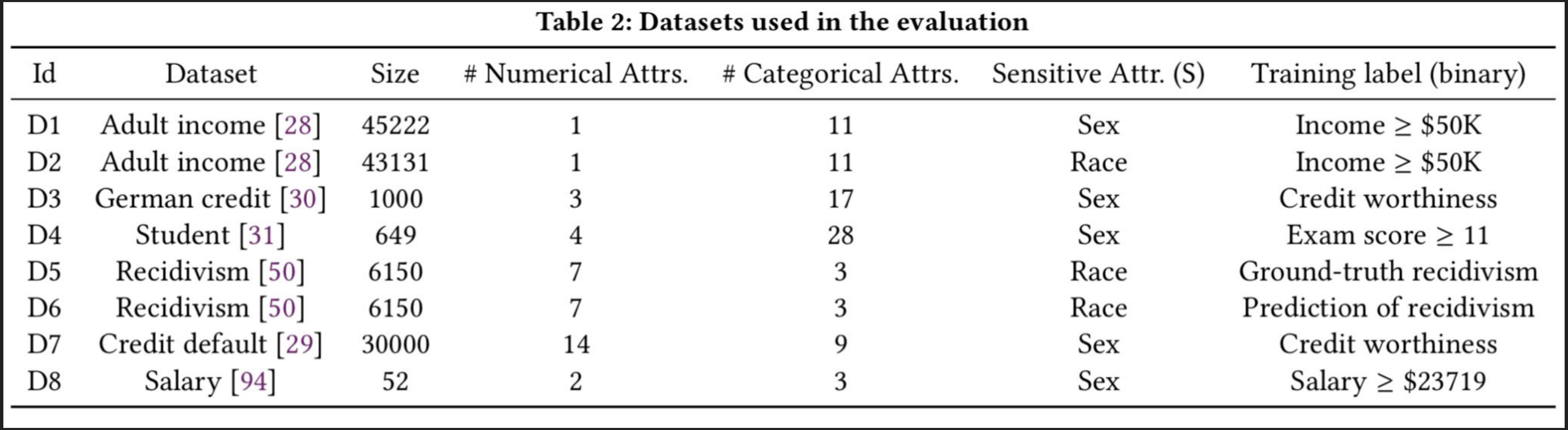
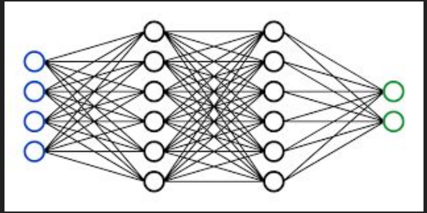


Table 2: Datasets used in the evaluation

Id	Dataset	Size	# Numerical Attrs.	# Categorical Attrs.	Sensitive Attr. (S)	Training label (binary)
D1	Adult income [28]	45222	1	11	Sex	Income ≥ $50K
D2	Adult income [28]	43131	1	11	Race	Income ≥ $50K
D3	German credit [30]	1000	3	17	Sex	Credit worthiness
D4	Student [31]	649	4	28	Sex	Exam score ≥ 11
D5	Recidivism [50]	6150	7	3	Race	Ground-truth recidivism
D6	Recidivism [50]	6150	7	3	Race	Prediction of recidivism
D7	Credit default [29]	30000	14	9	Sex	Credit worthiness
D8	Salary [94]	52	2	3	Sex	Salary ≥ $23719

ML Model Architecture



We used a neural network model for binary classification task.

It had the following choices for hyperparameters:

- In the first hidden layer, there were either 16, 24, or 32 neurons.
- In the second hidden layer, there were either 8 or 12 neurons.
- There were two batch sizes for optimization using SGD.
- And there were 20 random permutations of the whole dataset.

This makes a total of ***240 hyperparameter choices***.

Experimental Methodology

For each 240 hyperparameter choice:

- Split the dataset into first 80% training and rest 20% testing.
- Train the model using the training dataset.
- Debias the training dataset and retrain the model using it.
- Add the biased datapoints removed to a set named “Unfair datapoints”.

Individual Discrimination: Similarity Condition

We used two different *input* similarity condition in the experiments:

- Two individuals are considered similar if they have exactly the *same attribute value for all non-sensitive features* (λ = 0).
- Two individuals are considered similar if have the *same attribute value for all categorical features and within a 10% range for all numerical features*, barring sensitive features (λ > 0).

For both the cases, the *output* similarity condition required the same outcome for the pair of similar individuals.

Test Accuracy: On the unbiased test points.

Wick et al., 2019 emphasized that their exists a discrepancy between the discrimination mitigation approaches and their evaluation. Most prior works measure the accuracy on the full test set which leads to a fairness-accuracy trade-off.

We assess the model’s accuracy on the fair datapoints only. Our approach debunks this belief.

Our evaluation removes the “unfair datapoints” from the test set, when measuring test accuracy of a model. Note that there is no leak between the training and test set for a particular model.

Statistical Disparity

Statistical disparity is the difference in the probability of being classified in the desired class among the different demographic groups.

We measure the statistical disparity using the test set of the model.

Statistical disparity is an alternate measure of fairness, like individual discrimination.

Results

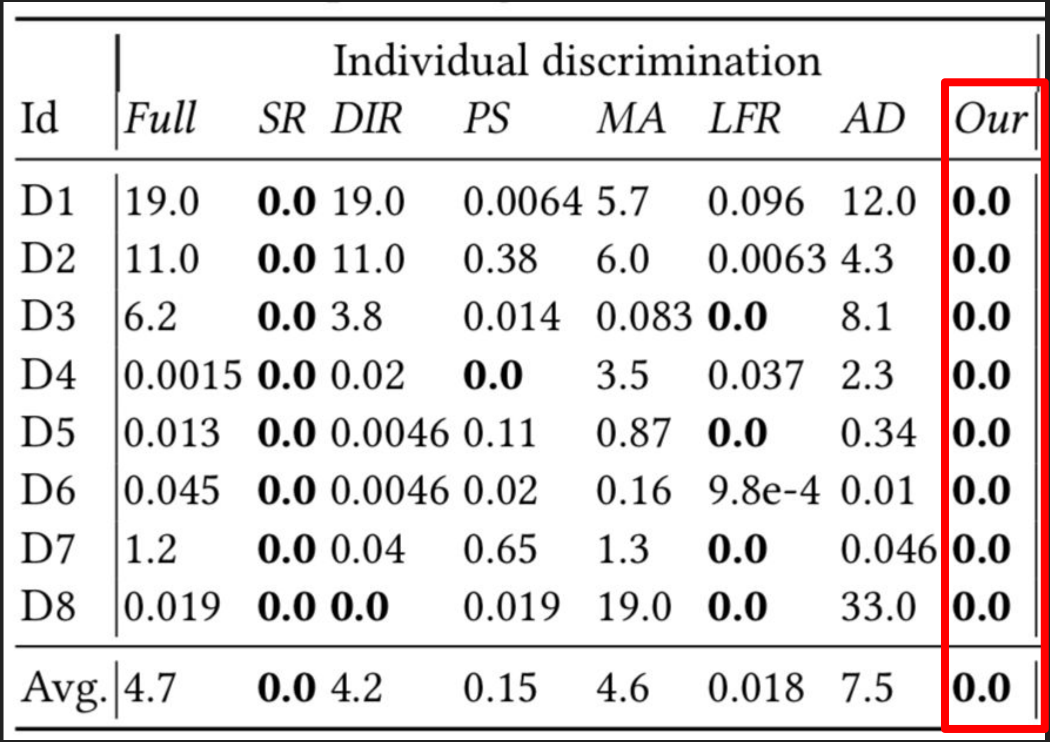
For all the datasets, we computed the individual discrimination, test accuracy, and statistical disparity for all 240 models trained.

In the paper, we show the three aforementioned metrics for the least discriminative, the most accurate, and the least statistically disparate model (among the 240 models). This answers the first five research questions.

To compare hyperparameter sensitivity, we plot boxplots for individual discrimination and test accuracy using the 240 models for all datasets and baselines.

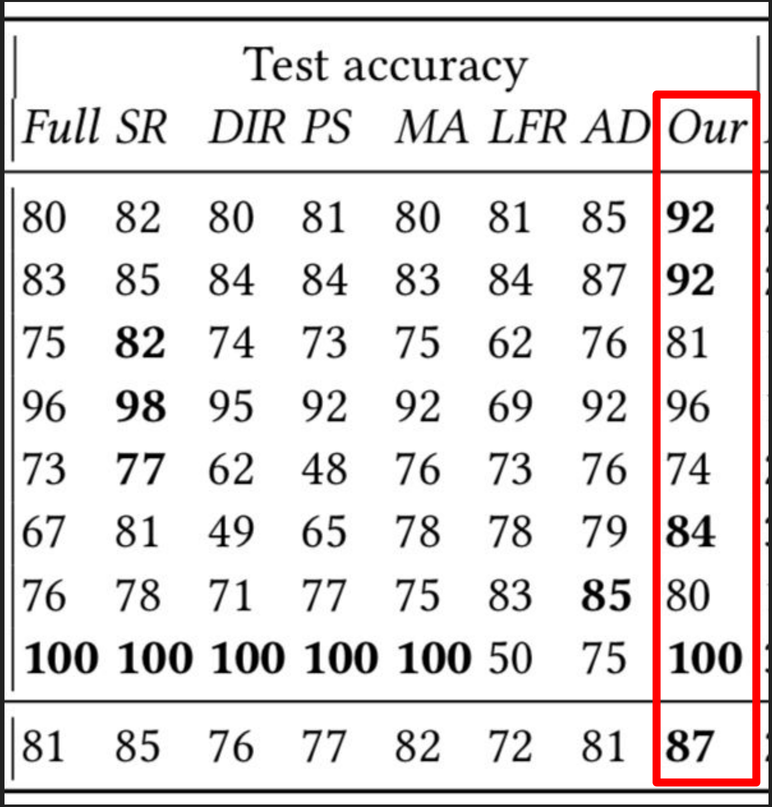
We answer the questions for both the input similarity conditions.

Least Discriminative Model (λ = 0)



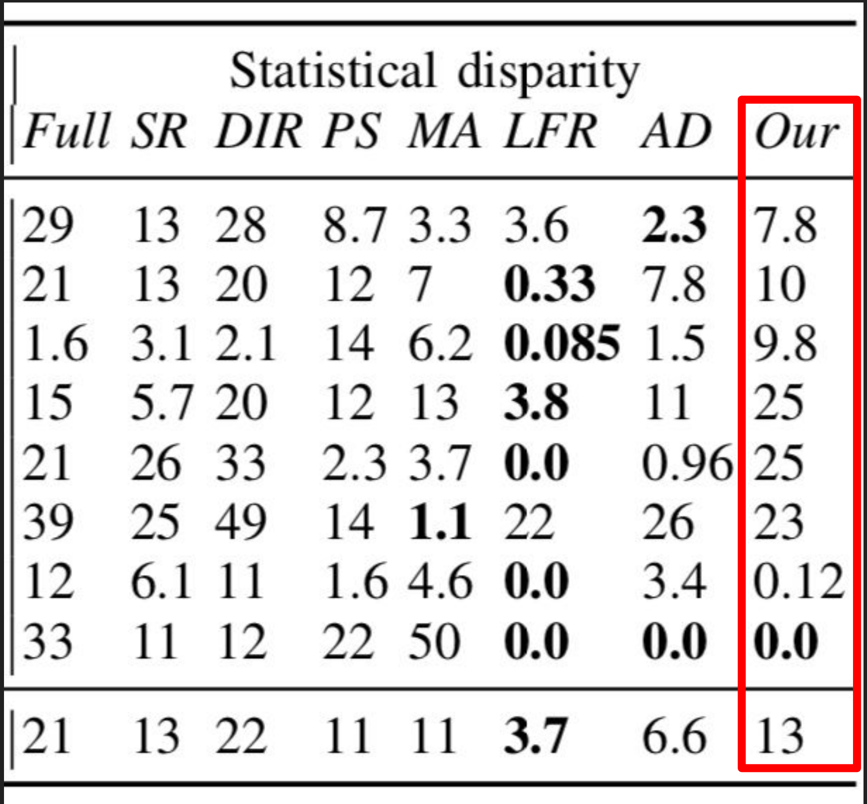
	Individual discrimination							
Id	*Full*	*SR*	*DIR*	*PS*	*MA*	*LFR*	*AD*	*Our*
D1	19.0	**0.0**	19.0	0.0064	5.7	0.096	12.0	**0.0**
D2	11.0	**0.0**	11.0	0.38	6.0	0.0063	4.3	**0.0**
D3	6.2	**0.0**	3.8	0.014	0.083	**0.0**	8.1	**0.0**
D4	0.0015	**0.0**	0.02	**0.0**	3.5	0.037	2.3	**0.0**
D5	0.013	**0.0**	0.0046	0.11	0.87	**0.0**	0.34	**0.0**
D6	0.045	**0.0**	0.0046	0.02	0.16	9.8e-4	0.01	**0.0**
D7	1.2	**0.0**	0.04	0.65	1.3	**0.0**	0.046	**0.0**
D8	0.019	**0.0**	**0.0**	0.019	19.0	**0.0**	33.0	**0.0**
Avg.	4.7	**0.0**	4.2	0.15	4.6	0.018	7.5	**0.0**

Least Discriminative Model (λ = 0)



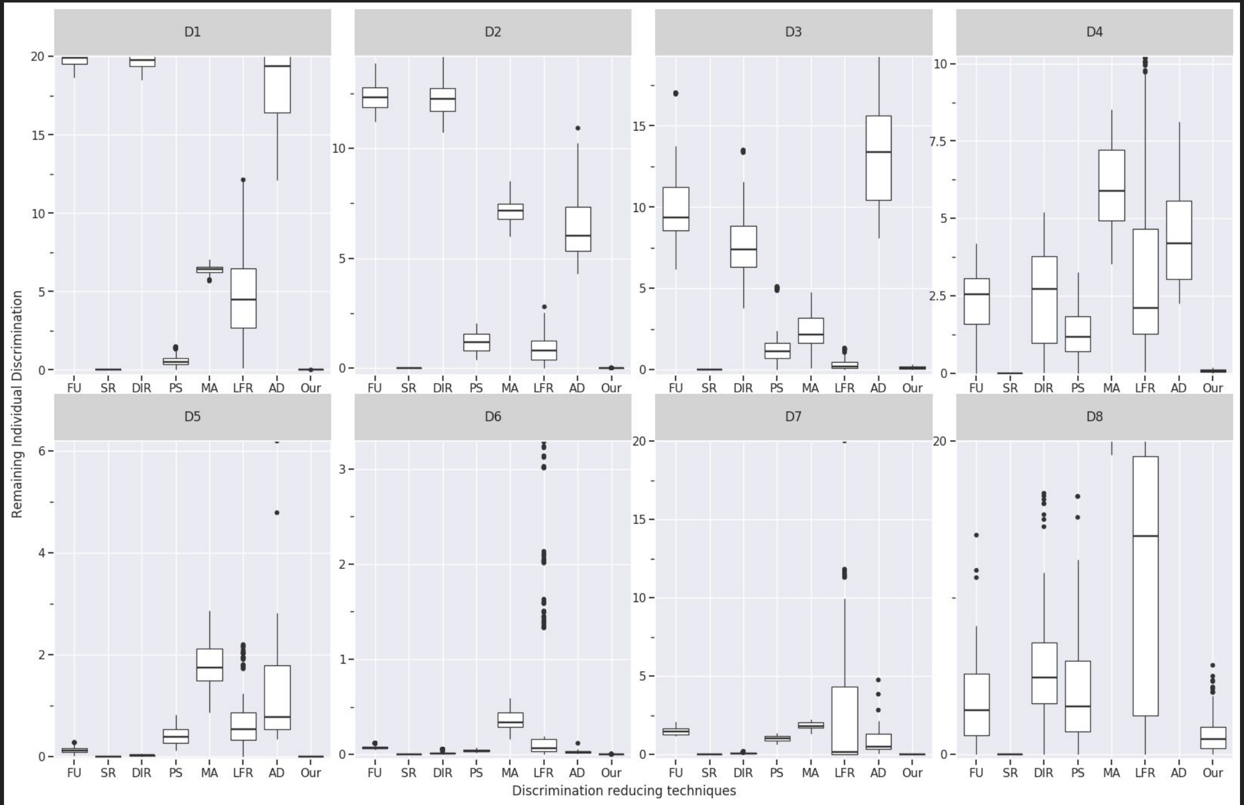
Test accuracy							
Full	*SR*	*DIR*	*PS*	*MA*	*LFR*	*AD*	*Our*
80	82	80	81	80	81	85	**92**
83	85	84	84	83	84	87	**92**
75	**82**	74	73	75	62	76	81
96	**98**	95	92	92	69	92	96
73	**77**	62	48	76	73	76	74
67	81	49	65	78	78	79	**84**
76	78	71	77	75	83	**85**	80
100	**100**	**100**	**100**	**100**	50	75	**100**
81	85	76	77	82	72	81	**87**

Least Discriminative Model ($\lambda = 0$)



Statistical disparity							
Full	*SR*	*DIR*	*PS*	*MA*	*LFR*	*AD*	*Our*
29	13	28	8.7	3.3	3.6	**2.3**	7.8
21	13	20	12	7	**0.33**	7.8	10
1.6	3.1	2.1	14	6.2	**0.085**	1.5	9.8
15	5.7	20	12	13	**3.8**	11	25
21	26	33	2.3	3.7	**0.0**	0.96	25
39	25	49	14	**1.1**	22	26	23
12	6.1	11	1.6	4.6	**0.0**	3.4	0.12
33	11	12	22	50	**0.0**	**0.0**	**0.0**
21	13	22	11	11	**3.7**	6.6	13

Least Discriminative Model ($\lambda = 0$)



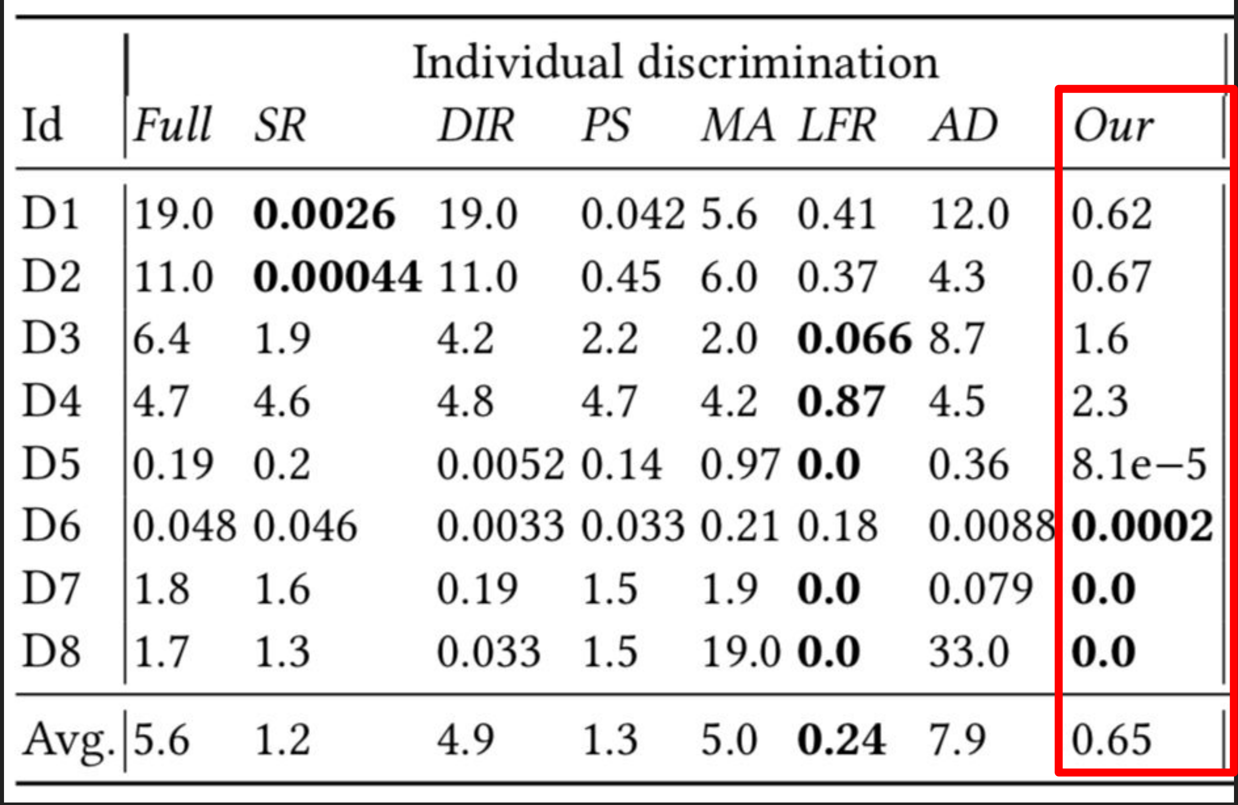
Research Questions

- How do the techniques compare in terms of reducing *individual discrimination*?
- How do the techniques compare in terms of *test accuracy*?
- How do the techniques compare in terms of *statistical disparity*?
- How sensitive are the techniques to *hyperparameter choices*?

Answers to the Research Questions (λ = 0)

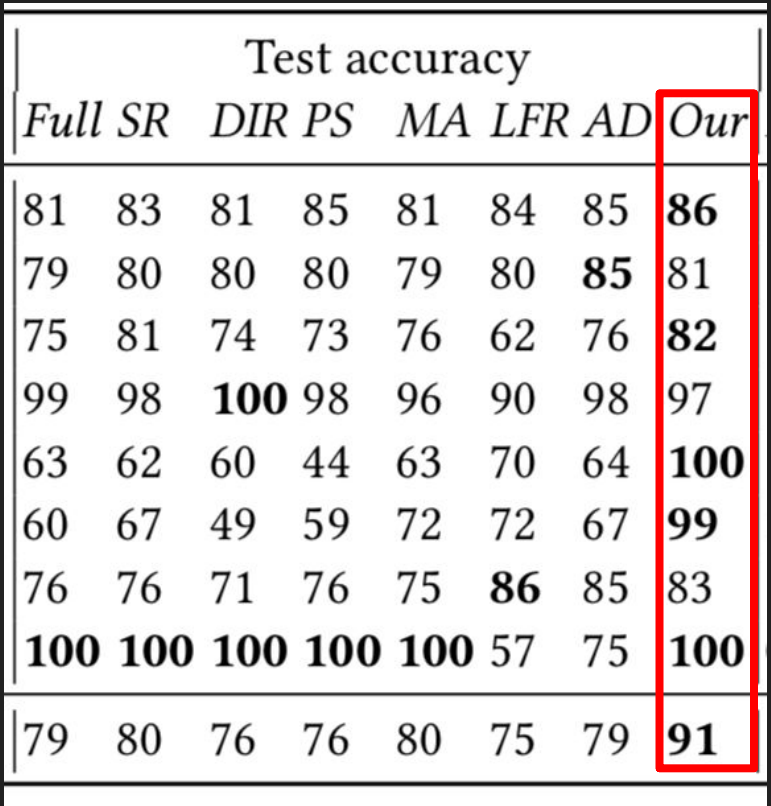
- Our technique ***achieves 0% remaining discrimination*** for all datasets.
- The remaining discrimination is ***also 0%*** if we simply remove the sensitive attribute ***(SR)***. LFR and PS are the next best ones.
- Our technique results in ***better test accuracy than all baselines***. It also debunks the accuracy fairness trade-off (better than “Full”).
- Our technique obtains ***lower statistical parity*** than the “Full” baseline. Compared to all baselines, it is in the middle of the pack.
- Our approach is ***much less sensitive*** to hyperparameter choices.

Least Discriminative Model ($\lambda > 0$)



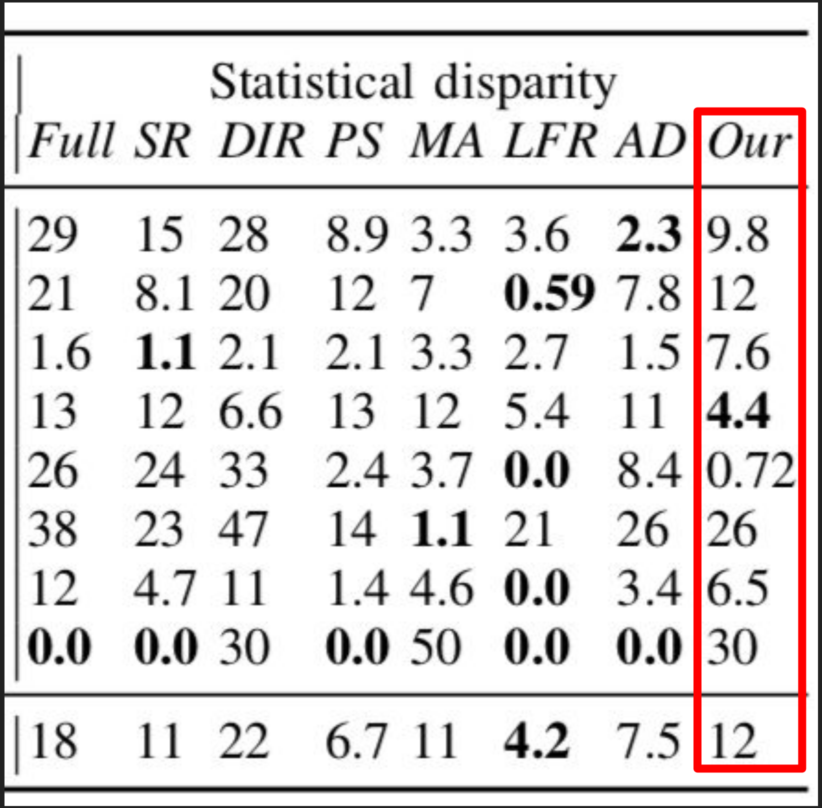
Id	Individual discrimination							
	Full	*SR*	*DIR*	*PS*	*MA*	*LFR*	*AD*	*Our*
D1	19.0	**0.0026**	19.0	0.042	5.6	0.41	12.0	0.62
D2	11.0	**0.00044**	11.0	0.45	6.0	0.37	4.3	0.67
D3	6.4	1.9	4.2	2.2	2.0	**0.066**	8.7	1.6
D4	4.7	4.6	4.8	4.7	4.2	**0.87**	4.5	2.3
D5	0.19	0.2	0.0052	0.14	0.97	**0.0**	0.36	8.1e−5
D6	0.048	0.046	0.0033	0.033	0.21	0.18	0.0088	**0.0002**
D7	1.8	1.6	0.19	1.5	1.9	**0.0**	0.079	**0.0**
D8	1.7	1.3	0.033	1.5	19.0	**0.0**	33.0	**0.0**
Avg.	5.6	1.2	4.9	1.3	5.0	**0.24**	7.9	0.65

Least Discriminative Model ($\lambda > 0$)



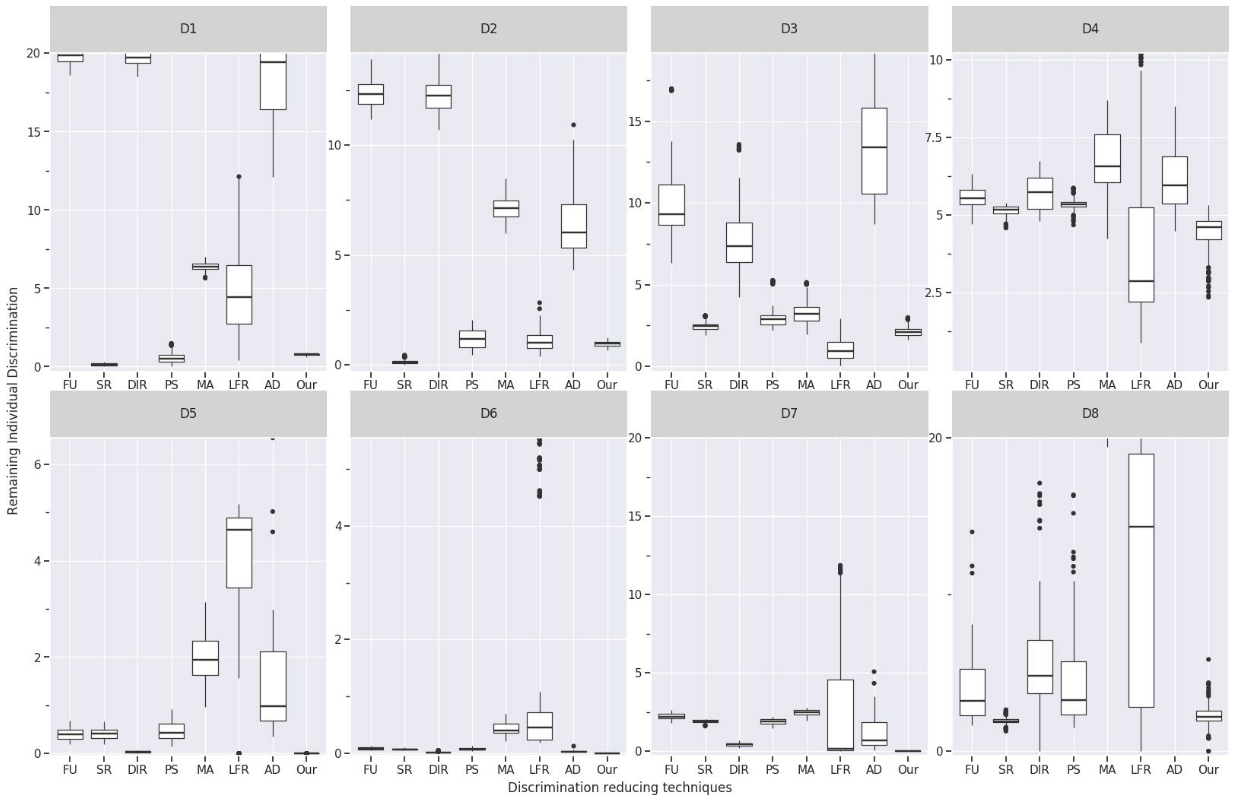
Test accuracy							
Full	*SR*	*DIR*	*PS*	*MA*	*LFR*	*AD*	*Our*
81	83	81	85	81	84	85	**86**
79	80	80	80	79	80	**85**	81
75	81	74	73	76	62	76	**82**
99	98	**100**	98	96	90	98	97
63	62	60	44	63	70	64	**100**
60	67	49	59	72	72	67	**99**
76	76	71	76	75	**86**	85	83
100	**100**	**100**	**100**	**100**	57	75	**100**
79	80	76	76	80	75	79	**91**

Least Discriminative Model ($\lambda > 0$)



Statistical disparity							
Full	*SR*	*DIR*	*PS*	*MA*	*LFR*	*AD*	*Our*
29	15	28	8.9	3.3	3.6	**2.3**	9.8
21	8.1	20	12	7	**0.59**	7.8	12
1.6	**1.1**	2.1	2.1	3.3	2.7	1.5	7.6
13	12	6.6	13	12	5.4	11	**4.4**
26	24	33	2.4	3.7	**0.0**	8.4	0.72
38	23	47	14	**1.1**	21	26	26
12	4.7	11	1.4	4.6	**0.0**	3.4	6.5
0.0	**0.0**	30	**0.0**	50	**0.0**	**0.0**	30
18	11	22	6.7	11	**4.2**	7.5	12

Least Discriminative Model ($\lambda > 0$)



Answers to the Research Questions ($\lambda > 0$)

- Our technique ***achieves very low remaining*** discrimination for all datasets.
- The discrimination is second to LFR and ***much lower than SR***.
- Our technique results in ***better test accuracy*** than all baselines.
- The conclusion for statistical parity remains the same.
- The conclusion for sensitivity to hyperparameter choices remains the same.

Related Work: Fairness Metrics

- Fairness metrics can be broadly categorized in three categories:
  - ***Group fairness***: All definitions other than statistical disparity are based on the confusion matrix and require access to ground-truth.
  - ***Individual fairness***: Does not require access to ground-truth.
  - ***Causal fairness*****:** Requires a causal model to reason about the effects of a feature on another. Causal models are not learnable and are designed by domain experts.

Related Work: Literature

Fairness literature in ML classification is broadly divided into discrimination detection and mitigation. We use such a detection approach.

Bias mitigation approaches are further categorized into:

- Pre-processing: Intervenes on the training data.
- In-processing: Intervenes when the model is being trained.
- Post-processing: Intervenes when the model is being used.

Since we remove biased datapoints, ours is a pre-processing approach.

Future Work

We plan to address several of our limitations in future:

- Generate realistic individuals for fairness testing.
- Design input similarity condition metrics which accommodates historical and ongoing discrimination within society.
- Develop an approach that estimates the parameters of the model when few training datapoints are removed, without retraining.
- Analyse the effect of removing datapoints on model properties like robustness and generalizability, and compare that to the case when the datapoints are down weighted.

Thank you!

Questions?