Fair Enough: Searching for Sufficient Measures of Fairness

Anonymous Author(s)

ABSTRACT

Testing machine learning software for ethical bias has become a pressing current concern. In response, recent research has proposed a plethora of new fairness metrics, for example, the dozens of fairness metrics in the IBM AIF360 toolkit. This raises the question: How can any fairness tool satisfy such a diverse range of goals?

While we cannot completely simplify the task of fairness testing, we can certainly reduce the problem. This paper shows that many of those fairness metrics effectively measure the same thing. Based on experiments using seven real-world datasets, we find that (a) 26 classification metrics can be clustered into seven groups, and (b) four dataset metrics can be clustered into three groups. Further, each reduced set may actually predict different things. Hence, it is no longer necessary (or even possible) to satisfy all fairness metrics.

In summary, to simplify the fairness testing problem, we recommend the following steps: (1) determine what type of fairness is desirable (and we offer a handful of such types); then (2) lookup those types in our clusters; then (3) just test for one item per cluster. To support that processing, all our scripts (and example datasets) are available at **https://github.com/Repoanonymous/Fairness_Metrics**.

KEYWORDS

Ethics in Software Engineering, Machine Learning with and for SE

1 INTRODUCTION

Increasingly, software is being used for critical decision-making processes, such as patient release from hospitals [15, 83], credit card applications [48], hiring [81], and admissions [19]. According to guidelines from the European Union [13] and IEEE [16], a software cannot be used in real-life applications if it is found to be discriminatory toward an individual based on any sensitive attribute such as gender, race, or age. Hence "fairness testing" is now an open and pressing problem in software engineering.

As shown in Table 1, researchers have proposed a plethora of fairness metrics for measuring fairness, and that number is growing. We can find multiple ways to measure "fairness". For example: (a) The Fairlearn [20] tool lists 16 metrics; (b) The Fairkit-learn toolkit [62] comes with 16 metrics, and (c) The IBM AIF360 toolkit [24] implements 45 fairness metrics. However, researchers in this area only use a few metrics in their papers [39, 53, 59, 64, 75, 92]. For example, in our literature review papers from the last three years, we see only a handful of papers (13 out of 60 to the best of our knowledge) using more than five fairness metrics to evaluate their method. This is surprising since all of them ignore more than half the known metrics. But is that wise?

- Should we reject papers that "only" use (e.g.) five metrics?
- Or should researchers always use dozens of metrics?
- When we use automatic tools to optimize for fairness, should we optimize for dozens of goals?
- Or is optimizing for a smaller set sufficient?

The conjecture that is tested by this paper is that *too many spurious metrics* which *all measure very similar things*. If that were true, then it should be possible to simplify fairness assessment as follows:

> *Run metrics on real-world data. Find clusters of correlated metrics. Prune "insensitive clusters[1]". Only use one metric per surviving cluster.*

This paper experiments with seven datasets and finds that (a) 26 classification fairness metrics can be clustered into just seven groups; (b) four dataset metrics can be clustered into three groups and that (c) these clusters actually predict for different things. That is, it is no longer necessary (or even possible) to satisfy all these fairness metrics. Hence, to simplify fairness testing, we recommend (a) determining what type of fairness is desirable (and we offer a handful of such types); then (b) looking up those types in our clusters; then (c) testing for one item per cluster.

This paper is structured around these research questions.

RQ1: *Do current fairness metrics agree with each other?* Our experiments show that current fairness metrics often disagree, markedly.

RQ2: *Can we group (cluster) fairness metrics based on similarity?* We find sets of similar metrics using agglomerative clustering [5].

RQ3: *Are some fairness metrics more sensitive to change than others?* While most are sensitive, some are not.

RQ4: *Can we achieve fairness based on all the metrics at the same time?* It is challenging to do so since some of them are competing goals and some are contradictory based on definitions.

We assert this work is **novel**, **significant**, **sound** & **verifiable**.

Novel: Our *metrics selection tactic* prunes spurious metrics (i.e., metrics which merely echo the results of other metrics). This tactic was tested in an *extensive case study* applying 30 fairness metrics and grouped them into clusters (RQ1 & RQ2). This study is the first one to perform such a *sensitivity meta-analysis* of fairness testing and to warn that some metrics are unresponsive to data changes (RQ3). This study also presents a *meta-analysis of metrics ability* to achieve fairness after application of bias mitigation technique (RQ4).

Significant: Our work significantly clarifies and simplifies fairness testing. We show not only how to assess fairness metrics but which metrics offer little additional information content. This is important since the art of software fairness testing is evolving rapidly. Studies like the one are essential to documenting what methods are "best" (as opposed to those that might distract from core issues).

Sound: All our empirical results were repeated 25 times. Also, all our analytical results (in §5.1) were double-checked via empirical analysis. Further, this study is far more detailed than prior work since it explores multiple bias mitigation algorithms on seven datasets (than prior work [38, 40–42, 58] was tested on far fewer metrics and far fewer datasets).

[1] Note: Here, by "insensitive" clusters, we mean those where changes to the data do not change the fairness scores.

Table 1: Mathematical definitions of the classification and dataset metrics used in this research. Definitions are collected from IBM AIF360 [24], Fairkit-learn [62] & Fairlearn [20]. For definitions of the terms used here, see Table 3.

Metric Id (MID)	Metric Name	Description	Ideal Value	AIF360	Fairkit	Fairlearn
		Classification Metrics				
C0	true_positive_rate_difference	$TPR_{D=unprivileged} - TPR_{D=privileged}$	0	✓	✓	✓
C1	false_positive_rate_difference	$FPR_{D=unprivileged} - FPR_{D=privileged}$	0	✓	✓	✓
C2	false_negative_rate_difference	$FNR_{D=unprivileged} - FNR_{D=privileged}$	0	✓	✓	✓
C3	false_omission_rate_difference	$FOR_{D=unprivileged} - FOR_{D=privileged}$	0	✓	✓	
C4	false_discovery_rate_difference	$FDR_{D=unprivileged} - FDR_{D=privileged}$	0	✓	✓	
C5	false_positive_rate_ratio	$FPR_{D=unprivileged}/FPR_{D=privileged}$	1	✓	✓	✓
C6	false_negative_rate_ratio	$FNR_{D=unprivileged}/FNR_{D=privileged}$	1	✓	✓	✓
C7	false_omission_rate_ratio	$FOR_{D=unprivileged}/FOR_{D=privileged}$	1	✓	✓	
C8	false_discovery_rate_ratio	$FDR_{D=unprivileged}/FDR_{D=privileged}$	1	✓	✓	
C9	average_odds_difference	$\frac{1}{2}(false\_positive\_rate\_difference$ $+ true\_positive\_rate\_difference)$	0	✓	✓	
C10	average_abs_odds_difference	$\frac{1}{2}(\vert false\_positive\_rate\_difference\vert$ $+ \vert true\_positive\_rate\_difference\vert)$	0	✓	✓	
C11	error_rate_difference	$ERR_{D=unprivileged} - ERR_{D=privileged}$	0	✓	✓	
C12	error_rate_ratio	$ERR_{D=unprivileged}/ERR_{D=privileged}$	1	✓	✓	
C13	selection_rate	$Pr(\hat{Y} = favorable)$	0	✓	✓	
C14	disparate_impact	$Pr(\hat{Y} = 1 \vert D = unprivileged)/Pr(\hat{Y} = 1 \vert D = privileged)$	1	✓	✓	✓
C15	statistical_parity_difference	$Pr(\hat{Y} = 1 \vert D = unprivileged) - Pr(\hat{Y} = 1 \vert D = privileged)$	0	✓	✓	✓
C16	generalized_entropy_index	$\frac{1}{n\alpha(\alpha-1)} \sum_{i=1}^{n} [(\frac{b_i}{\mu})^\alpha - 1]$ where $b_i = \hat{y}_i - y_i + 1$	0	✓		
C17	between_all_groups_generalized _entropy_index	generalized_entropy_index between all groups	0	✓		
C18	between_group_generalized _entropy_index	generalized_entropy_index between privileged and unprivileged groups	0	✓		
C19	theil_index	$\frac{1}{n} \sum_{i=1}^{n} \frac{b_i}{\mu} \ln \frac{b_i}{\mu}$	0	✓		
C20	coefficient_of_variation	$2 * \sqrt{generalized\_entropy\_index}$	0	✓		
C21	between_group_theil_index	theil_index between privileged and unprivileged groups	0	✓		
C22	between_group_coefficient _of_variation	coefficient_of_variation privileged and unprivileged groups	0	✓		
C23	between_all_groups_theil _index	theil_index between all groups	0	✓		
C24	between_all_groups_coefficient _of_variation	coefficient_of_variation between all groups	0	✓		
C25	differential_fairness_bias _amplification	Smoothed EDF between the classifier and the original dataset	0	✓		
		Dataset Metrics				
D0	consistency	$1 - \frac{1}{n*n\_neighbors} \sum_{i-1}^{n} \vert\hat{y}_i - \sum_{j \in N_{n_neighbors(x_i)}} \hat{j}y\vert$	1	✓		
D1	smoothed_empirical _differential_fairness	Smoothed EDF	0	✓		
D2	mean_difference	$Pr(\hat{Y} = 1 \vert D = unprivileged) - Pr(\hat{Y} = 1 \vert D = privileged)$	0	✓		
D3	disparate_impact	$Pr(\hat{Y} = 1 \vert D = unprivileged)/Pr(\hat{Y} = 1 \vert D = privileged)$	1	✓		

Verifiable: In order to support replication and reproduction of our results, all our datasets and scripts are publicly available at https://github.com/Repoanonymous/Fairness_Metrics.

Before beginning, we digress to make three points. *Firstly*, Table 1 lists dozens of metrics currently seen in the SE fairness testing literature. This paper makes an *empirical argument* that this list is too long since many of these metrics offer similar conclusions. One alternative to our *empirical argument* is an *analytical argument* that metric X (e.g.) is equivalent to metric Y. Later in this paper (see §5.1), we make the case that to reduce the space of metrics to be explored, that kind of analytical argument may actually be misleading.

Secondly to be clear, while we can reduce dozens of metrics down to ten, there will still be issues of how to trade-off within this reduced set. That said, we assert our work is valuable since debating the merits of, say, ten metrics is a far simpler task than trying to resolve all the conflicts between 30. Further, and more importantly, our methods could be used as a litmus test to prune away spurious new metrics that merely report old ideas but in a different way.

Also, *thirdly*, even after our mitigation algorithms, some fairness metrics still can contradict each other regarding presence of bias. Hence, in §5.3, we offer an extensive discussion on what to do in that situation.

2 BACKGROUND

2.1 The Problem of Fairness

There is much evidence of machine learning (ML) software showing biased behavior. For example, language processing tools are more accurate on English written by Anglo-Saxons than written by people of other races [32]. An Amazon hiring tool was found to be biased against women [12]. YouTube makes more mistakes while generating closed captions for videos with female voices than males [71, 84]. A popular risk-score predicting algorithm was found to be heavily biased against African Americans showing a higher error rate while predicting future criminals [8]. Gender bias is also prevalent in Google [34] and Bing [62] translators.

Due to so many undesirable events, academic researchers and big industries have started giving immense importance to ML software fairness. Microsoft has launched ethical principles of AI where "fairness" has been given the topmost priority [18]. IBM has built a toolkit called AI Fairness 360 [11] containing the most noted works in the fairness domain. In recent years, the software engineering research community has also started exploring this topic actively. ICSE'18 held a special workshop for "software fairness" [14]. ASE'19 held another workshop called EXPLAIN, where fairness and explainability of ML models were discussed [17]. Johnson et al. have created a public GitHub repository for data scientists to evaluate ML models based on quality and fairness metrics simultaneously [62].

As to technology developed to detect and fix these issues of fairness, we can see three groups: *fairness testing, model-based mitigation, and fairness metrics,* .

Fairness Testing: The idea here is to generating discriminatory test cases and finding whether the model shows discrimination or not. The first work on this was THEMIS, done by Galhotra, et al. [57]. THEMIS generates test cases by randomly perturbing attributes. AEQUITAS [86] improves the way of test case generation to become more efficient. Aggarwal, et al. combined local explanation and symbolic execution to generate a better black-box testing strategy [21].

Model Bias Mitigation: There are three techniques used to remove bias from model behavior. The first one is "pre-processing" where before model training, bias is removed from training data. Some popular prior work includes optimized pre-processing [35], Fair-SMOTE [41] and reweighing [65]. The second one is "in- processing" where after model training, the trained model is optimized for fairness. Popular prior work includes prejudice remover regularizer [68] and meta fair classifier [37]. The last one is "post-processing" where while making predictions, model output is changed to remove discrimination. Some noted works include reject option classification [67] and calibration [75]. There is some work that combines two or more of these techniques, such as Fairway [42], a combination of "pre-processing" and "in-processing".

While the fairness testing and model bias mitigation are important areas, we note that *before* we can declare success in those two areas, we *first* need some way to measure that success.

Accordingly, this paper focuses on the third area called:

Fairness Metrics: Early work in this area was done by Verma, et al. [89] who divided 20 fairness metrics into five groups based on the theoretical definitions. Hinnefeld, et al. made a comparative analysis of four fairness metrics [60]. Wang, et al. did a user study to find a relation between fairness metrics and human judgments [93]. There are also some papers coming from industry on the topic. LinkedIn has created a toolkit called LiFT for scalable computation of fairness metrics as part of large ML systems [88]. Recently, Amazon internally published an empirical study based on 18 fairness metrics [52].

While all that research is certainly insightful, in some sense that work has been too successful. As mentioned in the introduction, the above work has now generated a plethora of metrics. Hence, for the rest of this paper, we check if we can simplify the current space of metrics.

2.2 Metrics Used in this Study

In our work, we collected all the metric definitions from the IBM AI Fairness 360 GitHub repository. Table 1 lists the metrics studied in this paper. The *Fairkit* and *Fairlearn* columns in Table 1 show the metrics that are common among the IBM AIF360 metrics and metrics from Fairkit [62] (16 out of 16 available metrics) and Fairlearn [20] (7 out of 16 metrics) toolkit.

Before explaining fairness metrics, we need to understand some terminology. Table 2 contains seven binary classification datasets. The binary outcomes are *favorable* if it gives an advantage to the receiver (i.e., being hired for a job, getting credit card approved). Each of these datasets has at least one *protected attribute* that divides the population into two groups (*privileged & unprivileged*) that have differences in terms of benefits received. "sex", "race", "age" are examples of protected attributes. The goal of group fairness is, based on the protected attribute, privileged and unprivileged groups will be treated similarly. While individual fairness tries to provide similar outcomes to similar individuals.

A *fairness metric* is a quantification of unwanted bias in training data or models. Table 1 shows a sample of such metrics. When selecting these particular metrics, we skipped over:

- Metrics for which we could not access precise definitions and implementations in IBM AIF360 toolkit [24];
- Metrics for which we could not find publications to use as baselines in this paper.

These two selection rules resulted in the 30 metrics of Table 1, which divide as follows:

Classification Metrics: These measure fairness based on classification results and are labeled in Table 1 using a *Metric Id* beginning with *C*. Two inputs are needed to measure this: the first one is original dataset with true labels and the second one is predicted dataset. In case of binary classification, classification metrics can be calculated from confusion matrix. Table 3 shows a combined confusion matrix where every cell is divided based on the protected attribute.

Dataset Metrics: While classification metrics relate to predictions made from models, *dataset metrics* discuss learner-independent properties of the data. These are labeled in Table 1 using a *Metric Id* beginning with *D*. Only one input is needed to compute this: original dataset or transformed (by some bias mitigation algorithm) dataset. It can be applied for both *group and individual fairness.*

Distortion Metrics: For completeness, we note that AIF360 includes a third set of metrics called *distortion metrics.* While these metrics are not seen extensively in the current literature, they would be a worthy target for future research.

Table 2: Details of the datasets used in this research.

Dataset	#Rows	#Features	Protected Attribute		Class Label	
			Privileged	Unprivileged	Favorable	Unfavorable
Adult Census Income [2]	48,842	14	Sex-Male Race-White	Sex-Female Race-Non-white	High Income	Low Income
Compas [7]	7,214	28	Sex-Male Race-Caucasian	Sex-Female Race-Not Caucasian	Did not reoffend	Reoffended
German Credit [3]	1,000	20	Sex-Male	Sex-Female	Good Credit	Bad Credit
Heart Health [4]	297	14	Age-Young	Age-Old	Not Disease	Disease
Bank Marketing [9]	45,211	16	Age-Old	Age-Young	Term Deposit - Yes	Term Deposit - No
Student Performance [6]	1,044	33	Sex-Male	Sex-Female	Good Grade	Bad Grade
Titanic ML [10]	891	10	Sex-Male	Sex-Female	Survived	Not Survived

Table 3: Mathematical definition of various confusion matrix based rates. These are used to calculate fairness metrics defined in Table 1.

	Actual Positive	Actual Negative
Predicted Positive	TP PPV = TP/(TP+FP) TPR = TP/(TP+FN)	FP FDR = FP/(TP+FP) FPR = FP/(FP+TN)
Predicted Negative	FN FOR = FN/(TN+FN) FNR = FN/(TP+FN)	TN NPV = TN/(TN+FN) TNR=TN/(TN+FP)

In Table 1, each metric has an *ideal value* representing the best-case scenario. This means at ideal value according to the metric privileged and unprivileged groups are treated equally. For most of the metrics, the ideal value is zero, while in some cases where the metric is a ratio, the ideal value is one. If the ideal value for a metric is zero, a positive value denotes an advantage for the unprivileged group, while a negative value denotes an advantage for the privileged group. On the other hand, if the ideal value for a metric is one, a value < one denotes an advantage for the privileged group and > one denotes an advantage for the unprivileged group.

To use these metrics, some threshold must be applied to report "fair" or "unfair";

- For metrics with ideal value 0: the IBM AIF360 toolkit [24] uses the following definition of "fair": ranges between -0.1 to 0.1 as "fair" (so "unfair" means values outside that range).
- For metrics with ideal value 1: the IBM AIF360 toolkit [24] uses the following definition of "fair": ranges between 0.8 to 1.2 as "fair" (so "unfair" means values outside that range).

3 METHODOLOGY

3.1 Models

This paper analyzes the 30 fairness metrics in Table 1 using the seven datasets described in Table 2. In that work, we use one baseline model and two models tuned by pre-processing and in-processing algorithms to compare with:

- **Baseline:** We used a logistic regression model for creating baseline results. Logistic regression is widely used in the fairness domain as baseline model [36, 42–44, 68]. We used scikit-learn implementation with 'l2' regularization, 'lbfgs' solver, and maximum iteration of 1000.
- **Reweighing:** A widely used [22, 25, 40, 63, 78] pre-processing method proposed by Kamiran et al. [65]. Here, before model training, examples in each group, and label are given different weights to ensure fairness.
- **Meta Fair Classifier:** An in-processing method proposed by Celis et al. [37], which is a widely used meta algorithm [27, 38, 58, 74]. The optimization algorithm is developed to improve 11 fairness metrics with minimal loss in accuracy.

The last two bias mitigation algorithm implementations are taken from IBM AIF360 [26].

3.2 Agglomerative Clustering

Our metrics selection strategy, requires a clustering algorithm. Two class of such clustering algorithms are (a) partitioning clustering and (b) hierarchical clustering. Here we are grouping fairness metrics based on similarity, not on distance, and we have no prior idea about the number of clusters. Thus, in this case, the ideal choice is *hierarchical clustering*. Agglomerative clustering [5] is a hierarchical bottom-up clustering approach that is widely used in the ML community [23, 49–51, 54, 73, 77, 82, 95]. In this approach, the closest pairs of items are grouped together. These closest of these groups are then grouped into a higher-level group. This repeats until everything falls into one group. We used the average pairwise dissimilarity between objects in two different clusters as linkage method between groups. This process creates a dendrogram, a hierarchical structure of the groups/clusters obtained by between-cluster distance or dissimilarity. From this tree of groupings, we use the within-cluster similarity from the dendrogram, look for the largest distance that we can travel vertically without crossing any horizontal line [1, 47, 85], and extract the clusters at the largest change in SSE.

3.3 Spearman Rank Correlation

Figure 1 shows the dendrogram created for the classification metrics using the above described method. Table 4 shows that we get seven clusters from 26 classification metrics. Following a similar process for dataset metrics we get three clusters as shown in Table 5.

To build these clusters and dendrograms, we measure the similarity of two metrics. In this paper, by "similarity" we mean, they

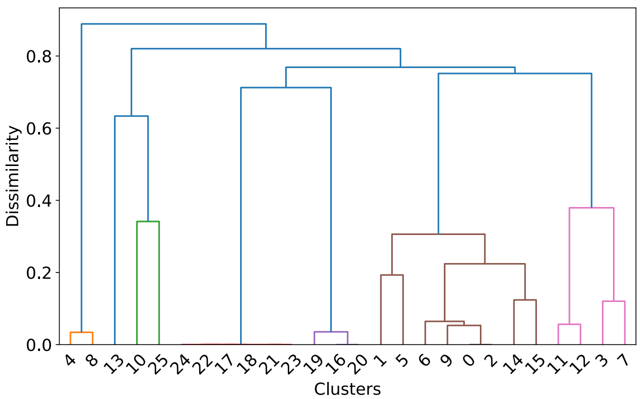

Figure 1: Agglomerative clustering of classification metrics (using Spearman rank correlation). Here x-axis shows the classification metric Ids from Table 1 and y-axis shows the dissimilarity measure between clusters.

are measuring the similar bias in the models/dataset. Such similar metrics will show a similar pattern of changes in bias when models are built using different parts of the data or different bias removal algorithms are used. To compute this similarity, we sample from our model training procedure (see §3.4.2) that computes our metrics 25 times, each time using different train/validation/test samples of the data. Next, for each dataset, for those 25 numbers, we use correlation to assess similarity.

Two widely used definitions of correlation [45, 49–51, 61, 76, 82, 95] are the (a) Pearson correlation (which evaluates the linear relationship between two continuous variables) and the (b) Spearman rank correlation (which is a non-parametric measure of rank correlation that evaluates the monotonic relationship between two continuous or ordinal variables). We choose Spearman rank correlation, as it measures the monotonic relationship between two variables and is less affected by outliers.

3.4 Experimental Setup

We summarize our experimental setup as follows.

3.4.1 ***Data Pre-processing****:* Three different pre-processing steps are performed before using the data [56, 72, 80] for model building. At first, each categorical value in the dataset is converted either using a label encoder or by one hot encoder. Then the protected attributes are changed into ones and zeros from their original values. Here we denote the privileged attribute as one and unprivileged as zero. Finally, we use min-max normalization in the datasets to normalize the data before building the models.

3.4.2 ***Model Training****:* We used five fold cross-validation repeated five times with random seeds build training/ test sets (as recommended by [66, 80, 87, 90]). This step is to divide the data into multiple subsets of data with various degrees of bias. We train three models in each iteration (a) *baseline model:* here we use the training data to build a logistic regression model; (b) *Reweighing model:* here we first train the reweighing method, then use the learned model to transform the training data to achieve group fairness. Using the transformed data, we train a logistic regression from scikit-learn with 'l2' regularization, 'lbfgs' solver and maximum iteration of 1000; and (c) *Meta Fair Classifier model:* here to train the meta fair classifier model, we use the training data to build multiple meta fair classifier model with different values of τ (a hyperparameter for fairness penalty in the model) and measure the bias in the model using the validation set. Then to build the final model, we select the τ for which the model had the lowest bias in the validation set and build the final meta fair classifier model.

3.4.3 ***Fairness Metric Calculation****:* We collect the performance of each model based on 26 classification and four dataset metrics for each iteration of the cross-validation. So for each iteration, we use the test data for prediction and then use the predicted values along with the ground truth to calculate the 26 classification metrics. Similarly, we collect the four dataset metrics on the baseline and reweighing method. Meta fair classifier is not applicable in the case of dataset metrics.

3.4.4 ***Measure for Fairness****:* Data Pre-processing, Model Training, and Fairness Metric Calculation steps are performed for each of seven datasets with five fold five repeat cross-validation. Then to measure if the model built on a dataset is fair or unfair according to a metric, we selected a threshold for each of the metrics. As mentioned in Section 2.2, that threshold is the *fair range*. If a metric value falls in that range, we say it "fair" otherwise "unfair".

3.4.5 ***Building Clusters****:* One of the main goals of this study is to group a set of metrics together that perform similarly and measure similar kinds of bias. We use 26 classification metrics calculated on seven datasets with three different methods to calculate metric to metric correlation based on Spearman rank correlation coefficient. We do the same for the four dataset metrics as well. This provides us two correlation matrices: one 26x26 and one 4x4. After that, to build the clusters using the agglomerative clustering, we convert the similarity matrix into a dissimilarity matrix [49, 61] using equation 1. We use this dissimilarity matrix to create the clusters. The agglomerative clustering process creates a dendrogram as shown in Figure 1. Now to select the number of clusters, we cut the dendrogram at a height, where the clusters will remain unchanged with the most increase/decrease of the cutting threshold. For classification metrics, we cut the dendrogram (Figure 1) at 0.57 as the clusters will remain unchanged between the cutoff value 0.49 and 0.64. Finally, we get the clusters containing classification metrics measuring similar kinds of bias. We perform the same process for dataset metrics and cut the dendrogram at a height of 0.4.

$$d(x, y) = 1 - |sim(x, y)| \tag{1}$$

3.4.6 ***Calculating Sensitivity****:* Research question four asks about the consistency of the metric values for three cases: (a) raw data, b) after applying Reweighing (RW), (c) after applying Meta Fair Classifier (MFC). As we are using five cross fold five repeats for all the datasets, we get 25 results for each dataset and report for all seven datasets:

- the median value: the 50th percentile (or Q_2);
- the IQR: the (75-25)th percentile (or $Q_3 - Q_1$)

Table 4: Cluster based results for 26 classification metrics on seven datasets. For a metric with ideal an value of zero, anything below -0.1 and above 0.1 is "unfair". For a metric with an ideal value of one, anything <0.8 or >1.2 is "unfair".

Cluster Id	MID	Metrics	Datasets: Adult	Compas	German	Health	Bank	Student	Titanic	Metric Type
0	C3	false_omission_rate_difference	Unfair	Fair	Fair	Unfair	Fair	Fair	Unfair	Mis-classification
0	C7	false_omission_rate_ratio	Unfair	Fair	Fair	Unfair	Fair	Unfair	Unfair	
0	C11	error_rate_difference	Unfair	Fair	Fair	Unfair	Fair	Fair	Fair	
0	C12	error_rate_ratio	Unfair	Fair	Fair	Unfair	Fair	Fair	Fair	
		Percentage of agreement	**100%**	**100%**	**100%**	**100%**	**100%**	**75%**	**50%**	
1	C10	average_abs_odds_difference	Unfair	Unfair	Unfair	Unfair	Unfair	Fair	Unfair	Differential Fairness
1	C25	differential_fairness_bias_amplification	Unfair	Unfair	Unfair	Unfair	Unfair	Fair	Unfair	
		Percentage of agreement	**100%**	**100%**	**100%**	**100%**	**100%**	**100%**	**100%**	
2	C16	generalized_entropy_index	Fair	Unfair	Fair	Fair	Fair	Fair	Unfair	Individual Fairness
2	C19	theil_index	Unfair	Unfair	Fair	Unfair	Unfair	Fair	Unfair	
2	C20	coefficient_of_variation	Unfair	Unfair	Unfair	Unfair	Unfair	Unfair	Unfair	
		Percentage of agreement	**67%**	**100%**	**67%**	**67%**	**67%**	**67%**	**100%**	
3	C4	false_discovery_rate_difference	Fair	Fair	Fair	Fair	Fair	Fair	Unfair	Mis-classification
3	C8	false_discovery_rate_ratio	Fair	Fair	Fair	Fair	Fair	Unfair	Unfair	
		Percentage of agreement	**100%**	**100%**	**100%**	**65%**	**100%**	**50%**	**100%**	
4	C0	true_positive_rate_difference	Unfair	Unfair	Fair	Unfair	Unfair	Fair	Unfair	Confusion Matrix Based Group Fairness
4	C1	false_positive_rate_difference	Fair	Unfair	Unfair	Unfair	Unfair	Fair	Unfair	
4	C2	false_negative_rate_difference	Unfair	Unfair	Unfair	Unfair	Unfair	Fair	Unfair	
4	C5	false_positive_rate_ratio	Fair	Unfair	Unfair	Unfair	Unfair	Fair	Unfair	
4	C6	false_negative_rate_ratio	Unfair	Unfair	Unfair	Unfair	Unfair	Unfair	Unfair	
4	C9	average_odds_difference	Unfair	Unfair	Unfair	Unfair	Unfair	Fair	Unfair	
4	C14	disparate_impact	Unfair	Unfair	Unfair	Unfair	Unfair	Unfair	Unfair	
4	C15	statistical_parity_difference	Unfair	Unfair	Unfair	Unfair	Unfair	Fair	Unfair	
		Percentage of agreement	**75%**	**100%**	**88%**	**100%**	**100%**	**75%**	**100%**	
5	C17	between_all_groups_generalized_entropy_index	Fair	Fair	Fair	Fair	Fair	Fair	Fair	Between Group Individual Fairness
5	C18	between_group_generalized_entropy_index	Fair	Fair	Fair	Fair	Fair	Fair	Fair	
5	C21	between_group_theil_index	Fair	Fair	Fair	Fair	Fair	Fair	Fair	
5	C22	between_group_coefficient_of_variation	Fair	Fair	Fair	Fair	Fair	Fair	Unfair	
5	C23	between_all_groups_theil_index	Fair	Fair	Fair	Fair	Fair	Fair	Fair	
5	C24	between_all_groups_coefficient_of_variation	Fair	Fair	Fair	Fair	Fair	Fair	Unfair	
		Percentage of agreement	**100%**	**100%**	**100%**	**100%**	**100%**	**100%**	**67%**	
6	C13	selection_rate	Unfair	Unfair	Unfair	Unfair	Unfair	Unfair	Unfair	Intermediate Metric
		Percentage of agreement	**100%**	**100%**	**100%**	**100%**	**100%**	**100%**	**100%**	
		Percentage of metrics marking dataset as unfair	**58%**	**54%**	**34%**	**65%**	**50%**	**23%**	**77%**	

Table 5: Cluster based results for four dataset metrics on seven datasets. For a metric with ideal value of zero, anything below -0.1 and above 0.1 is "unfair". For a metric with ideal value of one, anything <0.8 or >1.2 is "unfair".

Cluster Id	MID	Metrics	Datasets: Adult	Compas	German	Health	Bank	Student	Titanic	Metric Type
0	D0	consistency	Fair	Unfair	Fair	Unfair	Fair	Unfair	Fair	Individual Fairness
1	D1	smoothed_empirical_differential_fairness	Unfair	Unfair	Unfair	Unfair	Unfair	Unfair	Unfair	Differential Fairness
2	D2	mean_difference	Unfair	Unfair	Unfair	Fair	Unfair	Fair	Unfair	Confusion Matrix Based Group Fairness
2	D3	disparate_impact	Unfair	Unfair	Unfair	Fair	Unfair	Fair	Unfair	
		Percentage of metrics marking dataset as unfair	**75%**	**100%**	**75%**	**50%**	**75%**	**50%**	**75%**	

4 RESULTS

Our results are organized based on four research questions.

RQ1: Do current fairness metrics agree with each other?

At first, we need to verify our motivation. In real life, do the fairness metrics contradict? Table 4 contains results for 26 classification metrics; Table 5 contains results for four dataset metrics. The learner here is logistic regression. The last row contains the percentage of metrics marking the specific dataset as unfair in both tables. If we combine last rows of Table 4 & 5 and sort them in ascending order, we get the following list:

$$\{ 23, 34, 50, 50, 50, 54, 58, 65, 75, 75, 75, 75, 77, 100 \}\%$$

The median value here is 62%; i.e., nearly half the time the metrics make *different* conclusions about the *same* data. This means that researchers and practitioners will be spending much effort trying to understand their systems using disagreeing oracles (a result that motivates this entire paper).

RQ2: Can we group (cluster) fairness metrics based on similarity?

Table 4 shows that 26 classification metrics can be divided into seven clusters. Table 5 shows that four dataset metrics can be divided into three clusters. More importantly, we note that:

- RQ1 reported intra-project disagreement on "fair"-vs-"unfair";
- We note that there is much intra-cluster agreement for each data set in Table 4 and Table 5.

As evidence, we note that the majority fairness decision is always the same within the clusters for each dataset. In Table 4, the row *Percentage of agreement* comments on the uniformity of decisions within each cluster (for each dataset). Note that uniformity is very high (often 100%). That means metrics inside each cluster agree with each other for every dataset. Among the seven clusters, we see six clusters (except cluster two) show 100% agreement considering median value across seven datasets. For example, in case of cluster zero, percentage of agreement is 100% for five datasets; 75% for one; 50% for one. Majority is 100%. That is true for clusters 1,3,4,5,6 & 7. We see similar agreement pattern inside clusters in Table 5 also.

For reference purposes, the last column of Table 4 and Table 5 offers names for those clusters:

- **Misclassification**: these metrics try to measure the difference or ratio of misclassification errors between groups;
- **Differential fairness**: these metrics try to measure if probabilities of the outcomes are similar regardless of the combination of protected attributes [55];
- **Individual Fairness**: It measures if two similar individuals with respect to the classification task receive the same outcome or not;
- **Confusion matrix based group fairness**: these metrics measure difference or ratio between groups based on confusion matrix;
- **Between group individual fairness**: measures the difference or ratio of individual fairness between groups;
- **Intermediate metrics**: these are intermediate metrics.

From a practitioner viewpoint, this clustering is useful because:

- The clustering reduces the confusion of having too many metrics and not knowing their similarity.
- As the metrics inside the same cluster measure same kind of bias and behave in the same manner; we can choose just one metric from each cluster. Thus we measure a few metrics but can cover a much more comprehensive range of fairness notions.
- If we see agreement among all the metrics inside a cluster for a particular dataset, then one metric can be chosen as representative of the whole cluster.
- In case of intra-cluster conflicts, choosing only one metric can be risky. In these cases, practitioners need to do a proper risk assessment before selecting metrics. That said, if there is intra-cluster conflict among metrics, we can choose one from the 'fair' group and one from the 'unfair' group to mitigate that risk.

As part of this study, we further analyzed each cluster mathematically to verify if our cluster of metrics and their mathematical definitions coincide. A detailed analysis of these clusters and their mathematical analysis has been discussed in section 5.1.

RQ3: Are some fairness metrics more sensitive to change than others?

An ideal metric is responsive to the dataset it examines. An "insensitive" metric is one that delivers the same conclusions, no matter what data is being examined. An "insensitive" cluster is one containing mostly insensitive metrics. Such insensitive clusters could be ignored since they are not informative.

We measure sensitivity by looking at the variability of our metrics scores using the intra-quartile range (IQR=$Q_3 - Q1$). For each data set, we found the IQR across all clusters. Next, we highlight the sensitive results; i.e. those with an IQR greater that d*standard deviation. The remaining, unhighlighted results are the insensitive metrics.

As to what value of d to use in this analysis, we take the advice of a widely cited paper by Sawilowsky [79] (this 2009 paper has 1100 citations). That paper asserts that "small" and "medium" effects can be measured using $d = 0.2$ and $d = 0.5$ (respectively). We will analyze this data by splitting the difference looking for differences larger than $d = (0.5 + 0.2)/2 = 0.35$.

Turning now to Table 6 and Table 7 we see that most clusters have highlight IQR results. However, in Table 6, we see the clusters formed by metrics C16, C18, C20 (individual fairness) and C17, C18, C21, C22, C23, C24 (between group individual fairness) are insensitive. This, in turn, means that we should ***not*** criticize a fairness analysis that ignores these metrics.

RQ4: Can we achieve fairness based on all the metrics at the same time?

Different fairness metrics measure different kinds of bias. If any of the metrics complain about the fairness of the test results, then we can not trust the model blindly, and it should go through further scrutiny and improvement. Bias mitigation algorithms try to make unfair models fairer. Here we are verifying even after applying bias mitigation algorithms; can we achieve fairness based on all the metrics or not? We have chosen two highly cited algorithms from IBM AIF360: Reweighing (RW) by Kamiran et al. [65] and Meta Fair Classifier (MFR) by Celis et al [37].

Table 8 shows those results collected for seven datasets after using RW and MFC algorithms. For every dataset (row-wise), we show the number of metrics changed towards or away from its ideal value. In that table:

- FU denotes the metrics that changed towards ideal value;
- UF denotes the metrics that moved away from the ideal value,
- NC means the metrics which did not change.

Note that majority of the metrics move towards "fair", but there are some metrics that move towards "unfair". For Reweighing, some metrics show "no change", but we have verified they always remain in the *fair range.*

The main takeaway here is no longer necessary (or even possible) to satisfy all these fairness metrics. While our analysis can reduce dozens of metrics down to ten, there will still be issues of how to trade-off within this reduced set. Even after applying bias mitigation

Table 6: This table shows sensitivity of the classification metrics on the three different models used in this study (a) Baseline; (b) Reweighing(RW); and (c) Meta Fair Classifier(MFC). The table shows the median and IQR values of three datasets. Here the cells in IQR columns are marked with "red" those that change by more than a small amount (35th percentile of the standard deviation of the IQR values). The insensitive metrics are those that usually have white IQR values.

MID	Compas						Health						German					
	Baseline		RW		MFC		Baseline		RW		MFC		Baseline		RW		MFC	
	Med	IQR	Med	IQR	Med	IQR	Med	IQR	Med	IQR	Med	IQR	Med	IQR	Med	IQR	Med	IQR
C3	-0.067	0.079	-0.113	0.015	-0.061	0.035	-0.14	0.032	-0.211	0.068	-0.151	0.137	0	0.5	-0.5	0.667	0	0.592
C7	0.817	0.211	0.691	0.029	0.824	0.099	0.357	0.091	0.158	0.333	0.408	0.356	2.3	0.72	0	0.5	1	0.53
C11	-0.033	0.043	-0.016	0.039	-0.042	0.026	-0.108	0.01	-0.133	0.013	-0.098	0.111	0.059	0.074	0.059	0.114	0.049	0.062
C12	0.912	0.112	0.958	0.112	0.887	0.071	0.508	0.174	0.339	0.026	0.494	0.51	1.18	0.274	1.18	0.418	1.166	0.215
C10	0.252	0.058	0.029	0.02	0.181	0.035	0.162	0.103	0.106	0.087	0.161	0.064	0.221	0.167	0.043	0.048	0.031	0.121
C25	0.531	0.354	-0.22	0.141	0.359	0.148	0.193	0.249	-0.094	0.422	0.113	0.428	2.399	3.29	1.162	0.49	1.578	2.087
C16	0.193	0.001	0.189	0.012	0.192	0.007	0.091	0.004	0.087	0.017	0.091	0.025	0.076	0.011	0.071	0.011	0.066	0.011
C19	0.268	0.003	0.263	0.017	0.269	0.009	0.132	0.021	0.14	0.051	0.139	0.033	0.083	0.02	0.073	0.017	0.064	0.02
C20	0.878	0.003	0.87	0.027	0.876	0.016	0.602	0.015	0.589	0.058	0.602	0.082	0.553	0.041	0.532	0.041	0.513	0.043
C4	0.04	0.034	0.138	0.051	0.037	0.058	-0.091	0.133	-0.009	0.202	-0.016	0.152	0.059	0.135	0.059	0.114	0.045	0.064
C8	1.109	0.089	1.376	0.143	1.098	0.156	0	0.944	0.964	1.571	0.925	1.292	2.6	0.542	1.18	0.459	1.156	0.233
C0	-0.273	0.087	-0.004	0.052	-0.212	0.048	-0.106	0.139	0.13	0.227	-0.117	0.405	-0.077	0.092	0	0.038	-0.017	0.062
C1	-0.186	0.052	-0.014	0.02	-0.17	0.031	-0.214	0.053	-0.13	0.174	-0.109	0.114	-0.3	0.233	0	0.029	-0.053	0.176
C2	0.273	0.087	0.004	0.052	0.212	0.048	0.106	0.139	-0.13	0.227	0.117	0.405	0.077	0.092	0	0.038	0.017	0.062
C5	0.408	0.037	0.956	0.069	0.471	0.068	0	0.219	0.25	0.654	0.191	0.347	0.7	0.228	1	0.029	0.947	0.176
C6	1.71	0.242	1.009	0.124	1.514	0.133	1.467	0.5	0.429	1.222	1.453	2.056	3.4	0.56	0	5.459	11.532	3.4
C9	-0.252	0.058	-0.018	0.059	-0.181	0.035	-0.162	0.103	-0.06	0.158	-0.139	0.165	-0.221	0.167	0	0.043	-0.031	0.121
C14	0.432	0.073	0.89	0.133	0.541	0.062	0.314	0.14	0.435	0.322	0.387	0.274	0.836	0.12	1	0.045	0.971	0.105
C15	-0.264	0.05	-0.049	0.059	-0.205	0.03	-0.356	0.079	-0.289	0.179	-0.298	0.18	-0.164	0.122	0	0.043	-0.029	0.104
C17	0.002	0.002	0.001	0.002	0.001	0.001	0.001	0.003	0.002	0.001	0.001	0.001	0.001	0.002	0.002	0.001	0.001	0.001
C18	0.002	0.002	0.001	0.002	0.001	0.001	0.001	0.001	0.002	0.001	0.001	0.001	0.001	0.002	0.002	0.001	0.001	0.001
C21	0.002	0.002	0.001	0.003	0.001	0.001	0.003	0.001	0.002	0.001	0.001	0.001	0.003	0.002	0.003	0.001	0.002	0.001
C22	0.088	0.049	0.052	0.006	0.057	0.019	0.036	0.037	0.017	0.054	0.045	0.03	0.029	0.063	0.03	0.05	0.036	0.038
C23	0.002	0.002	0.001	0	0.001	0.001	0.003	0.003	0.006	0.001	0.001	0.001	0.001	0.002	0.001	0.001	0.004	0.001
C24	0.088	0.049	0.052	0.006	0.057	0.019	0.036	0.037	0.017	0.054	0.045	0.03	0.029	0.063	0.03	0.05	0.036	0.038
C13	0.405	0.025	0.436	0.016	0.41	0.017	0.407	0.05	0.39	0.133	0.411	0.056	0.945	0.015	0.975	0.03	0.99	0.047

Table 7: This table is similar to Table 6, showing the sensitivity of the dataset metrics on (a) Baseline; (b) Reweighing (RW).

MID	Compas						Health						German					
	Baseline		RW		MFC		Baseline		RW		MFC		Baseline		RW		MFC	
	Med	IQR	Med	IQR	Med	IQR	Med	IQR	Med	IQR	Med	IQR	Med	IQR	Med	IQR	Med	IQR
D1	0.568	0.021	0.568	0.021	-	-	0.804	0.02	0.804	0.02	-	-	0.632	0.008	0.632	0.008	-	-
D2	0.252	0.043	0	0	-	-	0.868	0.322	0.001	0	-	-	0.298	0.105	0.002	0	-	-
D3	-0.105	0.016	0	0	-	-	-0.313	0.067	0	0	-	-	-0.097	0.033	0	0	-	-
D4	0.777	0.033	1	0	-	-	0.411	0.137	1	0	-	-	0.865	0.043	1	0	-	-

Table 8: This table shows the number of classification metrics that move towards or away from the ideal value when either Reweighing or Meta Fair Classifier is used to remove bias in the models. Here "UF" shows the number of metrics that moved towards the ideal metric value, while "FU" shows the opposite. Finally "NC" shows the number of metrics that did not change at all.

Dataset	Reweighing (RW)			Meta Fair Classifier (MFC)		
	UF	FU	NC	UF	FU	NC
Adult	13	13	0	11	15	0
Compas	15	7	4	16	6	4
Health	17	5	4	17	7	1
German	19	6	1	19	7	0
bank	16	6	4	15	7	4
Titanic	11	15	0	17	9	0
Student	15	7	4	12	10	4

approaches, some metrics still conflict with others. This finding is similar to the claim made by others:

- Berk et al. [28] offer an "Impossibility Theorem" that says there is no way to satisfy all kinds of fairness together.
- As Yuriy Brun said at his keynote at ICSSP'2020 "*we need to work the system in a biased way sometimes*" [33].

5 DISCUSSION

We have described all of our results. Here we are summarizing the results in a comprehensible way to reach a stable conclusion. The main idea of this work is to reduce the complexity of measuring fairness. That said, it is imperative we narrate our conclusions in a very easy way. We discuss here three major concerns that arise from §4 and try to simplify fairness measurement to our best.

5.1 Why Not Group Metrics via their Analytical Structure?

This paper has offered an empirical analysis that many of the metrics in Table 4 are synonymous since, when clustered, they fell together into just a few similar groups. In this section, we check if the same conclusions can be achieved from a more analytical analysis that looked at the structure of the equations for the fairness metrics.

Sometimes, a group generated by formula's analytical structure is similar to the clusters we generated above. For example:

- In cluster three (from Table 4), all metrics are based on *FDR*, which suggests that both from an empirical and analytical point of view, they should be similar.
- Also, In cluster zero, we see that all those metrics are based on *FOR* and error rate. Intuitively, this seems sensible since here metrics try to measure amount of misclassification.

That said, as shown by the following three examples, there are many examples where an equation's analytical structure does *not* predict for its empirical cluster.

EXAMPLE #1: If we look at cluster five, all six metrics inside this cluster are related to "between group individual fairness". This metric is based on the same benefit function:

$$y = \hat{y} - y + 1 \tag{2}$$

(For more details on that. see Table 1 metric id C16). We note that cluster two is also based on Equation 2, but the metrics inside this cluster represent individual fairness for each group separately. That means

> *Although all metrics inside cluster two and cluster five are based on the same benefit function, they measure different definitions of fairness.*

That is, a formal analysis of the analysis might combine these clusters, whereas a data-oriented empirical analysis would argue for their separation.

EXAMPLE #2: In cluster four from Table 1, the metrics C0, C1, C2, C5, C6 and C9 dependent on *TPR*, *FPR* and *FNR*. Recall that *FPR* and *FNR* report type one and type two errors (misclassification on fairness); Now *TPR* can be expressed as *1 - FPR*, which means the change in *TPR* will mirror changes in *FPR*. In contrast, in this cluster, the other two metrics C14 and C15 are based on selection rate (ratio of number of predicted positive and number of instances). Although there is not much similarity in the formula between these two and other metrics in this cluster, we can see they perform similarly when measuring fairness. That is:

> *An analytical analysis does not always reflect the measurement of fairness in the real world scenario.*

Verma et al. [91] in their paper notice a similar phenomenon where they observe that: *Equal Predictive parity (a measure they explore) should also have equal* FDR *... but when measured from an empirical point of view, they showed they are not the same.*

EXAMPLE #3: In cluster one, metrics C10 and C25 have very different mathematical formulas. C10 is based on *FPR* while C25 is based on smoothed EDF– the Empirical differential fairness. *EDF* is calculated based on Dirichlet smoothed base rates for each intersecting group in the dataset, which is based on count of predicted positive. Here as well, we see that

> *Two formulas with a different analytical structure can have a similar performance w.r.t. fairness.*

To summarize the above, we quote Alfred Korzybski, who warned:

A map is not the territory.

While the analytical structure of the formula offers intuitions about the nature of fairness, those intuitions had better be checked via empirical analysis.

5.2 Is our Empirical Analysis Useful?

We have established the requirement of empirical analysis and we have also done that analysis. We need to find out whether this analysis would be helpful in real-life applications or not. Here we describe various scenarios of fairness contradiction and how our study helps to remove that.

Imagine a college admission decision scenario, where the system might be seen as biased against group B if applicants from group A are accepted more than group B. Here group A and group B are divided based on different values of a protected attribute. The college applies a bias mitigation approach to solve this problem using a group fairness metric by changing group A's or B's scoring threshold. Now, if a member of group A is rejected, while a member of group B has been accepted with an equal or lower score, then the system might be seen as biased against that individual. The main takeaway from this story is that there is a conflict between "individual fairness" and "group fairness" [30].

The concept of fairness is very much application-specific and choosing the appropriate metric is the sole responsibility of the policymaker. An ideal scenario will be building a machine learning model which does not show any kind of bias. However, that is too good to be true. Brun et al. found out that if a model is adjusted to be fair based on one protected attribute (e.g., sex), in some cases model becomes more biased based on another protected attribute (e.g., race) [14]. Kleinberg and other researchers argue that different notions of fairness are incompatible with each other and hence it is impossible to satisfy all kinds of fairness simultaneously [70]. Here one thing to remember while doing prediction is that fairness is not the only concern. Prediction performance is the most important goal. Berk et al. found out that accuracy and fairness are competing goals [29]. This trade-off makes the job even more complicated since damaging model performance while making it fair may be unacceptable.

As researchers, we know that satisfying all kinds of fairness together is not possible. A policymaker has to choose which fairness definitions are most important for the particular domain and ignore the rest. Our work of dividing fairness tries to make the choice easier, as choosing metrics from a group of 10 options is much simpler than choosing from 30 choices. Using our results of Table 4 and Table 5, in a specific domain, if group fairness is more important than individual fairness, then cluster four will be given more priority than clusters two and five (Table 4). Once a cluster is given priority, one or two metrics can be chosen to represent the whole cluster. That means our whole work boils down to minimizing the number of metrics to look at and covering a wide range of fairness. We believe future researchers and industry practitioners will use our work as a guide and that will be the fulfillment of this study.

5.3 What to do when the metrics contradict each other?

We have seen that there are scenarios where fairness metrics contradict each other. According to some metrics, the prediction is fair, where some other metrics disagree. Fairness metrics find out how critical the errors of a prediction model are. It is the decision of the policymaker or the domain expert to choose appropriate fairness metrics based on what kind of bias is more important for the specific domain. For example, consider the following two scenarios:

- Suppose we are predicting if a patient has cancer or not, depending on the symptoms. Here predicting a benign case as malignant is not very dangerous but predicting a malignant case as benign is extremely dangerous. A wrong diagnosis for an actual cancer patient will delay the treatment, and the patient may die. That means *false negative* is more important here.
- Suppose we are predicting if future performance of a student based on previous records. Here if we predict a good student as bad, that is not that fatal. However, if a student who really needs special attention and help from teachers, is given a good rating then it will be misery for that student. That means *false positive* is more important here.

If we know which metrics look at what kind of error, it will be easier for the decision-maker to choose. That said, based on the guidance we have provided, in case of contradiction among metrics, one metric over another will be given priority.

6 THREATS TO VALIDITY & FUTURE WORK

This paper explores machine learning methods for software engineering. One issue with any paper like this is a few selection and evaluation biases along with construct and external validity based on the choice of models, datasets, and methods. In the future, we plan to address the apparent threats to validity that this paper has not fully addressed.

Construct Validity: Here, we have used popular *hierarchical* clustering called *agglomerative* approach, as the number of clusters were not known beforehand. In future, we need to experiment with other clustering techniques to check for conclusion stability. This analysis used logistic regression (LR), as much prior work on fairness has also used LR [24, 42]. Nevertheless, in future work, we need to explore some other classification models including DL models. Also, the metric clusters found in Table 4 and Table 5 are created using the results of our choice ML models, dissimilarity measures, and cutting point in the dendrogram. Thus, choosing one metric from each cluster may contain some risk, and researchers need to be careful while making informed choices about metric selection.

Evaluation Bias: We have used 30 metrics taken from IBM AIF360 [24]. We have also covered most of the metrics from Fairkit-learn [62] and Fairlearn [20]. There are other metrics and definitions of fairness, thus the results of this study may not generalize to all available metrics. But the 30 metrics covered in this study are widely used in the fairness domain [31, 46, 56, 69, 94].

External Validity: We have used seven datasets. In the fairness domain, one big challenge is the availability of adequate datasets. It would be insightful to re-run this study on new datasets and also on other domains.

Sampling Bias: In this work we used thresholds recommended by IBM AIF360 (“fair” means -0.1,0.1 or 0.8,1.2 for different kinds of metric). Future work should explore the sensitivity of our conclusions to changes in those thresholds.

Another issue with sampling bias is that our analysis is based on the data of Table 2. We recommend that when new data becomes available, we test the conclusions of this paper against that new data. That would not be an arduous task (and to simplify that task, we have placed all our scripts online in order).

7 CONCLUSION

Fairness is a rapidly evolving domain and the number of fairness metrics is increasing exponentially. While performing our literature review we saw the current practice in this domain is to relying on a handful of metrics and ignoring the rest. But which metrics can be ignored? Which are essential?

To answer these questions, this paper has experimented with the following *metrics selection tactic*: When applied, the paper reported that this tactic could reduce dozens of metrics to just a handful. We found:

- RQ1 showed that all the metrics do not agree with each other when labeling a model as fair or unfair.
- RQ2 showed that metrics can be clustered together based on how they measure bias. Each of the resultant clusters measures different types of bias and selecting one metric from each cluster should be representative enough to measure increase or decrease in bias in other metrics in the same cluster.
- RQ3 showed that we could ignore at least two of those clusters, since they were not “sensitive”. Recall that by “insensitive” clusters, we mean those where changes to the data did not change the fairness scores.
- RQ4 showed this reduced set actually predicts for different things. That said, it is no longer necessary (or even possible) to satisfy all these fairness metrics.

From these results, we argue that:

- There are many spurious fairness metrics; i.e. metrics that measure very similar things.
- To simplify fairness testing, just (a) determine what type of fairness is desirable (for a list of types, see Table 4 and Table 5); then (b) look up those types in our clusters; then (c) just test for one item per cluster.
- While this approach does not completely remove all issues with fairness testing, it does reduce a very complex problem of (say) 30 metrics to a much smaller and manageable set.
- Also, the methods of this paper could be used as a litmus test to prune away spurious new metrics that merely report the same thing as existing metrics.

ACKNOWLEDGEMENTS

The work was partially funded by blinded for review.

REFERENCES

[1] 1953. Stanford Hlab. (1953). https://hlab.stanford.edu/brian/number_of_clusters_.html
[2] 1994. UCI:Adult Data Set. (1994). http://mlr.cs.umass.edu/ml/datasets/Adult
[3] 2000. UCI:Statlog (German Credit Data) Data Set. (2000). https://archive.ics.uci.edu/ml/datasets/Statlog+(German+Credit+Data)
[4] 2001. UCI:Heart Disease Data Set. (2001). https://archive.ics.uci.edu/ml/datasets/Heart+Disease
[5] 2011. sklearn.cluster.AgglomerativeClustering. https://scikit-learn.org/stable/modules/generated/sklearn.cluster.AgglomerativeClustering.html
[6] 2014. Student Performance Data Set. (2014). https://archive.ics.uci.edu/ml/datasets/Student+Performance
[7] 2015. propublica/compas-analysis. (2015). https://github.com/propublica/compas-analysis
[8] 2016. Machine Bias: There's software used across the country to predict future criminals. And it's biased against blacks. (2016). https://www.propublica.org/article/machine-bias-risk-assessments-in-criminal-sentencing
[9] 2017. Bank Marketing UCI. (2017). https://www.kaggle.com/c/bank-marketing-uci
[10] 2017. Titanic: Machine Learning from Disaster. (2017). https://www.kaggle.com/c/titanic/data
[11] 2018. AI Fairness 360: An Extensible Toolkit for Detecting, Understanding, and Mitigating Unwanted Algorithmic Bias. (10 2018). https://github.com/IBM/AIF360
[12] 2018. Amazon scraps secret AI recruiting tool that showed bias against women. (Oct 2018). https://www.reuters.com/article/us-amazon-com-jobs-automation-insight/amazon-scraps-secret-ai-recruiting-tool-that-showed-bias-against-women-idUSKCN1MK08G
[13] 2018. Ethics Guidelines for Trustworthy Artificial Intelligence. https://ec.europa.eu/digital-single-market/en/news/ethics-guidelines-trustworthy-ai
[14] 2018. FAIRWARE 2018:International Workshop on Software Fairness. (2018). http://fairware.cs.umass.edu/
[15] 2018. Health care start-up says A.I. can diagnose patients better than humans can, doctors call that 'dubious'. *CNBC* (June 2018). https://www.cnbc.com/2018/06/28/babylon-claims-its-ai-can-diagnose-patients-better-than-doctors.html
[16] 2019. Ethically-Aligned Design: A Vision for Prioritizing Human Well-Begin with Autonomous and Intelligence Systems.
[17] 2019. EXPLAIN 2019. (2019). https://2019.ase-conferences.org/home/explain-2019
[18] 2019. Microsoft AI principles. 2019. https://blogs.microsoft.com/eupolicy/artificial-intelligence-ethics/
[19] 2020. Improving the enrollment process through machine learning. https://www.ellucian.com/insights/improving-enrollment-process-through-machine-learning
[20] 2021. Fairlearn. https://fairlearn.org/
[21] Aniya Aggarwal, Pranay Lohia, Seema Nagar, Kuntal Dey, and Diptikalyan Saha. 2019. Black Box Fairness Testing of Machine Learning Models. In *Proceedings of the 2019 27th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering* (Tallinn, Estonia) *(ESEC/FSE 2019)*. ACM, New York, NY, USA, 625–635. https://doi.org/10.1145/3338906.3338937
[22] Ashrya Agrawal, Florian Pfisterer, Bernd Bischl, Jiahao Chen, Srijan Sood, Sameena Shah, Francois Buet-Golfouse, Bilal A Mateen, and Sebastian J Vollmer. 2020. Debiasing classifiers: is reality at variance with expectation? *Available at SSRN 3711681* (2020).
[23] Gabriele Bavota, Sebastiano Panichella, Nikolaos Tsantalis, Massimiliano Di Penta, Rocco Oliveto, and Gerardo Canfora. 2014. Recommending refactorings based on team co-maintenance patterns. In *Proceedings of the 29th ACM/IEEE international conference on Automated software engineering*. 337–342.
[24] Rachel Bellamy, Kuntal Dey, Michael Hind, Samuel C. Hoffman, Stephanie Houde, Kalapriya Kannan, Pranay Lohia, Jacquelyn Martino, Sameep Mehta, Aleksandra Mojsilovic, Seema Nagar, Karthikeyan Natesan Ramamurthy, John Richards, Diptikalyan Saha, Prasanna Sattigeri, Moninder Singh, Ramazon Kush, and Yunfeng Zhang. 2018. AI Fairness 360: An Extensible Toolkit for Detecting, Understanding, and Mitigating Unwanted Algorithmic Bias. (10 2018).
[25] Rachel KE Bellamy, Kuntal Dey, Michael Hind, Samuel C Hoffman, Stephanie Houde, Kalapriya Kannan, Pranay Lohia, Jacquelyn Martino, Sameep Mehta, Aleksandra Mojsilović, et al. 2019. AI Fairness 360: An extensible toolkit for detecting and mitigating algorithmic bias. *IBM Journal of Research and Development* 63, 4/5 (2019), 4–1.
[26] Rachel K. E. Bellamy, Kuntal Dey, Michael Hind, Samuel C. Hoffman, Stephanie Houde, Kalapriya Kannan, Pranay Lohia, Jacquelyn Martino, Sameep Mehta, Aleksandra Mojsilovic, Seema Nagar, Karthikeyan Natesan Ramamurthy, John Richards, Diptikalyan Saha, Prasanna Sattigeri, Moninder Singh, Kush R. Varshney, and Yunfeng Zhang. 2018. AI Fairness 360: An Extensible Toolkit for Detecting, Understanding, and Mitigating Unwanted Algorithmic Bias. https://arxiv.org/abs/1810.01943
[27] Suman K Bera, Deeparnab Chakrabarty, Nicolas J Flores, and Maryam Negahbani. 2019. Fair algorithms for clustering. *arXiv preprint arXiv:1901.02393* (2019).
[28] Richard Berk, Hoda Heidari, Shahin Jabbari, Michael Kearns, and Aaron Roth. 2017. Fairness in Criminal Justice Risk Assessments: The State of the Art. arXiv:1703.09207 [stat.ML]
[29] Richard Berk, Hoda Heidari, Shahin Jabbari, Michael Kearns, and Aaron Roth. 2017. Fairness in Criminal Justice Risk Assessments: The State of the Art. *Sociological Methods & Research* (03 2017). https://doi.org/10.1177/0049124118782533
[30] Reuben Binns. 2019. On the Apparent Conflict Between Individual and Group Fairness. arXiv:1912.06883 [cs.LG]
[31] Sumon Biswas and Hridesh Rajan. 2020. Do the machine learning models on a crowd sourced platform exhibit bias? an empirical study on model fairness. In *Proceedings of the 28th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering*. 642–653.
[32] Su Lin Blodgett and Brendan O'Connor. 2017. Racial Disparity in Natural Language Processing: A Case Study of Social Media African-American English. arXiv:1707.00061 [cs.CY]
[33] Yuriy Brun. 2020. Preventing Undesirable Behavior of Intelligent Machines (ICSSP and ICGSE 2020 Keynote). (2020). https://www.youtube.com/watch?v=6M2Y3EG4fik&start=835s&ab_channel=YuriyBrun
[34] Aylin Caliskan, Joanna J. Bryson, and Arvind Narayanan. 2017. Semantics derived automatically from language corpora contain human-like biases. *Science* 356, 6334 (2017), 183–186. https://doi.org/10.1126/science.aal4230 arXiv:https://science.sciencemag.org/content/356/6334/183.full.pdf
[35] Flavio Calmon, Dennis Wei, Bhanukiran Vinzamuri, Karthikeyan Natesan Ramamurthy, and Kush R Varshney. 2017. Optimized Pre-Processing for Discrimination Prevention. In *Advances in Neural Information Processing Systems 30*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (Eds.). Curran Associates, Inc., 3992–4001. http://papers.nips.cc/paper/6988-optimized-pre-processing-for-discrimination-prevention.pdf
[36] Flavio Calmon, Dennis Wei, Bhanukiran Vinzamuri, Karthikeyan Natesan Ramamurthy, and Kush R Varshney. 2017. Optimized pre-processing for discrimination prevention. In *Advances in Neural Information Processing Systems*. 3992–4001.
[37] L. Elisa Celis, Lingxiao Huang, Vijay Keswani, and Nisheeth K. Vishnoi. 2020. Classification with Fairness Constraints: A Meta-Algorithm with Provable Guarantees. arXiv:1806.06055 [cs.LG]
[38] L Elisa Celis and Vijay Keswani. 2019. Improved adversarial learning for fair classification. *arXiv preprint arXiv:1901.10443* (2019).
[39] L. Elisa Celis, Damian Straszak, and Nisheeth K. Vishnoi. 2018. Ranking with Fairness Constraints. arXiv:1704.06840 [cs.DS]
[40] Juliana Cesaro and Fabio Gagliardi Cozman. 2019. Measuring Unfairness Through Game-Theoretic Interpretability. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, 253–264.
[41] Joymallya Chakraborty, Suvodeep Majumder, and Tim Menzies. 2021. Bias in Machine Learning Software: Why? How? What to Do?. In *Proceedings of the 29th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering* (Athens, Greece) *(ESEC/FSE 2021)*. Association for Computing Machinery, New York, NY, USA, 429–440. https://doi.org/10.1145/3468264.3468537
[42] Joymallya Chakraborty, Suvodeep Majumder, Zhe Yu, and Tim Menzies. 2020. Fairway: A Way to Build Fair ML Software. (2020), 654–665. https://doi.org/10.1145/3368089.3409697
[43] J. Chakraborty, K. Peng, and T. Menzies. 2020. Making Fair ML Software using Trustworthy Explanation. In *2020 35th IEEE/ACM International Conference on Automated Software Engineering (ASE)*. 1229–1233.
[44] Joymallya Chakraborty, Tianpei Xia, Fahmid M. Fahid, and Tim Menzies. 2019. Software Engineering for Fairness: A Case Study with Hyperparameter Optimization. arXiv:1905.05786 [cs.SE]
[45] Tse-Hsun Chen, Mark D Syer, Weiyi Shang, Zhen Ming Jiang, Ahmed E Hassan, Mohamed Nasser, and Parminder Flora. 2017. Analytics-driven load testing: An industrial experience report on load testing of large-scale systems. In *2017 IEEE/ACM 39th International Conference on Software Engineering: Software Engineering in Practice Track (ICSE-SEIP)*. IEEE, 243–252.
[46] Andrew Cotter, Heinrich Jiang, Maya R Gupta, Serena Wang, Taman Narayan, Seungil You, and Karthik Sridharan. 2019. Optimization with Non-Differentiable Constraints with Applications to Fairness, Recall, Churn, and Other Goals. *Journal of Machine Learning Research* 20, 172 (2019), 1–59.
[47] David Dann, Matthias Hauser, and Jannis Hanke. 2017. Reconstructing the giant: Automating the categorization of scientific articles with deep learning techniques. *Proceedings der 13. Internationalen Tagung Wirtschaftsinformatik* (2017), 1538–1549.
[48] R. H. DAVIS, D. B. EDELMAN, and A. J. GAMMERMAN. 1992. Machine-learning algorithms for credit-card applications. *IMA Journal of Management Mathematics* 4, 1 (01 1992), 43–51. https://doi.org/10.1093/imaman/4.1.43 arXiv:https://academic.oup.com/imaman/article-pdf/4/1/43/6764690/4-1-43.pdf
[49] Patrik D'haeseleer. 2005. How does gene expression clustering work? *Nature biotechnology* 23, 12 (2005), 1499–1501.
[50] William Dickinson, David Leon, and A Fodgurski. 2001. Finding failures by cluster analysis of execution profiles. In *Proceedings of the 23rd International*

Conference on Software Engineering. ICSE 2001. IEEE, 339–348.

[51] Jin Hwan Do, D Choi, et al. 2008. Clustering approaches to identifying gene expression patterns from DNA microarray data. *Molecules and cells* 25, 2 (2008), 279.

[52] Sanjiv Das et al. 2020. Fairness Measures for Machine Learning in Finance. *AWS Cloud* (Oct 2020). https://pages.awscloud.com/rs/112-TZM-766/images/Fairness.Measures.for.Machine.Learning.in.Finance.pdf

[53] Michael Feldman, Sorelle Friedler, John Moeller, Carlos Scheidegger, and Suresh Venkatasubramanian. 2015. Certifying and removing disparate impact. arXiv:1412.3756 [stat.ML]

[54] Marios Fokaefs, Nikolaos Tsantalis, Eleni Stroulia, and Alexander Chatzigeorgiou. 2011. JDeodorant: identification and application of extract class refactorings. In *2011 33rd International Conference on Software Engineering (ICSE)*. IEEE, 1037–1039.

[55] J. Foulds, Rashidul Islam, Kamrun Keya, and Shimei Pan. 2019. Differential Fairness.

[56] Sorelle A Friedler, Carlos Scheidegger, Suresh Venkatasubramanian, Sonam Choudhary, Evan P Hamilton, and Derek Roth. 2019. A comparative study of fairness-enhancing interventions in machine learning. In *Proceedings of the conference on fairness, accountability, and transparency*. 329–338.

[57] Sainyam Galhotra, Yuriy Brun, and Alexandra Meliou. 2017. Fairness testing: testing software for discrimination. *Proceedings of the 2017 11th Joint Meeting on Foundations of Software Engineering - ESEC/FSE 2017* (2017). https://doi.org/10.1145/3106237.3106277

[58] Vincent Grari, Boris Ruf, Sylvain Lamprier, and Marcin Detyniecki. 2019. Fair adversarial gradient tree boosting. In *2019 IEEE International Conference on Data Mining (ICDM)*. IEEE, 1060–1065.

[59] Ben Green and Yiling Chen. 2019. Disparate Interactions: An Algorithm-in-the-Loop Analysis of Fairness in Risk Assessments. In *Proceedings of the Conference on Fairness, Accountability, and Transparency* (Atlanta, GA, USA) *(FAT* '19)*. Association for Computing Machinery, New York, NY, USA, 90–99. https://doi.org/10.1145/3287560.3287563

[60] J. Henry Hinnefeld, Peter Cooman, Nat Mammo, and Rupert Deese. 2018. Evaluating Fairness Metrics in the Presence of Dataset Bias. arXiv:1809.09245 [cs.LG]

[61] Bipul Hossen, Hoque A Siraj-Ud-Doulah, and Aminul Hoque. 2015. Methods for evaluating agglomerative hierarchical clustering for gene expression data: A comparative study. *Computational Biology and Bioinformatics* 3, 6 (2015), 88–94.

[62] Brittany Johnson, Jesse Bartola, Rico Angell, Katherine Keith, Sam Witty, Stephen J. Giguere, and Yuriy Brun. 2020. Fairkit, Fairkit, on the Wall, Who's the Fairest of Them All? Supporting Data Scientists in Training Fair Models. arXiv:2012.09951 [cs.LG]

[63] Gareth P Jones, James M Hickey, Pietro G Di Stefano, Charanpal Dhanjal, Laura C Stoddart, and Vlasios Vasileiou. 2020. Metrics and methods for a systematic comparison of fairness-aware machine learning algorithms. *arXiv preprint arXiv:2010.03986* (2020).

[64] Nathan Kallus and Angela Zhou. 2018. Residual Unfairness in Fair Machine Learning from Prejudiced Data. arXiv:1806.02887 [stat.ML]

[65] Faisal Kamiran and Toon Calders. 2012. Data preprocessing techniques for classification without discrimination. *Knowledge and Information Systems* 33, 1 (01 Oct 2012), 1–33. https://doi.org/10.1007/s10115-011-0463-8

[66] Faisal Kamiran and Toon Calders. 2012. Data preprocessing techniques for classification without discrimination. *Knowledge and Information Systems* 33, 1 (2012), 1–33.

[67] Faisal Kamiran, Sameen Mansha, Asim Karim, and Xiangliang Zhang. 2018. Exploiting Reject Option in Classification for Social Discrimination Control. *Inf. Sci.* (2018). https://doi.org/10.1016/j.ins.2017.09.064

[68] Toshihiro Kamishima, Shotaro Akaho, Hideki Asoh, and Jun Sakuma. 2012. Fairness-Aware Classifier with Prejudice Remover Regularizer. In *Machine Learning and Knowledge Discovery in Databases*, Peter A. Flach, Tijl De Bie, and Nello Cristianini (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 35–50.

[69] Michael Kearns, Seth Neel, Aaron Roth, and Zhiwei Steven Wu. 2018. Preventing fairness gerrymandering: Auditing and learning for subgroup fairness. In *International Conference on Machine Learning*. PMLR, 2564–2572.

[70] Jon Kleinberg, Sendhil Mullainathan, and Manish Raghavan. 2016. Inherent Trade-Offs in the Fair Determination of Risk Scores. arXiv:1609.05807 [cs.LG]

[71] Allison Koenecke, Andrew Nam, Emily Lake, Joe Nudell, Minnie Quartey, Zion Mengesha, Connor Toups, John R. Rickford, Dan Jurafsky, and Sharad Goel. 2020. Racial disparities in automated speech recognition. *Proceedings of the National Academy of Sciences* 117, 14 (2020), 7684–7689. https://doi.org/10.1073/pnas.1915768117 arXiv:https://www.pnas.org/content/117/14/7684.full.pdf

[72] Preethi Lahoti, Krishna P Gummadi, and Gerhard Weikum. 2019. ifair: Learning individually fair data representations for algorithmic decision making. In *2019 IEEE 35th International Conference on Data Engineering (ICDE)*. IEEE, 1334–1345.

[73] Qingwei Lin, Hongyu Zhang, Jian-Guang Lou, Yu Zhang, and Xuewei Chen. 2016. Log clustering based problem identification for online service systems. In *2016 IEEE/ACM 38th International Conference on Software Engineering Companion (ICSE-C)*. IEEE, 102–111.

[74] Kirtan Padh, Diego Antognini, Emma Lejal Glaude, Boi Faltings, and Claudiu Musat. 2020. Addressing Fairness in Classification with a Model-Agnostic Multi-Objective Algorithm. *arXiv preprint arXiv:2009.04441* (2020).

[75] Geoff Pleiss, Manish Raghavan, Felix Wu, Jon Kleinberg, and Kilian Q. Weinberger. 2017. On Fairness and Calibration. arXiv:1709.02012 [cs.LG]

[76] Pablo D Reeb, Sergio J Bramardi, and Juan P Steibel. 2015. Assessing dissimilarity measures for sample-based hierarchical clustering of RNA sequencing data using plasmode datasets. *PLoS One* 10, 7 (2015), e0132310.

[77] Pedro Pereira Rodrigues, Joao Gama, and Joao Pedroso. 2008. Hierarchical clustering of time-series data streams. *IEEE transactions on knowledge and data engineering* 20, 5 (2008), 615–627.

[78] Prasanna Sattigeri, Samuel C Hoffman, Vijil Chenthamarakshan, and Kush R Varshney. 2019. Fairness GAN: Generating datasets with fairness properties using a generative adversarial network. *IBM Journal of Research and Development* 63, 4/5 (2019), 3–1.

[79] Shlomo S Sawilowsky. 2009. New effect size rules of thumb. *Journal of Modern Applied Statistical Methods* 8, 2 (2009), 26.

[80] Sebastian Schelter, Yuxuan He, Jatin Khilnani, and Julia Stoyanovich. 2019. Fairprep: Promoting data to a first-class citizen in studies on fairness-enhancing interventions. *arXiv preprint arXiv:1911.12587* (2019).

[81] Kumba Sennaar. 2019. Machine Learning for Recruiting and Hiring – 6 Current Applications. https://emerj.com/ai-sector-overviews/machine-learning-for-recruiting-and-hiring/

[82] Ali Seyed Shirkhorshidi, Saeed Aghabozorgi, and Teh Ying Wah. 2015. A comparison study on similarity and dissimilarity measures in clustering continuous data. *PloS one* 10, 12 (2015), e0144059.

[83] E. Strickland. 2016. Doc bot preps for the O.R. *IEEE Spectrum* 53, 6 (June 2016), 32–60. https://doi.org/10.1109/MSPEC.2016.7473150

[84] Rachael Tatman. 2017. Gender and Dialect Bias in YouTube's Automatic Captions. In *Proceedings of the First ACL Workshop on Ethics in Natural Language Processing*. Association for Computational Linguistics, Valencia, Spain, 53–59. https://doi.org/10.18653/v1/W17-1606

[85] Robert L Thorndike. 1953. Who belongs in the family? *Psychometrika* 18, 4 (1953), 267–276.

[86] Sakshi Udeshi, Pryanshu Arora, and Sudipta Chattopadhyay. 2018. Automated directed fairness testing. *Proceedings of the 33rd ACM/IEEE International Conference on Automated Software Engineering - ASE 2018* (2018). https://doi.org/10.1145/3238147.3238165

[87] Inês Valentim, Nuno Lourenço, and Nuno Antunes. 2019. The Impact of Data Preparation on the Fairness of Software Systems. In *2019 IEEE 30th International Symposium on Software Reliability Engineering (ISSRE)*. IEEE, 391–401.

[88] Sriram Vasudevan and Krishnaram Kenthapadi. 2020. LiFT. *Proceedings of the 29th ACM International Conference on Information & Knowledge Management* (Oct 2020). https://doi.org/10.1145/3340531.3412705

[89] Sahil Verma and Julia Rubin. 2018. Fairness Definitions Explained. In *Proceedings of the International Workshop on Software Fairness* (Gothenburg, Sweden) *(FairWare '18)*. Association for Computing Machinery, New York, NY, USA, 1–7. https://doi.org/10.1145/3194770.3194776

[90] Sahil Verma and Julia Rubin. 2018. Fairness definitions explained. In *2018 ieee/acm international workshop on software fairness (fairware)*. IEEE, 1–7.

[91] S. Verma and J. Rubin. 2018. Fairness Definitions Explained. In *2018 IEEE/ACM International Workshop on Software Fairness (FairWare)*. 1–7. https://doi.org/10.23919/FAIRWARE.2018.8452913

[92] Christina Wadsworth, Francesca Vera, and Chris Piech. 2018. Achieving Fairness through Adversarial Learning: an Application to Recidivism Prediction. arXiv:1807.00199 [cs.LG]

[93] Hanchen Wang, Nina Grgic-Hlaca, Preethi Lahoti, Krishna P. Gummadi, and Adrian Weller. 2019. An Empirical Study on Learning Fairness Metrics for COMPAS Data with Human Supervision. arXiv:1910.10255 [cs.CY]

[94] Muhammad Bilal Zafar, Isabel Valera, Manuel Gomez Rogriguez, and Krishna P Gummadi. 2017. Fairness constraints: Mechanisms for fair classification. In *Artificial Intelligence and Statistics*. PMLR, 962–970.

[95] Feng Zhang, Quan Zheng, Ying Zou, and Ahmed E Hassan. 2016. Cross-project defect prediction using a connectivity-based unsupervised classifier. In *2016 IEEE/ACM 38th International Conference on Software Engineering (ICSE)*. IEEE, 309–320.