

CONTENTS

ABSTRACT	1
1. INTRODUCTION	1
1.1. RELATED WORK	1
1.2. CONTRIBUTIONS	2
2. DEFINITIONS	2
3. METHODOLOGICAL APPROACH	2
3.1. PARAMETRISING CORRECTION	2
3.2. SAMPLING STRATEGIES	2
3.3. FINDING THE OPTIMAL AMOUNT OF SAMPLING	3
3.4. MULTIPLE PROTECTED ATTRIBUTES	3
4. EXPERIMENTAL EVALUATION	4
4.1. SEPARABILITY	4
4.2. METHOD VALIDATION	4
4.3. COMPARISON WITH OTHER METHODS	5
5. CONCLUSION	5
REFERENCES	5

Optimising Fairness Through Parametrised Data Sampling

Vladimiro González-Zelaya
Newcastle University, UK
Universidad Panamericana, Mexico
Escuela de Ciencias Económicas y Empresariales
cvgonzalez@up.edu.mx

Dennis Prangle
Newcastle University, UK
School of Mathematics, Statistics and Physics
dennis.prangle@ncl.ac.uk

Julián Salas
Universitat Rovira i Virgili, Spain
Department of Computer Engineering and Mathematics
Center for Cybersecurity Research of Catalonia, Spain
julian.salas@urv.cat

Paolo Missier
Newcastle University, UK
School of Computing
paolo.missier@ncl.ac.uk

ABSTRACT

Improving machine learning models’ fairness is an active research topic, with most approaches focusing on specific definitions of fairness. In contrast, we propose PARDS, a parametrised data sampling method by which we can optimise the *fairness ratios* observed on a test set, in a way that is agnostic to both the specific fairness definitions, and the chosen classification model. Given a training set with one binary protected attribute and a binary label, our approach involves correcting the positive rate for both the *favoured* and *unfavoured* groups through resampling of the training set. We present experimental evidence showing that the amount of resampling can be optimised to achieve target fairness ratios for a specific training set and fairness definition, while preserving most of the model’s accuracy. We discuss conditions for the method to be viable, and then extend the method to include multiple protected attributes. In our experiments we use three different sampling strategies, and we report results for three commonly used definitions of fairness, and three public benchmark datasets: *Adult Income*, *COMPAS* and *German Credit*.

1 INTRODUCTION

The increasing presence of automated decisions in our lives has led to a rising concern about the way in which these decisions are taken, spurring research into the *fairness* of predictive models. These models are often learnt from biased data, reflecting historical disparities and discrimination [27]. We propose PARDS, a fairness-definition and classifier agnostic resampling method, which may be easily implemented on top of existing ML solutions and can satisfy specific classification model requirements. PARDS is modulated through the continuous parameter d , which determines the amount of resampling introduced into the training data, and has two possible use cases: to find the optimal amount of correction for a specific fairness/classifier combination and to control a classifier’s fairness/accuracy trade-off.

Standard data preparation techniques may be used to correct the fairness behaviour of a classification model [29]. PARDS is based on data resampling, which is well understood and part of the typical data management pipeline [23]. Being a preprocessing operator, PARDS may easily be incorporated along data cleaning into existing database solutions. Like other resampling techniques, PARDS can be computationally inexpensive and yield reduced classifier learning times, as shown in Subsection 4.2.

Our method offers the versatility of using both generic (random undersampling, random oversampling and *SMOTE* [6]) and fairness-specific (preferential sampling [20]) methods.

Multiple definitions of fairness have been proposed [25], which are sometimes in contrast with one another. A decision rule that satisfies one of the definitions may well prove to be very unfair for a different one [10]. For example, determining university admissions through gender quotas may achieve *demographic parity*, but it makes the acceptance rates for good students of different genders disparate.

A common resampling problem is the loss of predictive accuracy caused by such interventions [3]. In our setting, such loss can also be controlled through parameter d , allowing for a decision in the amount of accuracy/fairness trade-off the user is willing to accept. Furthermore, our experiments in Section 4 show that even at high correction levels, the accuracy loss for PARDS is relatively low.

1.1 Related Work

A classifier’s fairness may be corrected by *preprocessing* the training data, *in-processing* the learning algorithm [1, 5, 30–32] or *post-processing* a classifier’s predictions [18]. Our method belongs to the group of preprocessing solutions.

Fairness-aware preprocessing is defined [14] as a set of techniques that modify input data so that any classifier trained on such data will be fair. There are four main ways in which to make appropriate adjustments to data in order to enforce fairness [21]: suppressing certain features, also known as *fairness through unawareness (FTU)* [15], *massaging* variable values [4, 9, 13], reweighing features [19, 24], and resampling data instances [8, 20, 28, 29].

Data resampling, the category of PARDS, is less invasive in nature than *FTU* or *massaging*, since the original data is preserved and only the frequency with which the instances are represented is modified. In contrast, *FTU* disposes of large amounts of data without a guarantee on the effect of said intervention and *massaging* effectively creates synthetic data, which does not necessarily reflect the ground truth.

Preferential Sampling (PS) [20] is a similar method to PARDS, in the sense that it resamples the favoured/unfavoured and positive/negative combinations separately in order to equalise the favoured and unfavoured groups’ positive ratios. We empirically show that the optimal fairness correction depends on the selected sampling method, classifier and fairness definition. Equalising the positive ratios across protected attribute groups is not necessarily the best approach, hence we modulate our corrections via

parameter d . When using PS to resample, PARDS generalises it, with the unmodified PS corresponding to the $d = 0$ case.

SMOTEBoost [8] oversamples the minority group through synthetic data based on real data instances, with a focus on improved minority predictions and indirectly improving fairness.

Other related methods include *Capuchin* [29], a causal-fairness centric, non-parametrised resampling method and Feldman et al. [13], a *massaging* method where parameter λ is used to create linear interpolations of the original dataset and a *repaired* copy to find the optimal combination.

1.2 Contributions

We introduce PARDS, a parametrised resampling-based fairness-correcting method. PARDS is fairness-definition and classifier agnostic, and achieves close to optimal fairness correction with a small loss in predictive performance. We present extensive experiments to benchmark the effectiveness of the method using the *Adult Income*, *COMPAS* and *German Credit* datasets, and our implementation is available as a collection of *Jupyter Notebooks* at <https://github.com/vladoXNCL/fairCorrect>. This is a substantial extension of our preliminary workshop paper [17], presented at the *2019 KDD XAI Workshop*. Its additional contributions are four-fold:

- (1) We estimate the optimal fairness correction using Bayesian optimisation.
- (2) We present experimental evidence on synthetic datasets of our method's viability and effectiveness with respect to the linear separability of the training set.
- (3) We make an initial investigation into extending the method to multiple protected attributes.
- (4) We benchmark and compare our work with several existing fairness-correction methods.

2 DEFINITIONS

We will say a binary classifier's label can be *positive* or *negative* referring to the desirable and non-desirable outcome of a prediction, respectively.

A dataset's *protected attribute* (PA) refers to a variable that may be object of discrimination, due to historical bias or otherwise. In our particular case we will be dealing with a single binary PA .

We will call the ratio of the number of positive instances divided by the total number of instances in a specific group the *positive ratio* (PR) of the group.

Among the two PA groups, the one having the highest PR will be referred to as the *favoured* group F , while the other one will be referred to as the *unfavoured* group U . When required, we will refer to the positive and negative instances of F and U as F^+ , F^- , U^+ and U^- , respectively.

We based our analyses on three ratios, *Demographic Parity Ratio* (DPR), *Equality of Opportunity Ratio* (EOR) and *Proxy Fairness Ratio* (PFR), associated to their respective fairness definitions [22, 25]. In these definitions, the positive label is identified with $Y = 1$ and the negative label with $Y = 0$.

Definition 2.1 (Fairness Ratios).

$$\begin{aligned} DPR &:= \frac{\mathbb{P}(\hat{Y} = 1 \mid PA = U)}{\mathbb{P}(\hat{Y} = 1 \mid PA = F)}, \\ EOR &:= \frac{\mathbb{P}(\hat{Y} = 1 \mid PA = U, Y = 1)}{\mathbb{P}(\hat{Y} = 1 \mid PA = F, Y = 1)}, \\ PFR &:= \frac{\mathbb{P}(\hat{Y} = 1 \mid do(PA = U))}{\mathbb{P}(\hat{Y} = 1 \mid do(PA = F))}. \end{aligned}$$

For DPR and EOR , we evaluate the ratio of the positive classification probabilities for U and F . PFR is computed by intervening on the test set T twice, assigning every individual in T the PA -values U and F , resulting in $T_{PA=U}$ and $T_{PA=F}$, respectively. We then evaluate the quotient of the intervened sets' classification PR s; in all cases, the ratios quantify how close the classifier comes to optimal fairness.

3 METHODOLOGICAL APPROACH

We have focused on datasets with both binary protected attributes and labels. The plots in this section result from applying PARDS to the *Adult Income (Income)* dataset [12].

We introduce the *disparity correction* parameter $d \in [-1, 1]$, which may be used for two different objectives:

- To modulate a classifier's fairness/accuracy trade-off.
- To optimise a classifier with respect to a fairness definition.

Our main objective will be the third one, to estimate the d -value optimising a classifier's predictions with respect to a fairness definition. We summarise the method as follows:

- (1) Define PR -correcting functions for F and U .
- (2) Select a *sampling strategy* to correct the training set.
- (3) Estimate the fairness-specific optimal d -value.

Details on each of these steps now follow.

3.1 Parametrising Correction

The first step is to define linear functions that will yield corrected PR s for both PA groups. These functions, which we will call $f^+(d)$ and $u^+(d)$, should satisfy the constraints: $f^+(1) = PR(F)$, $f^+(-1) = PR(U)$ and $u^+(d) = f^+(-d)$.

The equations for these two linear functions are

$$f^+(d) = md + b, \quad u^+(d) = -md + b,$$

with coefficients

$$m = \frac{PR(F) - PR(U)}{2}, \quad b = \frac{PR(F) + PR(U)}{2}.$$

3.2 Sampling Strategies

In the second step, we use the resulting corrected ratios $f^+(d)$ and $u^+(d)$ to produce a resampled training set $\{\hat{U}, \hat{F}\}$ satisfying these ratios. The required amount of resampling for F and U will depend on d and the selected strategy.

PARDS can use one of four different sampling methods, modified to work on specific PA -label subgroups: random undersampling (*Under*), random oversampling (*Over*), *SMOTE* [7] and preferential sampling (*PS*) [20]. Depending on the sampling method, the following subgroups will be modified:

Under: Undersample F^+ and U^- .

Over: Oversample F^- and U^+ .

SMOTE: Oversample F^- and U^+ .

PS: Undersample F^+ and U^- , oversample F^- and U^+ .

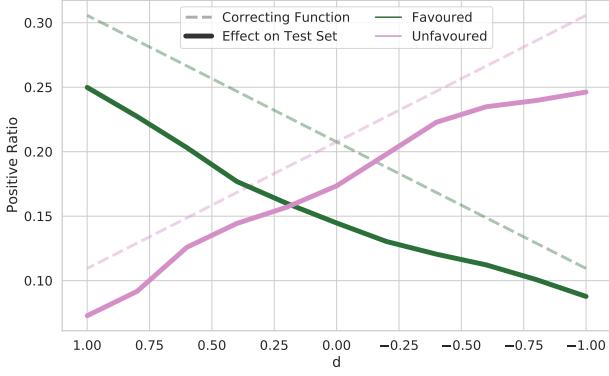


Figure 1: Correcting functions $f^+(d)$ and $u^+(d)$ applied to *Income* and their effect on the test set. The d -axis is reversed, going from 1 (no correction) to -1 (maximum correction). Note that the test-set PRs do not intersect at $d = 0$.

The resampled \hat{F} must satisfy

$$\frac{|\hat{F}^+|}{|\hat{F}^+| + |\hat{F}^-|} = f^+(d),$$

which may be rewritten as

$$\frac{|\hat{F}^+|}{|\hat{F}^-|} = \frac{f^+(d)}{1 - f^+(d)}. \quad (1)$$

The selected strategy will determine whether F^+ or F^- will be resampled to satisfy (1). Using *Under*, for example, \hat{F}^+ results from undersampling F^+ , while $\hat{F}^- = F^-$. In contrast, using *Over* produces \hat{F}^- from oversampling F^- while $\hat{F}^+ = F^+$. An analogous equation to (1) is used to resample U onto \hat{U} .

After the training-set has been resampled, a classifier learnt from the corrected training-set will display an improvement in fairness with respect to a classifier learnt from the original data. An example of the produced PR-correcting functions and their effect over *Income* is shown in Figure 1.

3.3 Finding the Optimal Amount of Sampling

Finally, the third step is to estimate the optimal correction for a specific fairness definition. As classification algorithms usually display non-linear—and sometimes unexpected—behaviours, it is not possible to deduce a closed-form solution to this optimisation problem. Hence, it becomes necessary to numerically approximate a solution.

A naïve approach is to compare the resulting fairness ratios for different values of d , and select the one producing the ratio closest to 1. As we will see on Section 4.2.1, it is easy to find d -values close to the optimal by trial and error, yet this optimal d -value will usually be different for distinct fairness definitions.

A more systematic way to approximate the optimal value of d is to use Bayesian optimisation [16]. This technique estimates the objective function on candidate values obtained from previous function estimations. The main reasons for choosing Bayesian over other optimisation methods are that every fairness ratio evaluation will be different due to the randomness in the sampling process and that Bayesian optimisation is good when estimating the objective function is expensive, e.g. our setting, since we work on large datasets and take the average over many estimations.

We have implemented a simple fairness optimiser using the *GPyOpt* [2] package, with a standard Gaussian process using d

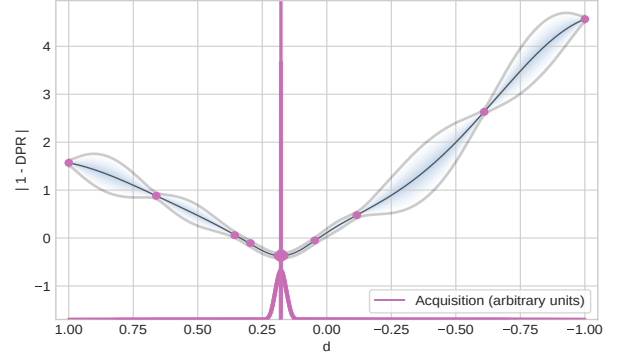


Figure 2: Plot of *GPyOpt*'s approximation of *DPR* as function of d for *Income*. The bottom red curve displays the resulting distribution for the optimal d -value.

as the only parameter and the distances of the different fairness ratios to 1 as objective functions, e.g. estimate the d -value yielding the fairest *DPR* expectation:

$$\arg \min_d |1 - \mathbb{E}[DPR(d)]| \quad \text{subject to} \quad -1 \leq d \leq 1.$$

An example run, used to approximate the optimal correction for Demographic Parity on *Income* may be seen in Figure 2.

3.4 Multiple Protected Attributes

We have generalised PARDS to multi-class PAs, as well as to multiple PA variables. In some cases, this could be addressed by binning several PA labels into just two categories, u and f . However, these arbitrary assignments would imply a loss of granularity in any subsequent fairness analysis. As an alternative, we have chosen to consider a *combined* PA, which may be obtained for every datapoint $p \in \text{train}$ as follows:

- (1) Evaluate $\text{PR}(D)$ for the training set D .
- (2) Define a set of PAs: $\{\text{PA}_1, \text{PA}_2, \dots, \text{PA}_k\}$.
- (3) Evaluate

$$\text{PR}_i(p) = \text{PR}(\text{PA}_i(p)) - \text{PR}(D)$$

for $i \in \{1, 2, \dots, k\}$.

- (4) Aggregate the partial PRs to obtain a combined value

$$\text{PR}^*(p) = \sum_{i=1}^k \text{PR}_i(p).$$

- (5) Define the combined PA of p as

$$\text{PA}^*(p) = \begin{cases} F & \text{if } \text{PR}^*(p) > 0, \\ U & \text{if } \text{PR}^*(p) \leq 0. \end{cases}$$

This solution allows for a much more granular approach on determining a datapoint's relative "prosperity" with respect to every PA, as some PA attributes may prove to be more determining of disparate treatment than others, and the effects of several PAs may cancel each other out.

Our experiments, carried out on *Income*, provide positive results, as described next. Figure 3 compares the effects—for unfavoured groups of different PAs (Gender, Nationality, Race and Age)—of applying disparity correction based on a single PA (Gender) to doing it based on a combined PA aggregating Gender, Age, Race and Country for *Income*. As can be seen, when correcting

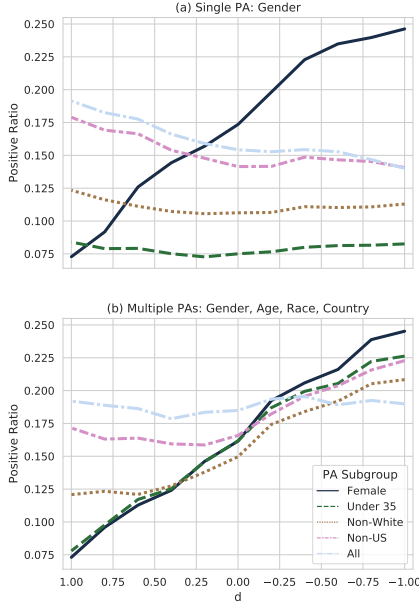


Figure 3: Positive ratios for the different PAs’ unfavoured groups on *Income*, correcting for (a) one PA and (b) multiple PAs.

for Gender alone, the other unfavoured groups’ PR remains relatively constant or gets worse. Likewise, the overall PR shows a drop on its PR as more correction is applied. When correcting for the combined PA, on the other hand, all of the unfavoured PRs improve at a similar rate, whilst the overall PR remains relatively constant across different correction levels. In short, this extension provides PARDS with the capability to correct for multiple biases simultaneously, at the individual level with similar optimal d -values across PAs. This method for combining several PAs into a single combined one, though, is not unique, and could further be improved by adding weights to the different PAs set as hyperparameters by experts.

4 EXPERIMENTAL EVALUATION

This section reports the effectiveness of PARDS regarding separability in Subsection 4.1, comparing sampling strategies and fairness definitions in Subsection 4.2 and benchmarking PARDS with existing fairness-correcting methods, in Subsection 4.3.

4.1 Separability

To verify the effect of separability on PARDS’ effectiveness, we created 11 simple datasets, consisting of one continuous feature f and one binary label l . These datasets were created using the *scikit-learn*’s [11] `make_classification` function, with varying levels of class separability s , ranging from 0 (completely mixed up) to 2 (over 95% probability of complete separation). What this function does is sample feature values from normal distributions centered at s and $-s$ for the two classes, respectively. A PA was then randomly added, ensuring a fixed 50/50 proportion of F vs U datapoints, with PRs of 0.9 and 0.1 for F and U , respectively.

As may be seen in Figure 4a, the greater separability data has, the less effective our correcting method becomes (represented by a near-flat demographic parity ratio curve as a function of d). However, adding random noise to a linearly separable dataset (effectively rendering it inseparable again) restores the effectiveness

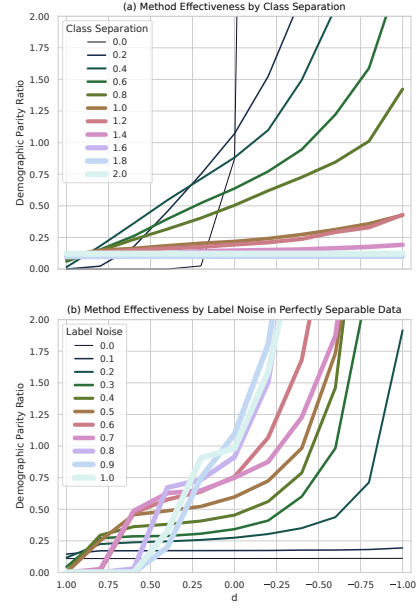


Figure 4: Correction effectiveness by separability. (a): On close-to-linearly separable data, the method becomes highly inefficient, or even stops working at all. (b): Introducing random noise into a separable dataset lets correction become effective again.

of PARDS. To test this, we created a linearly separable dataset with $s = 2$, and gradually introduced noise through parameter n taking values from 0 to 1, the proportion of randomly-assigned labels. As shown in Figure 4b, this intervention can render fairness correction effective again, even with a small amount of added noise.

4.2 Method Validation

We tested PARDS on three datasets commonly used in ML fairness research literature: *Adult Income (Income)* [12], *COMPAS* [26] and *German Credit* [12].

For every dataset, we performed the following experiment 50 times, and then averaged the results for robustness:

- (1) Random train/test split the data with 90/10 proportion.
- (2) For Proxy Fairness checking, make two copies of the *test* set T and intervene PA as either U or F , obtaining $T_{PA=U}$ and $T_{PA=F}$, respectively.
- (3) For each sampling function, obtain 11 training sets, corresponding to $d \in \{1, 0.8, 0.6, \dots, -1\}$.
- (4) For each of these training sets, fit a classifier.
- (5) For every model, get predictions for T , $T_{PA=U}$ and $T_{PA=F}$.
- (6) Compute metrics for accuracy, DPR , EOR and PFR , as well as the model coefficients.

We then proceeded to analyse the resulting fairness metrics, and compared our results with *PS*.

4.2.1 Results. As expected, fairness correction has an impact over a classifier’s predictive performance. Figure 5 shows the fairness-accuracy trade-off for the different sampling strategies for the three fairness ratios over *Income*. As may be seen, the trade-off is similar across the different sampling strategies, and the loss in predictive performance for optimal fairness will be definition dependant.

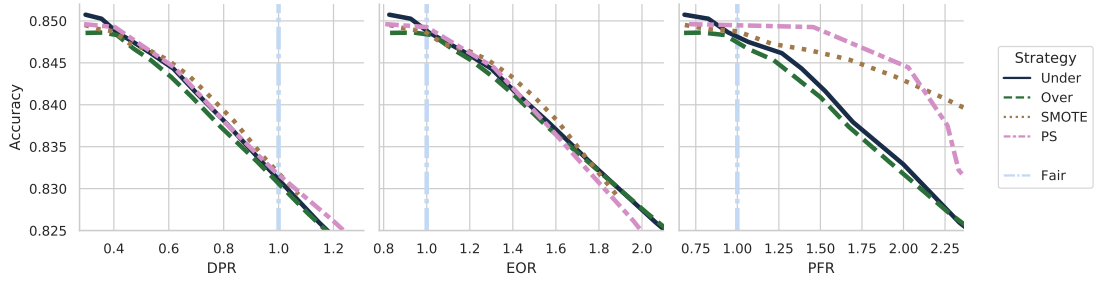


Figure 5: Fairness-accuracy trade-off for DPR, EOR and PFR on *Income*.

Table 1 shows diverse performance metrics for PARDS using the different sampling strategies to correct γ_{sr} on *Income*. The presented means and confidence intervals (CIs) result from 100 independent train/test splits, then using each sampling strategy with the optimal d -value for each estimated through Bayesian optimisation. As may be seen, there is a big difference in computing time across strategies, with *SMOTE* being over 10 times slower than *Under*. On the other hand, *SMOTE* produced the best scores for most performance metrics. Optimal fairness correction was achieved within the CIs for all methods, with roughly the same accuracy loss trade-off. Interestingly, running *Under* before training the classifier was 35% faster than just training the classifier over the full dataset. This would provide an additional advantage for *Under*-corrected training sets when learning models from large-scale datasets.

4.3 Comparison with Other Methods

An intrinsic advantage of PARDS is that it can optimise a classifier with respect to different group fairness definitions. Three definitions: γ_{sr} [5], *discrimination* (*disc*) [21] and *equalised odds* (*eOdds*) [19] were used for our comparisons.

Tables 2 and 3 compare PARDS with a variety of *preprocessing* [4, 8, 19, 20, 24, 29], *in-processing* [1, 5, 30–32] and *post-processing* [18] fairness-correcting methods.

Since four different classification algorithms were used on the papers we compared with—AdaBoost (AB), decision trees (DT), Gaussian naïve Bayes (GNB) and logistic regression (LR)—we present PARDS’ results using all three of them. We optimised our classifiers to compare with the state-of-the-art methods, hence three of our presented methods are optimised for *DPR* and two are optimised for *eOdds*. We evaluated our metrics using the same classification algorithms as the ones used in the papers we compare with. The objective functions to optimise were $|1 - DPR|$ and $|1 - eOdds|$ for *DPR* and *eOdds*, respectively, with 0 being the best value the objective function may take in both cases.

For every tested d -value we averaged the resulting *DPR* of 50 random 90/10 train/test splits, finding optimal d -values of {0.8338, -0.1803, -1.1528, -0.6083} for AB, DT, GNB and LR, respectively. All of the classifiers were trained using the default scikit-learn hyper-parameter values; using these parameter values, we ran PARDS 10 times and averaged the resulting metrics. The fairness and accuracy metrics for the compared methods refer to the *best* reported values in [5, 19, 20, 29, 32]. Likewise, for methods evaluated on more than one classifier, we present the best one.

Our *eOdds*-optimised AdaBoost classifier produced the best overall accuracy (86%), while showing an *eOdds* value within 3% of the best performing method [24]. Regarding our *DPR* optimised classifiers, although LR performed the best overall, both PARDS’

DT and GNB performed better than the other methods’ DTs and GNBs, respectively. Interestingly, PARDS’ LR produced the fairest classifiers with respect to definitions γ_{sr} and *disc*, even though they were actually optimised for *DPR*. While the accuracy of *DPR*-optimised PARDS LR was not the best (83%), it came within 1% of the best performing classifiers (PARDS’ DT, Kamiran and Calders [20] and Zafar et al. [30], with an accuracy of 84%).

5 CONCLUSION

In this paper we define PARDS, a parametrised fairness optimisation method agnostic to both fairness definitions and classification models. Correcting through training set resampling, we have shown that PARDS produces fairness-optimal predictions with a small loss in predictive power. When compared with the existing methods, in most cases PARDS produces the best fairness performance.

In future work we intend to further improve our data resampling methods, in order to optimise for different fairness definitions at once. Although PARDS shows a relatively low impact on prediction performance and its main objective is to estimate the optimal amount of correction with respect to fairness, we would like to find a way to consider predictive performance as well, either in the form of a restriction—e.g. a maximum loss in accuracy or a minimum level of fairness—or by setting an acceptable trade-off rate between both metrics.

ACKNOWLEDGMENTS

This research was partly supported by the Spanish Government under project RTI2018-095094-B-C21 “CONSENT”.

REFERENCES

- [1] Alekh Agarwal, Alina Beygelzimer, Miroslav Dudík, John Langford, and Hanna Wallach. 2018. A reductions approach to fair classification. *arXiv preprint arXiv:1803.02453* (2018).
- [2] The GPyOpt authors. 2016. GPyOpt: A Bayesian Optimization framework in Python. <http://github.com/SheffieldML/GPyOpt>.
- [3] Richard Berk, Hoda Heidari, Shahin Jabbari, Michael Kearns, and Aaron Roth. 2018. Fairness in criminal justice risk assessments: The state of the art. *Sociological Methods & Research* (2018), 0049124118782533.
- [4] Flavio Calmon, Dennis Wei, Bhanukiran Vinzamuri, Karthikeyan Natesan Ramamurthy, and Kush R Varshney. 2017. Optimized pre-processing for discrimination prevention. *Advances in Neural Information Processing Systems* 30 (2017), 3992–4001.
- [5] L Elisa Celis, Lingxiao Huang, Vijay Keswani, and Nisheeth K Vishnoi. 2019. Classification with fairness constraints: A meta-algorithm with provable guarantees. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*. 319–328.
- [6] Nitesh V. Chawla, Kevin W. Bowyer, Lawrence O. Hall, and W. Philip Kegelmeyer. 2002. SMOTE: Synthetic Minority Over-sampling Technique. *Journal of Artificial Intelligence Research* 16 (2002), 321–357.
- [7] Nitesh V Chawla, Kevin W Bowyer, Lawrence O Hall, and W Philip Kegelmeyer. 2002. SMOTE: synthetic minority over-sampling technique. *Journal of artificial intelligence research* 16 (2002), 321–357.

Table 1: Fairness and performance metrics comparison using the four sampling strategies on *Income*, optimising LR models for γ_{sr} with 95% CIs. The best score for each metric is highlighted.

	No Correction	Under	Over	SMOTE	PS
Estimated Optimal d		-0.5770	-0.6083	-0.8246	0.1784
Time (s/iteration)	1.324 \pm 0.018	0.857 \pm 0.014	1.952 \pm 0.030	10.974 \pm 0.085	1.237 \pm 0.016
Discrimination	0.178 \pm 0.003	-0.001 \pm 0.002	0.0 \pm 0.003	0.006 \pm 0.003	-0.003 \pm 0.003
γ_{sr}	0.295 \pm 0.007	1.008 \pm 0.018	1.005 \pm 0.019	0.966 \pm 0.018	1.023 \pm 0.018
Accuracy	0.85 \pm 0.001	0.832 \pm 0.001	0.831 \pm 0.001	0.833 \pm 0.001	0.831 \pm 0.001
Balanced Accuracy	0.761 \pm 0.002	0.695 \pm 0.002	0.692 \pm 0.002	0.717 \pm 0.002	0.712 \pm 0.002
Precision	0.737 \pm 0.003	0.77 \pm 0.004	0.77 \pm 0.004	0.727 \pm 0.004	0.726 \pm 0.004
Recall	0.59 \pm 0.003	0.43 \pm 0.004	0.425 \pm 0.003	0.493 \pm 0.003	0.482 \pm 0.003
F-Score	0.655 \pm 0.003	0.552 \pm 0.003	0.547 \pm 0.003	0.587 \pm 0.003	0.58 \pm 0.003
ROC AUC	0.761 \pm 0.002	0.695 \pm 0.002	0.692 \pm 0.002	0.717 \pm 0.002	0.712 \pm 0.002

Table 2: Fairness metrics and accuracy comparison of our *DPR*-optimised method with related fairness-correcting methods. The best result for each metric is highlighted.

Algorithm	CLF	disc	γ_{sr}	Accuracy
No Correction	LR	0.18	0.295	0.85
PARDS (Over)	LR	0.00	1.00	0.83
PARDS (PS)	GNB	0.00	0.98	0.82
PARDS (PS)	DT	0.01	0.97	0.84
Kamiran and Calders [20]	DT	0.03	—	0.84
Zemel et al. [32]	LR	0.20	—	0.68
Calmon et al. [4]	LR	0.03	—	0.79
Salimi et al. [29]	MLP	0.06	—	0.79
Zafar et al. [31]	GNB	—	0.87	0.77
Hardt et al. [18]	GNB	—	0.85	0.81
Zafar et al. [30]	GNB	—	0.42	0.84
Agarwal et al. [1]	GNB	—	0.72	0.79
Celis et al. [5]	GNB	—	0.95	0.77

Table 3: Equalised odds, accuracy and balanced accuracy comparison of our *eOdds*-optimised method with related fairness-correcting methods. The best result for each metric is highlighted.

Algorithm	CLF	eOdds	Accuracy
No Correction	AB	0.18	0.86
PARDS (PS)	AB	0.08	0.86
PARDS (PS)	LR	0.09	0.83
Krasanakis et al. [24]	LR	0.05	0.82
Iosifidis and Ntoutsi [19]	AB	0.08	0.83
Chawla et al. [8]	AB	0.47	0.81

- [8] Nitesh V Chawla, Aleksandar Lazarevic, Lawrence O Hall, and Kevin W Bowyer. 2003. SMOTEBoost: Improving prediction of the minority class in boosting. In *European conference on principles of data mining and knowledge discovery*. Springer, 107–119.
- [9] Silvia Chiappa and Thomas PS Gillam. 2018. Path-specific counterfactual fairness. *arXiv preprint arXiv:1802.08139* (2018).
- [10] Alexandra Chouldechova. 2017. Fair prediction with disparate impact: A study of bias in recidivism prediction instruments. *Big data* 5, 2 (2017), 153–163.
- [11] Sci-kit Learn Developers. 2019. scikit-learn: machine learning in Python.
- [12] Dheeru Dua and Casey Graff. 2017. UCI Machine Learning Repository. <http://archive.ics.uci.edu/ml>
- [13] Michael Feldman, Sorelle A Friedler, John Moeller, Carlos Scheidegger, and Suresh Venkatasubramanian. 2015. Certifying and removing disparate impact. In *proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*. 259–268.
- [14] 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*. ACM, 329–338.
- [15] Pratik Gajane and Mykola Pechenizkiy. 2017. On formalizing fairness in prediction with machine learning. *arXiv preprint arXiv:1710.03184* (2017).
- [16] Javier González, Michael Osborne, and Neil D Lawrence. 2016. GLASSES: Relieving the myopia of Bayesian optimisation. (2016).
- [17] Vladimiro González-Zelaya, Paolo Missier, and Dennis Prangle. 2019. Parametrised Data Sampling for Fairness Optimisation. (2019). Presented on the 2019 XAI Workshop at SIGKDD, Anchorage, AK, USA. Available at <http://homepages.cs.ncl.ac.uk/paolo.missier/doc/kddSubmission.pdf>.
- [18] Moritz Hardt, Eric Price, Nati Srebro, et al. 2016. Equality of opportunity in supervised learning. In *Advances in neural information processing systems*. 3315–3323.
- [19] Vasileios Iosifidis and Eirini Ntoutsi. 2019. AdaFair: Cumulative fairness adaptive boosting. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*. 781–790.
- [20] Faisal Kamiran and Toon Calders. 2010. Classification with no discrimination by preferential sampling. In *Proc. 19th Machine Learning Conf. Belgium and The Netherlands*. Citeseer, 1–6.
- [21] Faisal Kamiran and Toon Calders. 2012. Data preprocessing techniques for classification without discrimination. *Knowledge and Information Systems* 33, 1 (2012), 1–33.
- [22] Niki Kilbertus, Mateo Rojas Carulla, Giambattista Parascandolo, Moritz Hardt, Dominik Janzing, and Bernhard Schölkopf. 2017. Avoiding discrimination through causal reasoning. In *Advances in Neural Information Processing Systems*. 656–666.
- [23] SB Kotsiantis, Dimitris Kanellopoulos, and PE Pintelas. 2006. Data preprocessing for supervised learning. *International Journal of Computer Science* 1, 2 (2006), 111–117.
- [24] Emmanouil Krasanakis, Eleftherios Spyromitros-Xioufis, Symeon Papadopoulos, and Yiannis Kompatsiaris. 2018. Adaptive sensitive reweighting to mitigate bias in fairness-aware classification. In *Proceedings of the 2018 World Wide Web Conference on World Wide Web*. International World Wide Web Conferences Steering Committee, 853–862.
- [25] Matt J Kusner, Joshua Loftus, Chris Russell, and Ricardo Silva. 2017. Counterfactual fairness. In *Advances in Neural Information Processing Systems*. 4066–4076.
- [26] Jeff Larson, Surya Mattu, Lauren Kirchner, and Julia Angwin. 2016. How we analyzed the COMPAS recidivism algorithm. *ProPublica* (5 2016) 9 (2016).
- [27] David Madras, Elliot Creager, Toniann Pitassi, and Richard Zemel. 2019. Fairness through causal awareness: Learning causal latent-variable models for biased data. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*. 349–358.
- [28] Donald B Rubin. 1973. The use of matched sampling and regression adjustment to remove bias in observational studies. *Biometrics* (1973), 185–203.
- [29] Babak Salimi, Luke Rodriguez, Bill Howe, and Dan Suciu. 2019. Interventional fairness: Causal database repair for algorithmic fairness. In *Proceedings of the 2019 International Conference on Management of Data*. 793–810.
- [30] Muhammad Bilal Zafar, Isabel Valera, Manuel Gomez Rodriguez, and Krishna P Gummadi. 2017. Fairness beyond disparate treatment & disparate impact: Learning classification without disparate mistreatment. In *Proceedings of the 26th international conference on world wide web*. 1171–1180.
- [31] Muhammad Bilal Zafar, Isabel Valera, Manuel Gomez Rodriguez, and Krishna P Gummadi. 2015. Fairness constraints: Mechanisms for fair classification. *arXiv preprint arXiv:1507.05259* (2015).
- [32] Rich Zemel, Yu Wu, Kevin Swersky, Toni Pitassi, and Cynthia Dwork. 2013. Learning fair representations. In *International Conference on Machine Learning*. 325–333.

CONTENTS

ABSTRACT	1
1. INTRODUCTION	1
1.1. RELATED WORK	1
1.2. CONTRIBUTIONS	2
2. DEFINITIONS	2
3. METHODOLOGICAL APPROACH	2
3.1. PARAMETRISING CORRECTION	2
3.2. SAMPLING STRATEGIES	2
3.3. FINDING THE OPTIMAL AMOUNT OF SAMPLING	3
3.4. MULTIPLE PROTECTED ATTRIBUTES	3
4. EXPERIMENTAL EVALUATION	4
4.1. SEPARABILITY	4
4.2. METHOD VALIDATION	4
4.3. COMPARISON WITH OTHER METHODS	5
5. CONCLUSION	5
REFERENCES	5

Optimising Fairness Through Parametrised Data Sampling

Vladimiro González-Zelaya
Newcastle University, UK
Universidad Panamericana, Mexico
Escuela de Ciencias Económicas y Empresariales
cvgonzalez@up.edu.mx

Dennis Prangle
Newcastle University, UK
School of Mathematics, Statistics and Physics
dennis.prangle@ncl.ac.uk

Julián Salas
Universitat Rovira i Virgili, Spain
Department of Computer Engineering and Mathematics
Center for Cybersecurity Research of Catalonia, Spain
julian.salas@urv.cat

Paolo Missier
Newcastle University, UK
School of Computing
paolo.missier@ncl.ac.uk

ABSTRACT

Improving machine learning models’ fairness is an active research topic, with most approaches focusing on specific definitions of fairness. In contrast, we propose PARDS, a parametrised data sampling method by which we can optimise the *fairness ratios* observed on a test set, in a way that is agnostic to both the specific fairness definitions, and the chosen classification model. Given a training set with one binary protected attribute and a binary label, our approach involves correcting the positive rate for both the *favoured* and *unfavoured* groups through resampling of the training set. We present experimental evidence showing that the amount of resampling can be optimised to achieve target fairness ratios for a specific training set and fairness definition, while preserving most of the model’s accuracy. We discuss conditions for the method to be viable, and then extend the method to include multiple protected attributes. In our experiments we use three different sampling strategies, and we report results for three commonly used definitions of fairness, and three public benchmark datasets: *Adult Income*, *COMPAS* and *German Credit*.

1 INTRODUCTION

The increasing presence of automated decisions in our lives has led to a rising concern about the way in which these decisions are taken, spurring research into the *fairness* of predictive models. These models are often learnt from biased data, reflecting historical disparities and discrimination [27]. We propose PARDS, a fairness-definition and classifier agnostic resampling method, which may be easily implemented on top of existing ML solutions and can satisfy specific classification model requirements. PARDS is modulated through the continuous parameter d , which determines the amount of resampling introduced into the training data, and has two possible use cases: to find the optimal amount of correction for a specific fairness/classifier combination and to control a classifier’s fairness/accuracy trade-off.

Standard data preparation techniques may be used to correct the fairness behaviour of a classification model [29]. PARDS is based on data resampling, which is well understood and part of the typical data management pipeline [23]. Being a preprocessing operator, PARDS may easily be incorporated along data cleaning into existing database solutions. Like other resampling techniques, PARDS can be computationally inexpensive and yield reduced classifier learning times, as shown in Subsection 4.2.

Our method offers the versatility of using both generic (random undersampling, random oversampling and *SMOTE* [6]) and fairness-specific (preferential sampling [20]) methods.

Multiple definitions of fairness have been proposed [25], which are sometimes in contrast with one another. A decision rule that satisfies one of the definitions may well prove to be very unfair for a different one [10]. For example, determining university admissions through gender quotas may achieve *demographic parity*, but it makes the acceptance rates for good students of different genders disparate.

A common resampling problem is the loss of predictive accuracy caused by such interventions [3]. In our setting, such loss can also be controlled through parameter d , allowing for a decision in the amount of accuracy/fairness trade-off the user is willing to accept. Furthermore, our experiments in Section 4 show that even at high correction levels, the accuracy loss for PARDS is relatively low.

1.1 Related Work

A classifier’s fairness may be corrected by *preprocessing* the training data, *in-processing* the learning algorithm [1, 5, 30–32] or *post-processing* a classifier’s predictions [18]. Our method belongs to the group of preprocessing solutions.

Fairness-aware preprocessing is defined [14] as a set of techniques that modify input data so that any classifier trained on such data will be fair. There are four main ways in which to make appropriate adjustments to data in order to enforce fairness [21]: suppressing certain features, also known as *fairness through unawareness (FTU)* [15], *massaging* variable values [4, 9, 13], reweighing features [19, 24], and resampling data instances [8, 20, 28, 29].

Data resampling, the category of PARDS, is less invasive in nature than *FTU* or *massaging*, since the original data is preserved and only the frequency with which the instances are represented is modified. In contrast, *FTU* disposes of large amounts of data without a guarantee on the effect of said intervention and *massaging* effectively creates synthetic data, which does not necessarily reflect the ground truth.

Preferential Sampling (PS) [20] is a similar method to PARDS, in the sense that it resamples the favoured/unfavoured and positive/negative combinations separately in order to equalise the favoured and unfavoured groups’ positive ratios. We empirically show that the optimal fairness correction depends on the selected sampling method, classifier and fairness definition. Equalising the positive ratios across protected attribute groups is not necessarily the best approach, hence we modulate our corrections via

parameter d . When using PS to resample, PARDS generalises it, with the unmodified PS corresponding to the $d = 0$ case.

SMOTEBoost [8] oversamples the minority group through synthetic data based on real data instances, with a focus on improved minority predictions and indirectly improving fairness.

Other related methods include *Capuchin* [29], a causal-fairness centric, non-parametrised resampling method and Feldman et al. [13], a *massaging* method where parameter λ is used to create linear interpolations of the original dataset and a *repaired* copy to find the optimal combination.

1.2 Contributions

We introduce PARDS, a parametrised resampling-based fairness-correcting method. PARDS is fairness-definition and classifier agnostic, and achieves close to optimal fairness correction with a small loss in predictive performance. We present extensive experiments to benchmark the effectiveness of the method using the *Adult Income*, *COMPAS* and *German Credit* datasets, and our implementation is available as a collection of *Jupyter Notebooks* at <https://github.com/vladoXNCL/fairCorrect>. This is a substantial extension of our preliminary workshop paper [17], presented at the *2019 KDD XAI Workshop*. Its additional contributions are four-fold:

- (1) We estimate the optimal fairness correction using Bayesian optimisation.
- (2) We present experimental evidence on synthetic datasets of our method's viability and effectiveness with respect to the linear separability of the training set.
- (3) We make an initial investigation into extending the method to multiple protected attributes.
- (4) We benchmark and compare our work with several existing fairness-correction methods.

2 DEFINITIONS

We will say a binary classifier's label can be *positive* or *negative* referring to the desirable and non-desirable outcome of a prediction, respectively.

A dataset's *protected attribute* (PA) refers to a variable that may be object of discrimination, due to historical bias or otherwise. In our particular case we will be dealing with a single binary PA .

We will call the ratio of the number of positive instances divided by the total number of instances in a specific group the *positive ratio* (PR) of the group.

Among the two PA groups, the one having the highest PR will be referred to as the *favoured* group F , while the other one will be referred to as the *unfavoured* group U . When required, we will refer to the positive and negative instances of F and U as F^+ , F^- , U^+ and U^- , respectively.

We based our analyses on three ratios, *Demographic Parity Ratio* (DPR), *Equality of Opportunity Ratio* (EOR) and *Proxy Fairness Ratio* (PFR), associated to their respective fairness definitions [22, 25]. In these definitions, the positive label is identified with $Y = 1$ and the negative label with $Y = 0$.

Definition 2.1 (Fairness Ratios).

$$\begin{aligned} DPR &:= \frac{\mathbb{P}(\hat{Y} = 1 \mid PA = U)}{\mathbb{P}(\hat{Y} = 1 \mid PA = F)}, \\ EOR &:= \frac{\mathbb{P}(\hat{Y} = 1 \mid PA = U, Y = 1)}{\mathbb{P}(\hat{Y} = 1 \mid PA = F, Y = 1)}, \\ PFR &:= \frac{\mathbb{P}(\hat{Y} = 1 \mid do(PA = U))}{\mathbb{P}(\hat{Y} = 1 \mid do(PA = F))}. \end{aligned}$$

For DPR and EOR , we evaluate the ratio of the positive classification probabilities for U and F . PFR is computed by intervening on the test set T twice, assigning every individual in T the PA -values U and F , resulting in $T_{PA=U}$ and $T_{PA=F}$, respectively. We then evaluate the quotient of the intervened sets' classification PR s; in all cases, the ratios quantify how close the classifier comes to optimal fairness.

3 METHODOLOGICAL APPROACH

We have focused on datasets with both binary protected attributes and labels. The plots in this section result from applying PARDS to the *Adult Income (Income)* dataset [12].

We introduce the *disparity correction* parameter $d \in [-1, 1]$, which may be used for two different objectives:

- To modulate a classifier's fairness/accuracy trade-off.
- To optimise a classifier with respect to a fairness definition.

Our main objective will be the third one, to estimate the d -value optimising a classifier's predictions with respect to a fairness definition. We summarise the method as follows:

- (1) Define PR -correcting functions for F and U .
- (2) Select a *sampling strategy* to correct the training set.
- (3) Estimate the fairness-specific optimal d -value.

Details on each of these steps now follow.

3.1 Parametrising Correction

The first step is to define linear functions that will yield corrected PR s for both PA groups. These functions, which we will call $f^+(d)$ and $u^+(d)$, should satisfy the constraints: $f^+(1) = PR(F)$, $f^+(-1) = PR(U)$ and $u^+(d) = f^+(-d)$.

The equations for these two linear functions are

$$f^+(d) = md + b, \quad u^+(d) = -md + b,$$

with coefficients

$$m = \frac{PR(F) - PR(U)}{2}, \quad b = \frac{PR(F) + PR(U)}{2}.$$

3.2 Sampling Strategies

In the second step, we use the resulting corrected ratios $f^+(d)$ and $u^+(d)$ to produce a resampled training set $\{\hat{U}, \hat{F}\}$ satisfying these ratios. The required amount of resampling for F and U will depend on d and the selected strategy.

PARDS can use one of four different sampling methods, modified to work on specific PA -label subgroups: random undersampling (*Under*), random oversampling (*Over*), *SMOTE* [7] and preferential sampling (*PS*) [20]. Depending on the sampling method, the following subgroups will be modified:

Under: Undersample F^+ and U^- .

Over: Oversample F^- and U^+ .

SMOTE: Oversample F^- and U^+ .

PS: Undersample F^+ and U^- , oversample F^- and U^+ .

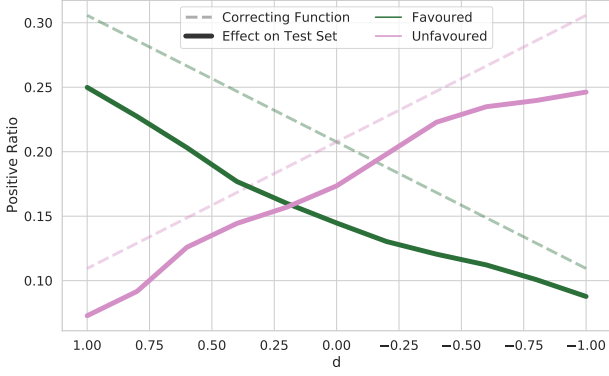


Figure 1: Correcting functions $f^+(d)$ and $u^+(d)$ applied to *Income* and their effect on the test set. The d -axis is reversed, going from 1 (no correction) to -1 (maximum correction). Note that the test-set PRs do not intersect at $d = 0$.

The resampled \hat{F} must satisfy

$$\frac{|\hat{F}^+|}{|\hat{F}^+| + |\hat{F}^-|} = f^+(d),$$

which may be rewritten as

$$\frac{|\hat{F}^+|}{|\hat{F}^-|} = \frac{f^+(d)}{1 - f^+(d)}. \quad (1)$$

The selected strategy will determine whether F^+ or F^- will be resampled to satisfy (1). Using *Under*, for example, \hat{F}^+ results from undersampling F^+ , while $\hat{F}^- = F^-$. In contrast, using *Over* produces \hat{F}^- from oversampling F^- while $\hat{F}^+ = F^+$. An analogous equation to (1) is used to resample U onto \hat{U} .

After the training-set has been resampled, a classifier learnt from the corrected training-set will display an improvement in fairness with respect to a classifier learnt from the original data. An example of the produced PR-correcting functions and their effect over *Income* is shown in Figure 1.

3.3 Finding the Optimal Amount of Sampling

Finally, the third step is to estimate the optimal correction for a specific fairness definition. As classification algorithms usually display non-linear—and sometimes unexpected—behaviours, it is not possible to deduce a closed-form solution to this optimisation problem. Hence, it becomes necessary to numerically approximate a solution.

A naïve approach is to compare the resulting fairness ratios for different values of d , and select the one producing the ratio closest to 1. As we will see on Section 4.2.1, it is easy to find d -values close to the optimal by trial and error, yet this optimal d -value will usually be different for distinct fairness definitions.

A more systematic way to approximate the optimal value of d is to use Bayesian optimisation [16]. This technique estimates the objective function on candidate values obtained from previous function estimations. The main reasons for choosing Bayesian over other optimisation methods are that every fairness ratio evaluation will be different due to the randomness in the sampling process and that Bayesian optimisation is good when estimating the objective function is expensive, e.g. our setting, since we work on large datasets and take the average over many estimations.

We have implemented a simple fairness optimiser using the *GPyOpt* [2] package, with a standard Gaussian process using d

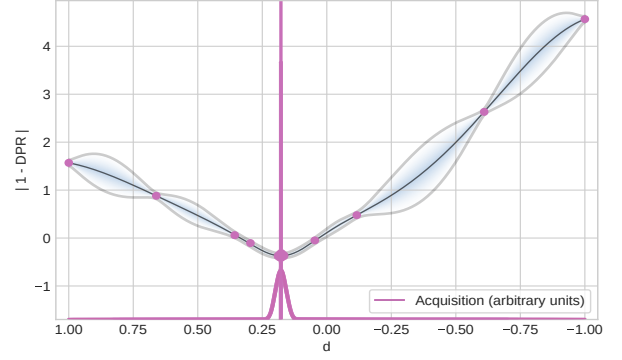


Figure 2: Plot of *GPyOpt*'s approximation of *DPR* as function of d for *Income*. The bottom red curve displays the resulting distribution for the optimal d -value.

as the only parameter and the distances of the different fairness ratios to 1 as objective functions, e.g. estimate the d -value yielding the fairest *DPR* expectation:

$$\arg \min_d |1 - \mathbb{E}[DPR(d)]| \quad \text{subject to} \quad -1 \leq d \leq 1.$$

An example run, used to approximate the optimal correction for Demographic Parity on *Income* may be seen in Figure 2.

3.4 Multiple Protected Attributes

We have generalised PARDS to multi-class PAs, as well as to multiple PA variables. In some cases, this could be addressed by binning several PA labels into just two categories, u and f . However, these arbitrary assignments would imply a loss of granularity in any subsequent fairness analysis. As an alternative, we have chosen to consider a *combined* PA, which may be obtained for every datapoint $p \in \text{train}$ as follows:

- (1) Evaluate $\text{PR}(D)$ for the training set D .
- (2) Define a set of PAs: $\{\text{PA}_1, \text{PA}_2, \dots, \text{PA}_k\}$.
- (3) Evaluate

$$\text{PR}_i(p) = \text{PR}(\text{PA}_i(p)) - \text{PR}(D)$$

for $i \in \{1, 2, \dots, k\}$.

- (4) Aggregate the partial PRs to obtain a combined value

$$\text{PR}^*(p) = \sum_{i=1}^k \text{PR}_i(p).$$

- (5) Define the combined PA of p as

$$\text{PA}^*(p) = \begin{cases} F & \text{if } \text{PR}^*(p) > 0, \\ U & \text{if } \text{PR}^*(p) \leq 0. \end{cases}$$

This solution allows for a much more granular approach on determining a datapoint's relative "prosperity" with respect to every PA, as some PA attributes may prove to be more determining of disparate treatment than others, and the effects of several PAs may cancel each other out.

Our experiments, carried out on *Income*, provide positive results, as described next. Figure 3 compares the effects—for unfavoured groups of different PAs (Gender, Nationality, Race and Age)—of applying disparity correction based on a single PA (Gender) to doing it based on a combined PA aggregating Gender, Age, Race and Country for *Income*. As can be seen, when correcting

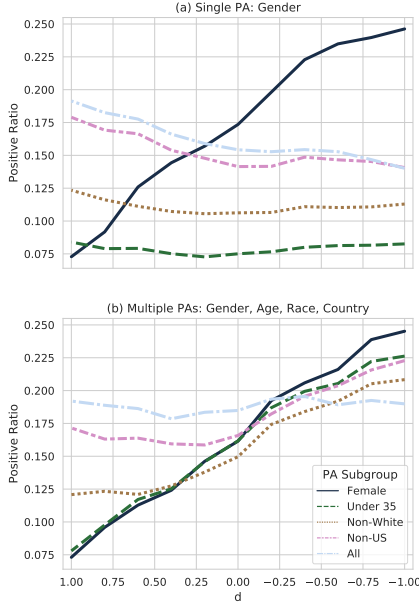


Figure 3: Positive ratios for the different PAs’ unfavoured groups on *Income*, correcting for (a) one PA and (b) multiple PAs.

for Gender alone, the other unfavoured groups’ PR remains relatively constant or gets worse. Likewise, the overall PR shows a drop on its PR as more correction is applied. When correcting for the combined PA, on the other hand, all of the unfavoured PRs improve at a similar rate, whilst the overall PR remains relatively constant across different correction levels. In short, this extension provides PARDS with the capability to correct for multiple biases simultaneously, at the individual level with similar optimal d -values across PAs. This method for combining several PAs into a single combined one, though, is not unique, and could further be improved by adding weights to the different PAs set as hyperparameters by experts.

4 EXPERIMENTAL EVALUATION

This section reports the effectiveness of PARDS regarding separability in Subsection 4.1, comparing sampling strategies and fairness definitions in Subsection 4.2 and benchmarking PARDS with existing fairness-correcting methods, in Subsection 4.3.

4.1 Separability

To verify the effect of separability on PARDS’ effectiveness, we created 11 simple datasets, consisting of one continuous feature f and one binary label l . These datasets were created using the *scikit-learn*’s [11] `make_classification` function, with varying levels of class separability s , ranging from 0 (completely mixed up) to 2 (over 95% probability of complete separation). What this function does is sample feature values from normal distributions centered at s and $-s$ for the two classes, respectively. A PA was then randomly added, ensuring a fixed 50/50 proportion of F vs U datapoints, with PRs of 0.9 and 0.1 for F and U , respectively.

As may be seen in Figure 4a, the greater separability data has, the less effective our correcting method becomes (represented by a near-flat demographic parity ratio curve as a function of d). However, adding random noise to a linearly separable dataset (effectively rendering it inseparable again) restores the effectiveness

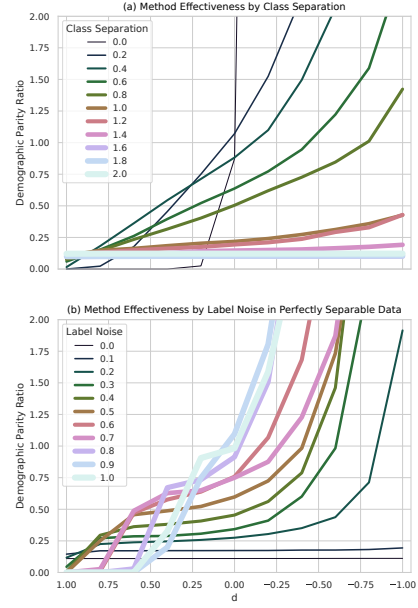


Figure 4: Correction effectiveness by separability. (a): On close-to-linearly separable data, the method becomes highly inefficient, or even stops working at all. (b): Introducing random noise into a separable dataset lets correction become effective again.

of PARDS. To test this, we created a linearly separable dataset with $s = 2$, and gradually introduced noise through parameter n taking values from 0 to 1, the proportion of randomly-assigned labels. As shown in Figure 4b, this intervention can render fairness correction effective again, even with a small amount of added noise.

4.2 Method Validation

We tested PARDS on three datasets commonly used in ML fairness research literature: *Adult Income (Income)* [12], *COMPAS* [26] and *German Credit* [12].

For every dataset, we performed the following experiment 50 times, and then averaged the results for robustness:

- (1) Random train/test split the data with 90/10 proportion.
- (2) For Proxy Fairness checking, make two copies of the *test* set T and intervene PA as either U or F , obtaining $T_{PA=U}$ and $T_{PA=F}$, respectively.
- (3) For each sampling function, obtain 11 training sets, corresponding to $d \in \{1, 0.8, 0.6, \dots, -1\}$.
- (4) For each of these training sets, fit a classifier.
- (5) For every model, get predictions for T , $T_{PA=U}$ and $T_{PA=F}$.
- (6) Compute metrics for accuracy, DPR , EOR and PFR , as well as the model coefficients.

We then proceeded to analyse the resulting fairness metrics, and compared our results with *PS*.

4.2.1 Results. As expected, fairness correction has an impact over a classifier’s predictive performance. Figure 5 shows the fairness-accuracy trade-off for the different sampling strategies for the three fairness ratios over *Income*. As may be seen, the trade-off is similar across the different sampling strategies, and the loss in predictive performance for optimal fairness will be definition dependant.

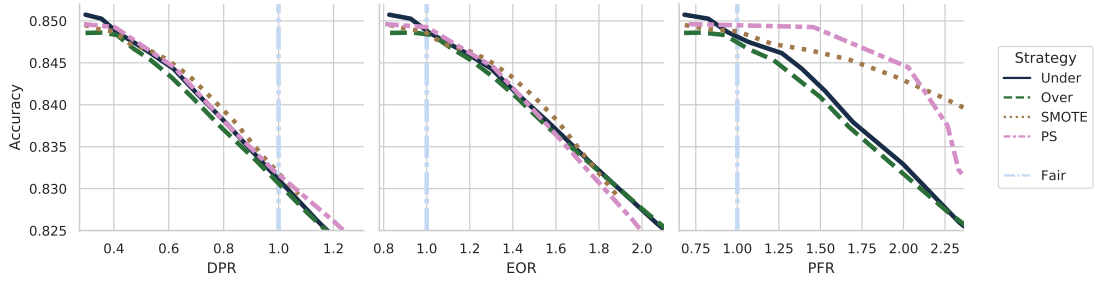


Figure 5: Fairness-accuracy trade-off for DPR, EOR and PFR on *Income*.

Table 1 shows diverse performance metrics for PARDS using the different sampling strategies to correct γ_{sr} on *Income*. The presented means and confidence intervals (CIs) result from 100 independent train/test splits, then using each sampling strategy with the optimal d -value for each estimated through Bayesian optimisation. As may be seen, there is a big difference in computing time across strategies, with *SMOTE* being over 10 times slower than *Under*. On the other hand, *SMOTE* produced the best scores for most performance metrics. Optimal fairness correction was achieved within the CIs for all methods, with roughly the same accuracy loss trade-off. Interestingly, running *Under* before training the classifier was 35% faster than just training the classifier over the full dataset. This would provide an additional advantage for *Under*-corrected training sets when learning models from large-scale datasets.

4.3 Comparison with Other Methods

An intrinsic advantage of PARDS is that it can optimise a classifier with respect to different group fairness definitions. Three definitions: γ_{sr} [5], *discrimination* (*disc*) [21] and *equalised odds* (*eOdds*) [19] were used for our comparisons.

Tables 2 and 3 compare PARDS with a variety of *preprocessing* [4, 8, 19, 20, 24, 29], *in-processing* [1, 5, 30–32] and *post-processing* [18] fairness-correcting methods.

Since four different classification algorithms were used on the papers we compared with—AdaBoost (AB), decision trees (DT), Gaussian naïve Bayes (GNB) and logistic regression (LR)—we present PARDS’ results using all three of them. We optimised our classifiers to compare with the state-of-the-art methods, hence three of our presented methods are optimised for *DPR* and two are optimised for *eOdds*. We evaluated our metrics using the same classification algorithms as the ones used in the papers we compare with. The objective functions to optimise were $|1 - DPR|$ and $|1 - eOdds|$ for *DPR* and *eOdds*, respectively, with 0 being the best value the objective function may take in both cases.

For every tested d -value we averaged the resulting *DPR* of 50 random 90/10 train/test splits, finding optimal d -values of {0.8338, -0.1803, -1.1528, -0.6083} for AB, DT, GNB and LR, respectively. All of the classifiers were trained using the default scikit-learn hyper-parameter values; using these parameter values, we ran PARDS 10 times and averaged the resulting metrics. The fairness and accuracy metrics for the compared methods refer to the *best* reported values in [5, 19, 20, 29, 32]. Likewise, for methods evaluated on more than one classifier, we present the best one.

Our *eOdds*-optimised AdaBoost classifier produced the best overall accuracy (86%), while showing an *eOdds* value within 3% of the best performing method [24]. Regarding our *DPR* optimised classifiers, although LR performed the best overall, both PARDS’

DT and GNB performed better than the other methods’ DTs and GNBs, respectively. Interestingly, PARDS’ LR produced the fairest classifiers with respect to definitions γ_{sr} and *disc*, even though they were actually optimised for *DPR*. While the accuracy of *DPR*-optimised PARDS LR was not the best (83%), it came within 1% of the best performing classifiers (PARDS’ DT, Kamiran and Calders [20] and Zafar et al. [30], with an accuracy of 84%).

5 CONCLUSION

In this paper we define PARDS, a parametrised fairness optimisation method agnostic to both fairness definitions and classification models. Correcting through training set resampling, we have shown that PARDS produces fairness-optimal predictions with a small loss in predictive power. When compared with the existing methods, in most cases PARDS produces the best fairness performance.

In future work we intend to further improve our data resampling methods, in order to optimise for different fairness definitions at once. Although PARDS shows a relatively low impact on prediction performance and its main objective is to estimate the optimal amount of correction with respect to fairness, we would like to find a way to consider predictive performance as well, either in the form of a restriction—e.g. a maximum loss in accuracy or a minimum level of fairness—or by setting an acceptable trade-off rate between both metrics.

ACKNOWLEDGMENTS

This research was partly supported by the Spanish Government under project RTI2018-095094-B-C21 “CONSENT”.

REFERENCES

- [1] Alekh Agarwal, Alina Beygelzimer, Miroslav Dudík, John Langford, and Hanna Wallach. 2018. A reductions approach to fair classification. *arXiv preprint arXiv:1803.02453* (2018).
- [2] The GPyOpt authors. 2016. GPyOpt: A Bayesian Optimization framework in Python. <http://github.com/SheffieldML/GPyOpt>.
- [3] Richard Berk, Hoda Heidari, Shahin Jabbari, Michael Kearns, and Aaron Roth. 2018. Fairness in criminal justice risk assessments: The state of the art. *Sociological Methods & Research* (2018), 0049124118782533.
- [4] Flavio Calmon, Dennis Wei, Bhanukiran Vinzamuri, Karthikeyan Natesan Ramamurthy, and Kush R Varshney. 2017. Optimized pre-processing for discrimination prevention. *Advances in Neural Information Processing Systems* 30 (2017), 3992–4001.
- [5] L Elisa Celis, Lingxiao Huang, Vijay Keswani, and Nisheeth K Vishnoi. 2019. Classification with fairness constraints: A meta-algorithm with provable guarantees. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*. 319–328.
- [6] Nitesh V. Chawla, Kevin W. Bowyer, Lawrence O. Hall, and W. Philip Kegelmeyer. 2002. SMOTE: Synthetic Minority Over-sampling Technique. *Journal of Artificial Intelligence Research* 16 (2002), 321–357.
- [7] Nitesh V Chawla, Kevin W Bowyer, Lawrence O Hall, and W Philip Kegelmeyer. 2002. SMOTE: synthetic minority over-sampling technique. *Journal of artificial intelligence research* 16 (2002), 321–357.

Table 1: Fairness and performance metrics comparison using the four sampling strategies on *Income*, optimising LR models for γ_{sr} with 95% CIs. The best score for each metric is highlighted.

	No Correction	Under	Over	SMOTE	PS
Estimated Optimal d		-0.5770	-0.6083	-0.8246	0.1784
Time (s/iteration)	1.324 \pm 0.018	0.857 \pm 0.014	1.952 \pm 0.030	10.974 \pm 0.085	1.237 \pm 0.016
Discrimination	0.178 \pm 0.003	-0.001 \pm 0.002	0.0 \pm 0.003	0.006 \pm 0.003	-0.003 \pm 0.003
γ_{sr}	0.295 \pm 0.007	1.008 \pm 0.018	1.005 \pm 0.019	0.966 \pm 0.018	1.023 \pm 0.018
Accuracy	0.85 \pm 0.001	0.832 \pm 0.001	0.831 \pm 0.001	0.833 \pm 0.001	0.831 \pm 0.001
Balanced Accuracy	0.761 \pm 0.002	0.695 \pm 0.002	0.692 \pm 0.002	0.717 \pm 0.002	0.712 \pm 0.002
Precision	0.737 \pm 0.003	0.77 \pm 0.004	0.77 \pm 0.004	0.727 \pm 0.004	0.726 \pm 0.004
Recall	0.59 \pm 0.003	0.43 \pm 0.004	0.425 \pm 0.003	0.493 \pm 0.003	0.482 \pm 0.003
F-Score	0.655 \pm 0.003	0.552 \pm 0.003	0.547 \pm 0.003	0.587 \pm 0.003	0.58 \pm 0.003
ROC AUC	0.761 \pm 0.002	0.695 \pm 0.002	0.692 \pm 0.002	0.717 \pm 0.002	0.712 \pm 0.002

Table 2: Fairness metrics and accuracy comparison of our *DPR*-optimised method with related fairness-correcting methods. The best result for each metric is highlighted.

Algorithm	CLF	disc	γ_{sr}	Accuracy
No Correction	LR	0.18	0.295	0.85
PARDS (Over)	LR	0.00	1.00	0.83
PARDS (PS)	GNB	0.00	0.98	0.82
PARDS (PS)	DT	0.01	0.97	0.84
Kamiran and Calders [20]	DT	0.03	—	0.84
Zemel et al. [32]	LR	0.20	—	0.68
Calmon et al. [4]	LR	0.03	—	0.79
Salimi et al. [29]	MLP	0.06	—	0.79
Zafar et al. [31]	GNB	—	0.87	0.77
Hardt et al. [18]	GNB	—	0.85	0.81
Zafar et al. [30]	GNB	—	0.42	0.84
Agarwal et al. [1]	GNB	—	0.72	0.79
Celis et al. [5]	GNB	—	0.95	0.77

Table 3: Equalised odds, accuracy and balanced accuracy comparison of our *eOdds*-optimised method with related fairness-correcting methods. The best result for each metric is highlighted.

Algorithm	CLF	eOdds	Accuracy
No Correction	AB	0.18	0.86
PARDS (PS)	AB	0.08	0.86
PARDS (PS)	LR	0.09	0.83
Krasanakis et al. [24]	LR	0.05	0.82
Iosifidis and Ntoutsi [19]	AB	0.08	0.83
Chawla et al. [8]	AB	0.47	0.81

- [8] Nitesh V Chawla, Aleksandar Lazarevic, Lawrence O Hall, and Kevin W Bowyer. 2003. SMOTEBoost: Improving prediction of the minority class in boosting. In *European conference on principles of data mining and knowledge discovery*. Springer, 107–119.
- [9] Silvia Chiappa and Thomas PS Gillam. 2018. Path-specific counterfactual fairness. *arXiv preprint arXiv:1802.08139* (2018).
- [10] Alexandra Chouldechova. 2017. Fair prediction with disparate impact: A study of bias in recidivism prediction instruments. *Big data* 5, 2 (2017), 153–163.
- [11] Sci-kit Learn Developers. 2019. scikit-learn: machine learning in Python.
- [12] Dheeru Dua and Casey Graff. 2017. UCI Machine Learning Repository. <http://archive.ics.uci.edu/ml>
- [13] Michael Feldman, Sorelle A Friedler, John Moeller, Carlos Scheidegger, and Suresh Venkatasubramanian. 2015. Certifying and removing disparate impact. In *proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*. 259–268.
- [14] 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*. ACM, 329–338.
- [15] Pratik Gajane and Mykola Pechenizkiy. 2017. On formalizing fairness in prediction with machine learning. *arXiv preprint arXiv:1710.03184* (2017).
- [16] Javier González, Michael Osborne, and Neil D Lawrence. 2016. GLASSES: Relieving the myopia of Bayesian optimisation. (2016).
- [17] Vladimiro González-Zelaya, Paolo Missier, and Dennis Prangle. 2019. Parametrised Data Sampling for Fairness Optimisation. (2019). Presented on the *2019 XAI Workshop at SIGKDD, Anchorage, AK, USA*. Available at <http://homepages.cs.ncl.ac.uk/paolo.missier/doc/kddSubmission.pdf>.
- [18] Moritz Hardt, Eric Price, Nati Srebro, et al. 2016. Equality of opportunity in supervised learning. In *Advances in neural information processing systems*. 3315–3323.
- [19] Vasileios Iosifidis and Eirini Ntoutsi. 2019. AdaFair: Cumulative fairness adaptive boosting. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*. 781–790.
- [20] Faisal Kamiran and Toon Calders. 2010. Classification with no discrimination by preferential sampling. In *Proc. 19th Machine Learning Conf. Belgium and The Netherlands*. Citeseer, 1–6.
- [21] Faisal Kamiran and Toon Calders. 2012. Data preprocessing techniques for classification without discrimination. *Knowledge and Information Systems* 33, 1 (2012), 1–33.
- [22] Niki Kilbertus, Mateo Rojas Carulla, Giambattista Parascandolo, Moritz Hardt, Dominik Janzing, and Bernhard Schölkopf. 2017. Avoiding discrimination through causal reasoning. In *Advances in Neural Information Processing Systems*. 656–666.
- [23] SB Kotsiantis, Dimitris Kanellopoulos, and PE Pintelas. 2006. Data preprocessing for supervised learning. *International Journal of Computer Science* 1, 2 (2006), 111–117.
- [24] Emmanouil Krasanakis, Eleftherios Spyromitros-Xioufis, Symeon Papadopoulos, and Yiannis Kompatsiaris. 2018. Adaptive sensitive reweighting to mitigate bias in fairness-aware classification. In *Proceedings of the 2018 World Wide Web Conference on World Wide Web*. International World Wide Web Conferences Steering Committee, 853–862.
- [25] Matt J Kusner, Joshua Loftus, Chris Russell, and Ricardo Silva. 2017. Counterfactual fairness. In *Advances in Neural Information Processing Systems*. 4066–4076.
- [26] Jeff Larson, Surya Mattu, Lauren Kirchner, and Julia Angwin. 2016. How we analyzed the COMPAS recidivism algorithm. *ProPublica* (5 2016) 9 (2016).
- [27] David Madras, Elliot Creager, Toniann Pitassi, and Richard Zemel. 2019. Fairness through causal awareness: Learning causal latent-variable models for biased data. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*. 349–358.
- [28] Donald B Rubin. 1973. The use of matched sampling and regression adjustment to remove bias in observational studies. *Biometrics* (1973), 185–203.
- [29] Babak Salimi, Luke Rodriguez, Bill Howe, and Dan Suciu. 2019. Interventional fairness: Causal database repair for algorithmic fairness. In *Proceedings of the 2019 International Conference on Management of Data*. 793–810.
- [30] Muhammad Bilal Zafar, Isabel Valera, Manuel Gomez Rodriguez, and Krishna P Gummadi. 2017. Fairness beyond disparate treatment & disparate impact: Learning classification without disparate mistreatment. In *Proceedings of the 26th international conference on world wide web*. 1171–1180.
- [31] Muhammad Bilal Zafar, Isabel Valera, Manuel Gomez Rodriguez, and Krishna P Gummadi. 2015. Fairness constraints: Mechanisms for fair classification. *arXiv preprint arXiv:1507.05259* (2015).
- [32] Rich Zemel, Yu Wu, Kevin Swersky, Toni Pitassi, and Cynthia Dwork. 2013. Learning fair representations. In *International Conference on Machine Learning*. 325–333.