Ding Ke (kd120)

1. Introduction

The report begins with a simple breakdown of the complex implementation details of model selection, as well as statements about methods adopted in empirical evaluations.

1.1. Implementation overview

The core of the model selection pipeline evolves around a custom-built gridsearchcv function named `cross_validation_search`, which is reminiscent of `sklearn`'s well-known `GridSearchCV`. It is designed specifically to process the `aif360` dataset object, enable model evaluation with `aif360` fairness metrics, and perform multi-model cross-validation and hyperparameter-tuning with fold number, different estimator types and parameter grid of user's choosing.

What makes this `cross_validation_search` unique from standard `GridSearchCV` is first a series of helper functions (chief among which `get_fold_idx_iterables`) that produces a list of index iterables for a given required number of folds such that `aif360` datasets' `subset` [1] built-in method can be exploited to greatly facilitate fold splitting process. However, note here that the aforementioned fold slicing mechanism (`get_fold_idx_iterables` followed by `subset`) supports only total fold number $num\_fold > 2$, for number of folds equal to 2 use the other `aif360` built method `split` and pass in a single split ratio.

Once the training and testing datasets are correctly sliced using the `subset` method at each fold, the rest of cross-validation operation follows standard cross-validation strategy, except for the hyperparameter search at each fold is conducted for each estimator with each hyperparameter value in the search range. Therefore it is worth noting that the custom-built `cross_validation_search` supports gridsearch on different types of estimators, unlike the standard `sklearn`'s `GridSearchCV`.

The performance of each model at each fold is stored in a an array. After the end of operation for all folds, this result array is averaged across all folds for each model, computing the average model performance on validation data across all folds. Finally each model with its corresponding performance is zipped into a pandas dataframe, which is then sorted from best to worst by each performance metric. Additionally the `visualisation` flag in `cross_validation_search` can be set to true to plot the model selection process.

In general, the gridsearchcv and model selection pipeline is implemented with maximised efficiency in mind, exploiting `aif360`'s built-in method to the fullest; helper functions are devised where possible, in-line with the principle of abstraction.

1.2. Setup of experimentations & methodologies

1.2.1 Dataset overview

In this work there are two `aif360` datasets chosen for empirical evaluation: `AdultIncome` dataset and `German` dataset. For the `AdultIncome` dataset, the protected/sensitive attribute used is "sex" while for the `German` dataset the protected attribute chosen is "age". Note these attributes are all binary. Authors of `aif360` [2] wrote: a protected attribute is one that should partition population into groups which would enjoy equal benefit; and that it contains privileged value that points to group with historical advantage over others (for Adult dataset under protected attribute "sex" the privileged value is "Male"; for German under protected attribute "age" the privileged value is $age \geq 25$). Thus in this work, the protected attribute is central to the group fairness study which is essentially detecting whether there is any bias w.r.t. to the chosen protected attribute.

1.2.2 Machine learning models

In total there are two types of classification estimator investigated in this work: `LogisticRegression` and `svm.LinearSVC` from `sklearn`. There is only one hyperparameter type that is being searched: the regularisation term C. Note each estimator type is simply set to share the same hyperparameter C search range, but should the reader wish otherwise each estimator can also be set to have different hyperparameter search ranges.

For all of the empirical evaluations in this work, the C search range in particular, is defined to be an array of 6 elements, spanning from 1×10^{-4} to 1×10^{1}, evenly incremented on a logscale with logbase of 10. Such logspace-spanned C search range is chosen considering the fact that C is inversely proportional to the regularisation strength in `sklearn`'s implementations, and the logspace C search is also usually adopted in other works involving classifier selections [5].

1.2.3 Evaluation metrics

For model evaluations, metrics from two aspects need to be considered: accuracy and fairness based. For accuracy evaluations the standard `sklearn` `accuracy_score` is used; for fairness evaluations the `aif360`'s `equal_opportunity_difference` is chosen, although reader can also check the similar `average_odds_difference` that is also stored in the evaluation dictionary. Equality of opportunity and equalised odds are chosen as the fairness metric in this work because as precisely discussed in M.Hardt et al.'s work [3], equalised odds enables the predictor C to be dependent on protected attribute A only through outcome Y allowing perfect classifier with highest accuracy, and superior to demographic parity which may lead to abusing the protected attribute as proxy for outcome i.e. predicts true for more individuals that are not even actually able(have false labels) from privileged group and decreases the accuracy. Equality of opportunity is a relaxation of equalised odds as it only requires no bias conditioned on labels being the positive class [3].

1.2.4 Fairness methodologies

For the fairness method in task 2 under main tasks, the default/baseline is chosen to be the `aif360 Reweighing`. However, there are different fairness methods and circumstances investigated in this work, and those will be discussed in another standalone section in section 3.

For task 3 suggesting a new criterion that combines both accuracy and fairness metrics (equal opportunity difference), based on knowledge of the fairness metric's behaviour, the combined metric is formulated as a weighted sum of accuracy and fairness metric:

$$combined\_metric = a \times accuracy + b \times \mid fairness \mid$$

where the a and b are respectively the weights and $\begin{cases} a > 0 \\ b < 0 \end{cases}$, $\forall a, b$. The negative sign requirement for b is mandated because of the observation that the absolute value of equal opportunity difference should be minimised (as close to 0 as possible) to achieve the maximisation of fairness. The value assignment to a and b essentially dictates the degree of accuracy-fairness trade-off desired, i.e. a very large a means accuracy is prioritised while fairness is sacrificed. The determination of exact values to be used is based on observation of task 1 ,2 and final results. To find the optimal weights, a is kept at $a = 1.0$ and $b = [-0.1, -0.5, -1, -1.5, -3]$ is searched. It occurs that when $\begin{cases} a = 1 \\ b = -1.5 \end{cases}$, the new criterion can reliably pick out model 5 and 6 with the performance on test data no worse than (in most cases) the best models in task 2 and task 1 respectively. This result is obtained by testing on both `Adult` and `German` datasets.

2. Main results for task 1,2 & 3

The main tasks are carried out on two datasets. Therefore there are two sets of results listed and discussed in two separate parts, first for `Adult` then for `German`.

	model_1_accuracy	model_1_eq_opp_diff
mean	0.801160172	-0.4495958
std	0.003579015	0.003813031
	model_2_accuracy	model_2_eq_opp_diff
mean	0.795359312	-0.236617798
std	0.004269458	0.021620834
	model_3_accuracy	model_3_eq_opp_diff
mean	0.787347301	0.011887436
std	0.004164368	0.025066703
	model_4_accuracy	model_4_eq_opp_diff
mean	0.787251757	0.00217556
std	0.004068938	0.012839719
	model_5_accuracy	model_5_eq_opp_diff
mean	0.787251757	0.00217556
std	0.004068938	0.012839719
	model_6_accuracy	model_6_eq_opp_diff
mean	0.795359312	-0.236617798
std	0.004269458	0.021620834

Table 1. Final results across 5 repeats for Adult dataset.

2.1. Results on Adult dataset

As the effect of initial random train/test split of the dataset is non-negligible (as can be seen by the considerable standard deviations especially for fairness metric recorded in Table 1), the entire pipeline is repeated five times, each time with a different seed for initial random split. For reproducibility, the random seed list containing five random seeds corresponding to the five repeats responsible for results shown here are included:$[1, 222, 444, 888, 248]$. The final results are averages of accuracy & fairness results for each model from each task across five repeats, shown in Table 1.

It is discovered that at repeat 5 the evaluation results for all models are the closest to the mean values from final results (as can be verified by comparing Table 1 with Table 2, 3 & 4). Therefore repeat 5 is deemed to be the representative split, and its specific Task 1, 2 & 3 results on test data can be seen in more detail below.

model	accuracy	eq_opp_diff
model_1: lr_C_0.001	0.80550	-0.45071
model_2: lr_C_0.0001	0.79936	-0.23880

Table 2. Task 1 results on held-out test data for Adult dataset, at repeat 5.

model	accuracy	eq_opp_diff
model_3: lr_C_0.0001	0.79219	-0.00048
model_4: svm_C_0.0001	0.79219	-0.00048

Table 3. Task 2 results on held-out test data for Adult dataset, at repeat 5.

model	accuracy	eq_opp_diff
model_5: lr_C_0.01	0.79219	-0.00048
model_6: lr_C_0.0001	0.79936	-0.23880

Table 4. Task 3 results on held-out test data for Adult dataset, at repeat 5.

2.1.1 Model selection and test results in task 1,2 & 3 at the representative split

From Table 2, it reads that model 1(best accuracy) is selected to be a Logistic Regression(lr) model with a C of 0.001; model 2(best fairness) is a also a lr model with a C of 0.0001. Such model selection comes from cross validation performed for all candidate models which is visualised in Figure 1 in Appendix A, where the yellow line represents lr models; red line as svm models; each scatter dot is a model with C value specified by its x-coordinate. For the accuracy subplot the dot with the highest y-value(accuracy) is chosen, that is the yellow dot at $C = 1 \times 10^{-3}$, indeed a lr model with C of 0.001. Similarly for the fairness subplot the best model is one with its y-value closest to 0, indeed it is a lr model with $C = 1 \times 10^{-3}$. These graphical interpretations correspond to the models shown selected in Table 2.

Similarly, models selected in Table 3 can be verified by checking the model selection plots in Figure 2. The model selection process in task 3 is a bit different. In Figure 4, model 5 is selected by ranking the highest scoring model by the combined metric, from task 2 models(fairness-based). In Figure 3, model 6 is the highest combined metric model from task 1 models(standard). And Table 4 confirms the models tested are indeed the models selected from cross-validation.

Looking closely at the models selected in task 1 and 2 (Table 2 and 3), it can be seen that model 2 with worse accuracy but significantly better fairness (almost doubles the performance of model 1 even with no fairness method applied) has a C value of 0.0001 whereas model 1 has a C of 0.001, meaning model 2 has a greater regularisation strength. For task 2, model 3 and 4 have the same regularisation strength, but their actual fairness scores on test data after `Reweighing` are identical and very close to 0, still suggesting a strong regularisation is a recipe for good fairness. These findings do support the hypothesis "Regularisation could help preventing the model of doing too well on the majority group".

2.1.2 Analysis of Adult dataset final results

Observing final results Table 1, one can spot some general trends: model 1 has better accuracy but worse fairness performance compared with model 2; model 3 has slightly better accuracy but again worse fairness performance than model 4. This confirms that there exists an accuracy-fairness trade-off and when optimising for one aspect you would lose performance on the other. And observing the fact that both model 3 and 4 have significantly better fairness performance than model 1 & 2 demonstrates the effectiveness of fairness method applied(in this case `Reweighing`).

What is interesting is that model 5 has the identical average performances as model 4, both in terms accuracy and fairness, and searching across all 6 models the performances of model 4/5 are actually the overall best: accuracy is only 0.0081 less than the highest one(model 2) but fairness is far better than anyone else. This a testament not only to the effectiveness of combine metric as it finds the overall best(both accuracy and fairness), but also a testament to the good accuracy-fairness trade-off of the `Reweighing` method as model 4 has already the best trade-off among all models.

2.2. Results on German dataset

	model_1_accuracy	model_1_eq_opp_diff
mean	0.690666667	-0.351493204
std	0.012995726	0.137928895
	model_2_accuracy	model_2_eq_opp_diff
mean	0.683333333	-0.328501421
std	0.028674418	0.103920727
	model_3_accuracy	model_3_eq_opp_diff
mean	0.693333333	-0.007457606
std	0.021473498	0.015932099
	model_4_accuracy	model_4_eq_opp_diff
mean	0.688666667	0.000838558
std	0.017094509	0.035686955
	model_5_accuracy	model_5_eq_opp_diff
mean	0.693333333	-0.007592559
std	0.022236107	0.020021364
	model_6_accuracy	model_6_eq_opp_diff
mean	0.683333333	-0.328501421
std	0.028674418	0.103920727

Table 5. Final results across 5 repeats for German dataset.

Random splits cause even greater effect than that in Adult dataset, as can be seen in Table 5 the standard deviations of most accuracy and fairness results are more than 10 times higher than those in Adult's final results. This phenomenon can be explained by the shear size difference of Adult (total 48842 entries) and German (total 1000 entries) datasets. The much smaller amount of training examples for model selection on German dataset means individual data points matter more i.e. model selection would be more likely to be affected by outliers.

For reproducibility, the seeds responsible for the results shown here are made available: [4, 5, 6, 7, 8]. And the representative split is found to with a seed of 5, that is repeat

2 has the closest results to the averages.

model	accuracy	eq_opp_diff
model_1: lr_C_0.1	0.68667	-0.23077
model_2: svm_C_0.01	0.68667	-0.23077

Table 6. Task 1 results on held-out test data for German dataset, at repeat 2.

model	accuracy	eq_opp_diff
model_3: svm_C_0.01	0.66333	0.0
model_4: svm_C_0.01	0.66333	0.0

Table 7. Task 2 results on held-out test data for German dataset, at repeat 2.

model	accuracy	eq_opp_diff
model_5: svm_C_0.01	0.66333	0.0
model_6: svm_C_0.01	0.68667	-0.23077

Table 8. Task 3 results on held-out test data for German dataset, at repeat 2.

2.2.1 Model selection and test results in task 1,2 & 3 at the representative split

The cross validation results and model selection process are visualised in Figure 5(for task 1), 6(for task 2), 7 & 8(for task 3). And the selected models are registered and tested on held-out test set, results are recorded in Table 6, 7, 8 for task 1, 2 & 3 respectively.

Looking closely at Table 6 and 7, this time model 2 which is selected according to highest fairness from cross-validation has a C of 0.01, while model 1 has a C of 0.1. This indicates model 2 has stronger regularisation than model 1, albeit estimator type being different(svm & lr). Also considering the actual test performances on fairness for both model 1 and 2 are identical, it is not guaranteed to support the hypothesis. Looking at the broader picture, on German dataset models selected for highest fairness seem to have bigger C values than those selected on Adult dataset (0.01 for German models compared with 0.0001 for Adult models). This suggests models selected on German dataset would prefer weaker regularisation strength than what is typically optimal for Adult models. It seems not to be the case that strongest regularisation would always lead the best fairness performance, at least not for every dataset.

2.2.2 Analysis of German dataset final results

From Table 5, overall trends for final results on German dataset are similar to those on Adult: (1) accuracy-fairness trade-off exists and task model pairs (model 1 & 2; model 3 & 4) in task 1 and 2 have either higher accuracy or fairness but not both; (2) `Reweighing` is again very effective in enhancing fairness performance on German dataset, and this time it actually increases the accuracy score on test data. This means task 2 with fairness method applied will guarantee better models than task 1 approach.

Focusing on task 3, first compare model 5 with model 3 & 4 which all have fairness method applied, model 5 has similar performances with model 3. And model 3 has accuracy score 0.00466 higher but fairness score 0.00661 lower than those of model 4. This implies that the combined metric still inclines slightly towards prioritising accuracy in the accuracy-fairness trade-off. Such behaviour is reasonable as in this case as one could argue a 0.00466 improvement in accuracy is worth sacrificing 0.00661 equal opportunity difference which is already quite close to 0 anyway.

3. Some Extrapolations

3.1. Other Fairness methods

Fairness methods/bias mitigating algorithms are ways for reducing unwanted bias in training/model [2] and there are three categories of fairness methods that can be applied to task 2: pre-processing, in-processing and post-processing.

In this work, Equalised Odds Post-processing(EOP) is also analysed and implemented: `cross_validation_search_w_eop` is a substitute for original reweighing-based task 2 method `cross_validation_search_w_reweigh`. Specifically, EOP is deployed inside the `evaluate_w_eop` where an EOP object is initialised with protected attribute information, and then used the `fit` method on the ground-truth and predicted labels to compute parameters for equalising the odds, then used the `predict` method to obtain new labels that satisfy the learned constraints. Note for the full running of task 1, 2 & 3 with task 2 fairness method being EOP, user can simply run the `full_run_for_ds_w_eop`.

Compared with pre-processing methods which take place before training, such as reweighing or representation learning, equalised odds post-processing(EOP) is trying to fix the model predictions after the training. As the equalised odds views fairness through the lens of probabilistic approach, EOP essentially constructs and solves a linear program whose solution is an optimal predictor that produces new prediction under the equalised odds constraints [3]. Whereas reweighing simply breaks the correlation between the final outcome Y and the protected attribute A by re-calculating and re-assigning weights, as discussed in Calders et al.'s work [4]. Representation learning, first investigated in Zemel et al.'s work [6], and also mentioned in Zhang, Y. and Zhou, L.s' work [7], is about achieving both group and individual fairness by training data clustering and representing/mapping onto a fairer space while preserving feature (X) information by minimising the reconstruction loss.

3.1.1 Final results with EOP as the fairness method

When switching the task 2 fairness method to EOP, the whole pipeline is re-run on the Adult dataset(as it is bigger and less prone to randomness). Final results averaged across 5 repeats are included in Table 9, with the same random seeds as the previous experiment on Adult dataset, enabling direct comparison between the fairness methods.

	model_1_accuracy	model_1_eq_opp_diff
mean	0.801160172	-0.4495958
std	0.003579015	0.003813031
	model_2_accuracy	model_2_eq_opp_diff
mean	0.795359312	-0.236617798
std	0.004269458	0.021620834
	model_3_accuracy	model_3_eq_opp_diff
mean	0.771650856	0.002385766
std	0.004542038	0.00215398
	model_4_accuracy	model_4_eq_opp_diff
mean	0.608380536	-0.0034486
std	0.146477416	0.020336129
	model_5_accuracy	model_5_eq_opp_diff
mean	0.771650856	0.002385766
std	0.004542038	0.00215398
	model_6_accuracy	model_6_eq_opp_diff
mean	0.795359312	-0.236617798
std	0.004269458	0.021620834

Table 9. Final results across 5 repeats for Adult dataset with EOP as the fairness method, for reproducibility random seeds for the five repeats are: [1, 222, 444, 888, 248].

repeats	model_4_accuracy	model_4_eq_opp_diff
Repeat_1	0.769603494	-0.000303109
Repeat_2	0.502286221	0.02496888
Repeat_3	0.504060602	-0.013474437
Repeat_4	0.76803385	0.001662754
Repeat_5	0.497918515	-0.030097087

Table 10. Model 4 performances on all of the 5 repeats, on Adult dataset with EOP applied as the fairness method.

Comparing Table 9 with Table 1, focusing on model 3 and 4's performances, it can be seen that model 3 from Table 1 (reweighing method), compared with model 3 from Table 9 (EOP) has much worse fairness score (0.0118 compared with 0.00238) but slightly better accuracy score (roughly 0.01 higher). This shows that EOP method can better enhance fairness performance but may come with minor costs on accuracy.

Examining model 4 performances from Table 9 and Table 1, one can spot a surprising drop in model 4 accuracy score with EOP method applied compared with that with reweighing applied, an almost 19 percent drop in accuracy! To further investigate this abnormality, the model 4's performances at all 5 splits/repeats are printed out, as seen in Table 10. It can be noticed that repeat number 2, 3 & 5 have incredibly low accuracies while at repeat 1 & 4 the accuracy scores are comparable to those under model 3, and the fairness results all look fine. And the log messages during repeat 2, 3 & 4 reveal that the EOP solver warned about ill-posed problem and suggesting a relaxation of constraints to find better solutions. This leads to a conclusion that repeat 2, 3 & 4 are "unlucky" data splits such that EOP find the constraints under those conditions ill-posed. This also reflects that EOP is more sensitive to random splits than reweighing.

3.2. Excluding the sensitive attribute from input feature

The other scenario investigated is when the input feature matrix X does not include the sensitive attribute A. This is implemented first via conversion of `aif360` datasets to pandas dataframes, then drop the sensitive/protected attribute column from the X, as there seems to be no easy way of dropping a specific column directly with `aif360`'s dataset object. To run the entire 3 tasks pipeline, simply run the `full_run_for_ds_nsf`.

3.2.1 Final results from excluding the sensitive attribute from input feature

Experiments are conducted on both Adult and German datasets, keeping all other variables including random split seeds constant(same from the original standard experiment from section 2), and only exclude the sensitive attribute from X. For Adult dataset, final results across 5 repeats are recorded in Table 11. For German dataset, final results across 5 repeats are recorded in Table 12.

Comparing how this exclusion of sensitive attribute would change results in Adult dataset (compare Table 1 and 11), one can observe that except for model 3 which has its accuracies and fairness unchanged before and after the exclusion, the rest of models all have their accuracies decreased by various degrees after the exclusion of sensitive feature. However, the fairness scores of models after this exclusion are all dramatically improved, except for model 3's remains unchanged. In particular, for model 1 & 2, their fairness scores after exclusion are enhanced as much as 65 times!(-0.449; -0.237 before versus 0.00699; 0.0245 after). This strongly suggests that excluding sensitive attribute from X can significantly increases fairness performance while only trading-off a minor loss in accuracy. The loss in accuracy is indeed sensible as one can argue that the input features to the model now contain less information and more training data is always useful.

	model_1_accuracy	model_1_eq_opp_diff
mean	0.787019723	0.006987805
std	0.004434744	0.024500849
	model_2_accuracy	model_2_eq_opp_diff
mean	0.786869583	0.004004723
std	0.004984018	0.036305065
	model_3_accuracy	model_3_eq_opp_diff
mean	0.787347301	0.011887436
std	0.004164368	0.025066703
	model_4_accuracy	model_4_eq_opp_diff
mean	0.786869583	0.004004723
std	0.004984018	0.036305065
	model_5_accuracy	model_5_eq_opp_diff
mean	0.786869583	0.004004723
std	0.004984018	0.036305065
	model_6_accuracy	model_6_eq_opp_diff
mean	0.786869583	0.004004723
std	0.004984018	0.036305065

Table 11. Final results across 5 repeats for Adult dataset with sensitive feature excluded from input feature X, for reproducibility random seeds for the five repeats are: [1, 222, 444, 888, 248].

	model_1_accuracy	model_1_eq_opp_diff
mean	0.692	-0.019674263
std	0.020357909	0.019733946
	model_2_accuracy	model_2_eq_opp_diff
mean	0.692666667	-0.012720764
std	0.021265517	0.014755004
	model_3_accuracy	model_3_eq_opp_diff
mean	0.692	-0.011551174
std	0.020763215	0.016178537
	model_4_accuracy	model_4_eq_opp_diff
mean	0.691333333	-0.011551174
std	0.019804601	0.016178537
	model_5_accuracy	model_5_eq_opp_diff
mean	0.691333333	-0.011551174
std	0.019804601	0.016178537
	model_6_accuracy	model_6_eq_opp_diff
mean	0.692666667	-0.012720764
std	0.021265517	0.014755004

Table 12. Final results across 5 repeats for German dataset with sensitive feature excluded from input feature X, for reproducibility random seeds for the five repeats are: [4, 5, 6, 7, 8].

Now for the exclusion effect on German datasets, compare Table 12 and 5. It is still the case that fairness scores of model 1 and 2 are drastically improved (-0.3515; -0.3285 before versus -0.0197; -0.0127 after), albeit fairness scores of model 3 and 4 actually worsen by a small negligible amount after exclusion. However, surprisingly again the accuracy scores for all models after the exclusion all increase a bit, as opposed to the believed trade-off. This suggests that it is not always the case that more information/training data is helpful for the model performance, at least not for every dataset and every sensitive feature removed. Some sensitive features may be redundant or even counter-productive in terms of achieving high accuracy for the model.

4. Conclusion

As discovered in section 2, final results of both Adult and German datasets show that accuracy and fairness are in most cases at odds with each other, although applying fairness method such as `Reweighing` or removing sensitive attribute from input features can sometimes enhance performances on both, as seen in German dataset. Empirical evidence from EOP experiments shows that EOP as a fairness method can achieve better fairness enhancement than reweighing method, but sacrifices a bit of accuracy. Excluding the sensitive attribute from X can significantly improves fairness scores, at least for the two datasets tested, but this may also come with small costs on accuracy depending on the sensitive feature dropped and dataset.

For future effort, there are still many fairness method not yet empirically and methodologically tested like another post-processing method Reject Option Classification (ROC). In particular, this work has not investigated in-processing methods like Adversarial Debiasing, for incorporating in-processing methods into current pipeline may require considerable functional changes to the train-evaluation scheme as in-processing method introduces changes to the model.

References

[1] AIF360Authors. Aif360 read the docs. https://aif360.readthedocs.io/en/latest/modules/datasets.html. 1

[2] R. K. Bellamy, K. Dey, M. Hind, S. C. Hoffman, S. Houde, K. Kannan, P. Lohia, J. Martino, S. Mehta, A. Mojsilović, et al. Ai fairness 360: An extensible toolkit for detecting and mitigating algorithmic bias. *IBM Journal of Research and Development*, 63(4/5):4–1, 2019. 1, 4

[3] M. Hardt, E. Price, and N. Srebro. Equality of opportunity in supervised learning. *arXiv preprint arXiv:1610.02413*, 2016. 2, 4

[4] F. Kamiran and T. Calders. Data preprocessing techniques for classification without discrimination. *Knowledge and Information Systems*, 33(1):1–33, 2012. 4

[5] C. Neale. Hyper-parameter tuning and model selection, like a movie star, 2019. https://towardsdatascience.com/hyper-parameter-tuning-and-model-selection-like-a-movie-star-a884b8ee8d68. 1

[6] R. Zemel, Y. Wu, K. Swersky, T. Pitassi, and C. Dwork. Learning fair representations. In *International conference on machine learning*, pages 325–333. PMLR, 2013. 4

[7] Y. Zhang and L. Zhou. Fairness assessment for artificial intelligence in financial industry. *arXiv preprint arXiv:1912.07211*, 2019. 4

A. Cross validation plots showing model selection process

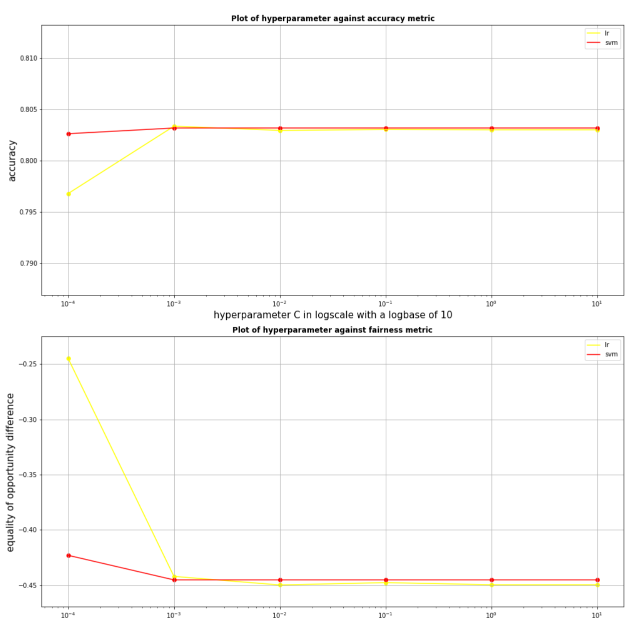

Figure 1. Cross validation plots of C values(x-axis in logscale) against accuracy and fairness for lr and svm models for task 1 model selection at repeat 5, for adult dataset.

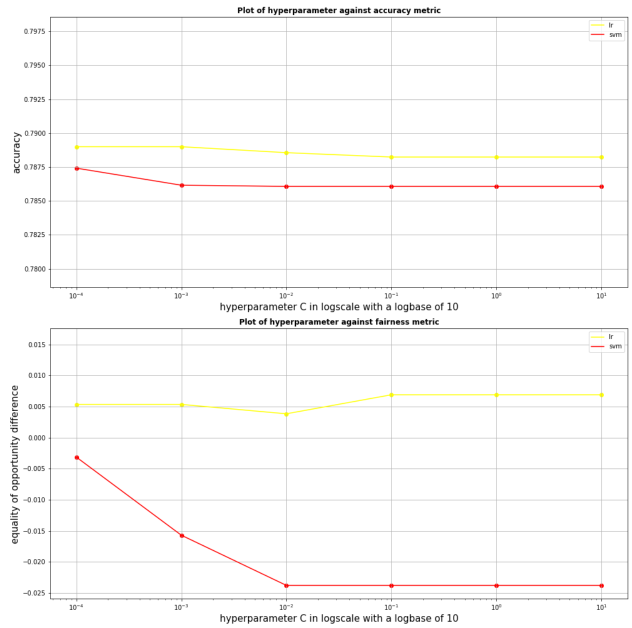

Figure 2. Cross validation plots of C values(x-axis in logscale) against accuracy and fairness for lr and svm models for task 2 model selection at repeat 5, for adult dataset.

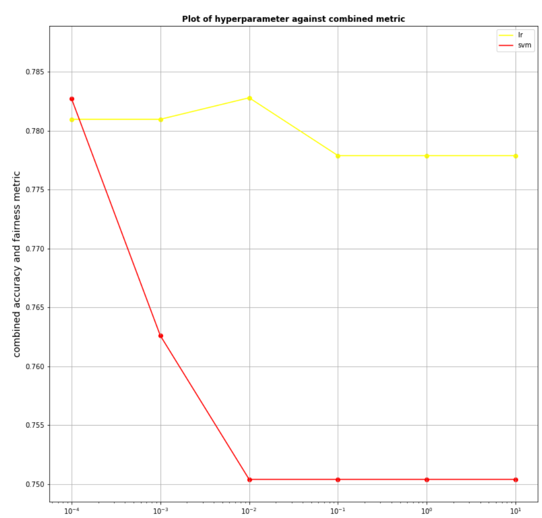

Figure 3. Cross validation plot of C values(x-axis in logscale) against the combined metric(accuracy + fairness) for selecting model 5(fairness-method based) in task 3, at repeat 5, for adult dataset.

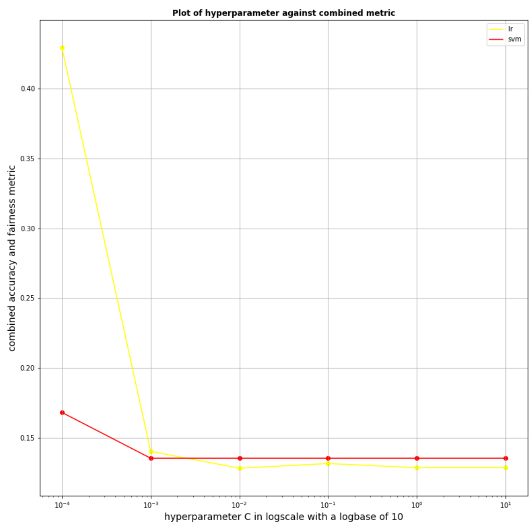

Figure 4. Cross validation plot of C values(x-axis in logscale) against the combined metric(accuracy + fairness) for selecting model 6(standard) in task 3, at repeat 5, for adult dataset.

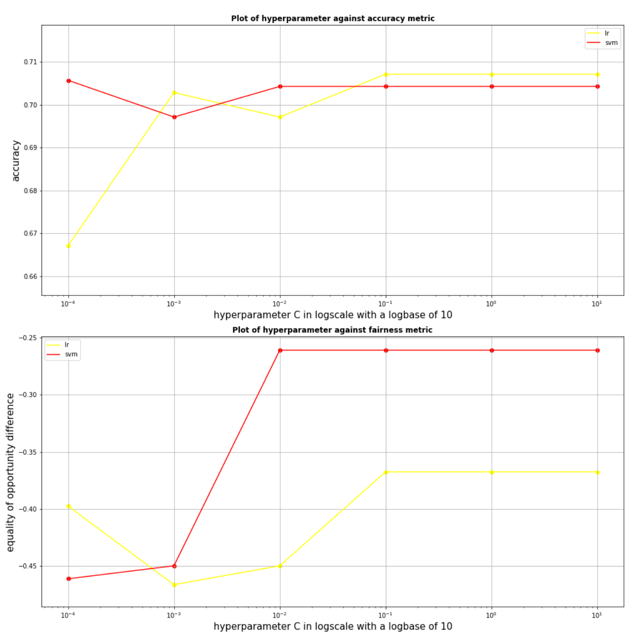

Figure 5. Cross validation plots of C values(x-axis in logscale) against accuracy and fairness for lr and svm models for task 1 model selection at repeat 2, for German dataset.

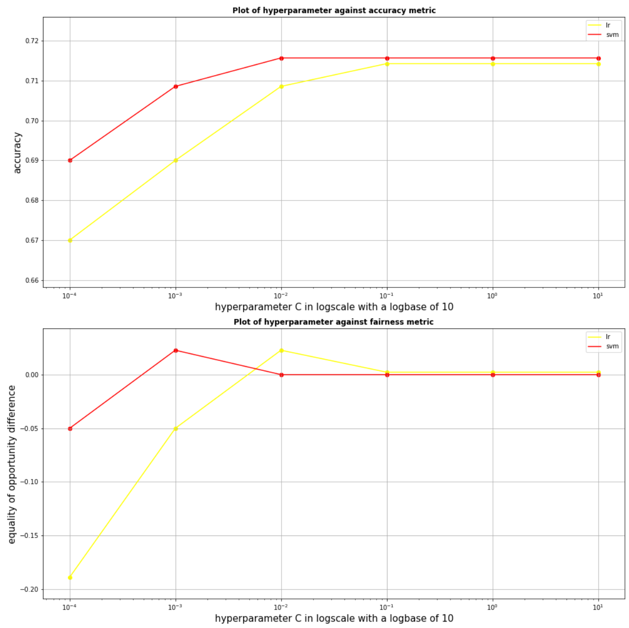

Figure 6. Cross validation plots of C values(x-axis in logscale) against accuracy and fairness for lr and svm models for task 2 model selection at repeat 2, for German dataset.

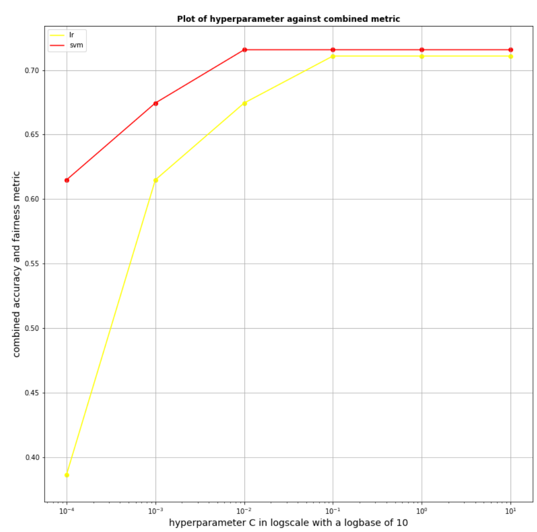

Figure 7. Cross validation plot of C values(x-axis in logscale) against the combined metric(accuracy + fairness) for selecting model 5(fairness-method based) in task 3, at repeat 2, for German dataset.

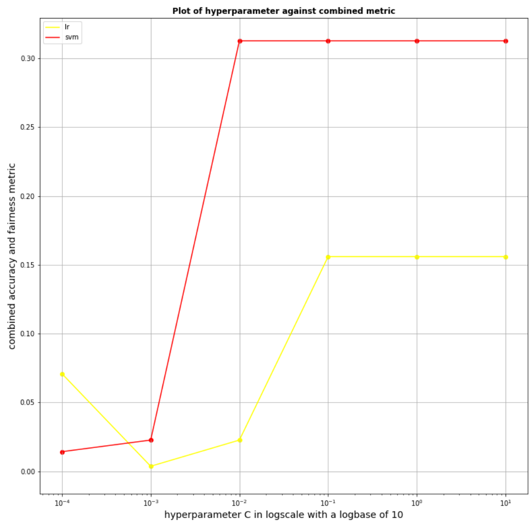

Figure 8. Cross validation plot of C values(x-axis in logscale) against the combined metric(accuracy + fairness) for selecting model 6(standard) in task 3, at repeat 2, for German dataset.