Appendix to: Pushing the limits of fairness impossibility: Who's the fairest of them all?

A Mapping from bins to scores

As mentioned, the most straightforward method of applying the score transformation after solving the optimization problem is to sample from a multinomial distribution. However, this is a less granular approach as we are assuming that all observations in the bin are indistinguishable. To overcome this, we recommend the idea proposed in ([**?**]) which is a linear projection. This strategy proposes that if an observation with score s falling into a bin a with upper and lower bounds $[a_l, a_u]$ gets mapped from the random draw into a new bin b_1 with bounds $[b_{1l}, b_{1u}]$, then we assign it a linearly interpolated score given by:

$$s' = b_{1l} + \frac{s - a_l}{a_u - a_l}(b_{1u} - b_{1l})$$

This allows us to maintain rank-ordering of scores that receive the same assignment from a to b.

A more deterministic manner of mapping from bin to score would be to take the expected score mapping. After solving the optimization problem, we know the transitions probabilities a to $\{b_1, b_2, \ldots, b_B\}$ (denoted as $P(a \longrightarrow b_i)$) based on the optimization variables and from the previous method, we also know the score assignment if a were moved into b_i (denote as s_i). Hence, a deterministic map would transform score s to $s' = \sum_{i \in B} s_i P(a \longrightarrow b_i)$.

B Details on the fractional LP subproblem for bound tightening

We elaborate on the methodology in Section **??**. Recall that our goal is to find bounds for:

$$v_{b'}^{[g]} = \sum_{b \in \mathcal{B}} x_{bb'}^{[g]} N_b^{[g]} \qquad \text{and} \qquad t_{b'}^{[g]} v_{b'}^{[g]} = \sum_{b \in \mathcal{B}} x_{bb'}^{[g]} N_{b+}^{[g]} \qquad \forall\ b' \in \mathcal{B}.$$

Where $t_{b'}^{[g]}$ is meant to represent the fractional quantity:

$$t_{b'}^{[g]} = \frac{\sum_{b \in \mathcal{B}} x_{bb'}^{[g]} N_{b+}^{[g]}}{\sum_{b \in \mathcal{B}} x_{bb'}^{[g]} N_b^{[g]}} = \frac{\sum_{b \in \mathcal{B}} x_{bb'}^{[g]} N_{b+}^{[g]}}{v_{b'}^{[g]}}$$

We will do this by fixing $\bar{g}$ and $\bar{b}$ such that we first tighten bounds for $v_{\bar{b}}^{\bar{g}}$ and then use the optimal solution to tighten bounds for $t_{\bar{b}}^{\bar{g}}$. First, it is easy to see that maximizing/minimizing $v_{\bar{b}}^{\bar{g}}$ is an LP as we have dropped the quadratic constraints, leaving us with a linear objective and linear constraint set. Now let $v_{\bar{b},min/max}^{\bar{g}*}$ represent the optimal values of the min/max objective for $v_{\bar{b}}^{\bar{g}}$. We now turn to bounding $t_{\bar{b}}^{\bar{g}}$ which has the same linear constraints but a fractional (nonlinear) objective. To deal with this, we utilize the Charnes-Cooper transformation ([**?**]). Essentially, this reformulation trick handles the denominator by removing it from the objective and passing it to all constraints while maintaining linearity. To illustrate this in detail, we first define new optimization variables:

$$\xi_{bb'}^{[g]} = \frac{x_{bb'}^{[g]}}{\sum_{b \in \mathcal{B}} x_{b\bar{b}}^{[\bar{g}]} N_b^{[\bar{g}]}} \qquad \phi_{\bar{b}}^{[\bar{g}]} = \frac{1}{\sum_{b \in \mathcal{B}} x_{b\bar{b}}^{[\bar{g}]} N_b^{[\bar{g}]}} \tag{1}$$

36th Conference on Neural Information Processing Systems (NeurIPS 2022).

Using (1), we can express the min/max problem for $t_{\bar{b}}^{[\bar{g}]}$ as problem (2).

$$
\begin{aligned}
\underset{\xi_{bb'}^{[g]},\phi}{\text{Min or Max}} \quad & t_{\bar{b}}^{[\bar{g}]} = \sum_{b\in\mathcal{B}} N_{b+}^{[\bar{g}]} \xi_{b\bar{b}}^{[\bar{g}]} \\
\text{subject to} \quad & \sum_{b\in\mathcal{B}} \xi_{b\bar{b}}^{[g]} N_b^{[\bar{g}]} = 1 \\
& \xi_{bb'}^{[g]} \geq (1-m)\phi \qquad \forall\ b = b' \\
& \xi_{bb'} = 0 \qquad \forall\ b'\ s.t.\ |b'-b| \geq w \\
& \left| \frac{1}{N^{[1]}} \sum_{b\in\mathcal{B}} \xi_{bb'}^{[1]} N_b^{[1]} - \frac{1}{N^2} \sum_{b\in\mathcal{B}} \xi_{bb'}^{[2]} N_b^{[2]} \right| \leq \epsilon_{DP}\phi \qquad \forall\ b' \in B \qquad (2) \\
& \left| \frac{1}{N_+^{[1]}} \sum_{b\in\mathcal{B}} \xi_{bb'}^{[1]} N_{b+}^{[1]} - \frac{1}{N_+^{[2]}} \sum_{b\in\mathcal{B}} \xi_{bb'}^{[2]} N_{b+}^{[2]} \right| \leq \epsilon_{EOdds}\phi \qquad \forall\ b' \in B \\
& \left| \frac{1}{N_-^{[1]}} \sum_{b\in\mathcal{B}} \xi_{bb'}^{[1]} N_{b-}^{[1]} - \frac{1}{N_-^{[2]}} \sum_{b\in\mathcal{B}} \xi_{bb'}^{[2]} N_{b-}^{[2]} \right| \leq \epsilon_{EOdds}\phi \quad \forall\ b' \in B \\
& \frac{1}{v_{b'max}^{g*}} \leq \phi \leq \frac{1}{v_{b'min}^{g*}} \qquad 0 \leq \xi_{bb'}^{[g]} \leq \frac{1}{v_{b'min}^{g*}}
\end{aligned}
$$

By solving these subproblems and taking the objective value as bounds for $t_b^{[g]}$, we can reduce the feasible region of the problem and enhance our solutions. The sub-problems are bounded and if any of them are infeasible, then it also implies that the MFOpt problem is also infeasible as we drop the PRP constraints in these sub-problems:

C Experiment data descriptions and problem parameters

We use three primary data sources for our experiments, the more recently developed American Community Survey (ACS) data as well as two more classical datasets, Heart Disease and COMPAS. We elaborate on each dataset in this section. The ACS data is a dataset made publicly available by the US Census Bureau. Specifically, Ding et. al [**?**] have created an excellent Python package[1] that enables users to pull model-ready data (for a requested year and geographic region) for a set of pre-defined binary classification tasks, such as predicting high income, health insurance coverage, whether they move or not, among others. The tasks are detailed in the paper and we use all of the pre-defined tasks without any additional modification except for Employment. We do not use the Employment task because of the assumption detailed in Section **??** regarding overlap. Experiments with this task occasionally yielded models that did not have overlap which made this task unsuitable for demonstrating our methodology. We reiterate that this is not a practical issue if one just ignored the non-overlapping bins, but requires a lengthy and technical fairness interpretation that we felt were beyond the purpose of our study. In terms of time and geography, we use 2020 data for all experiments while the geography varies. In the experiments shown on Table **??**, we use the West Coast US states (California, Oregon, Washington). In **??** we wanted a larger dataset as we required a sufficiently large testing split, hence we used the West Coast States ('CA', 'OR', 'WA', 'NV', 'AZ') with the "ACS Mobility" dataset and a 60/40 train-test split for the inprocessing comparison and East Coast States ('ME','NH','MA','RI', 'CT','NY', 'NJ', 'DE','MD','VA', 'NC','SC','GA','FL') and same with the "ACS Poverty" dataset for the postprocessing comparison. There was no particular reason for selecting these geographies aside from obtaining a large enough sample that we can feasibly run multiple trials on. Though we are using census data there is no PII information nor any endangerment to the subjects in the data. However, we note that in practice, it is important to exercise caution and equity in picking groups to mitigate for, as selective mitigation of favored groups by a malicious practitioner can result in underperformance for deserving groups.

The Heart Disease Dataset ([**?**]) is a publicly available dataset where the task is to predict whether or not an individual has heart disease. Most applications of this data use the standard processed

[1] https://github.com/zykls/folktables (MIT License)

Table 1: Experiment Problem Parameters

# Trials	Bins	ϵ	Max Movement	Window Size	Solve Time	Precision
10	50	0.03	0.5	13	600s	1e-5

Table 2: Comparison with other fairness methods

Method	Metric	Base	Train Method	MF-Opt
Rezaei	AUC	0.7471 ± 0.003	0.6619 ± 0.0022	0.747 ± 0.003
	ϵ_{DP}	0.0117 ± 0.0014	0.0124 ± 0.0013	0.0088 ± 0.001
	ϵ_{EOdds}	0.0266 ± 0.007	0.0291 ± 0.0059	0.0167 ± 0.0029
	ϵ_{PRP}	0.109 ± 0.0145	0.1091 ± 0.0143	0.0986 ± 0.0133
Pleiss	AUC	0.8314 ± 0.0045	0.8145 ± 0.0104	0.8306 ± 0.0044
	ϵ_{DP}	0.0208 ± 0.0029	0.0145 ± 0.0023	0.0105 ± 0.0008
	ϵ_{EOdds}	0.0325 ± 0.0062	0.0257 ± 0.005	0.0144 ± 0.0017
	ϵ_{PRP}	0.1405 ± 0.0214	0.4149 ± 0.1824	0.1319 ± 0.0204

"Cleveland" data and we use sex as the group variable. We could not find a standard and preprocessed version of this data and did it ourselves by one-hot-encoding categorical variables.

The COMPAS dataset is based on the recidivism study noted in ([**?**]). We use the preprocessed version made available in the publicly available AIF360 package ([**?**]) [2] without any additional modification. In this dataset, the task is to predict whether or not an individual will recidivate and we use ethnicity as the group variable.

We list the problem parameters used to create the results in Table **??** in Table 1. We use the same parameters for all tasks. A particular note is that in Table **??**, the IPOPT method performs very poorly compared on the COMPAS and ACS Coverage data. This is because across the 10 trials we ran, the IPOPT algorithm had frequently failed to converge within the 10-minute time limit for these two datasets. From the outputs, we saw that convergence failure is accompanied by heavy violation of the predictive rate parity constraint (demographic parity and equalized odds are still satisfied) and a high objective value. The exact reason for frequent failure in these two datasets is unclear, however, we hypothesize that it is due to a relatively high predictive rate parity gap in the data which led to numeric issues. Such failures were not observed in the IP formulation while both solvers were provided the exact same problem parameters.

All experiments were run on a MacBook Pro with a 2.4GHz 8-Core Intel Core i9 processor with 32 GB RAM. We did not use the GPU for solving. Data, preprocessing steps, and the random forest models utilize Python's scikit-learn ([**?**], BDS License) package. The optimization model is coded through Julia's JuMP package ([**?**] MPL License). We use the Gurobi ([**?**] Academic License) and IPOPT ([**?**] Eclipse Public License) solvers for all problems.

Due to lack of space, we only showed the method comparison Table **??** for the testing data. We show the results on the training data here in Table 2.

D Comparison to other fairness definitions

We compare our bin-wise worst-case fairness definition with other fairness definitions seen in literature and explain why it does not contradict previous impossibility theorem results. First, since we are considering score bins, our definition is a generalization of the definitions in ([**?**]), ([**?**]) and ([**?**]), which consider fairness metrics for binary $\{0, 1\}$ classifiers or assume that there is a threshold

[2] https://github.com/Trusted-AI/AIF360

for mapping probabilities to 0,1 outcomes. In these cases, the overall FPR/FNR can be computed and EOdds refers to the equality of those rates. Our framework is a generalization since the same fairness metrics for binary classification can be achieved by specifying that all scores be moved into exactly one of two bins (representing $\{0,1\}$ predictions) under our framework.

Since we are dealing with binned scores, our fairness definitions more resemble those seen in [**?**], which also has a notion of binned "risk assignments". The critical difference in fairness definitions is that Kleinberg's paper utilize the sum of scores in each bin compared against the number of positive or negative instances. Under this scheme, predictive rate parity refers to having the sum of scores be equal to the number of positive instances and true positive rate refers to the expected score of the positive instances in each bin (and analogously with FPR). The key difference is that rather than the sum of scores, our definitions are based on the expected number of $\{0,1\}$ instances moved into each bin, irrespective of the instance's original scores. As such, we are not faced with the same strict fairness trade-offs.

E Commentary on other solution methods and solvers

While investigating a solution to our nonconvex problem, we considered another global integer programming based approach known as spatial branch and branch (SBB), which relies on a combination of spatial partitioning and solving local partitions using McCormick relaxations ([**?**], [**?**]) and other outer approximation variants ([**?**]). In our testing, Gurobi's nonconvex QCQP solver, which applies these SBB heuristics, worked remarkably well despite being relatively new and was sometimes able to beat both the interior point solution and the MIP solution. Open-source solver SCIP ([**?**]) also features a gender nonconvex SBB solver that works reasonably well. However, our main goal was to provide a widely accessible method of solving the problem to global optimality and as of writing, there are significantly more developed open-source MILP solvers, such as SCIP, HiGHS ([**?**]), and CBC ([**?**]), than SBB solvers. Another reason we opted for the MILP approach is that we saw more potential in the NMDT reformulation for taking advantage of our reformulation and bound tightening procedure. Nonetheless, our bound tightening procedure is theoretically beneficial for both methods and as other open-source algorithms/solvers for SBB become more developed, such as Couenne ([**?**]) and Alpine ([**?**]), we encourage a future re-evaluation of solution methods and comparisons.

Finally, we note that we chose Gurobi in our experiments for its speed and effectiveness since we are repeatedly solving many problems. We acknowledge that Gurobi is a very powerful commercial solver and the results solved over 10 minutes may be worse with open source solvers such as HiGHS ([**?**]). Nonetheless, the important fact is that all MIP solvers target the global optimum and hence even less powerful solvers can yield strong solutions given more time.

F Additional Experiments Using Expected Assignment on Testing Data

We list additional experiments focusing on the performance of our method on the testing data in Tables 3 to 8. All results are based on 20 trials that are run with a similar procedure in the comparison section **??** except that we compute the testing metrics based on expected assignment rather than stochastic assignment (explained below), which we feel better reflects the *average* performance of MFOpt. In each trial, we tune a random forest via grid-search, find the base fairness violations, set up the parameters of MFOpt to reduce the violations by a half, and optimize. Then, we compute the results (AUC and fairness violations) on the testing data (baseline with no modifications vs. MFOpt) and show the 1-Standard deviation error margins as well as the p-value corresponds to a one-sided Wilcoxon signed rank test which evaluates if the distribution of differences of the Base - MFOpt stats (higher AUC, lower fairness violations) is symmetric around zero (null) or instead favors the base (alternative).

The major difference in this evaluation compared to the results shown in section **??** is that in section **??**, we assign observations to bins in the testing data based on random draws from a multinomial distribution (explained in Appendix A). However, a single draw per train-test split may not properly reflect the expected performance of MFOpt, even if we average over 20 trials. Instead, we feel a more accurate representation of the expected performance is if we apply the expected bin assignment to obtain the post-movement number of $\{0, 1\}$ and total samples for each bin in the testing set. Under this procedure, we do not move individual observations across bins, but rather move all of

them together. Concretely, suppose that we have optimized parameters $(\hat{x}^{[g]}_{1b'}, \hat{x}^{[g]}_{2b'}, \ldots, \hat{x}^{[g]}_{Bb'})$, which represented the probabilities of observations from each bin moving into bin b'. Then we propose that the number of $\{1\}$ outcomes at bin b' after the expected bin assignment procedure is:

$$\hat{N}^{[g]}_{b'+} = \sum_{b \in \mathcal{B}} \hat{x}^{[g]}_{bb'} N^{[g]}_{b+}$$

The same computation applies for the expected number of negative samples $\hat{N}^{[g]}_{b'-}$ and total samples $\hat{N}^{[g]}_{b'}$, which we use to compute the AUC and fairness violations and report in the tables. We find that across different datasets, the decrease in AUC is miniscule in terms of both absolute amount and variance (less than 1%). We obviously do not expect better AUC from the MFOpt solution compared to the unconstrained model and thus this result is remarkable as it indicates that some degree of fairness can be afforded practically for free under our framework. The second observation is that we can reduce all three fairness metrics simultaneously and consistently across all datasets, as we find p-values below 0.05 in all cases. We do observe variance of the PRP violation is relatively higher than that of DP or EOdds. We noted this in our conclusion Section **??** as an area for future work and provide some hypotheses for methods that can address this inconsistency. Nonetheless, improving conflicting definitions of fairness simultaneously is another significant result as it provides empirical evidence that there is a path forward towards multiple fairness.

Table 3: ACS West Travel

Metric	Base	MFOpt	Wilcoxon p-value
AUC	0.7439 ± 0.0039	0.7437 ± 0.0039	0.999999
DP	0.0313 ± 0.0057	0.0208 ± 0.0045	0.000001
EOdds	0.0404 ± 0.0055	0.0268 ± 0.0081	0.000001
PRP	0.1743 ± 0.0326	0.1481 ± 0.0306	0.000001

Table 4: ACS West Income

Metric	Base	MFOpt	Wilcoxon p-value
AUC	0.8907 ± 0.0012	0.8902 ± 0.0016	1.000000
DP	0.0172 ± 0.0021	0.0126 ± 0.0019	0.000001
EOdds	0.0362 ± 0.006	0.0231 ± 0.0035	0.000001
PRP	0.1994 ± 0.0745	0.1507 ± 0.0295	0.000024

Table 5: ACS West Mobility

Metric	Base	MFOpt	Wilcoxon p-value
AUC	0.7413 ± 0.0033	0.7412 ± 0.0033	0.999284
DP	0.0183 ± 0.0057	0.0149 ± 0.0033	0.008591
EOdds	0.0469 ± 0.0153	0.0327 ± 0.0068	0.000031
PRP	0.197 ± 0.0318	0.1738 ± 0.0311	0.000024

Table 6: ACS West Insurance

Metric	Base	MFOpt	Wilcoxon p-value
AUC	0.7183 ± 0.0028	0.7182 ± 0.0029	0.999916
DP	0.0603 ± 0.0089	0.0322 ± 0.0041	0.000001
EOdds	0.0676 ± 0.0072	0.0579 ± 0.0062	0.000052
PRP	0.3732 ± 0.11	0.205 ± 0.0446	0.000018

Table 7: ACS West Poverty

Metric	Base	MFOpt	Wilcoxon p-value
AUC	0.8319 ± 0.0039	0.8316 ± 0.0039	1.000000
DP	0.0246 ± 0.0038	0.0154 ± 0.0017	0.000001
EOdds	0.0396 ± 0.0086	0.0226 ± 0.0024	0.000001
PRP	0.1305 ± 0.0172	0.1173 ± 0.019	0.000018

Table 8: ACS West Public Coverage

Metric	Base	MFOpt	Wilcoxon p-value
AUC	0.7932 ± 0.0016	0.7924 ± 0.0018	1.000000
DP	0.03 ± 0.0041	0.0204 ± 0.0028	0.000001
EOdds	0.0403 ± 0.0061	0.0236 ± 0.0028	0.000001
PRP	0.159 ± 0.0245	0.1443 ± 0.0297	0.029129

G Role of Bins Parameter and Ablation Study

In this section, we first comment on the choice of the number of bins, $|\mathcal{B}|$, as a hyperparameter and then show results of additional testing with respect to the choice of the bin parameter. In practice, the choice of $|\mathcal{B}|$ should reflect the required bin granularity of the outputs for usage in, for example, rank-ordering or threshold-based classification. Our work therefore approaches choosing $|\mathcal{B}|$ purely from a computational complexity perspective as the number of optimization variables scales in the order of $\mathcal{O}(|\mathcal{G}||\mathcal{B}|^2)$. We used $|\mathcal{B}| = 50$ across our experiments as we found that this parameter gave a reasonable degree of granularity in the resulting bins while also being small enough to solve (our MIP solver could frequently find solutions with <10% optimality gap within 10 minutes). From a more theoretical perspective, the choice of $|\mathcal{B}|$ determines how well we can estimate the score transformation function, with higher $|\mathcal{B}|$ giving us better estimates in the training data. However, having too few samples within each bin (in the training data) results in low bias but high variance estimates of the score transformation function and the corresponding performance/fairness metrics, leading to bad generalization on the testing set. Having too few bins can, on the other hand, lead to an under-parameterized score transformation function. While it is clear that $|\mathcal{B}|$ should scale with respect to the per-bin sample size, we consider a rigorous analysis of this choice to be out of scope as a current limitation of our work. We hence recommend choosing $|\mathcal{B}|$ in practice based on the desired granularity of results and/or treating it as a parameter to optimize via cross-validation.

To run an experiment to study the effects of changing $|\mathcal{B}|$, we must first define $|\mathcal{B}|$-agnostic metrics to evaluate the post-processing output in terms of performance and fairness. The primary difficulty is that since fairness is defined on a bin-wise basis, evaluating the worst-case violation for example using $|\mathcal{B}| = 10$ is not comparable to $|\mathcal{B}| = 50$. Hence, we do the following: First, after optimizing for $x_{bb}^{[g]*}$, we apply the linear interpolation based mapping as described in Appendix A so that we have new scores s'. To compare the accuracy across different $|\mathcal{B}|$, we compute the ROC and precision-recall (PR) AUC based on s'. To compare the fairness, we discretize s' in 100 bins (regardless of which $|\mathcal{B}|$ we used) and compute the worst case violation for each fainess metric across all 100 bins. The reason we use 100 bins to assess fairness is that we specifically use a large dataset (500k+ instances in both training and testing) to obtain more granular metrics. To generate the results, we use the ACS Income data as presented in previous sections. The difference is that we now use data across all states to obtain a large sample for a single random 60/40 train-test split, each with over 500k samples. Following the procedure from before, we train and tune a random forest model, score the training and testing data, and proceed with our evaluation methodology. Since the linear interpolation mapping methodology utilizes stochastic draws, we sample 300 draws per bin and compute the 1 standard-deviation errors shown in the tables.

Table 9 shows the performance metrics across different $|\mathcal{B}|$, where we show the average metric and 1-standard deviation error margins. We observe that performance is very similar across the different choices, but is maximized at the higher bin counts of of 50-60. The performance also appears to plateau beyond a certain point, suggesting that it is not necessary to select a large $|\mathcal{B}|$ for performance

purposes. Interestingly, we also see that the solutions are near-optimal based on the optimality gap. This shows that although the number of variables scales with $|\mathcal{B}|$, it does not necessarily preclude us from high quality solutions. On the contrary, the optimality gap for lower bin ranges such as 25 is larger with lower performance. This could be due to the fact that since we are optimizing over the same ϵ fairness criteria, having fewer degrees of freedom with smaller bins makes it more difficult to find feasible solutions.

Table 10 shows the results for fairness, where we show the average worst fairness violation and the 1-standard deviation error margins. The story is less clear from this angle but we emphasize that training the problem for different $|\mathcal{B}| \neq 100$ bins using the same ϵ parameters and evaluating it on $|\mathcal{B}| = 100$ for fairness is an unintended method of using our framework which we are only doing to have comparable results for the ablation study. Here, we see that the average metrics are similar for different parameters with $|\mathcal{B}| = 40$ having the best overall result. Demographic parity violation is surprisingly minimized at $|\mathcal{B}| = 30$ in this example, but given the other fairness and performance metrics $|\mathcal{B}| = 40$ appears to be the best choice. Under this method of evaluation, the fairness parameters do not appear to be very sensitive to the choice of $|\mathcal{B}|$.

Table 9: Bin Ablation Study (Performance) - ACS Income

Num. Bins	Optimality Gap	ROC AUC	PR AUC
25	0.1483	0.8712 ± 0.0001	0.7933 ± 0.0002
30	0.0672	0.8733 ± 0.0	0.7969 ± 0.0002
35	0.0326	0.8739 ± 0.0	0.7981 ± 0.0002
40	0.0095	0.874 ± 0.0	0.8001 ± 0.0001
45	0.0096	0.8742 ± 0.0	0.8 ± 0.0001
50	0.0144	0.8743 ± 0.0	0.8002 ± 0.0001
55	0.0150	0.8742 ± 0.0	0.8005 ± 0.0001
60	0.0212	0.8743 ± 0.0	0.8011 ± 0.0001

Table 10: Bin Ablation Study (Fairness) - ACS Income

Num Bins	Demographic Parity	Equalized Odds	Predictive Rate Parity
25	0.0174 ± 0.0004	0.0224 ± 0.0008	0.1447 ± 0.013
30	0.0144 ± 0.0004	0.0206 ± 0.0005	0.1702 ± 0.0101
35	0.0177 ± 0.0003	0.0235 ± 0.0019	0.1535 ± 0.0149
40	0.0179 ± 0.0004	0.0195 ± 0.0005	0.1295 ± 0.013
45	0.0174 ± 0.0003	0.0234 ± 0.0008	0.1406 ± 0.0182
50	0.0215 ± 0.0003	0.0239 ± 0.0004	0.1456 ± 0.0143
55	0.0165 ± 0.0004	0.021 ± 0.0006	0.1323 ± 0.0127
60	0.0179 ± 0.0004	0.0237 ± 0.0007	0.1503 ± 0.0178

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes] In the discussion **??**
   (c) Did you discuss any potential negative societal impacts of your work? [Yes] In Section C
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...

(a) Did you state the full set of assumptions of all theoretical results? [N/A] We do not derive any theoretical results and all claims (e.g. NP-hardness of non-convex programs) are either well-known or a source is referenced.

(b) Did you include complete proofs of all theoretical results? [N/A]

3. If you ran experiments...

(a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] Yes, all code is included in a zip file uploaded as the supplementary materials. Instructions and data used in all experiments are clearly outlined in C and **??**

(b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] Referenced in Appendix C

(c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes]

(d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] Referenced in Section C

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...

(a) If your work uses existing assets, did you cite the creators? [Yes] License and references can be found in the Appendix Section C

(b) Did you mention the license of the assets? [Yes] License are mentioned alongside open-source package references

(c) Did you include any new assets either in the supplemental material or as a URL? [No]

(d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]

(e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [Yes] Under the Experimental Data Section C

5. If you used crowdsourcing or conducted research with human subjects...

(a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]

(b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]

(c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]