On Coresets for Fair Regression and Individually Fair Clustering

Rachit Chhaya
IIT Gandhinagar

Anirban Dasgupta
IIT Gandhinagar

Jayesh Choudhari
University of Warwick

Supratim Shit
Technion

Abstract

In this paper we present coresets for Fair Regression with Statistical Parity (SP) constraints and for Individually Fair Clustering. Due to the fairness constraints, the classical coreset definition is not enough for these problems. We first define coresets for both the problems. We show that to obtain such coresets, it is sufficient to sample points based on the probabilities dependent on combination of sensitivity score and a carefully chosen term according to the fairness constraints. We give provable guarantees with relative error in preserving the cost and a small additive error in preserving fairness constraints for both problems. Since our coresets are much smaller in size as compared to n, the number of points, they can give huge benefits in computational costs (from polynomial to polylogarithmic in n), especially when $n \gg d$, where d is the input dimension. We support our theoretical claims with experimental evaluations.

1 INTRODUCTION

Automated data driven decisions play an increasingly important role in a number of settings, and correspondingly, the need for ensuring fairness of such automated procedures has become critical (Barocas and Selbst, 2016; Caton and Haas, 2020; Corbett-Davies and Goel, 2018). This has naturally led the machine learning research community to propose various measures of fairness.

Fairness notions are introduced in the traditional machine learning problems either by imposing constraints on the solutions space (to ensure, e.g., that no group suffers a much smaller response than the others), or via changing the objective function itself (e.g. optimizing for the worst group-wise distortion instead of the aggregate). Incorporating fairness notions into the optimization can often increase the complexity of the resulting optimization that needs to be solved. For instance, to solve a regression problem (for n points in d dimensions) with the statistical parity fairness notion, it takes roughly $O(n^2)$ time (Agarwal et al., 2019), which is far worse than the $O(nd^2)$ time (Golub and Van Loan, 1996) required to solve linear regression by itself, especially when $n \gg d$. For individually fair clustering the dependence on n is n^4 which can be prohibitive even for medium sized datasets. This is not always desirable from a practical standpoint, as it could lead to the unintended consequence of fairness definitions not being adopted widely.

Our work aims to mitigate this added computational burden. We employ the use of coresets, a data summarization technique, in order to create a small dataset with the guarantee that solving the optimization problem on the smaller dataset will give a solution that maintains both the fairness guarantees as well as gives a close approximation to the objective function.

The main technical challenge is the following: incorporating the coreset technique into a fairness problem cannot always be done blindly. In particular, for certain fairness definitions (e.g. the ones we consider), the space of valid solutions (aka *query space*) depends on the dataset on which we are evaluating the solution. This implies that once we subsample the dataset, the valid query space changes. Existing coresets for non-fair versions of ML problems only preserve the cost. Since the coreset creation procedure is not even aware of any constraints, it is not clear what kind of theoretical guarantees such traditional coresets can give for fair versions of the ML problems. For example when the fairness notion involves some protected attribute like gender or race, it is possible that a traditional coreset might not have representation from all groups and hence it is not clear why it should work with guarantees. Such situations involving constraints and changing of feasible solutions with changing subsamples have

Proceedings of the 25$^{\text{th}}$ International Conference on Artificial Intelligence and Statistics (AISTATS) 2022, Valencia, Spain. PMLR: Volume 151.

not been tackled in the coreset literature. From this perspective, following are our main contributions:

1. We define coresets for fair regression with statistical parity (SP) that was defined by Agarwal et al. (2019). Using a combination of leverage scores and stratified uniform sampling, we create a coreset that ensures that the solution obtained from the coreset is a close approximation to the optimal and maintains the statistical parity based fairness. The size of our coreset for fair regression is independent of the number of input points, depending only on the dimension and the number of distinct values the protected attribute can take.
2. We define coresets for individually fair clustering defined by Jung et al. (2019) and studied by Mahabadi and Vakilian (2020). For this we modify the definition of fair radius and using this modified definition we define the fair coreset. We give an algorithm to build such coresets of size logarithmic in n, in time $\tilde{O}(nkd)$. This gives us a $O(nkd + k^8d^4 + (k \log n)^4)$ constant factor approximation algorithm, a significant improvement from the original $O(n^4)$ algorithm obtained by Mahabadi and Vakilian (2020).
3. We provide empirical results on real and semi synthetic data to support our theoretical claims.

To the best of our knowledge, this is the first work giving coresets for these specific fair regression and clustering problems. While our algorithms to construct coresets are obtained using modifications to the unified framework given by Feldman and Langberg (2011), the novelty lies in framing the definitions of fair coresets in the setting of optimization with data-dependent constraints and in showing that combining the stratified uniform sampling with (known) sensitivity scores i.e. little careful oversampling can indeed give fair coresets.

2 RELATED WORK AND NOTATION

Coresets, small summaries of data that can be used as proxy for original data for some optimization task, have been used in making number of optimization tasks efficient, see (Bachem et al., 2017; Feldman, 2020) and references therein. Coresets are extensively studied for problems like regression (Dasgupta et al., 2009; Chhaya et al., 2020c) and clustering (Har-Peled and Mazumdar, 2004; Chhaya et al., 2020a). Our work follows the unified coreset framework given by Feldman and Langberg (2011). Coresets for clustering without fairness are given by Langberg and Schulman (2010); Feldman and Langberg (2011); Bachem et al. (2018a) and many others. Similarly leverage scores based coresets for regression without fairness have been given by Drineas et al. (2006, 2011); Dasgupta et al. (2009)

There has been a significant work in incorporating fairness into various classification and clustering problems (Chierichetti et al., 2017; Ghadiri et al., 2021; Hardt et al., 2016; Kleinberg et al., 2017). In (Kamishima et al., 2012; Berk et al., 2017), authors used regularization to enforce fairness, Calders et al. (2013); Zink and Rose (2020); Berk et al. (2018) studied regression with fairness constraints by proposing linear models with constraints on residuals of model and hence controlled bias. For other such results see (Alabi et al., 2018; Komiyama et al., 2018; Pérez-Suay et al., 2017), (Mehrabi et al., 2021) being a nice survey.

Our work deals with the fair regression with statistical parity (SP) defined by Agarwal et al. (2019). Agarwal et al. (2018) also propose algorithms for SP-fair and bounded group loss (BGL) constraints based on reduction to classification and linear regression problems. Though their definition applies to any form of regression with the predictions constrained in $[0, 1]$, our coresets are for linear and ℓ_p regression problems with SP constraints.

Individually fair clustering was first introduced by Jung et al. (2019). Intuitively it requires every point to have a corresponding center within its closest n/k nearest neighbours. Jung et al. (2019) only gave an algorithm that gives a feasible clustering solution and does not consider optimizing the clustering cost. Mahabadi and Vakilian (2020) gave an algorithm that gives 7α approximation in terms of fairness and $O(1)$ approximation for k-median and k-means as clustering costs and $p^{O(p)}$ approximation for optimal cost with general ℓ_p norm. The algorithm's running time is $O(n^4)$ which makes it prohibitive. Vakilian and Yalçıner (2021); Chakrabarty and Negahbani (2021) further improve the algorithm either in terms of the fairness guarantees or the cost but *not the running time*. Our work tackles this open problem of reducing the time complexity.

Coresets have been given for other definitions of fair clustering problems. Schmidt et al. (2019) construct coresets for a *balance*-based fairness definition given by Chierichetti et al. (2017) for k-means. Huang et al. (2019) improved on the size of the fair coreset for the k-means problem and provide the first fair coreset for the fair k-median problem. Bandyapadhyay et al. (2020) further improve upon these coresets in terms of dependence on ϵ.

General Notation and Background: Matrices and sets will be denoted by bold face uppercase letters and

vectors by boldface lower case. By default all vectors are treated as column vectors. $\mathbf{A}$ denotes both a matrix or the set of its rows (data points) interchangeably. For some $p, q > 0$, the notation $p \in (1 \pm \epsilon)q$ means $(1-\epsilon)q \leq p \leq (1+\epsilon)q$. The symbol $\mathbf{a}_i$ represents the i^{th} row of matrix $\mathbf{A}$ represented as a column vector while b_i represents the i^{th} coordinate of a vector $\mathbf{b}$. All statements where we say "high probability", hold with probability at least some large constant, e.g. 0.99, unless otherwise stated. The notation $\tilde{O}(*)$ hides poly-logarithmic terms in complexity.

Coresets: Let $\mathbf{A}$ be a weighted dataset (weights w_a), $\mathbf{x}$ be a query from the query space $\mathbf{X}$, and F be a non-negative cost function of the form $F(\mathbf{A}, \mathbf{x}) = \sum_{\mathbf{a} \in \mathbf{A}} w_a F(\mathbf{a}, \mathbf{x})$. A weighted set $\mathbf{C}$ is called an ϵ-strong (fixed $\epsilon > 0$) coreset for F, if $|F(\mathbf{A}, \mathbf{x}) - F(\mathbf{C}, \mathbf{x})| \leq \epsilon F(\mathbf{A}, \mathbf{x})$ **for all** $\mathbf{x}$ in the query space. We consider coresets $\mathbf{C}$ which are subsamples (reweighted) of original data. If $\mathbf{x}_{opt}$ is the optimal solution on the full data and $\tilde{\mathbf{x}}_{opt}$ is the optimal obtained from the coreset, it follows that $F(\mathbf{A}, \tilde{\mathbf{x}}_{opt}) \leq (1+3\epsilon)F(\mathbf{A}, \mathbf{x}_{opt})$ (see e.g. Langberg and Schulman (2010)). One of the most successful coreset techniques is the importance sampling using sensitivity scores, defined by Langberg and Schulman (2010). For a function F, query set $\mathbf{X}$ and a point $\mathbf{a}_i$, the i^{th} point's sensitivity σ_i is defined as

$$\sigma_i = \sup_{\mathbf{x} \in \mathbf{X}} \frac{F(\mathbf{a}_i, \mathbf{x})}{F(\mathbf{A}, \mathbf{x})}$$

Sensitivity captures the relative importance of a point w.r.t the cost function. It has been shown by Langberg and Schulman (2010); Feldman and Langberg (2011) that sampling points proportional to their sensitivities scores (or their upper bounds) give coresets for F of size equal to the sum of these scores times the pseudo-dimension of the query space. In this work coreset refers to a strong coreset, unless otherwise mentioned. All proofs and additional experiments are in appendix.

3 FAIR REGRESSION WITH SP CONSTRAINTS

We consider the fair regression problem as described by Agarwal et al. (2019). The input is a tuple $(\mathbf{A}, \mathbf{b}, \mathbf{G}, \{\zeta_i\})$. $\mathbf{A} \in \mathbb{R}^{n \times d}$ contains the data points in the rows. The vector $\mathbf{b} \in [0,1]^n$ contains the targets. Each data point i has a protected attribute, denoted by g_i, and $\mathbf{G} = \{g_i\}$. Each g_i can be one out of ℓ finite values, denoted by $[\ell]$. There are also ℓ values $\zeta_i \in [0,1]$, these are called slack variables. Let $\mathbf{A} = \bigcup_{j \leq \ell} \mathbf{A}_j$ where $\mathbf{A}_j$ is the dataset formed by the rows for which the protected attribute takes the value j. Each query $\mathbf{x}$ is from a query space $Q \subseteq \mathbb{R}^d$ such that, $\forall i$, $\mathbf{a}_i^T \mathbf{x} \in [0,1]$.

Consider a set of points $\mathbf{S}$, where each $\mathbf{s}_i \in \mathbf{S}$ has an associated weight w_i. Let $\mathbf{w}$ be the vector of w_i values. For a fixed query $\mathbf{x}$ and $z \in [0,1]$ we define the following function $\mathcal{F}_{\mathbf{x},z}(\mathbf{S})$ as $\mathcal{F}_{\mathbf{x},z}(\mathbf{S}) = \frac{\sum_{i:\mathbf{s}_i \in \mathbf{S}} w_i \cdot \mathbb{1}_i}{\sum_{i:\mathbf{s}_i \in \mathbf{S}} w_i}$ where, $\mathbb{1}_i = 1$ when $\mathbf{s}_i^T \mathbf{x} > z$, else it is 0. $\mathcal{F}_{\mathbf{x},z}(\cdot)$ thus returns the fractional weight of points from $\mathbf{S}$ whose predicted response, given the query $\mathbf{x}$, exceeds z. Fair regression with statistical parity (SP) constraints is then defined as follows (Agarwal et al., 2019). Given $\zeta_k \geq 0$,

$$\min_{\mathbf{x} \in \mathbb{R}^d : \forall i, \mathbf{a}_i^T \mathbf{x} \in [0,1]} 1/n \sum_i (\mathbf{a}_i^T \mathbf{x} - \mathbf{b}_i)^2 \text{ such that,}$$
$$\forall k \in [\ell],\ \forall z \in [0,1], |\mathcal{F}_{\mathbf{x},z}(\mathbf{A}_k) - \mathcal{F}_{\mathbf{x},z}(\mathbf{A})| \leq \zeta_k$$

In this paper, for ease of presentation, we consider weight of each input point in original dataset to be 1 but other weights can be used.

3.1 Coreset for Fair Regression with SP Constraints

When we have a strong coreset, the solution obtained from the coreset gives a good approximation of the total cost even when used with the full data. However, notice that because of the fact that the constraints are data-dependent (unlike, say, a norm-constraint on the solution vector, which is data-independent), we have to extend the standard coreset definition to guarantee viability of the constraints under the coreset creation procedure. To circumvent this problem, one of our main technical contributions is to define a coreset for the fair regression with SP constraints that puts additional requirements on the feasibility of the solution obtained. We next define the coreset formally for this problem.

Definition 1. *Consider a tuple $(\mathbf{C}, \tilde{\mathbf{w}}, \mathbf{b}_c, \{\zeta_i'\})$ such that $\mathbf{C} = \bigcup_{i \leq \ell} \mathbf{C}_i$ is a subset of rows of $\mathbf{A}$, $\tilde{\mathbf{w}}$ has the weights associated with the rows of $\mathbf{C}$, and $\mathbf{b}_c$ denotes the corresponding responses from $\mathbf{b}$. $\{\zeta_i'\}$ be a set of non-negative values. Recall that $\{\zeta_i\}$ denotes the slack corresponding to the i^{th} group. Let $\epsilon > 0$. We call the tuple $(\mathbf{C}, \tilde{\mathbf{w}}, \mathbf{b}_c, \{\zeta_i'\})$ to be an ϵ-coreset for fair regression with statistical parity if the following conditions are true.*

1. *All feasible solutions for $(\mathbf{A}, \mathbf{b}, \{\zeta_i\})$ are also feasible solutions for $(\mathbf{C}, \tilde{\mathbf{w}}, \mathbf{b}_c, \{\zeta_i'\})$.*

2. *For each group i, all feasible solutions for $(\mathbf{C}, \tilde{\mathbf{w}}, \mathbf{b}_c, \{\zeta_i'\})$ satisfy the i^{th} group constraint for original dataset for $\zeta_i + O(\epsilon)$, i.e. any $\mathbf{x}$ that is*

feasible for the coreset satisfies, for all $z \in [0,1]$,

$$|\mathcal{F}_{\mathbf{x},z}\ (\mathbf{A}_i) - \mathcal{F}_{\mathbf{x},z}\ (\mathbf{A})| \leq \zeta_i + O(\epsilon).$$

3. $\forall \mathbf{x}$, $\sum_{\mathbf{c}_i \in \mathbf{C}} \tilde{w}_i(\mathbf{c}_i^T\mathbf{x} - b_i)^2 \in (1 \pm \epsilon)\|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2^2$.

Notice that the solution obtained using the coreset $\mathbf{C}$ now satisfies that original constraints only with an additional slack $O(\epsilon)$. The advantage of this definition of coreset in practice is that it allows us to use the solution obtained from the coreset and use it in the original data and still get good approximation to the objective function value as well as the fairness criteria. Corollary 5 gives a formal version of this statement.

In order to replace the infinite constraints in the original problem with a finite number of constraints Agarwal et al. (2019) used a discretization trick. We first present a useful result that uses this trick in the coreset setting.

Discretization: Since we need to preserve the $\mathcal{F}_{\mathbf{x},z}$ for all $z \in [0,1]$, the number of constraints is infinite. To handle this Agarwal et al. (2019) use a discretization grid over z as well as the $\mathbf{a}_i^T\mathbf{x}$ and b_i's for all i. For a given $\eta \in (0,1)$, we consider a grid $\mathbf{Z}$ for $[0,1]$ defined as $\mathbf{Z} = \{\eta, 2\eta, \ldots, 1\}$ such that $|\mathbf{Z}| = \frac{1}{\eta}$. We also define an $\eta/2$- cover for all b_i's.

For our problem and the least squared loss function $\ell(b_i, \mathbf{a}_i^T\mathbf{x}) = (b_i - \mathbf{a}_i^T\mathbf{x})^2$ for each point, we represent the total average loss as $\frac{1}{n}\sum_{i\in[n]} \ell(b_i, \mathbf{a}_i^T\mathbf{x})$. Notice that $\ell(b_i, \mathbf{a}_i^T\mathbf{x})$ is a loss term and different from the ℓ used to denote number of distinct protected attributes. The way they are used will make it clear from the context to the reader. Now similar to Agarwal et al. (2019), we also define $\ell_\eta(b_i, \mathbf{a}_i^T\mathbf{x}) = \ell(\underline{b_i}, \lfloor \mathbf{a}_i^T\mathbf{x} \rfloor_\eta + \eta/2)$. Here $\lfloor a \rfloor_\eta = \lfloor a/\eta \rfloor$ rounds a down to the nearest integer multiple of η and $\underline{b_i}$ is the smallest b' in the $\eta/2$ cover of b_i's s.t $|b_i - b'| \leq \eta/2$. Since $\ell(b_i, \mathbf{a}_i^T\mathbf{x})$ is 1-Lipschitz, the absolute difference between the loss values ℓ and ℓ_η is bounded by η. Therefore when the discretized version of the problem is solved , an additive loss of η is incurred i.e. $|\|\mathbf{A}\mathbf{x}\|_2^2 - \ell_\eta(\mathbf{A}\mathbf{x})| \leq n\eta$. Here for some $\mathbf{x}$, with a little abuse of notation, $\ell_\eta(\mathbf{A}\mathbf{x}) = \sum_{i\in[n]} \ell_\eta(b_i, \mathbf{a}_i^T\mathbf{x})$. Similarly $\ell(\mathbf{A}\mathbf{x}) = \sum_{i\in[n]} \ell(b_i, \mathbf{a}_i^T\mathbf{x})$. Theorem 1 from Agarwal et al. (2019) states that we can solve the problem after discretization essentially incurring only an additive error of $O(\eta)$ compared to the original problem. This allows us to just consider the z in the grid $\mathbf{Z}$. To use this technique in the coreset setting we present the following lemma.

For ease of notation, we define $\mathbf{C}$ to be the set of chosen rows from $\mathbf{A}$, i.e. the rows with non-zero $\tilde{w}_i$ values associated with them.

In: Matrix $\mathbf{A}$ with group of each data point, response vector $\mathbf{b}$, error parameter ϵ
Out: Subsampled matrix $\mathbf{C}$, corresponding response vector, and weights

Let $\mathbf{C}$ = empty matrix; $\mathbf{b}_c$ = empty vector
Set $r_1 = \frac{\log d}{\epsilon^2}$, $r_2 = \frac{1+\epsilon}{\epsilon^2}$, as in lemma 3.
for $i = 1, 2, \ldots n$ **do**
  Set $q_i = r_1\|\mathbf{u}_i\|_2^2 + r_2 t_i$, where, t_i is as given in Lemma 3
  With probability $p_i = \min(q_i, 1)$, add i^{th} row of $\mathbf{A}$ to $\mathbf{C}$ and corresponding entry $\mathbf{b}$ to $\mathbf{b}_c$, and set weight $\tilde{w}_i = 1/p_i$.
Return $(\mathbf{C}, \mathbf{b}_c)$ and corresponding weights.

Algorithm 1: Coreset for Fair Regression

Lemma 2. *For a coreset* $\mathbf{C}$ *satisfying,* $\forall \mathbf{x}$, $\sum_{\mathbf{c}_i \in \mathbf{C}} \tilde{w}_i(\mathbf{c}_i^T\mathbf{x} - b_i)^2 \in (1 \pm \epsilon)\|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2^2$, *we have* $\frac{1}{n}|\ell_\eta(\mathbf{A}\mathbf{x}) - \ell_\eta(\mathbf{C}\mathbf{x})| \leq (2+\epsilon)\eta + (\epsilon/n)\ell_\eta(\mathbf{A}\mathbf{x})$.

3.2 Coreset Construction Algorithm

Let $\mathbf{U}$ be any orthogonal basis of the augmented matrix $[\mathbf{A}, -\mathbf{b}]$. We denote the i^{th} row of $\mathbf{U}$ as $\mathbf{u}_i$. The procedure to construct coreset is described as pseudocode in Algorithm 1. Calculating exact leverage scores takes time $O(nd^2)$, faster techniques are available to calculate upper bounds e.g. (Mahoney et al., 2012). Next we describe the theoretical guarantees given by the coreset obtained using Algorithm 1.

3.2.1 Analysis of Algorithm

In this section we describe the theoretical guarantees obtained for the coreset using Algorithm 1. We give the following Lemma to relate the $\mathcal{F}$ constraint for each group in the coreset to that in the original data.

Lemma 3. *If we fix* $r_2 = \frac{1+\epsilon}{\epsilon^2}$ *and* $t_i = (d\log(2/\eta) + \log\ell + \log 1/\eta + \log 1/\delta)/|\mathbf{A}_k|$ *when* $\mathbf{a}_i$ *belongs to the* k^{th} *group, then with probability at least* $1-\delta$ *we have the following for each group* k, $\forall \mathbf{x}$ *and* $\forall z \in \mathbf{Z}$,

$$|\mathcal{F}\ (\mathbf{A}_k) - \mathcal{F}\ (\mathbf{C}_k)| \leq \frac{2\epsilon}{1-\epsilon} \tag{1}$$

$$|\mathcal{F}\ (\mathbf{A}) - \mathcal{F}\ (\mathbf{C})| \leq \frac{2\epsilon}{1-\epsilon} \tag{2}$$

The proof relies on first proving the statement for fixed $\mathbf{x}$ and then using an ϵ-net argument to get the result for all queries.

We next describe our main Theorem. It uses Lemma 3 to show the additional guarantees for the fair coreset.

Theorem 4. *Using sampling probabilities as* $p_i = \min(q_i, 1)$ *and reweighing the sampled points as* $\frac{1}{p_i}$ *and for* r_2 *and* t_i *values as given in Lemma 3 and*

$r_1 = \frac{\log d}{\epsilon^2}$, we get a set $(\mathbf{C}, \mathbf{b}_c)$ with associated weight $\tilde{w}_i$ for point i that satisfy the following three conditions with probability $1 - \delta$.

1. $\forall \mathbf{x}$, $\sum_{\mathbf{c}_i \in \mathbf{C}} \tilde{w}_i (\mathbf{c}_i^T \mathbf{x} - b_i)^2 \in (1 \pm \epsilon) \|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2^2$.
2. *All feasible solutions $\mathbf{x}_c$ for the dataset $(\mathbf{C}, \mathbf{b}_c, \tilde{\mathbf{w}})$ and appropriate values ζ_k' satisfy, $\forall k$, $\forall z \in \mathbf{Z}$, $|(\mathcal{F}(\mathbf{A}_k) - \mathcal{F}(\mathbf{A}))| \leq \zeta_k + O(\epsilon)$.*
3. *All feasible solutions for $(\mathbf{A}, \mathbf{b}, \{\zeta_k\})$ are also feasible solutions for $(\mathbf{C}, \tilde{\mathbf{w}}, \mathbf{b}_c, \{\zeta_k'\})$.*

The expected size of the coreset is $O\big(\frac{d \log d}{\epsilon^2} + \frac{1}{\epsilon^2}\ell(d \log \frac{2}{\eta} + \log \frac{\ell}{\eta\delta})\big)$. Here η is the step size in the grid.

Time to build the coreset is $O(nd \log d)$ by using 2-approximate leverage scores (Mahoney et al., 2012). The total time taken to run the algorithm is thus $O(nd \log d + C^2)$ where $C = \big(\frac{d \log d}{\epsilon^2} + \frac{1}{\epsilon^2}\ell(d \log \frac{2}{\eta} + \log \frac{\ell}{\eta\delta})\big)$. Here there will be some additional logarithmic dependence on C^2 and ℓ.

Notice the dependence of the coreset size on ℓ and d. This dependence cannot be avoided. We can show that there exists a data set $\mathbf{A} \in \mathbf{R}^{n \times (d+1)}$ having ℓ groups such that a coreset for fair regression with SP constraints that preserves the SP constraint for each of ℓ groups within an ϵ additive error, must be of size at least $(1-\epsilon)\ell d$. We formally show this in the appendix.

Now using the definition of our coreset and the guarantees obtained in Theorem 4, in the following Corollary we show how the solution obtained from the coreset also approximates the cost on the full data.

Corollary 5. *For input data $(\mathbf{A}, \mathbf{b}, \mathbf{G}, \{\zeta_k\})$ there exists a coreset $(\mathbf{C}, \tilde{\mathbf{w}}, \mathbf{b}_c, \{\zeta_k'\})$, where for k, $\zeta_k' = \zeta_k + \frac{4\epsilon}{1-\epsilon}$. Suppose the obtained optimal solution for coreset is $\mathbf{x}_{copt}$ and the optimal solution for full data is $\mathbf{x}_{opt}$, we have that, w.h.p. : 1) $\|\mathbf{A}\mathbf{x}_{copt} - \mathbf{b}\|_2^2 \leq (1 + 3\epsilon)\|\mathbf{A}\mathbf{x}_{opt} - \mathbf{b}\|_2^2$ and 2) $\forall k$ and $\forall z \in \mathbf{Z}$, $|\mathcal{F}_{\mathbf{x}_{copt}, z}(\mathbf{A}_k) - \mathcal{F}_{\mathbf{x}_{copt}, z}(\mathbf{A})| \leq \zeta_k + \frac{8\epsilon}{1-\epsilon}$*

It is not difficult to extend this technique to overlapping groups by considering all possible overlaps. Using Theorem 4 and Lemma 2, we can solve the fair regression problem using our coreset by using the discretization trick and incur only an additional $O(\eta)$ additive error.

Furthermore, if the regression loss function is $\ell(b_i, \mathbf{a}_i^T \mathbf{x}) = |\mathbf{a}_i^T \mathbf{x} - b_i|^p$, the problem becomes ℓ_p-regression problem with statistical parity constraints. Our coreset technique can also be applied to such ℓ_p-regression problems with statistical parity constraints with only slight change in sampling probability. We show this small result more formally in the appendix.

4 INDIVIDUALLY FAIR CLUSTERING

The individual fair clustering problem was first introduced by Jung et al. (2019). Consider a dataset $\mathbf{P}$ with n points in d dimensions, each having weight 1. For each $\mathbf{x} \in \mathbf{P}$, and for some k, we define the n/k-fair radius— $r_{n/k}^P(\mathbf{x})$ — as the distance of $\mathbf{x}$ to its $(n/k)^{th}$ nearest neighbour in $\mathbf{P}$. Here we assume that all the pairwise distances in the dataset are different. A k-clustering using a set of centers $\mathbf{S}$ is called α-fair if for any $\mathbf{x} \in \mathbf{P}$, $d(\mathbf{x}, \mathbf{S}) \leq \alpha r_{n/k}^P(\mathbf{x})$ where $d(\mathbf{x}, \mathbf{S})$ denotes the distance of $\mathbf{x}$ to its nearest neighbour in $\mathbf{S}$. Also note that the cost of a k-clustering is given as $\sum_{\mathbf{x} \in \mathbf{P}} d(\mathbf{x}, \mathbf{S})$. Here we consider the α-fair k-median problem for the Euclidean metric for analysis and then extend the results to other ℓ_p cost functions where the cost is given as $\sum_{\mathbf{x} \in \mathbf{P}} d(\mathbf{x}, \mathbf{S})^p$. Jung et al. (2019) gave an algorithm that gives a feasible clustering solution which does not necessarily optimize the clustering cost. Mahabadi and Vakilian (2020) gave an algorithm for a 7α approximation in terms of fairness, an $O(1)$ approximation for k-median and k-means as clustering costs and $p^{O(p)}$ approximation for optimal cost with general ℓ_p norm. The algorithm's running time is $O(n^4)$ which makes it prohibitive. Vakilian and Yalçıner (2021); Chakrabarty and Negahbani (2021) further improve the algorithm either in terms of the fairness guarantees or the cost but *not the running time*. To remedy this, we first generalize the notion of fair radius to weighted set and use it to define a coreset for individual fair clustering. then we build such a coreset using a combination of importance and uniform sampling.

Let $\mathbf{C}$ be a weighted data set, where the weight of the point $\tilde{\mathbf{x}}_i$ is denoted as $\tilde{w}_i$. We extend the definition of fair radius in the following manner. Let $\tilde{W} = \sum_{i \in \mathbf{C}} \tilde{w}_i$. We denote $r_{\tilde{W}/k}^{\mathbf{C}}(\tilde{\mathbf{x}})$ as the maximum radius of a ball around $\tilde{\mathbf{x}} \in \mathbf{C}$ which contains points from $\mathbf{C}$ whose total weight is **at most** $\tilde{W}/k$. This is the right generalization that preserves the intuition behind the notion of "fair-radius". From now on we will use this definition of fair radius. Notice that for a dataset having all points with weight 1, this definitions of fair radius corresponds exactly to the previous definition.

As mentioned before, we assume that all pairwise distances are distinct. This assumption is required for the generalized definition of fair radius, as the following example demonstrates. Suppose that $\mathbf{x}_2, \ldots, \mathbf{x}_n$ are all at distance exactly r from $\mathbf{x}_1$. This definition of (weighted) fair-radius at $\mathbf{x}_1$ needs to take a supremum instead of a maximum, and returns r, where $\mathbf{B}(\mathbf{x}_1, r)$ contains $> n/k$ points, violating the *at most* n/k condition. Hence we needed the assumption for the con-

sistency of this (weighted) fair-radius definition. Our algorithm does not use this assumption. Also, given any set of points P, the following simple preprocessing (doable in $O(n \log n)$ time) guarantees that P satisfies this assumption without disturbing the neighborhood relations— let ϵ be any value smaller than the minimum pairwise distance in P, and add a Gaussian noise vector $N(0, \frac{\epsilon}{\sqrt{d}} I)$ to each point.

We use $\mathbf{B}^{\mathbf{P}}(\mathbf{x}, r)$ to denote ball of radius r centered around $\mathbf{x}$ in a dataset $\mathbf{P}$. With a little abuse of notation we use $\left|\mathbf{B}^{\mathbf{P}}(\mathbf{x}, r)\right|$ to denote the total weight of points inside the ball $\mathbf{B}^{\mathbf{P}}(\mathbf{x}, r)$. Notice again that for $\mathbf{P}$ having all points with weights 1 it is same as the cardinality of points inside the ball.

Notice that since a coreset has points that are weighted, we need this generalized definition of fair radius. Also similar to the case of fair regression with SP constraints, we require solutions obtained from coreset to be feasible when used with full data and vice versa. Hence the coreset for individual fair clustering must incorporate this additional requirements. As our next important technical contribution, using this modified definition of fair radius, we formally define a coreset for individually fair k-median problem.

4.1 Coreset for Individually Fair clustering

For the individually fair clustering problem with k-median cost, we define a coreset as follows:

Definition 6. *For a set of points $\mathbf{P}$ with weights 1, a tuple $(\mathbf{C}, \tilde{w})$ is called an ϵ-coreset for individually fair clustering if the following properties hold*

1. $\sum_{\tilde{\mathbf{x}}_i \in \mathbf{C}} \tilde{w}_i d(\tilde{\mathbf{x}}_i, \mathbf{S}) \in (1 \pm \epsilon) \sum_{\mathbf{x} \in \mathbf{P}} d(\mathbf{x}, \mathbf{S}), \forall \mathbf{S}$.
2. *Any feasible solution $\mathbf{S}'$ on the coreset $\mathbf{C}$ with for α fair k-clustering with fair radius $r^{\mathbf{C}}_{\tilde{W}/k}(\tilde{\mathbf{x}})$ is also approximately feasible on the full data i.e $\forall \mathbf{x} \in \mathbf{P}$, $d(\mathbf{x}, \mathbf{S}') \leq O(\alpha)\ r^{\mathbf{P}}_{(1+O(\epsilon))n/k}(\mathbf{x})$.*
3. *Any solution $\mathbf{S}$ feasible on the full data for α-fair k-clustering with fair radius $r^{P}_{n/k}(\mathbf{x})$ is also approximately feasible on the coreset i.e. $\forall \tilde{\mathbf{x}} \in \mathbf{C}$, $d(\tilde{\mathbf{x}}, \mathbf{S}) \leq O(\alpha)\ r^{\mathbf{C}}_{\tilde{W}(1+O(\epsilon))/k}(\tilde{\mathbf{x}})$.*

This definition of the coreset allows us to obtain a solution set of centers from the coreset and use them with the full data while preserving the overall cost and fairness approximately. Next we describe an algorithm to construct such a coreset.

4.1.1 Coreset Algorithm and Analysis

The algorithm to construct for individual fair clustering is similar in spirit to the algorithm for fair regression. The high level idea is to again use a combination of sensitivity scores and carefully calibrated term to decide the sampling probabilities. The process is described as a pseudocode in Algorithm 2. Algorithm 2 takes as an input the dataset and error parameter and returns a subset of points along with corresponding weights. Theorem 7 gives main guarantees of the set returned by Algorithm 2.

In: A dataset $\mathbf{P}$ with n data points in d dimension, error parameter ϵ
Out: Coreset $\mathbf{C}$, associated weights $\tilde{w}_i$'s
Set $\mathbf{C} = \emptyset$; $r = (1/\epsilon^2)$ and $t = \frac{ck \log n}{\epsilon^3 n}$ for an appropriate c.
for $i = 1, 2, \ldots n$ **do**
  Set $q_i = r s_i + t$, where, s_i is an upper bound on the sensitivity of i^{th} point.
  With probability $p_i = \min(q_i, 1)$, add i^{th} point to $\mathbf{C}$ with weight $\tilde{w}_i = 1/p_i$
Return $(\mathbf{C}, \tilde{w}_i' s)$.

Algorithm 2: Coreset for Individually Fair Clustering

Theorem 7. *For individually fair clustering with k-median cost, Algorithm 2 returns a coreset as given in definition 6 with probability at least $(1 - 1/n)$. The expected size of the coreset is $O(k^2 \epsilon^{-2} d \log k + (k \log n)/\epsilon^3)$. The time taken to build the coreset is $\tilde{O}(nkd)$.*

The proof of the Theorem 7 relies on the following two Lemmas:

Lemma 8. *The set $\mathbf{C}$ returned by Algorithm 2 ensures the following property: $\forall \mathbf{x} \in \mathbf{P}$, $\exists \tilde{\mathbf{x}} \in \mathbf{C}$ s.t. $d(\mathbf{x}, \tilde{\mathbf{x}}) \leq r^{\mathbf{P}}_{\epsilon n/k}(\mathbf{x})$ with probability at least $1 - 1/n$.*

Lemma 9. *With probability $1 - 1/n$, the set $\mathbf{C}$ returned by Algorithm 2 ensures that for all $\mathbf{x} \in \mathbf{P}$ and for all radii $r \geq r^{\mathbf{P}}_{\epsilon n/k}(\mathbf{x})$:*
$(1 - \epsilon)|\mathbf{B}^{\mathbf{P}}(\mathbf{x}, r)| \leq |\mathbf{B}^{\mathbf{C}}(\mathbf{x}, r)| \leq (1 + \epsilon)|\mathbf{B}^{\mathbf{P}}(\mathbf{x}, r)|$.

Lemma 8 intuitively says that for any point in original dataset, there is a point not very far inside the coreset. Lemma 9 intuitively captures that the generalized fair radius is in some sense preserved by the coreset.

The proof of both the Lemmas and Theorem is in appendix. An obvious result obtained by combining the time to build the coreset with the fact that the coreset size is only logarithmic in n, solving the individual fair clustering approximately with the coreset is much faster and takes time $O(nkd + k^8 d^4 + (k \log n)^4)$. This is much faster than the time taken to solve for full data which is of the order of $n^4 k^4$. Note that the particular time and sample complexity are obtained because of calculating the sensitivities using a particular method. However as the general idea is the one

as given by Langberg and Schulman (2010); Feldman and Langberg (2011), one can use any upper bounds on sensitivity scores and get time and sampling complexity accordingly. The next result describes how the solution obtained from the coreset can also be used with the full data.

Corollary 10. *For coreset* $\mathbf{C}$ *if we solve the individually fair* k*-median with fair radius* $r^{\mathbf{C}}_{(1+3\epsilon)\tilde{W}/k}(\tilde{\mathbf{x}})$ *and obtain optimal solution* $\tilde{\mathbf{S}}_{opt}$ *and suppose the optimal solution for the full data with fair radius* $r^{\mathbf{P}}_{(1+\epsilon)\frac{n}{k}}(\mathbf{x})$ *is* $\mathbf{S}_{opt}$ *then we have that* $1.\forall \mathbf{x} \in \mathbf{P}$, $d(\mathbf{x}, \tilde{\mathbf{S}}_{opt}) \leq O(\alpha) r^{\mathbf{P}}_{(1+O(\epsilon))\frac{n}{k}}(\mathbf{x})$ *and* $2. \sum_{\mathbf{x}\in\mathbf{P}} d(\mathbf{x}, \tilde{\mathbf{S}}_{opt}) \leq (1 + 3\epsilon) \sum_{\mathbf{x}\in\mathbf{P}} d(\mathbf{x}, \mathbf{S}_{opt})$.

Notice that both the fair-radius as well as the clustering cost suffers an approximation factor that depends on the size of the coreset. Our results can also be extended to clusterings with other ℓ_p norm costs. This is more formally as shown in the appendix.

5 EXPERIMENTS

Here we evaluate our coresets on both the problems

5.1 Experiments on Fair regression with Statistical Parity

For evaluating coresets for fair regression with SP we use following datasets used in Agarwal et al. (2019): 1) **Law School Admissions** (*Law-School*): $\in \mathbb{R}^{20,469\times 11}$, protected attribute: Race. 2) **Communities & Crime** (*Communities*): $\in \mathbb{R}^{1994\times 22}$, protected attribute: Race.

For both datasets the prediction task is carried out via *squared loss minimization*. We compare performances of the models trained on the subsample of the dataset provided by following methods: 1) `FairRegCor`: coreset constructed using our method and 2) `Uniform`: coreset constructed by sampling points uniformly at random[1]. We note that uniform sampling is a known to be a strong baseline in real datasets. The constructed coreset is passed as an input data to Algorithm 1 in Agarwal et al. (2019)[2], which returns a model trained on the given input data by a reduction using the least squares (LS) oracle. The performance of the model trained on the complete training data is called as `FullModel`. We evaluate the performances of the models over range of slack values (ζ) and a fixed discretization grid of size 40: $\mathbf{Z} = \{1/40, 2/40, ..., 1\}$. Here for each group $k \in [\ell]$, we set $\zeta_k = \zeta$. For each value of ζ we run three different experiments and the results are presented as an average over these three runs.

[1]Code for the coreset based algorithm can be found here
[2]Code for the original paper can be found here

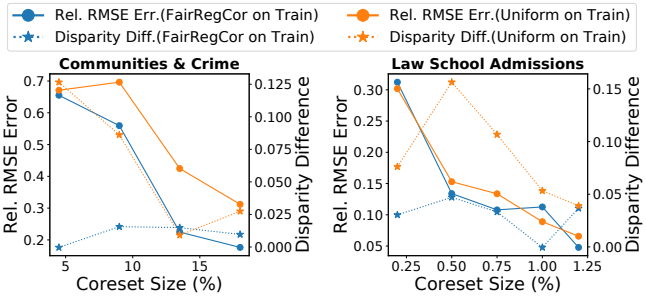

Figure 1: Rel. RMSE Err. & Disparity Diff. on Train data for different coreset sizes ($\zeta = 0.1$ for *Communities & Crime*, and 0.05 for *Law-School*).

Metrics: 1)*Relative RMSE Error*: It is defined as absolute difference of the *RMSE* incurred on `FullModel` with the *RMSE* incurred on the sampling based model divided by the *RMSE* incurred on `FullModel`. Here, *RMSE* is the root mean squared error incurred by the model. 2)*Disparity Difference*: It is defined as the difference of the disparity incurred by the coreset based model (when evaluated on the train/test data) and the disparity incurred by the `FullModel`. Here, disparity is the maximum slack incurred in the constraints over all groups and all z.

Results: First we compare the performances of models on `FairRegCor` and `Uniform` coresets. From Figure 1 we observe that for both *Law-School* and *Communities* datasets `FairRegCor` outperforms `Uniform` in terms of both *RMSE* and *Disparity Difference* for almost all the different sizes of the coreset. For the cases where `Uniform` performs better than `FairRegCor` in terms of *RMSE* error, the difference in the performance is not significant, and `FairRegCor` significantly outperforms `Uniform` with respect to *disparity*. Here we have not compared with leverage score sampling as our sampling probabilities already have leverage score as component and hence on real data we expect similar results.

Experiments on computation time, different parameters, and on test data are in appendix.

5.2 Experiments on Individually Fair Clustering

To evaluate coresets for Individual fair clustering, we used two of the datasets viz. 1) *Diabetes* : $101,765 \times 2$, and 2) *Census*: 48842×5, with same set of features as used by Mahabadi and Vakilian (2020) from the UCI Machine Learning Repository (Dua et al., 2017). It is important to note that that in (Mahabadi and

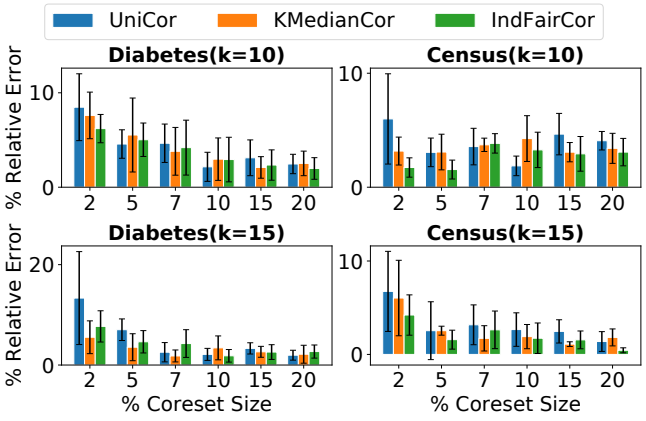

Figure 2: Rel. Err. in Cost (wrt. to Mahabadi and Vakilian (2020) algo.) on Real datasets

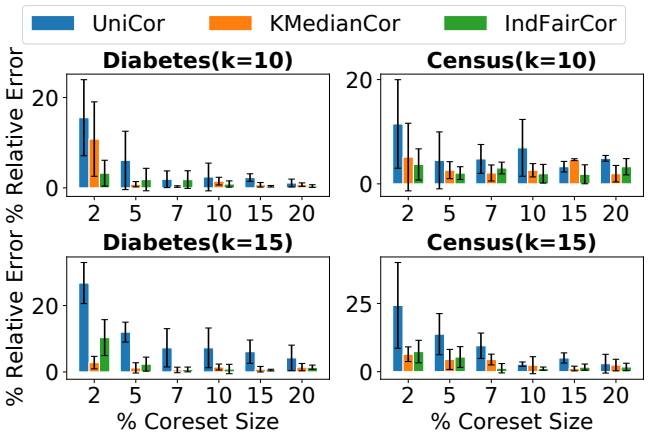

Figure 3: Rel. Err. in Cost (wrt. to Mahabadi and Vakilian (2020) algo.) on Semi-synthetic datasets

Vakilian, 2020), all the experiments are performed on random samples of size 1000 only. For our coreset experiments, we considered a random sample of 10000 points from the *Diabetes* and *Census* datasets. Henceforth, we will refer to these as full datasets for our experiments [3].

Additionally, to emphasize the effect of non-uniform sampling, we also create a semi-synthetic data using the above real datasets. We first sample (2500) points uniformly at random and then use a power law distribution over this set and make copies of sampled points to increase it to of size 10000.

Similar to Mahabadi and Vakilian (2020), after the initial estimate, the local search algorithm is executed with 1-swap. They used two metrics: 1) the k-median cost on the full data and 2) *Max. Fairness (maxF)*, which is the maximum over the distance of each point

[3]Code base for Individual Fairness can be found here

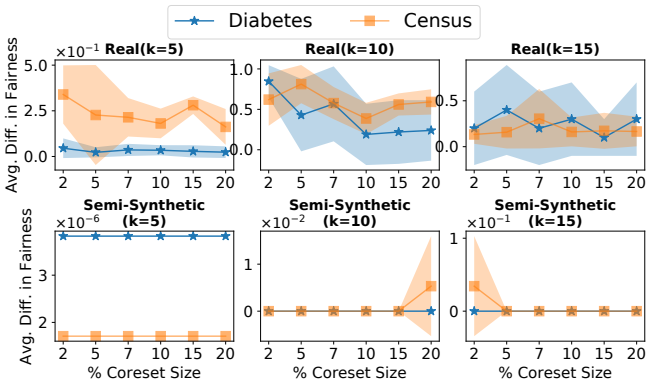

Figure 4: Difference in fairness by Mahabadi and Vakilian (2020) algo. and **IndFair** Coreset algo. on Real and Semi-synthetic datasets

Table 1: Computation Time Comparison (seconds) between IndFair Coreset and Full data

Coreset size in %	Full	2%	5%
Diabetes (k=10)	412.36	0.156	0.865
Diabetes (k=15)	930.57	0.321	2.127
Census (k=10)	506.81	0.31	1.859
Census (k=15)	1489.31	0.507	3.064

to its nearest center.

For our coresets experiments, we created coresets using three methods and appropriately reweighed the points. The three methods are: 1) **UNI**: uniform random sample 2) **k-Median** : k-median coreset using Bachem et al. (2018b) technique and 3) **IndFair**: our Individual fairness coreset algorithm. We modified the code by Mahabadi and Vakilian (2020) to accommodate weighted datasets and experiment with different coreset sizes and k values. For each method and each coreset size we ran 5 experiments and our results are the average of these five runs.

Cost Evaluation: We obtained a set of k-centers by running the algorithm on the coresets and then calculated the total cost on the full data using these obtained centers. Lets call the original cost on full data $C1$ and the cost obtained using centers from the coreset be $C2$. We report the metric: Percentage Relative Error: $(|C1 - C2| * 100)/C1$. Figures 2 and 3 show the percentage relative error for different coreset sizes for all coreset methods and for all datasets for the values of $k = 10, 15$ for individually fair clustering with k-median cost. As can be observed from the figure, our method achieves better or at par errors compared to other sampling strategies on both real and synthetic data.

Fairness Evaluation: To see the effect of coreset on fairness we compare the difference of *maxF* value obtained using centers from our coreset and the centers from the full data. Figure 4 shows the evaluation of fairness. The differences are very small and we observe that our method achieves comparable fairness.

Computation Time: The comparison of the computation time on our coreset to the one on full real data is shown in Table 1 in seconds. The column tagged "Full" shows the computation time on the full data. The time to create the coresets for most datasets and different values of k was negligible as compared to the total computation time and is thus ignored. Of course uniform coresets were created fastest. Computation time on all coreset methods are comparable and so we have reported only the time for our coresets. We see that our coresets have a huge advantage in terms of computation time over the full data. This can especially be useful in practice.

6 CONCLUSION

In this paper we define and construct coresets for Fair Regression with Statistical Parity and for Individually Fair Clustering problems. It is interesting to note that though the nature of both the problems is different viz, a supervised and an unsupervised learning problem and the nature of fairness constraints are also different, at a high level the algorithms to solve them have certain similarities. In both cases, algorithms use combination of importance and uniform sampling to satisfy the fairness constraints along with approximating the cost. This may give insights for designing coresets for ML problems with other fairness notions also. Though our coreset construction algorithms are simple and rely on existing coreset construction strategies, the technical novelty lies in formally defining coresets for problems with additional fairness constraints and then showing that such coresets with theoretical guarantees can actually be constructed using these simple strategies. Empirically the coresets show good accuracy and also preserve fairness. Specifically for Individually Fair Clustering they give very significant time benefits. Defining and designing such coresets for other fairness notions is an interesting question.

Acknowledgements

Supratim acknowledges the generous funding from the European Union's Horizon 2020 research and innovation programmed under grant agreement No 682203 -ERC-[Inf-Speed-Tradeoff]. Jayesh acknowledges the funding received from the Engineering and Physical Sciences Research Council, UK (EPSRC) under Grant Ref: EP/S03353X/1. Anirban acknowleges the kind support of the N. Rama Rao Chair Professor position, the Google India AI/ML award, and the CISCO University Award.

References

Abbasi, M., Bhaskara, A., and Venkatasubramanian, S. (2021). Fair clustering via equitable group representations. In *ACM FAccT*.

Ackermann, M. R. and Blömer, J. (2009). Coresets and approximate clustering for bregman divergences. In *Proceedings of the twentieth annual ACM-SIAM symposium on Discrete algorithms*, pages 1088–1097. SIAM.

Agarwal, A., Beygelzimer, A., Dudík, M., Langford, J., and Wallach, H. (2018). A reductions approach to fair classification. In *International Conference on Machine Learning*, pages 60–69. PMLR.

Agarwal, A., Dudík, M., and Wu, Z. S. (2019). Fair regression: Quantitative definitions and reduction-based algorithms. In *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*.

Agarwal, P. K., Har-Peled, S., and Varadarajan, K. R. (2004). Approximating extent measures of points. *Journal of the ACM (JACM)*, 51(4):606–635.

Agarwal, P. K., Har-Peled, S., Varadarajan, K. R., et al. (2005). Geometric approximation via coresets. *Combinatorial and computational geometry*, 52:1–30.

Alabi, D., Immorlica, N., and Kalai, A. (2018). Unleashing linear optimizers for group-fair learning and optimization. In *COLT*, pages 2043–2066. PMLR.

Avron, H., Clarkson, K. L., and Woodruff, D. P. (2017). Sharper bounds for regularized data fitting. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques (APPROX/RANDOM 2017)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik.

Bachem, O., Lucic, M., and Krause, A. (2015). Coresets for nonparametric estimation-the case of dp-means. In *ICML*, pages 209–217.

Bachem, O., Lucic, M., and Krause, A. (2017). Practical coreset constructions for machine learning. *stat*, 1050:4.

Bachem, O., Lucic, M., and Krause, A. (2018a). Scalable k-means clustering via lightweight coresets. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 1119–1127.

Bachem, O., Lucic, M., and Lattanzi, S. (2018b). One-shot coresets: The case of k-clustering. In *International conference on artificial intelligence and statistics*, pages 784–792.

Backurs, A., Indyk, P., Onak, K., Schieber, B., Vakilian, A., and Wagner, T. (2019). Scalable fair clustering. In *ICML*, pages 405–413. PMLR.

Balcan, M.-F. F., Ehrlich, S., and Liang, Y. (2013). Distributed k-means and k-median clustering on general topologies. In *Advances in Neural Information Processing Systems*, pages 1995–2003.

Bandyapadhyay, S., Fomin, F. V., and Simonov, K. (2020). On coresets for fair clustering in metric and euclidean spaces and their applications. *preprint arXiv:2007.10137*.

Barger, A. and Feldman, D. (2020). Deterministic coresets for k-means of big sparse data. *Algorithms*, 13(4):92.

Barocas, S. and Selbst, A. D. (2016). Big data's disparate impact. *Calif. L. Rev.*, 104:671.

Bercea, I. O., Groß, M., Khuller, S., Kumar, A., Rösner, C., Schmidt, D. R., and Schmidt, M. (2019). On the cost of essentially fair clusterings. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques (APPROX/RANDOM 2019)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik.

Berk, R., Heidari, H., Jabbari, S., Joseph, M., Kearns, M., Morgenstern, J., Neel, S., and Roth, A. (2017). A convex framework for fair regression. *preprint arXiv:1706.02409*.

Berk, R., Heidari, H., Jabbari, S., Kearns, M., and Roth, A. (2018). Fairness in criminal justice risk assessments: The state of the art. *Sociological Methods & Research*, page 0049124118782533.

Bhojanapalli, S. and Sanghavi, S. (2015). A new sampling technique for tensors. *stat*, 1050:19.

Boutsidis, C. and Magdon-Ismail, M. (2013). Deterministic feature selection for k-means clustering. *IEEE Transactions on Information Theory*, 59(9):6099–6110.

Braverman, V., Feldman, D., and Lang, H. (2016). New frameworks for offline and streaming coreset constructions. *arXiv preprint arXiv:1612.00889*.

Calders, T., Karim, A., Kamiran, F., Ali, W., and Zhang, X. (2013). Controlling attribute effect in linear regression. In *2013 IEEE 13th international conference on data mining*, pages 71–80. IEEE.

Caton, S. and Haas, C. (2020). Fairness in machine learning: A survey. *arXiv preprint arXiv:2010.04053*.

Chakrabarty, D. and Negahbani, M. (2021). Better algorithms for individually fair k-clustering. *arXiv preprint arXiv:2106.12150*.

Chhaya, R., Choudhari, J., Dasgupta, A., and Shit, S. (2020a). Online coresets for clustering with bregman divergences. *arXiv preprint arXiv:2012.06522*.

Chhaya, R., Choudhari, J., Dasgupta, A., and Shit, S. (2020b). Streaming coresets for symmetric tensor factorization. In *International Conference on Machine Learning*, pages 1855–1865. PMLR.

Chhaya, R., Dasgupta, A., and Shit, S. (2020c). On coresets for regularized regression. In *International conference on machine learning*, pages 1866–1876. PMLR.

Chierichetti, F., Kumar, R., Lattanzi, S., and Vassilvitskii, S. (2017). Fair clustering through fairlets. In *Neurips*.

Chzhen, E., Denis, C., Hebiri, M., Oneto, L., and Pontil, M. (2020). Fair regression with wasserstein barycenters. *Neurips*.

Cohen, M. B., Elder, S., Musco, C., Musco, C., and Persu, M. (2015a). Dimensionality reduction for k-means clustering and low rank approximation. In *Proceedings of the forty-seventh annual ACM symposium on Theory of computing*, pages 163–172.

Cohen, M. B., Lee, Y. T., Musco, C., Musco, C., Peng, R., and Sidford, A. (2015b). Uniform sampling for matrix approximation. In *Proceedings of the 2015 Conference on Innovations in Theoretical Computer Science*, pages 181–190. ACM.

Cohen, M. B., Musco, C., and Musco, C. (2017). Input sparsity time low-rank approximation via ridge leverage score sampling. In *Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1758–1777. SIAM.

Cohen, M. B. and Peng, R. (2015). L p row sampling by lewis weights. In *Proceedings of the forty-seventh annual ACM symposium on Theory of computing*, pages 183–192. ACM.

Corbett-Davies, S. and Goel, S. (2018). The measure and mismeasure of fairness: A critical review of fair machine learning. *arXiv preprint arXiv:1808.00023*.

Corbett-Davies, S., Pierson, E., Feller, A., Goel, S., and Huq, A. (2017). Algorithmic decision making and the cost of fairness. In *Proceedings of the 23rd acm sigkdd international conference on knowledge discovery and data mining*, pages 797–806.

Curtin, R., Im, S., Moseley, B., Pruhs, K., and Samadian, A. (2019). On coresets for regularized loss minimization. *arXiv preprint arXiv:1905.10845*.

Dan, C., Wang, H., Zhang, H., Zhou, Y., and Ravikumar, P. K. (2019). Optimal analysis of subset-selection based l_p low-rank approximation. In *Advances in Neural Information Processing Systems*, pages 2537–2548.

Dasgupta, A., Drineas, P., Harb, B., Kumar, R., and Mahoney, M. W. (2009). Sampling algorithms and coresets for \ell_p regression. *SIAM Journal on Computing*, 38(5):2060–2078.

Dickens, C., Cormode, G., and Woodruff, D. (2018). Leveraging well-conditioned bases: Streaming and distributed summaries in minkowski p-norms. In *International Conference on Machine Learning*, pages 1243–1251.

Drineas, P., Magdon-Ismail, M., Mahoney, M. W., and Woodruff, D. P. (2012). Fast approximation of matrix coherence and statistical leverage. *Journal of Machine Learning Research*, 13(Dec):3475–3506.

Drineas, P., Mahoney, M. W., and Muthukrishnan, S. (2006). Sampling algorithms for ℓ_2 regression and applications. In *Proceedings of the seventeenth annual ACM-SIAM symposium on Discrete algorithm*, pages 1127–1136. Society for Industrial and Applied Mathematics.

Drineas, P., Mahoney, M. W., and Muthukrishnan, S. (2008). Relative-error cur matrix decompositions. *SIAM Journal on Matrix Analysis and Applications*, 30(2):844–881.

Drineas, P., Mahoney, M. W., Muthukrishnan, S., and Sarlós, T. (2011). Faster least squares approximation. *Numerische mathematik*, 117(2):219–249.

Dua, D., Graff, C., et al. (2017). Uci machine learning repository.

Dubhashi, D. P. and Panconesi, A. (2009). *Concentration of measure for the analysis of randomized algorithms*. Cambridge University Press.

Feldman, D. (2020). Core-sets: Updated survey. *Sampling Techniques for Supervised or Unsupervised Tasks*, pages 23–44.

Feldman, D. and Langberg, M. (2011). A unified framework for approximating and clustering data. In *Proceedings of the forty-third annual ACM symposium on Theory of computing*, pages 569–578. ACM.

Feldman, D., Schmidt, M., and Sohler, C. (2020). Turning big data into tiny data: Constant-size coresets for k-means, pca, and projective clustering. *SIAM Journal on Computing*, 49(3):601–657.

Feldman, D., Volkov, M., and Rus, D. (2016). Dimensionality reduction of massive sparse datasets using coresets. In *Advances in Neural Information Processing Systems*, pages 2766–2774.

Fitzsimons, J., Al Ali, A., Osborne, M., and Roberts, S. (2019). A general framework for fair regression. *Entropy*, 21(8):741.

Fukuchi, K., Kamishima, T., and Sakuma, J. (2015). Prediction with model-based neutrality. *IEICE TRANSACTIONS on Information and Systems*, 98(8):1503–1516.

Ghadiri, M., Samadi, S., and Vempala, S. (2021). Socially fair k-means clustering. In *ACM FAccT*.

Golub, G. H. and Van Loan, C. F. (1996). *Matrix Computations*. The Johns Hopkins University Press, third edition.

Har-Peled, S. and Mazumdar, S. (2004). On coresets for k-means and k-median clustering. In *Proceedings of the thirty-sixth annual ACM symposium on Theory of computing*, pages 291–300. ACM.

Hardt, M., Price, E., and Srebro, N. (2016). Equality of opportunity in supervised learning. In *Neurips*.

Haussler, D. (1995). Sphere packing numbers for subsets of the boolean n-cube with bounded vapnik-chervonenkis dimension. *J. Comb. Theory, Ser. A*, 69(2):217–232.

Haussler, D. and Welzl, E. (1987). ϵ-nets and simplex range queries. *Discrete & Computational Geometry*, 2(2):127–151.

Huang, L., Jiang, S. H.-C., and Vishnoi, N. K. (2019). Coresets for clustering with fairness constraints. In *NeurIPS*.

Huang, L. and Vishnoi, N. K. (2020). Coresets for clustering in euclidean spaces: Importance sampling is nearly optimal.

Johnson, K. D., Foster, D. P., and Stine, R. A. (2016). Impartial predictive modeling: Ensuring fairness in arbitrary models. *Statistical Science*, page 1.

Jung, C., Kannan, S., and Lutz, N. (2019). A center in your neighborhood: Fairness in facility location. *arXiv preprint arXiv:1908.09041*.

Kamishima, T., Akaho, S., Asoh, H., and Sakuma, J. (2012). Fairness-aware classifier with prejudice remover regularizer. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 35–50. Springer.

Kleinberg, J., Mullainathan, S., and Raghavan, M. (2017). Inherent trade-offs in the fair determination of risk scores. In *8th Innovations in Theoretical Computer Science Conference (ITCS 2017)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik.

Kleindessner, M., Awasthi, P., and Morgenstern, J. (2020). A notion of individual fairness for clustering. *arXiv preprint arXiv:2006.04960*.

Komiyama, J., Takeda, A., Honda, J., and Shimao, H. (2018). Nonconvex optimization for regression with fairness constraints. In *ICML*.

Langberg, M. and Schulman, L. J. (2010). Universal ε-approximators for integrals. In *Proceedings of the twenty-first annual ACM-SIAM symposium on Discrete Algorithms*, pages 598–607. SIAM.

Lucic, M., Bachem, O., and Krause, A. (2016). Strong coresets for hard and soft bregman clustering with applications to exponential family mixtures. In *Artificial intelligence and statistics*, pages 1–9.

Maalouf, A., Jubran, I., and Feldman, D. (2019). Fast and accurate least-mean-squares solvers. In *Advances in Neural Information Processing Systems*, pages 8305–8316.

Mahabadi, S. and Vakilian, A. (2020). Individual fairness for k-clustering. In *International Conference on Machine Learning*, pages 6586–6596. PMLR.

Mahoney, M. W., Drineas, P., Magdon-Ismail, M., and Woodruff, D. P. (2012). Fast approximation of matrix coherence and statistical leverage. In *ICML*.

Makarychev, Y. and Vakilian, A. (2021). Approximation algorithms for socially fair clustering. In *Conference on Learning Theory*, pages 3246–3264. PMLR.

Mehrabi, N., Morstatter, F., Saxena, N., Lerman, K., and Galstyan, A. (2021). A survey on bias and fairness in machine learning. *ACM Computing Surveys (CSUR)*, 54(6):1–35.

Meng, X. and Mahoney, M. W. (2013). Low-distortion subspace embeddings in input-sparsity time and applications to robust linear regression. In *Proceedings of the forty-fifth annual ACM symposium on Theory of computing*, pages 91–100. ACM.

Pérez-Suay, A., Laparra, V., Mateo-García, G., Mu noz-Marí, J., Gómez-Chova, L., and Camps-Valls, G. (2017). Fair kernel learning. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 339–355. Springer.

Phillips, J. M. (2017). Coresets and sketches. In *Handbook of discrete and computational geometry*, pages 1269–1288. Chapman and Hall/CRC.

Pilanci, M. and Wainwright, M. J. (2015). Randomized sketches of convex programs with sharp guarantees. *IEEE Transactions on Information Theory*, 61(9):5096–5115.

Raj, A., Musco, C., and Mackey, L. (2020). Importance sampling via local sensitivity. In *International Conference on Artificial Intelligence and Statistics*, pages 3099–3109. PMLR.

Reddi, S. J., Póczos, B., and Smola, A. J. (2015). Communication efficient coresets for empirical loss minimization. In *UAI*, pages 752–761.

Redmond, M. and Baveja, A. (2002). A data-driven software tool for enabling cooperative information sharing among police departments. *European Journal of Operational Research*, 141(3):660–678.

Samadi, S., Tantipongpipat, U., Morgenstern, J. H., Singh, M., and Vempala, S. (2018). The price of fair pca: One extra dimension. *Advances in neural information processing systems*, 31.

Schmidt, M., Schwiegelshohn, C., and Sohler, C. (2019). Fair coresets and streaming algorithms for fair k-means. In *WAOA*.

Sharifi-Malvajerdi, S., Kearns, M., and Roth, A. (2019). Average individual fairness: Algorithms, generalization and experiments. *Advances in Neural Information Processing Systems*, 32:8242–8251.

Shit, S. (2021). ℓ_p subspace embedding in input sparsity time. In *8th ACM IKDD CODS and 26th COMAD*, pages 418–418.

Sohler, C. and Woodruff, D. P. (2011). Subspace embeddings for the ℓ_1-norm with applications. In *Proceedings of the forty-third annual ACM symposium on Theory of computing*, pages 755–764. ACM.

Tüfekci, P. (2014). Prediction of full load electrical power output of a base load operated combined cycle power plant using machine learning methods. *International Journal of Electrical Power & Energy Systems*, 60:126–140.

Vakilian, A. and Yalçıner, M. (2021). Improved approximation algorithms for individually fair clustering. *arXiv preprint arXiv:2106.14043*.

Wang, L., Gordon, M. D., and Zhu, J. (2006). Regularized least absolute deviations regression and an efficient algorithm for parameter tuning. In *Sixth International Conference on Data Mining (ICDM'06)*, pages 690–700. IEEE.

Wightman, L. (1998). Lsac national longitudinal bar passage study.(1998). *Newtown, PA: The Law School Admission Council*.

Woodruff, D. and Zhang, Q. (2013). Subspace embeddings and \ell_p-regression using exponential random variables. In *Conference on Learning Theory*, pages 546–567.

Woodruff, D. P. et al. (2014). Sketching as a tool for numerical linear algebra. *Foundations and Trends® in Theoretical Computer Science*, 10(1–2):1–157.

Yeh, I.-C. and Lien, C.-h. (2009). The comparisons of data mining techniques for the predictive accuracy of probability of default of credit card clients. *Expert Systems with Applications*, 36(2):2473–2480.

Zink, A. and Rose, S. (2020). Fair regression for health care spending. *Biometrics*, 76(3):973–982.

Supplementary Material: On Coresets for Fair Regression and Individually Fair Clustering

A APPENDIX

Here we present the proofs of all theoretical results in the paper as well some additional experiments and details. We will need Bernstein's inequality (Dubhashi and Panconesi, 2009)

Theorem 11. *(Bernstein's Inequality) Let the scalar random variables $x_1, x_2, \dots, x_n$ be independent that satisfy $\forall i \in [n]$, $|x_i - \mathbb{E}[x_i]| \leq b$. Let $X = \sum_{i=1}^{n} x_i$ and let $\sigma^2 = \sum_{i=1}^{n} \sigma_i^2$ be the variance of X. Then for any $m > 0$,*

$$Pr\big(X > \mathbb{E}[X] + m\big) \leq \exp\left(\frac{-m^2}{2\sigma^2 + bm/3}\right)$$

B PROOFS FOR FAIR REGRESSION WITH STATISTICAL PARITY

B.1 Proof of Lemma 2

Proof.

$$\begin{aligned}
|\ell_\eta(\mathbf{Ax}) - \ell_\eta(\mathbf{Cx})| &\leq |\ell_\eta(\mathbf{Ax}) - \|\mathbf{Ax}\|_2^2| + |\|\mathbf{Ax}\|_2^2 - \|\mathbf{Cx}\|_2^2| + |\|\mathbf{Cx}\|_2^2 - \ell_\eta(\mathbf{Cx})| \\
&\leq n\eta + \epsilon\|\mathbf{Ax}\|_2^2 + |\mathbf{C}|\eta \\
&\leq 2n\eta + \epsilon((n\eta) + \ell_\eta(\mathbf{Ax})) \\
&= (2+\epsilon)n\eta + \epsilon\ell_\eta(\mathbf{Ax})
\end{aligned}$$

Dividing both sides by n gives the result. □

B.2 Proof of Lemma 3

Proof. We show that $|\mathcal{F}(\mathbf{A}_k) - \mathcal{F}(\mathbf{C}_k)|$ is small. The proof for $|\mathcal{F}(\mathbf{A}) - \mathcal{F}(\mathbf{C})|$ being small is similar.

Note that, for all i, $\mathbb{E}[\tilde{w}_i] = 1$, and $|\tilde{w}_i| \leq \frac{1}{p_i} = \frac{1}{(r_1\|\mathbf{u}_i\|_2^2 + r_2 t_i)} \leq \frac{1}{r_2 t_i}$. For group k, $\mathrm{Var}(\sum_{i\in[|\mathbf{A}_k|]}(\tilde{w}_i)) \leq \mathbb{E}[\sum_{i\in[|\mathbf{A}_k|]}(\tilde{w}_i^2)] = \sum_{i\in[|\mathbf{A}_k|]}(\frac{1}{(r_1\|\mathbf{u}_i\|_2^2 + r_2 t_i)}) \leq \frac{|\mathbf{A}_k|}{r_2 t_i}$.

Applying Bernstein's inequality to the sum of $\tilde{w}_i$ for a fixed group $\mathbf{A}_k$ we get

$$\Pr(|\sum_{i\in[|\mathbf{A}_k|]} \tilde{w}_i - |\mathbf{A}_k|| \geq \epsilon|\mathbf{A}_k|) \leq \exp\left(\frac{-\epsilon^2|\mathbf{A}_k| r_2 t_i}{\epsilon + 1}\right)$$

We plug the given values of r_2 and t_i to get the above probability to be at most $\frac{\delta\eta}{\ell}\exp(-d\log(2/\eta))$.

Next for the numerator we fix a group, an $\mathbf{x}$ and a z. Applying similar analysis on the random variable $\tilde{w}_i \cdot \mathbb{1}_i$, we get an additive error of $\epsilon|\mathbf{A}_k|$ for a group k.

Now we show how to prove the statement for all groups, all z and all $\mathbf{x}$. Since we have discretized grid $\mathbf{Z}$ and finite number of groups ℓ, we can take a union bound over all $z \in \mathbf{Z}$ and ℓ. To show the statement for all $\mathbf{x}$, consider the set $\mathbf{Y} = \{\mathbf{y} \in \mathbb{R}^n | \mathbf{y} = \mathbb{1}(\mathbf{Ax}) \text{ for some } \mathbf{x} \in \mathbb{R}^d\}$. Here $\mathbf{y} = \mathbb{1}(\mathbf{Ax})$ is an indicator vector obtained by thresholding each $\mathbf{a}_i^T\mathbf{x}$ for some fixed z. Notice that for a fixed z, $\mathbb{1}(\mathbf{Ax})$ is just a thresholding classifier with dimension d and hence by a standard $\epsilon-$net argument with net of size $(2/\epsilon)^d$ suffices. Taking a union bound

over the ϵ-net along with all groups and z, we finally get for all $\mathbf{x} \in \mathbb{R}^d$, for all $z \in \mathbf{Z}$, and for all groups, we finally get an additive error of $\epsilon|\mathbf{A}_k|$ in the numerator with probability atleast $1-\delta$

Now using these results we can show bound $|(\mathcal{F}_{\mathbf{x},z}(\mathbf{A}_k) - \mathcal{F}_{\mathbf{x},z}(\mathbf{C}_k))| \leq \frac{2\epsilon}{1-\epsilon}$, $\forall k$. This is shown below. Hence $\forall k, \forall z \in \mathbf{Z}$ and $\forall \mathbf{x} \in \mathbb{R}^d$, we get equation 1. With similar analysis over the entire dataset $\mathbf{A}$, we get equation 2

We want to bound $|(\mathcal{F}_{\mathbf{x},z}(\mathbf{A}_k) - \mathcal{F}_{\mathbf{x},z}(\mathbf{C}_k))|$. Now $(\mathcal{F}_{\mathbf{x},z}(\mathbf{A}_k) - \mathcal{F}_{\mathbf{x},z}(\mathbf{C}_k))$ is given as

$$
\begin{aligned}
&\frac{\sum_{i\in[|\mathbf{A}_k|]} w_i \mathbb{1}_i}{\sum_{i\in[|\mathbf{A}_k|]} w_i} - \frac{\sum_{i\in[|\mathbf{A}_k|]} \tilde{w}_i \mathbb{1}_i}{\sum_{i\in[|\mathbf{A}_k|]} \tilde{w}_i} \\
&\leq \frac{(1+\epsilon)\sum_{i\in[|\mathbf{A}_k|]} w_i \mathbb{1}_i - \sum_{i\in[|\mathbf{A}_k|]} \tilde{w}_i \mathbb{1}_i}{(1-\epsilon)\sum_{i\in[|\mathbf{A}_k|]} w_i} \\
&\leq \frac{\epsilon \sum_{i\in[|\mathbf{A}_k|]} w_i + \epsilon \sum_{i\in[|\mathbf{A}_k|]} w_i \mathbb{1}_i}{(1-\epsilon)\sum_{i\in[|\mathbf{A}_k|]} w_i} \\
&\leq \frac{2\epsilon \sum_{i\in[|\mathbf{A}_k|]} w_i}{(1-\epsilon)\sum_{i\in[|\mathbf{A}_k|]} w_i} \\
&= \frac{2\epsilon}{1-\epsilon}
\end{aligned}
$$

This gives us the required result. □

B.3 Proof of Theorem 4

Proof. As each p_i satisfies $p_i \geq r_1 \|\mathbf{u}_i\|_2^2$ with $r_1 \geq \frac{\log d}{\epsilon^2}$, by employing the result of Drineas et al. (2006) we can show that, with high probability, the sampled set $\mathbf{C}$ and corresponding $\mathbf{b}_c$ satisfy the following property for all $\mathbf{x} \in \mathbb{R}^d$:

$$\sum_{i\in[|\mathbf{C}|]} \tilde{w}_i (\mathbf{c}_i^T \mathbf{x} - b_i)^2 \in (1 \pm \epsilon)\|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2^2 \tag{3}$$

Hence we have condition 1.

Now since we solve the fair regression over the coreset $\mathbf{C}$, any feasible solution $\mathbf{x}_c$ satisfies the following, $\forall k, \forall z \in \mathbf{Z}$,

$$|(\mathcal{F}\ (\mathbf{C}_k) - \mathcal{F}\ (\mathbf{C}))| \leq \zeta_k'. \tag{4}$$

Now using equation (4) in equation (1) we get for all $(k, z \in \mathbf{Z})$,

$$|(\mathcal{F}\ (\mathbf{A}_k) - \mathcal{F}\ (\mathbf{C}))| \leq \frac{2\epsilon}{1-\epsilon} + \zeta_k'.$$

Similarly using equation (2) in the above equation, we get, again for all (k, z),

$$|(\mathcal{F}\ (\mathbf{A}_k) - \mathcal{F}\ (\mathbf{A}))| \leq \frac{4\epsilon}{1-\epsilon} + \zeta_k'.$$

Setting $\zeta_k' = \zeta_k + \frac{4\epsilon}{(1-\epsilon)}$ we get

$$|(\mathcal{F}\ (\mathbf{A}_k) - \mathcal{F}\ (\mathbf{A}))| \leq \frac{8\epsilon}{1-\epsilon} + \zeta_k.$$

and hence gives the property 2. Also since each ζ_k' has an additional slack of $\frac{4\epsilon}{1-\epsilon}$, all feasible solutions for original problem are also feasible for the coreset giving property 3. Finally the expected size of the coreset is $r_1 \sum_{i\in[n]} (\|\mathbf{u}_i\|_2^2) + r_2 \sum_{i\in[n]} t_i$. Putting $r_1 = O(\frac{\log d}{\epsilon^2})$ and r_2 and t_i values as in Lemma 3 and summing over all groups, we get the result. □

B.4 Lower Bound for fair Regression with SP Constraints

There exists a dataset $\mathbf{A} \in \mathbf{R}^{n\times(d+1)}$ and having ℓ unique protected groups such that a coreset for fair regression with SP constraints that preserves the statistical parity constraint for each of ℓ groups, within an ϵ additive error must be of size at least $(1-\epsilon)\ell d$. Below is the construction.

Let $\mathbf{A}$ be a dataset with n points in $\mathbf{R}^{d+1}$ which are partitioned into ℓ groups. Let a group k has $|\mathbf{A}_k| = n_k$ points. Let the first index of every point identify the group, i.e., the first index value of every point in $\mathbf{A}_k$ is k. Let $\mathbf{A}_k$ has d unique points, e.g., $[k, 1, 0, 0, \ldots, 0], [k, 0, 1, 0, \ldots, 0]$,
$\ldots [k, 0, 0, 0, \ldots, 1]$. In $\mathbf{A}_k$ let there be $\frac{n_k}{d}$ copies of every unique point. Similarly for every other $j \in [\ell]$ we have $\frac{n_j}{d}$ copies of every unique point $[j, 1, 0, 0, \ldots, 0], [j, 0, 1, 0, \ldots, 0]$,
$\ldots [j, 0, 0, 0, \ldots, 1]$ in $\mathbf{A}_j$. For all $k \in [\ell]$ as $\mathbf{A}_k$ has only d linearly independent points, hence its rank is d. So the rank of $\mathbf{A}$ is also d. Suppose we get a coreset with at most $(1-\epsilon)d$ unique points. Then we choose a unit vector query $\mathbf{x}$ from the subspace spanned by ϵd unique points but orthogonal to $(1-\epsilon)d$ unique points in the coreset. Let $z = \gamma$, such that $\forall \mathbf{a} \in \mathbf{A}$ which is not perpendicular to ϵd unique points, we have $\mathbf{a}^T\mathbf{x} \geq \gamma$. Now let the missing ϵd unique points be from $\mathbf{A}_k$. Then notice that $\sum_{i\in[|\mathbf{A}_k|]} \tilde{w}_i \mathbb{1}_i = 0$ where as $\sum_{i\in[|\mathbf{A}_k|]} w_i \mathbb{1}_i \geq \epsilon d \frac{n_k}{d} = \epsilon n_k$. So for the group $\mathbf{A}_k$ we have $|\sum_{i\in[|\mathbf{A}_k|]} w_i \mathbb{1}_i - \sum_{i\in[|\mathbf{A}_k|]} \tilde{w}_i \mathbb{1}_i|$ is at least $\epsilon \cdot n_k$. So to bound the difference to be at most ϵn_k, the coreset must have at least $(1-\epsilon)d$ unique points. Now to ensure the same for all $k \in [\ell]$ group, the size of the coreset must be at least $(1-\epsilon)\ell d$

B.5 Proof of Corollary 5

Proof. From the proof of theorem 4, we know that $\mathbf{x}_{copt}$ satisfies property 2 in the above corollary. Now we use proof similar to one provided by Bachem et al. (2017) to prove property 1. By definition of coreset we have $\|\mathbf{C}\mathbf{x}_{copt} - \mathbf{b}_c\|_2^2 \leq (1+\epsilon)\|\mathbf{A}\mathbf{x}_{copt} - \mathbf{b}\|_2^2$.

$$
\begin{aligned}
\|\mathbf{A}\mathbf{x}_{copt} - \mathbf{b}_c\|_2^2 &\leq \frac{1}{(1-\epsilon)}\|\mathbf{C}\mathbf{x}_{copt} - \mathbf{b}_c\|_2^2 \\
&\overset{(i)}{\leq} \frac{1}{(1-\epsilon)}\|\mathbf{C}\mathbf{x}_{opt} - \mathbf{b}_c\|_2^2 \\
&\leq \frac{1+\epsilon}{(1-\epsilon)}\|\mathbf{A}\mathbf{x}_{opt} - \mathbf{b}\|_2^2 \leq (1+3\epsilon)\|\mathbf{A}\mathbf{x}_{opt} - \mathbf{b}\|_2^2
\end{aligned}
$$

For (i), notice that due to property 3 in theorem 4, we ensure that the $\mathbf{x}_{opt}$ is also a feasible solution for the problem with the coreset. This in turn brings the additive error of $O(\epsilon)$ in the constraints in property 2. □

Next we present the formal result that extends the results for fair ℓ_p - regression for $p \neq 2$.

Corollary 12. *Let $\hat{\mathbf{U}}$ be an (f, g, p) well-conditioned basis of $[\mathbf{A}, -\mathbf{b}]$, and $\|\hat{\mathbf{u}}_i\|_p^p$ be the p^{th} power of the p-norm of i^{th} row of $\hat{\mathbf{U}}$. If we sample the points using $p_i = \min(q_i, 1)$ where $q_i = r_1\|\hat{\mathbf{u}}_i\|_p^p + r_2 t$, we can get a coreset with properties similar to 4 for the ℓ_p-regression problem with statistical parity constraints with high probability. The expected coreset size is $\tilde{O}(d\epsilon^{-2}((fg)^p + \ell))$.*

As an example, if we want to have a coreset for ℓ_1 regression with SP constraint, we have $fg = d^{1.5}$ (Dasgupta et al., 2009) which gives us a coreset of size approximately $\tilde{O}((d^{2.5} + \ell d)\epsilon^{-2})$.

C PROOFS FOR INDIVIDUALLY FAIR CLUSTERING

Recall the generalized definition of fair radius. Let $\tilde{W} = \sum_{i\in\mathbf{C}} \tilde{w}_i$. We denote $r^{\mathbf{C}}_{\tilde{W}/k}(\tilde{\mathbf{x}})$ as the maximum radius of a ball around $\tilde{\mathbf{x}} \in \mathbf{C}$ which contains points from $\mathbf{C}$ whose total weight is at most $\tilde{W}/k$.

C.1 Proof of Lemma 8

Proof. For each $\mathbf{x} \in \mathbf{P}$, consider $\mathbf{X}_\epsilon$ as the set of $\epsilon n/k$ points around $\mathbf{x}$. Now for each $\mathbf{x_i} \in \mathbf{X}_\epsilon$ we define random variable $y_i = 1$ with probability p_i and 0 otherwise. $Y = \sum_{i:\mathbf{x_i}\in\mathbf{X}_\epsilon} y_i$. Now to prove the given property we need

$Y > 0$ with high probability. Here $\mathbb{E}[Y] = \sum p_i \geq \sum t \geq c \log n$. For $\delta = 1/2$, Using Chernoff bound we get $Pr[Y \leq (1-\delta)\mathbb{E}[Y]] \leq 1/n^c$ Now taking a union bound over all $\mathbf{x} \in \mathbf{P}$, we get the property with probability at most $1/n^{c-1}$. Hence with probability at least $1-1/n$, for appropriate value of c, we get the Lemma statement. □

C.2 Proof of Lemma 9

Proof. Consider any ball $\mathbf{B}^{\mathbf{P}}(\mathbf{x}, r)$ for some $\mathbf{x} \in \mathbf{P}$ and some radius r satisfying $r \geq r^{\mathbf{P}}_{\epsilon n/k}(\mathbf{x})$. Now the weight of points of $\mathbf{P}$ in this ball is $\left|\mathbf{B}^{\mathbf{P}}(\mathbf{x}, r)\right|$. The weight of the points $\tilde{\mathbf{x}}$'s of the coreset $\mathbf{C}$ inside the same ball is $\sum_{\tilde{\mathbf{x}}_i \in \mathbf{B}^{\mathbf{C}}(\mathbf{x},r)} \tilde{w}_i$. Note that in expectation, the weight of points of coreset in the ball $\mathbf{B}^{\mathbf{C}}(\mathbf{x}, r)$ is $\left|\mathbf{B}^{\mathbf{P}}(\mathbf{x}, r)\right|$.

Now notice that $|\tilde{w}_i| \leq 1/t \leq \frac{\epsilon^3 n}{ck \log n}$. Also the variance of sum of weights of points of the coreset inside the ball is given by $\text{Var}[\sum_{\tilde{\mathbf{x}}_i \in \mathbf{B}^{\mathbf{C}}(\mathbf{x},r)} \tilde{w}_i] \leq \sum \mathbb{E}{\tilde{w}_i}^2 \leq \frac{1}{t} \sum \mathbb{E}\tilde{w}_i = \left|\mathbf{B}^{\mathbf{P}}(\mathbf{x}, r)\right|/t = \frac{\left|\mathbf{B}^{\mathbf{P}}(\mathbf{x},r)\right|\epsilon^3 n}{kc \log n}$. Now using Bernstein's inequality we get the following result for appropriate values of c.

$$\Pr\left(\left|\sum_{i:\tilde{\mathbf{x}}_i \in \mathbf{B}^{\mathbf{C}}(\mathbf{x},r)} \tilde{w}_i - \mathbb{E}\left[\sum_{i:\tilde{\mathbf{x}}_i \in \mathbf{B}^{\mathbf{C}}(\mathbf{x},r)} \tilde{w}_i\right]\right| \geq \epsilon \mathbb{E}\left[\sum_{i:\tilde{\mathbf{x}}_i \in \mathbf{B}^{\mathbf{C}}(\mathbf{x})} \tilde{w}_i\right]\right) \leq 1/n^3$$

Notice that, there are at most n^2 possible distances between the points. If we can claim the above statement for all these n^2 distances, it follows for all $\mathbf{r}$. Hence, taking a union bound over these n^2 possible distances, we get the required property with probability at least $(1 - O(1/n))$. □

C.3 Proof of Theorem 7

Proof. For the proof of the first property in definition 6 it is known (Langberg and Schulman, 2010; Feldman and Langberg, 2011; Chhaya et al., 2020c) that by taking expected number of samples equal to $\sum_i s_i$ times the VC-dimension of the query space (in this case $kd \log k$) we can get a coreset for the k-median problem without the fairness constraints which is essentially the same as our first property. Here we can use any upper bound on sensitivity scores. We use the bound given by Bachem et al. (2018b) which gives us the summation of sensitivity scores as $O(k)$. Using any other technique to bound sensitivities will give results accordingly.

For second property first let $\tilde{\mathbf{S}}$ be the solution obtained by solving the problem on the coreset. For any point $\mathbf{x} \in \mathbf{P}$ for which $d(\mathbf{x}, \tilde{\mathbf{S}}) \leq r^{\mathbf{P}}_{\epsilon n/k}(x)$, we are already satisfying the fairness property, since $r^{\mathbf{P}}_{\epsilon n/k}(x) \leq \alpha r^{\mathbf{P}}_{(1+\epsilon)n/k}(x)$. For any other $\mathbf{x} \in \mathbf{P}$, such that $d(\mathbf{x}, \tilde{\mathbf{S}}) \geq r^{\mathbf{P}}_{\epsilon n/k}(x)$, by using Lemma 9, we have that $\left|\mathbf{B}^{\mathbf{P}}(\mathbf{x}, d(\mathbf{x}, \tilde{\mathbf{S}}))\right| \leq \frac{1}{1-\epsilon}|\mathbf{B}^{\mathbf{C}}(\mathbf{x}, d(\mathbf{x}, \tilde{\mathbf{S}}))|$. Now,

$$\begin{aligned}\left|\mathbf{B}^{\mathbf{C}}(\mathbf{x}, d(\mathbf{x}, \tilde{\mathbf{S}}))\right| &\leq \left|\mathbf{B}^{\mathbf{C}}\left(\mathbf{x}, d(\mathbf{x}, r^{\mathbf{C}}_{\tilde{W}/k}(\mathbf{x}))\right)\right| \\ &\leq \frac{\tilde{W}}{k} \leq (1+\epsilon)n/k\end{aligned}$$

It follows that $r^{\mathbf{C}}_{\tilde{W}/k}(\mathbf{x}) \leq r^{\mathbf{P}}_{(1+O(\epsilon))n/k}(\mathbf{x})$. Suppose using the coreset $\mathbf{C}$ we obtain a solution $\tilde{\mathbf{S}}$ for the fair clustering problem, then we have the following:

$$\begin{aligned}d(\mathbf{x}, \tilde{\mathbf{S}}) &\leq d(\mathbf{x}, \tilde{\mathbf{x}}) + d(\tilde{\mathbf{x}}, \tilde{\mathbf{S}}) \\ &\leq r^{\mathbf{P}}_{\epsilon n/k}(\mathbf{x}) + \alpha r^{\mathbf{C}}_{\tilde{W}/k}(\tilde{\mathbf{x}}) \\ &\leq r^{\mathbf{P}}_{\epsilon n/k}(\mathbf{x}) + \alpha r^{\mathbf{P}}_{(1+\epsilon)n/k}(\tilde{\mathbf{x}}) \\ &\leq r^{\mathbf{P}}_{\epsilon n/k}(\mathbf{x}) + \alpha\left(r^{\mathbf{P}}_{\epsilon n/k}(\mathbf{x}) + r^{\mathbf{P}}_{(1+O(\epsilon))n/k}(\mathbf{x})\right) \\ &= (\alpha+1) r^{\mathbf{P}}_{\epsilon n/k}(\mathbf{x}) + \alpha r^{\mathbf{P}}_{(1+O(\epsilon))n/k}(\mathbf{x}) \\ &\leq (\alpha+1) r^{\mathbf{P}}_{(1+O(\epsilon))n/k}(\mathbf{x}) + \alpha r^{\mathbf{P}}_{(1+O(\epsilon))n/k}(\mathbf{x}) \\ &= (2\alpha+1) r^{\mathbf{P}}_{(1+O(\epsilon))n/k}(\mathbf{x})\end{aligned}$$

Here the first inequality is due to triangle inequality, second due to Lemma 8, and the third inequality is due to Lemma 9.

For the third property $\forall \tilde{\mathbf{x}} \in \mathbf{C}$, we know that $d(\tilde{\mathbf{x}}, \mathbf{S}) \leq \alpha r^{\mathbf{P}}_{n/k}(\tilde{\mathbf{x}})$. This inequality is by definition. Now to prove the third property it is enough to consider the weight of points in the coreset $\mathbf{C}$ that are contained within the ball of radius $r^{\mathbf{P}}_{n/k}(\mathbf{x})$. The weight $\tilde{W}/k$ is less than $(1+\epsilon)n/k \leq \frac{(1+\epsilon)\tilde{W}}{(1-\epsilon)k} \leq (1+3\epsilon)\tilde{W}/k$

Finally, to get a bound on the number of samples we calculate $r\sum_{i\in[n]} s_i = O(k)/\epsilon^2$. Multiplying this by the dimension and setting appropriate value of the failure probability i.e. $1/n$ we get the first part $O(k^2\epsilon^{-2}d\log k + k\log n)$. Also summing over t, we get $O(k\log n/\epsilon^3)$.Hence our final expected coreset size is $O(k^2\epsilon^{-2}d\log k + (k\log n)/\epsilon^3)$. Now the time taken to build the coreset depends on the method to get the upper bounds on the sensitivity scores s_i's. The upper bounds we have used in the proof using the technique given by Bachem et al. (2018b) takes time $\tilde{O}(nkd)$ □

C.4 Proof of Corollary 10

Proof. Notice that the first part is satisfied by the definition of $\mathbf{C}$. Here the important point to notice is that $\mathbf{S}_{opt}$ is also feasible for the fair radius on the coreset $r^{\mathbf{C}}_{\frac{\sum\tilde{W}}{k}}(\tilde{\mathbf{x}})$. This is because of Lemma 9. Now we can apply the same technique as in the proof of corollary 5 and get the second part of the result. □

Next we present formal version of the result that extends our results for individually fair clustering for other ℓ_p-costs

Corollary 13. *Replacing the s_i's of Algorithm 2 for fair k-median clustering with the s_i's of fair clustering with other ℓ_p norm costs we can get coreset for fair individual clustering with any ℓ_p norm cost for $p \geq 1$. The expected coreset size is $O(8^p k^2\epsilon^{-2}d\log k + (k\log n)/\epsilon)$*

Proof. The fairness constraints for the α-fair k clustering with any ℓ_p norm remain the same. Hence the proof for properties 2 and 3 of definition 6 remain the same. We only need to prove property 1. For property 1 we only need a bound on the sum of s_i's for ℓ_p-norm k-clustering. By Bachem et al. (2018b) we can bound the sum of s_i's for any $p \geq 1$ as $8^p k$. the rest of the argument remains the same as that in the proof of Theorem 7. □

D ADDITIONAL EXPERIMENTS

Here we present a bunch of additional experiments and some more details about the dataset. We will release the code on acceptance of the paper.

D.1 Fair Regression with SP Constraints

More Details on Datasets:

1) **Law School Admissions** (*Law-School*): This is the dataset from Law School Admissions Council's National Longitudinal Bar Passage Study (Wightman, 1998) which has $20,649$ examples with 11 columns. 'Race' is the protected attribute and the task here is to predict student's GPA (normalized to $[0,1]$).

2) **Communities & Crime** (*Communities*): This dataset contains $1,994$ examples and 122 features and contains crime, socio-economic, law enforcement data about communities in the US (Redmond and Baveja, 2002). 'Race' is the protected attribute and the task is to predict the number of violent crimes per 100,000 population (normalized to $[0,1]$). For the *Law-School* dataset the train-test split is set to 50-50 and for *Communities* dataset it is set to 90-10 respectively.

In figure 1 in main paper, we had compared the performance of our coreset with uniform sampling on both datasets. Next we check the accuracy achieved by `FairRegCor` model when trained with varying sized coresets for different values of ζ. In Figure 5 we observe that as the size of the coreset increases the *Rel. RMSE Err.* for the `FairRegCor` model approaches to zero. The *Disparity Difference* for different sizes of the coreset is also very low, and thus we can say that the disparity incurred by the `FairRegCor` model when evaluated on the train data is very close to that obtained by `FullModel` on the train data.

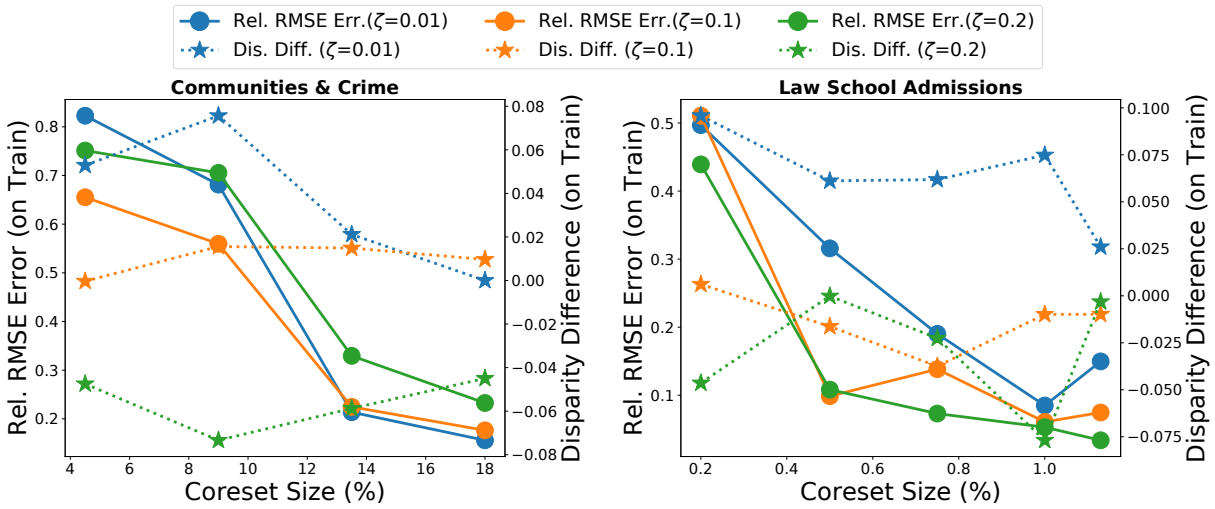

Figure 5: Relative RMSE Error and Disparity Difference for the `FairRegCor` model on Train data with increasing Coreset Size.

Table 2: Fair Regression Time(secs) on *Law-School* dataset for different sample sizes

ζ — size	200	400	600	800	1000	2000	4000	5000	7000	10000	15000	20000
0.01	29	79	129	187	235	351	836	994	672	1989	1914	2779
0.1	20	31	58	48	100	158	384	480	536	988	1439	1988
0.2	8	31	46	48	100	154	393	384	535	767	1436	1589
0.3	8	31	23	32	40	77	307	384	671	767	1482	1918
0.5	8	15	23	32	40	77	153	192	269	383	577	764
0.7	7	15	23	32	40	77	154	191	268	383	573	766

In Figure 6 the performance comparison of `FairRegCor` and `Uniform` algorithms on the train data for a fixed coreset size while changing the allowed value of slack i.e. disparity ζ. It can be observed that for almost all the values of ζ the *RMSE* error incurred by `FairRegCor` is low as compared to `Uniform`. In case where the *RMSE* in `Uniform` is better or lower, the disparity difference is high in case of `Uniform`, except for a point or two. We observe that as the slack ζ increases, the *RMSE* for the models decreases which is in line with the experiments by Agarwal et al. (2019).

To signify the importance of coresets we perform another experiment that records the time taken by the fair regression algorithm of Agarwal et al. (2019). Table 2 shows the time taken by the algorithm in seconds for different values of ζ and different sample sizes for the *Law-School* dataset. Currently the code can handle weights by making copies, however, it is clear from Table 2 and Figure 5 that if we have black box code for fair regression that can handle weighted points we can achieve comparable accuracy to full data using our coresets with huge speedup. Roughly speaking the speedup vs accuracy can be obtained by extrapolating from this table. For instance, take the LawSchool data with $\xi = 0.1$. For a 1% coreset (size 200), we have close to $100X$ speedup (20 secs vs 1988 seconds) , while having relative RMSE error being less than 0.1

Results on Test data: In Figures 7, and 8 we present the results when the models `FairRegCor` and `Uniform` are evaluated on the test data. Figure 7 shows the performance of `FairRegCor` model in terms of *RMSE* and *Disparity Difference* for increasing coreset size and for different values of the slack ζ. Here as well, as similar to that in case of the train data, the performance on the test data improves with the increase in the size of the data used for training. The curves seen here are similar to that observed when these models are evaluated on the train data.

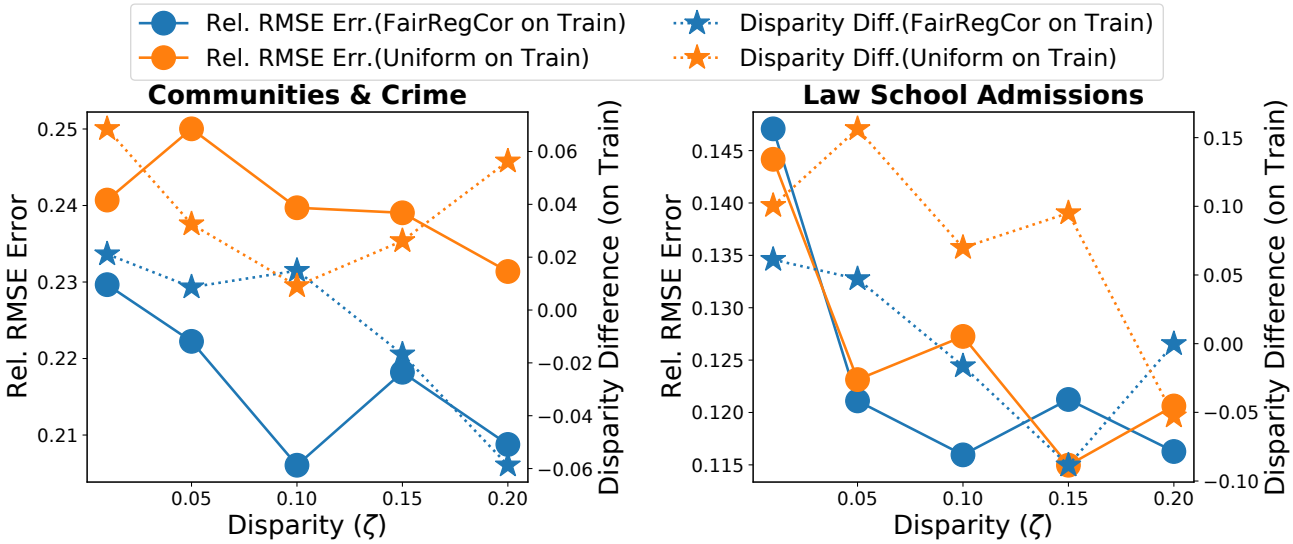

Figure 6: RMSE Error and Disparity Difference for the `FairRegCor` and `Uniform` models on Train data for fixed coreset size (13% *Communities & Crime*, 0.5% *Law-School*).

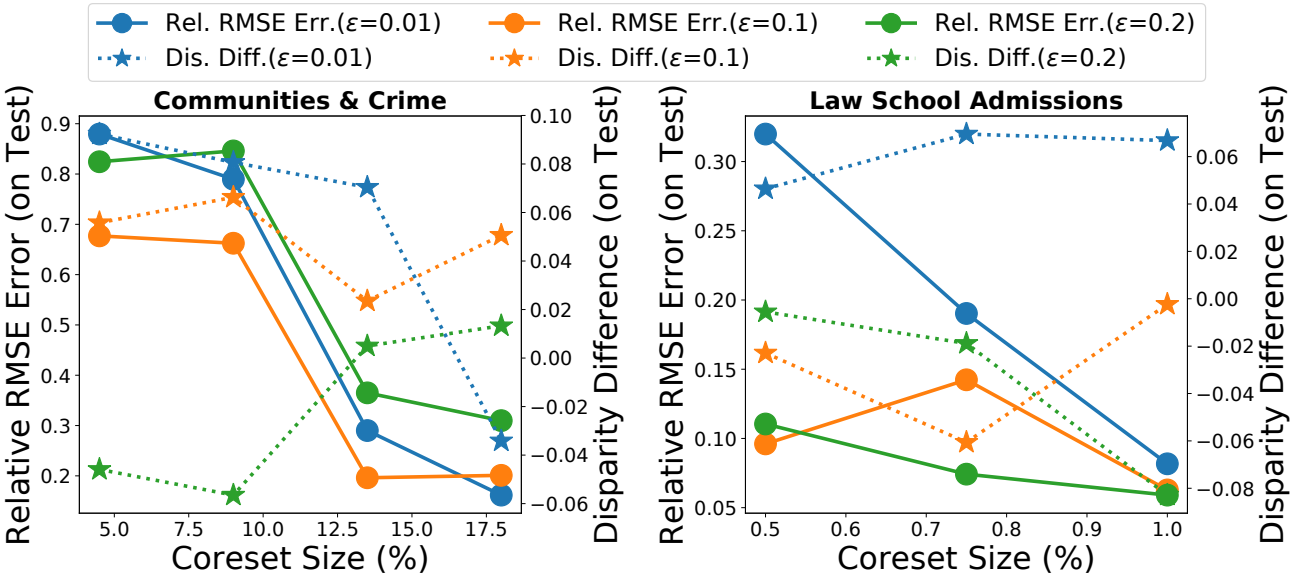

Figure 7: Relative RMSE Error and Disparity Difference for the `FairRegCor` model on Test data with increasing Coreset Size.

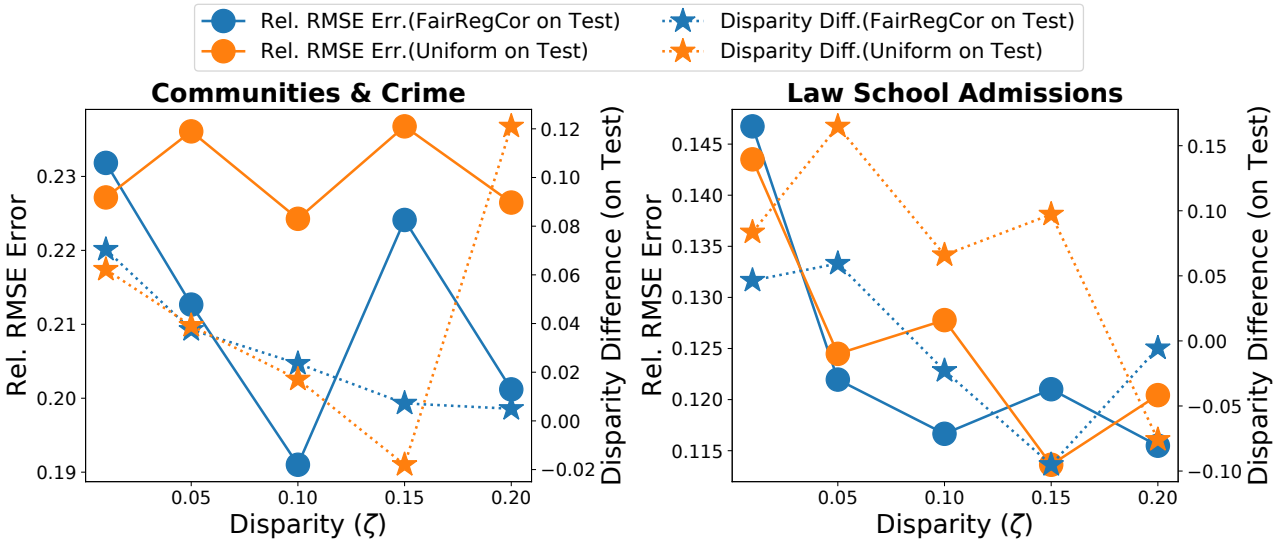

Figure 8: RMSE Error and Disparity Difference for the `FairRegCor` and `Uniform` models on Test data for fixed coreset size (13% *Communities & Crime*, 0.5% *Law-School*).

D.2 More Experiments and Details for Individually Fair Clustering

Additional Details about Datasets Following are details of the two datasets:

1. *Diabetes*: This dataset contains information of diabetes patients from hospitals across US[4]. The dataset has 101,765 points and like Mahabadi and Vakilian (2020) we considered two features : "age" and "time in hospitals".

2. *Census*: The new version of this dataset also known as Adult[5] contains 48,842 points and we have considered the five numeric attributes: "age" ,"fnlwgt", "education-num", "capital-gain" and "hours-per-week"

Semisynthetic Datasets: To emphasize the effect of non-uniform sampling, we also create a semi-synthetic data using the above real datasets. We first sample (2500) points uniformly at random and then use a power law distribution over this set and make copies of sampled points to increase it to of size 10000. The power law parameter (α) value was set to 1.5.

In figure 9 we show the performance of our coreset algorithm (`IndFair`) along with the Uniform coreset (`Uni`) and k-median based coreset (`KMedian`) in terms of the percentage relative error in cost as compared to Mahabadi and Vakilian (2020) algorithm. As expected for all the algorithms the relative error decreases with the increase in the coreset size. We observe that for different values of k, performance of `IndFair` is comparable or at par with other baseline algorithms.

To evaluate the fairness of `IndFair` we look at the difference in the average maximum fairness achieved by our `IndFair` algorithm and the fairness achieved by Mahabadi and Vakilian (2020) algorithm. From the plot in figure 10 we observe that difference in the fairness values obtained by our `IndFair` algorithm and that by Mahabadi and Vakilian (2020) algorithm is very small for different values of k.

[4]https://archive.ics.uci.edu/ml/datasets/diabetes+130-us+hospitals+for+years+1999-2008 Last Accessed: 26th September 2021

[5]https://archive.ics.uci.edu/ml/datasets/adult Last Accessed: 26th September 2021

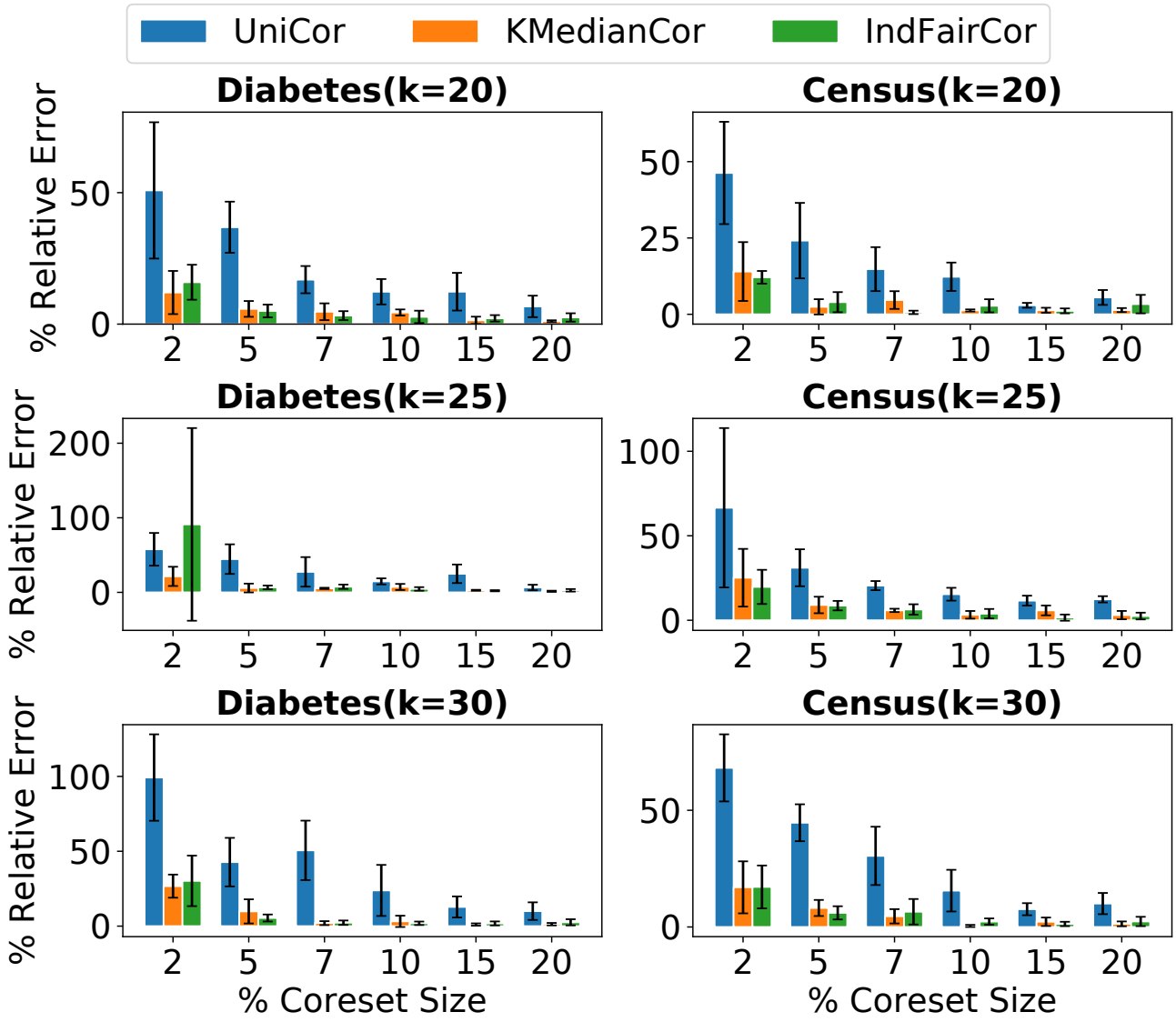

Figure 9: Rel. Err. in Cost (wrt. to Mahabadi and Vakilian (2020) algo.) on Semi-synthetic datasets

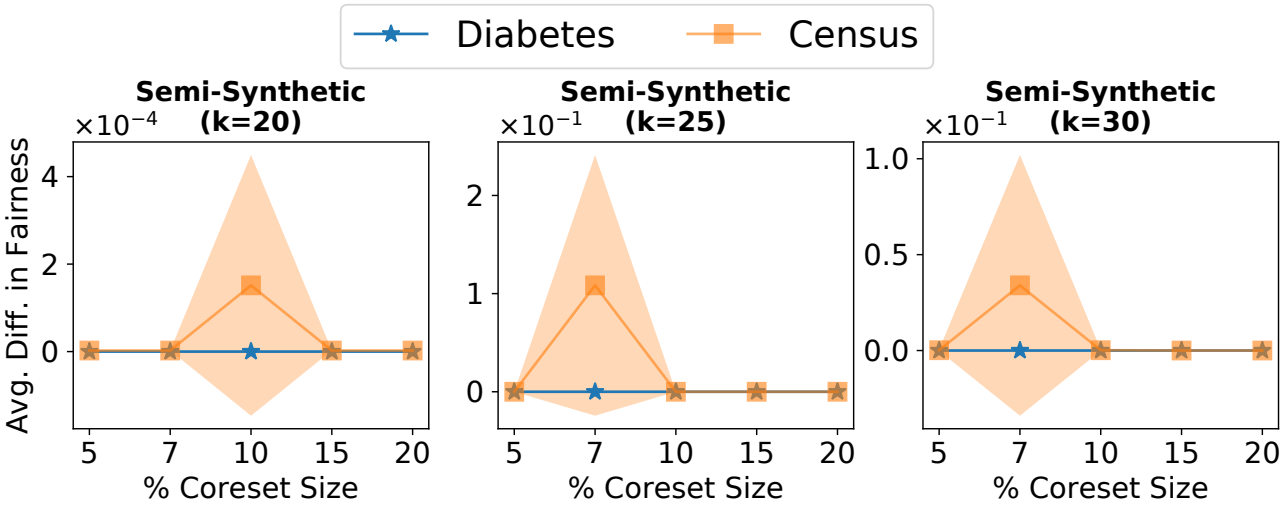

Figure 10: Difference in fairness by Mahabadi and Vakilian (2020) algo. and **IndFair** Coreset algo. on Semi-synthetic datasets