Sampling Algorithms and Coresets for ℓ_p Regression

Anirban Dasgupta * Petros Drineas † Boulos Harb ‡ Ravi Kumar *
Michael W. Mahoney *

Abstract

The ℓ_p *regression problem* takes as input a matrix $A \in \mathbb{R}^{n\times d}$, a vector $b \in \mathbb{R}^n$, and a number $p \in [1,\infty)$, and it returns as output a number $\mathcal{Z}$ and a vector $x_{\text{OPT}} \in \mathbb{R}^d$ such that $\mathcal{Z} = \min_{x\in\mathbb{R}^d} \|Ax-b\|_p = \|Ax_{\text{OPT}} - b\|_p$. In this paper, we construct coresets and obtain an efficient two-stage sampling-based approximation algorithm for the very overconstrained ($n \gg d$) version of this classical problem, for all $p \in [1,\infty)$. The first stage of our algorithm non-uniformly samples $\hat{r}_1 = O(36^p d^{\max\{p/2+1,p\}+1})$ rows of A and the corresponding elements of b, and then it solves the ℓ_p regression problem on the sample; we prove this is an 8-approximation. The second stage of our algorithm uses the output of the first stage to resample $\hat{r}_1/\epsilon^2$ constraints, and then it solves the ℓ_p regression problem on the new sample; we prove this is a $(1+\epsilon)$-approximation. Our algorithm unifies, improves upon, and extends the existing algorithms for special cases of ℓ_p regression, namely $p = 1, 2$ [10, 13]. In course of proving our result, we develop two concepts—well-conditioned bases and subspace-preserving sampling—that are of independent interest.

1 Introduction

An important question in algorithmic problem solving is whether there exists a *small* subset of the input such that if computations are performed only on this subset, then the solution to the given problem can be *approximated* well. Such a subset is often known as a *coreset* for the problem. The concept of coresets has been extensively used in solving many problems in optimization and computational geometry; e.g., see the excellent survey by Agarwal, Har-Peled, and Varadarajan [2].

In this paper, we construct coresets and obtain efficient sampling algorithms for the classical ℓ_p regression problem, for all $p \in [1,\infty)$. Recall the ℓ_p *regression problem*:

PROBLEM 1.1. (ℓ_p REGRESSION PROBLEM) *Let* $\|\cdot\|_p$ *denote the p-norm of a vector. Given as input a matrix* $A \in \mathbb{R}^{n\times m}$, *a target vector* $b \in \mathbb{R}^n$, *and a real number* $p \in [1,\infty)$, *find a vector* x_{OPT} *and a number* $\mathcal{Z}$ *such that*

$$\mathcal{Z} = \min_{x\in\mathbb{R}^m} \|Ax-b\|_p = \|Ax_{\text{OPT}} - b\|_p . \tag{1.1}$$

In this paper, we will use the following ℓ_p *regression coreset* concept:

DEFINITION 1.1. (ℓ_p REGRESSION CORESET) *Let* $0 < \epsilon < 1$. *A* coreset *for Problem 1.1 is a set of indices* $\mathcal{I}$ *such that the solution* $\hat{x}_{\text{OPT}}$ *to* $\min_{x\in\mathbb{R}^m} \|\hat{A}x - \hat{b}\|_p$, *where* $\hat{A}$ *is composed of those rows of* A *whose indices are in* $\mathcal{I}$ *and* $\hat{b}$ *consists of the corresponding elements of* b, *satisfies*

$$\|A\hat{x}_{\text{OPT}} - b\|_p \le (1+\epsilon)\min_x \|Ax-b\|_p .$$

If $n \gg m$, i.e., if there are many more constraints than variables, then (1.1) is an *overconstrained ℓ_p regression problem*. In this case, there does not in general exist a vector x such that $Ax = b$; thus $\mathcal{Z} > 0$. Overconstrained regression problems are fundamental in statistical data analysis and have numerous applications in applied mathematics, data mining, and machine learning [16, 9]. Even though convex programming methods can be used to solve the overconstrained regression problem in time $O((mn)^c)$, for $c > 1$, this is prohibitive if n is large.[1] This raises the natural question of developing more efficient algorithms that run in time $O(m^c n)$, for $c > 1$, while possibly relaxing the solution to Equation (1.1). In particular: Can we get a κ-approximation to the ℓ_p regression problem, i.e., a vector $\hat{x}$ such that $\|A\hat{x} - b\|_p \le \kappa\mathcal{Z}$, where $\kappa > 1$? Note that a coreset of

*Yahoo! Research, 701 First Ave., Sunnyvale, CA 94089. Email: {anirban, ravikumar, mahoney}@yahoo-inc.com

†Computer Science, Rensselaer Polytechnic Institute, Troy, NY 12180. Work done while the author was visiting Yahoo! Research. Email: drinep@cs.rpi.edu

‡Computer Science, University of Pennsylvania, Philadelphia, PA 19107. Work done while the author was visiting Yahoo! Research. Email: boulos@cis.upenn.edu

[1] For the special case of $p = 2$, vector space methods can solve the regression problem in time $O(m^2 n)$, and if $p = 1$ linear programming methods can be used.

small size would strongly satisfy our requirements and result in an efficiently computed solution that's almost as good as the optimal. Thus, the question becomes: Do coresets exist for the ℓ_p regression problem, and if so can we compute them efficiently?

Our main result is an efficient two-stage sampling-based approximation algorithm that constructs a coreset and thus achieves a $(1+\epsilon)$-approximation for the ℓ_p regression problem. The first-stage of the algorithm is sufficient to obtain a (fixed) constant factor approximation. The second-stage of the algorithm carefully uses the output of the first-stage to construct a coreset and achieve arbitrary constant factor approximation.

1.1 Our contributions

1.1.1 Summary of results For simplicity of presentation, we summarize the results for the case of $m = d = \text{rank}(A)$. Let $k = \max\{p/2 + 1,\ p\}$ and let $\phi(r, d)$ be the time required to solve an ℓ_p regression problem with r constraints and d variables. In the first stage of the algorithm, we compute a set of sampling probabilities $p_1, \ldots, p_n$ in time $O(nd^5 \log n)$, sample $\widehat{r_1} = O(36^p d^{k+1})$ rows of A and the corresponding elements of b according to the p_i's, and solve an ℓ_p regression problem on the (much smaller) sample; we prove this is an 8-approximation algorithm with a running time of $O\left(nd^5 \log n + \phi(\widehat{r_1}, d)\right)$. In the second stage of the algorithm, we use the residual from the first stage to compute a new set of sampling probabilities $q_1, \ldots, q_n$, sample additional $\widehat{r_2} = O(\widehat{r_1}/\epsilon^2)$ rows of A and the corresponding elements of b according to the q_i's, and solve an ℓ_p regression problem on the (much smaller) sample; we prove this is a $(1+\epsilon)$-approximation algorithm with a total running time of $O\left(nd^5 \log n + \phi(\widehat{r_2}, d)\right)$ (Section 4). We also show how to extend our basic algorithm to commonly encountered and more general settings of constrained, generalized, and weighted ℓ_p regression problems (Section 5).

We note that the l_p regression problem for $p = 1, 2$ has been studied before. For $p = 1$, Clarkson [10] uses a subgradient based algorithm to preprocess A and b and then samples the rows of the modified problem; these elegant techniques however depend crucially on the linear structure of the l_1 regression problem[2]. Furthermore, this algorithm does not yield coresets. For $p = 2$, Drineas, Mahoney, and Muthukrishnan [13] construct coresets by exploiting the singular value decomposition, a property peculiar to the l_2 space. Thus in order to efficiently compute coresets for the ℓ_p regression problem for all $p \in [1, \infty)$, we need tools that capture the geometry of l_p norms. In this paper we develop the following two tools that may be of independent interest (Section 3).

(1) *Well-conditioned bases.* Informally speaking, if U is a well-conditioned basis, then for all $z \in \mathbb{R}^d$, $\|z\|_p$ should be close to $\|Uz\|_p$. We will formalize this by requiring that for all $z \in \mathbb{R}^d$, $\|z\|_q$ multiplicatively approximates $\|Uz\|_p$ by a factor that can depend on d but is *independent* of n (where p and q are conjugate; i.e., $q = p/(p-1)$). We show that these bases exist and can be constructed in time $O(nd^5 \log n)$. In fact, our notion of a well-conditioned basis can be interpreted as a computational analog of the Auerbach and Lewis bases studied in functional analysis [24]. They are also related to the barycentric spanners recently introduced by Awerbuch and R. Kleinberg [5] (Section 3.1). J. Kleinberg and Sandler [17] defined the notion of an ℓ_1-independent basis, and our well-conditioned basis can be used to obtain an exponentially better "condition number" than their construction. Further, Clarkson [10] defined the notion of an "ℓ_1-conditioned matrix," and he preprocessed the input matrix to an ℓ_1 regression problem so that it satisfies conditions similar to those satisfied by our bases.

(2) *Subspace-preserving sampling.* We show that sampling rows of A according to information in the rows of a well-conditioned basis of A minimizes the sampling variance and consequently, the rank of A is not lost by sampling. This is critical for our relative-error approximation guarantees. The notion of subspace-preserving sampling was used in [13] for $p = 2$, but we abstract and generalize this concept for all $p \in [1, \infty)$.

We note that for $p = 2$, our sampling complexity matches that of [13], which is $O(d^2/\epsilon^2)$; and for $p = 1$, it improves that of [10] from $O(d^{3.5}(\log d)/\epsilon^2)$ to $O(d^{2.5}/\epsilon^2)$.

1.1.2 Overview of our methods Given an input matrix A, we first construct a well-conditioned basis for A and use that to obtain bounds on a slightly non-standard notion of a p-norm condition number of a matrix. The use of this particular condition number is crucial since the variance in the subspace preserving sampling can be upper bounded in terms of it. An ε-net argument then shows that the first stage sampling gives us a 8-approximation. The next twist is to use the output of the first stage as a feedback to fine-tune the sampling probabilities. This is done so that the "positional information" of b with respect to A is also preserved in addition to the subspace. A more careful use of a different ε-net shows that the second stage sampling achieves a $(1+\epsilon)$-approximation.

[2] Two ingredients of [10] use the linear structure: the subgradient based preprocessing itself, and the counting argument for the concentration bound.

1.2 Related work As mentioned earlier, in course of providing a sampling-based approximation algorithm for ℓ_1 regression, Clarkson [10] shows that coresets exist and can be computed efficiently for a *controlled* ℓ_1 regression problem. Clarkson first preprocesses the input matrix A to make it well-conditioned with respect to the ℓ_1 norm then applies a subgradient-descent-based approximation algorithm to guarantee that the ℓ_1 norm of the target vector is conveniently bounded. Coresets of size $O(d^{3.5}\log d/\epsilon^2)$ are thereupon exhibited for this modified regression problem. For the ℓ_2 case, Drineas, Mahoney and Muthukrishnan [13] designed sampling strategies to preserve the subspace information of A and proved the existence of a coreset of rows of size $O(d^2/\epsilon^2)$—for the *original* ℓ_2 regression problem; this leads to a $(1+\epsilon)$-approximation algorithm. While their algorithm used $O(nd^2)$ time to construct the coreset and solve the ℓ_2 regression problem—which is sufficient time to solve the regression problem—in a subsequent work, Sarlós [18] improved the running time for solving the regression problem to $\tilde{O}(nd)$ by using random sketches based on the Fast Johnson–Lindenstrauss transform of Ailon and Chazelle [3].

More generally, embedding d-dimensional subspaces of L_p into $\ell_p^{f(d)}$ using coordinate restrictions has been extensively studied [19, 7, 21, 22, 20]. Using well-conditioned bases, one can provide a constructive analog of Schechtman's existential L_1 embedding result [19] (see also [7]), that any d-dimensional subspace of $L_1[0,1]$ can be embedded in ℓ_1^r with distortion $(1+\epsilon)$ with $r = O(d^2/\epsilon^2)$, albeit with an extra factor of $\sqrt{d}$ in the sampling complexity. Coresets have been analyzed by the computation geometry community as a tool for efficiently approximating various extent measures [1, 2]; see also [15, 6, 14] for applications of coresets in combinatorial optimization. An important difference is that most of the coreset constructions are exponential in the dimension, and thus applicable only to low-dimensional problems, whereas our coresets are polynomial in the dimension, and thus applicable to high-dimensional problems.

2 Preliminaries

Given a vector $x \in \mathbb{R}^m$, its *p-norm* is $\|x\|_p = \sum_{i=1}^m (|x_i|^p)^{1/p}$, and the *dual norm* of $\|\cdot\|_p$ is denoted $\|\cdot\|_q$, where $1/p + 1/q = 1$. Given a matrix $A \in \mathbb{R}^{n\times m}$, its *generalized p-norm* is

$$|||A|||_p = (\sum_{i=1}^{n}\sum_{j=1}^{m}|A_{ij}|^p)^{1/p}.$$

This is a submultiplicative matrix norm that generalizes the Frobenius norm from $p = 2$ to all $p \in [1,\infty)$, but it is not a vector-induced matrix norm. The j-th column of A is denoted $A_{\star j}$, and the i-th row is denoted $A_{i\star}$. In this notation, $|||A|||_p = (\sum_j \|A_{\star j}\|_p^p)^{1/p} = (\sum_i \|A_{i\star}\|_p^p)^{1/p}$. For $x, x', x'' \in \mathbb{R}^m$, it can be shown using Hölder's inequality that

$$\|x - x'\|_p^p \le 2^{p-1}\left(\|x - x''\|_p^p + \|x'' - x'\|_p^p\right).$$

Two crucial ingredients in our proofs are ε-nets and tail-inequalities. A subset $\mathcal{N}(D)$ of a set D is called an *ε-net* in D for some $\varepsilon > 0$ if for every $x \in D$, there is a $y \in \mathcal{N}(D)$ with $\|x - y\| \le \varepsilon$. In order to construct an ε-net for D it is enough to choose $\mathcal{N}(D)$ to be the maximal set of points that are pairwise ε apart. It is well known that the unit ball of a d-dimensional space has an ε-net of size at most $(3/\varepsilon)^d$ [7].

Throughout this paper, we will use the following sampling matrix formalism to represent our sampling operations. Given a set of n probabilities, $p_i \in (0,1]$, for $i = 1,\ldots,n$, let S be an $n \times n$ diagonal sampling matrix such that S_{ii} is set to $1/p_i^{1/p}$ with probability p_i and to zero otherwise. Clearly, premultiplying A or b by S determines whether the i-th row of A and the corresponding element of b will be included in the sample, and the expected number of rows/elements selected is $r' = \sum_{i=1}^n p_i$. (In what follows, we will abuse notation slightly by ignoring zeroed out rows and regarding S as an $r' \times n$ matrix and thus SA as an $r' \times m$ matrix.) Thus, e.g., sampling constraints from Equation (1.1) and solving the induced subproblem may be represented as solving

$$\hat{\mathcal{Z}} = \min_{\hat{x}\in\mathbb{R}^m} \|SA\hat{x} - Sb\|_p. \tag{2.2}$$

A vector $\hat{x}$ is said to be a *κ-approximation* to the ℓ_p regression problem of Equation (1.1), for $\kappa \ge 1$, if $\|A\hat{x} - b\|_p \le \kappa\mathcal{Z}$.

Finally, several proofs are omitted from this extended abstract; all the missing proofs may be found in the technical report version of this paper [11].

3 Main technical ingredients

In this section, we describe two concepts that will be used in the proof of our main result but that are of independent interest. The first is the concept of a basis that is well-conditioned for a p-norm in a manner analogous to that in which an orthogonal matrix is well-conditioned for the Euclidean norm. The second is the idea of using information in that basis to construct subspace-preserving sampling probabilities.

3.1 Well-conditioned bases We introduce the following notion of a "well-conditioned" basis.

DEFINITION 3.1. (WELL-CONDITIONED BASIS) *Let A be an $n \times m$ matrix of rank d, let $p \in [1, \infty)$, and let q be its dual. Then an $n \times d$ matrix U is an* (α, β, p)-well-conditioned basis *for the column space of A if*

- $|||U|||_p \leq \alpha$, *and*
- *for all* $z \in \mathbb{R}^d$, $\|z\|_q \leq \beta \|Uz\|_p$.

We will say that U is a p-well-conditioned basis *for the column space of A if α and β are $d^{O(1)}$, independent of m and n.*

Recall that any orthonormal basis U for $\text{span}(A)$ satisfies both $|||U|||_2 = \|U\|_F = \sqrt{d}$ and also $\|z\|_2 = \|Uz\|_2$ for all $z \in \mathbb{R}^d$, and thus is a $(\sqrt{d}, 1, 2)$-well-conditioned basis. Thus, Definition 3.1 generalizes to an arbitrary p-norm, for $p \in [1, \infty)$, the notion that an orthogonal matrix is well-conditioned with respect to the 2-norm. Note also that duality is incorporated into Definition 3.1 since it relates the p-norm of the vector $z \in \mathbb{R}^d$ to the q-norm of the vector $Uz \in \mathbb{R}^n$, where p and q are dual.[3]

The existence and efficient construction of these bases is given by the following.

THEOREM 3.1. *Let A be an $n \times m$ matrix of rank d, let $p \in [1, \infty)$, and let q be its dual norm. Then there exists an* (α, β, p)-well-conditioned basis *U for the column space of A such that:*

- *if* $p < 2$, *then* $\alpha = d^{\frac{1}{p}+\frac{1}{2}}$ *and* $\beta = 1$,
- *if* $p = 2$, *then* $\alpha = d^{\frac{1}{2}}$ *and* $\beta = 1$, *and*
- *if* $p > 2$, *then* $\alpha = d^{\frac{1}{p}+\frac{1}{2}}$ *and* $\beta = d^{\frac{1}{q}-\frac{1}{2}}$.

Moreover, U can be computed in $O(nmd + nd^5 \log n)$ time (or in just $O(nmd)$ time if $p = 2$).

Proof. Let $A = QR$, where Q is any $n \times d$ matrix that is an orthonormal basis for $\text{span}(A)$ and R is a $d \times m$ matrix. If $p = 2$, then Q is the desired basis U; from the discussion following Definition 3.1, $\alpha = \sqrt{d}$ and $\beta = 1$, and computing it requires $O(nmd)$ time. Otherwise, fix Q and p and define the norm, $\|z\|_{Q,p} \triangleq \|Qz\|_p$. A quick check shows that $\|\cdot\|_{Q,p}$ is indeed a norm. ($\|z\|_{Q,p} = 0$ if and only if $z = 0$ since Q has full column rank; $\|\gamma z\|_{Q,p} = \|\gamma Qz\|_p = |\gamma| \|Qz\|_p = |\gamma| \|z\|_{Q,p}$; and $\|z + z'\|_{Q,p} = \|Q(z + z')\|_p \leq \|Qz\|_p + \|Qz'\|_p = \|z\|_{Q,p} + \|z'\|_{Q,p}$.)

[3]For $p = 2$, Drineas, Mahoney, and Muthukrishnan used this basis, i.e., an orthonormal matrix, to construct probabilities to sample the original matrix. For $p = 1$, Clarkson used a procedure similar to the one we describe in the proof of Theorem 3.1 to preprocess A such that the 1-norm of z is a $d\sqrt{d}$ factor away from the 1-norm of Az.

Consider the set $C = \{z \in \mathbb{R}^d : \|z\|_{Q,p} \leq 1\}$, which is the unit ball of the norm $\|\cdot\|_{Q,p}$. In addition, define the $d \times d$ matrix F such that $\mathcal{E}_{\text{LJ}} = \{z \in \mathbb{R}^d : z^T F z \leq 1\}$ is the Löwner–John ellipsoid of C. Since C is symmetric about the origin, $(1/\sqrt{d})\mathcal{E}_{\text{LJ}} \subseteq C \subseteq \mathcal{E}_{\text{LJ}}$; thus, for all $z \in \mathbb{R}^d$,

$$\|z\|_{\text{LJ}} \leq \|z\|_{Q,p} \leq \sqrt{d} \|z\|_{\text{LJ}} \quad , \tag{3.3}$$

where $\|z\|_{\text{LJ}}^2 = z^T F z$ (see, e.g. [8, pp. 413–4]). Since the matrix F is symmetric positive definite, we can express it as $F = G^T G$, where G is full rank and upper triangular. Since Q is an orthogonal basis for $\text{span}(A)$ and G is a $d \times d$ matrix of full rank, it follows that $U = QG^{-1}$ is an $n \times d$ matrix that spans the column space of A. We claim that $U \triangleq QG^{-1}$ is the desired p-well-conditioned basis.

To establish this claim, let $z' = Gz$. Thus, $\|z\|_{\text{LJ}}^2 = z^T F z = z^T G^T G z = (Gz)^T Gz = z'^T z' = \|z'\|_2^2$. Furthermore, since G is invertible, $z = G^{-1}z'$, and thus $\|z\|_{Q,p} = \|Qz\|_p = \left\|QG^{-1}z'\right\|_p$. By combining these expression with (3.3), it follows that for all $z' \in \mathbb{R}^d$,

$$\|z'\|_2 \leq \|Uz'\|_p \leq \sqrt{d} \|z'\|_2 \quad . \tag{3.4}$$

Since $|||U|||_p^p = \sum_j \|U_{\star j}\|_p^p = \sum_j \|Ue_j\|_p^p \leq \sum_j d^{\frac{p}{2}} \|e_j\|_2^p = d^{\frac{p}{2}+1}$, where the inequality follows from the upper bound in (3.4), it follows that $\alpha = d^{\frac{1}{p}+\frac{1}{2}}$. If $p < 2$, then $q > 2$ and $\|z\|_q \leq \|z\|_2$ for all $z \in \mathbb{R}^d$; by combining this with (3.4), it follows that $\beta = 1$. On the other hand, if $p > 2$, then $q < 2$ and $\|z\|_q \leq d^{\frac{1}{q}-\frac{1}{2}} \|z\|_2$; by combining this with (3.4), it follows that $\beta = d^{\frac{1}{q}-\frac{1}{2}}$.

In order to construct U, we need to compute Q and G and then invert G. Our matrix A can be decomposed into QR using the compact QR decomposition in $O(nmd)$ time. The matrix F describing the Löwner–John ellipsoid of the unit ball of $\|\cdot\|_{Q,p}$ can be computed in $O(nd^5 \log n)$ time. Finally, computing G from F takes $O(d^3)$ time, and inverting G takes $O(d^3)$ time.

3.1.1 Connection to barycentric spanners A point set $K = \{K_1, \ldots, K_d\} \subseteq D \subseteq \mathbb{R}^d$ is a *barycentric spanner* for the set D if every $z \in D$ may be expressed as a linear combination of elements of K using coefficients in $[-C, C]$, for $C = 1$. When $C > 1$, K is called a C-approximate barycentric spanner. Barycentric spanners were introduced by Awerbuch and R. Kleinberg in [5]. They showed that if a set is compact, then it has a barycentric spanner. Our proof shows that if A is an $n \times d$ matrix, then $\tau^{-1} = R^{-1}G^{-1} \in \mathbb{R}^{d \times d}$ is a $\sqrt{d}$-approximate barycentric spanner for $D = \{z \in \mathbb{R}^d : \|Az\|_p \leq 1\}$. To see this, first note that each $\tau_{\star j}^{-1}$ belongs to D since $\|A\tau_{\star j}^{-1}\|_p = \|Ue_j\|_p \leq \|e_j\|_2 = 1$, where

the inequality is obtained from Equation (3.4). Moreover, since τ^{-1} spans $\mathbb{R}^d$, we can write any $z \in D$ as $z = \tau^{-1}\nu$. Hence,

$$\frac{\|\nu\|_\infty}{\sqrt{d}} \leq \frac{\|\nu\|_2}{\sqrt{d}} \leq \|U\nu\|_p = \left\|A\tau^{-1}\nu\right\|_p = \|Az\|_p \leq 1 \ ,$$

where the second inequality is also obtained from Equation (3.4). This shows that our basis has the added property that every element $z \in D$ can be expressed as a linear combination of elements (or columns) of τ^{-1} using coefficients whose ℓ_2 norm is bounded by $\sqrt{d}$.

3.1.2 Connection to Auerbach bases An *Auerbach basis* $U = \{U_{\star j}\}_{j=1}^d$ for a d-dimensional normed space $\mathcal{A}$ is a basis such that $\|U_{\star j}\|_p = 1$ for all j and such that whenever $y = \sum_j \nu_j U_{\star j}$ is in the unit ball of $\mathcal{A}$ then $|\nu_j| \leq 1$. The existence of such a basis for every finite dimensional normed space was first proved by Herman Auerbach [4] (see also [12, 23]). It can easily be shown that an Auerbach basis is an (α, β, p)-well-conditioned basis, with $\alpha = d$ and $\beta = 1$ for all p. Further, suppose U is an Auerbach basis for span(A), where A is an $n \times d$ matrix of rank d. Writing $A = U\tau$, it follows that τ^{-1} is an *exact* barycentric spanner for $D = \{z \in \mathbb{R}^d : \|Az\|_p \leq 1\}$. Specifically, each $\tau_{\star j}^{-1} \in D$ since $\|A\tau_{\star j}^{-1}\|_p = \|U_{\star j}\|_p = 1$. Now write $z \in D$ as $z = \tau^{-1}\nu$. Since the vector $y = Az = U\nu$ is in the unit ball of span(A), we have $|\nu_j| \leq 1$ for all $1 \leq j \leq d$. Therefore, computing a barycentric spanner for the compact set D—which is the pre-image of the unit ball of span(A)—is equivalent (up to polynomial factors) to computing an Auerbach basis for span(A).

3.2 Subspace-preserving sampling In the previous subsection (and in the notation of the proof of Theorem 3.1), we saw that given $p \in [1, \infty)$, any $n \times m$ matrix A of rank d can be decomposed as

$$A = QR = QG^{-1}GR = U\tau \ ,$$

where $U = QG^{-1}$ is a p-well-conditioned basis for span(A) and $\tau = GR$. The significance of a p-well-conditioned basis is that we are able to minimize the variance in our sampling process by randomly sampling *rows* of the matrix A and elements of the vector b according to a probability distribution that depends on norms of the *rows* of the matrix U. This will allow us to preserve the subspace structure of span(A) and thus to achieve relative-error approximation guarantees.

More precisely, given $p \in [1, \infty)$ and any $n \times m$ matrix A of rank d decomposed as $A = U\tau$, where U is an (α, β, p)-well-conditioned basis for span(A), consider any set of sampling probabilities p_i for $i = 1, \ldots, n$, that satisfy:

$$p_i \geq \min\left\{1, \frac{\|U_{i\star}\|_p^p}{|||U|||_p^p} r\right\} \ , \tag{3.5}$$

where $r = r(\alpha, \beta, p, d, \epsilon)$ to be determined below. Let us randomly sample the i^{th} row of A with probability p_i, for all $i = 1, \ldots, n$. Recall that we can construct a diagonal sampling matrix S, where each $S_{ii} = 1/p_i^{1/p}$ with probability p_i and 0 otherwise, in which case we can represent the sampling operation as SA.

The following theorem is our main result regarding this subspace-preserving sampling procedure.

THEOREM 3.2. *Let A be an $n \times m$ matrix of rank d, and let $p \in [1, \infty)$. Let U be an (α, β, p)-well-conditioned basis for* span(A)*, and let us randomly sample rows of A according to the procedure described above using the probability distribution given by Equation* (3.5)*, where*

$$r \geq 32^p(\alpha\beta)^p(d\ln(\frac{12}{\epsilon}) + \ln(\frac{2}{\delta}))/(p^2\epsilon^2).$$

Then, with probability $1 - \delta$, the following holds for all $x \in \mathbb{R}^m$:

$$|\ \|SAx\|_p - \|Ax\|_p\ | \leq \epsilon \|Ax\|_p .$$

Several things should be noted about this result. First, it implies that $\text{rank}(SA) = \text{rank}(A)$, since otherwise we could choose a vector $x \in \text{null}(SA)$ and violate the theorem. In this sense, this theorem generalizes the subspace-preservation result of Lemma 4.1 of [13] to all $p \in [1, \infty)$. Second, regarding sampling complexity: if $p < 2$ the sampling complexity is $O(d^{\frac{p}{2}+2})$, if $p = 2$ it is $O(d^2)$, and if $p > 2$ it is $O(dd^{\frac{1}{p}+\frac{1}{2}}d^{\frac{1}{q}-\frac{1}{2}})^p = O(d^{p+1})$. Finally, note that this theorem is analogous to the main result of Schechtman [19], which uses the notion of Auerbach bases.

4 The sampling algorithm

In this section, we present our main sampling algorithm for the ℓ_p-regression problem; we present a quality-of-approximation theorem; and we outline a proof of this threorem. Recall that omitted parts of the proof may be found in the technical report [11].

4.1 Statement of our main algorithm and theorem Our main sampling algorithm for approximating the solution to the ℓ_p regression problem is presented in Figure 1.[4] The algorithm takes as input an $n \times m$ matrix A of rank d, a vector $b \in \mathbb{R}^n$, and a number

[4] It has been brought to our attention by an anonymous reviewer that one of the main results of this section can be

Input: An $n \times m$ matrix A of rank d, a vector $b \in \mathbb{R}^n$, and $p \in [1, \infty)$.

Let $0 < \epsilon < 1/7$, and define $k = \max\{p/2 + 1, p\}$.

- Find a p-well-conditioned basis $U \in \mathbb{R}^{n \times d}$ for span(A) (as in the proof of Theorem 3.1) .
- **Stage 1:** Define $p_i = \min\left\{1, \frac{\|U_{i\star}\|_p^p}{\|U\|_p^p} r_1\right\}$ where $r_1 = 8^2 \cdot 36^p d^k \left(d \ln(8 \cdot 36) + \ln(200)\right)$.
  - Generate (implicitly) S where $S_{ii} = 1/p_i^{1/p}$ with probability p_i and 0 otherwise.
  - Let $\hat{x}_c$ be the solution to $\min_{x \in \mathbb{R}^m} \|S(Ax - b)\|_p$.
- **Stage 2:** Let $\hat{\rho} = A\hat{x}_c - b$, and unless $\hat{\rho} = 0$ define $q_i = \min\left\{1, \max\left\{p_i, \frac{|\hat{\rho}_i|^p}{\|\hat{\rho}\|_p^p} r_2\right\}\right\}$ with $r_2 = \frac{36^p d^k}{\epsilon^2}\left(d \ln(\frac{36}{\epsilon}) + \ln(200)\right)$.
  - Generate (implicitly, a new) T where $T_{ii} = 1/q_i^{1/p}$ with probability q_i and 0 otherwise.
  - Let $\hat{x}_{\text{OPT}}$ be the solution to $\min_{x \in \mathbb{R}^m} \|T(Ax - b)\|_p$.

Output: $\hat{x}_{\text{OPT}}$ (or $\hat{x}_c$ if only the first stage is run).

Figure 1: Sampling algorithm for ℓ_p regression.

$p \in [1, \infty)$. It is a two-stage algorithm that returns as output a vector $\hat{x}_{\text{OPT}} \in \mathbb{R}^m$ (or a vector $\hat{x}_c \in \mathbb{R}^m$ if only the first stage is run). In either case, the output is the solution to the induced ℓ_p regression subproblem constructed on the randomly sampled constraints.

The algorithm first computes a p-well-conditioned basis U for span(A), as described in the proof of Theorem 3.1. Then, in the first stage, the algorithm uses information from the norms of the rows of U to sample constraints from the input ℓ_p regression problem. In particular, roughly $O(d^{p+1})$ rows of A, and the corresponding elements of b, are randomly sampled according to the probability distribution given by

$$p_i = \min\left\{1, \frac{\|U_{i\star}\|_p^p}{\|U\|_p^p} r_1\right\}, \quad \text{where } r_1 = 8^2 \cdot 36^p d^k \left(d \ln(8 \cdot 36) + \ln(200)\right), \tag{4.6}$$

implicitly represented by a diagonal sampling matrix S, where each $S_{ii} = 1/p_i^{1/p}$. For the remainder of the paper, we will use S to denote the sampling matrix for the first-stage sampling probabilities. The algorithm then solves, using any ℓ_p solver of one's choice, the smaller subproblem. If the solution to the induced subproblem is denoted $\hat{x}_c$, then, as we will see in Theorem 4.1, this is an 8-approximation to the original problem.[5]

In the second stage, the algorithm uses information from the residual of the 8-approximation computed in the first stage to refine the sampling probabilities. Define the residual $\hat{\rho} = A\hat{x}_c - b$ (and note that $\|\hat{\rho}\|_p \leq 8\,\mathcal{Z}$). Then, roughly $O(d^{p+1}/\epsilon^2)$ rows of A, and the corresponding elements of b, are randomly sampled according to the probability distribution

$$q_i = \min\left\{1, \max\left\{p_i, \frac{|\hat{\rho}_i|^p}{\|\hat{\rho}\|_p^p} r_2\right\}\right\}, \quad \text{where } r_2 = \frac{36^p d^k}{\epsilon^2}\left(d \ln(\frac{36}{\epsilon}) + \ln(200)\right). \tag{4.7}$$

As before, this can be represented as a diagonal sampling matrix T, where each $T_{ii} = 1/q_i^{1/p}$ with probability q_i and 0 otherwise. For the remainder of the paper, we will use T to denote the sampling matrix for the second-stage sampling probabilities. Again, the algorithm solves, using any ℓ_p solver of one's choice, the smaller subproblem. If the solution to the induced subproblem at the second stage is denoted $\hat{x}_{\text{OPT}}$, then, as we will see in Theorem 4.1, this is a $(1+\epsilon)$-approximation to the original problem.[6]

The following is our main theorem for the ℓ_p regression algorithm presented in Figure 1.

obtained with a simpler analysis. In particular, one can show that one can obtain a relative error (as opposed to a constant factor) approximation in one stage, if the sampling probabilities are constructed from subspace information in the augmented matrix $[Ab]$ (as opposed to using just subspace information from the matrix A), i.e., by using information in both the data matrix A and the target vector b.

[5] For $p = 2$, Drineas, Mahoney, and Muthhukrishnan show that this first stage actually leads to a $(1+\epsilon)$-approximation. For $p = 1$, Clarkson develops a subgradient-based algorithm and runs it, after preprocessing the input, on all the input constraints to obtain a constant-factor approximation in a stage analogous to our first stage. Here, however, we solve an ℓ_p regression problem on a small subset of the constraints to obtain the constant-factor approximation. Moreover, our procedure works for all $p \in [1, \infty)$.

[6] The subspace-based sampling probabilities (4.6) are similar to those used by Drineas, Mahoney, and Muthukrishnan [13], while the residual-based sampling probabilities (4.7) are similar to those used by Clarkson [10].

THEOREM 4.1. *Let A be an $n \times m$ matrix of rank d, let $b \in \mathbb{R}^n$, and let $p \in [1, \infty)$. Recall that $r_1 = 8^2 \cdot 36^p d^k \left(d \ln(8 \cdot 36) + \ln(200)\right)$ and $r_2 = \frac{36^p d^k}{\epsilon^2}\left(d \ln(\frac{36}{\epsilon}) + \ln(200)\right)$. Then,*

- ***Constant-factor approximation.*** *If only the first stage of the algorithm in Figure 1 is run, then with probability at least 0.6, the solution $\hat{x}_c$ to the sampled problem based on the p_i's of Equation (3.5) is an 8-approximation to the ℓ_p regression problem;*
- ***Relative-error approximation.*** *If both stages of the algorithm are run, then with probability at least 0.5, the solution $\hat{x}_{\text{OPT}}$ to the sampled problem based on the q_i's of Equation (4.7) is a $(1+\epsilon)$-approximation to the ℓ_p regression problem;*
- ***Running time.*** *The i^{th} stage of the algorithm runs in time $O(nmd + nd^5 \log n + \phi(20ir_i, m))$, where $\phi(s,t)$ is the time taken to solve the regression problem $\min_{x \in \mathbb{R}^t} \|A'x - b'\|_p$, where $A' \in \mathbb{R}^{s \times t}$ is of rank d and $b' \in \mathbb{R}^s$.*

Note that since the algorithm of Figure 1 constructs the (α, β, p)-well-conditioned basis U using the procedure in the proof of Theorem 3.1, our sampling complexity depends on α and β. In particular, it will be $O(d(\alpha\beta)^p)$. Thus, if $p < 2$ our sampling complexity is $O(d \cdot d^{\frac{p}{2}+1}) = O(d^{\frac{p}{2}+2})$; if $p > 2$ it is $O(d(d^{\frac{1}{p}+\frac{1}{2}} d^{\frac{1}{q}-\frac{1}{2}})^p) = O(d^{p+1})$; and (although not explicitly stated, our proof will make it clear that) if $p = 2$ it is $O(d^2)$. Note also that we have stated the claims of the theorem as holding with constant probability, but they can be shown to hold with probability at least $1-\delta$ by using standard amplification techniques.

4.2 Proof for first-stage sampling – constant-factor approximation To prove the claims of Theorem 4.1 having to do with the output of the algorithm after the first stage of sampling, we begin with two lemmas. First note that, because of our choice of r_1, we can use the subspace preserving Theorem 3.2 with only a constant distortion, i.e., for all x, we have

$$\frac{7}{8}\|Ax\|_p \leq \|SAx\|_p \leq \frac{9}{8}\|Ax\|_p$$

with probability at least 0.99. The first lemma below now states that the optimal solution to the original problem provides a small (constant-factor) residual when evaluated in the sampled problem.

LEMMA 4.1. *$\|S(Ax_{\text{OPT}} - b)\| \leq 3\mathcal{Z}$, with probability at least $1 - 1/3^p$.*

The next lemma states that if the solution to the sampled problem provides a constant-factor approximation (when evaluated in the sampled problem), then when this solution is evaluated in the original regression problem we get a (slightly weaker) constant-factor approximation.

LEMMA 4.2. *If $\|S(A\hat{x}_c - b)\| \leq 3\,\mathcal{Z}$, then $\|A\hat{x}_c - b\| \leq 8\,\mathcal{Z}$.*

Clearly, $\|S(A\hat{x}_c - b)\| \leq \|S(Ax_{\text{OPT}} - b)\|$ (since $\hat{x}_c$ is an optimum for the sampled ℓ_p regression problem). Combining this with Lemmas 4.1 and 4.2, it follows that the solution $\hat{x}_c$ to the the sampled problem based on the p_i's of Equation (3.5) satisfies $\|A\hat{x}_c - b\| \leq 8\,\mathcal{Z}$, i.e., $\hat{x}_c$ is an 8-approximation to the original $\mathcal{Z}$.

To conclude the proof of the claims for the first stage of sampling, note that by our choice of r_1, Theorem 3.2 fails to hold for our first stage sampling with probability no greater than 1/100. In addition, Lemma 4.1 fails to hold with probability no grater than $1/3^p$, which is no greater than 1/3 for all $p \in [1, \infty)$. Finally, let $\widehat{r_1}$ be a random variable representing the number of rows actually chosen by our sampling schema, and note that $E[\widehat{r_1}] \leq r_1$. By Markov's inequality, it follows that $\widehat{r_1} > 20r_1$ with probability less than 1/20. Thus, the first stage of our algorithm fails to give an 8-approximation in the specified running time with a probability bounded by $1/3 + 1/20 + 1/100 < 2/5$.

4.3 Proof for second-stage sampling – relative-error approximation The proof of the claims of Theorem 4.1 having to do with the output of the algorithm after the second stage of sampling will parallel that for the first stage, but it will have several technical complexities that arise since the first triangle inequality approximation in the proof of Lemma 4.2 is too coarse for relative-error approximation. By our choice of r_2 again, we have a finer result for subspace preservation. Thus, with probability 0.99, the following holds for all x

$$(1-\epsilon)\|Ax\|_p \leq \|SAx\|_p \leq (1+\epsilon)\|Ax\|_p$$

As before, we start with a lemma that states that the optimal solution to the original problem provides a small (now a relative-error) residual when evaluated in the sampled problem. This is the analog of Lemma 4.1. An important difference is that the second stage sampling probabilities significantly enhance the probability of success.

LEMMA 4.3. *$\|T(Ax_{\text{OPT}} - b)\| \leq (1+\epsilon)\mathcal{Z}$, with probability at least 0.99.*

Next we show that if the solution to the sampled problem provides a relative-error approximation (when

evaluated in the sampled problem), then when this solution is evaluated in the original regression problem we get a (slightly weaker) relative-error approximation. We first establish two technical lemmas.

The following lemma says that for all optimal solutions $\hat{x}_{\text{OPT}}$ to the second-stage sampled problem, $A\hat{x}_{\text{OPT}}$ is not too far from $A\hat{x}_c$, where $\hat{x}_c$ is the optimal solution from the first stage, in a p-norm sense. Hence, the lemma will allow us to restrict our calculations in Lemmas 4.5 and 4.6 to the ball of radius $12\,\mathcal{Z}$ centered at $A\hat{x}_c$.

LEMMA 4.4. $\|A\hat{x}_{\text{OPT}} - A\hat{x}_c\| \leq 12\,\mathcal{Z}$.

Thus, if we define the affine ball of radius $12\,\mathcal{Z}$ that is centered at $A\hat{x}_c$ and that lies in span(A),

(4.8)
$$B = \{y \in \mathbb{R}^n : y = Ax, x \in \mathbb{R}^m, \|A\hat{x}_c - y\| \leq 12\,\mathcal{Z}\}\ ,$$

then Lemma 4.4 states that $A\hat{x}_{\text{OPT}} \in B$, for all optimal solutions $\hat{x}_{\text{OPT}}$ to the sampled problem. Let us consider an ε-net, call it B_ε, with $\varepsilon = \epsilon\,\mathcal{Z}$, for this ball B. Using standard arguments, the size of the ε-net is $\left(\frac{3\cdot 12\,\mathcal{Z}}{\epsilon\,\mathcal{Z}}\right)^d = \left(\frac{36}{\epsilon}\right)^d$. The next lemma states that for all points in the ε-net, if that point provides a relative-error approximation (when evaluated in the sampled problem), then when this point is evaluated in the original regression problem we get a (slightly weaker) relative-error approximation.

LEMMA 4.5. *For all points Ax_ε in the ε-net, B_ε, if $\|T(Ax_\varepsilon - b)\| \leq (1+3\epsilon)\mathcal{Z}$, then $\|Ax_\varepsilon - b\| \leq (1+6\epsilon)\mathcal{Z}$, with probability* 0.99.

Finally, the next lemma states that if the solution to the sampled problem (in the second stage of sampling) provides a relative-error approximation (when evaluated in the sampled problem), then when this solution is evaluated in the original regression problem we get a (slightly weaker) relative-error approximation. This is the analog of Lemma 4.2, and its proof will use Lemma 4.5.

LEMMA 4.6. *If $\|T(A\hat{x}_{\text{OPT}} - b)\| \leq (1+\epsilon)\mathcal{Z}$, then $\|A\hat{x}_{\text{OPT}} - b\| \leq (1+7\epsilon)\mathcal{Z}$.*

Clearly, $\|T(A\hat{x}_{\text{OPT}} - b)\| \leq \|T(Ax_{\text{OPT}} - b)\|$, since $\hat{x}_{\text{OPT}}$ is an optimum for the sampled ℓ_p regression problem. Combining this with Lemmas 4.3 and 4.6, it follows that the solution $\hat{x}_{\text{OPT}}$ to the the sampled problem based on the q_i's of Equation (4.7) satisfies $\|A\hat{x}_{\text{OPT}} - b\| \leq (1+\epsilon)\,\mathcal{Z}$, i.e., $\hat{x}_{\text{OPT}}$ is a $(1+\epsilon)$-approximation to the original $\mathcal{Z}$.

To conclude the proof of the claims for the second stage of sampling, recall that the first stage failed with probability no greater than 2/5. Note also that by our choice of r_2, Theorem 3.2 fails to hold for our second stage sampling with probability no greater than 1/100. In addition, Lemma 4.3 and Lemma 4.5 each fails to hold with probability no greater than 1/100. Finally, let $\widehat{r_2}$ be a random variable representing the number of rows actually chosen by our sampling schema in the second stage, and note that $E[\widehat{r_2}] \leq 2r_2$. By Markov's inequality, it follows that $\widehat{r_2} > 40r_2$ with probability less than 1/20. Thus, the second stage of our algorithm fails with probability less than $1/20 + 1/100 + 1/100 + 1/100 < 1/10$. By combining both stages, our algorithm fails to give a $(1+\epsilon)$-approximation in the specified running time with a probability bounded from above by $2/5 + 1/10 = 1/2$.

5 Extensions

In this section we outline several immediate extensions of our main algorithmic result.

5.1 Constrained ℓ_p regression Our sampling strategies are transparent to constraints placed on x. In particular, suppose we constrain the output of our algorithm to lie within a convex set $\mathcal{C} \subseteq \mathbb{R}^m$. If there is an algorithm to solve the constrained ℓ_p regression problem $\min_{z\in\mathcal{C}} \|A'x - b'\|$, where $A' \in \mathbb{R}^{s\times m}$ is of rank d and $b' \in \mathbb{R}^s$, in time $\phi(s,m)$, then by modifying our main algorithm in a straightforward manner, we can obtain an algorithm that gives a $(1+\epsilon)$-approximation to the constrained ℓ_p regression problem in time $O(nmd + nd^5 \log n + \phi(40r_2, m))$.

5.2 Generalized ℓ_p regression Our sampling strategies extend to the case of generalized ℓ_p regression: given as input a matrix $A \in \mathbb{R}^{n\times m}$ of rank d, a target matrix $B \in \mathbb{R}^{n\times p}$, and a real number $p \in [1,\infty)$, find a matrix $X \in \mathbb{R}^{m\times p}$ such that $\|\|AX - B\|\|_p$ is minimized. To do so, we generalize our sampling strategies in a straightforward manner. The probabilities p_i for the first stage of sampling are the same as before. Then, if $\hat{X}_c$ is the solution to the first-stage sampled problem, we can define the $n \times p$ matrix $\hat{\rho} = A\hat{X}_c - B$, and define the second stage sampling probabilities to be $q_i = \min\left(1, \max\{p_i, r_2\|\hat{\rho}_{i\star}\|_p^p / \|\|\hat{\rho}\|\|_p^p\}\right)$. Then, we can show that the $\hat{X}_{\text{OPT}}$ computed from the second-stage sampled problem satisfies $\|\|A\hat{X}_{\text{OPT}} - B\|\|_p \leq (1+\epsilon)\min_{X\in\mathbb{R}^{m\times p}} \|\|AX - B\|\|_p$, with probability at least 1/2.

5.3 Weighted ℓ_p regression Our sampling strategies also generalize to the case of ℓ_p regression involving weighted p-norms: if $w_1, \ldots, w_m$ are a set of non-negative weights then the weighted p-norm of a vector

$x \in \mathbb{R}^m$ may be defined as $\|x\|_{p,w} = \left(\sum_{i=1}^m w_i |x_i|^p\right)^{1/p}$, and the weighted analog of the matrix p-norm $|||\cdot|||_p$ may be defined as $|||U|||_{p,w} = \left(\sum_{j=1}^d \|U_{\star j}\|_{p,w}\right)^{1/p}$. Our sampling schema proceeds as before. First, we compute a "well-conditioned" basis U for $\mathrm{span}(A)$ with respect to this weighted p-norm. The sampling probabilities p_i for the first stage of the algorithm are then $p_i = \min\left(1, r_1 w_i \|U_{i\star}\|_p^p / |||U|||_{p,w}^p\right)$, and the sampling probabilities q_i for the second stage are $q_i = \min\left(1, \max\{p_i, r_2 w_i |\hat{\rho}_i|^p / \|\hat{\rho}\|_{p,w}^p\}\right)$, where $\hat{\rho}$ is the residual from the first stage.

5.4 General sampling probabilities More generally, consider *any* sampling probabilities of the form: $p_i \geq \min\left\{1, \max\left\{\frac{\|U_{i\star}\|_p^p}{|||U|||_p^p}, \frac{|(\rho_{\mathrm{OPT}})_i|^p}{\mathcal{Z}^p}\right\} r\right\}$, where $\rho_{\mathrm{OPT}} = Ax_{\mathrm{OPT}} - b$ and $r \geq \frac{36^p d^k}{\epsilon^2}\left(d\ln(\frac{36}{\epsilon}) + \ln(200)\right)$ and where we adopt the convention that $\frac{0}{0} = 0$. Then, by an analysis similar to that presented for our two stage algorithm, we can show that, by picking $O(36^p d^{p+1}/\epsilon^2)$ rows of A and the corresponding elements of b (in a single stage of sampling) according to these probabilities, the solution $\hat{x}_{\mathrm{OPT}}$ to the sampled ℓ_p regression problem is a $(1+\epsilon)$-approximation to the original problem, with probability at least $1/2$. (Note that these sampling probabilities, if an equality is used in this expression, depend on the entries of the vector $\rho_{\mathrm{OPT}} = Ax_{\mathrm{OPT}} - b$; in particular, they require the solution of the original problem. This is reminiscent of the results of [13]. Our main two-stage algorithm shows that by solving a problem in the first stage based on coarse probabilities, we can refine our probabilities to approximate these probabilities and thus obtain an $(1+\epsilon)$-approximation to the ℓ_p regression problem more efficiently.)

Acknowledgments. We would like to thank Robert Kleinberg for pointing out several useful references. We would also like to acknowledge an anonymous reviewer who pointed out that we can can obtain a relative error approximation in one stage, if we are willing to look at the target vector b and use information in the augmented matrix $[Ab]$ to construct sampling probabilities.

References

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

[2] P. K. Agarwal, S. Har-Peled, and K. R. Varadarajan. Geometric approximation via coresets. In J. E. Goodman, J. Pach, and E. Welzl, editors, *Combinatorial and Computational Geometry*, volume 52, pages 1–30. Cambridge University Press, 2005.

[3] N. Ailon and B. Chazelle. Approximate nearest neighbors and the fast Johnson–Lindenstrauss transform. In *Proceedings of the 38th Annual ACM Symposium on Theory of Computing*, pages 557–563, 2006.

[4] H. Auerbach. *On the Area of Convex Curves with Conjugate Diameters (in Polish)*. PhD thesis, University of Lwów, 1930.

[5] B. Awerbuch and R. D. Kleinberg. Adaptive routing with end-to-end feedback: Distributed learning and geometric approaches. In *Proceedings of the 36th Annual ACM Symposium on Theory of Computing*, pages 45–53, 2004.

[6] M. Bädoiu and K. L. Clarkson. Smaller core-sets for balls. In *SODA '03: Proceedings of the fourteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 801–802, Philadelphia, PA, USA, 2003. Society for Industrial and Applied Mathematics.

[7] J. Bourgain, J. Lindenstrauss, and V. Milman. Approximation of zonoids by zonotopes. *Acta Math.*, 162:73–141, 1989.

[8] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.

[9] S. Chatterjee, A. S. Hadi, and B. Price. *Regression Analysis by Example*. Wiley Series in Probability and Statistics, 2000.

[10] K. L. Clarkson. Subgradient and sampling algorithms for ℓ_1 regression. In *Proceedings of the 16th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 257–266, 2005.

[11] A. Dasgupta, P. Drineas, B. Harb, R. Kumar, and M. W. Mahoney. Sampling algorithms and coresets for ℓ_p regression. Technical report. Preprint: arXiv:0707.1714v1 (2007).

[12] M. Day. Polygons circumscribed about closed convex curves. *Transactions of the American Mathematical Society*, 62:315–319, 1947.

[13] P. Drineas, M. W. Mahoney, and S. Muthukrishnan. Sampling algorithms for ℓ_2 regression and applications. In *Proceedings of the 17th Annual ACM-SIAM Symposium on Discrete Algorithm*, pages 1127–1136, 2006.

[14] D. Feldman, A. Fiat, and M. Sharir. Coresets for weighted facilities and their applications. In *Proceedings of the 47th Annual IEEE Symposium on Foundations of Computer Science (FOCS'06)*, pages 315–324. IEEE Computer Society, 2006.

[15] S. Har-Peled and S. Mazumdar. On coresets for k-means and k-median clustering. In *STOC '04: Proceedings of the thirty-sixth annual ACM symposium on Theory of computing*, pages 291–300, New York, NY, USA, 2004. ACM Press.

[16] T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning Learning*. Springer, 2003.

[17] J. Kleinberg and M. Sandler. Using mixture models for collaborative filtering. In *Proceedings of the 36th Annual ACM Symposium on Theory of Computing*, pages 569–578, 2004.

[18] T. Sarlós. Improved approximation algorithms for

large matrices via random projections. In *Proceedings of the 47th Annual IEEE Symposium on Foundations of Computer Science (FOCS'06)*, pages 143–152. IEEE Computer Society, 2006.

[19] G. Schechtman. More on embedding subspaces of L_p in ℓ_r^n. *Compositio Math*, 61(2):159–169, 1987.

[20] G. Schechtman and A. Zvavitch. Embedding subspaces of L_p into ℓ_p^N, $0 < p < 1$. *Math. Nachr.*, 227:133–142, 2001.

[21] M. Talagrand. Embedding subspaces of L_1 into ℓ_1^N. *Proceedings of the American Mathematical Society*, 108(2):363–369, February 1990.

[22] M. Talagrand. Embedding subspaces of L_p into ℓ_p^N. *Oper. Theory Adv. Appl.*, 77:311–325, 1995.

[23] A. Taylor. A geometric theorem and its application to biorthogonal systems. *Bulletin of the American Mathematical Society*, 53:614–616, 1947.

[24] P. Wojtaszczyk. *Banach Spaces for Analysts*, volume 25 of *Cambridge Studies in Advanced Mathematics*. Cambridge University Press, 1991.