Sampling algorithms and core-sets for L_p regression

Petros Drineas

Rensselaer Polytechnic Institute
Computer Science Department

For papers, etc.

Google drineas

Problem definition and motivation

In many applications (e.g., statistical data analysis and scientific computation), one has n observations of the form:

$$y_i = y(t_i), i = 1, \ldots, n$$

Model y(t) (unknown) as a linear combination of d basis functions:

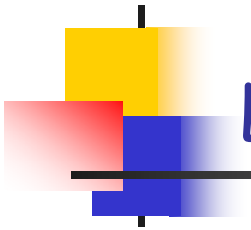
$$y(t) \approx x_1\phi_1(t) + \cdots + x_d\phi_d(t)$$

A is an n x d "design matrix" (n >> d):

$$A_{ij} = \phi_j(t_i)$$

In matrix-vector notation,

$$y \approx Ax$$



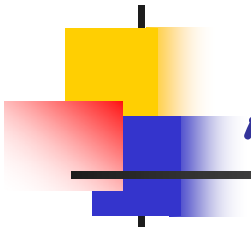
Least-norm approximation problems

Recall a linear measurement model:

$$y = Ax + \varepsilon \quad \begin{cases} y \text{ are the measurements} \\ x \text{ is the unknown} \\ \varepsilon \text{ is an error process} \end{cases}$$

In order to estimate x, solve:

$$\widehat{x} = \arg\min \|y - Ax\|$$



Application: (all of) data analysis in science

- First application: Astronomy

  Predicting the orbit of the asteroid Ceres (in 1801!).

  Gauss (1809) -- see also Legendre (1805) and Adrain (1808).

  First application of "least squares optimization" and runs in $O(nd^2)$ time!

- Data analysis: Fit parameters of a biological, chemical, economical, physical (astronomical), social, internet, etc. model to experimental data.

Norms of common interest

Let y = b and define the residual: $r = Ax - b \in R^n$

Least-squares approximation:

$$\text{minimize: } ||Ax - b||_2^2 = r_1^2 + r_2^2 + \cdots + r_n^2$$

Chebyshev or mini-max approximation:

$$\text{minimize: } ||Ax - b||_\infty = \max\{|r_1|, \ldots, |r_n|\}$$

Sum of absolute residuals approximation:

$$\text{minimize: } ||Ax - b||_1 = |r_1| + |r_2| + \cdots + |r_n|$$

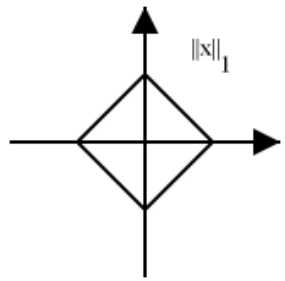
L_p norms and their unit balls

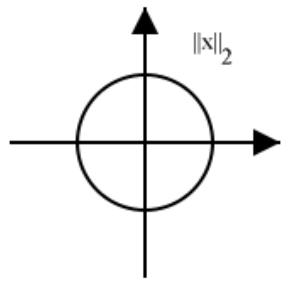
Recall the L_p norm for $z \in R^n$:

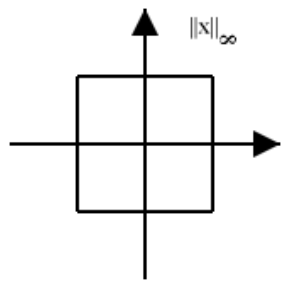
$$||z||_p = \left(\sum_{i=1}^{n} |z_i|^p \right)^{1/p} \quad , p \in [1, \infty)$$

$$||z||_\infty = \max_i |z_i|$$

$$||z||_2^2 = \sum_i z_i^2 = z^T z$$





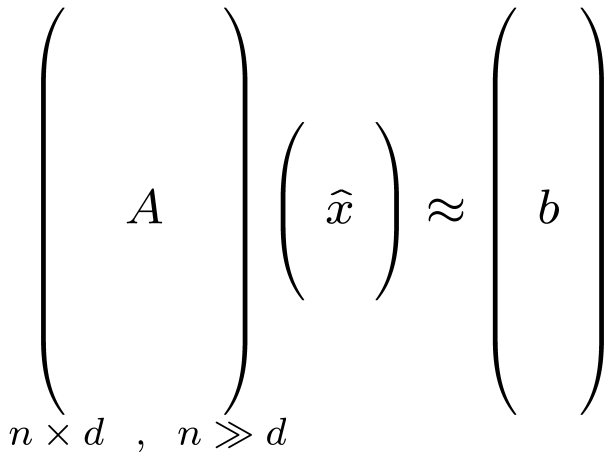


Some inequality relationships include:

$$\frac{1}{\sqrt{n}}||z||_2 \leq ||z||_\infty \leq ||z||_2 \leq ||z||_1 \leq \sqrt{n}||z||_2$$

L_p regression problems

$$\begin{aligned} \mathcal{Z}_p &= \min_{x \in R^d} ||b - Ax||_p \\ &= ||b - A\hat{x}||_p \end{aligned}$$



$$\underset{n \times d \quad , \quad n \gg d}{\begin{pmatrix} A \end{pmatrix}} \begin{pmatrix} \hat{x} \end{pmatrix} \approx \begin{pmatrix} b \end{pmatrix}$$

We are interested in over-constrained L_p regression problems, $n \gg d$.

- Typically, there is no x such that $Ax = b$.
- Want to find the "best" x such that $Ax \approx b$.

L_p regression problems are convex programs (or better).

- There exist poly-time algorithms.
- We want to solve them faster.

Singular Value Decomposition (SVD)

$$\underset{n \times d}{\begin{pmatrix} \\ A \\ \\ \end{pmatrix}} = \underset{n \times \rho}{\begin{pmatrix} \\ U \\ \\ \end{pmatrix}} \cdot \underset{\rho \times \rho}{\begin{pmatrix} \\ \Sigma \\ \\ \end{pmatrix}} \cdot \underset{\rho \times d}{\begin{pmatrix} \\ V \\ \\ \end{pmatrix}^T}$$

ρ: rank of A

U (V): orthogonal matrix containing the left (right) singular vectors of A.

Σ: diagonal matrix containing the singular values of A.

Exact computation of the SVD takes $O(nd^2)$ time (assuming $n > d$).

Exact solution to L_2 regression

Cholesky Decomposition:

If A is full rank and well-conditioned,

decompose $A^TA = R^TR$, where R is upper triangular, and

solve the normal equations: $R^TRx = A^Tb$.

$$\begin{aligned} \mathcal{Z}_2 &= \min_{x \in R^d} ||b - Ax||_2 \\ &= ||b - A\hat{x}||_2 \end{aligned}$$

QR Decomposition:

Slower but numerically stable, esp. if A is rank-deficient.

Write $A = QR$, and solve $Rx = Q^Tb$.

Projection of b on the subspace spanned by the columns of A

$$\begin{aligned} \mathcal{Z}_2^2 &= ||b||_2^2 - ||AA^+b||_2^2 \\ \hat{x} &= A^+b \end{aligned}$$

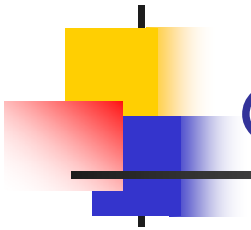
Pseudoinverse of A

Singular Value Decomposition:

Most expensive, but best if A is very ill-conditioned.

Write $A = U\Sigma V^T$, in which case: $\mathbf{x_{OPT} = A^+b = V\Sigma^{-1}U^Tb}$.

Complexity is $O(nd^2)$, but constant factors differ.



Questions …

$$\mathcal{Z}_p = \min_{x \in R^d} ||b - Ax||_p = ||b - A\hat{x}||_p$$

Approximation algorithms:

Can we approximately solve L_p regression faster than "exact" methods?

Core-sets (or induced sub-problems):

Can we find a small set of constraints such that solving the L_p regression on those constraints gives an approximation to the original problem?

Overview of four L_p regression algorithms

$$\mathcal{Z}_p = \min_{x \in R^d} ||b - Ax||_p = ||b - A\hat{x}||_p$$

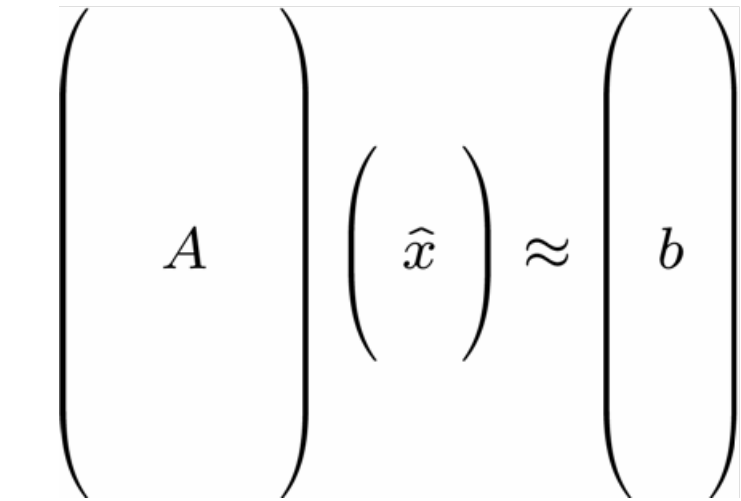
Alg. 1	p=2	Sampling (core-set)	(1+ε)-approx	$O(nd^2)$	D., Mahoney, & Muthukrishnan '06 D., Mahoney, & Muthukrishnan '07
Alg. 2	p=2	Projection (no core-set)	(1+ε)-approx	$O(nd^2)$	"obvious"
Alg. 3	p=2	Projection (no core-set)	(1+ε)-approx	$o(nd^2)$	Sarlos '06 D., Mahoney, & Sarlos '07
Alg. 4	p ε [1,∞)	Sampling (core-set)	(1+ε)-approx	$O(nd^5)$ +o("exact")	DasGupta, D., Harb, Kumar, & Mahoney '07

Note: Clarkson '05 gets a (1+ε)-approximation for L1 regression in $O^*(d^{3.5}/\varepsilon^4)$ time.

He *preprocessed* [A,b] to make it "well-rounded" or "well-conditioned" and *then sampled*.

Algorithm 1: Sampling for L2 regression

$$\mathcal{Z}_2 = \min_{x \in \mathbb{R}^d} \|b - Ax\|_2 = \|b - A\hat{x}\|_2$$



$n \times d, \quad n >> d$

Algorithm

1. Fix a set of probabilities p_i, i=1…n, summing up to 1.
2. Pick the i-th row of A and the i-th element of b with probability

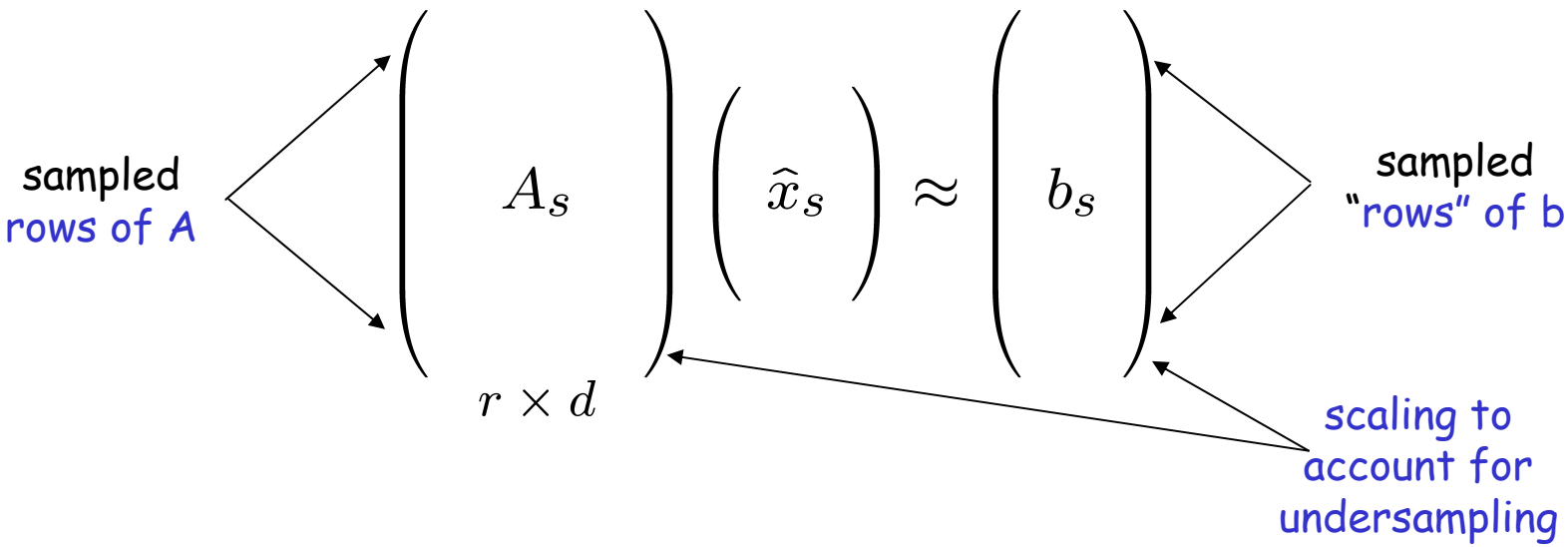
   $$\min\{1, rp_i\},$$

   and rescale both by $(1/\min\{1,rp_i\})^{1/2}$.
3. Solve the induced problem.

Note: in expectation, at most r rows of A and r elements of b are kept.

Random sampling algorithm for L2 regression

$$\mathcal{Z}_{2,s} = \min_{x \in \mathbb{R}^d} \|b_s - A_s x\|_2 = \|b_s - A_s \hat{x}_s\|_2$$



$$\left|\mathcal{Z}_2 - \mathcal{Z}_{2,s}\right| \leq ? \qquad \|\hat{x} - \hat{x}_s\|_2 \leq ? \qquad \|A\hat{x}_s - b\|_2 \leq ?$$

Our results for p=2

If the p_i satisfy a condition, then with probability at least 1-δ,

$$\mathcal{Z}_{2,s} \leq (1+\epsilon)\,\mathcal{Z}_2$$

$$\mathcal{Z}_2 \leq \|A\widehat{x}_s - b\|_2 \leq (1+\epsilon)\,\mathcal{Z}_2$$

$$\|\widehat{x} - \widehat{x}_s\|_2 \leq \epsilon \left(\frac{\kappa(A)}{\gamma}\right) \|\widehat{x}\|_2$$

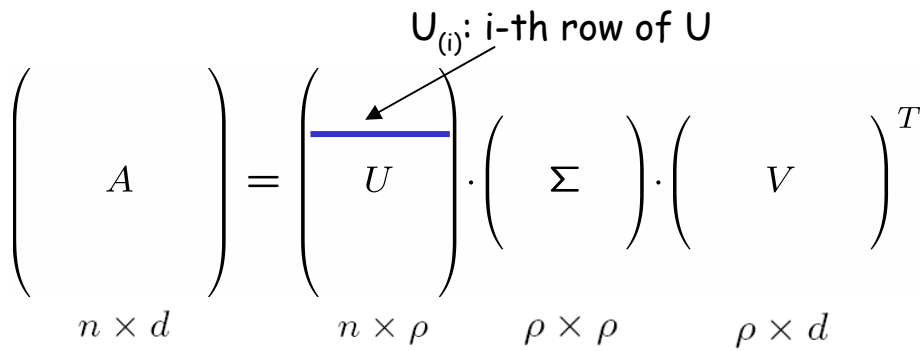
$\kappa(A)$: condition number of A

$$\gamma = \left\|AA^{+}b\right\|_2 / \|b\|_2$$

The sampling complexity is

$$r = O(d\log(d)\log(1/\delta)/\epsilon^2)$$

Notation



ρ: rank of A

U: orthogonal matrix containing the left singular vectors of A.

Condition on the probabilities

The condition that the p_i must satisfy is, for some $\beta \in (0,1]$:

lengths of **rows** of matrix of left singular vectors of *A*

$$p_i \geq \frac{\beta \left\| U_{(i)} \right\|_2^2}{\sum_{i=1}^{n} \left\| U_{(i)} \right\|_2^2} = \frac{\beta \left\| U_{(i)} \right\|_2^2}{d}$$

Notes:

- Using the norms of the rows of *any orthonormal basis suffices*, e.g., Q from QR.
- *$O(nd^2)$ time suffices* (to compute probabilities and to construct a core-set).
- Important question:

  Is $O(nd^2)$ necessary? Can we compute the p_i's, or construct a core-set, faster?

- Original conditions (D, Mahoney, & Muthukrishnan '06) were stronger and more complicated.

Condition on the probabilities, cont'd

- **Important**: Sampling process must NOT loose any rank of A.

  (Since pseudoinverse will amplify that error!)

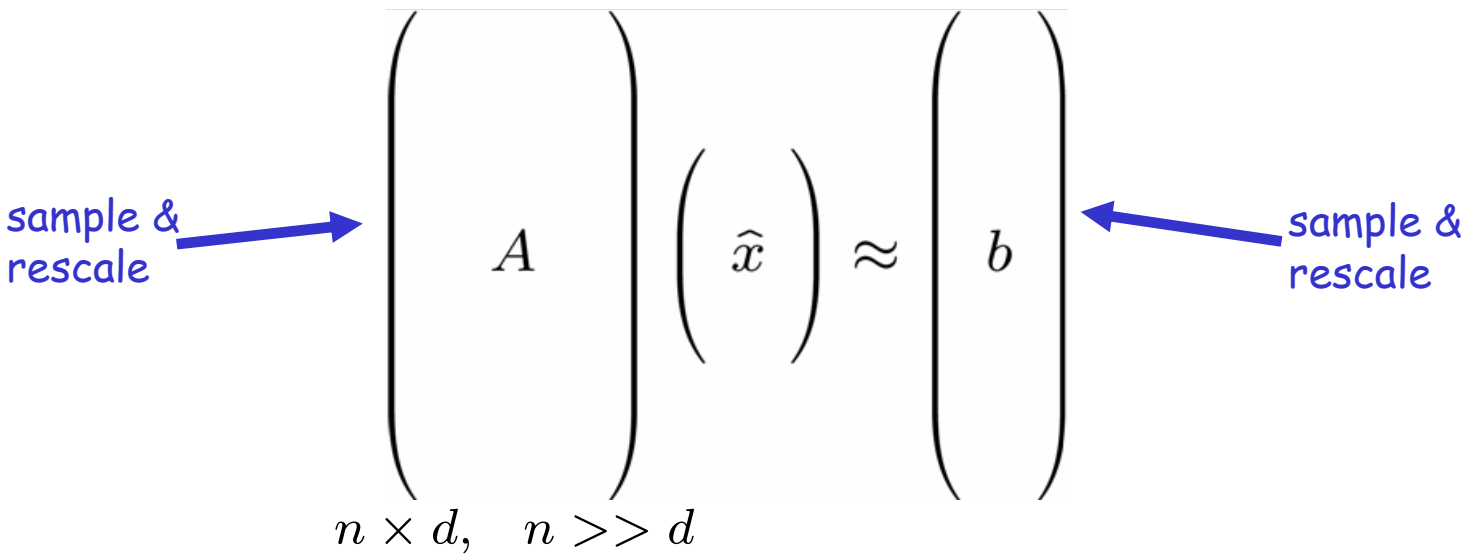
$$A\hat{x} \approx b \;\rightarrow\; \hat{x} = A^{+}b$$
$$SA\hat{x}_s \approx Sb \;\rightarrow\; \hat{x}_s = (SA)^{+}Sb$$

Notation:

- S is an r $\times$ n matrix that samples and rescales a small number of rows of A and elements of b.
- Each row of S has exactly one non-zero element (corresponding to the selected row of A); this non-zero element is set to the "rescaling value."

Critical observation

$$\mathcal{Z}_2 \;=\; \min_{x\in\mathbb{R}^d} \|b - Ax\|_2 = \|b - A\widehat{x}\|_2$$



Recall: $A = U_A \Sigma_A V_A^T$

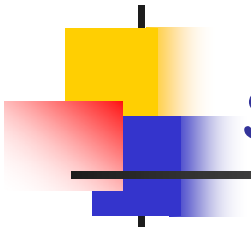
Critical observation, cont'd

$$
\begin{aligned}
SA\hat{x}_s \approx Sb \rightarrow \hat{x}_s &= (SA)^{+} Sb \\
&= \left(SU_A \Sigma_A V_A^T\right)^{+} b \\
&= V_A \left(SU_A \Sigma_A\right)^{+} b
\end{aligned}
$$

SU_A is approx. orthogonal

The approx. orthogonality of SU_A allows us to prove that:

- The left singular vectors of SA are (approx.) SU_A.
- The singular values of SA are (approx.) equal to the singular values of A.
- A corollary of the above is that SA is full rank.



Sampling rows from orthogonal matrices

An old question:

Given an orthogonal matrix, sample a subset of its rows uniformly at random and argue that the resulting matrix is almost orthogonal.

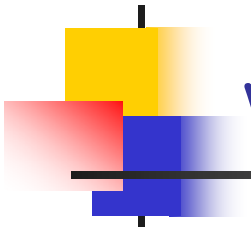
Important question in functional analysis (see, e.g., early work of Kashin and Tzafriri)

What if we are allowed to keep non-uniform samples of the rows of an orthogonal matrix (scaled appropriately)?

Then, in our case (n >> d), we can prove that:

$$\left\| U_A^T S^T S U_A - I \right\|_2 \leq \epsilon \qquad r = O(d \log d / \epsilon^2)$$

(Similar arguments in Frieze, Kannan, and Vempala '98, D. and Kannan '01, D. Kannan, and Mahoney '06, Rudelson and Virshynin '04).



What made the L_2 result work?

The L_2 sampling algorithm worked because:

- For p=2, an orthogonal basis (from SVD, QR, etc.) is a "good" or "well-conditioned" basis.

  (This came for free, since orthogonal bases are the obvious choice.)

- Sampling w.r.t. the "good" basis allowed us to perform "subspace-preserving sampling."

  (This allowed us to preserve the rank of the matrix.)

Can we generalize these two ideas to $p \neq 2$?

p-well-conditioned basis (definition)

Let p be in [1,∞), and let q be its dual (thus, 1/p + 1/q = 1).

Definition: An n x d matrix U is an *(α,β,p)-well-conditioned basis* for the subspace spanned by the columns of A if:

$$\left(\sum_{i,j}\left|U_{ij}\right|^p\right)^{1/p} = |||U|||_p \leq \alpha,$$

$$\forall z \in \mathbb{R}^d, \|z\|_q \leq \beta\|Uz\|_p.$$

(Obviously, U also has to be a basis for the span of A.) U is a *p-well-conditioned basis* if α, $\beta = d^{O(1)}$, independent of n.

Our definition is motivated by Clarkson '05 (L_1 case).

p-well-conditioned basis (existence)

Let A be an $n \times d$ matrix and let p be in $[1,\infty)$, and let q be its dual.

Theorem: There exists an *(α,β,p)-well-conditioned basis* U for span(A) s.t.:

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

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

(The theorem can be generalized to $n \times m$ matrices A of rank $d << m,n$.)

p-well-conditioned basis (construction)

Algorithm:

• Let A=QR be any QR decomposition of A (stop if p=2).

Thus, Q is an orthogonal basis for the subspace spanned by the columns of A.

• (Q is an n $\times$ d matrix.) Let

$$\mathcal{C} = \left\{ z \in \mathbb{R}^d : \|Qz\|_p \leq 1 \right\}$$

• Compute ε_{LJ} , the Lowner-John ellipsoid of C, which satisfies:

$$\left(\frac{1}{\sqrt{d}}\right) \mathcal{E}_{LJ} \subseteq \mathcal{C} \subseteq \mathcal{E}_{LJ}$$

ε_{LJ} is of the form:

$$\mathcal{E}_{LJ} = \left\{ z \in \mathbb{R}^d : z^T F z \leq 1 \right\}, \qquad F = GG^T$$

F is SPD and full rank, hence G is invertible.

p-well-conditioned basis (construction, cont'd)

Algorithm:

- Let A=QR be any QR decomposition of A (stop if p=2).
- (Q is an n $\times$ d matrix.) Let

$$\mathcal{C} = \left\{ z \in \mathbb{R}^d : \|Qz\|_p \leq 1 \right\}$$

- Compute ε_{LJ} , the Lowner-John ellipsoid of C:

$$\mathcal{E}_{LJ} = \left\{ z \in \mathbb{R}^d : z^T F z \leq 1 \right\}, \qquad F = GG^T$$

- Return U = Q^*G^{-1} as the p-well-conditioned basis.

(Similar preprocessing of the matrix A was done in Clarkson '05 in order to get a matrix A' on which to apply gradient-descent and sampling algorithms.)

Subspace-preserving sampling

Let A be an n x d matrix and let p be in [1,∞).

Let U be an *(α,β,p)-well-conditioned basis* for span(A),

Basic lemma: Randomly sample rows of A where the i-th row is sampled with probability

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

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- δ, the following holds for all vectors x in R^d:

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

Sampling matrix S

Notation: S is an r $\times$ n matrix that samples and rescales rows of A.

Each row of S has exactly one non-zero element (corresponding to the selected row of A); this non-zero element is set to $p_i^{1/p}$.

Basic lemma: Randomly sample rows of A where the i-th row is sampled with probability

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

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- δ, the following holds for all vectors x in R^d:

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

Algorithm 4a: constant-factor L_p regression

Input: An n x d matrix A, a vector b in R^n, and p ε [1,∞).
Output: x', such that Ax' $\approx$ b in the p-norm.

- Find a *p-well-conditioned* basis U for span(A).
- Using the p_i's that we defined, create the sampling and rescaling matrix S.
- Solve: $\min_x \left(\|S\,(Ax-b)\|_p = \|S\,(Ax'-b)\|_p\right.$

Theorem: With probability at least .6,

$$\|Ax'-b\|_p \leq 8 \min_x \|Ax-b\|_p$$

Notes : this is a constant factor approximation
: r may be chosen to be smaller than the value of the Basic Lemma.

Algorithm 4b: relative-error L_p regression

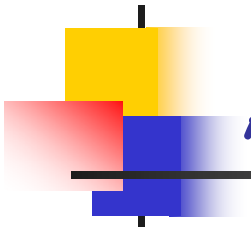
(this approach was suggested by an anonymous reviewer of a draft of this paper)

Input: An n x d matrix A, a vector b in R^n, and p ε [1,∞).
Output: x", such that Ax" $\approx$ b (w.r.t the p-norm)

- Form the n-by-(d+1) matrix C = [A -b].
- Compute U_C, the well-conditioned basis for C.
- Form the sampling matrix T, using probabilities that depend on the row-lengths of U_c.
- Solve

$$\min_x \left(\|T\,(Ax-b)\|_p = \|T\,(Ax''-b)\|_p\right.$$

Note: T is constructed in exactly the same way as S.



Algorithm 4b: relative-error theorem

Theorem: With probability at least .5,

$$\|Ax'' - b\|_p \leq (1+\epsilon)\min_x \|Ax - b\|_p$$

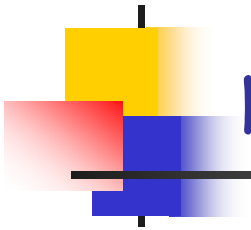
Note: this is a relative-error approximation.

Running time: constructing the p-well-conditioned basis takes

$O(nd^2 + nd^5 \log n)$

time.

We also need to solve induced L_p optimization problems, whose sizes are independent of n (but depend polynomially on d and exponentially on p).



Notes

- Notice that for the relative error approximation we need to know the right hand side vector b.
- This was not necessary in the l_2 regression case.
- Our constant-factor algorithm is much simpler than Clarkson '05, but is randomized instead of deterministic.
- Our approach generalizes/unifies the L_1, L_2 results to general p in $[1,\infty)$.
- **Extensions:** algorithm and analysis extend to:

  Constrained L_p regression: if we have oracle for constrained subproblem.

  Generalized L_p regression: if right hand side b is a matrix B.

  Weighted L_p regression: if we use a weighted L_p norm.