Quantile Regression for Large-scale Applications

Jiyan Yang

Stanford University

June 19, 2013

International Conference on Machine Learning, 2013
Joint work with Xiangrui Meng and Michael Mahoney

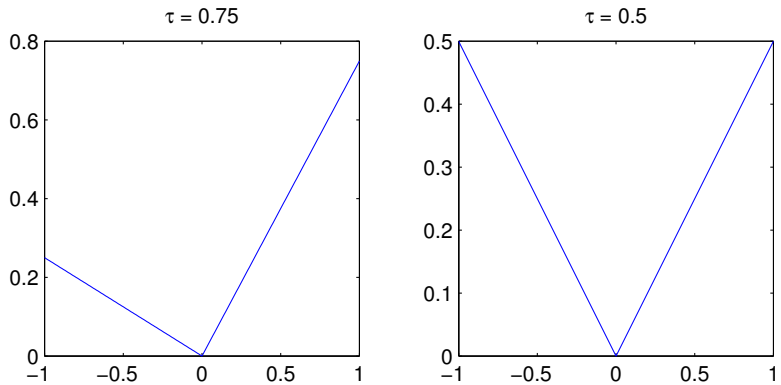
1. Overview to quantile regression
2. Technical ingredients
   - Important notions
   - Sampling lemma
   - Conditioning
   - Estimating row norms
3. Main algorithm
4. Empirical evaluation
   - Medium-scale Empirical evaluation
   - Large-scale Empirical evaluation
5. Conclusion

What is quantile regression?

- Quantile regression is a method to estimate the quantiles of the conditional distribution of response.
- Quantile regression involves minimizing asymmetrically weighted absolute residuals:

$$\rho_\tau(z) = \begin{cases} \tau z, & z \geq 0; \\ (\tau - 1)z, & z < 0. \end{cases}$$

- ℓ_1 regression is a special case of quantile regression with $\tau = 0.5$.



Formulation of quantile regression

- Given matrix $A \in \mathbb{R}^{n \times d}$, a vector $b \in \mathbb{R}^n$, and a parameter $\tau \in (0, 1)$, quantile regression problem can be solved via the optimization problem

$$\text{minimize}_{x \in \mathbb{R}^d} \quad \rho_\tau(Ax - b), \tag{1}$$

  where $\rho_\tau(y) = \sum_{i=1}^n \rho_\tau(y_i)$, for $y \in \mathbb{R}^n$.
- We use A to denote $\begin{bmatrix} A & -b \end{bmatrix}$, the quantile regression problem (1) can equivalently be expressed as the following,

$$\text{minimize}_{x \in \mathcal{C}} \quad \rho_\tau(Ax), \tag{2}$$

  where $\mathcal{C} = \{x \in \mathbb{R}^d \mid c^T x = 1\}$ and c is a unit vector with the last coordinate 1.
- **Goal:** For $A \in \mathbb{R}^{n \times d}$ with $n \gg d$, find $\hat{x}$ such that

$$\rho_\tau(A\hat{x}) \leq (1 + \epsilon)\rho_\tau(Ax^*),$$

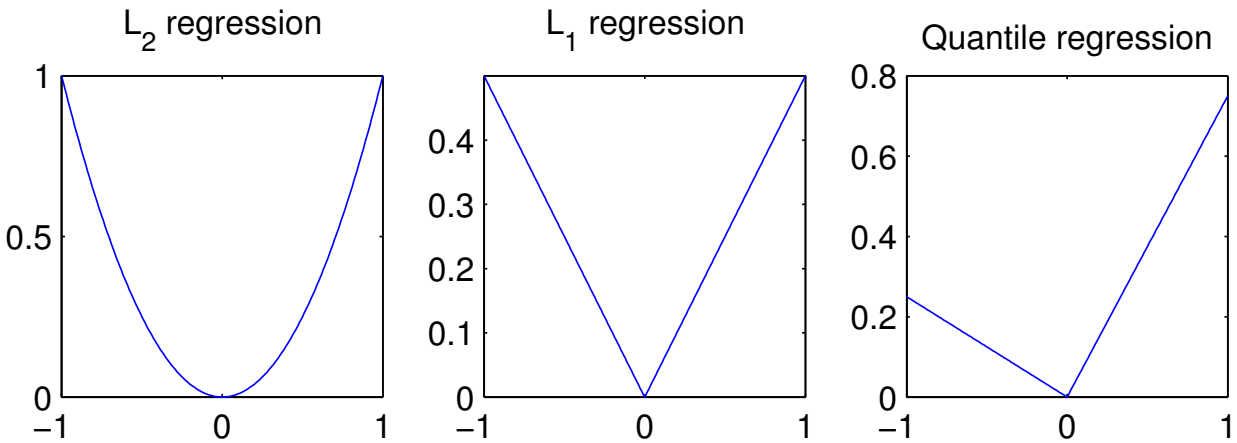
  where x^* is an optimal solution.

Background

- The standard solver for quantile regression problem is interior-point method `ipm` [Portnoy and Koenker, 1997], which might be applicable for medium-large scale problem with size $1e6$ by 50.
- The best previous sampling algorithm, namely `prqfn`, for quantile regression problems is using an interior-point method on a smaller problem that has been preprocessed by randomly sampling a subset of the data; see [Portnoy and Koenker, 1997].
- Inspired by recent work using randomized algorithms to compute approximate solutions for least-squares regression and related problems. For example, [Dasgupta et al., 2009] and [Clarkson et al., 2013].

Comparison of three types of regression problems

	ℓ_2 regression	ℓ_1 regression	quantile regression
estimation	mean	median	quantile τ
loss function	x^2	$\lvert x\rvert$	$\rho_\tau(x)$
formulation	$\lVert Ax-b\rVert_2^2$	$\lVert Ax-b\rVert_1$	$\rho_\tau(Ax-b)$
is a norm?	yes	yes	no



Two important notions

Definition ((α, β)-conditioning and well-conditioned basis (Dasgupta et al., 2009))

Given $A \in \mathbb{R}^{n \times d}$, A is (α, β)-conditioned if $\|A\|_1 \leq \alpha$ and for all $x \in \mathbb{R}^d$, $\beta \|Ax\|_1 \geq \|x\|_\infty$. Define $\kappa(A)$ as the minimum value of $\alpha\beta$ such that A is (α, β)-conditioned. We will say that a basis U of range(A) is a well-conditioned basis if $\kappa = \kappa(U)$ is a low-degree polynomial in d, independent of n.

Definition (ℓ_1 leverage scores (Clarkson et al., 2013))

Given a well-conditioned basis U for the range(A), the leverage scores of A are defined by the ℓ_1 norms of U's rows: $\|U_{(i)}\|_1$, $i = 1, \ldots, n$.

A useful tool

Definition ($(1 \pm \epsilon)$-distortion Subspace-preserving Embedding)

Given $A \in \mathbb{R}^{n \times d}$, $S \in \mathbb{R}^{s \times n}$ is a $(1 \pm \epsilon)$-distortion subspace-preserving matrix if $s = \text{poly}(d)$ and for all $x \in \mathbb{R}^d$,

$$(1 - \epsilon)\rho_\tau(Ax) \leq \rho_\tau(SAx) \leq (1 + \epsilon)\rho_\tau(Ax). \tag{3}$$

Solving the subproblem $\min_{x \in \mathcal{C}} \rho_\tau(SAx)$ gives a $(1 + \epsilon)/(1 - \epsilon)$-approximate solution to the original problem. This is because

$$\rho_\tau(A\hat{x}) \leq \frac{1}{1 - \varepsilon}\rho_\tau(SA\hat{x}) \leq \frac{1}{1 - \varepsilon}\rho_\tau(SAx^*) \leq \frac{1 + \varepsilon}{1 - \varepsilon}\rho_\tau(Ax^*).$$

Sampling lemma

Lemma (Subspace-preserving Sampling Lemma)

Given $A \in \mathbb{R}^{n\times d}$, let $U \in \mathbb{R}^{n\times d}$ be a well-conditioned basis for range(A) with condition number κ. For $s > 0$, choose

$$\hat{p}_i \geq \min\{1, s \cdot \|U_{(i)}\|_1 / \|U\|_1\},$$

and let $S \in \mathbb{R}^{n\times n}$ be a random diagonal matrix with $S_{ii} = 1/\hat{p}_i$ with probability $\hat{p}_i$, and 0 otherwise. Then when $\epsilon < 1/2$ and

$$s \geq \frac{\tau}{1-\tau}\frac{27\kappa}{\epsilon^2}\left(d\log\left(\frac{\tau}{1-\tau}\frac{18}{\epsilon}\right) + \log\left(\frac{4}{\delta}\right)\right), \tag{4}$$

with probability at least $1-\delta$, for every $x \in \mathbb{R}^d$,

$$(1-\varepsilon)\rho_\tau(Ax) \leq \rho_\tau(SAx) \leq (1+\varepsilon)\rho_\tau(Ax).$$

Strategy

- Find a well-conditioned basis U.
- Compute or estimate the ℓ_1 row norms of U and construct sampling matrix S.
- Solve the subproblem $\text{minimize}_{x \in \mathcal{C}} \ \rho_\tau(SAx)$.

Conditioning

- We call the procedure for finding U as conditioning.
- There are many existing conditioning methods. See [Clarkson et al., 2013] and [Dasgupta et al., 2009].
- We care about two important properties: the condition number κ of the resulting basis U and the running time for construction. In general, there is a trade-off between these two quantities.

Comparison of conditioning methods

name	running time	κ	type
SC[SW11]	$\mathcal{O}(nd^2 \log d)$	$\mathcal{O}(d^{5/2} \log^{3/2} n)$	QR
FC [CDMMMW13]	$\mathcal{O}(nd \log d)$	$\mathcal{O}(d^{7/2} \log^{5/2} n)$	QR
Ellipsoid rounding [Cla05]	$\mathcal{O}(nd^5 \log n)$	$d^{3/2}(d+1)^{1/2}$	ER
Fast ER [CDMMMW13]	$\mathcal{O}(nd^3 \log n)$	$2d^2$	ER
SPC1 [MM13]	$\mathcal{O}(\text{nnz}(A))$	$\mathcal{O}(d^{\frac{13}{2}} \log^{\frac{11}{2}} d)$	QR
SPC2 [MM13]	$\mathcal{O}(\text{nnz}(A) \cdot \log(n)) + \texttt{ER\_small}$	$6d^2$	QR+ER
SPC3 (this work)	$\mathcal{O}(\text{nnz}(A) \cdot \log(n)) + \texttt{QR\_small}$	$\mathcal{O}(d^{\frac{19}{4}} \log^{\frac{11}{4}} d)$	QR+QR

Table: Summary of running time, condition number, and type of conditioning methods proposed recently. QR and ER refer, respectively, to methods based on the QR factorization and methods based on Ellipsoid Rounding.

SC := Slow Cauchy Transform
FC := Fast Cauchy Transform
SPC := Sparse Cauchy Transform

Estimating row norms of well-conditioned basis

- Recall, that we choose our sampling probabilities based on the ℓ_1 row norms of a well-conditioned basis:

$$\hat{p}_i \geq \min\{1, s \cdot \|U_{(i)}\|_1 / \|U\|_1\}.$$

- Generally, we find a matrix R such that AR^{-1} is a well-conditioned basis.
- We post-multiply a random projection matrix $\Pi \in \mathbb{R}^{d \times \mathcal{O}(\log n)}$ on AR^{-1} and compute the median of each row of the resulting matrix.
- This gives us an estimation of the ℓ_1 row norms of AR^{-1} up to some constant factor running in $\mathcal{O}(\text{nnz}(A) \cdot \log n)$ time; see [Clarkson et al., 2013].

$$\begin{matrix} A & & R^{-1} & & \Pi \end{matrix}$$
$$\begin{pmatrix} \quad \end{pmatrix} \cdot \begin{pmatrix} \quad \end{pmatrix} \cdot \begin{pmatrix} \quad \end{pmatrix}$$

Fast Randomized Algorithm for Quantile Regression

Input: $A \in \mathbb{R}^{n \times d}$ with full column rank, $\epsilon \in (0, 1/2)$, $\tau \in [1/2, 1)$.
Output: An approximate solution $\hat{x} \in \mathbb{R}^d$ to problem minimize$_{x \in \mathcal{C}}$ $\rho_\tau(Ax)$.

1: Compute $R \in \mathbb{R}^{d \times d}$ such that AR^{-1} is a well-conditioned basis for range(A).
2: Compute a $(1 \pm \epsilon)$-distortion subspace-preserving embedding $S \in \mathbb{R}^{s \times n}$.
3: Return $\hat{x} \in \mathbb{R}^d$ that minimizes $\rho_\tau(SAx)$ with respect to $x \in \mathcal{C}$.

Theorem (Fast Quantile Regression)

Given $A \in \mathbb{R}^{n \times d}$ and $\varepsilon \in (0, 1/2)$, the above algorithm returns a vector $\hat{x}$ that, with probability at least 0.8, satisfies

$$\rho_\tau(A\hat{x}) \leq \left(\frac{1+\varepsilon}{1-\varepsilon}\right) \rho_\tau(Ax^*),$$

where x^* is an optimal solution to the original problem. In addition, the algorithm to construct $\hat{x}$ runs in time

$$\mathcal{O}(\text{nnz}(A) \cdot \log n) + \phi\left(\mathcal{O}(\mu d^3 \log(\mu/\epsilon)/\epsilon^2), d\right), \tag{5}$$

where $\mu = \frac{\tau}{1-\tau}$ and $\phi(s, d)$ is the time to solve a quantile regression problem of size $s \times d$.

Outline of empirical evaluation

We will show

- numerical results for medium-scale data with size about $1e6$ by 50 as well as large-scale data with size $1.1e10$ by 10;
- plots of relative errors versus sampling size, lower dimension and so on by using different conditioning-based methods;
- comparison of running time performance with existed methods.

Types of data

Synthetic data

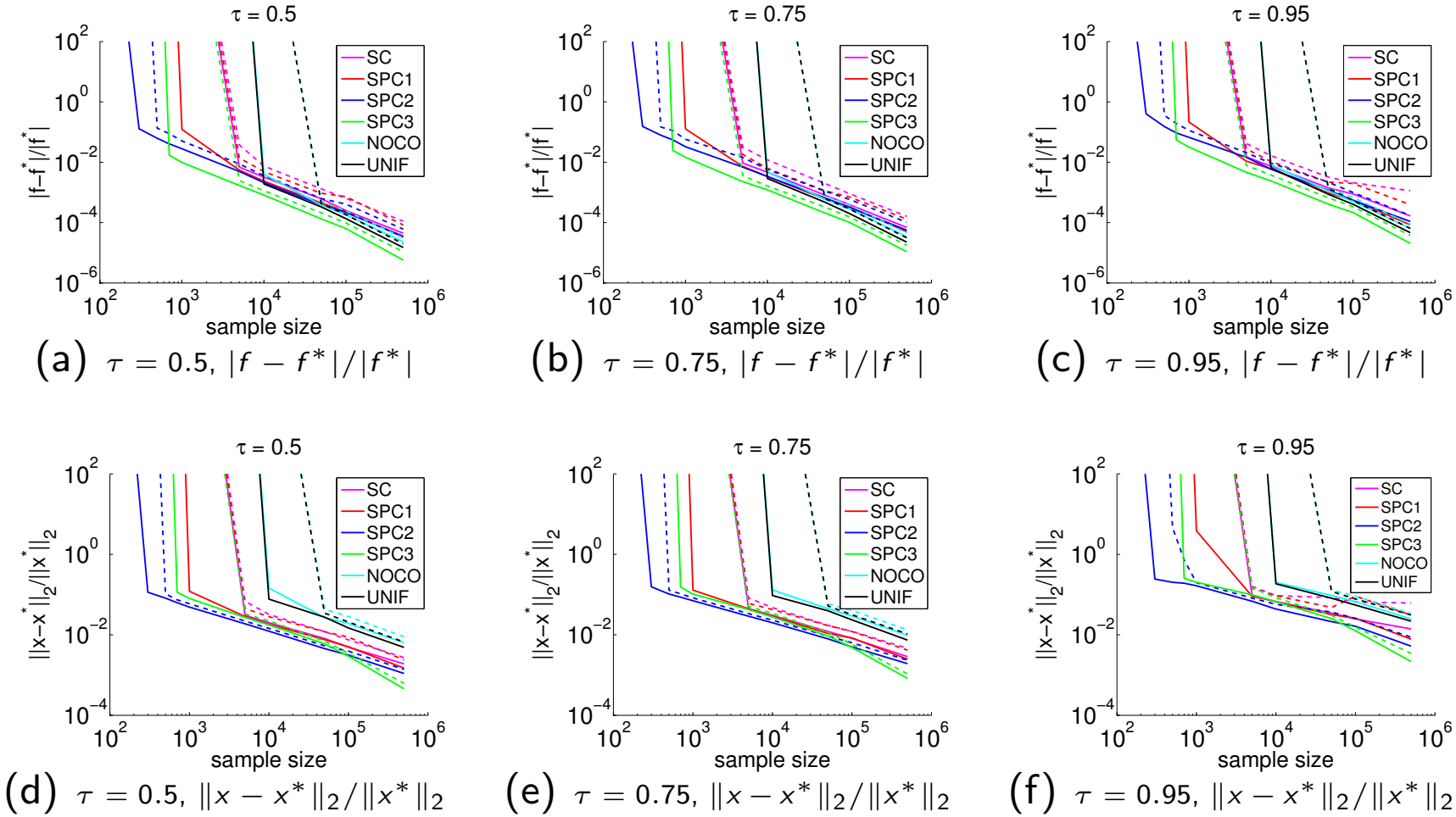
We simulate our data in the following manner. A similar construction for the test data appeared in [Clarkson et al., 2013].

- Each row of the design matrix A is a canonical vector. Suppose the number of measurements on the j-th column are c_j, where $c_j = qc_{j-1}$, for $j = 2, \ldots, d$. Here $1 < q \leq 2$. A is a $n \times d$ matrix.
- The true vector x^* with length d is a vector with independent Gaussian entries. Let $b^* = Ax^*$.
- The response vector b is obtained by adding noise to b^*.

Real data

We consider a data set consisting of a 5% sample of the U.S. 2000 Census data consisting of annual salary and related features. The size of the design matrix is 5×10^6 by 11.

Relative error when the sampling size s changes



(a) $\tau = 0.5$, $|f - f^*|/|f^*|$ (b) $\tau = 0.75$, $|f - f^*|/|f^*|$ (c) $\tau = 0.95$, $|f - f^*|/|f^*|$

(d) $\tau = 0.5$, $\|x - x^*\|_2/\|x^*\|_2$ (e) $\tau = 0.75$, $\|x - x^*\|_2/\|x^*\|_2$ (f) $\tau = 0.95$, $\|x - x^*\|_2/\|x^*\|_2$

Figure: The first (solid lines) and the third (dashed lines) quartiles of the relative errors of the objective value and solution vector. The test is on synthetic data with size 1e6 by 50.

Relative error of each method measured in three different norms

	$\|x - x^*\|_2/\|x^*\|_2$	$\|x - x^*\|_1/\|x^*\|_1$	$\|x - x^*\|_\infty/\|x^*\|_\infty$
SC	[0.0121, 0.0172]	[0.0093, 0.0122]	[0.0229, 0.0426]
SPC1	[0.0108, 0.0170]	[0.0081, 0.0107]	[0.0198, 0.0415]
SPC2	[0.0079, 0.0093]	[0.0061, 0.0071]	[0.0115, 0.0152]
SPC3	[0.0094, 0.0116]	[0.0086, 0.0103]	[0.0139, 0.0184]
NOCO	[0.0447, 0.0583]	[0.0315, 0.0386]	[0.0769, 0.1313]
UNIF	[0.0396, 0.0520]	[0.0287, 0.0334]	[0.0723, 0.1138]

Table: The first and the third quartiles of relative errors of the solution vector, measured in ℓ_1, ℓ_2, and ℓ_∞ norms. The test data set is the synthetic data, with size $1e6 \times 50$, the sampling size $s = 5e4$, and $\tau = 0.75$.

Comparison of the running time of each conditioning method

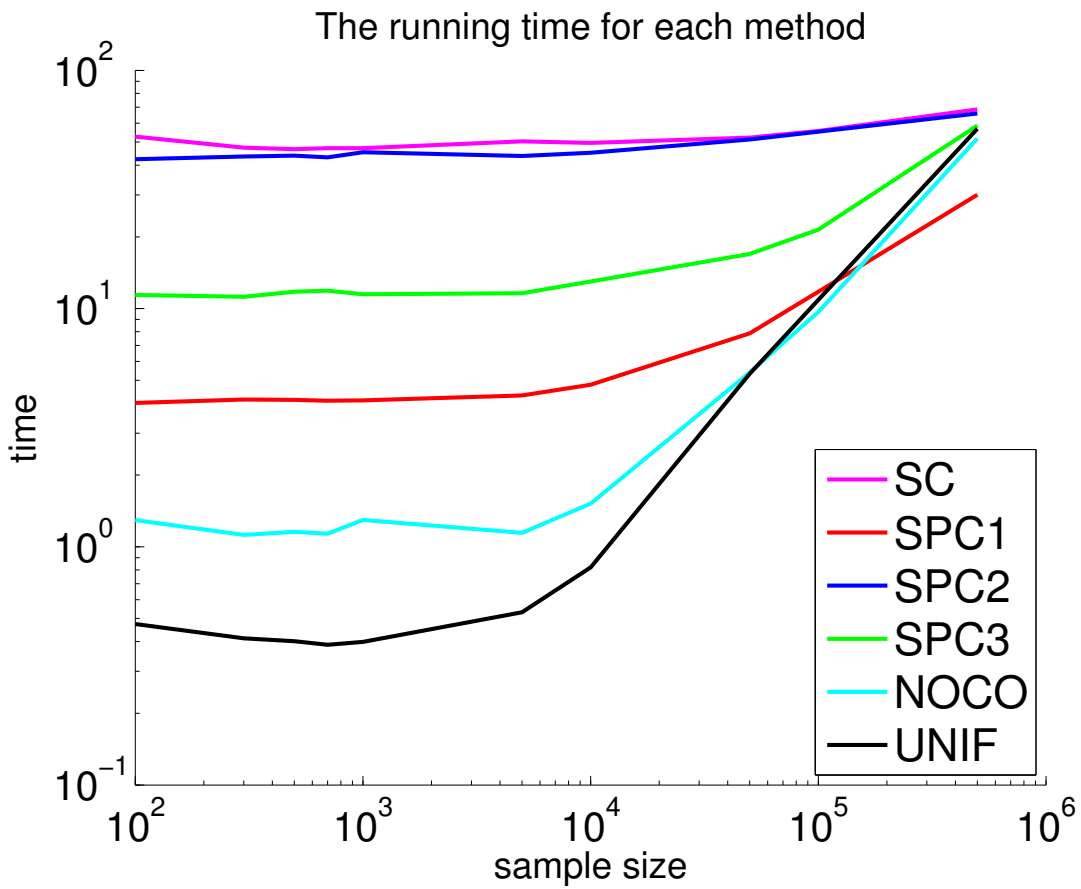
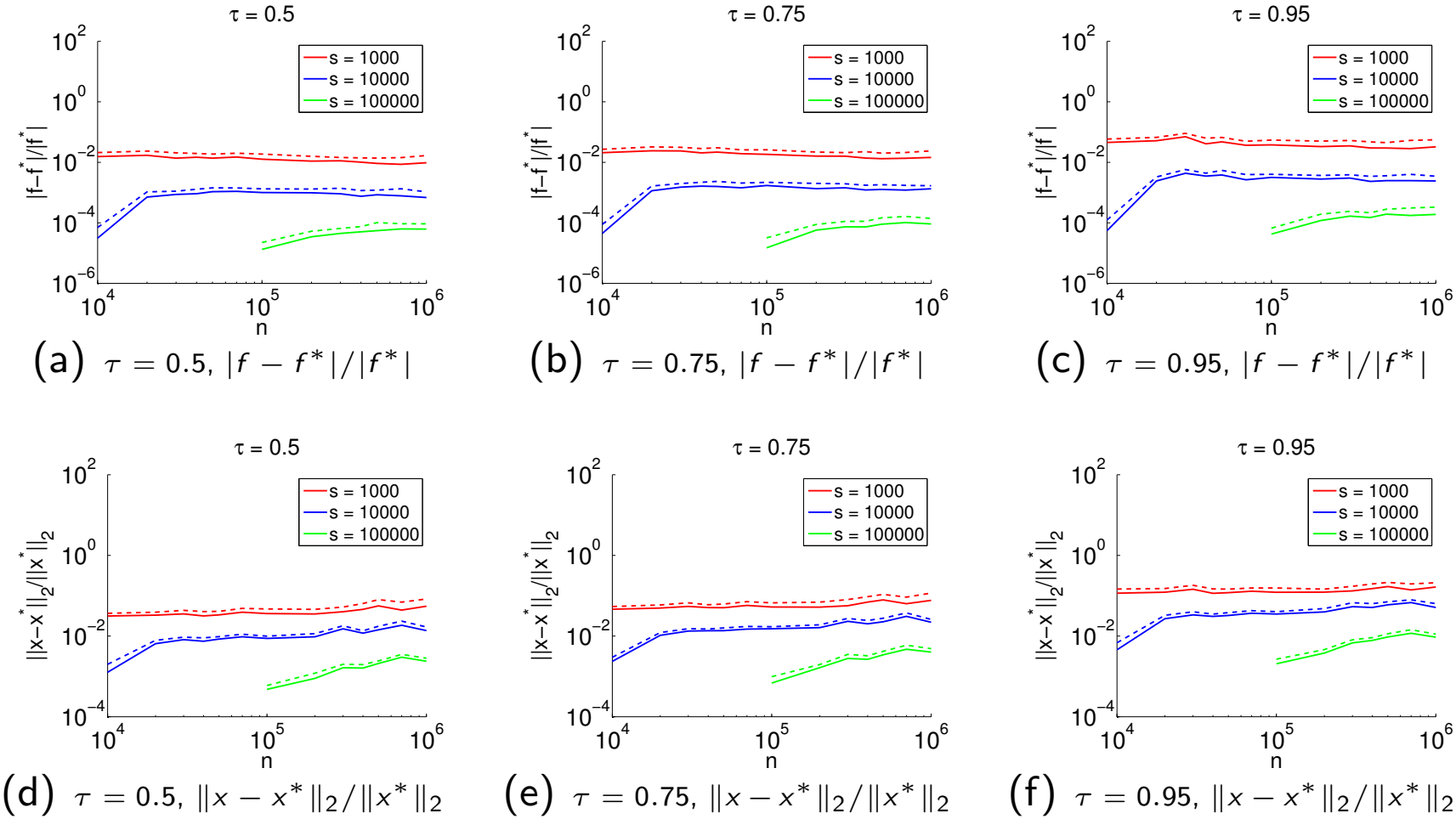


Figure: The running time for solving the problems associated with three different τ values when the sampling size s changes.

Relative error when the higher dimension n changes

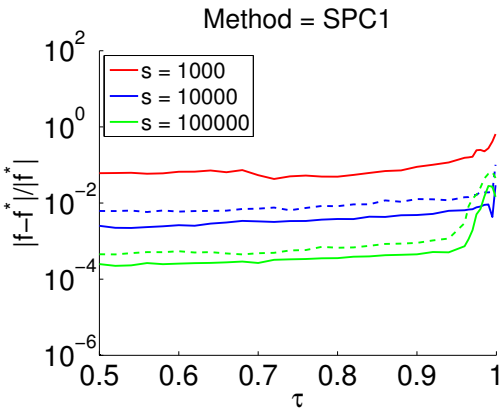


(a) $\tau = 0.5$, $|f - f^*|/|f^*|$ (b) $\tau = 0.75$, $|f - f^*|/|f^*|$ (c) $\tau = 0.95$, $|f - f^*|/|f^*|$

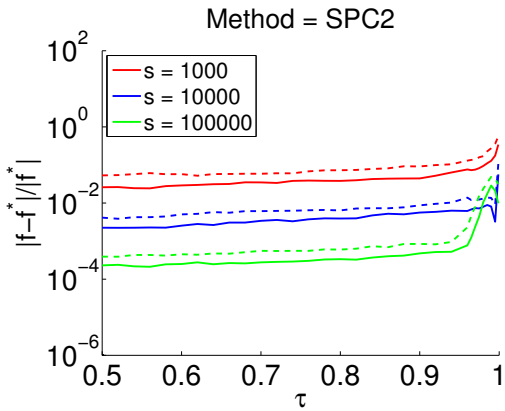
(d) $\tau = 0.5$, $\|x - x^*\|_2/\|x^*\|_2$ (e) $\tau = 0.75$, $\|x - x^*\|_2/\|x^*\|_2$ (f) $\tau = 0.95$, $\|x - x^*\|_2/\|x^*\|_2$

Figure: The first (solid lines) and the third (dashed lines) quartiles of the relative errors of the objective value and solution vector, when n varying from 1e4 to 1e6 and $d = 50$ by using SPC3.

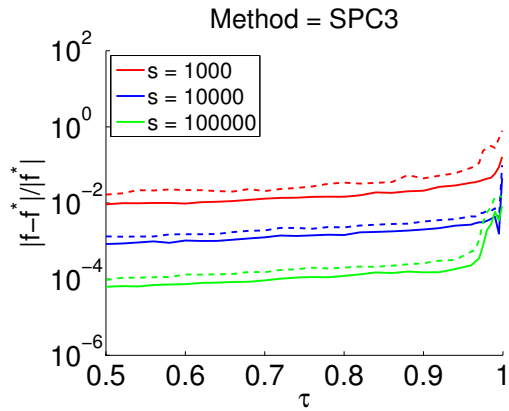
Relative error when the quantile τ changes



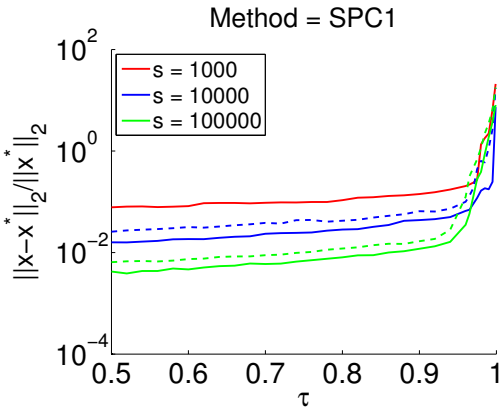
(a) SPC1, $|f - f^*|/|f^*|$



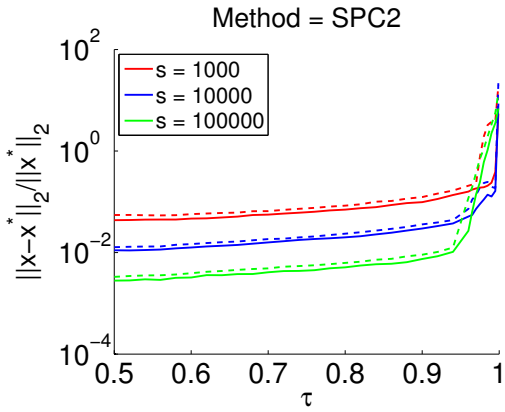
(b) SPC2, $|f - f^*|/|f^*|$



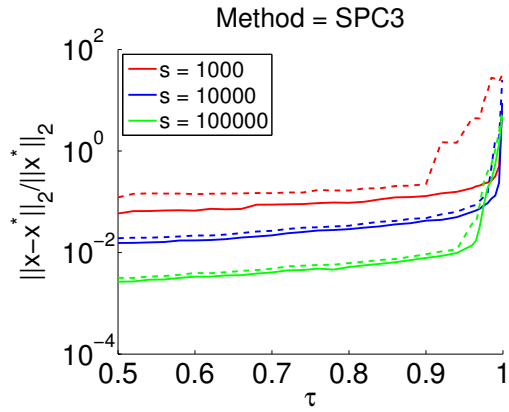
(c) SPC3, $|f - f^*|/|f^*|$



(d) SPC1, $\|x - x^*\|_2/\|x^*\|_2$



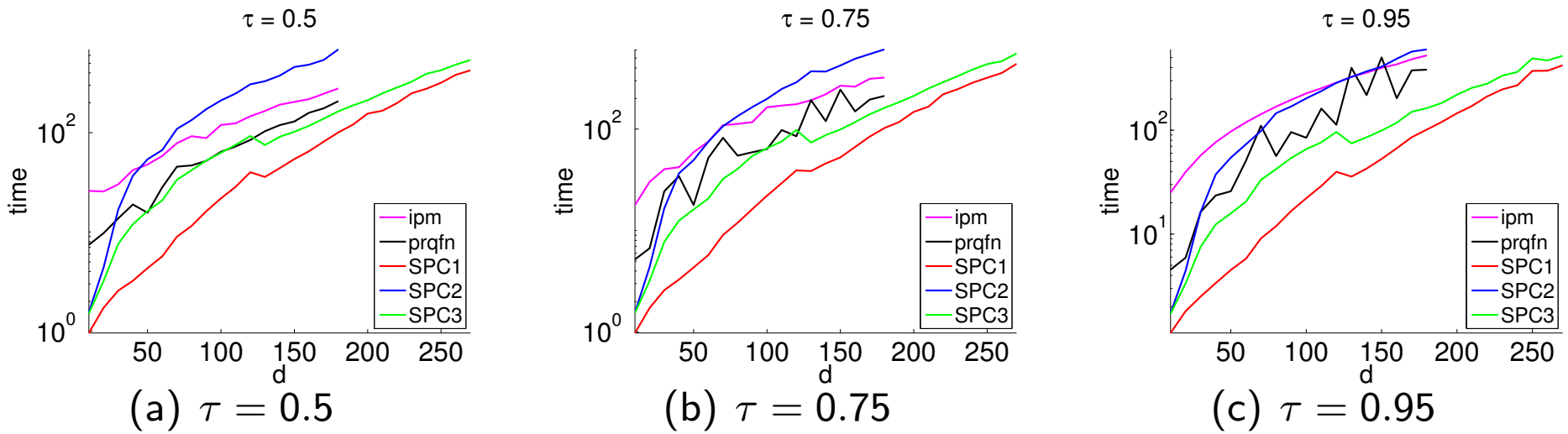
(e) SPC2, $\|x - x^*\|_2/\|x^*\|_2$



(f) SPC3, $\|x - x^*\|_2/\|x^*\|_2$

Figure: The first (solid lines) and the third (dashed lines) quartiles of the relative errors of the objective value, and solution vector. The test data size 1e6 by 50.

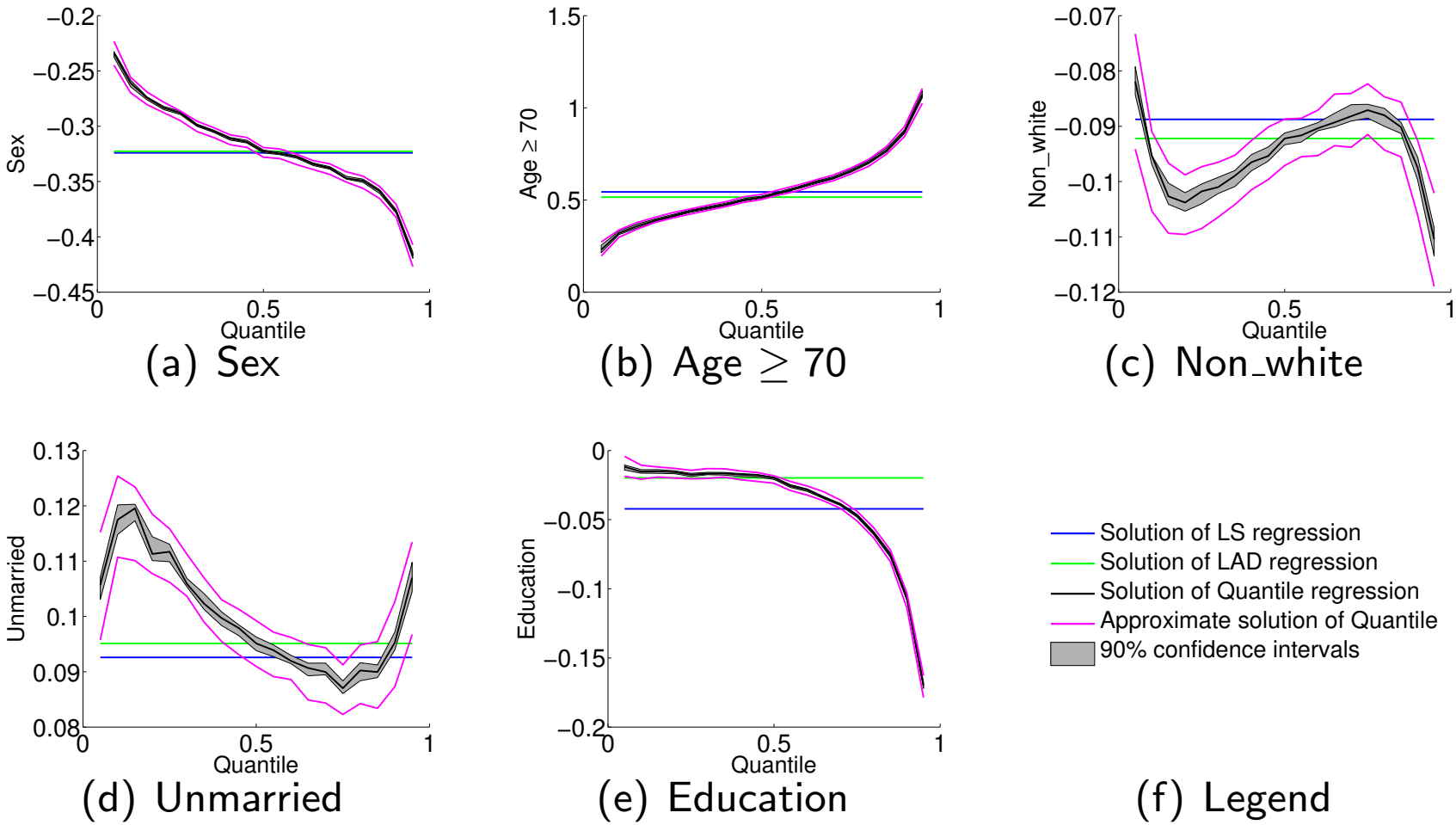
Running time when the lower dimension d changes



(a) $\tau = 0.5$ (b) $\tau = 0.75$ (c) $\tau = 0.95$

Figure: The running time for five methods for solving simulated problem, with $n = 1e6$, when d varies.

Plots for real data



(a) Sex (b) Age $\geq$ 70 (c) Non_white

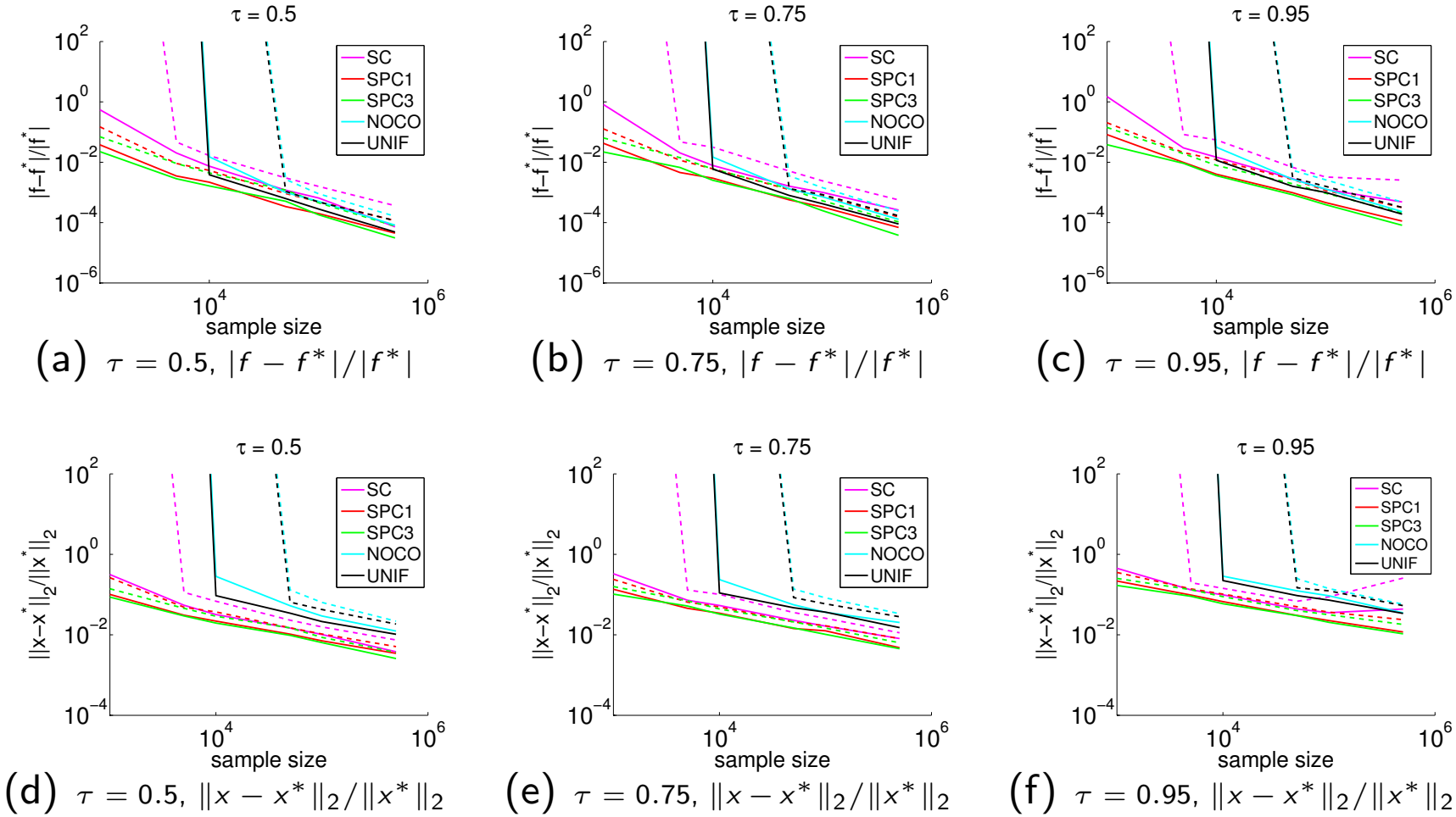
(d) Unmarried (e) Education (f) Legend

Figure: Each subfigure is associated with a coefficient in the census data. The two magenta curves show the first and third quartiles of solutions obtained by using SPC3, among 200 independent trials with sampling size $s = 5e4$.

Large-scale data and MapReduce

- At terabyte scale, interior-point method `ipm` has two major issues: memory requirement and running time.
- The MapReduce framework is the *de facto* standard parallel environment for large data analysis.
- Since our sampling algorithm only needs 3 passes through the data and it is embarrassingly parallel, it is straightforward to implement it on Hadoop.
- For a simulated data with size $5e6 \times 10$, we stack it vertically 2200 times. This leads to a data with size $1.1e10 \times 10$.

Relative error when the sampling size s changes



(a) $\tau = 0.5$, $|f - f^*|/|f^*|$ (b) $\tau = 0.75$, $|f - f^*|/|f^*|$ (c) $\tau = 0.95$, $|f - f^*|/|f^*|$

(d) $\tau = 0.5$, $\|x - x^*\|_2/\|x^*\|_2$ (e) $\tau = 0.75$, $\|x - x^*\|_2/\|x^*\|_2$ (f) $\tau = 0.95$, $\|x - x^*\|_2/\|x^*\|_2$

Figure: The first (solid lines) and the third (dashed lines) quartiles of the relative errors of the objective value and solution vector. The test is on replicated synthetic data with size 1.1e10 by 10.

Relative error of each method measured in three different norms

	$\|x - x^*\|_2/\|x^*\|_2$	$\|x - x^*\|_1/\|x^*\|_1$	$\|x - x^*\|_\infty/\|x^*\|_\infty$
SC	[0.0081, 0.0112]	[0.0073, 0.0098]	[0.0078, 0.0140]
SPC1	[0.0048, 0.0080]	[0.0048, 0.0074]	[0.0047, 0.0082]
SPC3	[0.0045, 0.0063]	[0.0043, 0.0060]	[0.0043, 0.0062]
NOCO	[0.0203, 0.0335]	[0.0176, 0.0251]	[0.0209, 0.0413]
UNIF	[0.0151, 0.0281]	[0.0131, 0.0230]	[0.0180, 0.0347]

Table: The first and the third quartiles of relative errors of the solution vector, measured in ℓ_1, ℓ_2, and ℓ_∞ norms. The test is on replicated synthetic data with size 1.1e10 by 10, the sampling size $s = 5e5$, and $\tau = 0.75$.

Conclusion

- Proposed, analyzed, and evaluated new randomized algorithm for solving medium-scale and large-scale quantile regression problems.
- Uses a subsampling technique that involves constructing an ℓ_1-well-conditioned basis.
- Runs in nearly input-sparsity time, plus the time needed for solving a subsampled problem whose size depends only on the lower dimension of the design matrix.
- Provided a detailed empirical evaluation of our main algorithm.