Fast Inference for Quantile Regression with Tens of Millions of Observations

Sokbae Lee, Yuan Liao, Myung Hwan Seo & Youngki Shin

Columbia, Rutgers, SNU, & McMaster

University of California, Riverside
December 6, 2022

Main Object

In this paper, we tackle on the inference problem of quantile regression (QR) with $(n, d) \sim (10^7, 10^3)$:

$$y_i = x_i'\beta^* + \varepsilon_i, \quad P(\varepsilon \leq 0|x_i) = \tau.$$

- We estimate the wage structure (college premium) using the data from IPUMS USA. The sample size of each year is over 14 millions.
- We also apply many controls to mitigate the bias, which turns out to be over 1,000.
- For a smaller sample size, we need additional assumptions, e.g. sparsity in the lasso.

Big Picture

- Standard asymptotics: $n \gg d$. E.g. $n = 1000$ and $d = 20$.
- High-dimensional approach: $n \sim d$ or $n \ll p$. E.g. $n \sim \exp(d)$
    - Threshold regression: LSS (2016, JRSSB), LLSS (2018, JASA), LLSS (2021a, AOS).
    - Filtering and prediction: LLSS (2021b, JOE), LS (forthcoming, JOE)
- The current problem is $n = 10^7$ and $d = 10^3$. Can we go back to the standard framework?
- It turns out that we need a novel approach because of the implementation issue.

Standard QR Estimator

Let $\{Y_i \equiv ((y_i, x_i) \in \mathbb{R}^(1+d) : i = 1, \ldots, n\}$ be a random sample generated from $y_i = x_i' \beta^* + \varepsilon_i, \quad P(\varepsilon \leq 0 | x_i) = \tau$.

The object of interest is

$$\beta^* := \arg\min_{\beta \in \mathbb{R}^d} Q(\beta),$$

where

$$\begin{aligned} Q(\beta) &:= \mathbb{E}[q(\beta, Y_i)] \\ q(\beta, Y_i) &:= (y_i - x_i'\beta)(\tau - I\{y_i - x_i'\beta \leq 0\}). \end{aligned}$$

The QR estimator is defined as

$$\widehat{\beta}_n := \arg\min_{\beta \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} q(\beta_, Y_i).$$

Standard QR Estimator (cont.)

The standard M-estimator theory gives us

$$\sqrt{n}(\widehat{\beta}_n - \beta^*) \xrightarrow{d} N(0, \tau(1-\tau)H^{-1}\mathbb{E}[x_i x_i']H^{-1}),$$

where $H = \mathbb{E}[f_\varepsilon(0|x_i)x_i x_i']$ and $f_\varepsilon(\cdot|x_i)$ is the conditional distribution of ε_i given x_i

- Point estimator: Linear programming through interior-point algorithms or smoothing type estimators
- Covariance estimator: The `conquer` method in He et al. (2021) has received attention and boast its capability to make inference with $(n, p) = (4000, 100)$.

New approach: we propose a stochastic subgradient descent (S-subGD) method with a random scaling.

Gradient Descent

Let β^* be the parameter of interest:

$$\beta^* := \arg\min_{\beta\in\mathbb{R}^d} Q\left(\beta\right)$$

where $Q := E[q(\beta, Y)]$ and q is diff. and convex. Let $\{Y_t\}_{t=1}^n$ be a random sample. The sample analogue of the FOC is

$$\frac{1}{n}\sum_{t=1}^{n} \nabla q\left(\hat{\beta}, Y_t\right) = 0.$$

If we don't have a reduced form solution, we can solve it iteratively:

$$\hat{\beta}_m = \hat{\beta}_{m-1} - \gamma_m \frac{1}{n}\sum_{t=1}^{n} \nabla q\left(\hat{\beta}_{m-1}, Y_t\right).$$

Stochastic Gradient Descent

Limitations of gradient descent:

- It calculates the derivatives for the entire dataset.
- It requires a larger memory size as the dataset increases.

Binding time budget or the memory size occurs more often in modern empirical applications.

Robbins and Monro (1951) proposed the stochastic gradient descent (SGD) solution path as

$$\beta_t = \beta_{t-1} - \gamma_t \nabla q \left(\beta_{t-1}, Y_t\right).$$

SGD has advantages when we face a large-scale dataset or online machine learning.
Examples: Chen and White (2002), Khan, Lan, and Tamer (2021).

SGD Averaging

Recall that we aim to develop online inference with SGD.

We study the classical Polyak-Ruppert averaging estimator (Polyak (1990) and Ruppert(1988)): $\bar{\beta}_n := n^{-1} \sum_{t=1}^{n} \beta_t$.

Polyak and Juditsky (1992) established regularity conditions under which the averaging estimator $\bar{\beta}_n$ is asymptotically normal:

$$\sqrt{n} \left(\bar{\beta}_n - \beta^* \right) \xrightarrow{d} \mathcal{N}(0, \Upsilon),$$

where the asymptotic variance Υ has a sandwich form

$$\Upsilon := H^{-1} S H^{-1},$$

and $H := \mathbb{E}[\nabla^2 Q(\beta^*)]$ is the Hessian matrix and $S := \mathbb{E}\left[\nabla q(\beta^*, Y) \nabla q(\beta^*, Y)' \right]$ is the score variance.

SGD Averaging in Online Learning

In online learning, data arrive sequentially.

The Polyak-Ruppert estimator $\bar{\beta}_n$ can be computed recursively by the updating rule

$$\bar{\beta}_t = \bar{\beta}_{t-1}\frac{t-1}{t} + \frac{\beta_t}{t},$$

which implies that it is well suited to the online setting.

Examples include

- Linear regression (with a large dataset)
- Logistic regression
- Quantile regression (using a subgradient):

$$\nabla q(\beta; x, y) = -x[u - 1\{y \leq x'\beta\}],$$

Overview

- Introduction
- Inference Methods
- Theoretical Results
- Monte Carlo Experiments
- Application
- Conclusion

Inference Methods

Online Inference

Although the asymptotic normality result by Polyak and Juditsky (1992) was established about three decades ago, it is only past few years that online inference has gained increasing interest in the literature.

It is challenging to estimate the asymptotic variance Υ in an online fashion.

This is because the naive implementation of estimating it requires storing all data, thereby losing the advantage of online learning.

Method 1: Plug-In

Chen et al. (2020) addressed this issue by estimating H and S using the online iterated estimator β_t, and recursively updating them whenever a new observation is available.

However, the plug-in estimator requires that the Hessian matrix be computed to estimate H.

In other words, it is necessary to have strictly more inputs than the SGD solution paths β_t to carry out inference. It is the case even when a t-statistic is computed for each regression coefficient.

In applications, it can be demanding to compute the Hessian matrix and its inverse.

Method 2: Batch-Means

This method proposed by Chen et al. (2020) and Zhu et al. (2021) directly estimates the variance of the averaged online estimator $\bar{\beta}_n$ by dividing $\{\beta_1, \ldots, \beta_n\}$ into batches with increasing batch size.

The batch-means estimator is based on the idea that correlations among batches that are far apart decay exponentially fast; therefore, one can use nonparametric empirical covariance to estimate Υ.

However, this approach requires the batch size should increase exponentially fast, and it turns out that the performance is not satisfactory.

Method 3: Bootstrap

Instead of estimating the asymptotic variance, Fang et al. (2018) proposed a bootstrap procedure for online inference.

Specifically, they proposed to use a large number (say, B) of randomly perturbed SGD solution paths: for all $b = 1, \ldots, B$, starting with $\beta_0^{(b)} = \beta_0$ and then iterating

$$\beta_t^{(b)} = \beta_{t-1}^{(b)} - \gamma_t \eta_t^{(b)} \nabla q \left(\beta_{t-1}^{(b)}, Y_t \right),$$

where $\eta_t^{(b)} > 0$ is an independent and identically distributed random variable that has mean one and variance one.

The bootstrap procedure needs strictly more inputs than computing $\bar{\beta}_n$ and can be time-consuming.

Our Approach: Random Scaling

- Lee, Liao, Seo, Shin (LLSS, 2022) proposed not to estimate the asymptotic variance Υ, but to studentize $\sqrt{n}\left(\bar{\beta}_n - \beta^*\right)$ via $\widehat{V}_n^{1/2}$, where

$$\widehat{V}_n := \frac{1}{n}\sum_{s=1}^{n}\left\{\frac{1}{\sqrt{n}}\sum_{t=1}^{s}\left(\beta_t - \bar{\beta}_n\right)\right\}\left\{\frac{1}{\sqrt{n}}\sum_{t=1}^{s}\left(\beta_t - \bar{\beta}_n\right)\right\}'.$$

- It converges in distribution to a pivotal distribution up to an unknown scale, which is the same as the asymptotic variance of the average SGD estimator. This leverages insights from the time series literature (e.g. Kiefer et al. (2000)).
- It has been already being adopted in other machine-learning literature like in federated learning (Li et al. 2022), Kiefer-Wolfowitz method (Chen et al. 2021) among others.

Algorithm

```
Input: function $\nabla q(\cdot)$, parameters $(\gamma_0, a)$ for step size $\gamma_t = \gamma_0 t^{-a}$ for $t \geq 1$
Initialize: set initial values for $\beta_0, \bar{\beta}_0, A_0, b_0$
for $t = 1, 2, \ldots$ do
    Receive: new observation $Y_t$
    $\beta_t = \beta_{t-1} - \gamma_t \nabla q\left(\beta_{t-1}, Y_t\right)$
    $\bar{\beta}_t = \bar{\beta}_{t-1} \frac{t-1}{t} + \frac{\beta_t}{t}$
    $A_t = A_{t-1} + t^2 \bar{\beta}_t \bar{\beta}_t'$
    $b_t = b_{t-1} + t^2 \bar{\beta}_t$
    $c_t = c_{t-1} + t^2$
    Obtain $\widehat{V}_t$ by
        $$\widehat{V}_t = t^{-2}\left(A_t - \bar{\beta}_t b_t' - b_t \bar{\beta}_t' + \bar{\beta}_t \bar{\beta}_t' c_t\right)$$
    Output: $\bar{\beta}_t$, $\widehat{V}_t$
end
```

Additional Remarks/Features

- We set an initial value β_0 from He et al (2021). Works well with only 1% or 5% of the sample.
- We set the step size γ_t is set closer to upper bound, $t^{-0.501}$.
- It does not involve any inverse matrix. We can iterate only a sub-matrix of $\widehat{V}_n$. E.g. a scalar vs. a (1000×1000) matrix.

Theoretical Results

Functional Central Limit Theorem for Online SGD

We first extend Polyak and Juditsky (1992)'s central limit theorem (CLT) to a *functional* CLT (FCLT) for partial sum process:

$$\frac{1}{\sqrt{n}}\sum_{t=1}^{[nr]}(\beta_t - \beta^*) \Rightarrow \Upsilon^{1/2}W(r), \quad r \in [0,1],$$

where $\Rightarrow$ stands for the weak convergence in $\ell^\infty[0,1]$ and $W(r)$ stands for a vector of the independent standard Wiener processes on $[0,1]$.

That is, the partial sum of the online updated estimates β_t converges weakly to a rescaled Wiener process,

Note that the scaling $\Upsilon^{1/2}$ is equal to the square root asymptotic variance of the Polyak-Ruppert average.

PJ's approximation

Their CLT is built on a brilliant stochastic approximation that

$$\frac{1}{\sqrt{n}}\sum_{t=1}^{n}(\beta_t-\beta^*)=\frac{1}{\sqrt{n}}\sum_{t=1}^{n}H^{-1}\xi_t+o_p(1),$$

where ξ_t is a martingale difference sequence (MDS) whose variance converges to the score variance.

For FCLT, it is required to extend the approximation to the uniform approximation.

Conditions for Quantile Regression

Let the partial derivative $\frac{d}{de} f_\varepsilon(\cdot|x_i)$ exist and assume that

1. There exist positive constants ϵ and c_0 such that

$$\inf_{|\beta-\beta^*|<\epsilon} \lambda_{\min}\left(\mathbb{E}[x_i x_i' f_\varepsilon(x_i'(\beta-\beta^*)|x_i)]\right) > c_0,$$

2. $\sup_b \mathbb{E}[\|x_i\|^3 A(b,x_i)] < C$ for some constant $C < \infty$, where $A(b,x_i) := \left|\frac{d}{de} f_\varepsilon(x_i' b|x_i)\right| + f_\varepsilon(x_i' b|x_i)$,
3. $\mathbb{E}[(\|x_i\|^6 + 1)\exp(\|x_i\|^2)] < C$ for some constant $C < \infty$,

This is a set of low-level conditions to meet Gadat and Panloup (2022)'s consistency without strong convexity.

Main Theorem

Let for any $\ell \leq d$ linear restrictions

$$H_0 : R\beta^* = c,$$

where R is an $(\ell \times d)$-dimensional known matrix of rank ℓ and c is an ℓ-dimensional known vector.

Theorem

Suppose rank$(R) = \ell$. Under the stated Assumptions and H_0,

$$n\left(R\bar{\beta}_n - c\right)'\left(R\widehat{V}_n R'\right)^{-1}\left(R\bar{\beta}_n - c\right) \xrightarrow{d} W(1)'\left(\int_0^1 \bar{W}(r)\bar{W}(r)'dr\right)^{-1} W(1),$$

where $W(\cdot)$ is an ℓ-dimensional vector of the standard Wiener processes and $\bar{W}(r) := W(r) - rW(1)$.

Special Case: t-Statistic

The t-statistic for each j converges in distribution:

$$\frac{\sqrt{n}\left(\bar{\beta}_{n,j} - \beta_j^*\right)}{\sqrt{\widehat{V}_{n,jj}}} \xrightarrow{d} W_1(1)\left[\int_0^1 \{W_1(r) - rW_1(1)\}^2\, dr\right]^{-1/2},$$

- The asymptotic distribution is mixed normal and symmetric around zero.
- It is the same as the distribution of the statistics observed in the estimation of the cointegration vector by Johansen (1991). They are different statistics but have the identical distribution as functions of the standard Wiener process as shown by Abadir and Paruolo (2002).

Monte Carlo Experiments

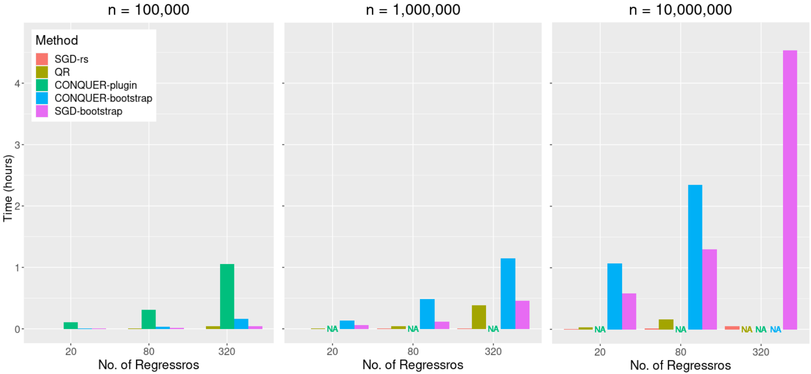
Settings

- $\dim(x_t) = d \in \{10, 30, 180, 320, 1000\}$
- $n \in \{10^5, 10^6, 10^7\}$.
- We compare 5 methods:
  - S-subGD: our method
  - QR: "standard", R package `quantreg`.
  - CONQUER-plugin: He et al (2021), R package `conquer`.
  - CONQUER-bootstrap: He et al (2021), R package `conquer`.
  - SGD-bootstrap: Fang et al. (2018). Bootstrap-online learning
- Constraints: 10 hour and 192 Gb RAM for one replication.

Can you compute?

- S-subGD: Yes for all cases.
- QR: out-of memory when $n = 10^7$ and $d = 320$.
- Conquer-plugin: out-of memory when $n = 10^6$.
- Conquer-bootstrap: out-of-time when $n = 10^7$ and $d = 320$.
- SGD-bootstrap: Survived but 100 times slower than S-subGD.

Figure: Computation Time



Note: Observe 'NA' for several cases.

Figure: Relative Computation Time

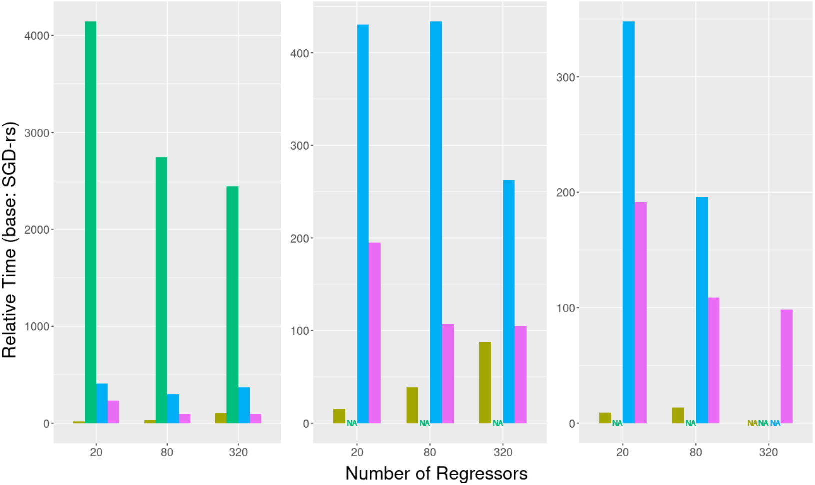


Figure: Coverage Rate

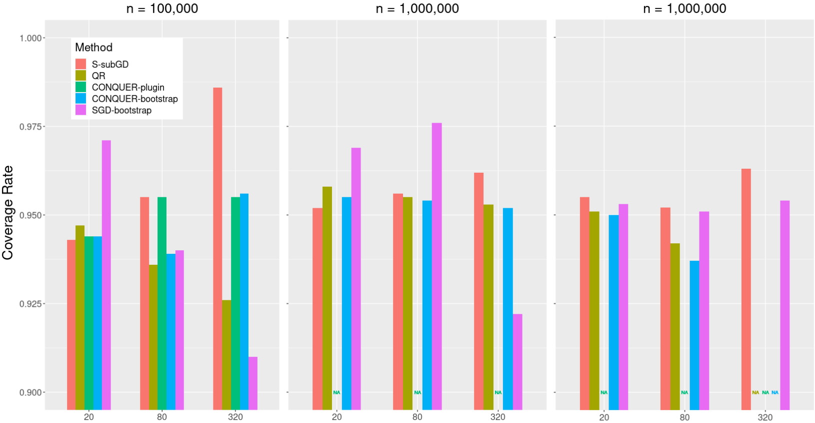
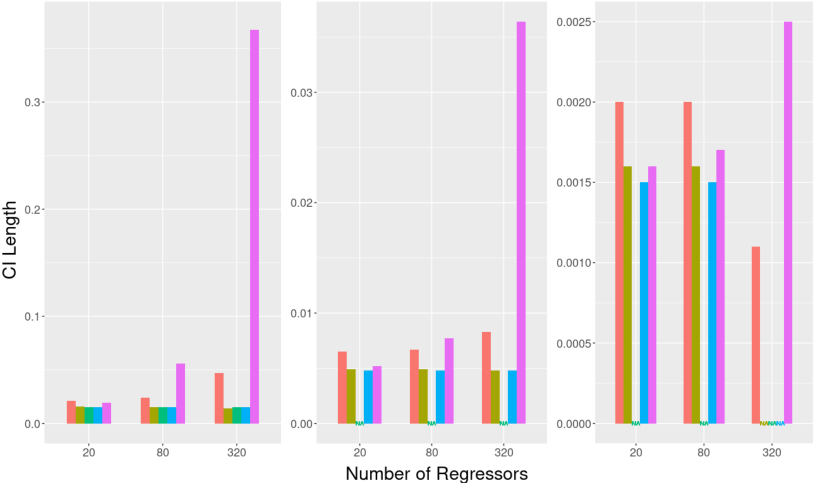


Figure: Confidence Interval Length



Stretch Out

Table: Performance of S-subGD: $n = 10^7$

d	Time (sec.)	Coverage Rate	CI Length
10	5.87	0.965	0.0020
20	11.05	0.955	0.0020
40	21.86	0.954	0.0020
80	43.12	0.952	0.0020
160	81.35	0.953	0.0021
320	166.40	0.963	0.0011
1000	762.16	0.925	0.0461

Application: College Wage Premium

Gender Gap in College Wage Premium

- The higher college wage premium for women has been suggested as a major cause for attracting more women to colleges than men (e.g., Goldin et al. (2006); Chiappori et al. (2009)).
- Top coded wage issue: Hubbard (2011) estimated the model by censored regression and quantile regression
- Our Goals:
  1. To identify (if any) heterogeneous effects across quantiles.
  2. To understand the difference in gender trends in the college wage premium.
  3. To utilize a large number of controls (census data).

Samples from IPUMS

- We use the samples over several different years (1980, 1990, 2000-2015) from IPUMS USA.
- In the years from 1980 to 2000, we use the 5% State sample which is a 1-in-20 national random sample of the population. In the remaining years, we use the American Community Survey (ACS) each year.
- The sampling ratio varies from 1-in-261 to 1-in-232 in 2001-2004, but it is set to a 1-in-100 national random sample of the population after 2005.
- To balance the sample size, we bunch the sample every 5 year after 2001.

Summary Statistics

Year	Sample Size	$\mathbb{E}(F)$	$\mathbb{E}(Educ)$	$\mathbb{E}(Educ \mid M)$	$\mathbb{E}(Edu \mid F)$
1980	3,659,684	0.390	0.433	0.444	0.416
1990	4,192,119	0.425	0.543	0.537	0.550
2000	4,479,724	0.439	0.600	0.578	0.629
2001-2005	2,493,787	0.447	0.642	0.619	0.670
2006-2010	4,708,119	0.447	0.663	0.631	0.701
2011-2015	4,542,874	0.447	0.686	0.646	0.735

- *Educ* denotes an education dummy for some college or above.
- *White*, $18 \leq Age \leq 65$, and Full time workers (30 hours per week)
- $Wage \geq \$62$, which is a half of minimum wage earnings in 1980 ($\$3.10 \times 40\text{hours} \times 1/2$).
- The data cleaning leaves us with 3.6-4.7 million observations besides 2001-2005, where we have around 2.5 million observations.

Regression Model

$$\begin{aligned}\log(Wage_i) =& \beta_0 + \beta_1 Female_i + \beta_2 Educ_i + \beta_3 Female_i \cdot Educ_i \\ &+ \theta_1' X_i + \theta_2'(X_i \cdot Female_i) + \varepsilon_i,\end{aligned}$$

- For control variable X_i, we use 12 age group dummies with a four-year interval, 51 states dummies (including D.C.), and their interactions. Note that $(X_i \cdot Female_i)$ implies that there exist up to 3-way interactions.
- The model contains 1226 covariates in total.
- We also add 4 additional year dummies for the 5-year combined samples after 2001.

Figure: College Wage Premium: Combining 5-Year Data

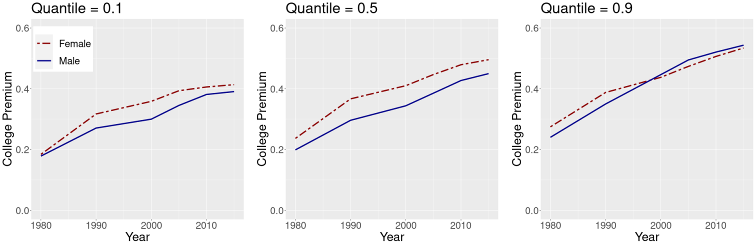


Figure: Difference of College Premium: Combining 5-Year Data

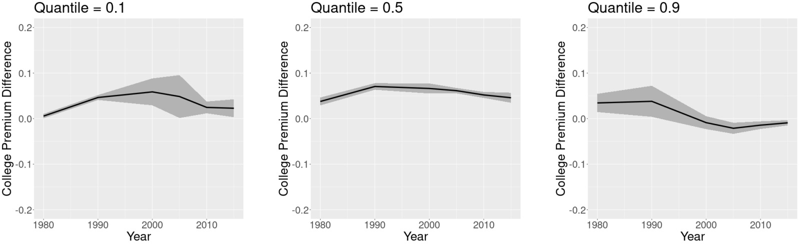


Table: College Wage Premium: $\tau = 0.5$

Year	Female	Male	Difference	Time (min.)
$\underline{\tau = 0.5}$				
1980	0.2365	0.1988	0.0377	29.4
	[0.2294,0.2435]	[0.1945,0.2030]	[0.0291,0.0463]	
1990	0.3667	0.2962	0.0705	34.2
	[0.3603,0.3732]	[0.2942,0.2982]	[0.0634,0.0777]	
2000	0.4101	0.3439	0.0662	36.7
	[0.4056,0.4146]	[0.3372,0.3506]	[0.0552,0.0772]	
2001-2005	0.4468	0.3854	0.0613	20.2
	[0.4369,0.4567]	[0.3765,0.3944]	[0.0554,0.0673]	
2006-2010	0.4791	0.4271	0.0520	47.7
	[0.4748,0.4834]	[0.4174,0.4368]	[0.0454,0.0585]	
2011-2015	0.4957	0.4498	0.0458	46.0
	[0.4887,0.5027]	[0.4455,0.4542]	[0.0348,0.0568]	

Conclusion

- We provide a new scalable on-line inference method for quantile regression with "ultra-large" sample sizes and a large number of covariates.
- It is efficient in terms of both computation time and the memory usage.
- Simulations and the empirical application shows that the S-subGD method could open a new realm of empirical studies.
- We are currently working on:
    - Endogenous regressor (GMM or Structural models)
    - Cluster robust inference (federated learning)
    - High-dimensional settings

thank
you