Penalized Quantile Regression: Concepts, Properties and Algorithm

Zi-Yi ZHANG, Zhong Ruijuan

Report Summary

Quantile regression is a robust analyzing tool. In this report, we first go through the basics of quantile regression, then turn to its usage in high dimensional cases, we review the theoretical results like convergence rate and oracle property of penalized (composite) quantile regression (PQR and CQR in short, respectively). Lasso PQR solution path is also checked. After that, we list the advantages and shortcomings of several algorithms. Finally, we briefly review tuning parameter selection and simulation in PQR/CQR.

Quantile Regression Overview

Introduction

Nowadays, quantile regression is becoming a more and more popular tool in research fields like economics and survival analysis, one famous example is using quantile regression to analyze infants' low birth weight. Besides, compared to traditional mean regression problems (ordinary least square problems), quantile regression makes no assumptions about the distribution of the target variable, and it tends to resist the influence of outlying observations.

Let us introduce some basic concepts, starting with the univariate unconditional quantile. Koenker and Basset (1978) proved that: if we define τth quantile for a random variable Y as $Q_Y(\tau) = \inf\{y : F_Y(y) \geq \tau\}$, then $Q_Y(\tau) = \arg\min_t \mathrm{E}\left[\rho_\tau(Y-t)\right]$, where the loss ρ_τ is called the check loss function:

$$\rho_\tau(u) = u\{\tau - I(u<0)\} = \tau u^+ + (1-\tau)u^- \tag{1}$$

The result in the median case may be more popular: $m = \arg\min_t E|X-t|$. The figure below shows the check loss function for τ = 0.75. From the figure we can see that check loss plays a role in distributing weights, for example when τ = 0.75, it puts more attention to the right positive side. When τ = 0.5, the check loss becomes the absolute value $L1$-loss.

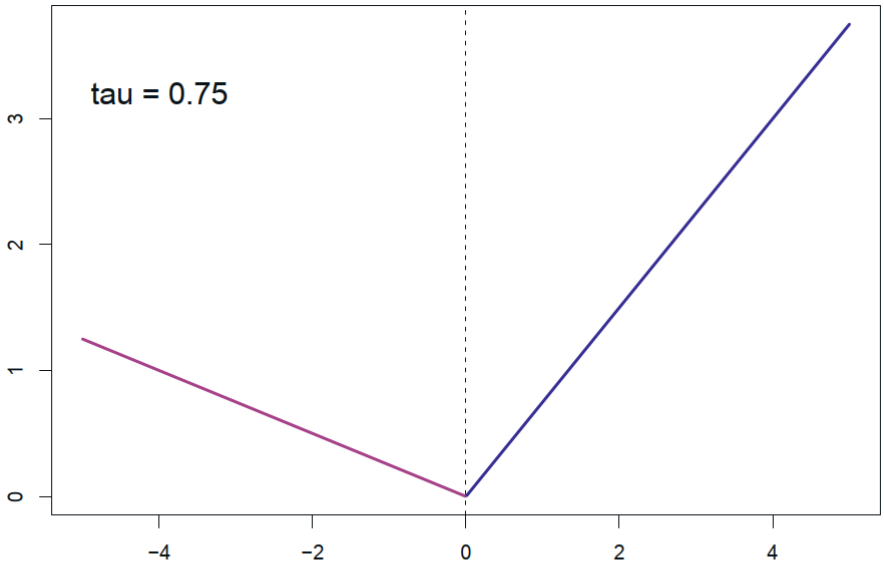


Figure 6.1: The check loss function with $\tau = 0.75$.

It is natural to extend the quantile concept to regression field. Consider a response variable Y and a vector of covariates $\mathbf{X} = (X_1, \ldots, X_p)^T$, now τth conditional quantile is defined as $Q_Y(\tau \mid \mathbf{x}) = \inf \{y : F_Y(y \mid \mathbf{X} = \mathbf{x}) \geq \tau\}$, where $F_Y(y \mid \mathbf{X})$ is the conditional cumulative distribution function, then it will have:

$$Q_Y(\tau \mid \mathbf{x}) = \arg\min_t \mathrm{E}\left[\rho_\tau(Y - t) \mid \mathbf{X} = \mathbf{x}\right] \tag{2}$$

Consider the linear quantile regression case in which $Q_Y(\tau \mid \mathbf{x})$ is a linear function of $\mathbf{x}$:

$$Q_Y(\tau \mid \mathbf{x}) = \mathbf{x}^T \boldsymbol{\beta}_\tau \tag{3}$$

Replace (2)'s expectation with the sample mean, one can estimate β^* with samples:

$$\widehat{\boldsymbol{\beta}}_\tau = \arg\min_{\boldsymbol{\beta}} \frac{1}{n} \sum_{i=1}^{n} \rho_\tau \left(Y_i - \mathbf{X}_i^T \boldsymbol{\beta}\right) \tag{4}$$

Methodology

Since the check loss is non-smooth at the origin, the Newton-Raphson algorithm and its variants cannot be used directly. A standard method for solving the quantile regression problem is to recast the corresponding optimization problem into linear programming, which can then be solved by many existing optimization software packages.

The original problem (4) is equivalent to:

$$\min_{b \in \Re^p} \sum^{n} \rho_\tau \left(y_i - x_i' b\right) \qquad \min_{b \in \Re^p} \sum_{i=1}^{n} \rho_\tau \left(y_i - x_i' b\right) \tag{5}$$

$$\min_{b \in \mathbb{R}^p} \sum_{i=1}$$

By making use of the second form of check loss in (1), one can get the following linear programming problem:

$$\min_{u,v} \left\{ \tau e'u + (1-\tau)e'v \mid y = Xb + u - v, (u,v) \in \Re_+^{2n} \right\} \tag{6}$$

where e denotes an n-vector of ones.

Koenker and Ng (2005) showed how to apply the interior-point method in quantile regression. Alternatively, Hunter and Lange (2000) proposed a majorization-minimization (MM) algorithm for doing that.

Property: Asymptotic Normality

Theoretical properties of linear quantile regression have been well studied in Koenker (2005). Asymptotic normality is one of them, suppose that $\{\mathbf{X}_i, Y_i\}_{i=1}^n$ is a random sample from a linear model: $Y = \mathbf{X}^T\boldsymbol{\beta}^* + \epsilon$, $\epsilon \sim f(\cdot)$ and is independent of $\mathbf{X}$, $\lim_{n\to\infty} \frac{1}{n}\mathbf{X}^T\mathbf{X} = \boldsymbol{\Sigma}$, under these assumptions along with other weak regularity conditions, it has been shown (Koenker, 2005):

> ***Theorem 1***
>
> $$\sqrt{n}\left(\widehat{\boldsymbol{\beta}}_\tau - \boldsymbol{\beta}^*\right) \xrightarrow{D} N\left(0, \frac{\tau(1-\tau)}{f^2\left(b_\tau^*\right)}\boldsymbol{\Sigma}^{-1}\right) \tag{7}$$
>
> *where b_τ^* is the τ-quantile of the error distribution.*

Under the same setting, if $E(\epsilon) = 0$ and $Var(\epsilon) = \sigma^2$, it is known that the corresponding least squares estimator has the following asymptotic normality:

> ***Theorem 2***
>
> $$\sqrt{n}\left(\widehat{\boldsymbol{\beta}}^{\text{ols}} - \boldsymbol{\beta}^*\right) \xrightarrow{D} N\left(0, \sigma^2\boldsymbol{\Sigma}^{-1}\right) \tag{8}$$

However, comparing these two results shows the robustness of quantile regression: quantile regression does not need assumptions of finite variance. It means even in extreme cases like error follows Cauchy distribution ($Var(\epsilon) = \infty$), (7) still holds, provided that $f\left(b_\tau^*\right)$ is not zero.

Properties of PQR Estimator

The penalized least squares method can be naturally extended to penalized quantile regression, actually they are all included in a general form (Fan, Xue and Zou 2014):

$$\min_{\boldsymbol{\beta}} \ell_n(\boldsymbol{\beta}) + P_\lambda(|\boldsymbol{\beta}|) \tag{9}$$

with $\ell_n(\boldsymbol{\beta})$ is a convex loss that does not need to be differentiable and $P_\lambda(|\boldsymbol{\beta}|)$ is a penalty term, this setting covers many important statistical models, $\ell_n(\boldsymbol{\beta})$ can be the squared error loss in penalized least squares, while in the PQR problem, it is the check loss.

$L1$/Lasso Penalty

Belloni and Chernozhukov (2011) studied the $L1$ penalized quantile regression (with $P_\lambda(|\boldsymbol{\beta}|)$ being $L1$ penalty) when $p \gg n$, following their notation, the $L1$ quantile regression is defined as:

$$\widehat{\boldsymbol{\beta}}^{\text{lasso}} = \arg\min_{\boldsymbol{\beta}} \frac{1}{n} \sum_{i=1}^{n} \rho_\tau \left(Y_i - \mathbf{X}_i^T \boldsymbol{\beta}\right) + \frac{\lambda}{n} \sum_{j=1}^{n} \widehat{\sigma}_j \left|\beta_j\right| \tag{10}$$

where $\widehat{\sigma}_j^2 = \frac{1}{n} \sum_{i=1}^{n} X_{ij}^2$. Dividing the penalty level to n is to make PQR converge (the original lasso estimator does not), and multiplying $\widehat{\sigma}_j$ because in practice design matrix will be normalized. This formula can be viewed as a special lasso penalty case of (9).

Near-oracle Convergence Rate

Suppose that $Q_Y(\tau \mid \mathbf{x}) = \mathbf{x}^T \boldsymbol{\beta}^*$ and $\left\|\beta^*\right\|_0 = s$. Let $\mathcal{S}$ denote the support of β^*. Then under several conditions:

> ***Conditions***
>
> *D.1 (Regular condition): Assume that the conditional density $f(y|x)$ is continuously differentiable in y and upper bounded by a constant $\bar{f}$. In addition, assume that $f(y|x)$ evaluated at its τ-quantile is bounded away from zero, i.e. $f(\mathbf{x}^T \boldsymbol{\beta}^* \mid \mathbf{x}) > \underline{f} > 0$ uniformly, and $\frac{\partial f(y|\mathbf{x})}{\partial y}$ is upper bounded by a constant $\bar{f}'$.*
>
> *D.2 (Well-behaved covariates):* $P\left(\max_j \left|\widehat{\sigma}_j - 1\right| \leq 1/2\right) \geq 1 - \gamma \to 0$.

D.3 (Restricted identifiability and nonlinearity):

For a vector $\mathbf{v}$ in $\mathbb{R}^p$ and $\mathcal{S}'$ any subset of $\{1,\ldots,p\}$, $\mathbf{v}_{\mathcal{S}'}$ is a vector in $\mathbb{R}^p$ such that $(\mathbf{v}_{\mathcal{S}'})_j = v_j$ if $j \in \mathcal{S}'$ and $(\mathbf{v}_{\mathcal{S}'})_j = 0$ if $j \notin \mathcal{S}'$. A restricted set is defined as

$$\mathcal{A} = \{\mathbf{v} \in \mathbb{R}^p : \|\mathbf{v}_{\mathcal{S}^c}\|_1 \leq c_0 \|\mathbf{v}_{\mathcal{S}}\|_1, \|\mathbf{v}_{\mathcal{S}^c}\|_0 \leq n\}. \tag{6.8}$$

It can be shown that with a high probability $\widehat{\boldsymbol{\beta}}^{\text{lasso}} - \boldsymbol{\beta}^* \in \mathcal{A}$. To bound $\widehat{\boldsymbol{\beta}}^{\text{lasso}} - \boldsymbol{\beta}^*$ additional regularity conditions are required. Define $\bar{\mathcal{S}}(\mathbf{v}, m) \subset \{1,\ldots,p\} \setminus \mathcal{S}$ as the support of the m largest in absolute value elements of $\mathbf{v}$ outside $\mathcal{S}$. Assume that for some $m > 0$ we have

$$\kappa_m^2 = \inf_{\mathbf{v}\in\mathcal{A},\mathbf{v}\neq 0} \frac{\mathbf{v}^T \mathbf{E}[\mathbf{X}\mathbf{X}^T]\mathbf{v}}{\|\mathbf{v}_{\mathcal{S}\cup\bar{\mathcal{S}}(\mathbf{v},m)}\|^2} > 0 \tag{6.9}$$

$$q_m = \frac{3}{8}\frac{\bar{f}^{3/2}}{\bar{f}'} \inf_{\mathbf{v}\in\mathcal{A},\mathbf{v}\neq 0} \frac{\mathrm{E}[|\mathbf{X}^T\mathbf{v}|^2]^{3/2}}{\mathrm{E}[|\mathbf{X}^T\mathbf{v}|^3]} > 0 \tag{6.10}$$

The condition (6.9) is similar to the restricted eigenvalue condition for the ℓ_1 least squares estimator.

Belloni and Chernozhukov proved that under D.1-D.3, lasso PQR shows the near-oracle convergence rate:

Theorem 6.1 *If we choose $\lambda_0(\alpha) \leq \lambda \leq C\sqrt{s\log(p)/n}$ for some constant C, where $\lambda_0(\alpha)$ only depends on α, with probability at least $1 - 4\gamma - \alpha - 3p^{-b^2}$,*

$$\|\widehat{\boldsymbol{\beta}}^{\text{lasso}} - \boldsymbol{\beta}^*\| \leq \frac{1 + c_0\sqrt{s/m}}{\kappa_m} \frac{8C(1+c_0)}{\underline{f}\kappa_0\sqrt{\tau(1-\tau)}} b\sqrt{\frac{s\log p}{n}},$$

where b is any constant $b > 1$.

The rate is related to τ, which is as expected, extreme quantiles (close to 0 or 1) can slow down the rates of convergence.

Model Selection Consistency

Besides, they also analyzed the model selection properties:

3.4. *Model selection properties.* Next, we turn to the model selection properties of ℓ_1-QR.

THEOREM 4 (Model selection properties of ℓ_1-QR). *Let* $r^o = \sup_{u\in\mathcal{U}} \|\widehat{\beta}(u) - \beta(u)\|$. *If* $\inf_{u\in\mathcal{U}} \min_{j\in T_u} |\beta_j(u)| > r^o$, *then*

$$T_u := \text{support}(\beta(u)) \subseteq \widehat{T}_u := \text{support}(\widehat{\beta}(u)) \qquad \text{for all } u \in \mathcal{U}. \tag{3.15}$$

Moreover, the hard-thresholded estimator $\bar{\beta}(u)$, *defined for any* $\gamma \geq 0$ *by*

$$\bar{\beta}_j(u) = \widehat{\beta}_j(u) 1\{|\widehat{\beta}_j(u)| > \gamma\}, \qquad u \in \mathcal{U}, j = 1, \ldots, p, \tag{3.16}$$

provided that γ *is chosen such that* $r^o < \gamma < \inf_{u\in\mathcal{U}} \min_{j\in T_u} |\beta_j(u)| - r^o$, *satisfies*

$$\text{support}(\bar{\beta}(u)) = T_u \qquad \text{for all } u \in \mathcal{U}.$$

The first result says that if nonzero coefficients are well separated from zero, then the support of $L1$ quantile regression includes the support of the true model, but the support of the estimator can include some unnecessary components having true coefficients equal to zero. The second result states that if the further conditions are satisfied, additional hard thresholding can eliminate inclusions of such unnecessary components. The value of the hard threshold must explicitly depend on the unknown value $\min_{j\in T_u} |\beta_j(u)|$, characterizing the separation of nonzero coefficients from zero. The additional conditions stated in this theorem are strong and perfect model selection appears **quite unlikely in practice**.

SCAD Penalty: Strong Oracle Property

When the penalty becomes the SCAD, we have the PQR form:

$$\widehat{\boldsymbol{\beta}}^{\text{scad}} = \arg\min_{\beta} \frac{1}{n} \sum_{i=1}^{n} \rho_\tau \left(Y_i - \mathbf{X}_i^T \boldsymbol{\beta}\right) + \sum_{j=1}^{p} p_\lambda \left(|\beta_j|\right) \tag{11}$$

where $P_\lambda(|\boldsymbol{\beta}|)$ now represents SCAD penalty function with the regularization parameter λ.

An oracle knows the true support set (but does not know true parameter values) and defines the oracle estimator as:

$$\hat{\boldsymbol{\beta}}^{\text{oracle}} = \left(\hat{\boldsymbol{\beta}}_{\mathcal{S}}^{\text{oracle}}, \mathbf{0}\right) = \arg\min_{\boldsymbol{\beta}:\boldsymbol{\beta}_{\mathcal{S}^c}=\mathbf{0}} \ell_n(\boldsymbol{\beta}) \tag{12}$$

The oracle estimator is not a feasible estimator but it can be used as a theoretic benchmark for other estimators to compare with. An estimator is said to have the oracle property if it has the same asymptotic distribution as the oracle estimator (Fan and Li, 2001; Fan and Peng, 2004). Moreover, an estimator is said to have the strong oracle property if the estimator equals the oracle estimator with overwhelming probability (Fan and Lv, 2011).

Problem (12) can be iteratively reweighted and solved by local linear approximation (LLA, Zou and Li, 2008), and Fan, Xue and Zou (2014) showed how to construct an estimator with the strong oracle property through LLA:

> ***Theorem** 3*
>
> *Under the regularity conditions for the $L1$ penalized quantile regression for Theorem 6.1, with a high probability (approaches to 1), the LLA algorithm initialized by $\widehat{\boldsymbol{\beta}}^{\text{lasso}}$ converges to $\widehat{\boldsymbol{\beta}}^{\text{oracle}}$ after two iterations.*

We only provide the proof ideas here, for details, please refer to the original paper.

First Fan, Xue and Zou proved that as long as the problem is localizable and the oracle estimator is well behaved, one can obtain the oracle estimator by using the one-step LLA, which mainly means, a good initialization can approach to oracle estimator by LLA in one step. Then they showed that once the oracle estimator is obtained, the LLA algorithm converges, namely it will be trapped in the same oracle estimator in following iterations. The second result simultaneously reflects the equivalence between the two-step and fully converged LLA solutions. The above two results are the main contributions of that paper, then in the case of sparse quantile regression, the authors specified constant values (for computing probabilities), and showed that $\widehat{\boldsymbol{\beta}}^{\text{lasso}}$ is a good starting point that will converge after two iterations.

Lasso PQR Solution Path

In 2008, inspired by the LARS/lasso (Efron et al. 2004) algorithm Li and Zhu draw PQR solution path of the lasso penalized quantile regression. The basic idea is to solve the problem's KKT conditions, and keep tracking the change of KKT conditions in the process.

Unlike Belloni and Chernozhukov (2011), Li and Zhu considered directly adding lasso penalty because they wanted to compare result with the original lasso paper (Tibshirani 1996), so their goal is:

$$\begin{aligned}
&\min_{\beta_0,\boldsymbol{\beta}} \tau \sum_{i=1}^{n} \xi_i + (1-\tau) \sum_{i=1}^{n} \zeta_i, \\
&\text{subject to} \quad \sum_{j=1}^{p} |\beta_j| \le s, \\
&\qquad -\zeta_i \le y_i - f(\boldsymbol{x}_i) \le \xi_i, \\
&\qquad \zeta_i, \xi_i \ge 0, \quad i = 1, \ldots, n,
\end{aligned}$$

$$\min_{\beta_0,\boldsymbol{\beta}} \sum_{i=1}^{n} \rho_\tau \left(y_i - \beta_0 - \boldsymbol{\beta}^\top \boldsymbol{x}_i \right) \tag{13}$$

subject to $|\beta_1| + \cdots + |\beta_p| \le s$

$$\text{subject to } |\rho_1| + \cdots + |\rho_p| \leq s$$

which can be rewritten equivalently:

$$\begin{aligned} &\min_{\beta_0, \boldsymbol{\beta}} \tau \sum_{i=1}^{n} \xi_i + (1-\tau) \sum_{i=1}^{n} \zeta_i, \\ \text{subject to} \quad &\sum_{j=1}^{p} |\beta_j| \leq s, \\ &-\zeta_i \leq y_i - f(\boldsymbol{x}_i) \leq \xi_i, \\ &\zeta_i, \xi_i \geq 0, \quad i = 1, \ldots, n, \end{aligned} \tag{14}$$

where $f(\boldsymbol{x}_i) = \beta_0 + \boldsymbol{\beta}^\top \boldsymbol{x}_i$.

Corresponding Lagrangian primal function:

$$\begin{aligned} L_p :& \tau \sum_{i=1}^{n} \xi_i + (1-\tau) \sum_{i=1}^{n} \zeta_i + \lambda^* \left(\sum_{j=1}^{p} |\beta_j| - s \right) + \sum_{i=1}^{n} \alpha_i \left(y_i - f(\boldsymbol{x}_i) - \xi_i \right) \\ &- \sum_{i=1}^{n} \gamma_i \left(y_i - f(\boldsymbol{x}_i) + \zeta_i \right) - \sum_{i=1}^{n} \kappa_i \xi_i - \sum_{i=1}^{n} \eta_i \zeta_i \end{aligned} \tag{15}$$

Setting derivative to 0:

$$\begin{aligned} &\frac{\partial}{\partial \boldsymbol{\beta}} : \lambda^* \cdot \operatorname{sign}(\beta_j) - \sum_{i=1}^{n} (\alpha_i - \gamma_i) x_{ij} = 0, \quad \forall j \text{ with } \beta_j \neq 0 \\ &\frac{\partial}{\partial \beta_0} : \sum_{i=1}^{n} (\alpha_i - \gamma_i) = 0 \\ &\frac{\partial}{\partial \xi_i} : \tau = \alpha_i + \kappa_i \\ &\frac{\partial}{\partial \zeta_i} : 1 - \tau = \gamma_i + \eta_i \end{aligned} \tag{16}$$

and the KKT conditions (complementarity slackness conditions) are:

$$\begin{aligned} \alpha_i \left(y_i - f(\boldsymbol{x}_i) - \xi_i \right) &= 0 \\ \gamma_i \left(y_i - f(\boldsymbol{x}_i) + \zeta_i \right) &= 0 \\ \kappa_i \xi_i &= 0 \\ \eta_i \zeta_i &= 0. \end{aligned} \tag{17}$$

Since Lagrange multipliers are non-negative, from equations of $\frac{\partial}{\partial \xi_i}$, $\frac{\partial}{\partial \zeta_i}$ and above KKT conditions one can conclude that:

$$\begin{array}{lllllll} y_i - f(\boldsymbol{x}_i) > 0 & \Rightarrow & \alpha_i = \tau, & \xi_i > 0, & \gamma_i = 0, & \zeta_i = 0; \\ y_i - f(\boldsymbol{x}_i) < 0 & \Rightarrow & \alpha_i = 0, & \xi_i = 0, & \gamma_i = 1 - \tau, & \zeta_i > 0; \\ y_i - f(\boldsymbol{x}_i) = 0 & \Rightarrow & \alpha_i \in [0, \tau], & \xi_i = 0, & \gamma_i \in [0, 1-\tau], & \zeta_i = 0. \end{array} \tag{18}$$

Notice that in KKT conditions only $\alpha_i - \gamma_i$ matters, denote it as θ_i, and then define four sets that are relative to KKT conditions changing:

- $\mathcal{E} = \{i : y_i - f(\boldsymbol{x}_i) = 0, -(1-\tau) \leq \theta_i \leq \tau\}$ (elbow)
- $\mathcal{L} = \{i : y_i - f(\boldsymbol{x}_i) < 0, \theta_i = -(1-\tau)\}$ (left of the elbow)
- $\mathcal{R} = \{i : y_i - f(\boldsymbol{x}_i) > 0, \theta_i = \tau\}$ (right of the elbow)

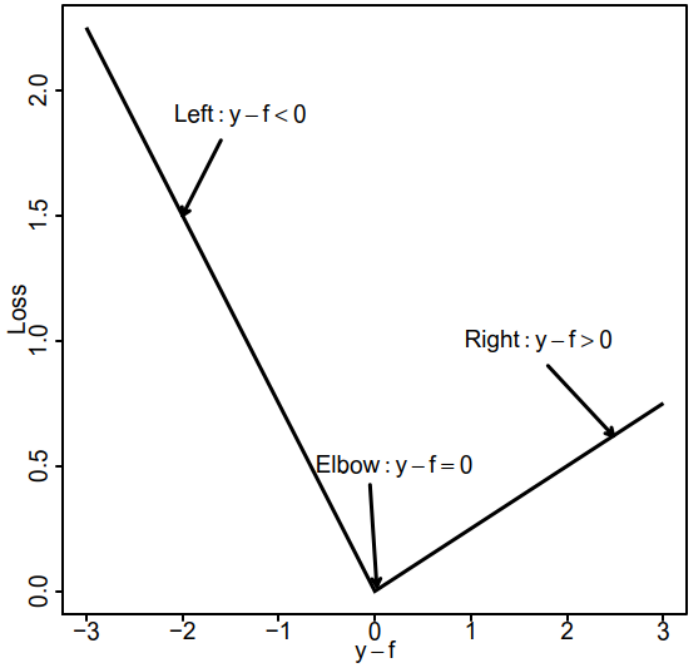


Figure 1. The check function with $\tau = 0.25$. We divide data points into three sets based on their associated residuals $y_i - f(\boldsymbol{x}_i)$. The three sets are left, elbow, and right.

- $\mathcal{V} = \{j : \beta_j \neq 0\}$ (active set)

Since the goal is to compute the solution path $\boldsymbol{\beta}(s)$, we are interested in how KKT conditions change when the regularization parameter s increases. The authors defined an *event* to be:

- either a data point hits the elbow, that is, a residual $y_i - \beta_0 - \boldsymbol{\beta}^\top \boldsymbol{x}_i$ changes from nonzero
  to zero, or
- a coefficient β_j changes from nonzero to zero, that is, a variable leaves the active set, V.

The two cases actually correspond to the non-smooth points of $\sum_{i=1}^{n} \rho_\tau \left(y_i - \beta_0 - \boldsymbol{\beta}^\top \boldsymbol{x}_i\right)$ and $||\beta||_1$ respectively. Note that it is also possible for a residual to change from zero to nonzero, or a coefficient to change from zero to nonzero, but those situations are similar so we omit them.

Given the above definition of the events, we can see:

- As s increases, the sets $\mathcal{V}, \mathcal{L}, \mathcal{R}$, and $\mathcal{E}$ will not change (or equivalently, the KKT conditions will not change), unless an event happens. When the KKT conditions do not change, from the first two equations in (16), there are $|\mathcal{E}| + 1$ unknowns and $|\mathcal{V}| + 1$ equations. For the solution to be unique, we must have the number of observations in the elbow equal to the number of variables in the active set, that is, $|\mathcal{E}| = |\mathcal{V}|$.
- As s increases, points in $\mathcal{E}$ stay in the elbow, unless an event happens. Therefore, nonzero β_j 's satisfy:

$$y_i - \left(\beta_0 + \sum_{j \in \mathcal{V}} \beta_j x_{ij}\right) = 0 \quad \text{for} \quad i \in \mathcal{E} \tag{19}$$

  Since $|\mathcal{E}| = |\mathcal{V}|$, there is one free unknown in this set of equations, which allows $\boldsymbol{\beta}$ to change when s increases, unless an event happens.

The basic idea of the algorithm is as follows: starting with s= 0 and increasing it, keeping track of the location of all data points relative to the elbow and also of the magnitude of the fitted coefficients along the way. As s increases, for a point to pass through $\mathcal{E}$, the corresponding θ_i must change from τ to $-(1-\tau)$ or vice versa, hence by continuity, points in $\mathcal{E}$ must linger in the elbow. Since all points in the elbow have $y_i - \left(\beta_0 + \sum_{j \in \mathcal{V}} \beta_j x_{ij}\right) = 0$, we can establish a path for $\boldsymbol{\beta}$. The elbow set will stay stable until either some other point comes to the elbow or one nonzero fitted coefficient has dropped to zero.

Then authors consider different choices of starting points which we omit here.

Now use the subscript ℓ to index the sets above immediately after the ℓth event has occurred, and let β_0^ℓ, β_j^ℓ, and s^ℓ be the parameter values immediately after the ℓth event.

- In the first case, $s^\ell < s < s^{\ell+1}$, **no event happens**, the KKT conditions are clear and one can get a close form for $\boldsymbol{\beta}$ (omit details in paper):

$$\begin{aligned} \beta_0 &= \beta_0^\ell + \left(s - s^\ell\right) v_0, \\ \beta_j &= \beta_j^\ell + \left(s - s^\ell\right) v_j, \quad \forall j \in \mathcal{V}^\ell. \end{aligned} \tag{20}$$

  where v_0, v_j are the update size that needed to be calculated. From (20) we can see that β_0^ℓ, β_j^ℓ change linearly unless an event happens.
- Another case is when **an event happens**, authors said "there will be $|\mathcal{V}|$ variables with nonzero coefficients and $|\mathcal{V}| + 1$ points in the elbow", and thus "to maintain the KKT conditions, we need to either add a variable not in $\mathcal{V}$ into $\mathcal{V}$, or remove a point in $\mathcal{E}$ from $\mathcal{E}$". The choice will result in that the sample sum of check loss decreasing with the fastest rate, and they calculated the fastest rate: $\frac{\Delta \text{loss}}{\Delta s} = -\lambda^*$, where λ^* is the parameter in (16).

After these analyses, we get the solution path for lasso PQR:

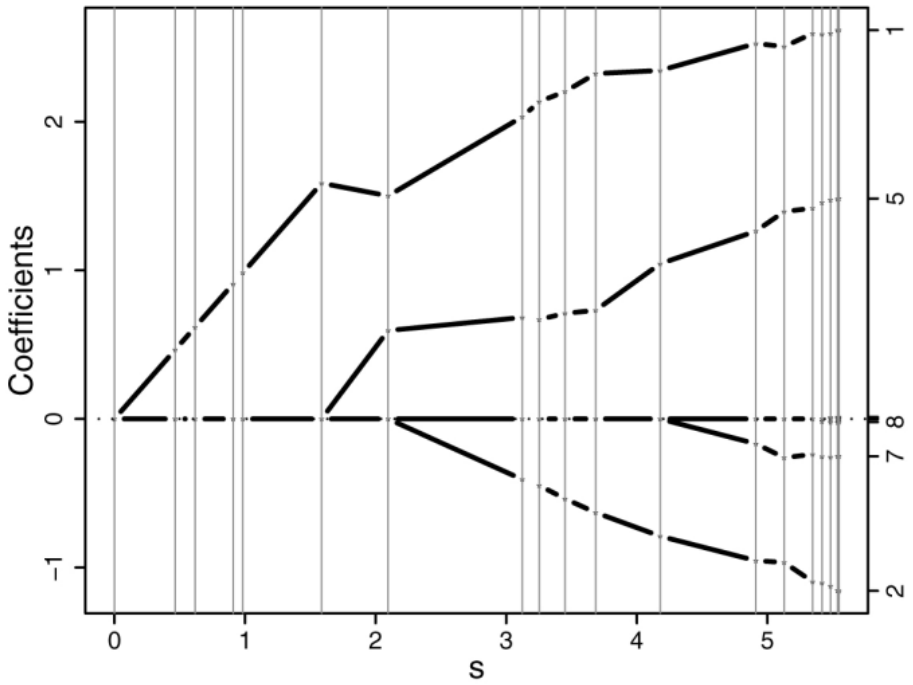


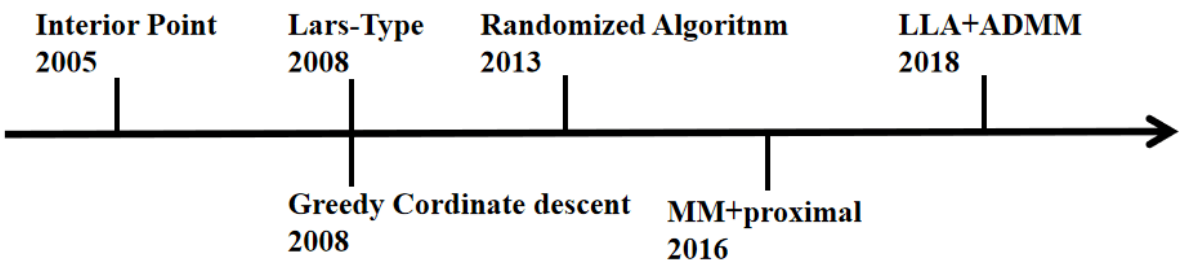
Figure 2. The solution path $\boldsymbol{\beta}(s)$ as a function of s. Any segment between two adjacent vertical lines is linear, hence the whole solution path is piecewise linear. The indices of predictor variables are labeled on the right side axis. Predictors 1, 2, and 5 are relevant variables, and the corresponding true coefficients are 3, -1.5, and 2, respectively. As we can see, over a range of s, only these three predictors have nonzero fitted coefficients.

The plot is piecewise linear as a function of s. Notice that a shortage of the plot is that it is in $p < n$ case, despite this, the solution path helps us understand PQR.

Algorithm for Solving PQR

Algorithm Review

The main hardness of solving PQR is that it uses a non-smooth loss function: check loss.



This picture shows the development of algorithms solving PQR problems, interior point method (Koenker and Ng, 2005) can handle both quantile regression and its penalized version, but it is very inefficient for large scale problems with large p. The LARS-type algorithm (Li and Zhu, 2008) is the algorithm for constructing the solution path above, it is very efficient for solving the lasso linear regression, but it loses its efficiency when the

squared error loss is replaced with the non-smooth check loss. More recently, Wu and Lange (2008) suggested using a greedy coordinate descent algorithm for solving sparse penalized quantile regression, which is similar to Peng and Wang (2015). Coordinate descent is very successful in solving the lasso penalized least squares (Friedman et al. 2010). However, when the loss function is non-smooth and non-separable (which is the case for quantile regression), the coordinate descent algorithm does not have a theoretical guarantee to give the right solution. Here is a simple example to demonstrate this point. Consider quantile regression ($\tau = 0.5$) with a 'fake' dataset:

$$\begin{array}{ccc} y & x_1 & x_2 \\ 0 & 1 & 1 \\ 0 & 1 & -1 \\ 0 & 1 & -1 \end{array} \tag{21}$$

the corresponding optimization problem is:

$$\arg\min_{\beta_1,\beta_2} \frac{1}{2}|\beta_1 + \beta_2| + |\beta_1 - \beta_2| \tag{22}$$

Obviously, the minimizer is $\beta_1 = \beta_2 = 0$. It can be directly shown that the cyclic coordinate descent algorithm is trapped at the initial value (for example, one can initialize $\beta_1 = \beta_2 = 1$ and try the simple gradient descent method to find the next updated value).

Lv, He, and Wang (2016) developed their algorithm by combining the MM technique in Hunter and Lange (2000) and the proximal gradient method (Parikh and Boyd 2013). Yang, Meng, and Mahoney (2013) considered a randomized algorithm for solving large-scale quantile regression with small to moderate dimensions.

We can see that many algorithms useful in linear regression do not work well in high dimensional quantile regression cases, an fast algorithm in large-scale high dimension cases is needed.

ADMM-based Algorithm

In 2018, Gu, Fan and Kong et al. proposed a proximal ADMM (pADMM) algorithm and a sparse coordinate descent ADMM (scdADMM) algorithm to solve the penalized quantile regression with a variety of penalties (lasso, adaptive lasso, and folded concave penalties, etc.).

The authors reviewed the interior point algorithm (Koenker and Ng, 2005), it fails to scale well with high dimensions. Also, they noticed that Yi and Huang (2016) built a coordinate descent algorithm to solve the penalized Huber regression and used its solutions to approximate those of the penalized quantile regression. It is worth mentioning that both the interior point algorithm and their ADMM-based algorithm solve the exact quantile regression problem in theory, while the coordinate descent method offers an approximate solution.

Recall that a general PQR is in the form:

$$\min_{\boldsymbol{\beta}} \frac{1}{n} \sum_{i=1}^{n} \rho_\tau \left(Y_i - \mathbf{X}_i^T \boldsymbol{\beta}\right) + \sum_{j=1}^{p} p_\lambda \left(|\beta_j|\right) \tag{23}$$

This form can be converted to the following weighted $L1$-PQR problem by LLA (Zou and Li, 2008) algorithm:

$$\min_{\boldsymbol{\beta}} \frac{1}{n} \sum_{i=1}^{n} \rho_\tau \left(y_i - \mathbf{x}_i^{\mathrm{T}} \boldsymbol{\beta}\right) + \lambda \|\mathbf{w} \circ \boldsymbol{\beta}\|_1 \tag{24}$$

where $\|\mathbf{w} \circ \boldsymbol{\beta}\|_1 = \sum_{j=1}^{p} |w_j \beta_j| = \sum_{j=1}^{p} w_j |\beta_j|$, for example, for lasso penalty: $\mathbf{w} = \mathbf{1}_p$, while for classical adaptive Lasso penalty: $w_j = \left(\left|\hat{\beta}_j^{\text{Lasso}}\right| + 1/n\right)^{-1}$.

Thus author put their attention on how to efficiently solve (24), once this is done, a general PQR problem (23) can be first transferred to form (24) and then be solved, via the following LLA iterations:

$$\min_{\boldsymbol{\beta}} \frac{1}{n} \sum_{i=1}^{n} \rho_\tau \left(y_i - \mathbf{x}_i^{\mathrm{T}} \boldsymbol{\beta}\right) + \sum_{j=1}^{p} p_\lambda(|\beta_j|),$$

via the following iterations:

(a) Initialize $\boldsymbol{\beta}$ with $\widehat{\boldsymbol{\beta}}^0$.

(b) For $k = 1, 2, \ldots, M$,

(b.1) Compute the weights $w_j = \hat{w}_j^{k-1} = \lambda^{-1} p'_\lambda(|\hat{\beta}_j^{k-1}|)$, $j = 1, \ldots, p$.

(b.2) Solve problem (1) using the weights from step (b.1) to obtain the update $\widehat{\boldsymbol{\beta}}^k$.

Their "problem (1)" is just our (24), so now focus on (24), denote $ \mathbb{Q}_{\tau}(\mathbf{z})=(1 / n) \sum_{i=1}^{n} \rho_{\tau}\left(z_{i}\right) $ for $\mathbf{z} = (z_1, \ldots, z_n)^{\mathrm{T}}$, then our problem is, then by convexity, (24) is equivalent to:

$$\begin{array}{ll} \min_{\boldsymbol{\beta},\mathbf{z}} & \mathbb{Q}_\tau(\mathbf{z}) + \lambda \|\mathbf{w} \circ \boldsymbol{\beta}\|_1 \\ \text{subject to} & \mathbf{X}\boldsymbol{\beta} + \mathbf{z} = \mathbf{y}. \end{array} \tag{25}$$

$$\text{subject to } \mathbf{X}\boldsymbol{\beta} + \mathbf{z} = \mathbf{y}.$$

Fix $\sigma > 0$ and the augmented Lagrangian function is:

$$\mathcal{L}_\sigma(\boldsymbol{\beta}, \mathbf{z}, \boldsymbol{\theta}) := \mathbb{Q}_\tau(\mathbf{z}) + \lambda\|\mathbf{w} \circ \boldsymbol{\beta}\|_1 - \langle \boldsymbol{\theta}, \mathbf{X}\boldsymbol{\beta} + \mathbf{z} - \mathbf{y}\rangle + \frac{\sigma}{2}\|\mathbf{X}\boldsymbol{\beta} + \mathbf{z} - \mathbf{y}\|_2^2 \quad (26)$$

Following Boyd et al. (2011), the iterations for the standard ADMM algorithm are given by:

$$\boldsymbol{\beta}^{k+1} := \arg\min_{\boldsymbol{\beta}} \mathcal{L}_\sigma\left(\boldsymbol{\beta}, \mathbf{z}^k, \boldsymbol{\theta}^k\right) \quad (27)$$

$$\mathbf{z}^{k+1} := \arg\min_{\mathbf{z}} \mathcal{L}_\sigma\left(\boldsymbol{\beta}^{k+1}, \mathbf{z}, \boldsymbol{\theta}^k\right) \quad (28)$$

$$\boldsymbol{\theta}^{k+1} := \boldsymbol{\theta}^k - \sigma\left(\mathbf{X}\boldsymbol{\beta}^{k+1} + \mathbf{z}^{k+1} - \mathbf{y}\right) \quad (29)$$

More specifically, the iterations are:

$$\begin{aligned} \boldsymbol{\beta}\text{ step}: \quad \boldsymbol{\beta}^{k+1} &:= \arg\min_{\boldsymbol{\beta}} \lambda\|\mathbf{w} \circ \boldsymbol{\beta}\|_1 - \left\langle \boldsymbol{\theta}^k, \mathbf{X}\boldsymbol{\beta}\right\rangle + \tfrac{\sigma}{2}\left\|\mathbf{X}\boldsymbol{\beta} + \mathbf{z}^k - \mathbf{y}\right\|_2^2 \\ \mathbf{z}\text{ step}: \quad \mathbf{z}^{k+1} &:= \arg\min_{\mathbf{z}} \mathbb{Q}_\tau(\mathbf{z}) - \left\langle \boldsymbol{\theta}^k, \mathbf{z}\right\rangle + \tfrac{\sigma}{2}\left\|\mathbf{z} + \mathbf{X}\boldsymbol{\beta}^{k+1} - \mathbf{y}\right\|_2^2 \\ \boldsymbol{\theta}\text{ step}: \quad \boldsymbol{\theta}^{k+1} &:= \boldsymbol{\theta}^k - \sigma\left(\mathbf{X}\boldsymbol{\beta}^{k+1} + \mathbf{z}^{k+1} - \mathbf{y}\right). \end{aligned} \quad (30)$$

First notice that update of $\mathbf{z}$ can be carried out component-wisely:

$$\begin{aligned} z_i^{k+1} &: = \arg\min_{z_i} \frac{1}{n}\rho_\tau\left(z_i\right) - \theta_i^k z_i + \frac{\sigma}{2}\left(z_i + \mathbf{x}_i^{\mathrm{T}}\boldsymbol{\beta}^{k+1} - y_i\right)^2 \\ &= \arg\min_{z_i} \rho_\tau\left(z_i\right) + \frac{n\sigma}{2}\left[z_i - \left(y_i - \mathbf{x}_i^{\mathrm{T}}\boldsymbol{\beta} + \frac{1}{\sigma}\theta_i^k\right)\right]^2 \end{aligned} \quad (31)$$

The solution is:

$$z_i^{k+1} = \mathrm{Prox}_{\rho_\tau}\left[y_i - \mathbf{x}_i^{\mathrm{T}}\boldsymbol{\beta} + \frac{1}{\sigma}\theta_i^k, n\sigma\right], \quad i = 1, \ldots, n \quad (32)$$

where $\mathrm{Prox}_{\rho_\tau}[\xi, \alpha] = \begin{cases} \xi - \frac{\tau}{\alpha}, & \text{if } \xi > \frac{\tau}{\alpha} \\ 0, & \text{if } \frac{\tau-1}{\alpha} \le \xi \le \frac{\tau}{\alpha} \\ \xi - \frac{\tau-1}{\alpha}, & \text{if } \xi < \frac{\tau-1}{\alpha} \end{cases}$ (we suggest using subgradient method to get the solution instead of the original paper's way).

Now problem focuses on how to efficiently update $\boldsymbol{\beta}$, there is no closed-form now:

$$\boldsymbol{\beta}^{k+1} := \arg\min_{\boldsymbol{\beta}} \lambda\|\mathbf{w} \circ \boldsymbol{\beta}\|_1 - \left\langle \boldsymbol{\theta}^k, \mathbf{X}\boldsymbol{\beta}\right\rangle + \frac{\sigma}{2}\left\|\mathbf{X}\boldsymbol{\beta} + \mathbf{z}^k - \mathbf{y}\right\|_2^2 \quad (33)$$

Gu, Fan and Kong et al. provided two ways, one named *pADMM*, uses a "linearization" trick (proximal algorithms) to modify the subproblem (33) such that the updated $\boldsymbol{\beta}^{k+1}$ also has a closed-form formula (Parikh and Boyd, 2013):

$$\begin{aligned}\boldsymbol{\beta}^{k+1} =& \arg\min_{\boldsymbol{\beta}} \lambda\|\mathbf{w}\circ\boldsymbol{\beta}\|_1 \\ &+ \frac{\sigma\eta}{2}\left\|\boldsymbol{\beta} - \frac{\sigma\eta\boldsymbol{\beta}^k + \mathbf{X}^{\mathrm{T}}\left(\boldsymbol{\theta}^k + \sigma\mathbf{y} - \sigma\mathbf{X}\boldsymbol{\beta}^k - \sigma\mathbf{z}^k\right)}{\sigma\eta}\right\|_2^2 \\ =& \left(\text{Shrink}\left[\beta_j^k + \frac{1}{\sigma\eta}X_j^{\mathrm{T}}\left(\boldsymbol{\theta}^k + \sigma\mathbf{y} - \sigma\mathbf{X}\boldsymbol{\beta}^k - \sigma\mathbf{z}^k\right), \frac{\lambda w_j}{\sigma\eta}\right]\right)_{1\le j\le p}\end{aligned} \tag{34}$$

where $\text{Shrink}[u,\alpha] = \text{sgn}(u)\max(|u|-\alpha, 0)$.

Another is called sparse coordinate descent ADMM (*scdADMM*), it views (33) as an adaptive Lasso linear regression problem, then uses efficient cyclic coordinate descent algorithm (CCD, Friedman, Hastie and Tibshirani, 2010) to get $\boldsymbol{\beta}^{k+1}$ (CCD works well in least square loss while not in check loss). Authors said that *scdADMM* performs more efficiently when p large.

After solving problem (24) efficiently, LLA can be added up to construct a complete algorithm:

pADMM

Algorithm 1 pADMM – Proximal ADMM algorithm for solving the weighted L_1-penalized quantile regression.

1. Initialize the algorithm with $(\boldsymbol{\beta}^0, \mathrm{z}^0, \boldsymbol{\theta}^0)$.
2. For $k = 0, 1, 2, \ldots,$ repeat steps (2.1)–(2.3) until the convergence criterion is met.

(2.1) Update $\boldsymbol{\beta}^{k+1} \leftarrow$
$$\left(\text{Shrink}\left[\beta_j^k + \tfrac{1}{\sigma\eta}X_j^{\mathrm{T}}(\boldsymbol{\theta}^k + \sigma\mathbf{y} - \sigma\mathbf{X}\boldsymbol{\beta}^k - \sigma\mathbf{z}^k), \tfrac{\lambda w_j}{\sigma\eta}\right]\right)_{1\le j\le p}.$$

(2.2) Update $\mathbf{z}^{k+1} \leftarrow \left(\text{Prox}_{\rho_\tau}\left[y_i - \mathbf{x}_i^{\mathrm{T}}\boldsymbol{\beta}^{k+1} + \sigma^{-1}\theta_i^k, n\sigma\right]\right)_{1\le i\le n}.$

(2.3) Update $\boldsymbol{\theta}^{k+1} \leftarrow \boldsymbol{\theta}^k - \gamma\sigma(\mathbf{X}\boldsymbol{\beta}^{k+1} + \mathbf{z}^{k+1} - \mathbf{y})$.

Algorithm 2 scdADMM – Sparse coordinate descent ADMM algorithm for solving the weighted L_1-penalized quantile regression with coordinate descent steps.

1. Initialize the algorithm with $(\boldsymbol{\beta}^0, \mathbf{z}^0, \boldsymbol{\theta}^0)$.
2. For $k = 0, 1, 2, \ldots$, repeat Steps (2.1)–(2.3) until the convergence criterion is met.

(2.1) Carry out the coordinate descent Steps (2.1.1)–(2.1.3).

(2.1.1) Initialize $\boldsymbol{\beta}^{k,0} = \boldsymbol{\beta}^k$.

(2.1.2) For $m = 0, 1, 2, \ldots$, repeat Step (2.1.2.1) until convergence.

(2.1.2.1) For $j = 1, \ldots, p$, update

$$\beta_j^{k,m+1} \leftarrow \frac{\text{Shrink}\left[\sum_{i=1}^n x_{ij}\left\{\theta_i^k + \sigma\left(y_i - z_i^k - \sum_{t\neq j} x_{it}\beta_t^{k,m+I(t<j)}\right)\right\}, \lambda w_j\right]}{\sigma\|X_j\|_2^2}.$$

(2.1.3) Set $\boldsymbol{\beta}^{k+1} \leftarrow \boldsymbol{\beta}^{k,m+1}$.

(2.2) Update $\mathbf{z}^{k+1} \leftarrow \left(\text{Prox}_{\rho_\tau}\left[y_i - \mathbf{x}_i^{\mathrm{T}}\boldsymbol{\beta}^{k+1} + \sigma^{-1}\theta_i^k, n\sigma\right]\right)_{1\le i\le n}$.

(2.3) Update $\boldsymbol{\theta}^{k+1} \leftarrow \boldsymbol{\theta}^k - \sigma(\mathbf{X}\boldsymbol{\beta}^{k+1} + \mathbf{z}^{k+1} - \mathbf{y})$.

Gu, Fan and Kong et al. also checked the convergence properties of two algorithms. Convergence of the *scdADMM* algorithm can be directly obtained from Boyd et al. (2011), while for *pADMM*, they proved a theorem:

Theorem 1. For given $\lambda > 0, \sigma > 0, 0 < \tau < 1, 0 < \gamma < (\sqrt{5}+1)/2$ and a component-wisely nonnegative weight vector $\mathbf{w}$, let $\{(\boldsymbol{\beta}^k, \mathbf{z}^k, \boldsymbol{\theta}^k)\}$ be generated by the pADMM algorithm as described in Algorithm 1.

Then, the sequence $\{(\boldsymbol{\beta}^k, \mathbf{z}^k),\ k = 0, 1, 2, \ldots\}$ converges to an optimal solution $(\boldsymbol{\beta}^*, \mathbf{z}^*)$ to (4) and $\{\boldsymbol{\theta}^k,\ k = 0, 1, 2, \ldots\}$ converges to an optimal solution $\boldsymbol{\theta}^*$ to the dual problem of (4). Equivalently, $\{\boldsymbol{\beta}^k,\ k = 0, 1, 2, \ldots\}$ converges to a global minimizer of problem (1). Moreover, when $\gamma = 1$, the sequence of norms $\{\|\boldsymbol{\beta}^k - \boldsymbol{\beta}^*\|_{\mathbf{S}}^2 + \sigma\|\mathbf{z}^k - \mathbf{z}^*\|_2^2 + \sigma^{-1}\|\boldsymbol{\theta}^k - \boldsymbol{\theta}^*\|_2^2,\ k \ge 0\}$ is nonincreasing and satisfies $\|\boldsymbol{\beta}^k - \boldsymbol{\beta}^*\|_{\mathbf{S}}^2 + \sigma\|\mathbf{z}^k - \mathbf{z}^*\|_2^2 + \sigma^{-1}\|\boldsymbol{\theta}^k - \boldsymbol{\theta}^*\|_2^2 = \mathcal{O}(1/k)$ as $k \to \infty$.

Note that the convergence of the *pADMM* is guaranteed regardless of the value σ takes. According to the theorem, when $\gamma = 1$, the worst-case convergence rate of the algorithm is at least of order $1/k$ in terms of the iterate norms defined in the theorem, where k is the iteration number.

Tuning Parameter Selection

Tuning parameters play a crucial role in the optimization problem for the penalized estimators to achieve consistent selection and optimal estimation. Typically, prediction error is used as the criterion to choose regularization parameter. For traditional penalized least square method, similar criterion has been raised.

First one is extension of C_p:

$$C_P(\lambda) = \|\mathbf{y} - \widehat{\boldsymbol{\mu}}(\lambda)\|^2 + \gamma\sigma^2 df \tag{35}$$

Second one is information criterion:

$$IC(\lambda) = \log\left(\|\mathbf{y} - \widehat{\boldsymbol{\mu}}(\lambda)\|^2/n\right) + \gamma df \tag{36}$$

Third one is generalized cross-validation criterion:

$$GCV(\lambda) = \frac{\|\mathbf{y} - \widehat{\mu}(\lambda)\|^2}{(1 - df/n)^2} \tag{37}$$

The above three criterion are all based on the prediction error. However, for quantile regression the loss function is changed to $\rho_\tau(y - f(x))$. The following are three changed loss function for quantile regression.

Robust cross-validation (RCV):

$$RCV(\lambda) = \frac{1}{n}\sum_{i=1}^{n} \rho_\tau\left(y_i - \hat{f}_\tau^{[-i]}\left(\mathbf{x}_i\right)\right) \tag{38}$$

Schwarz information criterion (SIC):

$$\mathrm{SIC}(\lambda) = \ln\left(\frac{1}{n}\sum_{i=1}^{n} \rho_\tau\left(y_i - f\left(x_i\right)\right)\right) + \frac{\ln n}{2n} df \tag{39}$$

Generalized approximate cross-validation criterion (GACV):

$$\mathrm{GACV}(\lambda) = \frac{\sum_{i=1}^{n} \rho_\tau\left(y_i - f\left(x_i\right)\right)}{n - df} \tag{40}$$

Nychka et al. (1995) and Yuan (2006) proposed to use

$$\sum_{i=1}^{n} \frac{\partial \hat{f}\left(\boldsymbol{x}_i\right)}{\partial y_i} \tag{41}$$

to estimate df, where $\hat{f}(x)$ is a fitted model.

It turns out that in the case of L1 − norm QR, for every fixed s and almost all $y = (y_1, \ldots, y_n)^\top$, $\sum_{i=1}^{n} \partial \hat{f}\left(\mathbf{x}_i\right) / \partial y_i$ has an extremely simple formula:

$$\sum_{i=1}^{n} \frac{\partial \hat{f}\left(\boldsymbol{x}_i\right)}{\partial y_i} = |\mathcal{E}| \tag{42}$$

Composite Quantile Regression

The quantile regression considers only one quantile at a time and may not fully grasp the distributional information to always produce efficient estimation. To its extreme, when the error density at a specified quantile approaches zero, the asymptotic variance of the corresponding QR estimator explodes to infinity, which results in an estimator having arbitrarily small efficiency.

Issues with oracle

Although Fan and Li (2001) treated the (penalized) least squares as a special case in a general (penalized) likelihood based framework when the noise follows a normal distribution, we should note that the LS-oracle model selection theory does not need the normal error assumption, as long as the error distribution has a finite variance. However, the finite variance assumption is crucial for the oracle model selection theory based on the least squares. The reason is simple. If the error variance is infinite, $\hat{\beta}^{LS}(oracle)$ is no longer a root-n consistent estimator and can not serve as an ideal method for variable selection and coefficient estimation.

We wish to find an alternative oracle that can overcome the breakdown issue of the LS oracle. There are several important considerations when designing a new oracle estimator.

Let $\hat{\beta}^{new}(oracle)$ be a new oracle estimator.

Firstly, the new oracle estimator should be root-n consistent and enjoy asymptotic normality even when the LS-oracle fails to do so.

Secondly, we are interested in the relative efficiency of the new oracle estimator $\hat{\beta}^{new}(oracle)$ with respect to $\hat{\beta}^{LS}(oracle)$ when $\sigma^2 < \infty$. Since $\hat{\beta}^{LS}(oracle)$ is of full efficiency when the error follows a normal distribution, it is impossible to have an oracle that is universally more efficient than the LS-oracle. However, it would be very nice to have the relative efficiency of $\hat{\beta}^{new}(oracle)$ with respect to $\hat{\beta}^{LS}(oracle)$ be bounded from below. This will prevent severe loss of statistical efficiency even in the worst scenario.

Furthermore, we would like to see that $\hat{\beta}^{new}(oracle)$ can be significantly more efficient than $\hat{\beta}^{LS}(oracle)$ for commonly used nonnormal error distributions.

Composite quantile regression

Denote $0 < \tau_1 < \tau_2 < \cdots < \tau_K < 1$, We consider estimating β^* as follows :

$$\left(\hat{b}_1, \ldots, \hat{b}_K, \widehat{\boldsymbol{\beta}}^{\mathrm{CQR}}\right) = \underset{b_1,\ldots,b_k,\boldsymbol{\beta}}{\arg\min} \sum_{k=1}^{K} \left\{ \sum_{i=1}^{n} \rho_{\tau_k} \left(y_i - b_k - \mathbf{x}_i^T \boldsymbol{\beta}\right) \right\} \tag{43}$$

We now establish the asymptotic normality of $\hat{\beta}^{CQR}$. The following two regularity conditions are assumed throughout the rest of our discussions:

1. There is a $p \times p$ positive definite matrix C such that

$$\lim_{n\to\infty} \frac{1}{n} \mathbf{X}^T \mathbf{X} = \mathbf{C} \tag{44}$$

   where C is a $p \times p$ positive definite matrix.

2. ϵ has cumulative distribution function F(·) and density function f (·). For each p-vector u,

$$\begin{aligned} \lim_{n\to\infty} \frac{1}{n} \sum_{i=1}^{n} \int_0^{u_0 + \mathbf{x}_i^T \mathbf{u}} \sqrt{n}[F(a + t/\sqrt{n}) - F(a)]dt \\ = \frac{1}{2} f(a) \left(u_0, \mathbf{u}^T\right) \begin{bmatrix} 1 & 0 \\ 0 & \mathbf{C} \end{bmatrix} \left(u_0, \mathbf{u}^T\right)^T \end{aligned} \tag{45}$$

Conditions 1–2 are basically the same conditions for establishing the asymptotic normality of a single quantile regression. Under these conditions, we have the following result for the CQR estimates.

Theorem (The limiting distribution) Under the regularity conditions 1 and 2, the limiting distribution of $\sqrt{n}\left(\widehat{\boldsymbol{\beta}}^{\mathrm{CQR}} - \boldsymbol{\beta}^*\right)$ is $N\left(0, \boldsymbol{\Sigma}_{\mathrm{CQR}}\right)$ where

$$\boldsymbol{\Sigma}_{\mathrm{CQR}} = \mathbf{C}^{-1} \frac{\sum_{k,k'=1}^{K} \min\left(\tau_k, \tau_{k'}\right)\left(1 - \max\left(\tau_k, \tau_{k'}\right)\right)}{\left(\sum_{k=1}^{K} f\left(b_{\tau_k}^*\right)\right)^2} \tag{46}$$

$$\left(\sum_{k=1} f\left(b_{\tau_k}^*\right)\right)$$

This theorem says that $\hat{\beta}^{CQR}$ is root-n consistent and enjoy asymptotic normality.

Then we are interested in the asymptotic relative efficiency. The asymptotic relative efficiency (ARE) of the CQR with respect to the least squares is investigated. The same results can be applied to compute the relative efficiency of the CQR-oracle with respect to the LS-oracle.

Theorem The universal lower bound.

$$\lim_{K\to\infty} \frac{\sum_{k,k'=1}^{K} \min\left(\tau_k, \tau_{k'}\right)\left(1-\max\left(\tau_k, \tau_{k'}\right)\right)}{\left(\sum_{k=1}^{K} f\left(b_{\tau_k}^*\right)\right)^2} = \frac{1}{12(E_\varepsilon[f(\varepsilon)])^2} \tag{47}$$

and

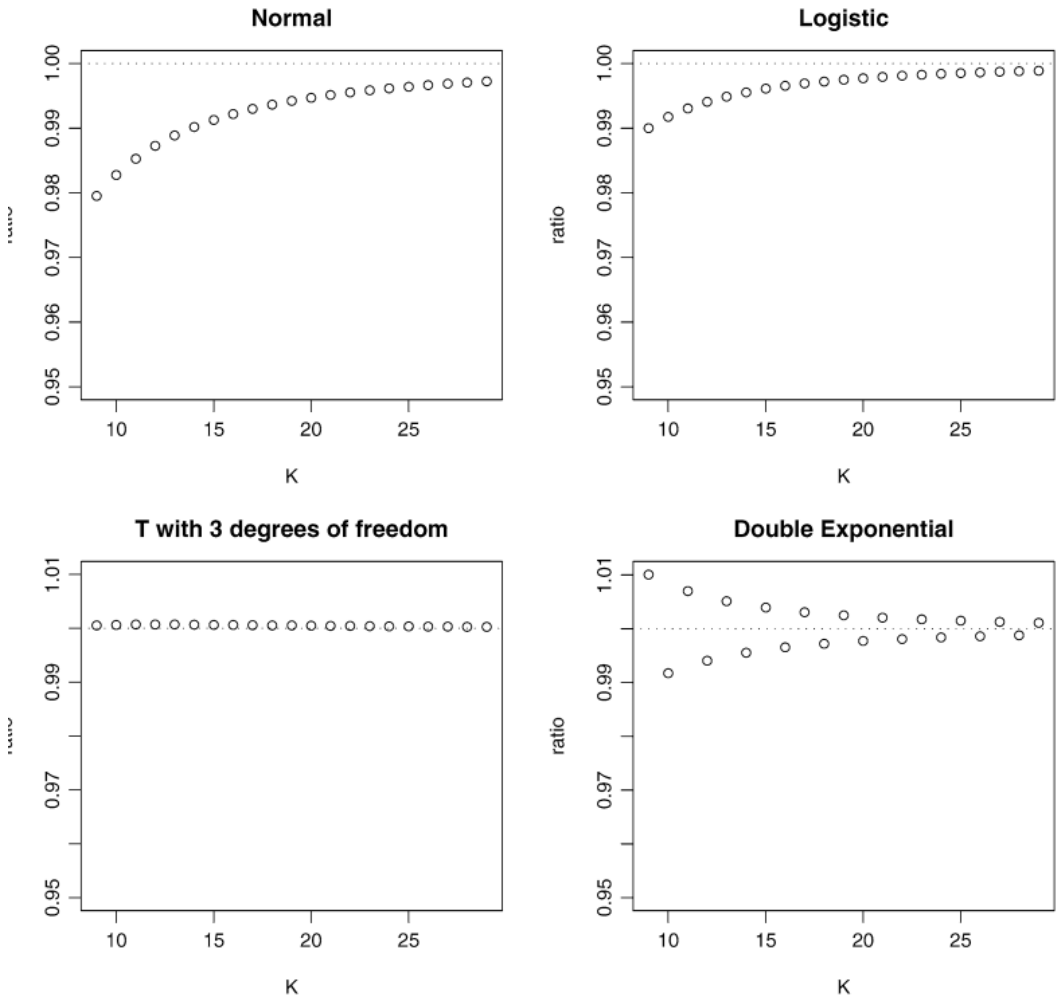
$$\delta(f) \equiv \lim_{K\to\infty} \mathrm{ARE}(K, f) = 12\sigma^2(E_\varepsilon[f(\varepsilon)])^2 \tag{48}$$

Denote by $\mathcal{F}$ the collection of all density functions that satisfy condition (2) and have a finite variance. We have

$$\inf_{f\in\mathcal{F}} \delta(f) > \frac{6}{e\pi} = 0.7026 \tag{49}$$

Although $\delta(f)$ explicitly depends on σ^2, it is actually scale-invariant. We should also point out that the lower bound 70.26% given above is conservative. For commonly used error distributions in practice, δ is often much larger than the lower bound. Having the lower bound is a very useful property. It prevents severe loss in efficiency when using the CQR estimator instead of the LS estimator. Even in the worst possible scenario, the potential loss in efficiency is less than 30%.

Empirically, we have found that for a reasonably large K, RE(K,f) is already very close to its limit.



As can be seen from Figure , the ratio is very close to 1 for K ≥ 9 in all the four different distributions considered there. In practice, it seems that K = 19 is a good choice, which amounts to using the 5%, 10%, 15%,..., 95% quantiles.

Simulation

We use simulation to compare the LS-oracle and the CQR-oracle and examine the performance of the ACQR with finite samples. Our simulated data consist of a training set and an independent validation set. Models were fitted on training data only, and the validation data were used to select the tuning parameters. We simulated 100 data consisting of 100 training observations and 100 validation observations from the model

$$y = x^T \beta + \epsilon \tag{50}$$

where $\beta = (3, 1.5, 0, 0, 2, 0, 0, 0)$ and the predictors $(x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8)$ follow a multivariate normal distribution $N(0, \mathbf{\Sigma}_x)$ with $(\mathbf{\Sigma}_x)_{i,j} = 0.5^{|i-j|}$ for $1 \leq i, j \leq 8$. This regression model was considered in Tibshirani (1996) and Fan and Li (2001). Here we considered five different error distributions.

Example 1

$$\varepsilon \sim N(0, 3) \tag{51}$$

Example 2

$\varepsilon = \sigma * \varepsilon^*$ where ε^* follows the mixture of normal distribution as

$$\varepsilon \sim (1-r)N(0,1) + rN\left(0, r^6\right) \tag{52}$$

with $r = 0.5, \sigma = \sqrt{6}$

Example 3

$\varepsilon = \sigma * \varepsilon^*$ where ε^* follows the mixture of normal double gamma as

$$\varepsilon \sim e^{-\alpha}\frac{1}{2}e^{-|\varepsilon|} + \left(1-e^{-\alpha}\right)\frac{1}{\Gamma(\alpha+1)}\varepsilon^{\alpha}e^{-|\varepsilon|} \tag{53}$$

with $\alpha = 14, \sigma = \frac{1}{9}$

Example 4

The error distribution is T-distribution with 3 degrees of freedom

Example 5

The error distribution is Cauchy

We used the quantiles $\tau_k = k/20$ for k = 1, 2,..., 19 in the CQR-oracle and the ACQR. The model error is computed by

$$ME = E[(\hat{\beta} - \beta)^T \mathbf{\Sigma}_x (\hat{\beta} - \beta)]. \tag{54}$$

Following is the simulation result:

Table: Simulation Results

		Example 1	Example 2	Example 3	Example 4	Example 5
Model error	LS-oracle	0.079	0.104	0.091	0.082	2788
	CQR-oracle	0.085	0.033	0.043	0.060	0.143

Table shows the average model errors and variable selection results over 100 replications. In the asymptotic sense, the LS-oracle is the best in Example 1, while the CQR-oracle works better in Examples 2–4. The numerical experiments agree with the theory. Example 5 is different from Examples 1–4, because the error distribution has infinite variance in Example 5. The LS-oracle is not the optimal estimator in this case. The simulation

confirmed the theory. The model error of the LS-oracle is more than 2500. The CQR-oracle still works very well in Example 5.

Penalized composite quantile regression

Lasso penalized composite quantile regression

For $\lambda > 0$, we define the lasso penalized CQR estimator as

$$(\widehat{\boldsymbol{\alpha}}_\lambda, \widehat{\boldsymbol{\beta}}_\lambda) := \underset{\boldsymbol{\alpha}, \boldsymbol{\beta}}{\arg\min}\, Q_n(\boldsymbol{\alpha}, \boldsymbol{\beta}) + \lambda \sum_{j=1}^{p} |\beta_j|,$$

We can typically choose the tuning parameter $\lambda = C\sqrt{log(p/n)}$ for the CQR lasso estimator, where $C > \sqrt{2M_0}$ is some constant, where

$$M_0 = \max_{0 \le j \le p} \|X_j\|_2^2/n \tag{55}$$

For example, one can choose $C = 2\sqrt{M_0}$.

Therefore, in principle, the parameter λ in the lasso penalized CQR is tuning free. This is in similar spirit to the square-root lasso. With such a choice of λ, we can see that $p_1(\lambda) = O(1)$ as $n, p \to \infty$, which leads to

$$||\hat{\beta}_\lambda - \beta^*||_2 = O_p(\frac{1}{\sqrt{\kappa_o \kappa_s}} \sqrt{\frac{s \log(p)}{n}}) \tag{56}$$

provided $q^{-1}\sqrt{s \log(p)/(n\kappa_0)} = O(1)$ and $\kappa_0 (s \log)^{-1} = O(1)$, by taking $m = s$. When κ_o and κ_s are both positive constants the CQR lasso estimator achieves the near-optimal rate $\sqrt{s \log(p/n)}$, which implies that it is a consistent estimator even when p is of exponential order of n.

Folded concave penalized composite quantile regression

Folded concave penalized regression has been widely adopted in the statistical analysis of high-dimensional data due to its strong oracle optimality. In order to establish the oracle property of the folded concave penalized CQR estimator, let us first define the CQR oracle estimator,

$$(\widehat{\boldsymbol{\alpha}}^{\mathrm{o}}, \widehat{\boldsymbol{\beta}}^{\mathrm{o}}) := \underset{\boldsymbol{\alpha}, \boldsymbol{\beta}:\, \boldsymbol{\beta}_{\mathcal{A}^{\mathrm{c}}} = \mathbf{0}}{\arg\min} \sum_{i=1}^{n} \sum_{k=1}^{K} \rho_{\tau_k}(y_i - \alpha_k - \mathbf{x}_i^{\mathrm{T}} \boldsymbol{\beta}).$$

The oracle estimator (α_0, β_0) is the ideal estimator one could possibly get using the CQR. It is not feasible in practice since A is unknown, but it serves as a benchmark estimator to which one can compare a penalized CQR estimator. In the following lemma, we show the rate of convergence of the CQR oracle estimator under the growing-dimension regime, i.e., the true dimensionality s is allowed to grow with n.

For the analysis of the minimizer, we consider the local linear approximation (LLA algorithm, where the initial estimator is chosen to be the CQR lasso estimator:

1. Initial α and β with α_0 and β_0, respectively, and compute weights

$$\hat{w}_j^{(0)} = p'_\lambda \left(\left| \hat{\beta}_j^{(0)} \right| \right), j = 1, \ldots, p. \tag{57}$$

2. For m = 1, 2,..., repeat the LLA iterations in the following two steps.

2.a) Solve the following convex optimization problem for $\widehat{\boldsymbol{\alpha}}^{(m)}$ and $\widehat{\boldsymbol{\beta}}^{(m)}$

$$\min_{\boldsymbol{\alpha}, \boldsymbol{\beta}} Q_n(\boldsymbol{\alpha}, \boldsymbol{\beta}) + \sum_{j=1}^{p} \hat{w}_j^{(m-1)} |\beta_j|. \tag{58}$$

2.b) Calculate the weights

$$\hat{w}_j^{(m)} = p'_\lambda \left(\left| \hat{\beta}_j^{(m)} \right| \right), j = 1, \ldots, p. \tag{59}$$

For the folded concave penalized CQR, there exists a tuning sequence λ_n (we write λn here to signify its dependence on n) such that the LLA algorithm yields the CQR oracle estimator in two iterations with probability approaching one. However, as we pointed out already, there is no direct way to use such λ_n, since it relies on unknown quantities. We thus pursue a data-driven approach to the selection of λ. Consider the following high-dimensional Bayesian information criterion (BIC):

$$\mathrm{BIC}^{\mathrm{H}}(\lambda) = \frac{1}{nK} \sum_{i=1}^{n} \sum_{k=1}^{K} \rho_{\tau_k} \left(y_i - \hat{\alpha}_k^\lambda - \mathbf{x}_i^{\mathrm{T}} \hat{\boldsymbol{\beta}}^\lambda \right) + \left| \hat{A}_\lambda \right| \frac{C_n \log(p)}{n} \tag{60}$$

We compare the values of $BIC^H(\lambda)$ for

$$\lambda \in \Xi_n = \left\{ \lambda : \left| \hat{A}_\lambda \right| \leq J_n \right\}, \tag{61}$$

where $J_n > s$ represents a rough estimate of the upper bound of the model sparsity and is allowed to (slowly) diverge as $n \to \infty$.Typically, Jn is much smaller than p, so that one can avoid searching over a notoriously large model space. The tuning parameter selected via BIC is given by

$$\hat{\lambda}_n = \arg\min_{\lambda \in \Xi_n} \mathrm{BIC}^{\mathrm{H}}(\lambda) \tag{62}$$

$$\lambda_n = \arg\min_{\lambda \in \Xi_n} BIC^{HL}(\lambda) \qquad (62)$$

Remark: The selection consistency of $BIC^{HL}(\lambda)$ requires additionally that $E(|\varepsilon|) < \infty$. Indeed, in the numerical comparison there, we see that $BIC^{HL}(\lambda)$ does not perform well under the Cauchy error.

Optimization

However, linear programming does not scale well when p is large. Hence an efficient alternating direction method of multipliers (ADMM) algorithm is proposed. The algorithm is based on a reformulation that turns the original problem into one that can harness the power of ADMM. Following is the algorithm for ADMM:

Algorithm 1: The ADMM algorithm for solving the weighted lasso penalized composite quantile regression

1) Initialize the algorithm with $(\boldsymbol{\varphi}^0, \mathbf{Z}^0, \boldsymbol{\gamma}^0, \mathbf{U}^0, \mathbf{v}^0)$, where $\boldsymbol{\varphi}^0 = ((\boldsymbol{\alpha}^0)^{\mathrm{T}}, (\boldsymbol{\beta}^0)^{\mathrm{T}})^{\mathrm{T}}$.
2) For $r = 0, 1, 2, \ldots,$ repeat steps (2.1) – (2.3) until convergence.

(2.1) Update

$$\boldsymbol{\varphi}^{r+1} = ((\boldsymbol{\alpha}^{r+1})^{\mathrm{T}}, (\boldsymbol{\beta}^{r+1})^{\mathrm{T}})^{\mathrm{T}} \leftarrow \frac{1}{\sigma}(\mathbb{X}_1^{\mathrm{T}}\mathbb{X}_1 + \mathbb{X}_2^{\mathrm{T}}\mathbb{X}_2)^{-1} \cdot \{\mathbb{X}_1^{\mathrm{T}}(\sigma\mathbf{Y} - \sigma\mathrm{vec}(\mathbf{Z}^r) - \mathrm{vec}(\mathbf{U}^r)) + \mathbb{X}_2^{\mathrm{T}}(\sigma\boldsymbol{\gamma}^r + \mathbf{v}^r)\}.$$

(2.2) Update

$$z_{ik}^{r+1} \leftarrow \mathrm{Prox}_{\rho\tau_k}\left(y_i - \alpha_k^{r+1} - \mathbf{x}_i^{\mathrm{T}}\boldsymbol{\beta}^{r+1} - \frac{u_{ik}^r}{\sigma}, nK\sigma\right), 1 \le i \le n, 1 \le k \le K,$$

and

$$\gamma_j^{r+1} \leftarrow \mathrm{Shrink}\left(\beta_j^{r+1} - \frac{v_j^r}{\sigma}, \frac{\lambda_1 d_j}{\sigma}\right), 1 \le j \le p.$$

(2.3) Update

$$\mathrm{vec}(\mathbf{U}^{r+1}) \leftarrow \mathrm{vec}(\mathbf{U}^r) + \sigma\{\mathrm{vec}(\mathbf{Z}^{r+1}) + \mathbb{X}_1\boldsymbol{\varphi}^{r+1} - \mathbf{Y}\}$$

and

$$\mathbf{v}^{r+1} \leftarrow \mathbf{v}^r + \sigma\{\boldsymbol{\gamma}^{r+1} - \mathbb{X}_2\boldsymbol{\varphi}^{r+1}\}.$$

Main References

Note: we do not list all references here, for interested readers, we suggest turning to these main materials to get all references. Sorry for the inconvenience due to limited time.

1. Belloni A , Chernozhukov V . L1-Penalized Quantile Regression in High-Dimensional Sparse Models[J]. The Annals of Statistics, 2011(1).
2. Gu, Y., Fan, J., Kong, L., Ma, S., and Zou, H. (2018), "ADMM for High-Himensional Sparse Penalized Quantile Regression," Technometrics, 60, 319–331.
3. Fan, J., Li, R., Zhang, C. H., & Zou, H. (2020). *Statistical foundations of data science*. Chapman and Hall/CRC.
4. Fan J, Xue L, Zou H. (2014) STRONG ORACLE OPTIMALITY OF FOLDED CONCAVE PENALIZED ESTIMATION. Ann Stat.

5. Koenker, R. (2005). Quantile Regression, Econometric Society Monograph Series, Cambridge University Press.
6. Li, Y. , Zhu, J. .(2012) L1-norm quantile regression.
7. Y. Gu and H. Zou,(2020) ”Sparse Composite Quantile Regression in Ultrahigh Dimensions With Tuning Parameter Calibration,” in IEEE Transactions on Information Theory.
8. Zou, H., Yuan, M. (2008). Composite quantile regression and the oracle model selection theory. Annals of Statistics