Introduction Quantile Regression

Thomas Wiemann
University of Chicago

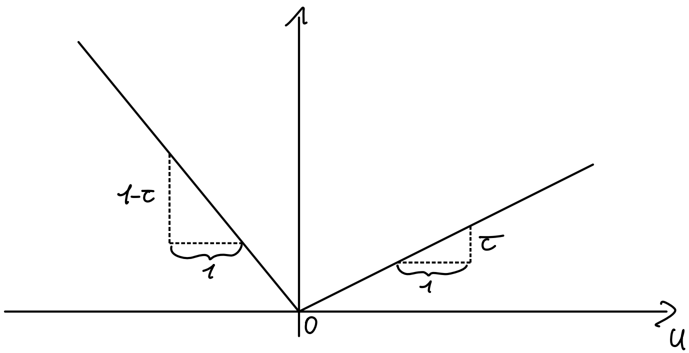
TA Discussion # 3
Econ 31740

February 1, 2022

Outline*

We are familiar with OLS's quadratic loss function $L(u) = u^2$. In the population, OLS solves the problem of $\min_\beta E\left[L(Y - X^\top \beta)\right] = \min_\beta E\left[(Y - X^\top \beta)^2\right]$.

Quantile regression (QR) uses the check functions as the loss function. The check function is given by $\rho_\tau(u) = u\left(\tau - \mathbb{1}\{u < 0\}\right)$. E.g.,



QR solves the population problem of $\min_{\beta_\tau} E\left[\rho_\tau(Y - X^\top \beta_\tau)\right]$ for $\tau \in [0, 1]$. Today's discussion gives an introduction to QR with a focus on its interpretation and computation.

*This discussion is based in large part on the TA material of Joshua Shea.

Interpretation

Let $Y \sim F_y$ and define $q^* := \arg\min_q E\left[\rho_\tau(Y-q)\right]$. The name *quantile* regression is then justified because q^* will satisfy $\tau = F_y(q^*)$.

To show this, it is convenient to work with positive component-notation – i.e., decompose $u = u_+ - u_-$ with $u_+, u_- \geq 0$, and therefore we have $\rho_\tau(u) = \tau u_+ + (1-\tau)u_-$. Then QR solves

$$\min_q \tau \int_q^\infty (Y-q)dF_y(Y) + (1-\tau)\int_{-\infty}^q (Y-q)dF_y(Y). \tag{1}$$

The first order condition (derived using the Leibniz rule) gives

$$\begin{aligned} 0 =& -\tau \int_{q^*}^\infty 1 dF_y(Y) + (1-\tau)\int_{-\infty}^{q^*} 1 dF_y(Y) \\ =& -\tau(1-F_y(q^*)) + (1-\tau)F_y(q^*) \\ \Rightarrow \quad \tau =& F_y(q^*). \end{aligned} \tag{2}$$

Intuition: the check function puts different weight on over-estimating and under-estimating. Using τ, you can defined how much mass should be under-estimated and how much mass should be over-estimated.

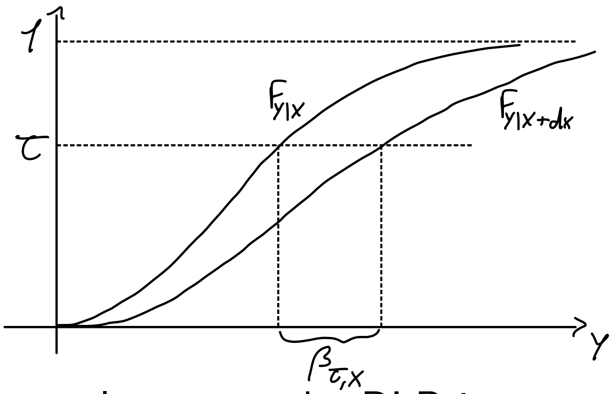
Interpretation (Contd.)

The notion from this simple example can be extended to linear regression, where QR solves the population problem given by

$$\beta_\tau^* = \arg\min_{\beta_\tau} E\left[\rho_\tau(Y - X^\top \beta_\tau)\right], \tag{3}$$

for $\tau \in [0, 1]$.

How do you interpret β_τ^*? (See problem set 3.)



(For an interpretation analogous to the BLP interpretation of OLS, see Angrist et al., 2006.)

Interpretation in Canonical Treatment Effects Model

Quantile regression also has a useful interpretation within the canonical treatment effects model where $Y = Y(0)(1 - D) = Y(1)D$ with $D \in \{0, 1\}$ and $Y(d)$ denotes the potential outcome with $D = d$ fixed.

The population objective function associated with a quantile regression of Y on D is given by

$$
\begin{aligned}
E\big[\rho_\tau(Y - \beta_0 - \beta_1 D)\big] &= E\Big[\rho_\tau(Y - \beta_0 - \beta_1)D\Big] + E\Big[\rho_\tau(Y - \beta_0)(1 - D)\Big] \\
&= E\Big[\rho_\tau(Y(1) - \beta_0 - \beta_1)\Big| D = 1\Big] P(D = 1) \\
&\quad + E\Big[\rho_\tau(Y(0) - \beta_0)\Big| D = 0\Big] P(D = 0).
\end{aligned}
$$

Under random assignment of D – i.e., $D \perp\!\!\!\perp Y(0), Y(1)$ – we have

$$
\begin{aligned}
E\big[\rho_\tau(Y - \beta_0 - \beta_1 D)\big] &= E\Big[\rho_\tau(Y(1) - \beta_0 - \beta_1)\Big] P(D = 1) \\
&\quad + E\Big[\rho_\tau(Y(0) - \beta_0)\Big] P(D = 0).
\end{aligned}
$$

Interpretation in Canonical Treatment Effects Model (Contd.)

Note further that

$$
\begin{aligned}
&\arg\min_{\beta_0,\beta_1} E\big[\rho_\tau(Y - \beta_0 - \beta_1 D)\big] \\
&= \arg\min_{\beta_0,\beta_1} \Big[E\Big[\rho_\tau(Y(1) - \beta_0 - \beta_1)\Big] P(D=1) + E\Big[\rho_\tau(Y(0) - \beta_0)\Big] P(D=0)\Big] \\
&= [\beta_0^{(\tau)}, \quad \beta_1^{(\tau)}],
\end{aligned}
$$

where

$$
\begin{aligned}
\beta_0^{(\tau)} &= \arg\min_{\beta_0} E\Big[\rho_\tau(Y(0) - \beta_0)\Big] \\
\beta_1^{(\tau)} &= \arg\min_{\beta_1} E\Big[\rho_\tau(Y(1) - \beta_0^{(\tau)} - \beta_1)\Big].
\end{aligned}
\tag{4}
$$

Consequently, under random assignment of a binary treatment, we may interpret $\beta_1^{(\tau)}$ as the difference in the τ^{th} quantile of the potential outcome distributions.

Interpretation in Canonical Treatment Effects Model (Contd.)

For example, $\beta_1^{(0.5)} = 1$ implies that the median of $Y(1)$ is 1 higher than the median of $Y(0)$.

Note that, unless we impose stronger assumptions such as rank invariance, we may *not* interpret the outcome as a shift to any individual (i.e., the individual at the median of $Y(0)$ may not be at the median of $Y(1)$ under treatment).

Computation

How is QR sample problem $\arg\min_{\beta_\tau} E_n\left[\rho_\tau(Y - X^\top \beta_\tau)\right]$ computed in practice? Note that ρ_τ is a non-differentiable loss function.

Koencker uses the subgradient function. Alternatively, QR can be formulated as a linear programming (LP) problem.

Linear programming is a type of optimization problem where the objective function and all constraints enter linearly. Linear programming problems, even of large dimensions, can be readily solved in practice. Future weeks will delve into, for example, the Simplex algorithm.

Today we'll focus on the basics necessary for putting turning QR into an LP problem in *standard form*.

Introduction to Linear Programming

A LP in standard form is given by

$$\begin{aligned} \min_{x} \quad & c'x \\ \text{s.t.} \quad & Ax = b, \\ & x \geq 0, \end{aligned} \tag{5}$$

where x are the non-negative decision variables, (A, b) define the constraints, and c is the objective vector. Most real-world problems, when translated into a linear programming problem, don't immediately take this form. For example, we might instead have

$$\begin{aligned} \min_{x} \quad & c'x \\ \text{s.t.} \quad & Ax = b, \\ & Dx \geq e, \\ & x \in \mathbb{R}^n. \end{aligned} \tag{6}$$

Fortunately, any linear program can be converted into standard form. Theory and optimization procedures thus focus on discussion of standard from LPs (e.g., Bertsimas and Tsitsiklis, 1997).

Introduction to Linear Programming (Contd.)

Consider again the LP in non-standard form:

$$\begin{aligned} \min_x \quad & c'x \\ \text{s.t.} \quad & Ax = b, \\ & Dx \geq e, \\ & x \in \mathbb{R}^n. \end{aligned} \tag{7}$$

Define $x_+, x_- \geq 0 : x = x_+ - x_-$ and introduce *slack variables* s. Take

$$Ax = b \Rightarrow Ax_+ - Ax_- = b \Rightarrow \begin{bmatrix} A & -A \end{bmatrix} \begin{bmatrix} x_+ \\ x_- \end{bmatrix} = b, \tag{8}$$

and similarly

$$\begin{aligned} Dx \geq e \Rightarrow Dx - s = e, \ \text{where } s \geq 0 \\ \Rightarrow Dx_+ - Dx_- - s = e \\ \Rightarrow \begin{bmatrix} D & -D & -\mathbf{I} \end{bmatrix} \begin{bmatrix} x_+ \\ x_- \\ s \end{bmatrix} = e. \end{aligned} \tag{9}$$

Introduction to Linear Programming (Contd.)

Then redefine parameters and variables of the linear program as follows:

$$\tilde{A} := \begin{bmatrix} A & -A & \mathbf{0} \\ D & -D & -\mathbf{I} \end{bmatrix}, \quad \tilde{b} := \begin{bmatrix} b \\ e \end{bmatrix}, \quad \tilde{x} := \begin{bmatrix} x_+ \\ x_- \\ s \end{bmatrix}, \quad \text{and} \quad \tilde{c} := \begin{bmatrix} c \\ -c \\ 0 \end{bmatrix},$$

where $\tilde{x} \geq 0$ by construction.

Then the linear program in (7) is then equivalent to

$$\begin{aligned} \min_{\tilde{x}} \quad & \tilde{c}'\tilde{x} \\ \text{s.t.} \quad & \tilde{A}\tilde{x} = \tilde{b}, \\ & \tilde{x} \geq 0. \end{aligned} \tag{10}$$

Quantile Regression as a Linear Program

We'll now apply this technique to quantile regression. Let $\{(y_i, x_i)\}_{i=1}^n$ denote the observed sample where $y_i \in \mathbb{R}$ and $x_i \in \mathbb{R}^k$. Recall that $\rho_\tau(z) = \tau z_+ + (1-\tau)z_-$ and define $u_i = (y_i - x_i^\top \beta_\tau)_+$ and $v_i = (y_i - x_i^\top \beta_\tau)_-$.

Then quantile regression $\arg\min_{\beta_\tau} \sum_{i=1}^n \rho_\tau(y_i - x_i^\top \beta_\tau)$ is equivalent to

$$
\begin{aligned}
&\arg\min_{\beta_\tau} \tau \mathbf{1}_n^\top u + (1-\tau)\mathbf{1}_n^\top v \\
&\quad \text{s.t.} \quad u_i - v_i = y_i - x_i^\top \beta_\tau, \forall i, \\
&\qquad\qquad \beta_\tau \in \mathbb{R}^k, u_i, v_i \geq 0, \forall i.
\end{aligned} \tag{11}
$$

Quantile regression can thus be written as a linear program, but (11) is not yet in standard form. We therefore decompose $\beta_\tau = \beta_{\tau+} - \beta_{\tau-}$ with $\beta_{\tau+}, \beta_{\tau-} \geq 0$.

Quantile Regression as a Linear Program (Contd.)

Putting everything together, we have

$$\tilde{A} := \begin{bmatrix} \mathbf{I} & -\mathbf{I} & X & -X \end{bmatrix}, \; \tilde{b} := Y, \; \tilde{x} := \begin{bmatrix} u \\ v \\ \beta_{\tau+} \\ \beta_{\tau-} \end{bmatrix}, \text{ and } \tilde{c} := \begin{bmatrix} \tau \mathbf{1}_n \\ (1-\tau)\mathbf{1}_n \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix}$$

so that QR is equivalent to

$$\begin{aligned} \min_{\tilde{x}} \quad & \tilde{c}'\tilde{x} \\ \text{s.t.} \quad & \tilde{A}\tilde{x} = \tilde{b}, \\ & \tilde{x} \geq 0. \end{aligned} \tag{12}$$

Implementation in Julia

The code snippets on the following two slides implement quantile regression for the simple case in Julia using the `JuMP` interface and the `GLPK` solver. (For mode advanced solvers, check Gurobi – free academic licenses are available.)

```
# Load additional dependencies
using Statistics # for pre-implemented quantiles
using JuMP, GLPK # for linear optimization

# Generate some example data from a standard uniform
nobs = 1001
Y = rand(nobs)

# initialize model
m = Model(GLPK.Optimizer)

# initialize variables (all weakly positive)
@variable(m, u[1:nobs] >= 0)
@variable(m, v[1:nobs] >= 0)
@variable(m, q_p >= 0)
@variable(m, q_n >= 0)

# set constraints
@constraint(m, c[i = 1:nobs], u[i] - v[i] + q_p - q_n  == Y[i])
```

Implementation in Julia (Contd.)

Last, we choose a value $\tau \in [0, 1]$ and solve the linear program.

```
# Choose tau and define the objective function
tau = 0.5
@objective(m, Min, tau*sum(u[1:nobs]) + (1 - tau)*sum(v[1:nobs]))

# solve model
MOI.set(m, MOI.Silent(), true)
optimize!(m)
```

To confirm our implementation, we compare the calculated value for q^* with the pre-implemented `quantile` function.

```
# Check implementation
value(m[:q_pos]) - value(m[:q_neg]) == quantile(Y, 0.5)
#> true
```

Success!

References

Angrist, J., Chernozhukov, V., and Fernández-Val, I. (2006). Quantile regression under misspecification, with an application to the us wage structure. *Econometrica*, 74(2):539–563.

Bertsimas, D. and Tsitsiklis, J. N. (1997). *Introduction to linear optimization*, volume 6. Athena Scientific Belmont, MA.