

Čížek, Pavel

**Working Paper**

## Quantile regression

SFB 373 Discussion Paper, No. 1999,78

**Provided in Cooperation with:**

Collaborative Research Center 373: Quantification and Simulation of Economic Processes,  
Humboldt University Berlin

*Suggested Citation:* Čížek, Pavel (1999) : Quantile regression, SFB 373 Discussion Paper, No. 1999,78, Humboldt University of Berlin, Interdisciplinary Research Project 373: Quantification and Simulation of Economic Processes, Berlin, <https://nbn-resolving.de/urn:nbn:de:kobv:11-10046663>

This Version is available at:

<https://hdl.handle.net/10419/61696>

**Standard-Nutzungsbedingungen:**

Die Dokumente auf EconStor dürfen zu eigenen wissenschaftlichen Zwecken und zum Privatgebrauch gespeichert und kopiert werden.

Sie dürfen die Dokumente nicht für öffentliche oder kommerzielle Zwecke vervielfältigen, öffentlich ausstellen, öffentlich zugänglich machen, vertreiben oder anderweitig nutzen.

Sofern die Verfasser die Dokumente unter Open-Content-Lizenzen (insbesondere CC-Lizenzen) zur Verfügung gestellt haben sollten, gelten abweichend von diesen Nutzungsbedingungen die in der dort genannten Lizenz gewährten Nutzungsrechte.

**Terms of use:**

*Documents in EconStor may be saved and copied for your personal and scholarly purposes.*

*You are not to copy documents for public or commercial purposes, to exhibit the documents publicly, to make them publicly available on the internet, or to distribute or otherwise use the documents in public.*

*If the documents have been made available under an Open Content Licence (especially Creative Commons Licences), you may exercise further usage rights as specified in the indicated licence.*

# Quantile regression

Pavel Čížek  
October 13, 1999

**Quantile regression** (QR) is a statistical technique that allows to estimate conditional quantile functions (e.g., the conditional median function) and obtain statistical inference about them in much the same way as classical regression methods based on minimizing sums of residuals facilitate estimation of conditional mean functions.

This chapter helps you to understand the main principles of quantile regression and demonstrates how to use XploRe for quantile regression analysis. As it is impossible to provide a profound introduction into this area here, we refer readers for further information to bibliography.

Before proceeding to the next section, please type at the XploRe command line

```
library("metrics")
```

to load the necessary libraries. Library `metrics` automatically loads `xploRe`, `kernel`, `glm`, and `multi` libraries.

## 1 Introduction

The purpose of the classical least squares estimation is to answer the question “How does the conditional expectation of a random variable  $Y$ ,  $E(Y|X)$ , depends on some explanatory variables  $X$ ?” usually under some assumptions about the functional form of  $E(Y|X)$ , e.g., linearity. On the other hand, quantile regression enable to pose such a question at any quantile of the conditional distribution. Let us remind that a real-valued random variable  $Y$  is fully characterized by its distribution function  $F(y) = P(Y \leq y)$ . Given  $F(y)$ , we can for any  $\tau \in (0, 1)$  define  $\tau$ th *quantile* of  $Y$  by

$$Q_Y(\tau) = \inf\{y \in \mathbb{R} | F(y) \geq \tau\}. \quad (1)$$

The *quantile function*, i.e.,  $Q_Y(\tau)$  as a function of  $\tau$ , completely describes the distribution of the random variable  $Y$ . Hence, the estimation of conditional quantile functions allows to obtain a more complete picture about the dependence of the conditional distribution of  $Y$  on  $X$ . In other words, this means that we have a possibility to investigate the influence of explanatory variables on the *shape* of the distribution.

Probably the simplest presentation of the above outlined idea is possible within the framework of the classical two-sample treatment-control model. There are two groups—one control group that is left without treatment, and the other one to which a particular treatment is applied—and the researcher is, of course, interested in the effect of the treatment on the performance of individuals. Doksum (1974) showed that the treatment effect can be uniquely defined by

$F(x) = G\{x + \Delta(x)\}$  as “horizontal distance”  $\Delta(x)$  between the original distribution,  $F$ , that describes the control group and the new one,  $G$ , that describes individuals after treatment. Denoting, for the sake of simplicity,  $F^{-1}, G^{-1}$  quantile functions corresponding to the distribution functions  $F, G$ , respectively, we get  $\Delta(x) = G^{-1}\{F(x)\} - x$ , and hence for  $\tau = F(x)$  the *quantile treatment effect* can be expressed as

$$\delta(\tau) = \Delta\{F^{-1}(\tau)\} = G^{-1}(\tau) - F^{-1}(\tau),$$

which describes the true effect of treatment at every quantile of the original distribution  $F$ .

On the estimation of the quantile treatment effect, we demonstrate now one important difference between the traditional expectation-oriented approach and quantile regression. If we neglect for now eventual difficulties related to the estimation of such models in general, the outlined treatment-control model can be easily described and estimated with the help of the treatment dummy  $D_i$  that is zero for the members of the control group and one for the rest of observations. Let us consider as an example a part of `vitaminc` data which contains observations on the effect of a single 600 mg dose of ascorbic acid versus a sugar placebo on the muscular endurance (measured by repetitive grip strength trials) of fifteen male volunteers in the first round (see Data Sets).

1. Starting with the standard linear regression model

$$\mathbf{E}(Y_i|D_i) = \alpha + \delta \cdot D_i, \quad (2)$$

the least squares method estimates  $\delta$  by

$$\hat{\delta} = \sum_{\{i, D_i=1\}} y_i - \sum_{\{i, D_i=0\}} y_i = E_{\hat{F}_n} Y - E_{\hat{G}_n} Y, \quad (3)$$

where  $\hat{F}_n, \hat{G}_n$  denotes empirical distributions of samples  $\{y_i|D_i = 0\}, \{y_i|D_i = 1\}$ , respectively. Numerical results for the given example are presented in table 1.

$\hat{\alpha}$	$\hat{\delta}$
4.514	-2.197

Table 1: The OLS estimate of model (2). [qr01.xpl](#)

2. In the quantile regression framework, the model is for a given  $\tau \in (0, 1)$  characterized by

$$Q_{Y_i}(\tau|D_i) = \alpha(\tau) + \delta(\tau) \cdot D_i \quad (4)$$

(note that the parameters are now functions of  $\tau$ ). It is possible to derive that the quantile regression estimate of  $\delta(\tau)$  is given by

$$\hat{\delta}(\tau) = \hat{G}_n^{-1}(\tau) - \hat{F}_n^{-1}(\tau), \quad (5)$$

where  $\hat{F}_n^{-1}, \hat{G}_n^{-1}$  denotes empirical quantile functions for samples  $\{y_i|D_i = 0\}, \{y_i|D_i = 1\}$ , respectively. Numerical results for several choices of  $\tau$  are presented in table 2.

$\tau$	$\hat{\alpha}(\tau)$	$\hat{\delta}(\tau)$
0.1	1.80	-0.63
0.3	2.79	-1.20
0.5	5.26	-2.78
0.7	5.82	-3.18
0.9	6.99	-3.12

Table 2: The QR estimate of model (4). [qr02.xpl](#)

Comparing given two methods, it is easy to see that while the traditional estimation of the conditional expectation provides only information about the difference of averages between groups, see (3), the quantile regression allows to identify the effect of treatment at various quantiles. Importance of this fact emerges once we examine carefully tables 1 and 2 (regression lines are also depicted in figure 1). Whereas the first one tells us just something about the average effect of treatment, which can be roughly compared with quantile regression for  $\tau = 0.5$ , the latter one indicates, for example, that the better the endurance is without the application of ascorbic acid, the bigger is the effect of the treatment, and vice versa. Moreover, the effect of treatment is nearly negligible (compared to its average) for individuals with a rather low endurance.

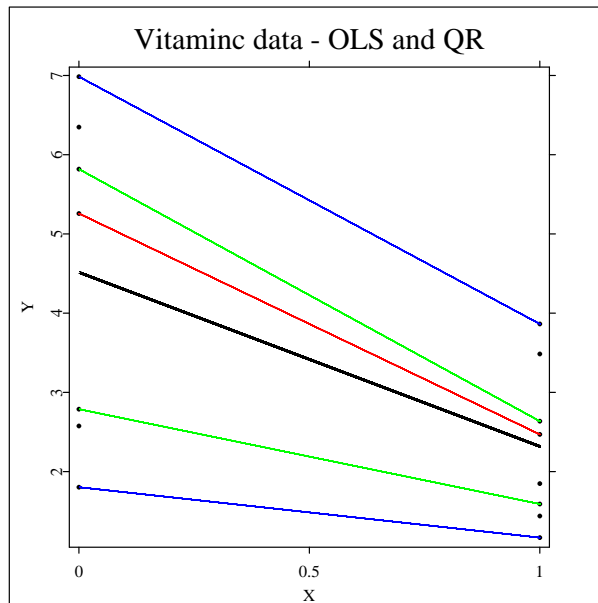


Figure 1: Least squares (the black thick line) and quantile regression for  $\tau = 0.1, 0.3, 0.5, 0.7, 0.9$  (blue, green, and red lines) [qr12.xpl](#)

## 2 Quantile regression estimation

Some key definitions related to quantile regression are introduced in this section. Besides that, we demonstrate how to use XploRe for the estimation of quantile regression models.

### 2.1 Definitions

Given a random sample  $y_1, \dots, y_n$ , it seems natural to find the approximation of a quantile (e.g., the median  $G_Y(1/2)$ ), in terms of the order statistics  $y_{[1]} \leq \dots \leq y_{[n]}$ , i.e., by means of sorting. The crucial point for the concept of quantile regression estimation is that the sample analogue of  $Q_Y(\tau)$  can be also found as the argument of the minimum of a specific objective function, because the optimization approach yields a natural generalization of the quantiles to the regression context. The  $\tau$ th sample quantile can be found as

$$\operatorname{argmin}_{\theta \in \mathbb{R}} \sum_{i=1}^n \rho_{\tau}(y_i - \theta), \quad (6)$$

where

$$\rho_{\tau}(x) = x \cdot \{\tau - I(x < 0)\} \quad (7)$$

(see figure 2) and  $I(\cdot)$  represents the indicator function.

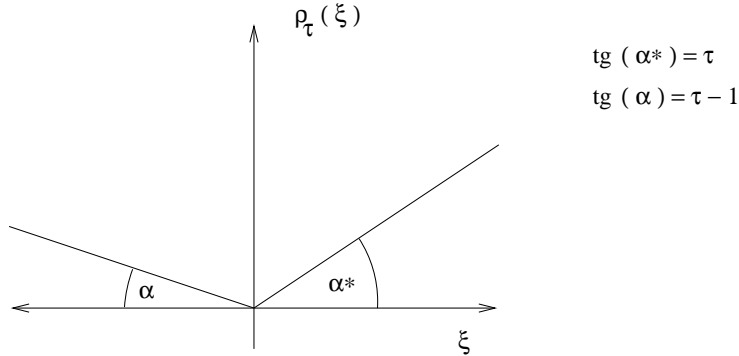


Figure 2: Quantile regression function  $\rho_{\tau}$

Any one-dimensional  $M$ -statistics (including the least squares estimator and (6)) for estimating a parameter of location

$$\hat{\mu} = \operatorname{argmin}_{\mu \in \mathbb{R}} \sum_{i=1}^n \psi(y_i - \mu)$$

can be readily extended to the regression context, i.e., to the estimation of conditional expectation function  $E(Y|X = x) = x^T \beta$  by solving

$$\hat{\beta} = \operatorname{argmin}_{\beta \in \mathbb{R}^p} \sum_{i=1}^n \psi(y_i - x_i^T \beta),$$

where  $y = (y_1, \dots, y_n)$  is a vector of responses and  $X = (x_1, \dots, x_n)^T$  is an  $n \times p$  matrix of explanatory variables. From now on,  $n$  will always refer to the number of observations and  $p$  to the number of unknown parameters. As the sample quantile estimation is just a special case of  $M$ -statistics for  $\psi = \rho_\tau$ , it can be adapted for the estimation of the conditional quantile function along the same way. Thus, the unknown parameters in the *conditional quantile function*  $Q_Y(\tau|X = x) = x_i^T \beta$  are to be estimated as

$$\hat{\beta}(\tau) = \operatorname{argmin}_{\beta \in \mathbb{R}^p} \sum_{i=1}^n \rho_\tau(y_i - x_i^T \beta). \quad (8)$$

The special case of  $t = 1/2$  is equivalent to minimizing the sum of absolute values of residuals, the well-known  $L_1$ -estimator.

Before proceeding to the description of how such an estimate can be computed in XploRe, two issues have to be discussed. First, given formula (8), it is clear that there probably exists no general closed-form solution like in the case of the least squares estimator. Therefore, it is natural to ask whether any solution of (8) exists at all and whether it is unique. The answer is positive under some rather general conditions. Let  $\mathcal{H}_m, m \in \{1, \dots, n\}$ , represents the set of all  $m$ -element subsets of  $\{1, \dots, n\}$ , and let for any  $m \in \{1, \dots, n\}$  and  $h \in \mathcal{H}_m$   $X_h$  denotes a  $m \times p$  submatrix of  $X$  composed from rows  $X_{h_1}, \dots, X_{h_m}$ . Similarly, let for a vector  $y$  be  $y_h = (y_{h_1}, \dots, y_{h_m})^T$ . Notice that this convention applies also for  $h \in \mathcal{H}_1$ , that is, for single numbers. The rows of  $X$  taken as column vectors are referred by  $x_1, \dots, x_n$ —therefore,  $X = (x_1, \dots, x_n)^T = (X_1^T, \dots, X_n^T)^T$ . Now we can write theorem 3.3 of Koenker and Bassett (1978) in the following way:

*Let  $(y, X)$  are regression observations,  $\tau \in (0, 1)$ . If  $(y, X)$  are in general position, i.e., the system of linear equations  $y_h = X_h b$  has no solution for any  $h \in \mathcal{H}_{p+1}$ , then there exists a solution to the quantile regression problem (8) of the form  $\hat{\beta}(\tau, h) = X_h^{-1} y_h, h \in \mathcal{H}_p$ , if and only if for some  $h \in \mathcal{H}_p$  holds*

$$(\tau - 1)1_p \leq \xi_p \leq \tau 1_p, \quad (9)$$

where  $\xi_h^T = \sum_{i \notin h} \rho_\tau\{y_i - X_i \hat{\beta}(\tau, h)\} \cdot X_i X_h^{-1}$ ,  $\rho_\tau$  is defined by (7), and  $1_p$  is the  $p \times 1$  vector of ones. Moreover,  $\hat{\beta}(\tau, h)$  is the unique solution if and only if the inequalities are strict, otherwise the solution set is the convex hull of several solutions of the form  $\hat{\beta}(\tau, h)$ .

The presented result deserves one additional remark. Whereas situations in which observations  $(y, X)$  are not in general position are not very frequent unless the response variable is of discrete nature, weak inequality in (9), and consequently multiple optimal solutions, can occur when all explanatory variables are discrete.

The second issue we have to mention is related to the numerical computation of estimates. The solution of (8) can be found by techniques of the *linear programming*, because

$$\min_{\beta \in \mathbb{R}^p} \sum_{i=1}^n \rho_\tau(y_i - X_i \beta)$$

may be rewritten as the minimization of a linear function subject to linear

constraints

$$\begin{aligned} \min_{\beta \in \mathbb{R}^p; u, v \in \mathbb{R}_+^n} \quad & \tau \cdot 1_n^T u + (1 - \tau) \cdot 1_n^T v \\ \text{subject to} \quad & y - X\beta = u - v. \end{aligned} \tag{10}$$

The linearity of the objective function and constraints implies that the solution has to lie in one of the vertices of the polyhedron defined by the constraints in (10). It is possible to derive that these vertices correspond to elements  $h$  of  $\mathcal{H}_p$  and take form

$$\begin{aligned} \hat{\beta}(\tau) &= X_h^{-1} y_h \\ u &= \max\{y - X\hat{\beta}(\tau), 0_n\}, \text{ especially } u_h = 0 \\ v &= -\min\{y - X\hat{\beta}(\tau), 0_n\}, \text{ especially } v_h = 0. \end{aligned}$$

Apparently, there are always at least  $p$  indices from  $\{1, \dots, n\}$  such that the corresponding residuals are equal to zero. Therefore, traversing between vertices of the polyhedron corresponds to switching between  $h_1, h_2 \in \mathcal{H}_p$ —hence the method belongs to the group of the so-called *exterior-point methods*. In order to find the optimal  $h$  (or equivalently vertex), we usually employ a modified simplex method (Koenker and D’Orey, 1987). Although this minimization approach has some considerable advantages (for small problems, it is even faster than the least squares computation), it becomes rather slow with an increasing number of observations. Thus, it is not very suitable for large problems ( $n \geq 100000$ ). Koenker and Portnoy (1997) developed an *interior-point method* that is rather fast when applied on large data sets.

## 2.2 Computation

```
z = rqfit(x, y, {tau, ci, alpha, iid, interp, tcrit})
      estimates noninteractively a quantile regression model
```

The quantlet of `metrics` library which serves for the quantile regression estimation is `rqfit`. We explain just the basic usage of `rqfit` quantlet in this section, other features will be discussed in the following sections. See appendix 5.1 for detailed description of the quantlet.

The quantlet expects at least two input parameters: an  $n \times p$  matrix  $X$  that contains  $n$  observations of  $p$  explanatory variables and an  $n \times 1$  vector  $y$  of  $n$  observed responses. If the intercept is to be included in the regression model, the  $n \times 1$  vector of ones can be concatenated to the matrix  $X$  in the following way:

```
x = matrix(rows(x)) ~ x.
```

Neither the matrix  $X$ , nor the vector  $y$  should contain missing (NaN) or infinite values (Inf, -Inf). Their presence can be identified by `isNaN` or `isNumber` and the invalid observations should be processed before running `rqfit`, e.g., omitted using `paf`.

Quantlet `rqfit` provides a noninteractive way for quantile regression estimation. The basic invocation method is quite simple:

```
z = rqfit(x,y,tau),
```

where parameter `tau` indicates which conditional quantile function  $Q_Y(\tau|X)$  has to be estimated. It is even possible to omit it:

```
z = rqfit(x,y).
```


In this case, the predefined value  $\tau = 0.5$  is used. The output of `rqfit` might be little bit too complex, but for now it is sufficient to note that `z.coefs` refers to the vector of the estimated coefficients  $\hat{\beta}(\tau)$  and `z.res` is the vector of regression residuals. If you want to have also the corresponding confidence intervals, you have to specify extra parameters in the call of `rqfit`—the fourth one, `ci`, equal to one, which indicates that you want to get confidence intervals, and optionally the fifth one, `alpha`, that specifies the nominal coverage probability for the confidence intervals (its default value is 0.1):

```
z = rqfit(x,y,tau,1,alpha).
```

Then `z.intervals` gives you the access to the  $p \times 2$  matrix of confidence intervals (the first column contains lower bounds, the second one upper bounds). Read section 4.3 for more information.

To have a real example, let us use data set `nicfoo` supplied with XploRe. The data set is two-dimensional, having only one explanatory variable  $x$ , a household's net income, in the first column and the response variable  $y$ , food expenditures of the household, in the second column (see Data Sets). In order to run, for example, the median regression ( $\tau = 0.5$ ) of  $y$  on constant term,  $x$  and  $x^2$ , you have to type at the command line or in the editor window

```
data = read("nicfoo")
x = matrix(rows(data)) ~ data[,1] ~ (data[,1]^2)
y = data[,2]
z = rqfit(x,y)
z.coefs
```

 `qr03.xpl`

Do not forget to load `metrics` library before running `rqfit`:

```
library("metrics").
```

The result of the above example should appear in the XploRe output window as follows:

```
Contents of coefs
[1,] 0.12756
[2,] 1.1966
[3,] -0.24616
```



### 3 Essential properties of QR

The practical usefulness of any estimation technique is determined, besides other factors, by its invariance and robustness properties, because they are essential for coherent interpretation of regression results. Although some of these properties are often perceived as granted (probably because of their validity in the case of the least squares regression), it does not have to be the case for more evolved regression procedures. Fortunately, quantile regression preserves many of these invariant properties, and even adds to them several other distinctive qualities, which we are going to discuss now.

#### 3.1 Equivariance

In many situations it is preferable to adjust the scale of original variables or reparametrize a model so that its result has a more natural interpretation. Such changes should not affect our qualitative and quantitative conclusions based on the regression output. Invariance to a set of some elementary transformations of the model is called *equivariance* in this context. Koenker and Bassett (1978) formulated four equivariance properties of quantile regression. Once we denote the quantile regression estimate for a given  $\tau \in (0, 1)$  and observations  $(y, X)$  by  $\hat{\beta}(\tau; y, X)$ , then for any  $p \times p$  nonsingular matrix  $A$ ,  $\gamma \in \mathbb{R}^p$ , and  $a > 0$  holds

1.  $\hat{\beta}(\tau; ay, X) = a\hat{\beta}(\tau; y, X)$
2.  $\hat{\beta}(\tau; -ay, X) = a\hat{\beta}(1 - \tau; y, X)$
3.  $\hat{\beta}(\tau; y + X\gamma, X) = \hat{\beta}(\tau; y, X) + \gamma$
4.  $\hat{\beta}(\tau; y, XA) = A^{-1}\hat{\beta}(\tau; y, X)$ .

This means, for example, that if we use as the measurement unit of  $y$  millimeters instead of meters, that is  $y$  multiplied by 1000, then our estimate scales appropriately:  $\hat{\beta}(\tau; y[\text{mm}], X) = 1000 \cdot \hat{\beta}(\tau; y[\text{m}], X)$ .

#### 3.2 Invariance to monotonic transformations

Quantiles exhibit besides “usual” equivariance properties also equivariance to monotone transformations. Let  $f(\cdot)$  be a nondecreasing function on  $\mathbb{R}$ —then it immediately follows from the definition of the quantile function that for any random variable  $Y$

$$Q_{f(Y)}(\tau) = f\{Q_Y(\tau)\}. \tag{11}$$

In other words, the quantiles of the transformed random variable  $f(Y)$  are the transformed quantiles of the original variable  $Y$ . Please note that this is not the case of the conditional expectation— $E\{f(Y)\} \neq f(EY)$  unless  $f(\cdot)$  is a linear function. This is why a careful choice of the transformation of the dependent variable is so important in various econometrics models when the ordinary least squares method is applied (unfortunately, there is usually no guide which one is correct).

We can illustrate the strength of equivariance with respect to monotone transformation on the so-called censoring models. We assume that there exists, for example, a simple linear regression model with i.i.d. errors

$$y_i = x_i^T \beta + \varepsilon_i, \quad i \in \{1, \dots, n\},$$

and that the response variable  $y_i$  is unobservable for some reason. Instead, we observe  $\tilde{y}_i = \max\{y_i, a\}$ , where  $a \in \mathbb{R}$  is the censoring point. Because of censoring, the standard least squares method is not consistent anymore (but a properly formulated maximum likelihood estimator can be used). On the contrary, the quantile regression estimator, thanks to the equivariance to monotone transformations, does not run into such problems as noted by Powell (1986). Using  $f(x) = \max\{x, a\}$  we can write

$$Q_{\tilde{y}_i}(\tau|x_i) = Q_{f(y_i)}(\tau|x_i) = f\{Q_{y_i}(\tau|x_i)\} = f(x_i^T \beta) = \max\{x_i^T \beta, a\}.$$

Thus, we can simply estimate the unknown parameters by

$$\hat{\beta}(\tau) = \operatorname{argmin}_{\beta \in \mathbb{R}^p} \sum_{i=1}^n \rho_\tau(y_i - \max\{x_i^T \beta, a\}).$$

### 3.3 Robustness

Sensitivity of an estimator to departures from its distributional assumptions is another important issue. The long discussion concerning relative qualities of the mean and median is an example of how significant this kind of robustness (or sensitivity) can be. The sample mean, being a superior estimate of the expectation under the normality of the error distribution, can be adversely affected even by a single observation if it is sufficiently far from the rest of data points. On the other hand, the effect of such a distant observation on the sample median is bounded no matter how far the outlying observation is. This robustness of the median is, of course, outweighed by lower efficiency in some cases. Other quantiles enjoy similar properties—the effect of outlying observations on the  $\tau$ th sample quantile is bounded, given that the number of outliers is lower than  $n \cdot \min\{\tau, 1 - \tau\}$ .

Quantile regression inherits these robustness properties since the minimized objective functions in the case of sample quantiles (6) and in the case of quantile regression (8) are the same. The only difference is that regression residuals  $r_i(\beta) = y_i - x_i^T \beta$  are used instead of deviations from mean  $y_i - \mu$ . Therefore, quantile regression estimates are reliable in presence of outlying observations that have large residuals. To illustrate this property, let us use a set of ten simulated pseudo-random data points to which one outlying observations is added (the complete code of this example is stored in `qr04.xpl`).

```


outlier = #(0.9,4.5)      ; outlying observation
;
; data initialization
;

```

```

randomize(17654321)      ; sets random seed
n = 10                  ; number of observations
beta = #(1, 2)         ; intercept and slope
x = matrix(n)~uniform(n) ; randomly generated data
x = sort(x)
x = x | (1~outlier[1])  ; add outlier
;
; generate regression line and noisy response variable
;
regline = x * beta
y = regline[1:n] + 0.05 * normal(n)
y = y | outlier[2]      ; add outlier

```


 qr04.xpl

Having the data in hand, we can advance to estimation in the same way as in section 2.2. To make results more obvious, they are depicted in a simple graph.

```

z = rqfit(x,y,0.5)      ; estimation
betahat = z.coefs
;
; create graphical display, draw data points and regressions line
;
d = createdisplay(1,1)
data = x[,2]~y          ; data points
outl = outlier[1]~outlier[2] ; outlier
setmaskp(outl,1,12,15)  ; is blue big star
;
line = x[,2]~regline    ; true regression line
setmaskp(line, 0, 0, 0)
setmaskl(line, (1:rows(line))', 1, 1, 1)
;
yhat = x * betahat
qrline = x[,2]~yhat     ; estimated regression line
setmaskp(qrline, 0, 0, 0)
setmaskl(qrline, (1:rows(qrline))', 4, 1, 3)
;
; display all objects
;
show(d, 1, 1, data[1:n], outl, line, qrline)
setgopt(d, 1, 1, "title", "Quantile regression with outlier")

```

 qr04.xpl

As a result, you should see a graph like one on figure 3, in which observations are denoted by black circles and the outlier is represented by the big blue star in the right upper corner of the graph. Further, the blue line depicts the true regression line, while the thick red line shows the estimated regression line.

As you may have noticed, we mentioned the robustness of quantile regression with respect to observations that are far in the direction of the dependent vari-

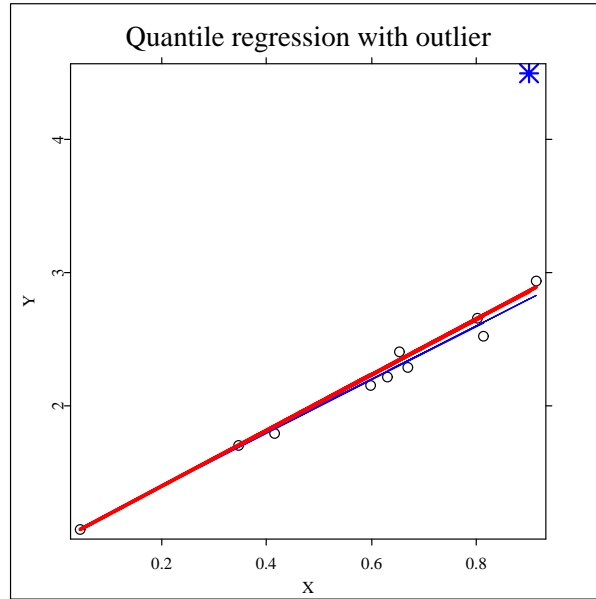


Figure 3: Robustness of QR estimates to outliers. [qr04.xpl](#)

able, i.e., that have large residuals. Unfortunately, this cannot be said about the effect of observations that are distant in the space of explanatory variables—a single point dragged far enough toward infinity can cause that all quantile regression hyperplanes go through it. As an example, let us consider the previous data set with a different outlier:

```
outlier = #(3,2)
```

Running example [qr05.xpl](#) with this leverage point gives dramatically different results than in the previous case—see figure 4.

## 4 Inference for QR

In this section we will discuss possibilities regarding statistical inference for quantile regression models. Although there are nearly no usable finite sample results for statistical inference compared to the often used theory of least squares under the normality assumption, the asymptotic theory offers several competing methods, namely tests based on the Wald statistics, rank tests, and likelihood ratio-like tests. Some of them are discussed in this section.

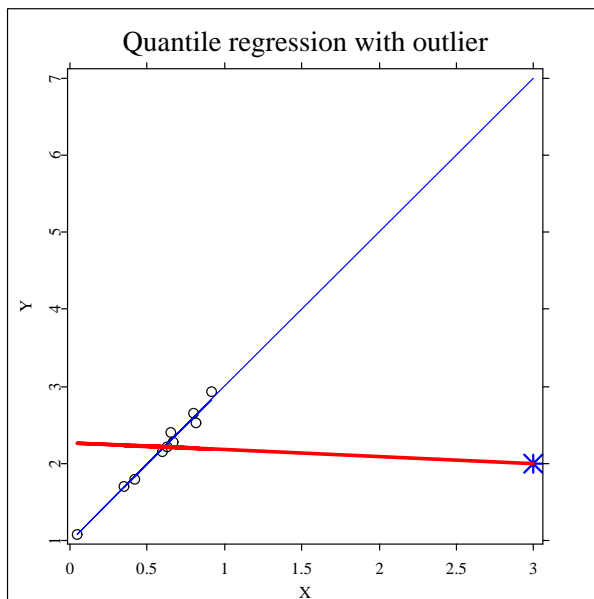


Figure 4: Non-robustness of QR estimates to leverage points. [qr05.xpl](#)

## 4.1 Main asymptotic results

The asymptotic behavior of ordinary sample quantiles generalizes relatively easily to the quantile regression case. A fundamental result was derived in Koenker and Bassett (1978). Let  $\{\hat{\beta}(\tau)|\tau \in (0, 1)\}$  is the *quantile regression process* and consider the classical regression model

$$y_i = X_i\beta + \varepsilon_i, \quad i \in \{1, \dots, n\}$$

with i.i.d. errors  $\varepsilon_i \sim F$ ,  $F$  has a density  $f$  such that for all  $x \in \mathbb{R}$ ,  $0 < F(x) < 1$ , holds  $f(x) > 0$ . Assuming that  $n^{-1} \sum_{i=1}^n X_i^T X_i \equiv Q_n \rightarrow Q_0$  as  $n \rightarrow +\infty$ , where  $Q_0$  is a positive definite matrix, Koenker and Bassett (1978) showed that the joint asymptotic distribution of  $m$   $p$ -variate quantile regression estimators  $\hat{B}_n = (\hat{\beta}_n(\tau_1)^T, \dots, \hat{\beta}_n(\tau_m)^T)^T$  takes the form

$$\sqrt{n} (\hat{B}_n - B) = \left[ \sqrt{n} \left\{ \hat{\beta}_n(\tau_j) - \beta(\tau_j) \right\} \right]_{j=1}^m \xrightarrow{F} N(0, \Omega \otimes Q_0^{-1}), \quad (12)$$

where  $\Omega = (\omega_{ij})_{i,j=1}^m = [(\min\{\tau_i, \tau_j\} - \tau_i\tau_j)/(f\{F^{-1}(\tau_i)\}f\{F^{-1}(\tau_j)\})]_{i,j=1}^m$  and  $F^{-1}$  refers again to the quantile function corresponding to the distribution  $F$ . Having the asymptotic normality of the estimates in hand, one can use the Wald test for testing hypotheses that can be expressed as a set of linear restrictions on  $\hat{\beta}_n(\tau_1), \dots, \hat{\beta}_n(\tau_m)$  for some  $\tau_1, \dots, \tau_m, m \in \mathbb{N}$ .

The situation is little bit more complicated in the case of non-i.i.d. errors, but the normality of the quantile regression estimator is preserved under heteroscedasticity. If we denote the estimate of coefficients for  $\tau \in (0, 1)$  by  $\hat{\beta}_n(\tau)$ , then for

$$n \rightarrow +\infty \quad \sqrt{n} \left\{ \hat{\beta}_n(\tau) - \beta(\tau) \right\} \xrightarrow{F} N(0, H^{-1}(\tau) J(\tau) H^{-1}(\tau)), \quad (13)$$

where

$$J(\tau) = \lim_{n \rightarrow +\infty} J_n(\tau) = \lim_{n \rightarrow +\infty} \tau(1-\tau)n^{-1} \sum_{i=1}^n X_i^T X_i$$

and

$$H(\tau) = \lim_{n \rightarrow +\infty} H_n(\tau) = \lim_{n \rightarrow +\infty} n^{-1} \sum_{i=1}^n X_i^T X_i f_i\{F^{-1}(\tau)\}$$

( $f_i(\cdot)$  denotes conditional density of  $y$  given  $X$ ). A guide to the estimation of  $H_n(\tau)$  is given, for example, in Powell (1989), and Koenker and Portnoy (2000).

## 4.2 Wald test

As was already mentioned in the previous section, the asymptotic normality of quantile regression estimates gives us possibility to test various linear hypotheses formulated through regression quantiles by means of the Wald test. For a general linear hypothesis about the vector  $B = (\beta(\tau_1)^T, \dots, \beta(\tau_m)^T)^T$

$$H_0 : HB = h \quad (14)$$

( $H$  being a  $J \times mp$  matrix,  $h$  a vector of length  $J$ ), the Wald test statistics can be written as

$$W_n = (H\hat{B} - h)^T [H \{ \Omega \otimes (X^T X)^{-1} \} H^T]^{-1} (H\hat{B} - h), \quad (15)$$

which has under the validity of  $H_0$  asymptotically  $\chi^2$  distribution with  $J$  degrees of freedom. The only difficult point is the estimation of the asymptotic covariance matrix  $\Omega$ . There are several strategies available, see Koenker and Portnoy (2000) for their discussion.

To present a possible application of this test procedure, let us explain a simple test of heteroscedasticity. Following Koenker and Bassett (1982a), homoscedasticity is equivalent to the equality of slope parameters across quantiles. Consider, for example, model (2)

$$y_i = \alpha + \delta D_i + \varepsilon_i, \quad i \in \{1, \dots, n\}.$$

Then the test of the equality of the slope parameter  $\delta$  across quantiles  $\tau_1, \tau_2$  is nothing but the test of the hypothesis

$$H_0 : \delta(\tau_2) - \delta(\tau_1) = 0.$$

Since the  $\tau$ th quantile regression estimate  $\hat{\delta}(\tau)$  is in this case simply the difference of the  $\tau$ th quantiles for samples  $\{y_i | D_i = 0\}$  and  $\{y_i | D_i = 1\}$  (remember,  $D_i \in \{0, 1\}$ ),

$$\begin{aligned} \hat{\delta}(\tau_2) - \hat{\delta}(\tau_1) &= \\ &= \{Q_Y(\tau_2 | D_i = 1) - Q_Y(\tau_2 | D_i = 0)\} - \{Q_Y(\tau_1 | D_i = 1) - Q_Y(\tau_1 | D_i = 0)\} \\ &= \{Q_Y(\tau_2 | D_i = 1) - Q_Y(\tau_1 | D_i = 1)\} - \{Q_Y(\tau_2 | D_i = 0) - Q_Y(\tau_1 | D_i = 0)\} \end{aligned}$$

Further it is possible to derive the variance  $\sigma^2(\tau_1, \tau_2)$  of  $\hat{\delta}(\tau_2) - \hat{\delta}(\tau_1)$  from formula (12), and to construct the statistics

$$T_n = \{\hat{\delta}(\tau_2) - \hat{\delta}(\tau_1)\} / \hat{\sigma}(\tau_1, \tau_2)$$

with an asymptotically normal distribution. For a practical realization of the test, it would be necessary to approximate  $\hat{\sigma}(\tau_1, \tau_2)$ , but this goes beyond the scope of this tutorial. For more information on the so-called *sparsity estimation*, see for example Siddiqui (1960), Bofinger (1975), Welsh (1988), or Hall and Sheather (1988).

### 4.3 Rank tests

```
z = rqfit(x,y{,tau,ci,alpha,iid,interp,tcrit})
      estimates noninteractively a given quantile regression model

chi = rrstest(x0,x1,y{,score})
      performs the regression rankscore test
```

The classical theory of rank test (Hájek and Šidák, 1967) employs the *rankscore functions*

$$a_{ni}(t) = \begin{cases} 1 & \text{if } t \leq (R_i - 1)/n \\ R_i - tn & \text{if } (R_i - 1)/n < t \leq R_i/n, \\ 0 & \text{if } R_i/n < t \end{cases} \quad (16)$$

where  $R_i$  represents the rank of the  $i$ th observation  $y_i$  in  $(y_1, \dots, y_n)$ . The integration of  $a_{ni}(t)$  with respect to various score generating functions  $\psi$  produces vectors of scores, rank-like statistics, which are suitable for testing. For instance, integrating  $a_{ni}(t)$  using the Lebesgue measure (generating function  $\psi(t) = t$ ) generates the Wilcoxon scores

$$s_i = \int_0^1 a_{ni}(t) dt = \frac{R_i - 1/2}{n}, \quad i = 1, \dots, n, \quad (17)$$

$\psi(t) = \text{sgn}(t - 1/2)$  yields the sign scores  $s_i = a_{ni}(1/2)$  and so on. The way how this theory can be applied in the regression context was found by Gutenbrunner and Jurečková (1992), who noticed that the rankscores (16) may be viewed as a solution of a linear-programming model

$$\begin{aligned} & \max_{a \in \langle 0,1 \rangle^n} && y^T a \\ & \text{subject to} && Xa = (1 - t)X1_n. \end{aligned} \quad (18)$$

The important property of this model is its duality to model (10) that defines regression quantiles—see also Koenker and D’Orey (1993) for details.

The uncovered link to rankscore tests enabled to construct tests of significance of regressors in quantile regression without necessity to estimate some nuisance parameters (such as  $\Omega$  in the case of the Wald test). Given the model  $y = X_0\beta_0 + X_1\beta_1 + \varepsilon, \beta_0 \in \mathbb{R}^{p-J}, \beta_1 \in \mathbb{R}^J$ , Gutenbrunner, Jurečková, Koenker,

and Portnoy (1993) designed a test of hypothesis  $H_0 : \beta_1 = 0$  based on the regression rankscore process. It is constructed in the following way: first, compute  $\{a_{ni}(t)\}_{i=1}^n$  at the restricted model  $y = X_0\beta_0 + \varepsilon$  and the corresponding rankscores vector  $s = (s_i)_{i=1}^n = \left\{ - \int a_{ni}(t)d\psi(t) \right\}_{i=1}^n$ . Next, form the vector

$$S_n = n^{-1/2}X_1s,$$

which converges in distribution to  $N(0, A^2(\psi)Q_0)$  under the null hypothesis, where  $A^2(\psi) = \int_0^1 \psi^2(t)dt$ ,  $Q_0 = \lim_{n \rightarrow \infty} (X_1 - \hat{X}_1)^T(X_1 - \hat{X}_1)/n$ , and  $\hat{X}_1 = X_0(X_0^T X_0)^{-1}X_0^T X_1$ . Finally, the test statistics


$$T_n = S_n^T Q^{-1} S_n / A^2(\psi) \quad (19)$$

has asymptotically  $\chi^2_J$  distribution. To do this test in XploRe (given some  $(y, X)$ ), the only thing you have to do is to split the matrix  $X$  to two parts  $X_0$  and  $X_1$  (leaving the intercept usually in  $X_0$ ) and to call the quantlet `rrstest`, which requires  $X_0, X_1$ , and  $y$  on input. Optionally, you can specify the type of the score generating function to be used (see appendix 5 for more details); the Wilcoxon scores are employed by default. For demonstration of the quantlet, let us use again a simulated data set—we generate a pseudo-random  $100 \times 3$  matrix  $X$  and two response vectors  $y_1 = X \cdot (1, 2, -1)^T + \varepsilon_1$  and  $y_2 = X \cdot (1, 2, 0)^T + \varepsilon_2$ . The resulting test statistics differ significantly (in the first case, the test statistics is significant, while in the latter one not) as is documented both by their values and  $p$ -values.

```

; simulate data matrix
n = 100
randomize(1101)
x = matrix(n) ~ uniform(n,2)
; generate y1 and y2
y1 = x[,1] + 2*x[,2] - x[,3] + normal(n)
y2 = x[,1] + 2*x[,2] + normal(n)
; test the hypothesis that the coefficient of x[,3] is zero
; first case
chi1 = rrstest(x[,1:2], x[,3], y1)
chi1
cdfc(chi1,1)
; second case
chi2 = rrstest(x[,1:2], x[,3], y2)
chi2
cdfc(chi2,1)

```

 qr06.xpl.xpl

When the script ends, the XploRe output should look like

```

Contents of chi1
[1,] 19.373
Contents of cdfc
[1,] 0.99999

```



```

Contents of chi2
[1,] 0.018436
Contents of cdfc
[1,] 0.10801

```

The existence of a testing strategy for quantile regression motivated the search for a reverse procedure that would provide a method for estimating confidence intervals without actual knowledge of the asymptotic covariance matrix. Quite general results in this area were derived in Hušková (1994). Although the computation of these confidence intervals is rather difficult, there are some special cases for which the procedure is tractable (Koenker, 1994). An adaptation of the technique for non-i.i.d. errors have been done recently. Now, it was already mentioned that quantlet `rqfit` can compute also confidence intervals for quantile regression estimates. This is done by the above mentioned method of *inverting rank tests*, which has several practical implications. Above all, the computation of confidence intervals at an exact significance level  $\alpha$  would require knowledge of the entire quantile regression process  $\{\hat{\beta}(\tau)|\tau \in (0, 1)\}$ . This is not possible because we always work with finite samples, hence we have only an approximation of the process in the form  $\{\hat{\beta}(\tau)|\tau \in \{\tau_1, \dots, \tau_m\}\}$ . Therefore, two confidence intervals are computed for every parameter at a given significance level  $\alpha$  (parameter `alpha`)—the largest one with true significance level higher than  $\alpha$ , let us call it  $I_1$ , and the smallest one with true significance level lower than  $\alpha$ ,  $I_2$ . Then, according to the value of parameter `interp`, various results are returned. If its value is nonzero or the parameter is not specified, e.g.,

```
z = rqfit(x, y, 0.5, 1),
```

then the bounds of the returned intervals are interpolated from the lower and upper bounds of the pairs of intervals, and the result in `z.intervals` is a  $p \times 2$  matrix of confidence intervals—the first column holds the interpolated lower bounds, the second one upper bounds. In the other case, i.e., `interp = 0`,

```
z = rqfit(x, y, 0.5, 1, 1, 0),
```

`z.intervals` is a  $p \times 4$  matrix of pairs of confidence intervals—the first column contains the lower bounds of intervals  $I_2$ , the second one lower bounds of  $I_1$ 's, the third one embody upper bounds of  $I_1$ 's, and the fourth one upper bounds of intervals  $I_2$ , which implies that the bounds in rows of `z.intervals` are numerically sorted. In this case, the matrix `z.pval` will contain the correct  $p$ -values corresponding to bounds in `z.intervals`.

Finally, before closing this topic, we make one small remark on `iid` switch. Its value specifies, whether the procedure should presume i.i.d. errors (this is the default setting), or whether it should make some non-i.i.d. errors adjustments. We can disclose the effect of this parameter using the already discussed `nicfoo` data. The data seems to exhibit some kind of heteroscedasticity (as is often the case if the set of significant explanatory variables involve individuals with diverse levels of income), see figure 5.

To compare the resulting confidence intervals for median regression under the

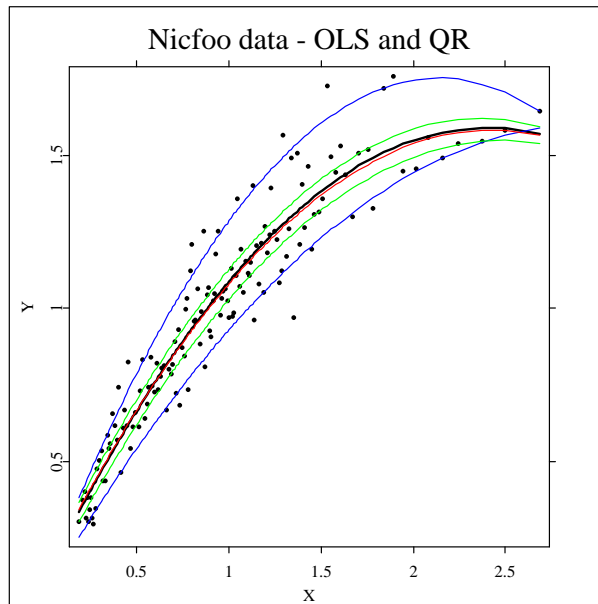


Figure 5: Least squares (the black thick line) and quantile regression for  $\tau = 0.1, 0.3, 0.5, 0.7, 0.9$  (blue, green, and red lines) [qr07.xpl](#)

i.i.d. errors assumption and without it, you can type at the command line or in the editor window

```

data = read("nicfoo")
x = matrix(rows(data)) ~ data[,1] ~ (data[,1]^2)
y = data[,2]
;
z = rqfit(x,y,0.5,1,0.1,1)
z.intervals
;
z = rqfit(x,y,0.5,1,0.1,0)
z.intervals

```

[qr08.xpl](#)

Once you run this example, the output window will contain the following results:

```

Contents of intervals
[1,] 0.12712 0.13194
[2,] 1.1667 1.2362
[3,] -0.24616 -0.24608

```

```

Contents of intervals
[1,] 0.024142 0.20241
[2,] 1.0747 1.3177

```

[3,] -0.29817 -0.2014

Please, notice the difference between the first group of intervals (i.i.d. errors assumption) and the second one.

## 5 Description of quantlets for QR

The complete description of XploRe quantlets for quantile regression and the related test follows in next two sections. There are also several final remarks and notes that are important for the use of these quantlets. In both sections holds that all the input parameters are discussed first, the output values are commented later.

### 5.1 Quantlet `rqfit`

```
z = rqfit(x,y{,tau,ci,alpha,iid,interp,tcrit})
      estimates noninteractively a quantile regression model
```

The main purpose of the quantlet is to estimate the quantile regression model given by regression observations  $(y,x)$  for a quantile  $\tau$ . For the sake of simplicity, we will assume throughout this section that the output of `rqfit` is stored in a list called `z` as shown in the template.

- x** An  $n \times p$  matrix of explanatory variables. It should not contain missing (NaN) or infinite values (Inf, -Inf). See also section 2.2.
- y** An  $n \times 1$  vector of observations for the dependent variable. It should not contain missing (NaN) or infinite values (Inf, -Inf). See also section 2.2.
- tau** A regression quantile to be estimated. If the parameter is omitted, the predefined value 0.5 is used. There are two different modes of operation, depending on the value of this parameter:
- tau inside**  $\langle 0, 1 \rangle$ : A single quantile solution for the given  $\tau$  is computed and returned. The estimated parameters are stored in `z.coefs` and the corresponding residuals are accessible via `z.res`.
  - tau outside**  $\langle 0, 1 \rangle$ : Solutions for all possible quantiles are sought and the approximation of the quantile regression process  $\{\hat{\beta}(\tau) | \tau \in \{\tau_1, \dots, \tau_m\}\}$  is computed. In this case, `z.coefs` is a matrix containing  $\hat{\beta}(\tau_1), \dots, \hat{\beta}(\tau_m)$ . The array containing both  $\tau_1, \dots, \tau_m$  and  $\hat{\beta}(\tau_1), \dots, \hat{\beta}(\tau_m)$  is to be found in `z.sol`.  
It should be emphasized that this regime can be quite memory and CPU intensive. On typical machines it is not recommended for problems with  $n > 10000$ .

- `ci` A logical flag for confidence intervals (nonzero values mean *true*) with the default value equal to 0 (*false*). If `ci` is zero, only regression coefficients and the corresponding residuals are calculated. In the other cases, confidence intervals for the parameters are computed using the rank inversion method of Koenker (1994) and returned in `z.intervals`.  
Be aware that the computation of confidence intervals can be rather slow for large problems. Note also that rank inversion works only for  $p > 1$ , but this should not be very restrictive, since you include intercept in the regression in most cases.
- `alpha` A nominal coverage probability for the confidence intervals, which default value is 0.1. The value is called nominal because the confidence intervals are computed from an approximation of the quantile regression process  $\{\hat{\beta}(\tau) | \tau \in \{\tau_1, \dots, \tau_m\}\}$ . Therefore, the “available” significance levels are given by the breakpoints  $\tau_1, \dots, \tau_m$ , and consequently, by the size of the used data set. Given a nominal significance level `alpha`, some breakpoints are chosen so that they most closely approximate the required coverage probability. Then either two confidence intervals are returned (the best ones with significance levels just above and below `alpha`), or interpolation takes place. See section 4.3 and the description of parameter `interp` for more details.
- `iid` A logical flag indicating i.i.d. errors (nonzero values mean *true*), the value used if the parameter is omitted is 1 (*true*). If `iid` is nonzero, then the rank inversion method employs the assumption of i.i.d. errors and the original version of the rank inversion intervals is used (Koenker, 1994). In the opposite case, possible heterogeneity of errors is taken into account. See also section 4.3.
- `interp` A logical flag for interpolated confidence intervals (again, nonzero values mean *true*), the default value is 1 (*true*). As confidence intervals (and any other test statistics) based on order statistics are discrete, it is reasonable to consider intervals that are an interpolation of two intervals with significance levels just below the specified `alpha` and just above the specified `alpha`. If `interp` is nonzero (and, of course, `ci` is nonzero, otherwise no confidence intervals are computed), `rqfit` returns for every parameter a single interval based on linear interpolation of the two intervals. Therefore, `z.intervals` is a  $p \times 2$  matrix, each row contains a confidence interval for the corresponding parameter in `z.coefs`. On the other hand, if `interp` equals to zero, two “exact” intervals with significance levels above and below `alpha` (that two on which the interpolation would be based) are returned. Thus, `z.intervals` is a  $p \times 4$  matrix, each row contains first the lower bounds, then the upper bounds of confidence intervals, i.e., all four bounds are sorted in ascending order. Moreover, matrices `z.cval` and `z.pval`, which contain the critical values and  $p$ -values of the upper and lower bounds of intervals, are returned in this case. See also sections 2.2 and 4.3.

**tcrit** A logical flag for finite sample adjustment using  $t$ -statistics, its default value is 1 (*true*). In the default case, the Student critical values are used for the computation of confidence intervals, otherwise, normal ones are employed.  
It might sometimes happen that confidence intervals for some parameter have a form  $(-\text{Inf}, \text{Inf})$  or  $(-10^{300}, 10^{300})$ . Setting this parameter to zero, i.e., decreasing the absolute value of critical values, can help you to obtain finite confidence intervals.

Now, the discussion of output values is ahead.

**z.coefs** A  $p \times 1$  or  $p \times m$  matrix. If parameter **tau** is inside interval  $\langle 0, 1 \rangle$ , the only column of **z.coefs** contains the estimated coefficients. If **tau** falls outside  $\langle 0, 1 \rangle$ , **z.coefs** is a  $p \times m$  matrix that contains the estimated coefficients for all breakpoints  $\tau_1, \dots, \tau_m$ . This matrix is actually composed of the last  $p$  rows of **z.sol** array, see **z.sol** for more detailed description. See also section 2.2.

**z.res** An  $n \times 1$  vector of regression residuals, that is returned only if **tau** is inside interval  $\langle 0, 1 \rangle$ . See also section 2.2.

**z.intervals** A  $p \times 2$  or  $p \times 4$  matrix containing confidence intervals that are computed only if **ci** is nonzero and **tau** belongs to interval  $\langle 0, 1 \rangle$ . In the first case, one interpolated interval per parameter is returned, in the second one, two intervals per parameter are returned (bounds of the intervals are sorted in ascending order). See the description of parameters **alpha** and **interp** for more details as well as sections 2.2 and 4.3.

**z.cval** A  $p \times 4$  matrix of critical values for (non-interpolated) confidence intervals. It is returned only when **tau** is inside interval  $\langle 0, 1 \rangle$ , **ci** is nonzero, and **interp** equals zero. See the description of parameter **interp** for further information.

**z.pval** A  $p \times 4$  matrix of  $p$ -values (probabilities) for (non-interpolated) confidence intervals. It is returned only when **tau** falls to interval  $\langle 0, 1 \rangle$ , **ci** is nonzero, and **interp** equals zero. See the description of parameter **interp** for further information.

**z.sol** The primal solution array, which is a  $(p + 3) \times m$  matrix. Its first row contains the breakpoints  $\tau_1, \dots, \tau_m$  of the quantile function, i.e., the values in  $(0, 1)$  at which the solution changes. The second row contains the corresponding quantiles evaluated at the mean design point, i.e., the inner product of  $\overline{X} = (\overline{X}_{\cdot, i})_{i=1}^p$  and  $\hat{\beta}(\tau_i), i = 1, \dots, m$ . The third row contains the value of the objective function evaluated at the corresponding  $\tau_i, i = 1, \dots, m$ , see (8), and the last  $p$  rows of the matrix give  $\hat{\beta}(\tau_1), \dots, \hat{\beta}(\tau_m)$ . The solution  $\hat{\beta}(\tau_i)$  prevails from  $\tau_i$  to  $\tau_{i+1}, i = 1, \dots, m$ . Portnoy (1989) showed that  $m = \mathcal{O}_p(n \ln n)$ . See also section 4.3.

`z.dsol` The dual solution array, an  $n \times m$  matrix containing the dual solution corresponding to `z.sol`. The  $ij$ th entry,  $i \in \{1, \dots, n\}$ ,  $j \in \{1, \dots, m\}$ , is equal to  $x$ , where

$$\begin{aligned} x &= 1 && \text{if } y_i > x_i \hat{\beta}(\tau_j), \\ x &= 0 && \text{if } y_i < x_i \hat{\beta}(\tau_j), \\ 0 &< x < 1 && \text{otherwise.} \end{aligned}$$

See Gutenbrunner and Jurečková (1992) for a detailed discussion of the statistical interpretation of `z.dsol`. The use of `z.dsol` in statistical inference is described in Gutenbrunner, Jurečková, Koenker, and Portnoy (1993).

## 5.2 Quantlet `rrstest`

```
chi = rrstest(x0,x1,y{,score})
      executes the regression rankscore test
```

The main purpose of the quantlet `rrstest` is to test significance of some explanatory variables in regression using rankscore tests. For this purpose, the quantlet invokes already described `rqfit` with parameter `tau` equal to  $-1$ . Therefore, the note related to this choice of `tau` applies here. The test is described in section 4.3.

`x0` An  $n \times (p - J)$  matrix of maintained regressors. If there is an intercept term in the regression, `x0` should contain it. The same restrictions as in the case of `x` and `rqfit` applies on `x0`—it should not contain missing (`NaN`) or infinite values (`Inf`, `-Inf`).

`x1` An  $n \times J$  matrix of regressors under test. The explanatory variables placed in `x1` are tested for their significance in regression. Again, `x1` should not contain missing (`NaN`) or infinite values (`Inf`, `-Inf`).

`y` An  $n \times 1$  vector of observations for the response variable. It should not contain missing (`NaN`) or infinite values (`Inf`, `-Inf`).

`score` The desired score function for test. Possible values are:

- `score = 1`: Wilcoxon scores (this is the default case); they are asymptotically optimal for logistic error model.
- `score = 2`: Normal scores, which are asymptotically optimal for Gaussian error model.
- `score = 3`: Sign scores, which are asymptotically optimal for Laplace error model.
- `score ∈ (0, 1)`: A generalization of sign scores to the quantile given by the value in  $(0, 1)$ , i.e., scores generated by the function  $\psi(t) = \text{sgn}(t - \text{score})$ .

See also section 4.3.

Let us discuss now the only output value of the quantlet.

**chi**      A test statistics that is asymptotically distributed according to  $\chi^2$  with  $J$  degrees of freedom. See also (19) in section 4.3.

## References

- Bassett, G. W. and Koenker, R. (1982). An empirical quantile function for linear models with iid errors, *Journal of the American Statistical Association* **77**: 407–415.
- Bofinger, E. (1975). Estimation of a density function using order statistics, *Australian Journal of Statistics* **17**: 1–7.
- Doksum, K. (1974). Empirical probability plots and statistical inference for nonlinear models in the two sample case, *Annals of Statistics* **2**: 267–277.
- Falk, M. (1986). On the estimation of the quantile density function, *Statistics & Probability Letters* **4**: 69–73.
- Fitzenberger, B. (1996). A Guide to Censored Quantile Regression, forthcoming in *Handbook of Statistics* **15**, North-Holland, New York.
- Gutenbrunner, C. and Jurečková, J. (1992). Regression quantile and regression rank score process in the linear model and derived statistics, *Annals of Statistics* **20**: 305–330.
- Gutenbrunner, C., Jurečková, J., Koenker, R., and Portnoy, S. (1993). Tests of linear hypotheses based on regression rank scores, *Journal of Nonparametric Statistics* **2**: 307–333.
- Hájek, J. and Šidák, Z. (1967). *Theory of rank tests*, Academia, Prague.
- Hall, P. and Sheather, S. (1988). On the distribution of a studentized quantile, *JRSS-B* **50**: 381–391.
- Hušková (1994). Some sequential procedures based on regression rank scores, *Journal of Nonparametric Statistics*, **3**: 285–298.
- Koenker, R. and Bassett, G. W. (1978). Regression quantiles, *Econometrica* **46**: 33–50.
- Koenker, R. and Bassett, G. W. (1982). Robust tests for heteroscedasticity based on regression quantiles, *Econometrica* **50**: 43–61.
- Koenker, R. and Bassett, G. W. (1982). Tests of linear hypotheses and  $l_1$  estimation, *Econometrica* **50**: 1577–1584.
- Koenker, R. and D’Orey, V. (1987). Computing Regression Quantiles, *Applied Statistics* **36**: 383–393.
- Koenker, R. and D’Orey, V. (1993). A Remark on Computing Regression Quantiles, *Applied Statistics* **43**: 410–414.
- Koenker, R. and Zhao, Q. (1994).  $L$ -estimation for the linear heteroscedastic models, *Journal of Nonparametric Statistics* **3**: 223–235.
- Koenker, R. (1994). Confidence Intervals for Regression Quantiles, in Mandl, P. and Hušková, M. (eds.) *Asymptotic Statistics*, Springer-Verlag, New York.



- Koenker, R. and Portnoy, S. (1997). The Gaussian Hare and the Laplacian Tortoise: Computability of Squared-error vs. Absolute-error Estimators, with discussion, *Statistical Science*, **12**: 279–300.
- Koenker, R. and Portnoy, S. (2000). *Quantile regression*, manuscript.
- Portnoy, S. (1989). Asymptotic behavior of the number of regression quantile breakpoints, *Journal of Scientific and Statistical Computing* **12**: 867–883.
- Powell, J. L. (1986). Censored regression quantiles, *Journal of Econometrics* **32**: 143–155.
- Powell, J. L. (1989). Estimation of monotonic regression models under quantile restrictions, in Barnett, W.A., Powell, J. L., and Tauchen, G. (eds) *Nonparametric and Semiparametric Methods in Econometrics*, Cambridge University Press, Cambridge.
- Sheather, S. J. and Maritz, J. S. (1983). An estimate of the asymptotic standard error of the sample median, *Australian Journal of Statistics* **25**: 109–122.
- Siddiqui, M. (1960). Distribution of Quantiles from a Bivariate Population, *Journal of Research of the National Bureau of Standards* **64B**: 145–150.
- Welsh, A. H. (1988). Asymptotically efficient estimation of the sparsity function at a point, *Statistics and Probability Letters* **6**: 427–432.