L_1-Statistical Procedures and Related Topics
IMS Lecture Notes – Monograph Series (1997) Volume 31

Computational aspects of censored quantile regression

Bernd Fitzenberger

University of Konstanz, Germany

Abstract: Similar to standard quantile regressions, the censored quantile regression estimate interpolates some data points. This paper discusses the algorithms used in empirical research in light of this interpolation property and compares their performance in a simulation study. The results show that the ranking between algorithms differs depending on the criterion used. The algorithm BRCENS, suggested by the author in the past, performs best in terms of the frequency that the exact censored quantile regression estimates are obtained, it is very competitive in terms of the computation times required and its performance can be noticeably improved, when trying out various starting values. However, BRCENS is not optimal in terms of the root–mean–squared deviation of the coefficient estimates, indicating a high skewness of the distribution of the deviation from the exact estimates. Overall, BRCENS can be recommended for moderate degrees of censoring, whereas all practical algorithms perform quite poorly when a lot of censoring is present.

Key words: Censored quantile regression, algorithms, BRCENS, ILPA, NLRQ.

AMS subject classification: 65U05, 62J05.

1 Introduction

Censored quantile regressions (CQR's), introduced by Powell (1984, 1986), are an attractive approach to the estimation of the censored regression model with fixed known censoring points.[1] First, compared to Tobit maximum likelihood estimation, cf. Amemiya (1985, chapter 10), CQR's provide consistent estimates under far weaker distributional assumptions. Second,

[1] Buchinsky (1997, section 8) and Fitzenberger (1997) provide general guides to CQR's.

CQR's allow us to model the conditional quantiles of the dependent variable as a function of the regressors, cf. Buchinsky (1994), Chamberlain (1994), or Fitzenberger et al. (1995) modelling the conditional wage distribution.

The computation of the CQR estimates involves minimizing a non–differentiable and non–convex distance function. When introducing the censored least absolute deviation (LAD) regression (i.e. the CQR for the special case of the median), Powell (1984) suggested generic optimization routines, which do not take account of the special characteristics of the problem.[2] Paarsch (1984) even resorted to grid search when evaluating the finite sample properties of censored LAD regression. Womersley (1986) analyzed the numerical properties of censored LAD regression for the first time and suggested an algorithm using a finite direct descent method. However, to my knowledge, his code is not available to applied researchers.

More recently, Buchinsky (1994) suggested an Iterative Linear Programming Algorithm (ILPA) involving an iteration of the Barrodale–Roberts–Algorithm (BRA) first developed for standard LAD regression.[3] However, ILPA is not guaranteed to converge and convergence does not guarantee a local minimum of the CQR optimization problem. Building on the characterization of the CQR estimates by the interpolation property presented in the following, Fitzenberger (1994) developed the algorithm BRCENS as an adaptation of the BRA guaranteeing convergence to a local minimum. Koenker and Park (1996) developed a general interior point algorithm for nonlinear quantile regression problems (NLRQ), which they apply to the CQR case. The simulation studies in Fitzenberger (1994, 1997) show that BRCENS performs best in comparison to ILPA and NLRQ in terms of the frequency that the exact global minimum of the CQR optimization problem is obtained. However, the simulation studies show that all algorithms perform quite poorly in the presence of a lot of censoring.

The purpose of this paper is to give an overview on the computational aspects of CQR's and to provide more extensive simulation evidence on the performance of various algorithms currently in use. In the remainder of this section, I present the interpolation property characterizing the CQR estimates. Section 2 describes various algorithms in detail and presents modified versions of the algorithms BRCENS and NLRQ. Section 3 extends the available evidence from simulation studies. Summarizing the simulation results, BRCENS performs quite well in comparison and it can be recommended for moderate degrees of censoring relative to the quantile

[2]Following Powell's suggestion, Horowitz and Neumann (1987) present an empirical application based on the Nelder and Mead (1965) algorithm.

[3]Koenker and d'Orey (1987) provide an extension of the BRA to the general quantile regression case. For the latter, the following discussion refers to this extension.

considered. However, in the presence of a lot of censoring all algorithms perform quite poorly.

1.1 Censored quantile regression and interpolation property

Introducing some notation, for a sample of size N, let the dependent variable be the $N \times 1$ vector, $y = (y_1, ..., y_N)$, the design matrix be the $N \times k$ matrix $X = (x_1, ..., x_N)'$, with $x_i = (x_{i,1}, ..., x_{i,k})$, the $N \times 1$ vector of fixed known observation specific censoring values be $yc = (yc_1, ..., yc_N)$, the $N \times 1$ vector of disturbances be $\epsilon = (\epsilon_1, ..., \epsilon_N)'$ and the $k \times 1$ parameter vector be β.

The following discussion considers a censored regression model with censoring from above. For a given quantile $\theta \in (0, 1)$, the CQR estimation problem[4] is to minimize the piecewise linear distance function given by

$$\widehat{\beta_\theta} \in \underset{\beta}{\text{argmin}} \sum_{i=1}^{N} sgn_\theta(y_i - min[x_i'\beta, yc_i]) \cdot (y_i - min[x_i'\beta, yc_i]) \ , \qquad (1)$$

where the θ weighted sign function is given by

$$sgn_\theta(\epsilon_i) \ \equiv \ \theta I(\epsilon_i > 0) \ - \ (1 - \theta) I(\epsilon_i < 0)$$

and I(.) denotes the indicator function. The expression $x_i'\beta_\theta$ captures the θ–quantile of the underlying uncensored dependent variable conditional on x_i.

Since the CQR distance function (1) is piecewise linear, the CQR minimization problem does not necessarily have a unique solution.[5] Analogous to standard quantile regressions, cf. Koenker and Bassett (1978), the following interpolation property can also be established for CQR's.

Interpolation Property: If the design matrix X has full rank k, then there exists a global minimizer $\widehat{\beta_\theta}$ of the CQR distance function such that $\widehat{\beta_\theta}$ interpolates at least k data points, i.e. there are k observations $\{(y_{i_1}, x_{i_1}), ..., (y_{i_k}, x_{i_k})\}$ with

(IP) $y_{i_l} = x_{i_l}'\widehat{\beta_\theta}$ for $l = 1, ..., k$ and the rank of $(x_{i_1}, ..., x_{i_k})'$ equals k.

[4] Fitzenberger (1997) treats censoring both from above and below and provides the asymptotic distribution of the CQR estimator. The expression "$sgn_\theta(\epsilon_i) \cdot \epsilon_i$" is mostly referred to as "check function" $\rho_\theta(\epsilon_i)$. The notation used here has advantages when studying the asymptotic distribution of the estimator.

[5] Womersley (1986, p. 112) and Fitzenberger (1994) provide a more complete characterization of the set of minimizers.

When evaluating the IP, the following three points deserve attention. First, if the CQR distance function exhibits a unique minimizer, it must satisfy IP. Second, the CQR can interpolate a censoring point where an observation is censored. And third, in contrast to standard quantile regressions, Koenker and Bassett (1978), it is not guaranteed that a share of at most θ [$(1-\theta)$] of the observations lies above [below] the estimated CQR line with an intercept.

The IP is established by analyzing the kinks of the piecewise linear distance function (1) for which the directional derivative proves an important tool. The directional derivative evaluated at β in direction $w \in \Re^k$ is given by

$$H'_\theta(\beta,w) = \sum_{i=1}^{N}[I(x_i'\beta < yc_i)\{-sgn_\theta(y_i - x_i'\beta) - I(x_i'\beta = y_i)sgn_\theta(-x_i'w)\}$$
$$+I(x_i'\beta = yc_i)\{(1-\theta)I(y_i < yc_i, x_i'w < 0) - \theta I(y_i = yc_i, x_i'w < 0)\}] \cdot x_i'w \quad . \tag{2}$$

2 Algorithms

Whereas previous simulation studies relied on grid search to determine the CQR estimates, the interpolation property (IP) discussed above suggests an enumeration algorithm to determine the CQR estimate exactly, i.e. an element out of the set of global minimizers. This algorithm, which I denote by IPOL, consists of an enumeration of the set of all k–tuples of data points with linear independent regressor vectors and the corresponding interpolating regression line. Then the ones minimizing the CQR distance function are in the set of global minimizers. IPOL involves the evaluation of at most $\binom{N}{k}$ k–tuples. In contrast to grid search, IPOL guarantees to find a global minimum exactly and is typically much faster than grid search. The computational advantage of IPOL relative to grid search increases with the required accuracy of the estimates and the number of the regressors k and it decreases with the number of observations N. The algorithms discussed in the following, which are typically much faster than IPOL, will be contrasted in Section 3 with the exact CQR estimates obtained by IPOL.

2.1 BRCENS

The algorithm BRCENS is developed in Fitzenberger (1994) as an adaptation of the standard **B**arrodale–**R**oberts–Algorithm (BRA) for standard Quantile Regressions to the **Cens**ored Quantile Regression case. A standard quantile regression exhibits a linear programming structure. Barrodale and Roberts (1973) notice that the IP allows for a more efficient, condensed simplex approach. Only kinks of the distance function need to be considered for which k (design matrix exhibits full rank) observations are interpolated and for which the rank of the matrix formed by the regressor vectors is equal to k.

Analogous to BRA, the algorithm BRCENS involves two parts. First, the algorithm starts with all coefficients being zero as the set of nonbasic variables (NB).[6] The algorithm proceeds in k steps, where in each step that coefficient from NB changes into a direction for which the directional derivative (2) indicates the strongest decline of the objective. This defines a one–dimensional search direction and the coefficient is changed along this direction until the objective starts increasing again. At this point, according to the expression for the directional derivative in (2), there is at least one data point being interpolated. One of these interpolated data points now replaces the coefficient leaving NB. At the end of the first part, the algorithm has reached a situation where IP is satisfied. The second part of BRCENS considers exchanging one of the interpolated observations in NB with a different observation. With all other data points in the NB remaining interpolated, considering one data point defines a one–dimensional search direction. The algorithm keeps moving into the search direction, for which the directional derivative indicates the largest decline of the objective, until the objective cannot be reduced further. At this new point there exists at least one data point which is added to NB, the set of interpolated data points. The algorithm stops when the directional derivatives for all interpolated data points in NB are non–negative, thus guaranteeing that a local minimum has been achieved.[7]

BRCENS does not guarantee convergence to a global minimum of the CQR objective function. The contribution to the directional derivative becomes zero at data points i, for which the current estimate of the CQR line yields a strictly censored fitted value, i.e. $x_i'\beta > yc_i$ at the current β. Therefore, the simulation study considers a heuristically modified version of BRCENS, denoted by MBRCENS, which uses different starting values. MBRCENS starts with BRCENS. Based on the set of coefficients obtained, new starting values are used which tend to put the current CQR line out of

[6] In the Simplex setup, the set of nonbasic variables comprises those coefficients and residuals (including the corresponding data points) which are currently zero and for which the Simplex tableau provides the linear representation in terms of the variables in the basis. The variables in the basis (coefficients, data points represented by the corresponding residuals) are typically different from zero.

[7] In contrast to the original version of BRCENS, the current version performs the final directional derivative check for all combinations of k interpolated data points with rank k as NB. This change improves the reported performance relative to earlier simulation studies, cf. Fitzenberger (1994, 1997).

the censored region. The estimates from the first step are used as starting values for BRCENS in the second step except for the intercept coefficient being shifted to the 25%–quantile of the estimated residuals. Again the estimated coefficients from the second step are modified such that the intercept is now shifted to the 40%–quantile of the estimated residuals and the slope coefficients are multiplied by 0.8 yielding the starting values for a third round with BRCENS. Among the three available estimates, MBRCENS finally takes the one yielding the lowest value of the CQR objective function.

2.2 Iterative linear programming algorithm

Buchinsky (1994, p. 412) suggests the following Iterative Linear Program ming Algorithm (ILPA), which has also been applied in Honoré and Powell (1993). ILPA consists of successions of standard quantile regressions. Starting with an initial coefficient estimate $\hat{\beta}_0$ and a counter $j = 1$, the following iterative steps are continued until either convergence is achieved or a maximal number of iterations is reached:

Step 1: For the j^{th} iteration, determine the set M_j of observations with $x_i'\hat{\beta}_j < yc_i$. If $j = 1$ or $M_j \neq M_{j-1}$ then continue with Step 2, otherwise terminate and take $\widehat{\beta_\theta} = \hat{\beta}_{j-1}$ as the CQR estimate.

Step 2: Calculate $\hat{\beta}_j$ as the standard quantile regression estimate for the set of observations M_j by means of the BRA. Set $j := j + 1$ and repeat Step 1.

Buchinsky states that ILPA is not guaranteed to converge. He motivates ILPA by the following two claims. First, for an optimal solution $\widehat{\beta_\theta}$ of the CQR problem, the set of observations for which the predicted value lies on or above the censoring point, i.e. $x_i'\widehat{\beta_\theta} \geq yc_i$, could have been excluded from the estimation and one would still obtain the same estimate. And second, when convergence is reached, the coefficient estimate represents a local minimum of the CQR distance function.

In Fitzenberger (1994), I provide counter examples showing that there exist both designs for which the CQR estimate interpolates a censoring point and designs for which convergence of ILPA does not result in a local mi-nimum. The main issue is that the contribution of a single observation to the directional derivative (2) changes when the regression line hits the censoring point depending on whether the observation itself is censored and depending on whether the regression line moves towards or away from the censored region.

However, under the assumption that the exact CQR estimate does not interpolate any censored observation, Buchinsky's rationale is true following from results on standard quantile regressions. Therefore, ILPA has a lot of appeal, since asymptotic consistency of the CQR estimate relies on the fact that population quantiles can be estimated consistently as long as they are uncensored, cf. Powell (1984, 1986).

There exists an alternative version of ILPA, which coincides with ILPA except that in step 1 "<" is replaced by "$\leq$" when defining the set M_j.[8] This allows for the CQR estimates to interpolate censored observations, but again convergence does not guarantee a local minimum, cf. Fitzenberger (1994). Since earlier simulation studies showed that this modification does not lead to an improvement relative to the version of ILPA presented first, it is not considered further in this paper.

2.3 NLRQ

The algorithm NLRQ ("Nonlinear Regression Quantile") developed in Koenker and Park (1996) is a generic interior point algorithm for nonlinear quantile regressions defined by minimizing the distance function $\sum_{i=1}^{N} sgn_\theta[y_i - f_i(x_i,\beta)] \cdot [y_i - f_i(x_i,\beta)]$ with respect to β. The algorithm is built on $f_i(x_i,\beta)$ being differentiable in β almost everywhere. NLRQ considers successions of linearized quantile regression problems and at each succession the algorithm performs two steps. First, it considers the dual problem to obtain a one–dimensional search direction by interior point methods. Second, the search direction from the dual problem is translated into a search direction in the primal nonlinear problem and a conventional one–dimensional line search is performed. After the two steps, the f_i's and their gradients are updated. The algorithm stops when the new iterate fails to improve the objective function. Koenker and Park provide an S–code, cf. Becker et al. (1988), for NLRQ. For the subsequent simulation study, I have translated their code into Fortran. This makes the timing comparison somewhat unfair for NLRQ since I use grid search for the line search in the second step.

Considering the use of NLRQ for CQR's, one should note that the $f_i(.)$–function is not everywhere differentiable in β. At points satisfying the IP there could be observations for which the regression line interpolates a censoring value involving a kink in $f_i(.)$, see the expression for the directional derivative in (2), i.e. $f_i(.)$ is not differentiable here. Therefore, I also consider a modification of NLRQ (denoted as MNLRQ) in the subsequent simulation study, which takes account of the fact that the directional derivative differs depending on the direction taken when the current CQR

[8]In contrast to Buchinsky (1994, p. 412), Buchinsky (1997, section 8.1) refers to this version as the ILPA.

interpolates a censoring point. In such a situation, MNLRQ tries all possible permutations of the contribution of interpolated censoring points to the directional derivative when forming the gradient until a search direction is found along which the CQR objective function can be improved.[9] MNLRQ stops when the CQR objective function cannot be improved further resulting in a local minimum of the CQR optimization problem. It has to be emphasized, that MNLRQ is not constructed to be a serious competitor in terms of computation time. In fact, with a lot of censoring in the data, my current implementation of MNLRQ performs very poorly in this respect, cf. Section 3. The goal here is rather to explore whether NLRQ could be improved by considering more precisely the directional derivative information at censoring points.

3 Simulation results

This section analyzes the performance of the algorithms described in Section 2 by means of a simulation study whose design is similar to Fitzenberger (1994, 1997). Table 1 describes the data generating processes (DGP's), (A)–(H). For each scenario, 1000 random samples of size 100 are drawn. The estimation problem is a censored LAD regression with one regressor and an intercept. A sample is dismissed if the exact CQR estimates (determined by the enumeration algorithm IPOL) are not unique.

DGP	Censoring Values	True Coefficients	Regressor Values
(A)	$yc_i = Const$	$(\beta_1, \beta_2) = (0, 0)$	$x_{i,2} \sim N(0,1)$
(B)	$yc_i = Const$	$(\beta_1, \beta_2) = (0, 0)$	$x_{i,2} = -9.9 + 0.2 \cdot i$
(C)	$yc_i = Const + 0.5$	$(\beta_1, \beta_2) = (0.5, 0.5)$	$x_{i,2} \sim N(0,1)$
(D)	$yc_i = Const + 0.5$	$(\beta_1, \beta_2) = (0.5, 0.5)$	$x_{i,2} = -9.9 + 0.2 \cdot i$
(E)	$yc_i \sim N(Const, 1)$	$(\beta_1, \beta_2) = (0, 0)$	$x_{i,2} \sim N(0,1)$
(F)	$yc_i \sim N(Const, 1)$	$(\beta_1, \beta_2) = (0, 0)$	$x_{i,2} = -9.9 + 0.2 \cdot i$
(G)	$yc_i \sim N(Const + 0.5, 1)$	$(\beta_1, \beta_2) = (0.5, 0.5)$	$x_{i,2} \sim N(0,1)$
(H)	$yc_i \sim N(Const + 0.5, 1)$	$(\beta_1, \beta_2) = (0.5, 0.5)$	$x_{i,2} = -9.9 + 0.2 \cdot i$

a) *Const* denotes some constant taking various values, $N(Const, 1)$ denotes the normal distribution with mean *Const* and variance one, and I(.) denotes the indicator function. The random variables ϵ_i are distributed as i.i.d. $N(0,1)$ and $i = 1, ..., N$

Table 1: Data generating processes (DGP) (A) – (H) used in simulation study for the model $y_i = min(yc_i, \beta_1 + \beta_2 \cdot x_{i,2} + \epsilon_i)^a$.

[9] In cases, when there are too many interpolated censored observations (more than $4 \times k$), each component of the current coefficient vector is perturbed by a very small number. This is done to limit the number of permutations to be explored.

The DGP's differ in four dimensions. First, by whether the coefficients to generate the data are both 0 (A,B,E,F) or both 0.5 (C,D,G,H). Since all algorithms start with both coefficients at 0, it could make a difference whether the starting values are close to the truth. Second, the DGP's differ by whether the censoring points are the same for all observations (A,B,C,D) or differ across observations (E,F,G,H). In the first case, if the CQR line is above the censoring value for a certain regressor value then this is true for all regressor values in a certain neighborhood. In the second case, the CQR line can be censored and uncensored for the same regressor values. This could have an influence on the directional derivative information used locally. Third, the DGP's differ by whether the regressor is a random variable (A,C,E,G) or a fixed sequence of numbers (B,D,F,H). The two scenarios differ by whether all observations exhibit the same a priori distribution. And fourth, the DGP's differ by the degree of censoring depending on $Const = 1, 0.5, 0$. Table 2 shows the average share of censored observations depending on the DGP and the choice of $Const$. $Const = 0$ represents a situation where on average 50% of the observations are censored, i.e. the exact CQR ($\theta = 0.5$) typically reaches the censored region.

DGP	Const = 1.0	Const = 0.5	Const = 0.0
(A)	15.9	30.9	49.7
(B)	15.9	30.9	49.8
(C)	18.7	32.9	49.9
(D)	41.0	46.0	50.9
(E)	24.1	36.4	50.1
(F)	24.1	36.3	50.0
(G)	25.4	37.1	50.0
(H)	41.0	46.0	51.0

Table 2: Average share of censored observations in random samples for various data generating processes (DGP) – In percent.

Convergence: Table 3 presents the absolute frequencies that an algorithm converges. For ILPA, the algorithm is terminated after 20 iterations, if no convergence is achieved. Increasing this number to 100 in some scenarios did not change the results. In such a case, the best coefficient vector during these 20 iterations (in terms of the CQR distance function) is taken as the ILPA estimate.[10] MNLRQ and NLRQ are considered not to have converged

[10] In an earlier simulation study, I used the final coefficient estimate after 20 iterations,

BRCENS, MBRCENS, and NLRQ converged under all DGP's for all 1000 random samples						
DGP	Const = 1.0		Const = 0.5		Const = 0.0	
	ILPA	MLRQ	ILPA	MNLRQ	ILPA	MNLRQ
(A)	999	1000	925	1000	537	707
(B)	1000	1000	962	1000	508	1000
(C)	752	1000	494	1000	298	989
(D)	662	1000	656	1000	642	1000
(E)	750	1000	618	1000	485	1000
(F)	747	1000	641	1000	506	1000
(G)	727	1000	621	1000	526	1000
(H)	726	1000	720	1000	710	1000

Table 3: Absolute frequencies among 1000 random samples that algorithms converged.

after 200 iterations. BRCENS, MBRCENS, and NLRQ converge for all scenarios. Convergence is a serious problem for ILPA, especially with a high degree of censoring ($Const = 0$), with bad starting values (C,D) or with random censoring points. When ILPA does not converge, it typically oscillates between two or three coefficient vectors.[11] Additional results (not reported here) indicate that along the iterations ILPA reaches a local minimum as the best coefficient estimate in almost all cases for $Const = 1, 0.5$ and in at least 76% of all cases for $Const = 0$. MNLRQ converges in all cases with low or moderate censoring ($Const = 1, 0.5$) but it exhibits some convergence problems in the presence of a lot of censoring ($Const = 0$, DGP (A) and (C)). Overall, lack of convergence is a very serious drawback for the application of ILPA.

Optimality: Table 4 is concerned with the frequencies that the global minimum of the CQR distance function is achieved where the latter (the exact CQR estimate) is obtained by the enumeration algorithm IPOL. An algorithm is assumed to have achieved the optimum, if the value of the objective at the solution is within a tolerance of 10^{-7} to the value of the exact

cf. Fitzenberger (1994).

[11] If ILPA does not converge, it must oscillate, since a finite sample allows only for a finite number of subsamples which a standard quantile regression can be based upon. Oscillation arises, since an observation, for which the current standard quantile regression implies a censored fitted value, still contributes to the distance function. In the next iteration, this observation is excluded from the sample, which can result in a new estimate for which the fitted value at the aforementioned observation is now uncensored.

CQR estimates. The results are very favorable for BRCENS in comparison. The relative performance of BRCENS is better, when the degree of censoring is higher and when there are more observation specific censoring points. However, for large degrees of censoring, all algorithms (including BRCENS) perform very poorly. Given that BRCENS, MBRCENS, and MNLRQ only guarantee convergence to a local minimum, this poor performance is to be expected, since the CQR distance function is already highly nonconvex with moderate censoring. Unfortunately, a local minimum does not guarantee a solution close to the global minimum, see also the following results on the properties of the coefficient estimates. The modified version MBRCENS yields a substantial improvement compared to BRCENS. ILPA performs better than NLRQ. Again the modified version, MNLRQ improves upon NLRQ and performs better than ILPA in many cases. All algorithms perform quite satisfactorily with moderate degrees of censoring, a common censoring point, and good starting values, DGP (A) and (B) and $Const = 1$.

Properties of Coefficient Estimates: The quality of the coefficient estimates obtained by various algorithms is an important issue, which has been mostly neglected in my previous simulation studies. Therefore, Tables 5 and 6 also provide results on the root–mean–squared deviation and the 90%–percentile of the absolute deviation of the respective estimates from the exact CQR estimates for moderate degrees of censoring ($Const = 0.5$).[12] According to the root–mean–squared deviations criterion, BRCENS and ILPA exhibit almost the same performance and NLRQ performs noticeably better. This is in contrast to the optimality results presented before. Considering results for $Const = 0$ (not reported here) reverses the relative performance of BRCENS and NLRQ (BRCENS also outperforms ILPA in this case).[13] The modified algorithm MBRCENS improves upon BRCENS, whereas there is no clear ranking between MNLRQ and NLRQ. MBRCENS performs slightly worse than MNLRQ and NLRQ. Turning to the 90%-percentiles, the numbers are considerably smaller than for the root–mean–squared–deviations, however, the relative performance between the algorithms is only slightly changed (MBRCENS performs better than NLRQ/MNLRQ in a considerable number of cases). Overall, these findings

[12]Further results for $Const = 1, 0.5$ and results on the median and the 75%–percentile are available from the author upon request.

[13]At this point, it remains a topic for further research to investigate why NLRQ proves less outlier sensitive than BRCENS or ILPA for moderate censoring. It seems to pay off that interior point methods do not move too close to the contraint set (in linear programming terminology) in an early stage of the iteration process. Some local minima on the contraint set can actually imply quite extreme coefficient values.

indicate that for all algorithms the distribution of the absolute deviation from the exact CQR estimates is very much skewed to the right. This effect is strongest for BRCENS and ILPA.

DGP	ILPA	BRCENS	MBRCENS	NLRQ	MNLRQ
			Const = 1.0		
(A)	993	995	995	829	983
(B)	1000	1000	1000	863	995
(C)	655	864	908	445	705
(D)	526	843	938	332	633
(E)	655	866	904	387	499
(F)	642	874	932	400	487
(G)	630	855	913	229	454
(H)	628	874	958	426	646
			Const = 0.5		
(A)	873	936	942	730	870
(B)	924	942	958	782	934
(C)	353	687	745	219	467
(D)	526	850	932	361	586
(E)	501	818	861	283	397
(F)	511	802	876	310	393
(G)	511	804	884	222	433
(H)	599	837	936	451	639
			Const = 0.0		
(A)	90	330	379	2	7
(B)	81	174	408	53	149
(C)	159	474	596	100	261
(D)	487	806	911	358	592
(E)	332	702	774	197	293
(F)	374	720	845	189	296
(G)	391	706	809	150	339
(H)	603	826	942	465	646

Table 4: Absolute frequencies among 1000 random samples that algorithms achieved global optimum of CQR distance function.

Timing: Table 7 provides the relative average CPU time requirements of the different algorithms depending on the degree of censoring in the data (*Const*). Timing is of particular importance when bootstrapping the CQR estimates. The results show that BRCENS exhibits the lowest time requirements. ILPA and MBRCENS are next with no clear ranking between the two. In comparison, NLRQ and MNLRQ exhibit a much larger time

DGP	ILPA	BRCENS	MBRCENS	NLRQ	MNLRQ
		Const = 0.5 – Estimates for β_1			
(A)	.565	.566	.566	.475	.440
(B)	6.409	6.409	6.409	6.408	6.408
(C)	.181	.179	.175	.141	.146
(D)	.113	.108	.075	.075	.078
(E)	.045	.036	.033	.029	.027
(F)	.048	.043	.039	.034	.038
(G)	.051	.045	.037	.037	.030
(H)	.121	.114	.058	.073	.079
		Const = 0.5 – Estimates for β_2			
(A)	.309	.309	.309	.260	.247
(B)	.662	.662	.662	.662	.662
(C)	.162	.156	.155	.110	.122
(D)	.018	.018	.012	.012	.013
(E)	.051	.048	.046	.037	.036
(F)	.007	.007	.006	.005	.006
(G)	.050	.049	.041	.037	.030
(H)	.019	.018	.009	.013	.013

Table 5: Root–mean–squared deviation of coefficient estimates from exact CQR estimates.

requirement. For a low degree of censoring, BRCENS is about 60 times faster than the exact determination of the CQR estimates by means of IPOL. This advantage is reduced to a factor of 10 when a lot of censoring is present. In comparison, the incremental time requirement for MBRCENS is fairly small, whereas NLRQ and MNLRQ are considerably more expensive. For the case with a lot of censoring, MNLRQ even uses more time than IPOL. However, for fairness sake, it has to mentioned that my Fortran implementation of NLRQ and MNLRQ is likely to be somewhat inefficient. Given the poor performance of all other algorithms in the presence of a lot of censoring, IPOL appears a viable alternative in such a situation.

Summary and Recommendations: Summarizing the simulation results, BRCENS performs quite well in comparison. It outperforms ILPA and NLRQ with respect to the frequencies that the exact CQR estimates are reached. BRCENS is very competitive in terms of the computation times involved and when trying out different starting values its performance can be improved considerably at a fairly small computational cost. All algo-

DGP	ILPA	BRCENS	MBRCENS	NLRQ	MNLRQ
		Const = 0.5 – Estimates for β_1			
(A)	.000	.000	.000	.013	.000
(B)	.000	.000	.000	.011	.000
(C)	.276	.256	.243	.054	.030
(D)	.239	.221	.000	.040	.027
(E)	.108	.072	.046	.049	.051
(F)	.097	.084	.022	.057	.051
(G)	.099	.091	.030	.080	.057
(H)	.267	.213	.000	.050	.029
		Const = 0.5 – Estimates for β_2			
(A)	.000	.000	.000	.013	.000
(B)	.000	.000	.000	.002	.000
(C)	.324	.267	.259	.073	.045
(D)	.039	.034	.000	.014	.008
(E)	.112	.084	.050	.051	.050
(F)	.014	.017	.007	.010	.010
(G)	.096	.112	.032	.091	.055
(H)	.043	.036	.000	.012	.007

Table 6: 90%–quantile of absolute deviation of coefficient estimates from exact CQR estimates.

rithms perform the worse, the higher the degree of censoring. Based on the simulation results reported here and recognizing that BRCENS guarantees convergence to a local optimum, its application can be highly recommended in situations with low or moderate degrees of censoring (relative to the quantile θ being estimated). For high degrees of censoring, one might want to determine the exact CQR estimates by means of IPOL. The results for NLRQ show that its performance can be improved when using more precise directional derivative information. However, this could result in considerably higher computation cost (or lack of convergence, when only a small number of iterations is allowed for).

Acknowledgements

I am grateful to Roger Koenker, Steve Portnoy, and an anonymous referee for helpful comments. All errors are my sole responsibility.

Const	IPOL	ILPA	BRCENS	MBRCENS	NLRQ	MNLRQ
1.0	66.24	1.77	1.00	1.69	24.50	21.90
0.5	64.19	2.12	2.05	2.69	27.25	24.19
0.0	61.96	6.86	6.41	7.27	25.55	700.87

a) The reported numbers are the ratios of average computation times across DGP's (A)–(H) for different degrees of censoring ($Const = 1, 0.5, 0$), cf. table 2, relative to BRCENS, $Const = 1$. The time results are obtained with the UNIX 'time' command on an IBM RS 6000 workstation, based on a Fortran implementation of the various algorithms.

Table 7: Average computation times[a].

References

[1] Amemiya, T. (1985). *Advanced Econometrics.* Cambridge, Massachusetts: Harvard University Press.

[2] Barrodale, I. and F.D.K. Roberts (1973). An improved algorithm for discrete l_1 linear approximation. *SIAM J. Numer. Anal.* **10** 839–848.

[3] Becker, R.A., Chambers, J.M., and A.R. Wilks (1988). *The New S Language.* Wadsworth & Brooks/Cole, Pacific Grove.

[4] Buchinsky, M. (1994). Changes in the U.S. wage structure 1963–1987: application of quantile regression. *Econometrica* **62** 405–458.

[5] Buchinsky, M. (1997). Recent advances in quantile regression models: a practical guideline for empirical research.. *Journal of Human Resources.* To appear.

[6] Chamberlain, G. (1994). Quantile regression, censoring, and the structure of wages. In *Advances in Econometrics: Sixth World Congress,* Ed. C. Sims. Econometric Society Monograph, Volume 1.

[7] Fitzenberger, B. (1994). A note on estimating censored quantile regressions. Center for International Labor Economics, Discussion Paper Nr. 14, University of Konstanz.

[8] Fitzenberger, B. (1997). A guide to censored quantile regressions. In *Handbook of Statistics,* Volume 15: Robust Inference, Eds. G.S. Maddala, and C.R. Rao. North–Holland. To appear.

[9] Fitzenberger, B., R. Hujer, T.E. MaCurdy, and R. Schnabel (1995). The dynamic structure of wages in Germany 1976–1984, a cohort analysis. Center for International Labor Economics, Discussion Paper Nr. 22, University of Konstanz.

[10] Honoré, B.E. and J.L. Powell (1993). Quantile regression under random censoring. Unpublished Manuscript, Princeton University and University of California at Berkeley.

[11] Horowitz, J. and G. Neumann (1987). Semiparametric estimation of

employment duration models. *Econometric Reviews* **6** 5–40.

[12] Koenker, R. and G. Bassett (1978). Regression quantiles. *Econometrica* **46** 33–50.

[13] Koenker, R. and V. D'Orey (1987). Algorithm AS 229. Computing regression quantiles. Statistical algorithms, 383–393.

[14] Koenker, R. and B.J. Park (1996). An interior point algorithm for nonlinear quantile regression. *Journal of Econometrics* **71** 265–283.

[15] Nelder, J.A. and R. Mead (1965). A simplex method for function minimization. *Computer Journal* **7** 308–313.

[16] Paarsch, H. (1984). A Monte Carlo comparison of estimators for censored regression modells. *Journal of Econometrics* **24** 197–213.

[17] Powell, J.L. (1984). Least absolute deviations estimation for the censored regression model. *Journal of Econometrics* **25** 303–325.

[18] Powell, J.L. (1986). Censored regression quantiles. *Journal of Econometrics* **32** 143–155.

[19] Womersley, R.S. (1986). Censored discrete l_1 approximation. *SIAM J. Scient. Statist. Comput.* **7** 105–122.