An efficient algorithm for structured sparse quantile regression

Vahid Nassiri · Ignace Loris

Abstract An efficient algorithm is derived for solving the quantile regression problem combined with a group sparsity promoting penalty. The group sparsity of the regression parameters is achieved by using a $\ell_{1,\infty}$-norm penalty (or constraint) on the regression parameters. The algorithm is efficient in the sense that it obtains the regression parameters for a wide range of penalty parameters, thus enabling easy application of a model selection criteria afterwards. A Matlab implementation of the proposed algorithm is provided and some applications of the methods are studied.

Keywords structured sparsity · variable selection · convex optimization

1 Introduction

The quantile regression model of Koenker and Bassett (1978) is an important alternative to the ordinary least squares regression. Such a model allows for studying the effect of explanatory variables on the entire conditional distribution of the response variable, and not only on its center. Whether using quantile regression or least squares regression, parsimony is often seen as a key property of a good model. Variable selection techniques have therefore attracted a great deal of attention in recent statistical literature (Claeskens and Hjort 2008). In this paper we discuss an effective numerical algorithm for solving the minimization problem that arises in quantile regression under the assumption of group sparsity of the regression parameters as introduced by Zou and Yuan (2008).

Vahid Nassiri
Department of Mathematics, Vrije Universiteit Brussel, Belgium. E-mail: vnas-siri@vub.ac.be

Ignace Loris
Department of Mathematics, Université Libre de Bruxelles, Belgium. E-mail: iglo-ris@ulb.ac.be
Consider a typical linear regression model:

\[ y = X\beta + \epsilon, \]  

where \( y \) is the response variable and \( X \) is a matrix containing the explanatory variable as its columns. The vector \( \beta \) contains the model parameters, and \( \epsilon \) is the model error. In the maximum likelihood context (Lehmann and Casella, 1998), the estimated parameter vector \( \hat{\beta} \) can be formulated as the following minimization problem:

\[ \hat{\beta} = \arg \min_{\beta} f(y - X\beta), \]  

where the loss \( f(y - X\beta) \) is defined as \(-\log L(\beta|y, X)\) where \( L \) is the likelihood function of the observed sample considering the conditional probability distribution of \( y|X \). The difference between least squares regression and quantile regression lies in the choice of loss function \( f \). The former uses a sum of squares as loss function, while the latter uses the check function of Koenker and Bassett (1978) which is akin to a sum of absolute values (see formulas (5) and (6) for an explicit expression).

An often used approach for imposing desirable properties on \( \hat{\beta} \) consist of penalizing the loss function with a penalty \( g(\beta) \). Such a penalized model therefore depends on the solution of a minimization problem of type:

\[ \hat{\beta}(\lambda) = \arg \min_{\beta} f(y - X\beta) + \lambda g(\beta). \]  

The penalty parameter \( \lambda \geq 0 \) fine-tunes the relative importance of the loss and the penalty in the cost function. Among the many penalty functions that have been proposed, one finds the \( \ell_2 \)-norm squared, \( g(\beta) = \sum_j \beta_j^2 \), for ridge regression of Hoel and Kennard (1970), and the \( \ell_1 \)-norm, \( g(\beta) = \sum_j |\beta_j| \), for Lasso regression of Tibshirani (1996). While ridge regression leads to a non-sparse solution, Lasso regression estimates a sparse parameter vector \( \hat{\beta} \) (i.e. with many parameters exactly zero), thus giving rise to variable selection.

As Tibshirani (1996) discussed, a penalized model also has a Bayesian interpretation and the minimizer \( \hat{\beta} \) in (3) can be regarded as a maximum a posteriori (MAP) estimator of \( \beta \). Therefore, considering a penalized model, one can find other estimates besides the MAP one. In the Lasso case, Park and Casella (2008) discussed the general Bayes estimate using Markov chain Monte Carlo (MCMC) algorithms. In the case of quantile regression some authors have considered the Bayes estimates (Alhamzawi et al. 2012; Alhamzawi and Yu 2012; Yu et al. 2013). In this paper we will focus on the practical computation of the MAP estimate for a particular penalization of the quantile regression loss.

Yuan and Lin (2006) have introduced a penalty for group Lasso regression. It takes the form of a mixed \( \ell_1 - \ell_\infty \)-norm (or \( \ell_{1,\infty} \)-norm for short): \( g(\beta) = ||\beta||_{1,\infty} \) (see expression (8) for its precise definition). As explained in detail in Yuan and Lin (2006) and in Section 2, the \( \ell_{1,\infty} \)-norm makes it possible to select groups of variables instead of individual variables, so a form of
structured sparsity can be imposed on the estimated parameter vector (examples where explanatory variables exhibit an a priori groups structure are given in Section 5).

Zou and Yuan (2008) have introduced group sparse quantile regression through an application. However, the numerical algorithm used by these authors, solves the relevant version of equation (3) for just a single value of the penalty parameter $\lambda$. For model selection, selecting an appropriate value of $\lambda$ remains an important issue. In order to apply a model selection criterion, such as the Bayesian information criterion of Schwarz (1978), it is desirable to have an algorithm that produces the minimizer $\hat{\beta}(\lambda)$ for a wide range of values of $\lambda$. The derivation and implementation of such an algorithm is the subject of this paper.

Efron et al. (2004); Osborne et al. (2000) have introduced such an efficient algorithm for the $\ell_1$-norm penalized least squares regression problem. Li and Zhu (2008) have proposed a similar algorithm for $\ell_1$-norm penalized quantile regression problem (see also Fuchs (2009)). Yuan and Lin (2006) have discussed this type of algorithm for the group sparse least squares regression problem. The main contribution of this article is the derivation of such an efficient algorithm for solving the minimization problem (3) for group sparse quantile regression, i.e. for the $\ell_{1,\infty}$ norm penalty combined with the quantile regression loss function. The implementation of such an algorithm is not straightforward. The second contribution of this paper is a Matlab implementation of this algorithm, available on the authors' web page Loris and Nassiri (2013). This algorithm applies only to the case of non-overlapping groups.

Quantile regression problems in which the design matrix $X$ is sparse have been discussed by Koenker and Ng (2005). This is completely different from our framework. We do not assume any sparsity of the design matrix $X$, but we are interested in the case where the model parameters $\beta$ exhibited sparsity. Quantile regression in high-dimensional sparse models was also studied by Belloni and Chernozhukov (2011).

In order to illustrate the application of the proposed optimization algorithm, we also briefly study two main examples: the analysis of a data set pertaining to ‘low birth weight’ and the use of a simultaneous variable selection approach.

Finally, we also compare the proposed algorithm with a general-purpose convex optimization toolbox (Grant and Boyd 2012) and with a simple iterative algorithm.

2 Problem statement

In this paper, $X_{n\times m} = (X_1, \ldots, X_m)$ is the design matrix containing the explanatory variables. The linear model which we study is:

$$y = X\beta + \epsilon.$$  (4)
The loss function which quantile regression (Koenker and Bassett 1978) tries to minimize in the objective function:

$$
\sum_{i=1}^{n} g_{\tau} ((y - X\beta)_i),
$$

is the pinball loss function $g_{\tau}$ defined as:

$$
g_{\tau}(t) = \begin{cases} 
2\tau t & \text{if } t \geq 0 \\
-2(1 - \tau) t & \text{if } t \leq 0,
\end{cases}
$$

for $t \in \mathbb{R}$ and $0 < \tau < 1$. For $\tau = 1/2$, one recovers $g_{\tau}(t) = |t|$. The estimator resulting from this loss function is more robust to outliers than the usual quadratic one: the minimizer of the least squares loss is the mean, while the minimizer of the least absolute deviations loss (expression (5) with $\tau = 1/2$) is the median. The optimization problem encountered in quantile regression thus is:

$$
\hat{\beta}_{QR} = \arg \min_{\beta} \sum_{i=1}^{n} g_{\tau} ((y - X\beta)_i).
$$

Koenker and D’Orey (1987) proposed an algorithm to solve this problem.

The so-called $\ell_1$-norm $||\beta||_1 = \sum_{j=1}^{m} |\beta_j|$ is well known to promote sparsity (Tibshirani 1996) when used as a penalty (added to a loss function) or as a constraint (imposed on the minimizer of a loss function). Two major examples are the Lasso of Tibshirani (1996) (which combines a quadratic loss with an $\ell_1$-norm penalty or constraint) and sparse quantile regression (Li and Zhu 2008; Fuchs 2009) which combines the loss (5) with an $\ell_1$-norm penalty.

The advantage of using the $\ell_1$-norm of $\beta$ as a penalty instead of the number of nonzero $\beta_j$ is computational: the former leads to a convex minimization problem while the latter gives rise to a combinatorial minimization problem.

In this paper our goal is to study a numerical algorithm that can be used for quantile regression in combination with a group sparsity constraint. Let the variables $\beta_1, \ldots, \beta_m$ be divided into $g$ non-overlapping groups ($g \leq m$) and let $G_1, \ldots, G_g$ represent the indices in each group. The mixed norm $||\beta||_{1,\infty}$ is defined as

$$
||\beta||_{1,\infty} = \sum_{k=1}^{g} ||\beta_{G_k}||_{\infty},
$$

where $\beta_{G_k}$ are the components of $\beta$ in group $k$, and $||\beta_{G_k}||_{\infty} = \max_{j \in G_k} |\beta_j|$. Within a group, the mixed norm behaves as a max-norm and between groups it behaves as an $\ell_1$-norm (sum of max-norms). Therefore, within a given group, when a single member is active (nonzero), the other members of that group are allowed to reach the same magnitude without penalty. Indeed, the value of $\max_{j \in G_k} |\beta_j|$ does not increase as long as the largest of the $|\beta_j|$ (with $j \in G_k$) does not increase, regardless of the actual size of the smaller ones. Furthermore, as a result of the $\ell_1$-norm behavior between groups, it can be used to select a limited number of active groups.
The structured sparse $\hat{\beta}$ for the quantile regression loss function in (5) is therefore defined as the minimizer:

$$
\hat{\beta} = \arg \min_{\beta} \sum_{i=1}^{n} \varphi_{\tau}((y - X\beta)_i) + \lambda \|\beta\|_{1,\infty}
$$

(9)

where $\lambda \geq 0$ is a penalty parameter or as the minimizer:

$$
\hat{\beta} = \arg \min_{\|\beta\|_{1,\infty} \leq R} \sum_{i=1}^{n} \varphi_{\tau}((y - X\beta)_i),
$$

(10)

where $R$ is a nonnegative parameter. By suitably choosing the parameters $\lambda$ and $R$, the minimizers of equation (9) and (10) are identical. We therefore use the same symbol $\hat{\beta}$, both for the minimizer of the penalized problem (9) and for the minimizer of constrained problem (10). We do not always explicitly indicate their dependence on $\lambda$ or $R$.

It is worth mentioning that Koenker et al. (1994) have studied the following minimization problem

$$
\min_{g \in \mathcal{G}} \sum_{i} \varphi_{\tau}(y_i - g(x_i)) + \lambda \left( \int_{0}^{1} |g''(x)|^p \, dx \right)^{1/p},
$$

(1)

(9)

(where $\mathcal{G}$ should be chosen appropriately) and have characterized the solution of this problem as splines for the particular choices $p = 1$ and $p = \infty$. For these choices, the problem can be solved using standard $\ell_1$-type linear programming. An algorithm to find the solution of this problem is also proposed in Ng (1996).

The mixed norm penalty $\|\beta\|_{1,\infty}$ was studied in the framework of least squares loss functions by Yuan and Lin (2006). A special case of that least squares problem was already introduced by Turlach et al. (2005). In contrast to mixed norm penalized least squares, expressions (9) and (10) combine the structure-imposing properties of the mixed norm with the robustness properties of quantile regression. This combination was introduced by Zou and Yuan (2008). However, the numerical optimization algorithm used by Zou and Yuan (2008) solves the problems (9) and (10) for just a single value of the penalty parameter $\lambda$ or a single value of the parameter $R$. The aim of this paper is to present an efficient algorithm specifically tailored to the solution of the minimization problems (9) and (10), for various values of $\lambda$ and $R$.

Standard general purpose convex optimization packages like e.g. (Grant and Boyd 2012) can solve instances of problems (9) and (10), i.e. they can solve such problem for a given value of $R$ or $\lambda$. In the context of regression problems, those values of $R$ or $\lambda$ are not given a priori and must be determined as well. This involves comparing several instances of $\hat{\beta}(R)$ and selecting one of them using a model selection criterion such as e.g. the BIC (Schwarz 1978). This is where the advantages of the proposed algorithm will be apparent: Instead of solving the problem (10) for a single value of $R$, it will compute the solution incrementally for a wide range of values of $R$ starting from $R = 0$ (with trivial solution $\hat{\beta} = 0$) up to some $R_{\text{max}}$. This can be done exactly (up
to numerical round-off) if one exploits the piece-wise linear nature of $\hat{\beta}$ as a function of $R$ (see Section 3).

General purpose convex optimization packages do not recognize or exploit the piece-wise linear behavior of $\hat{\beta}(R)$, and can only be used for computing several instances of $\hat{\beta}(R)$ separately. It thus turns out that the exact algorithm is more efficient than such algorithms or than purpose-designed iterative algorithms (see Section 5.3 for numerical simulations).

Non-iterative exact algorithms, such as the one proposed below for the pinball loss combined with the $\ell_{1,\infty}$-norm penalty, have already been introduced and applied for problems combining other loss functions and penalties (Efron et al. 2004; Osborne et al. 2000; Li and Zhu 2008; Fuchs 2009; Yuan and Lin 2006). The difficulty is that a completely new algorithm must be constructed for each combination of loss and penalty, based on their particular properties. Our main contribution is therefore the implementation and publication (in the spirit of reproducible computational research of Donoho (2010)) of a non-iterative exact algorithm for solving problems (9) and (10) for a range of values of $R$ and $\lambda$ (see Section 4).

3 Solution of the minimization problem

Due to the presence of the non-smooth functions $\varphi_\tau$ and $\|\beta\|_{1,\infty}$, the optimization problems (9) and (10) are not differentiable. They are however convex minimization problems for which a general theory exists. We refer to e.g. Rockafellar (1970) for an introduction to convex analysis. In particular, for any convex function $f$, the symbol $\partial f$ denotes the subdifferential of $f$.

We express the condition for minimizing the cost function of expression (9) using subdifferentials instead of usual derivatives. Necessary and sufficient conditions for optimality of the optimization problem (9) are found by expressing that $0$ belongs to the subdifferential of the functional (9). Therefore, we will have that $\beta$ is the minimizer of (9) if and only if there exists a vector $w \in \partial \sum_{i=1}^n \varphi_\tau(r_i)$ and a vector $u \in \partial \|\beta\|_{1,\infty}$ such that:

$$-X^T w + \lambda u = 0,$$

where $r = y - X\beta$. The equations that we need to solve are:

$$\begin{cases} 
-X^T w + \lambda u = 0 \\
w_i \in \partial \varphi_\tau(r_i) \\
u \in \partial \|\beta\|_{1,\infty} \\
r = y - X\beta.
\end{cases}$$

(12)

Solving the minimization problems (9) and (10) therefore requires detailed knowledge of the subdifferentials of $\varphi_\tau$ and of $\|\beta\|_{1,\infty}$. 
An efficient algorithm for structured sparse quantile regression

In case of the function \( g_r \) the subdifferential is equal to:

\[
\partial g_r(r_i) = \begin{cases} 
2\tau & \text{if } r_i > 0, \\
[-2(1 - \tau), 2\tau] & \text{if } r_i = 0, \\
-2(1 - \tau) & \text{if } r_i < 0.
\end{cases}
\]  

(13)

It is important to remark that, conversely, the knowledge of the value of the subgradient \( w_i \in \partial g_r(r_i) \) also gives a certain knowledge on \( r_i \). In particular, if \( w_i = 2\tau \), then \( r_i \) must be non-negative, if \( w_i = -2(1 - \tau) \), then \( r_i \) must be non-positive and if \( w_i \) belongs to the open interval \((-2(1 - \tau), 2\tau)\), then \( r_i \) must be zero. This will be important when describing the algorithm that solves the equations (12).

The subdifferential \( \partial \|\beta\|_{1,\infty} \) for the penalty part \( \|\beta\|_{1,\infty} \) is more difficult. As the groups are disjoint, each term in the sum (8) can be handled separately.

For simplicity, let us first consider a group with just two members (e.g. \( \beta_1, \beta_2 \)) and derive the subdifferential of \( \max\{\|\beta_1\|, \|\beta_2\|\} \). The subdifferential of the max function is the convex hull of the union of the subdifferentials of the ‘maximal’ arguments (see e.g. Boyd and Vandenberghe (2004)). We distinguish three cases.

If \( \beta_1 = \beta_2 = 0 \), then the maximum is found by both \( \|\beta_1\| \) and \( \|\beta_2\| \): \( \max\{\|\beta_1\|, \|\beta_2\|\} = \|\beta_1\| = \|\beta_2\| \). As \( \partial \|\beta_1\| = [-1, 1] \) (at \( \beta_1 = 0 \)) and \( \partial \|\beta_2\| = 0 \), the subdifferential of \( \|\beta_1\| \) (with respect to \( \beta_1 \) and \( \beta_2 \)) is the line segment connecting \((-1, 0)\) with \((1, 0)\). Similarly, the subdifferential of \( \|\beta_2\| \) (with respect to \( \beta_1 \) and \( \beta_2 \)) is the line segment connecting \((0, -1)\) to \((0, 1)\). The subdifferential of \( \max\{\|\beta_1\|, \|\beta_2\|\} \) at \( \beta_1 = \beta_2 = 0 \) is the convex hull of these two line segments, i.e. the \( \ell_1 \)-ball of radius 1 (see Figure 1, left).

If \( \|\beta_1\| = \|\beta_2\| > 0 \), the subdifferential of \( \|\beta_1\| \) (with respect to \( \beta_1 \) and \( \beta_2 \)) is \( (\text{sign}(\beta_1), 0) \), and the subdifferential of \( \|\beta_2\| \) (with respect to \( \beta_1 \) and \( \beta_2 \)) is \( (0, \text{sign}(\beta_2)) \). The subdifferential of \( \max\{\|\beta_1\|, \|\beta_2\|\} \) is therefore given by the line segment connecting \((\text{sign}(\beta_1), 0)\) with \((0, \text{sign}(\beta_2))\) (see Figure 1, center).

If \( \|\beta_1\| > \|\beta_2\| \), the subdifferential of \( \|\beta_1\| \) is \( (\text{sign}(\beta_1), 0) \). The subdifferential of \( \max\{\|\beta_1\|, \|\beta_2\|\} \) is therefore the single point \( (\text{sign}(\beta_1), 0) \) (see Figure 1, right). The case \( \|\beta_1\| < \|\beta_2\| \) is handled similarly.

Consider now group \( k \) with \( \|\beta_{G_k}\|_{\infty} = \max_{j \in G_k} \|\beta_j\| \). When all components of \( \beta \) in group \( G_k \) are zero, we have that \( \partial \|\beta_{G_k}\|_{\infty} \) equals the \( \ell_1 \)-ball of radius
When all components of $\beta$ in group $G_k$ are equal in absolute value (and nonzero), the subdifferential $u_{G_k}$ of $\max_{j \in G_k} |\beta_j|$ is characterized by $\|u_{G_k}\|_1 = 1$ and $\text{sign}(u_j) \text{sign}(\beta_j) \geq 0$ for all $j \in G_k$. In the remaining case, the subdifferential of $u_{G_k}$ of $\max_{j \in G_k} |\beta_j|$ is characterized by $\|u_{G_k}\|_1 = 1$ and $\text{sign}(u_j) \text{sign}(\beta_j) \geq 0$ for $j \in G_k$ such that $|\beta_j| < \|\beta_{G_k}\|_\infty$ (sub-maximal $\beta_j$).

Here too, knowledge of $u_{G_k} \in \partial \|\beta_{G_k}\|_\infty$ yields partial information on $\beta_{G_k}$. E.g. if $\|u_{G_k}\|_1 < 1$ then all $\beta_j$ in this group are zero; if $\|u_{G_k}\|_1 = 1$ and all $u_j \neq 0$ then all $\beta_j$ in that group have the same absolute value. If $u_j = 0$ then $\beta_j$ may not be maximal in its group. This type of interplay between $\beta$, $u$, $w$ and $r$ will be used frequently in the algorithm which we describe in Section 4.

The optimality conditions (12) are not analytically solvable for $\beta$. However, the problems (9) and (10) fall into the class of problems described by Rosset and Zhu (2007) which allow for a piece-wise linear solution path. This means that the minimizers of (9) and (10) are piecewise linear functions in terms of $\lambda$ or $R$. Moreover the nodes that determine these piecewise linear functions can be calculated analytically (i.e. using linear algebra), provided one starts the calculation from $R = 0$ (which corresponds to $\beta = 0$ and $\lambda$ large) and proceeds carefully for increasing values of $R$ (decreasing values of $\lambda$). As explained in the next section, a new node of the minimizer $\hat{\beta}$ (as a function of $R$) appears e.g. when a new group becomes active (becomes nonzero) or when an equation among the $(y - X \beta)_i = 0$ is satisfied. There are several more events like this possible and they are explained in more detail in the next section.

4 Efficient non-iterative algorithm

The non-iterative algorithm we propose for finding the minimizers of (9) and (10) is described in this section. It is similar to the LARS algorithm of Efron et al. (2004) for the Lasso and the algorithm of Li and Zhu (2008) and Fuchs (2009) for sparse quantile regression (where the number of groups equals the number of explanatory variables). In our case, we are not restricted to singleton groups; however, only non-overlapping groups are allowed.

4.1 Algorithm

Following Fuchs (2009), we set $u = u^{(0)} + u^{(1)}/\lambda$ and $w = w^{(0)} + \lambda v^{(1)}$, where $u^{(0)}$, $u^{(1)}$, $w^{(0)}$ and $w^{(1)}$ do not depend on $\lambda$. In this way equation (11) is equivalent to:

$$-X^T w^{(0)} + u^{(1)} = 0 \quad \text{and} \quad -X^T u^{(1)} + w^{(0)} = 0. \quad (14)$$

As the minimizer $\hat{\beta}$ of the problem (10) is a piecewise linear function of $R = \|\hat{\beta}\|_{1,\infty}$, it is characterized by a set of (interpolation) nodes. If the value of $\hat{\beta}$ is known in these nodes, then $\hat{\beta}$ can also be found for other (intermediate) values
of $R$ by linear interpolation. For two successive nodes located at $R = R_1$ and $R = R_2$ the interpolation formula is:

$$\beta(R) = \frac{(R_2 - R)}{R_2 - R_1}\beta(R_1) + \frac{(R - R_1)}{R_2 - R_1}\beta(R_2) \quad R_1 \leq R \leq R_2.$$  

(15)

The proposed algorithm starts from $\hat{\beta} = 0$ for $\lambda$ sufficiently large ($\lambda \geq \lambda_{\text{max}}$). As $\hat{\beta} = 0$, one finds the value of $r = y - X\hat{\beta} = y$ and of $w$: $w^{(0)}_i = 2\tau$ if $r_i > 0$, $w^{(0)}_i = -2(1 - \tau)$ if $r_i < 0$ (assuming that all coefficients of $r$ at this stage are nonzero) and $u^{(1)} = 0$. Using equations (14), one then calculates $u = u^{(0)} + u^{(1)}/\lambda$. The value of $\lambda_{\text{max}}$ is now found by solving the equations $\|u_{G_k}\|_1 = 1$ (for all groups $k : 1 \ldots g$). This can be done on a computer as $\|u^{(0)} + u^{(1)}/\lambda\|_\infty$ is a piecewise linear function of $\lambda^{-1}$. The largest positive value among those numbers is the desired value of $\lambda_{\text{max}}$. The group $k$ for which $\|u_{G_k}\|_1 = 1$ will be the first group to enter the set of active groups (in the next step) and become non-zero. Once this break point value of $\lambda$ is known, it can be used to update $u$ and $w$.

The algorithm then enters a loop which consists of several parts schematically described below (see also example in Subsection 4.2):

1. Express $\hat{\beta}$ as a linear function of $R$: This uses the knowledge of the active groups, of the maximal set within each group (from the knowledge of the subgradient $u$), of the relative signs of these components of $\beta$ (also from the knowledge of $u$), and of the components of $y - X\hat{\beta}$ that are zero (from the knowledge of the subgradient $w$).

2. Determine the largest value of $R$ for which the expression $\hat{\beta}(R)$ found in stage 1 is valid. As this expression is based on the values of the subdifferentials $u$ and $w$, it ceases to be valid when $u$ or $w$ change. This happens when:
   (a) an additional component of $r = y - X\hat{\beta}$ becomes zero (change of $w$),
   (b) an active group becomes non-active (all members are zero) (change of $u$),
   (c) a non-maximal component of $\hat{\beta}$ (in some group) becomes equal in absolute value to the maximal value in that group (change of $u$).

In practice, the breakpoint values of $R$ are determined by a linear equation (see example).

3. Once the value of $R$ is calculated, update the variables $u$ and $w$ (as a function of $\lambda$). Here equations (14) are used together with the knowledge of $u_i$ for the nonzero $r_i$. The knowledge of the non-maximal $\beta_j$ (in each group) is also used to set some $u_j$’s to zero.

4. Calculate the smallest value of $\lambda$ for which these expressions for $u$ and $w$ are valid. The expressions for $u$ and $w$ cease to be valid when:
   (a) One of the $\|u_{G_k}\|_1$ will reach 1 (a new active group will be added to the active set in the next step),
   (b) A coefficient of $w$ equals $2\tau$ or $-2(1 - \tau)$ (in this case, an equation $r_j = 0$ that is satisfied in the current step, will no longer be satisfied in the next step),
5. Continue with step 1 or stop.

The algorithm may be stopped when the desired maximum number of active groups is reached, when $\lambda = 0$, or when some other suitable stopping criteria is satisfied. Let us remark that a good understanding of the algorithm can only be obtained by going through the different steps manually on a toy example. Such a worked-out example is provided in Subsection 4.2.

In the algorithm above, steps 1 and 3 require the solution of a linear system of equations. Steps 2 and 4 require the solution of simple linear equations to determine $R$ or $\lambda$ at break points. Here numerical round-off error may affect the accuracy of these calculations. Unfortunately, the decisions (groups entering or leaving the active set, coefficients becoming submaximal in a group, . . .) depend on these numerical results. Round-off errors may therefore lead to the wrong decisions being taken by the algorithm. In that case the algorithm fails. This shortcoming is common to all the algorithms of this type (Efron et al. 2004; Li and Zhu 2008; Fuchs 2009).

When dealing with data $X$ and $y$ containing small integers, or when rows or columns of $y$ and $X$ repeat, it is possible that different events (2a–c or 4a–c) occur simultaneously. One could e.g. have two groups enter the active set at the same step. Another possibility is that a new group becomes active at the same step when a component of an active group becoming sub-maximal. Such possibilities are not accounted for in the current implementation of the algorithm (Loris and Nassiri 2013). This “one-at-a-time condition” (Efron et al. 2004, p417) is also common to algorithms of this type. (Efron et al. 2004; Li and Zhu 2008; Fuchs 2009) also do not handle such cases. (Efron et al. 2004, p438) have proposed to add a small amount of jitter to the variables to overcome the problem.

4.2 Illustrative Example

The most effective way of understanding the proposed algorithm is by going through a worked-out example step-by-step. Table 1 lists the complete solution path of a simple example with:

$$X = \begin{pmatrix} -4 & 3 & 5 \\ -4 & 5 & 1 \\ 4 & -3 & 0 \end{pmatrix}, \quad y = \begin{pmatrix} 8 \\ 7 \\ -11 \end{pmatrix},$$

(16)
groups $G_1 = \{1\}$ and $G_2 = \{2, 3\}$ and $\tau = 1/2$. The solution path is given as a function of $R$ and $\lambda$, and the intermediate values of the subgradients $u$ and $w$ are also given. This example was chosen in such a way that every possibility in steps 2 and 4 of the algorithm occurs at least once. The first two steps are explained in detail below.
At $R = 0$, we start with no variable in the model $\hat{\beta} = (0, 0, 0)^T$. The residuals corresponding to such a $\beta$ are $r = y - X\hat{\beta} = y = (8, 7, -11)^T$.

Therefore, the corresponding subdifferential $w$ is $w = \text{sign}(r) = (1, 1, -1)^T$.

Now one needs to determine the subdifferential $u = (u_1, u_2, u_3)^T \in \partial \|\hat{\beta}\|_{1,\infty}$ for this $\hat{\beta}$. The optimality conditions (11), $-X^Tw + \lambda u = 0$, yield:

$$
\begin{pmatrix}
4 & 4 & -4 \\
-3 & -5 & 3 \\
-5 & -1 & 0
\end{pmatrix}
\begin{pmatrix}
u_1 \\
u_2 \\
u_3
\end{pmatrix}
+ \lambda
\begin{pmatrix}
u_1 \\
u_2 \\
u_3
\end{pmatrix}
= \begin{pmatrix} 0 \\
0 \\
0
\end{pmatrix}
\Rightarrow
\begin{pmatrix} u_1 \\
u_2 \\
u_3
\end{pmatrix}
= \frac{1}{\lambda}
\begin{pmatrix}
-12 \\
11 \\
6
\end{pmatrix}
$$

On the other hand, as explained in Section 3, for the singleton group $G_1$, one has $|u_1| \leq 1$ and for group $G_2$, one has $|u_1| + |u_2| \leq 1$. It follows that $\lambda$ should be larger than 17, $\lambda \geq 17$. This completes the zeroth step. The results are summarized in the first line of Table 1.

Since the break point value $\lambda = 17$ is found using the components of $u$ of group $G_2$, it will become the first active group in the next step. Therefore, in the next step, $\beta$ will have the form $\beta = (0, \hat{\beta}_2, \hat{\beta}_3)^T$ (absence of first variable). Since none of the components of $u_{G_2}$ is zero, we also know that $|\hat{\beta}_2| = |\hat{\beta}_3|$. Furthermore, $u$ also tells us that $\hat{\beta}_2, \hat{\beta}_3 \geq 0$. We therefore parametrize $\hat{\beta}$ as

$$
\hat{\beta} = \begin{pmatrix} 0 \\
R \\
R
\end{pmatrix},
$$

where $R$ is a parameter (it is equal to $\|\hat{\beta}\|_{1,\infty}$). This parametrization is not valid for all values of $R \geq 0$. It stops being valid when the subdifferentials $u$ or $w$ change. The subdifferential $w$ is related to $r$, which now equals:

$$
r = y - X\hat{\beta} = \begin{pmatrix} 8 \\
7 \\
-11
\end{pmatrix} - \begin{pmatrix} -4 & 3 & 5 \\
-4 & 5 & 1 \\
4 & -3 & 0
\end{pmatrix} \begin{pmatrix} 0 \\
R \\
R
\end{pmatrix} = \begin{pmatrix} 8 - 8R \\
7 - 6R \\
-11 + 3R
\end{pmatrix}.
$$

The subdifferential $w$ changes when a component of $r$ changes sign. The three possibilities are:

$$
\begin{cases}
8 - 8R = 0 & \Rightarrow R = 1, \\
7 - 6R = 0 & \Rightarrow R = \frac{7}{6}, \\
-11 + 3R = 0 & \Rightarrow R = \frac{11}{3}.
\end{cases}
$$

Therefore, the smallest value of $R$ which makes one component in $r$ zero is $R = 1$. Thus, the next break point is located at $R = 1$ and expression (17) is valid for $0 \leq R \leq 1$.

At $R = 1$ one has $\hat{\beta} = (0, 1, 1)^T$ and $r = (0, 1, -8)^T$. The subdifferential $w$ corresponding to this value of $r$ is $w = (w_1, 1, -1)^T$, where $|w_1| \leq 1$. For determining $w_1$ and the subdifferential $u = (u_1, u_2, u_3)^T$ one may use the optimality conditions in (14) (three linear equations) together with the linear equation $u_2 + u_3 = 1$ (which must be satisfied because $\hat{\beta}_2$ and $\hat{\beta}_3$ are maximal
in group 2, and both are positive). Solving the resulting system of four linear equations:

\[
\begin{pmatrix}
4 & 4 & -4 \\
-3 & -5 & 3 \\
-5 & -1 & 0
\end{pmatrix}
\begin{pmatrix}
w_1 \\
u_1 \\
u_2 \\
u_3
\end{pmatrix}
+ \lambda
\begin{pmatrix}
w_1 \\
u_2 \\
u_3
\end{pmatrix}
= \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}
\text{ and } u_2 + u_3 = 1
\]

for the four unknowns \(w_1, u_1, u_2, u_3\) yields:

\[
w_1 = \frac{-9}{8} + \frac{\lambda}{8}, \quad u_1 = -\frac{1}{2} - \frac{7}{2\lambda}, \quad u_2 = \frac{3}{8} + \frac{37}{8\lambda}, \quad u_3 = \frac{5}{8} - \frac{37}{8\lambda},
\]

where \(\lambda\) remains as a parameter. This expression is not valid for all values of \(\lambda\). We already know that \(\lambda \leq 17\). We must also express that \(|w_1| \leq 1\), \(|u_1| \leq 1\) and \(|u_2| + |u_3| \leq 1\). One finds:

\[
\begin{cases}
|\frac{-9}{8} + \frac{\lambda}{8}| = 1 & \Rightarrow \lambda = 1, 17 \\
|\frac{-1}{2} - \frac{7}{2\lambda}| = 1 & \Rightarrow \lambda = -\frac{7}{3}, 7 \\
\left|\frac{3}{8} + \frac{37}{8\lambda}\right| + \left|\frac{5}{8} - \frac{37}{8\lambda}\right| = 1 & \Rightarrow \lambda \geq \frac{37}{5}.
\end{cases}
\]

The largest \(\lambda\) from these possibilities is \(\lambda = \frac{37}{5}\) (corresponding to \(u_3 = 0\) which indicates that \(\hat{\beta}_3\) will become sub-maximal in its group in the next step). The values of \(u\) and \(w\) at \(\lambda = \frac{37}{5}\) are: \(u = (-36/37, 1, 0)^T\) and \(w = (-1/5, 1, -1)^T\). This completes the first step. The results are summarized in the corresponding lines of Table 1.

As it is already mentioned, \(\hat{\beta}_1\) is sub-maximal in the next step; therefore, \(\hat{\beta}_1 = 0\) and \(\hat{\beta}_2 = R\) where \(R = \|\hat{\beta}\|_{1,\infty}\). The component \(\hat{\beta}_3\) can also be expressed in terms of \(R\) by imposing that \(r_1\) is zero (as found from the previous step). One has:

\[
0 = r_1 = (y - X\hat{\beta})_1 = 8 + 4\hat{\beta}_1 - 3\hat{\beta}_2 - 5\hat{\beta}_3 = 8 - 3R - 5\hat{\beta}_3
\]

or \(\hat{\beta}_3 = \frac{8}{5} - \frac{3}{2}R\) such that

\[
\hat{\beta} = \begin{pmatrix} 0 \\ \frac{8}{5} - \frac{3}{2}R \\ \frac{3}{2}R \end{pmatrix}.
\]

This expression is only valid for certain values of \(R\). We already know that \(R \geq 1\). We compute the residuals \(r\) and find:

\[
r = y - X\hat{\beta} = \begin{pmatrix} 0 \\ \frac{37}{5} - \frac{37}{5}R \\ -11 - 3R \end{pmatrix}
\]

The subddiffential \(u\) changes when a component of \(r\) changes sign. There are two possibilities:

\[
\begin{cases}
\frac{37}{5} - \frac{37}{5}R = 0 & \Rightarrow R = \frac{27}{5} \\
-11 - 3R = 0 & \Rightarrow R = \frac{11}{3}
\end{cases}
\]
The solution of this system is:

\[
\begin{align*}
\begin{pmatrix}
4 & 4 & -4 \\
-3 & -5 & 3 \\
-5 & 1 & 0 \\
\end{pmatrix}
\begin{pmatrix}
w_1 \\
w_2 \\
u_2 \\
u_3 \\
\end{pmatrix}
+ \lambda
\begin{pmatrix}
u_1 \\
u_2 \\
u_3 \\
\end{pmatrix}
= 
\begin{pmatrix}
0 \\
0 \\
0 \\
\end{pmatrix}
\text{ and } u_2 + u_3 = 1, \quad u_3 = 0.
\end{align*}
\]

The solution of this system is:

\[
w_1 = \frac{3}{22} - \frac{1}{22} \lambda, \quad w_2 = -\frac{15}{22} + \frac{5}{22} \lambda, \quad u_1 = -\frac{8}{11} - \frac{20}{11\lambda}, \quad u_2 = 1, \quad u_3 = 0.
\]

We already know that \( \lambda \leq \frac{20}{3} \) but we must still express the constraints \(|w_1|, |w_2| \leq 1, |u_1| \leq 1 \) and \(|u_2| + |u_3| \leq 1\):
The largest value of $\lambda$, smaller than $\frac{37}{5}$, in this list is $\frac{20}{3}$. Therefore, we have that expressions (19) are valid for $\frac{20}{3} \leq \lambda \leq \frac{37}{5}$ and at $\lambda = \frac{20}{3}$ one finds $u = (-1, 1, 0)^T$ and $w = (-\frac{1}{5}, \frac{5}{6}, -1)^T$. As the value $\lambda = \frac{20}{3}$ was found from $u_1 = -1$, group 1 will become active in the next step (and $\beta_1$ will be negative). The results are summarized in the corresponding lines of Table 1.

The algorithm can be continued in the same manner until $\lambda = 0$. It is clear that performing the calculations by hand is tedious and error prone. A set of Matlab functions that implement the above algorithm was written by the authors, and is available on their web page (Loris and Nassiri 2013). It also includes the details of the above example. The development and implementation of this algorithm is the main result of this paper.

5 Applications

The algorithm presented in Section 4 gives the entire solution path for different values of $R = \|\hat{\beta}\|_{1,\infty}$ in some interval $[0, R_{\text{max}}]$. Selecting the appropriate value of $R$ (and the corresponding coefficients $\hat{\beta}$) is an important issue for practical purposes. As Koenker et al. (1994) have proposed, the Bayesian information criterion (BIC) of Schwarz (1978) is a promising information criterion for model selection in quantile regression. In view of Theorem 2 of Li and Zhu (2008), and the loss function in (5), the adapted BIC for quantile regression is as follows:

$$
\text{BIC}(R) = \log \left( \frac{1}{n} \sum_{i=1}^{n} \varphi_{\tau} \left( (y - X\hat{\beta})_i \right) \right) - \log \left( \frac{n}{2n} \right) - \frac{\log(n)}{2n} \cdot nR, \quad (20)
$$

Fig. 2 Minimizers corresponding to the example $\mathbf{X}$ and $\mathbf{y}$ of equation (16). Left: parameters $\hat{\beta}$ as a function of $R = \|\hat{\beta}\|_{1,\infty}$. Right: the corresponding trade-off curve plotting the loss $\sum_{i=1}^{n} \varphi_{\tau}((y - X\hat{\beta})_i)$ as a function of the penalty $R = \|\hat{\beta}\|_{1,\infty}$. The slope of this curve is equal to $-\lambda$. The results are summarized in the corresponding lines of Table 1.
where \( \hat{\beta} \) is a function of \( R \) and \( n_R \) is defined as the number of zeros in the residual vector \( r = y - X\hat{\beta} \). The model with smaller BIC is more desirable (see e.g. Claeskens and Hjort (2008)). In this section two main applications of structured sparse quantile regression are studied using real data sets as illustrations.

5.1 Low birth weight data set

According to World Health Organization (1992), low birth weight (LBW) is defined as a birth weight of a liveborn infant of less than 2500g regardless of gestational age. LBW has negative effect both on the infants and the parents, e.g mothers of LBW babies have a greater chance of having postpartum depression and they need more time before returning to work, infants who are born with LBW are at greater risk of having learning or vision difficulties. Also, LBW infants would impose large costs on society. The risks of LBW are discussed by many authors such as, Boardman et al. (2002), Auslander et al. (2003) and Almond et al. (2005). Therefore, determining the effective factors in LBW infants is very important. If one is interested in studying the effects of different factors on the lower tail of the conditional distribution of infants’ weight, Koenker and Hallock (2001) have remarked that using the least squares regression methods (e.g. Yuan and Lin (2006)) is not reasonable. Using quantile regression with \( \tau \)-th quantile (\( \tau \leq 0.5 \)) would give the possibility to study lower tail of the infants weight given the explanatory variables.

As in Yuan and Lin (2006), the data are taken from Hosmer and Lameshow (1989). The data set contains the birth weight (expressed in grams) of 189 infants as the response variable and 8 explanatory variables: mother’s age (in years), mother’s weight (in pounds), mother’s race ((1, 0) = black, (0, 1) = white or (0, 0) = other), smoking status during pregnancy (1 =yes or 0 =no), number of previous premature labours (0, 1, 2, . . .), history of hypertension (1 = yes or 0 = no), presence of uterine irritability (1 = yes or 0 = no), number of physician visits during the first trimester (0, 1, 2, . . .). The data were collected at Baystate Medical Center, Springfield, Massachusetts, in the year 1986.

Although the \( \ell_1 \)-norm penalty does a good job in selecting individual variables, there is no control on the groups of variables it selects. There are cases where one is interested in selecting a group of variables instead of individual variables. In some cases it is inevitable, see Cohen (1991). Consider e.g. race as a predictor with three levels: black, white, and other. The standard approach is making two dummy variables \((X_1, X_2)\) out of it: (0, 0) for other, (1, 0) for black, and (0, 1) for white. Obviously in such a case one factor is represented by two variables. Therefore, one may select the pair \((X_1, X_2)\) together, or select none of them. The usual \( \ell_1 \)-norm penalty cannot guarantee such behavior.

As Koenker and Hallock (2001) and Yuan and Lin (2006) have suggested, some non-linear effects of the two quantitative predictors (mother’s weight and age) may exist. Therefore, in accordance with Koenker and Hallock (2001) we
consider a second-order polynomial for both of them. We put each corresponding pair of variables in one group. As mentioned, another variable which needs a non-singleton group is race (it is a nominal variable with more than two levels). All other groups are singletons.

As it is mentioned in Koenker and Hallock (2001), and considering Tukey’s dictum: *Never estimate intercepts, always estimate centercepts*, the quantitative variables in the model are centered and re-scaled by dividing by their standard deviations. Therefore, the estimated intercept may be interpreted as the weight of an infant born to a 23 year old mother, whose weight was 130 pounds, her race was other (neither black nor white), she was a non-smoker during her pregnancy, with on average 0.16 previous premature labours, no history of hypertension, no presence of irritability, and an average of 0.47 physician visits during the first trimester.

A small amount of jitter is added to the variables $y$ and $X$, to guarantee that the one-at-a-time condition mentioned at the end of Section 4 is satisfied (see e.g. also (Efron et al. 2004, p438)). We have verified that this does not change the outcome of the numerical experiments (see also Subsection 5.4).

Figure 3 (top) presents (part of) the solution path of the model coefficients $\hat{\beta}$ for $\tau = 0.1$, $\tau = 0.5$, and $\tau = 0.9$. A vertical dashed line is drawn at $R = R_{BIC}$ to indicate the coefficients chosen by the BIC (20). The bottom row of this figure contains the three corresponding model coefficients (chosen by BIC). As one may see, the models chosen for various values of $\tau$ are different, i.e. the effective variables for the lower tail of the conditional distribution are different from the ones for the median and for the upper tail. Since we are interested in studying the effective variables on the LBW, using loss functions such as least-squares (with minimizer equal to the conditional mean) would only give partial information. This result is in accordance with Koenker and Machado (1999) who remarked that the effects of the explanatory variables in LBW data set are not constant across the conditional distribution of the infant’s weight. In other words, for different quantiles one may have different models.

5.2 Simultaneous variable selection for a vector of response variables

Consider $Y = (y_1, y_2, \ldots, y_p)$ a vector of possibly correlated response variables. One is interested in modelling these variables using a common set of explanatory variables $X = (x_1, x_2, \ldots, x_m)$. For variable selection purposes, one may consider $p$ linear models $y_j = X\beta + \epsilon$, $(j = 1, \ldots, p)$ separately and find e.g. the penalized model solution for $\beta$ for each model. But as Turlach et al. (2005) have suggested, it is sometimes interesting to select variables by considering all $p$ response variables simultaneously, specially when these $p$ variables are correlated.

Suppose we have $n$ observations $y_{ij}$ (with $i = 1, \ldots, n$ and $j = 1, \ldots, p$) for $Y = (y_1, y_2, \ldots, y_p)$ as response variables and $X = (x_1, x_2, \ldots, x_m)$ as explanatory variables. Let $\beta_{kj}$, $(k = 1, \ldots, m$ and $j = 1, \ldots, p)$ be the regres-
An efficient algorithm for structured sparse quantile regression

0 5 0
0.5 1 1.5 RBIC
∥\hat{\beta}_{Gk}\parallel_{\infty} (k = 1, \ldots, 9)

0 2 4 0
0.5 1 1.5 RBIC
∥\hat{\beta}_{Gk}\parallel_{\infty} (k = 1, \ldots, 9)

0 2 4 0
0.5 1 1.5 RBIC
∥\hat{\beta}_{Gk}\parallel_{\infty} (k = 1, \ldots, 9)

Fig. 3 Analysis of the LBW data set. The first row shows (part of) the solution path given by the algorithm in Section 4 for \(\tau = 0.1, 0.5, 0.9\). The vertical dashed line indicates the model chosen by the BIC (for \(R = R_{BIC}\)). The second row shows the corresponding coefficients \(\hat{\beta}\) for the BIC model.

sion coefficient of \(y_j\) regressed on \(x_k\). As Turlach et al. (2005) have observed, \(\| (\beta_{k1}, \beta_{k2}, \ldots, \beta_{kp}) \|_{\infty} = \max \{|\beta_{k1}|, |\beta_{k2}|, \ldots, |\beta_{kp}|\}\) is a reasonable measure of the explanatory power of the regressor \(x_k\) on all \(p\) response variables \(y_j\), simultaneously. If the least-squares loss of Turlach et al. (2005) is replaced by the more robust loss function \((6)\), the following optimization problem:

\[
\hat{\beta} = \arg \min_{\beta} \sum_{i=1}^{n} \sum_{j=1}^{p} \varrho_{\tau} \left( y_{ij} - \sum_{k=1}^{m} x_{ik}\hat{\beta}_{kj} \right) + \lambda \sum_{k=1}^{m} \max \{|\beta_{k1}|, |\beta_{k2}|, \ldots, |\beta_{kp}|\}.
\]

(21)

should be solved to select the variables. Equivalently, one could also use the constrained formulation taking the form:

\[
\hat{\beta} = \arg \min_{\beta} \sum_{i=1}^{n} \sum_{j=1}^{p} \varrho_{\tau} \left( y_{ij} - \sum_{k=1}^{m} x_{ik}\beta_{kj} \right) \leq R \sum_{i=1}^{n} \sum_{j=1}^{p} \varrho_{\tau} \left( y_{ij} - \sum_{k=1}^{m} x_{ik}\beta_{kj} \right).
\]

(22)

As the argument of \(\varrho_{\tau}\) in these last two expressions is a linear function of the \(\beta_{kj}\), problems (21) and (22) are special cases of problems (9) and (10), and can therefore be solved by the algorithm of Section 4. As Turlach et al. (2005) pointed out, this is an exploratory tool for identifying a suitable subset of regressor variables, not for actual parameter estimation. The problem (21) has already been proposed by Zou and Yuan (2008) who solved it for a fixed value of \(R\) using a generic linear programming code. The algorithm of Section 4 finds the minimizer for a whole range of values of \(R\). The latter algorithm is
therefore more useful as the BIC criterion (20) (also used by Zou and Yuan (2008)) requires a further minimization over many values of the parameter $R$.

As an example, we consider the ‘93CARS’ data set which contains information on 93 new cars for the 1993 model year which is obtained from Lock (1993). Table 2 presents the variables we have considered. Some of the observations have been omitted due to missing values, so in total 82 observations are used.

Table 2 Variables in 93CARS data set.

| Description | Description |
|-------------|-------------|
| $y_1$: Minimum price (in $1,000) | $x_6$: Fuel tank capacity |
| $y_2$: Midrange price (in $1,000) | $x_7$: Passenger capacity |
| $y_3$: Maximum price (in $1,000) | $x_8$: Length |
| $y_4$: City MPG | $x_9$: Wheelbase |
| $y_5$: Highway MPG | $x_{10}$: Width |
| $x_1$: Number of cylinders | $x_{11}$: U-turn space |
| $x_2$: Engine size | $x_{12}$: Rear seat room |
| $x_3$: Horsepower | $x_{13}$: Luggage capacity |
| $x_4$: RPM | $x_{14}$: Weight |
| $x_5$: Engine revolutions per mile |

In this case, the 5 response variables are correlated, and there are some outliers in the data. Therefore, a simultaneous variables selection using least absolute deviation (QR with $\tau = 0.5$) seems reasonable here. Both response variables $y$ and regression variables $X$ are standardized, so one would be able to compare variables in different measures.

The algorithm of Section 4 is used with $\tau = 0.5$. In this application too, a small amount of jitter is added to the variables $y$, so as to guarantee that the one-at-a-time condition mentioned at the end of Section 4 is satisfied (see e.g. also (Efron et al. 2004, p438)). We have again verified that this does not change the outcome of the numerical experiments (see also Subsection 5.4). Figure 4 (left) shows the presence or absence of each group (as a function of $R = \|\hat{\beta}\|_{1,\infty}$).

The BIC (20) is used to select a model among all possible models calculated along the solution path. The corresponding value of $\|\hat{\beta}\|_{1,\infty}$ is called $R_{BIC}$, and is indicated on the first panel with a dotted line. The final model is given in Figure 4, (right). As one may see, the variables $x_2, x_8, x_{11}$ and $x_{12}$ were not selected.

5.3 Efficiency of the proposed algorithm

Firstly, the proposed algorithm is able to compute the minimizer of the cost functions (9) and (10) exactly. I.e. if no rounding errors are made, the algorithm of Section 4 will yield the exact result in a finite number of steps (each step only using addition, multiplication, comparison etc). Secondly, the algorithm also returns the values of $R$ in problem (10) for which a new group of
variables enters the active set. In other words, it gives a complete description of $\hat{\beta}(R)$ as a piece-wise linear function of $R$.

In contrast, iterative algorithms need a certain amount of iterations to arrive at an approximate minimizer of problem (9) or (10). The accuracy of the output depends on the number of iterations. Such iterative algorithms are typically applied to an instance of problem (10) and yield an approximate minimizer for that single problem (that single value of $R$). If another value of the parameter $R$ is used, the iteration must be run again. Obviously, such types of algorithms do not give as much information on $\hat{\beta}$ (as a function of $R$) as the non-iterative algorithm proposed in Section 4.

If $\beta$ is needed for just one (or a few) value(s) of the parameter $R$, then an iterative algorithm may be successfully applied. However, for applying the BIC criterion, it is necessary to know the minimizer $\hat{\beta}(R)$ for a wide range of values of $R$.

One example of a simple iterative algorithm for solving problem (10) is given by Chambolle and Pock (2011):

$$
\begin{align*}
\beta^{(n+1)} &= P_{R}^{1,\infty} \left[ \beta^{(n)} + \alpha X^T (2w^{(n)} - w^{(n-1)}) \right] \\
w^{(n+1)} &= P_{\tau}^{\infty} \left[ w^{(n)} - \alpha (X \hat{\beta}^{(n+1)} - y) \right],
\end{align*}
$$

(23)

where $P_{R}^{1,\infty}$ is the projection on the $\ell_{1,\infty}$ ball of radius $R$, $P_{\tau}^{\infty}$ is given by

$$
P_{\tau}^{\infty}(w)_i = \begin{cases} 
-2(1 - \tau) & \text{if } w_i < -2(1 - \tau) \\
w_i & \text{if } -2(1 - \tau) \leq w_i \leq 2\tau \\
2\tau & \text{if } w_i > 2\tau
\end{cases}
$$

(24)

and $\alpha$ is a step size parameter $\alpha < 1/\|X\|$. The projection $P_{R}^{1,\infty}$ can be implemented on a computer although no explicit expression exists (as in the case of $P_{\tau}^{\infty}$).

As an illustration we apply the iterative algorithm (23) to the problem (10) for the 93CARS data set (see previous Subsection). We first run the non-iterative algorithm of Section 4 to fully determine the piece-wise linear function
\( \hat{\beta}(R) \) (minimizer of (10) with \( \tau = 1/2 \)). The algorithm runs for approximately 20 seconds and finds that there are 751 knots in the function \( \hat{\beta}(R) \).

Next we run the algorithm (23) for each of the 751 values of \( R \) corresponding to a knot in \( \hat{\beta}(R) \), and for 100 iterations each. Figure 5 (Left) shows the convergence of \( \beta^{(n)} \) to \( \hat{\beta}(R) \) for one such value of \( R \) (\( R = 2.9269 \)). After 100 iterations, the distance to the exact minimizer \( \hat{\beta} \) (obtained from the proposed algorithm) is down to about 20% of \( \hat{\beta} \). In Figure 5 (Right) the remaining error on the approximate minimizer (after 100 iterations) is shown as a function of \( R \) (for all 751 values of \( R \) obtained previously).

The time spent by the iterative algorithm is 864s (for 751 times 100 iterations). The time spent by the proposed algorithm is 20s. The accuracy of the iterative algorithm depends on the value of \( R \). After 100 iterations the algorithm (23) may still be about 30% to 40% from the actual minimizer \( \hat{\beta} \).

One can also use existing convex optimization packages for solving problem (10) for specific values of \( R \). The CVX package (Grant and Boyd 2012, 2008) takes about 4 seconds to solve a single instance of problem (10) for the 93CARS data set (the difference between the CVX result and the proposed algorithm is on the order of \( 10^{-5} \)). As the proposed algorithm takes about 20s to compute the complete solution path, the CVX package can only compute about 5 points in the same amount of time (instead of the 751 that determine the solutions path completely). As in the case of the iterative algorithm (23), the CVX package cannot compute the values of \( R \) that correspond to the knots of the minimizer \( \hat{\beta}(R) \).

5.4 Choice of the amount of jitter

In this Subsection we provide some further details on the choice of the amount of jitter added to the variables in the LBW and 96CARS examples.

As was mentioned in Section 4, at each stage the algorithm determines which groups are active and which variables are maximal in their group from the numerical values of the subdifferentials \( u \) and \( w \). The algorithm also keeps track of which linear equations among the \( (y - X\hat{\beta})_i = 0 \) that are satisfied.
However, computers use floating point arithmetic which does not follow usual arithmetic exactly (e.g, on a computer one may have \( a - (a/b)b \neq 0 \)). The computer implementation (Loris and Nassiri 2013) of the proposed algorithm therefore uses a tolerance parameter to decide if a quantity is zero or nonzero, and to determine whether two quantities are equal in absolute value. As this tolerance parameter is strictly positive, it becomes possible in practice that two events occur simultaneously (e.g, two groups become active at the same step). The algorithm cannot handle this difficulty, an issue shared with all algorithms of this type (Efron et al. 2004; Osborne et al. 2000; Li and Zhu 2008; Fuchs 2009; Yuan and Lin 2006). The standard solution is to add a small amount of jitter to the variables \( y \) and \( X \) to attempt to avoid the problem.

It now becomes necessary to verify that this jitter does not significantly change the value of the estimated parameters \( \beta \). In order to study the effect of the jitter, a simulation study was designed for both applications. The estimation procedure (calculation of \( \hat{\beta}(R) \) for a range of values of \( R \) and selection of \( \hat{\beta} \) through the BIC criterion) was repeated 20 times. In each of these 20 trials, random jitter taken from a zero-mean normal distribution is added to each variable (to \( y \) and to each column of \( X \)). The standard deviation of this jitter equals \( 10^{-6}, 10^{-5}, 10^{-4} \) times the standard deviation of the variable it is added to.

The mean \( \overline{\beta} \) (over 20 trials) of the estimated parameter vectors \( \hat{\beta}^{(i)} \) (\( i : 1 \ldots 20 \)) is computed and the average of the norm of the differences of the estimated \( \hat{\beta}^{(i)} \) with the mean \( \overline{\beta} \) is calculated:

\[
\text{Mean difference} = \frac{1}{20} \sum_{i=1}^{20} \| \overline{\beta} - \hat{\beta}^{(i)} \|_2 / \| \overline{\beta} \|_2. \tag{25}
\]

In case of the LBW data, the experiment is also repeated for several values of \( \tau \). The results of the simulation study show that the mean difference between the different parameter estimations can be kept under \( 10^{-2} \) by choosing the jitter in this way. The Matlab code to run such a simulation study are provided in (Loris and Nassiri 2013).

6 Conclusions

A structured sparse solution (or group sparse solution) of a quantile regression model, based on penalizing or constraining the quantile regression loss function by a mixed \( l_1,\infty \)-norm of regression coefficients, was discussed. In particular, an algorithm to compute the solution of the corresponding minimization problem was presented. This algorithm computes the minimizer of the penalized or constrained loss function for all values of \( R = \| \hat{\beta} \|_{1,\infty} \) within a range \([0, R_{\text{max}}]\) instead of just for a single value of \( R \). This is a strong point when using the BIC criterion (which needs a further minimization over \( R \)) for model selection. Furthermore, the algorithm terminates in a finite number of steps and yields the exact minimizer (up to round-off errors) using only elementary operations.
(addition, multiplication, comparison, . . . ). We illustrated the effectiveness of the proposed algorithm by comparing it to an iterative minimization algorithm, and to a state-of-art convex optimization package.

The implementation of the algorithm is not straightforward. Therefore, such an implementation in Matlab is provided on the authors’ webpage (Loris and Nassiri 2013), together with the scripts for two applications and the comparison with an iterative algorithm and with a general-purpose convex optimization package.

Our proposed algorithm is not able to handle models such as those based on the extended quantile loss function of Chen et al. (2009) and Chen and Gerlach (2013). The present algorithm is tied to the special form of the objective function and penalty function in expressions (9) and (10), and cannot involve logarithms or square roots.

Acknowledgements I. L. is a research associate of the F.R.S.-FNRS (Belgium). This research was supported by VUB GOA-062 and by the FWO-Vlaanderen grant G.0564.09N.

References

R. Alhamzawi and K. Yu. Variable selection in quantile regression via Gibbs sampling. *J. Appl. Stat.*, 39(4):799–813, 2012.
R. Alhamzawi, K. Yu, and D. F. Benoit. Bayesian adaptive Lasso quantile regression. *Statistical Modelling*, 12(3):279–297, 2012.
D. Almond, K. Y. Chay, and D. S. Lee. The costs of low birth weight. *Q. J. Econ.*, 120(3):1031–1083, 2005.
G. K. Auslander, D. Netzer, and I. Arad. Parental anxiety following discharge from hospital of their very low birth weight infants. *Family Relations*, 52(1):12–21, 2003.
A. Belloni and V. Chernozhukov. $\ell_1$-penalized quantile regression in high-dimensional sparse models. *Ann. Stat.*, 39(1):82–130, 2011.
J. D. Boardman, D. A. Powers, Y. C. Padilla, and R. A. Hummer. Low birth weight, social factors, and developmental outcomes among children in the United States. *Demography*, 39(2):353–368, 2002.
S. Boyd and L. Vandenberghe. *Convex optimization*. Cambridge University Press, Cambridge, 2004.
A. Chambolle and T. Pock. A first-order primal-dual algorithm for convex problems with applications to imaging. *J. Math. Imaging Vis.*, 40(1):120–145, 2011.
C. W. S. Chen and R. Gerlach. Semi-parametric quantile estimation for double threshold autoregressive models with heteroskedasticity. *Computational Statistics*, 28(3):1103–1131, 2013.
C. W. S. Chen, R. Gerlach, and D. C. M. Wei. Bayesian causal effects in quantiles: accounting for heteroscedasticity. *Computational Statistics & Data Analysis*, 53:1993–2007, 2009. Special issue on Computational Econometrics.
G. Claeskens and N. L. Hjort. *Model Selection and Model Averaging*. Cambridge University Press, Cambridge, 2008.
A. Cohen. Dummy variables in stepwise regression. *The American Statistician*, 45(3):226–228, 1991.
D. L. Donoho. An invitation to reproducible computational research. *Biostatistics*, 11(3):385–388, 2010.
B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani. Least angle regression. *Ann. Statist.*, 32(2):407–499, 2004.
An efficient algorithm for structured sparse quantile regression

J. J. Fuchs. Fast implementation of a $\ell_1 - \ell_1$ regularized representations algorithm. *IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 3329–3332, 2009.

M. Grant and S. Boyd. Graph implementations for nonsmooth convex programs. In V. Blondel, S. Boyd, and H. Kimura, editors, *Recent Advances in Learning and Control*, Lecture Notes in Control and Information Sciences, pages 95–110. Springer-Verlag Limited, 2008.

M. Grant and S. Boyd. CVX: Matlab software for disciplined convex programming, version 2.0 beta. [http://cvxr.com/cvx](http://cvxr.com/cvx), sep 2012.

A. Hoel and R. Kennard. Ridge regression: bias estimation of nonorthogonal problems. *Technometrics*, 12(1):55–67, 1970.

D. W. Hosmer and S. Lameshow. *Applied Logistic Regression*. Wiley, 1989.

R. Koenker and G. Bassett. Regression quantiles. *Econometrica*, 46(1):33–50, 1978.

R. Koenker and V. D’Orey. Algorithm AS 229: Computing Regression Quantiles. *J. R. Stat. Soc. C*, 36(3):383–393, 1987.

R. Koenker and K. F. Hallock. Quantile regression. *J. Econ. Perspect.*, 15(4):143–156, 2001.

R. Koenker and J. A. F. Machado. Goodness of fit and related inference processes for quantile regression. *J. Amer. Statist. Assoc.*, 94(448):1296–1310, 1999.

R. Koenker and P. Ng. A Frisch-Newton algorithm for sparse quantile regression. *Acta Math. Appl. Sin.*, 21(2):225–236, 2005.

R. Koenker, P. Ng, and S. Portnoy. Quantile smoothing splines. *Biometrika*, 81(4):673–680, 1994.

E. L. Lehmann and G. Casella. *Theory of point estimation*. Springer Texts in Statistics. Springer-Verlag, New York, second edition, 1998.

Y. Li and J. Zhu. $\ell_1$-norm quantile regression. *J. Comp. Graph. Stat.*, 17(1):163–185, 2008.

R. H. Lock. 1993 new car data. *J. Stat. Educ.*, 1(1), 1993. URL [http://www.amstat.org/publications/jse/v1n1/datasets.lock.html](http://www.amstat.org/publications/jse/v1n1/datasets.lock.html).

I. Loris and V. Nassiri. Group quantile regression Matlab code, 2013. URL [http://homepages.vub.ac.be/~igloris/pub.html](http://homepages.vub.ac.be/~igloris/pub.html).

P. Ng. An algorithm for quantile smoothing splines. *Comput. Stat. Data Anal.*, 22(2):99–118, 1996.

M. R. Osborne, B. Presnell, and B. A. Turlach. A new approach to variable selection in least squares problems. *IMA J. Numer. Anal.*, 20(3):389–403, July 2000.

T. Park and G. Casella. The Bayesian lasso. *J. Amer. Statist. Assoc.*, 103(482):681–686, 2008.

R. T. Rockafellar. *Convex Analysis*. Princeton University Press, 1970.

S. Rosset and J. Zhu. Piecewise linear regularized solution paths. *Ann. Stat.*, 35(3):1012–1030, 2007.

G. Schwarz. Estimating the dimension of a model. *Ann. Statist.*, 6(2):461–464, 1978.

R. Tibshirani. Regression shrinkage and selection via the lasso. *J. Roy. Statist. Soc. Ser. B*, 58(1):267–288, 1996.

B. A. Turlach, W. N. Venables, and S. J. Wright. Simultaneous variable selection. *Technometrics*, 47(3):349–363, 2005.

World Health Organization. International statistical classification of diseases and related health problems (ICD-10). [http://www.who.int/classifications/icd/en/](http://www.who.int/classifications/icd/en/), 1992. 10th Revision.

K. Yu, C. W. S. Chen, C. Reed, and D. B. Dunson. Bayesian variable selection in quantile regression. *Statistics and Its Interface*, 6(2):261–274, 2013.

M. Yuan and Y. Lin. Model selection and estimation in regression with grouped variables. *J. R. Stat. Soc. Ser. B Stat. Methodol.*, 68(1):49–67, 2006.

H. Zou and M. Yuan. Regularized simultaneous model selection in multiple quantiles regression. *Comput. Stat. Data Anal.*, 52(12):5296–5304, 2008.