Journal of the Korean Data & Information Science Society
2010, **21**(5), 973–981

한국데이터정보과학회지

M-quantile regression using kernel machine technique†

Changha Hwang[1]

[1]Department of Statistics, Dankook University

Received 13 July 2010, revised 31 August 2010, accepted 6 September 2010

Abstract

Quantile regression investigates the quantiles of the conditional distribution of a response variable given a set of covariates. M-quantile regression extends this idea by a "quantile-like" generalization of regression based on influence functions. In this paper we propose a new method of estimating M-quantile regression functions, which uses kernel machine technique. Simulation studies are presented that show the finite sample properties of the proposed M-quantile regression.

Keywords: Expectile, iteratively reweighted least squares procedure, kernel machine, M-quantile, quantile.

1. Introduction

Regression analysis is a statistical tool for the investigation of relationships between a response variable y and some covariates $\boldsymbol{x} \in R^d$. Typical regression models have the form

$$y = m(\boldsymbol{x}) + \epsilon,$$

where the function m is usually referred to as the regression function or regression mean, and ϵ is a known random variable. In general, the regression function describes the middle of the point-cloud, in the y direction, as a function of the covariates $\boldsymbol{x}$. Note that the mean regression function is a symmetric least squares estimator. However, new insights about the underlying structure can be gained by investigating the higher or lower regions of the point-cloud. This leads us to study the estimation of the conditional percentiles of y given $\boldsymbol{x}$. These quantities are found to be useful descriptors of regression data sets. It started with a paper of Koenker and Bassett (1978) where they defined conditional quantiles or regression quantiles via an asymmetric absolute loss. Such a modeling exercise is referred to as quantile regression. Later Newey and Powell (1987) criticized the use of regression quantiles in linear models and instead of asymmetric absolute loss, they proposed an asymmetric squared loss. This leads to the concept of the conditional expectiles. Thus this makes it possible to investigate the higher or lower regions of the point-cloud via asymmetric squared loss. These authors called the resulting curves regression expectiles. Expectile regression is the

† The present research was conducted by the research fund of Dankook university in 2009.

[1] Professor, Department of Statistics, Dankook University, Gyeonggido 448-701, Korea.
E-mail: chwang@dankook.ac.kr

least squares analogue of quantile regression. Abdous and Remillard (1995) supported the theoretical work on the existence of a one-to-one mapping from expectiles to quantiles. More details and results on regression expectiles are given by Efron (1991) and Abdous and Remillard (1995). Taylor (2008) used expectiles as estimators of quantiles.

Similar to the conditional quantiles, the conditional expectiles also characterize the underlying conditional distribution. Thus, it provides an effective diagnostic tool such as testing the heteroscedasticity of regression models and the conditional symmetry of the noise terms. Quantiles and expectiles both characterize a distribution function although they are different in nature. Abdous and Remillard (1995) gave a sufficient condition under which a quantile and an expectile coincide. Although the asymmetric least squares (ALS) method is not as robust as the asymmetric least absolute deviations (ALAD) method against outliers, it has some desired features. For example, an ALS estimator is easier to compute and reasonably efficient under normality conditions (Efron, 1991). For interval prediction the conditional quantile is more appealing than the conditional expectile because of its conventional probability interpretation, whereas for general statistical diagnoses the conditional expectile is a valuable alternative to the conditional quantiles. Because neither is uniformly superior, the choice will usually depend on the particular application at hand. This situation is similar to the comparison between the conditional mean and the conditional median in conventional regression. On the other hand, Abdous and Remillard (1995) observed that for quite a large class of nonlinear regression models, the conditional expectiles as functions of $\boldsymbol{x}$ are in a one-one correspondence with the conditional quantiles. Thus, the ALS approach can be adapted to estimate conditional quantile directly.

Quantile regression can be viewed as a generalization of median regression. In the linear case, quantile regression leads to a family of hyperplanes indexed by the value of the corresponding quantile coefficient $q \in (0,1)$. In the same way, expectile regression is a "quantile-like" generalization of mean regression. *M*-quantile regression integrates these concepts within a common framework defined by a "quantile-like" generalization of regression based on influence functions (Breckling and Chambers, 1988). *M*-quantile regression has been successfully used in many real life problems. See for details Chambers and Tzavidis (2006), Taylor (2008), Tzavidis *et al.* (2008) and Vinciotti and Yu (2009). All the previous works of *M*-quantile regression concentrated on parametric models. Pratesi *et al.* (2009) studied nonparametric *M*-quantile regression based on penalized splines. In this paper we extend *M*-quantile with nonparametric modeling using kernel machine technique. Kenel machine is superior to spline based method in prediction performance and is simpler to use (Seok *et al.*, 2002). The outline of the paper is the following. Section 2 briefly reviews the *M*-quantile regression. Section 3 is devoted to *M*-quantile regression using kernel machine technique. Section 4 stretches the simulation study and its results. Section 5 presents the conclusions.

2. *M*-quantile regression

In this section we briefly review the basic concepts of quantile, expectile and *M*-quantile regressions. Koenker and Bassett (1978) introduced quantile regression as a generalization of median regression. In the linear case, quantile regression leads to a family of hyperplanes indexed by the value of the corresponding quantile coefficient $q \in (0,1)$. For each value of q, the corresponding model $Q_q(\boldsymbol{x}) = \boldsymbol{x}^t\boldsymbol{\beta}(q)$ explains how the qth quantile of the conditional distribution of y given $\boldsymbol{x}$ varies with $\boldsymbol{x}$. Given a sample of n observations $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^n$, the estimate of regression quantiles parameter vector $\boldsymbol{\beta}(q)$ is obtained by minimizing the ALAD function,

$$\sum_{i=1}^n |y_i - \boldsymbol{x}_i^t\boldsymbol{\beta}(q)| \left[(1-q)I(y_i - \boldsymbol{x}_i^t\boldsymbol{\beta}(q) \le 0) + qI(y_i - \boldsymbol{x}_i^t\boldsymbol{\beta}(q) > 0)\right] \tag{2.1}$$

with respect to $\boldsymbol{\beta}(q)$ by using linear programming methods. However, regression quantile hyperplanes are not comparable with the regression ones based on ordinary least squares that describe how the mean of y changes with $\boldsymbol{x}$. In fact, the former are based on an absolute deviations criterion, whereas the latter on a least squares one.

Newey and Powell (1987) criticized the use of regression quantiles in linear models and considered a generalization of expectation through expectile functions using an asymmetric squared loss instead of asymmetric absolute loss. The estimate of regression expectiles parameter vector $\boldsymbol{\beta}(q)$ is obtained by minimizing the ALS function,

$$\sum_{i=1}^n (y_i - \boldsymbol{x}_i^t\boldsymbol{\beta}(q))^2 \left[(1-q)I(y_i - \boldsymbol{x}_i^t\boldsymbol{\beta}(q) \le 0) + qI(y_i - \boldsymbol{x}_i^t\boldsymbol{\beta}(q) > 0)\right] \tag{2.2}$$

with respect to $\boldsymbol{\beta}(q)$. Expectile regression is a "quantile-like" generalization of mean regression. Expectile regression is computationally simpler than quantile regression. Schnabel and Eilers (2009) studied nonparametric expectile regression based on smoothing spline.

It is well known how linear least squares estimates can behave badly when the error distribution is not normal, particularly when the errors are heavy-tailed. To address this issue, two different approaches have been developed in the literature. One approach, called robust regression, is to employ a fitting criterion that is not as vulnerable as least squares to unusual data. The most common general method of robust regression is *M*-estimation. The second approach, which is also robust to large outliers, is the one of quantile regression. *M*-quantile regression integrates both *M*-regression and quantile regression, by providing a "quantile-like" generalization of regression based on influence functions (Breckling and Chambers, 1988).

We now need to extend the ALS to get *M*-quantile regression. Following Huber (1981) and Stone (2005), let $\rho(\cdot)$ be a (necessarily continuous) convex function on $(-\infty, \infty)$. Then, the *M*-regression can be formulated via the log-likelihood function for robust estimation of $m(\boldsymbol{\beta})$ in the classical sense

$$\sum_{i=1}^n \rho(y_i - m(\boldsymbol{x}_i)). \tag{2.3}$$

Usually, $\rho(u)' = \psi(u)$ is the influence function. Since the focus of our paper is on *M*-estimation, we may use different robust loss functions such as the Huber or the Hampel functions. In this paper, we employ the Huber function as follows: $\rho(u) = u^2/2$ for $|u| \le c$ and $\rho(u) = c|u| - u^2/2$ for $|u| > c$, where c is a positive constant. Then the corresponding influence function, termed Huber Proposal 2, is given by $\psi(u) = uI(|u| \le c) + c\ \mathrm{sign}(u)$, provided c is bounded away from zero. Therefore, corresponding to

$$\psi_q(u) = 2\psi(s^{-1}u)\left[(1-q)I(u \le 0) + qI(u > 0)\right], \tag{2.4}$$

we have

$$\rho_q(u) = 2\rho(\frac{u}{s})\left[(1-q)I(u \le 0) + qI(u > 0)\right], \tag{2.5}$$

where s is a robust estimate of scale. Here we use $s = \text{median}\{|u|\}/0.6745$. Then, in a linear M-quantile regression model the estimate of parameter vector $\boldsymbol{\beta}(q)$ is obtained by minimizing the asymmetric robust function,

$$\sum_{i=1}^{n} \rho_q(y_i - \boldsymbol{x}_i^t \boldsymbol{\beta}(q)) \tag{2.6}$$

with respect to $\boldsymbol{\beta}(q)$.

Setting the first derivative of (2.6) leads to the following estimating equations

$$\sum_{i=1}^{n} \psi_q(y_i - \boldsymbol{x}_i^t \boldsymbol{\beta}_\psi(q))\boldsymbol{x}_i = \mathbf{0}_d, \tag{2.7}$$

where subscript ψ denotes the influence function associated with the M-quantile. Provided that the tuning constant c is strictly greater than zero, the estimate of $\boldsymbol{\beta}_\psi(q)$ is obtained using iterative reweighted least squares (IRWLS) algorithm that guarantees the convergence to a unique solution (Kokic *et al.*, 1997). The steps of the algorithm for fitting the M-quantile regression model $Q_q(\boldsymbol{x}, \psi) = \boldsymbol{x}^t \boldsymbol{\beta}_\psi(q)$ are as follows:

1. Select initial estimates $\boldsymbol{\beta}_\psi(q)$ and s.
2. Estimate the residuals $e_i(q) = y_i - \boldsymbol{x}_i^t \boldsymbol{\beta}_\psi(q)$.
3. Define weights $w_i(q) = \frac{\psi_q(e_i(q))}{e_i(q)}$.
4. Update the estimate of $\boldsymbol{\beta}_\psi(q)$ using weighted least squares regression with weights $w_i(q)$
5. Iterate steps 2, 3 and 4 until convergence.

3. Kernel M-quantile regression

In this section we devise an algorithm of estimating semiparametric M-quantile regression model based on kernel machine, which is referred here to as kernel M-quantile regression (KMQR) model. Kernel machine is a class of algorithms for pattern analysis and whose best known element is the support vector machine (SVM). General semiparametric regression models contain both nonparametric and ordinary linear (parametric) components. Semiparametric models can be used instead of fully nonparametric models when some covariates are linear and/or discrete, as is often the case in real data analysis. Semiparametric regression model can be viewed as an extension of nonparametric regression. Pratesi *et al.* (2009) proposed nonparametric M-quantile regression using penalized splines. Kernel machine tackles the nonlinear problem using kernel trick, which is a method for using a linear algorithm only through kernel function to solve a nonlinear problem by mapping the original observations into a high dimensional feature space. Kernel machine owes its name to the use of kernel functions. Kernel machine has better prediction performance than spline based method (Seok *et al.*, 2002).

We will now illustrate how to construct a semiparametric *M*-quantile regression model using the kernel trick. The principle is very similar to that of Pratesi *et al.* (2009). Let $\boldsymbol{x}_i \in R^d$ be partitioned into $\boldsymbol{x}_{1i} \in R^{d_1}$ and $\boldsymbol{x}_{2i} \in R^{d_2}$ components such that $d_1 + d_2 > d$. This implies that some covariates can be included in both components. There is no loss of generality that we assume the first element of $\boldsymbol{x}_{1i}$ is 1. We assume that the y_i is related to covariate vector $\boldsymbol{x}_i$ in a semiparametric form as

$$y_i = \boldsymbol{x}_{1i}^t \boldsymbol{\beta} + \boldsymbol{k}_i^t \boldsymbol{\gamma} + \epsilon_i, i = 1, 2, \cdots, n, \tag{3.1}$$

where $\boldsymbol{\beta}$ and $\boldsymbol{\gamma}$ are $d_1 \times 1$ and $d_2 \times 1$ regression parameter vectors, $\boldsymbol{\beta}$ includes constant β_0 as the first element, and $\boldsymbol{k}_i$ is the ith column of the $n \times n$ kernel matrix $\boldsymbol{K}$ with elements $K_{ij} = \boldsymbol{\phi}(\boldsymbol{x}_{2i})^t \boldsymbol{\phi}(\boldsymbol{x}_{2j}) = K(\boldsymbol{x}_{2i}, \boldsymbol{x}_{2j})$ for a nonlinear feature mapping function $\boldsymbol{\phi}, i, j = 1, 2, \cdots, n$. Here K is a kernel function. Several choices of kernel functions are possible but Gaussian kernel function is the most frequently used kernel function. For semiparametric model, we assume the covariates $\boldsymbol{x}_{1i}$ in the parametric part of the regression function have a linear effect on y_i, and the effect of covariates $\boldsymbol{x}_{2i}$ in the nonparametric part on y_i is not specified.

Similar to estimating equations (2.7) we have for semiparametric model the following estimating equations

$$\sum_{i=1}^{n} \psi_q(y_i - \boldsymbol{x}_{1i}^t \boldsymbol{\beta}_\psi(q) - \boldsymbol{k}_i^t \boldsymbol{\gamma}_\psi(q)) \begin{pmatrix} \boldsymbol{x}_{1i} \\ \boldsymbol{k}_i \end{pmatrix} = \mathbf{0}_{(d_1+n)}. \tag{3.2}$$

Note that the estimates obtained from (3.2) tend to be overfitted. In order to avoid this problem, additional penalization term can be added. We construct the following set of $(d_1 + n)$ penalized estimating equations

$$\sum_{i=1}^{n} \psi_q(y_i - \boldsymbol{x}_{1i}^t \boldsymbol{\beta}_\psi(q) - \boldsymbol{k}_i^t \boldsymbol{\gamma}_\psi(q)) \begin{pmatrix} \boldsymbol{x}_{1i} \\ \boldsymbol{k}_i \end{pmatrix} + \lambda \begin{pmatrix} \mathbf{0}_{d_1} \\ \boldsymbol{\gamma}_\psi(q) \end{pmatrix} = \mathbf{0}_{(d_1+n)}, \tag{3.3}$$

where λ is the penalization parameter that controls the level of smoothness of the resulting fit. The approximation ability of this final estimate will heavily depend on hyperparameters which are the penalization parameter and the kernel parameter. To determine the optimal values of hyperparameters we use here generalized cross validation (GCV) which has been successfully applied to kernel machines in Hwang (2007, 2008, 2010) and Shim and Lee (2009).

Let us rewrite the set of estimating equations in (3.3) as follows:

$$\sum_{i=1}^{n} \psi_q(y_i - \boldsymbol{u}_i^t \boldsymbol{\eta}_\psi(q)) \boldsymbol{u}_i + \lambda \boldsymbol{G} \boldsymbol{\eta}_\psi(q) = \mathbf{0}_{(d_1+n)}, \tag{3.4}$$

where $\boldsymbol{u}_i = (\boldsymbol{x}_i^t, \boldsymbol{k}_i^t)^t$, $\boldsymbol{\eta}_\psi(q) = (\boldsymbol{\beta}_\psi(q)^t, \boldsymbol{\gamma}_\psi(q)^t)^t$ and $\boldsymbol{G} = \text{diag}\{\mathbf{0}_{d_1}, \mathbf{1}_n\}$. If we define the weight function $w(e) = \psi_q(e)/e$ and let $w_i = w(e_i)$, then (3.4) can be written as

$$\sum_{i=1}^{n} w_i(y_i - \boldsymbol{u}_i^t \boldsymbol{\eta}_\psi(q)) \boldsymbol{u}_i + \lambda \boldsymbol{G} \boldsymbol{\eta}_\psi(q) = \mathbf{0}_{(d_1+n)}. \tag{3.5}$$

Solving this set of estimating equations is a penalized weighted least squares problem in which weights, residuals and coefficients depend one upon another. Further, the values of the hyperparameters have to be chosen. The GCV criterion to be optimized (minimized) to this end is the following

$$GCV(\boldsymbol{\theta}) = \frac{\|(\boldsymbol{I} - \boldsymbol{S}_{\boldsymbol{\theta}})\boldsymbol{y}\|^2}{\left(1 - n^{-1}\mathrm{tr}(\boldsymbol{S}_{\boldsymbol{\theta}})\right)^2}, \tag{3.6}$$

where $\boldsymbol{\theta}$ denotes the vector of hyperparameter, $\boldsymbol{S}_{\boldsymbol{\theta}}$ is the smoother matrix associated with $\hat{m}_{\psi,q}(\boldsymbol{u}_i) = \boldsymbol{S}_{\boldsymbol{\theta}}\boldsymbol{y}$ and $\boldsymbol{y} = (y_1, \cdots, y_n)^t$. Here $\hat{m}_{\psi,q}(\boldsymbol{u})$ denotes the estimator of $m(\boldsymbol{u})$ for fixed ψ and q.

The steps of the IRWLS algorithm for fitting the KMQR model are here proposed. In this algorithm we consider the influence function ψ and the quantile coefficient q. However, we will drop subscripts and indexes for notational convenience when this will not lead to ambiguity.

1. Select initial estimates $\boldsymbol{\eta}^0$.
2. At each iteration ℓ, calculate residuals $e_i^{(\ell-1)} = y_i - \boldsymbol{u}_i^t\boldsymbol{\eta}^{(\ell-1)}$ and associated weights $w_i^{(\ell-1)}$ from the previous iteration.
3. Optimize the $GCV(\boldsymbol{\theta})$ criterion over a grid of $\boldsymbol{\theta}$ values and obtain $\boldsymbol{\theta}^*$.
4. Calculate the new weighted penalized least squares estimates as
$$\boldsymbol{\eta}^{(\ell)} = [\boldsymbol{U}^t\boldsymbol{W}^{(\ell-1)}\boldsymbol{U} + \lambda^*\boldsymbol{G}]^{-1}\boldsymbol{U}^t\boldsymbol{W}^{(\ell-1)}\boldsymbol{y},$$
where $\boldsymbol{U} = \{\boldsymbol{u}_i\}_{i=1}^n$ and $\boldsymbol{W}^{(\ell-1)} = \mathrm{diag}\{w_1^{(\ell-1)}, \cdots, w_n^{(\ell-1)}\}$ are the current weight matrices.
5. Iterate steps 2, 3 and 4 until convergence.

4. Simulation studies

In this section, we illustrate the performance of the quantile estimation of the proposed KMQR model through two simulated data sets. We compare the performance of the KMQR model with those of Shim and Hwang (2010) and Takeuchi *et al.* (2006), which are referred to as support vector quantile regression using IRWLS (SVQR_IRWLS) and support vector quantile regression using quadratic programming (SVQR_QP), respectively. The tuning constant c associated with Huber proposal 2 influence function is set to $c = 1.345$. The numerical studies are conducted in MATLAB environment.

Example 4.1 We generate one training data set of size 300 and 100 test data sets of size 300 in the same manner to Shim and Hwang (2010). The univariate input observations x's are equally spaced ranging from 0 to π, the corresponding responses y's are drawn from a univariate normal distribution with mean and variance that vary smoothly with x as follows:

$$y = \sin\left(\frac{5x}{2}\right)\sin\left(\frac{3x}{2}\right) + \sigma(x)\epsilon \quad \text{with} \quad \sigma(x) = \sqrt{\frac{1}{100} + \frac{1}{4}\left\{1 - \sin\left(\frac{5x}{2}\right)\right\}^2}, \ \epsilon \sim N(0,1).$$

The Gaussian kernel function is utilized in this example, which is

$$K(x_i, x_j) = \exp\left(-\frac{1}{\sigma^2}\|x_i - x_j\|^2\right),$$

where σ^2 is the kernel parameter. In SVQR_IRWLS the values of the hyperparameters are chosen by its own GCV function. In SVQR_QP the values of the hyperparameters are chosen by CV function. For details see Shim and Hwang (2010). To illustrate the prediction performance of KMQR, we compare it with SVQR_IRWLS and SVQR_QP via 100 data sets, where the mean squared errors (MSEs) are used as the estimation performance measures defined by

$$\text{MSE} = \frac{1}{300}\sum_{i=1}^{300}\left(Q_q(\boldsymbol{x}_i) - \hat{m}_{\psi,q}(\boldsymbol{x}_i)\right)^2$$

and

$$\text{MSE} = \frac{1}{300}\sum_{i=1}^{300}\left(Q_q(\boldsymbol{x}_i) - \hat{Q}_q(\boldsymbol{x}_i)\right)^2$$

for KMQR and SVQR_IRWLS, SVQR_QP, respectively. The averages of 100 MSEs from KMQR, SVQR_IRWLS and SVQR_QP are given in Table 4.1. We can see that KMQR has somewhat bigger average of MSEs than SVQR_IRWLS and SVQR_QP for each quantile coefficient except 0.5.

Table 4.1 Average of 100 MSEs for Example 4.1.

Methods	$q = 0.05$	$q = 0.1$	$q = 0.5$	$q = 0.9$	$q = 0.95$
KMQR	0.1055	0.0695	0.0102	0.0513	0.0879
SVQR_IRWLS	0.0236	0.0173	0.0082	0.0153	0.0262
SVQR_QP	0.0237	0.0174	0.0084	0.0160	0.0224

Example 4.2 We generate one training data set of size 300 and 100 test data sets of size 300 in the same manner to Shim and Hwang (2010). The univariate input observations x's are equally spaced ranging from 0 to 2, the corresponding responses y's are drawn from a univariate χ^2 distribution with mean and variance that vary smoothly with x as follows:

$$y = \sin(2\pi x) + \sigma(x)\epsilon \quad \text{with} \quad \sigma(x) = \sqrt{\frac{2.1 - x}{4}},\ \epsilon \sim \chi^2_{(2)} - 2.$$

Here $\chi^2_{(2)}$ is the chi-squared distribution with degree of freedom 2. The Gaussian kernel function is also utilized in this example. As in Example 4.1, the values of the hyperparameters are chosen by GCV or CV function. The averages of 100 MSEs from KMQR, SVQR_IRWLS and SVQR_QP are given in Table 4.2. As in Example 4.1, we can see that KMQR has somewhat bigger average of MSEs than SVQR_IRWLS and SVQR_QP for each quantile coefficient except 0.5.

Table 4.2 Average of 100 MSEs for Example 4.2.

Methods	$q = 0.05$	$q = 0.1$	$q = 0.5$	$q = 0.9$	$q = 0.95$
KMQR	0.0403	0.0529	0.0480	0.2191	0.4984
SVQR_IRWLS	0.0020	0.0034	0.0303	0.2452	0.4191
SVQR_QP	0.0019	0.0033	0.0456	0.2162	0.3871

5. Conclusions

In this paper, we dealt with estimating the KMQR model, which has lots of potential applications. Through the examples we recognized that the proposed KMQR model has significant differences in estimating quantiles from SVQR_IRWLS and SVQR_QP except for $q = 0.5$. We made sure that quantiles and expectiles are different in nature, although expectile regression and more generally M-quantile regression can be used to characterize the relationship between a response variable and covariates when the behaviour of "nonaverage" individuals is of interest. We also found that both KMQR and SVQR_IRWLS are much faster than SVQR_QP, which implies that the proposed KMQR method is appropriate for the large training data sets.

References

Abdous, B. and Remillard, B. (1995). Relating quantiles and expectiles under weighted-symmetry. *Annals of the Institute of Statistical Mathematics*, **47**, 371-384.

Breckling, J. and Chambers, R. (1988). *M*-quantiles. *Biometrika*, **75**, 761-771.

Chambers, R. and Tzavidis, N. (2006). M-quantile models for small area estimation. *Biometrika*, **93**, 255-268.

Efron, B. (1991). Regression percentiles using asymmetric squared error loss. *Statistica Sinica*, **1**, 93-125.

Huber, P. J. (1981). *Robust Statistics*, Wiley, New York.

Hwang, C. (2007). Kernel machine for Poisson regression. *Journal of Korean Data & Information Science Society*, **18**, 767-772 .

Hwang, C. (2008). Mixed effects kernel binomial regression. *Journal of Korean Data & Information Science Society*, **19**, 1327-1334 .

Hwang, C. (2010). Support vector quantile regression for longitudinal data. *Journal of Korean Data & Information Science Society*, **21**, 309-316.

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

Kokic, P., Chambers, R., Breckling, J. and Beare, S. (1997). A measure of production performance. *Journal of Business and Economic Statistics*, **10**, 419-435.

Newey, N. K. and Powell, J. L. (1987). Asymmetric least squares estimates and testing. *Econometrica*, **55**, 819-847.

Pratesi, M., Ranalli, M. G. and Salvati, N. (2009). Nonparametric M-quantile regression using penalized splines. *Journal of Nonparametric Statistics*, **21**, 287-304.

Schnabel, S. K. and Eilers, P. H. C. (2009). Optimal expectile smoothing. *Computational Statistics and Data Analysis*, **53**, 4168-4177.

Seok, K., Hwang, C. and Cho, D. (2002). Prediction intervals for support vector machine regression. *Communications in Statistics-Theory and Methods*, **31**, 1887-1898.

Shim, J. and Lee, J. (2009). Kernel method for autoregressive data. *Journal of Korean Data & Information Science Society*, **20**, 949-954.

Shim, J. and Hwang, C. (2010). Support vector quantile regression with weighted quadratic loss function. *Communications of the Korean Statistical Society*, **17**, 183-191.

Stone, C. J. (2005). Nonparametric M-regression with free knot splines. *Journal of Statistical Planning and Inference*, **130**, 183-206.

Takeuchi, I., Le, Q. V., Sears, T. D. and Smola, A. J. (2006). Nonparametric quantile estimation. *Journal of Machine Learning Research*, **7**, 1231-1264.

Taylor, J. W. (2008). Estimating value at risk and expected shortfall using expectiles. *Journal of Financial Econometrics*, **6**, 231-252.

Tzavidis, N., Salvati, N., Pratesi, M. and Chambers, R. (2008). M-quantile models with application to poverty mapping. *Statistical Methods and Applications*, **17**, 393-411.

Vinciotti, V. and Yu, K. (2009). M-quantile regression analysis of temporal gene expression data. *Statistical Applications in Genetics and Molecular Biology*, **8**, Article 41.