Quantile autoregression neural network model

Qifa Xu[a,b], Xi Liu[a], Cuixia Jiang[a,*], Keming Yu[c]

[a]*School of Management, Hefei University of Technology, Hefei 230009, Anhui, PR China*
[b]*Key Laboratory of Process Optimization and Intelligent Decision-making, Ministry of Education, Hefei 230009, Anhui, PR China*
[c]*Department of Mathematics, Brunel University London, Kingston Lane, Uxbridge, Middlesex UB8 3PH,, UK*

Abstract

We develop a new quantile autoregression neural network (QARNN) model based on an artificial neural network architecture. The proposed QARNN model is flexible and can be used to explore potential nonlinear relationships among quantiles in time series data. By optimizing an approximate error function and standard gradient based optimization algorithms, QARNN outputs conditional quantile functions recursively. The utility of our new model is illustrated by Monte Carlo simulation studies and empirical analyses of three real stock indices from the Hong Kong Hang Seng Index (HSI), the US S&P500 Index (S&P500) and the Financial Times Stock Exchange 100 Index (FTSE100).

Keywords:
Artificial neural network, Quantile autoregression neural network (QARNN), Quantile autoregression, Quantile regression, Value-at-Risk

1. Introduction

Quantile regression (QR) proposed by Koenker & Bassett (1978) provides an alternative way to explore the true relationship among variables. Instead of mean regression, quantile regression focuses on the entire conditional quantiles of a response y given predictors z. QR has been widely applied in many disciplines, such as

*Corresponding author at: School of Management, Hefei University of Technology, Hefei 230009, Anhui, PR China. Tel: +86 551 62919150; fax: +86 551 62905263.

Email addresses: xuqifa1975@gmail.com (Qifa Xu), lisal1992@163.com (Xi Liu), jiangcuixia@hfut.edu.cn (Cuixia Jiang), Keming.Yu@brunel.ac.uk (Keming Yu)

financial economics, survival analysis, environmental modelling, see Yu et al. (2003) for more details.

While the linear-in-parameters QR offers an effective approach in many applications and can be tested using methods of Jiang et al. (2014), it may misspecify the underlying functional relationships, which are often nonlinear-in-parameters. However, as pointed out by Koenker (2005), it is still challenging to specify an appropriate nonlinear function for nonlinear quantile regression. A seminal work in this area is Engle & Manganelli (2004) who provided the conditional autoregressive value at risk (CAViaR) model, which has been widely extended, see Chen et al. (2012) and Jeon & Taylor (2013). But the nonlinear relationship in CAViaR needs to be set according to experience in advance. This prior specification of the nonlinear functional form is probably not appropriate for some complicated data.

Inspired by Cannon (2011), in order to overcome the nonlinearity issue, we consider a nonlinear quantile regression in the context of time series and develop a quantile autoregression neural network (QARNN) model by adding an artificial neural network (ANN) structure to quantile autoregression (QAR) model. The QARNN model is flexible and can implement a nonlinear quantile autoregression for time series data and estimate nonlinear relationships without the need to specify a precise functional form. To illustrate the efficacy of QARNN, we conduct Monte Carlo simulation studies and real world applications. The numerical results show that the QARNN model is able to describe dynamics in various financial time series and generally outperforms other classical models, such as GARCH-type models, CAViaR models, Riskmetrics and QRNN model, in terms of the accuracy for VaR evaluation.

The rest of the paper is organized as follows. Section 2 explores the literature review from both technical and application aspects. Section 3 gives a brief overview of quantile autoregression methods and presents the QARNN model in details. Monte Carlo simulation studies and real world applications are carried out in Section 4. We conclude with a summary in Section 5. More details of QARNN model is elaborated in the Appendix.

2. Literature review

Quantile regression presents a comprehensive strategy for giving an impression of the entire conditional distribution of a response y given x instead of the conditional mean only. The idea behind quantile regression can be traced back to the advance in loss functions.The asymmetric loss function (check function) in Koenker (2005)

$$\rho_\tau(u) = \begin{cases} \tau u & u \geq 0 \\ (\tau - 1)u & u < 0 \end{cases} \tag{1}$$

with $\tau \in (0,1)$ can yield quantiles. It captures the rationale, recalling that the quadratic loss function yields mean, while the absolute value loss function yields median.

The parametric and nonparametric quantile regression have been extensively studied in the last decades. In an important generalization of the quantile regression model, Powell (1984, 1986) introduced the censored quantile regression model, which consistently estimates conditional quantiles when observations on the dependent variable are censored. Yu & Jones (1998) proposed the nonparametric regression quantile estimation by kernel weighted local linear fitting. Auśin et al. (2014) developed a Bayesian semiparametric approach to relax the constraints of GARCH models, utilizing mixtures of Gaussian distribution with a Dirichlet process prior. See more details in Koenker (2005), which gave an overview of theory, methodology, and application of quantile regression.

As mentioned above, nonlinear quantile regression experience is much more limited, compared to extensive computational experience with quantile regression computation for linear-in-parameters models. In order to address the problem, Chen et al. (2009) proposed a copula-based nonlinear quantile autoregression, which addressed the possibility of deriving nonlinear parametric models for families of conditional quantile functions. This method can address nonlinearity, asymmetry, serial dependence as well as the heavy-tails of financial assets marginal and joint probability distribution. Inspired by Chen et al. (2009), Righi & Ceretta (2015) proposed a pair-copula construction approach, considering the dependence with past observations isolating the effect for lags. Koenker (2005) and Ghouch & Genton (2009) developed local polynomial quantile regression model via adopting the local polynomial expansion. Koenker et al. (1994) explored a class of quantile smoothing splines, which extended by Koenker & Mizera (2004) with introducing a continuous, piecewise linear function over the trigram model.

Moreover, nonparametric methods, such as ANN, has been widely employed to explore complex nonlinearities due to its flexibility (Saravanan et al., 2010). ANN is also used to develop a nonlinear quantile regression approach. Taylor (2000) proposed a nonparametric and nonlinear quantile regression neural network (QRNN) method, which combined the advantages of both quantile regression and neural network. This method efficiently reveals the entire conditional distribution of the response variable and also simulates the nonlinearity of financial system comprehensively. The QRNN model (Taylor, 2000) is also applied to estimate the conditional distribution of multi-period returns, which avoids the need to specify appropriate explanatory variables in Taylor (1999).

A nonparametric conditional quantile estimation via nerual network structure,

developed by Feng et al. (2010), has been applied to credit portfolio data analysis. The results showed that QRNN is more robust in fitting outliers compared to both local linear regression and spline regression. Cannon (2011) implemented QRNN model in R, which provided a comprehensive package and Yeh (2014) applied QRNN into estimating distribution of concrete strength through adding an ANN structure to the QR model for cross sectional data.

In the field of time series analysis, autocorrelation is a common phenomenon and can be fully revealed by an Autoregression (AR) method. Following the idea of the AR, QAR model is developed by Koenker & Xiao (2006) and Galvao Jr et al. (2013). In their work, the model is still linear-in-parameters. Additive models for quantile regression, proposed by Gouriéroux & Jasiak (2008) and Koenker (2011), show both autoregressive and dynamic properties of quantile. Although a threshold quantile autoregressive model is proposed by Galvao Jr et al. (2011) to investigate the asymmetric dynamics in financial time series, there is relatively little literature that considers nonlinear quantile regressions in the context of time series. Moreover, it is worth noting that the QRNN model only takes explanatory variables as inputs, without considering the nonlinear-in-parameters effects of exogenous variables or the lagged endogenous variables.

Therefore, in order to address the problem, we develop a general strategy QARNN for deriving families of nonlinear-in-parameters conditional quantile functions in time series data analysis via adding an ANN structure to QAR model. The novel proposed method is elaborated in Section 3.

3. Model Setup

In this section we first give a brief overview of some popular quantile autoregression models in the context of time series. Then we propose our QARNN model, utlizing the architecture of QRNN model.

3.1. Quantile Autoregression

The AR model plays an important role in time series analysis. Following the idea of the AR, QAR model is developed by Koenker & Xiao (2006) under the framework of quantile regression. The QAR model can be written as

$$Q_{y_t}(\tau|\mathbb{F}_t) = \alpha_0(\tau) + \alpha_1(\tau)y_{t-1} + \cdots + \alpha_p(\tau)y_{t-p} = \boldsymbol{z}_t'\boldsymbol{\theta}(\tau), \tag{2}$$

where $0 < \tau < 1$, $\boldsymbol{z}_t = (1, y_{t-1}, \cdots, y_{t-p})'$ is a predictor vector, $\boldsymbol{\theta}(\tau) = (\alpha_0(\tau), \alpha_1(\tau), \cdots, \alpha_p(\tau))'$ is a coefficient vector to be estimated, and $\mathbb{F}$ is the σ-field generated by $\{y_{t-1}, \cdots, y_{t-p}\}$.

Then, the QAR model in (2) is fitted by

$$\hat{\boldsymbol{\theta}}(\tau) = \arg\min_{\boldsymbol{\theta}\in\mathbb{R}^{p+1}} \frac{1}{T-T_0}\sum_{t=T_0+1}^{T} \rho_\tau(y_t - \boldsymbol{z}_t'\boldsymbol{\theta}), \tag{3}$$

where $T_0 = p$, $\rho_\tau(u)$ is a check function given by equation (1).

Given $\hat{\boldsymbol{\theta}}(\tau)$, the τ th quantile of y_t conditional on $\mathbb{F}_t$ can be estimated by

$$\hat{Q}_{y_t}(\tau|\mathbb{F}_t) = \boldsymbol{z}_t'\hat{\boldsymbol{\theta}}(\tau). \tag{4}$$

3.2. Quantile Autoregressive Distributed Lag

The QAR model is generalized by Galvao Jr et al. (2013) to the quantile autoregressive distributed lag (QARDL) model in an autoregressive dynamic framework. By adding exogenous covariates to the QAR model (2), QARDL is given by

$$Q_{y_t}(\tau|\mathbb{F}_t) = \alpha_0(\tau) + \sum_{j=1}^{p}\alpha_j(\tau)y_{t-j} + \sum_{l=0}^{q}\boldsymbol{x}_{t-1}'\boldsymbol{\beta}_l(\tau) = \boldsymbol{z}_t'\boldsymbol{\theta}(\tau), \tag{5}$$

where $\boldsymbol{z}_t = (1, y_{t-1}, \cdots, y_{t-p}, \boldsymbol{x}_t, \cdots, \boldsymbol{x}_{t-q})'$, $\boldsymbol{\theta}(\tau) = (\alpha_0(\tau), \alpha_1(\tau), \cdots, \alpha_p(\tau), \boldsymbol{\beta}_0'(\tau), \boldsymbol{\beta}_1'(\tau), \cdots, \boldsymbol{\beta}_q'(\tau))'$, and $\mathbb{F}_t$ is the σ-field generated by $\{y_s, \boldsymbol{x}_s; s \le t\}$.

The estimation procedure of QARDL is also based on standard linear quantile regression and is similar to equation (3) except for $T_0 = \max\{p, q\}$. In QARDL the choice of p and q is very important and can be implemented using Akaike information criterion (AIC) or Bayesian information criterion (BIC).

3.3. Conditional autoregressive value at risk

An early quantile autoregressive framework is proposed by Engle & Manganelli (2004) which focuses on value-at-risk (VaR) estimation via autoregressive process. The model is called CAViaR and has the following generic specification

$$Q_{y_t}(\tau|\mathbb{F}_t) = \alpha_0(\tau) + \sum_{j=1}^{p}\alpha_j(\tau)Q_{y_{t-j}}(\tau) + \sum_{l=0}^{q} f(\boldsymbol{x}_{t-1}')\boldsymbol{\beta}_l(\tau) = \boldsymbol{z}_t'\boldsymbol{\theta}(\tau), \tag{6}$$

where f is a nonlinear function, $z_t = (1, Q_{y_{t-1}}(\tau), \cdots, Q_{y_{t-p}}(\tau), \boldsymbol{x}_t, \cdots, \boldsymbol{x}_{t-q})'$, $\boldsymbol{\theta}(\tau) = (\alpha_0(\tau), \alpha_1(\tau), \cdots, \alpha_p(\tau), \boldsymbol{\beta}_0'(\tau), \boldsymbol{\beta}_1'(\tau), \cdots, \boldsymbol{\beta}_q'(\tau))$, and $\mathbb{F}$ is the σ-field generated by $\{\boldsymbol{x}_s; s \le t\}$. The autoregressive terms $\alpha_j(\tau)Q_{y_{t-j}}(\tau)$ ensure that the conditional quantile changes smoothly over time. The role of $f(\boldsymbol{x}_{t-i}')$ is to link the conditional quantile to observable variables in the information set $\mathbb{F}_t$. Comparing to QAR and

QARDL models, which apply QR to the conventional linear time series depending on past values of the response and model a linear conditional quantile function, CAViaR models the quantile functions directly as an autoregressive process.

There are four commonly used CAViaR models defined as follows, with CAViaR-1, CAViaR-2, CAViaR-3, and CAViaR-4 denoting CAViaR Symmetric Absolute Value, CAViaR-Asymmetric Slope, CAViaR-Indirect GARCH(1,1) model and CAViaR-Adaptive, separately.

$$\begin{aligned}
\text{CAViaR-1} : Q_t(\theta) &= \omega + \alpha Q_{t-1}(\theta) + \beta|y_{t-1}|, && (7)\\
\text{CAViaR-2} : Q_t(\theta) &= \omega + \alpha Q_{t-1}(\theta) + \beta_1(y_{t-1})^+ + \beta_2(y_{t-1})^-, && (8)\\
\text{CAViaR-3} : Q_t(\theta) &= (1 - 2I(\theta < 0.5))(\omega + \alpha Q_{t-1}(\theta)^2 + \beta y_{t-1}^2)^{1/2}, && (9)\\
\text{CAViaR-4} : Q_t(\theta) &= Q_{t-1}(\theta) + \alpha[\theta - I(y_{t-1} < Q_{t-1}(\theta)], && (10)
\end{aligned}$$

where $(y_{t-1})^+ = \max(y_{t-1}, 0)$ and $(y_{t-1})^- = -\min(y_{t-1}, 0)$.

The parameters of the CAViaR model can be estimated by equation (3) through recursive forms, and the conditional quantile functions can be recursively estimated using equation (4).

3.4. Quantile Autoregression Neural Network

Inspired by the above, we develop the QARNN model by combining QAR method with an ANN structure. We first present the model specification and then discuss its estimation procedure. In addition, two criteria are used for parameter selection in the QARNN model.

3.4.1. Model Specification

Inspired by the QRNN of Cannon (2011) which implements a flexible nonlinear quantile regression for cross-section data without prior specification of the form of the relationships, we consider a nonlinear quantile regression in the context of time series and develop the QARNN model.

We extend CAViaR model to a general dependent structure as

$$Q_{y_t}(\tau|\mathbb{F}_t) = f(Q_{y_{t-1}}(\tau), \cdots, Q_{y_{t-p}}(\tau), \boldsymbol{x}_t, \boldsymbol{x}_{t-1}, \cdots, \boldsymbol{x}_{t-q}; \hat{\boldsymbol{\theta}}(\tau)), \qquad (11)$$

where f is an arbitrary nonlinear function and we consider the use of ANN to simulate the nonlinear structure in f. As the recursive terms need to be initialized for training the neural network at the beginning, we compute the empirical quantile corresponding to each τ using the first m observations, specifically, m is often chosen as 1/10 of the sample size. Given predictors $Q_{y_{t-1}}(\tau), \cdots, Q_{y_{t-p}}(\tau), \boldsymbol{x}_t, \boldsymbol{x}_{t-1}, \cdots, \boldsymbol{x}_{t-q}$

and a predicted $Q_{y_t}(\tau)$, outputs from a QARNN, which is pictured in Fig.1, are calculated as follows. First, an output from the k-th hidden layer node $g_{k,t}(\tau)$ is given by applying a sigmoid transfer function to the inner product between the predictors and the hidden layer weights $w_{pk}^{(h)}$ plus the hidden layer bias $b_k^{(h)}$

$$g_{k,t}(\tau) = f^{(h)}\left(\sum_{i=1}^{p} w_{ik}^{(h)}(\tau) Q_{y_{i-1}}(\tau) + \sum_{j=0}^{q} w_{j+p+1,k}^{(h)}(\tau)\boldsymbol{x}_{t-j} + b_k^{(h)}(\tau)\right), \tag{12}$$

where $\boldsymbol{w}^{(h)} = (w_{1k}^{(h)}, w_{2k}^{(h)}, \cdots, w_{p+q+1,k}^{(h)})$ is a weight vector of hidden layer, $\boldsymbol{b}^{(h)}(\tau) = (b_1^{(h)}(\tau), b_2^{(h)}(\tau), \cdots, b_K^{(h)})(\tau)$ is a bias vector of hidden layer, and $f^{(h)}$ denotes a sigmoid transfer function such as the hyperbolic tangent. Second, an estimate of the conditional τth conditional quantile of response is then given by

$$Q_{y_t}(\tau) = f^{(o)}\left(\sum_{k=1}^{K} w_k^{(o)} g_{k,t}(\tau) + b^{(o)}(\tau)\right) \tag{13}$$

where $w_k^{(o)}$ are the output layer weights, $b^{(o)}(\tau)$ is the output layer bias, and $f^{(o)}$ is the output layer transfer function such as the identity function.

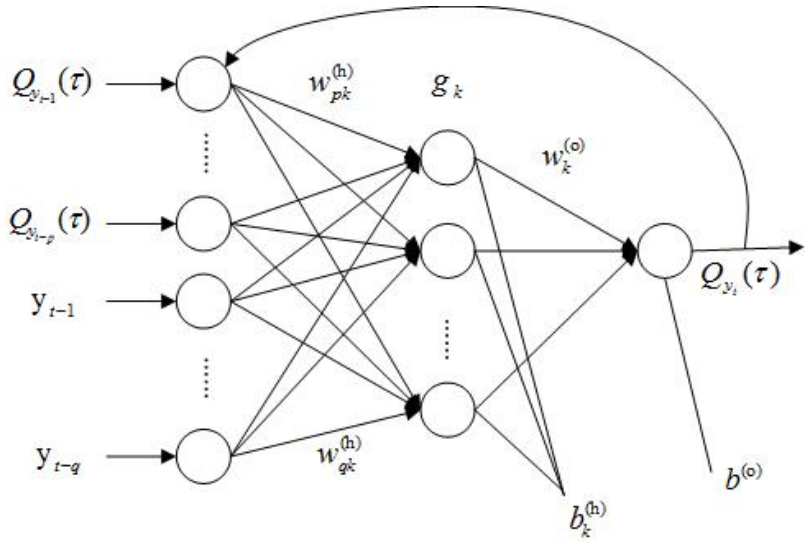


Figure 1: **Schematic diagram showing a QARNN model**

3.4.2. Model Estimation

Outputs in QARNN are the estimates of conditional regression quantiles. In order to estimate (or train) the QARNN model, we define the quantile regression

error (or cost) function as

$$C(\tau) = \frac{1}{T - T_0} \sum_{t=T_0+1}^{T} \rho_\tau(y_t - \hat{Q}_{y_t}(\tau)) \tag{14}$$

where $T_0 = \max\{p, q\}$.

In general, parameters in the QARNN model are trained through gradient based nonlinear optimization algorithm. The check function $\rho_\tau(u)$ defined in equation (1) is, however, not differentiable everywhere, as its derivative is invalid at the origin. Following the suggestion by Cannon (2011) for QRNN, the Huber norm, which can provide a smooth transition between absolute and squared errors around the origin

$$h(u) = \begin{cases} \dfrac{u^2}{2\varepsilon}, & 0 \le |u| \le \varepsilon \\ |u| - \dfrac{\varepsilon}{2}, & |u| > \varepsilon \end{cases} \tag{15}$$

is used here to approximate the check function by

$$\rho_\tau^{(a)}(u) = \begin{cases} \tau h(u), & u \ge 0 \\ (\tau - 1)h(u), & u < 0 \end{cases} \tag{16}$$

where ε is a given threshold magnitude, with fixed schedule of values ($\varepsilon = 2^i$ for $i = -8, -9, \cdots, -32$), which is used by default in the R package "qrnn". Cannon (2011) developed this package to fit a QRNN with optional left censoring using a variant of the finite smoothing algorithm. Using the approximate check function, we revise the quantile regression error and get the approximate error function

$$C^{(a)}(\tau) = \frac{1}{T - T_0} \sum_{t=T_0+1}^{T} \rho_\tau^{(a)}(y_t - \hat{Q}_{y_t}(\tau)). \tag{17}$$

Therefore, we can estimate the QARNN model by optimizing the approximate error function via standard gradient based optimization algorithms. In the whole optimization procedure, ε in the Huber norm is first set to a relatively large value, but a smaller value is required during the second optimization run. Moreover, ε decreases with the process, repeating until ε goes to zero and the algorithm converges.

3.4.3. Model Selection

The QARNN model is flexible and can represent nonlinear predictor-predicted relationships, including those involving interactions between predictors. The level of

model complexity is usually controlled by the size (p, q) of predictors and the number of hidden layer nodes K. A model that is too complex may result in over-fitting, but this can be avoided by penalizing large weights in the input-hidden layer by adding a quadratic penalty terms to the approximate error function

$$C^{pen}(\tau) = \frac{1}{T-T_0}\sum_{t=T_0+1}^{T}\rho_\tau^{(a)}(y_t - \hat{Q}_{y_t}(\tau)) + \lambda\frac{1}{MK}\sum_{m=1}^{M}\sum_{k=1}^{K}\left(w_{mk}^{(h)}(\tau)\right)^2, \tag{18}$$

where λ is a positive constant that controls the trade-off between the error and the weight decay term.

An important issue in QARNN modeling is to find appropriate (p, q) and K. To this end, we use the AIC and the generalized approximate cross validation (GACV) criteria defined as

$$AIC(p, q, K; \tau) = \ln\left(\frac{1}{T-T_0}\sum_{t=T_0+1}^{T}\rho_\tau^{(a)}(y_t - \hat{Q}_{y_t}(\tau))\right) + \frac{(p+q+1)K+1}{T}, \tag{19}$$

$$GACV(p, q, K; \tau) = \frac{\sum_{t=T_0+1}^{T}\rho_\tau^{(a)}(y_t - \hat{Q}_{y_t}(\tau))}{T - df} \tag{20}$$

where $(p+q+1)K+1$ denotes the number of parameters for AIC in the QARNN model, while df for GACV is a measure of the effective dimensionality of the fitted model, and can be estimated by the SURE divergence formula $\sum \partial\hat{Q}_{y_t}(\tau)/\partial y_t$ proposed in Yuan (2006). Presumably, the number of parameters in AIC would no longer be valid if weight decay regularization is used to avoid overfitting. Therefore, we replace the number of parameters for AIC: $(p+q+1)K+1$ with the df estimate used in equation (20). In our particular case, grid search method can be used to minimize AIC and GACV, where p, q, K take values from 1 to 10 leading to 1000 different combinations.

4. Numerical experiments

In terms of implementing the proposed method, we focus our application to estimate and predict VaR, which is defined as the maximum potential loss on the portfolio over a prescribed holding period with a confidence level. From the perspective of statistics, the $100(1-\tau)\%$ VaR of a portfolio Y, a standard tool in financial risk management, equals the negative of τ-th quantile of the distribution of Y, i.e. $VaR_Y(1-\tau) = -Q_Y(\tau)$. QR methods are, therefore, often used to evaluate VaR. In particular, QR has been successfully applied by Taylor (1999) to estimate the

tails of the distribution of multi-period financial returns or VaRs. Recently, there are many articles dedicated to the estimation of VaR (Abad et al., 2014). The existing methodologies, such as some asymmetric extensions of CAViaR method(Engle & Manganelli, 2004) and the parametric method under the skewed and fat-tail distributions lead to promising results, especially when the assumption that the standardised returns is iid is abandoned and the conditional high-order moments are considered to be time-varying.

In this section we conduct both Monte Carlo simulation studies and real applications to illustrate the performance of the proposed QARNN model for VaR evaluation. We mainly compare the QARNN model with some classical models and a state of arts approach, including CAViaR model, QRNN model, ARMA-APARCH model, Riskmetric model, GARCH-EVT model and serial pair-copula constructions (PCC) model.

4.1. Classical VaR estimation methods

A vast array of VaR estimation method have been proposed. We recall two more classical VaR estimation methods, RiskMetric model and GARCH-EVT model and one state of arts VaR evaluation approach serial PCC model, proposed in Righi & Ceretta (2015) for further comparison.

4.1.1. RiskMetric model

Apart from the aforementioned methods, RiskMetric methodology to VaR calculation developed by Morgan (1996) has been widely used in financial risk management. Let y_t denote a portfolio return with the distribution function $F_Y(y) = P(Y \leq y)$. According to the definition of VaR, the VaR of y_t with the confidence level $100 \times (1-\theta)\%$ is the negative of the θ-th quantile of F_Y: $VaR_Y(1-\theta) = -Q_Y(\theta)$. In application, RiskMetrics assumes that y_t follows the conditional normal distribution $y_t|\mathcal{F}_{t-1} \sim N(\mu_t, \sigma_t^2)$ and can be described as follows

$$\begin{cases} y_t = \mu_t + \epsilon_t, \quad \epsilon_t = \sigma_t z_t \\ \sigma_t^2 = \gamma\sigma_{t-1}^2 + (1-\gamma)\epsilon_{t-1}^2, \end{cases} \tag{21}$$

where $\mathcal{F}_{t-1}$ denotes the information set available until time t, μ_t is the conditional mean, σ_t is the conditional variance that evolves over time according to the EWMA model with a weighting parameter γ (often taking the value of 0.94), ϵ_i is a random disturbance term, and the residual sequence z_t is usually set to follow the standard normal distribution.

Under the assumptions in RiskMetrics, financial risk with the confidence level $100 \times (1-\theta)\%$ can be estimated by

$$VaR_t(1-\theta) = -\mu_t - \sigma_t z(\theta) \tag{22}$$

where $z_\theta = F_z^{-1}(\theta)$ is the θth quantile of the standard normal distribution.

4.1.2. GARCH-EVT model

The GARCH-EVT model proposed by McNeil & Frey (2000) and Allen et al. (2013) combines GARCH models to estimate the current volatility and EVT to estimate the tail of the innovation distribution of the GARCH model. The model has been widely used to estimate extreme financial risk.

The EWMA model is a special case of a generalized autoregressive conditional heteroscedasticity model proposed by Bollerslev (1986) with the GARCH(1,1) form

$$\begin{cases} y_t = \mu_t + \epsilon_t, \quad \epsilon_t = \sigma_t z_t, \\ \sigma_t^2 = \omega + \alpha_1 \epsilon_{t-1}^2 + \beta_1 \sigma_{t-1}^2, \end{cases} \tag{23}$$

where $z_t \sim iid.N(0,1)$ and $\omega, \alpha_1, \beta_1$ are parameters to be estimated. We set the conditions on parameters $\omega > 0, \alpha_1 > 0, \beta_1 > 0$ and $\alpha_1 + \beta_1 < 1$, to ensure strong positivity and stationarity of the conditional variance.

If F represents the distribution function of the residual series z_t, the conditional excess distribution function can be obtained as follows

$$F_u(y) = \Pr(z - u \le y | z > u) = \frac{F(z) - F(u)}{1 - F(u)}, \tag{24}$$

where u is a given threshold, $0 \le y < z_F - u$, $z_F < \infty$ is the right endpoint of F and $y = z - u$. For a large class of underlying distribution functions F, the conditional excess distribution function $F_u(y)$, for large u, is well approximated by the generalized Pareto distribution (GPD)

$$G_{\xi,\sigma}(y) = \begin{cases} 1 - \left(1 + \dfrac{\xi y}{\sigma}\right)^{-1/\xi}, & \text{if } \xi \neq 0 \\ 1 - \exp^{-y/\sigma}, & \text{if } \xi = 0, \end{cases} \tag{25}$$

where ξ and σ are called the shape and scale parameters, respectively. From equation (24), we have

$$F(z) = (1 - F(u))\, F_u(y) + F(u). \tag{26}$$

If we use the random proportion of the data $(n - n_u)/n$ to estimate $F(u)$ and use $G_{\xi,\sigma}(y)$ to approximate $F_u(y)$, we get the tail estimator

$$F(z) = 1 - \frac{n_u}{n}\left(1 + \frac{\xi}{\sigma}(z-u)\right)^{-1/\xi}, \tag{27}$$

for $z > u$. Here, n_u is the number of observations above u in all n observations.

The negative inverse of (27) with a probability θ gives the VaR

$$VaR_z(1-\theta) = -u - \frac{\sigma}{\xi}\left[\left(\frac{n}{n_u}(1-\theta)\right)^{-\xi} - 1\right] \tag{28}$$

For $\xi < 1$, borrowing the idea of equation (22), the financial risk of y_t with the confidence level $100 \times (1-\theta)\%$ can be further estimated by

$$VaR_t(1-\theta) = -\mu_t - \sigma_t[-VaR_z(1-\theta)] \tag{29}$$

4.1.3. PCC model

Righi & Ceretta (2015) used a serial dependence structure of financial assets based on PCC to estimate risk measures. This PCC structure considers dependence with past observations isolating the effect for other lags. Consider that y has the distribution function F, aligned with the procedures elaborated in section 3 in Righi & Ceretta (2015), financial risk of with the confidence level $100 \times (1-\theta)\%$ can be estimated by

$$VaR(\theta) = q_\theta(y) = inf\{q : F(q) \geq \alpha\} \tag{30}$$

$$ES(\theta) = E\left[y | y < VaR_t(\theta) = q_\theta(y)\right] = \theta^{-1}\int_0^\theta q_s(y)ds, \tag{31}$$

Consider Y has a marginal specification,

$$y_t = \mu_t + \sigma_t z_t, \tag{32}$$

where μ_t is the conditional mean, σ_t is the conditional variance that evolves over time, z_t represents the innovations white noise series. Following the specifications in Righi & Ceretta (2015), we estimate the ARMA(m,n)-GARCH(p,q) models with normal innovations for the marginal.

Therefore, for the parametric approach based on marginal models, VaR and ES can be calculated as follows:

$$VaR_t(\theta) = \mu_t + N^{-1}(\theta)\sigma_t, \tag{33}$$

$$ES_t(\theta) = \mu_t + \sigma_t \left[\frac{1}{\alpha} \int_0^{\theta} yN^{-1}(\theta)dy \right], \tag{34}$$

where, μ_t and σ_t are the conditional mean and standard deviation for each asset in period t, respectively; $N^{-1}(\theta)$ is the inverse of normal distribution N on each asset.

4.2. Monte Carlo simulations

4.2.1. Data Generation

Autoregressive conditional heteroscedasticity (ARCH) of Engle (1982) and generalized ARCH (GARCH) of Bollerslev (1986) provide effective ways to describe volatility of time series and have been widely used in financial risk management. In subsequent literature, many (G)ARCH-type models are proposed to explore stylized facts in financial markets [see Ali (2013) for details]. In particular, the asymmetric power ARCH (APARCH) model, proposed by Ding et al. (1993) and subsequently used in Ghourabi et al. (2015), is capable of representing a general class of models that include both ARCH and GARCH models. In addition, ARMA model with APARCH errors is also popular for time series analysis [see Wurtz et al. (2006) for details]. A general ARMA(m, n)-APARCH(p, q) specification, which refers as simulation 1, might be expressed as following

$$\begin{cases} y_t = c + \sum_{i=1}^{m} \phi_i y_{t-i} + \varepsilon_t - \sum_{j=1}^{n} \varphi_j \varepsilon_{t-j}, \quad \varepsilon_t = \eta_t \sigma_t \\ \sigma_t^{\delta} = \omega + \sum_{i=1}^{p} \alpha_i \left(|\varepsilon_{t-1}| - \gamma_i \varepsilon_{t-i} \right)^{\delta} + \sum_{j=1}^{q} \beta_j \sigma_{t-j}^{\delta} \end{cases} \tag{35}$$

where $\delta > 0$ and $-1 < \gamma_i < 1$. The random error η_t often uses standard normal distribution, skewed normal distribution, Student-t distribution and skewed Student-t distribution.

As one may consider, the performance measure (35) would give only an estimate of the performance, as the true conditional quantiles are not known. In consideration of a "ground truth" from our proposed QARNN model, we account for another simulation data via a synthetic time series sample, where a theoretical value of the conditional quantiles can be derived. Following Galvao Jr et al. (2011), we consider a baseline linear location-scale two-regime switching self-exciting threshold autoregressive model (simulation 2):

$$\begin{cases} y_t = 0.05 + 0.05 y_{t-1} - y_{t-1}\eta_t, & y_{t-1} \leq \gamma_0, \\ y_t = 0.05 + 0.05 y_{t-1} + y_{t-1}\eta_t, & y_{t-1} \geq \gamma_0, \end{cases} \tag{36}$$

where $\gamma_0 = 0$ and $\eta_t \sim N(0,1)$ for simplicity but without loss of generality. Aligning with Galvao Jr et al. (2011), the derived true value of conditional τ-quantile process is:

$$\begin{cases} Q_{y_t}(\tau|y_{t-1} \leq 0) = 0.05 + 0.05(0.5 - F_u^{-1}(\tau))y_{t-1}, \\ Q_{y_t}(\tau|y_{t-1} \leq 0) = 0.05 + 0.05(0.5 + F_u^{-1}(\tau))y_{t-1}, \end{cases} \tag{37}$$

In simulation, we first generate y_t via ARMA(1,0)-APARCH(1,1) model defined in (35), which is referred as simualtion 1, with given parameters: $c = 0.001, \phi_1 = 0.05, \omega = 0, \alpha_1 = 0.05, \beta_1 = 0.8, \gamma_1 = 0, \delta = 1.8$. Further, we evaluate the model defined in (36) by conducting Monte Carlo experiments. Specifically, we generate both simulation data with sample size $T = 1500$, in which the first 1000 sample are used as training data for model estimation and the remaining 500 sample are left as test data for the out-of-sample evaluation. Moreover, we generate four types of data sets for each model in simulation 1, named as s1.norm, s1.snorm, s1.std and s1.sstd, respectively, using the listed four different random errors above. We name data generated via model defined in (36) as s2.norm, for clarity.

4.2.2. VaR Evaluation

We estimate ARMA-APARCH model, CAViaR model, QRNN model, Riskmetric model, GARCH-EVT and our QARNN model, respectively. To be convenient for explanation, we modify the proposed model with predictor $|y_{t-j}|$ rather than y_{t-j} for $j = 1, 2, \cdots, q$, which is in accordance with CAViaR settings, and we set the penalty parameter λ as 0 for simplicity. We consider three quantiles: $\tau = 1\%, 5\%, 10\%$. The optimal number of predictors and hidden nodes at each quantile τ and each stock index return is selected through AIC and GACV criteria, which are either $p = 2, q = 1, k = 3$ or $p = 1, q = 1, k = 3$, shown in Table 1. This implies that we do not need a very complicated neural network structure for practical use. To further demonstrate the stability of the QARNN model, we implement it for both cases of $p = 2, q = 1, k = 3$ and $p = 1, q = 1, k = 3$. Without loss of generality, we assume QRNN the same structure as our QARNN, but excluding recursive terms. Following the optimal hidden nodes for QARNN and balancing both learning and generalization errors comprehensively, we let the number of hidden nodes for QRNN equal to 3.

Table 1: **The optimal parameters in QARNN model determined by AIC and GACV for simulation data.**

	$\tau = 1\%$			$\tau = 5\%$			$\tau = 10\%$		
Data set	p, q, K	GACV	AIC	p, q, K	GACV	AIC	p, q, K	GACV	AIC
s1.norm*	(1,1,3)	2.212	-15.324	(2,1,3)	4.385	-14.640	(2,1,3)	5.704	-14.377
s1.snorm*	(1,1,3)	1.635	-15.430	(2,1,3)	4.019	-14.727	(2,1,3)	5.292	-14.452
s1.std*	(2,1,3)	1.180	-15.952	(2,1,3)	3.561	-14.848	(2,1,3)	5.220	-14.466
s1.sstd*	(1,1,3)	1.185	-15.948	(1,1,3)	3.543	-14.375	(1,1,3)	5.414	-13.687
s2.norm	(2,1,3)	0.923	-0.080	(2,1,3)	3.561	1.270	(1,1,3)	5.935	1.780

NOTE: (1) * denotes GACV $\times 10^{-7}$; (2)norm, snorm, std, sstd denote the generated data with random error η_t as standard normal distribution, skewed normal distribution, Student-t distribution and skewed Student-t distribution, respectively.

To evaluate the accuracy of VaR estimation, we use the likelihood ratio (LR) test of Kupiec (2006) and the independence and conditional coverage test of Christoffersen (1998) to backtesting. To distinguish these two tests, we name them as uc.LR and cc.LR, respectively. Define the observed proportion of failures as

$$p = \frac{N}{T} = \frac{1}{T}\sum_{t=1}^{T} I(-y_t > VaR_t(1-\theta)). \tag{38}$$

The ideas of both uc.LR test and cc.LR test are to check whether $H_0 : p = p^*$, where $p^* = \theta$ denotes the expected probability of failures. Under the null hypothesis, the corresponding uc.LR statistic

$$uc.LR = 2\ln\left[(1-p)^{T-N}p^N\right] - 2\ln\left[(1-\theta)^{T-N}\theta^N\right], \tag{39}$$

is asymptotically $\chi^2(1)$ distributed.

Based on the uc.LR test, Christoffersen (1998) proposed the cc.LR test which is extended to include a separate statistic for independence of exceptions. The test defines an indicator variable

$$I_t = \begin{cases} 0 & \text{if no violation occurs} \\ 1 & \text{if violation occurs} \end{cases}$$

Following the definition, the test statistic for independence of exceptions is

$$ind.LR = -2\ln\left[(1-\pi)^{n_{00}+n_{10}}\pi^{n_{01}+n_{11}}\right] - 2\ln\left[(1-\pi_0)^{n_{00}}\pi_0^{n_{01}}(1-\pi_1)^{n_{10}}\pi_1^{n_{11}}\right] \tag{40}$$

where $n_{ij}, i, j = 0, 1$ denote the number of times that $I_{t-1} = i, I_t = j$ occurs; $\pi_i, i = 0, 1$ represent the probability that a violation occurs conditional on the previous day, that is $\pi_0 = \frac{n_{01}}{n_{00}+n_{01}}$, $\pi_1 = \frac{n_{10}}{n_{00}+n_{11}}$ and $\pi = \frac{n_{01}+n_{11}}{n_{00}+n_{01}+n_{10}+n_{11}}$. Under the null hypothesis, the corresponding conditional coverage cc.LR statistic:

$$cc.LR = uc.LR + ind.LR \tag{41}$$

is asymptotically $\chi^2(2)$ distributed (Christoffersen, 1998).

Furthermore, in order to demonstrate that our QARNN model outperforms the aforementioned available approaches, three measurements are employed to evaluate the prediction accuracy for simulation 2, namely, the empirical quantile risk (Risk), the root mean square error (RMSE) and the mean absolute error (MAE).They are defined as:

$$Risk(\tau) = \frac{1}{n}\sum_{i=1}^{n} \rho_\tau \left(Q_{y_i}(\tau|x) - \hat{Q}_{y_i}(\tau|x) \right), \tag{42}$$

$$RMSE(\tau) = \sqrt{\frac{1}{n}\sum_{i=1}^{n} \left(Q_{y_i}(\tau|x) - \hat{Q}_{y_i}(\tau|x) \right)^2}, \tag{43}$$

$$MAE(\tau) = \frac{1}{n}\sum_{i=1}^{n} \left| \left(Q_{y_i}(\tau|x) - \hat{Q}_{y_i}(\tau|x) \right) \right|, \tag{44}$$

where $\hat{Q}_{y_i}(\tau|x)$ is the prediction of the true quantile $Q_{y_i}(\tau|x)$.

4.2.3. Performance results

Table 2 presents the observed proportion of failures of each method for the two simulation data at three confidence levels 90%, 95%, and 99% corresponding to quantiles 10%, 5%, and 1% respectively. The last two columns of Table 2 report the value of NS1 and NS2, which represent a count for the number of uc.LR test and cc.LR test separately for which the null is rejected at 5% significance level. The closer the proportion of failures to 10%, 5% and 1% under 90%, 95% and 99% confidence level separately, the better the model is.

Table 2: **Out-of-sample VaR backtesting on simulation data.**

τ	Models	s1.norm			s1.snorm			s1.std			s1.sstd			s2.norm			NS1	NS2
		F	P1	P2	F	P1	P2	F	P1	P2	F	P1	P2	F	P1	P2		
1%	Riskmetric	1.23	0.32	0.22	2.53	0	0	0.87	0.60	0.41	1.60	0.03	0.02	0.73	0.28	0	2	3
	Garch-POT	0.80	0.64	0.33	0.06	0.33	0	1.40	0.14	0.07	0.60	0.33	0	2.20	0	0	2	3
	CAViaR1	0.93	0.78	0.11	1.00	0.99	0.58	1.40	0.14	0.10	0.93	0.78	0.41	0.67	0.17	0.02	0	1
	CAViaR2	1.00	0.99	0.30	1.14	0.59	0.42	1.40	0.14	0.07	0.93	0.78	0.23	0.73	0.28	0.03	0	1
	CAViaR3	1.13	0.62	0.42	0.93	0.78	0.56	1.07	0.79	0.52	1.14	0.59	0.12	0.60	0.09	0.01	0	1
	CAViaR4	0.93	0.78	0.01	1.00	0.99	0.77	1.13	0.62	0.42	1.13	0.62	0.44	0.27	0	0	1	2
	QRNN	1.20	0.45	0.32	1.13	0.62	0.01	0.93	0.78	0.41	0.93	0.78	0.83	1.13	0.61	0.37	0	1
	APARCH	1.27	0.31	0.22	2.20	0	0	1.14	0.59	0.41	1.55	0.05	0.05	5.60	0	0	2	2
	PCC	0.80	0.64	0.33	0.80	0.64	0.33	1.14	0.59	0.41	0.60	0.33	0.15	0	0	0	1	1
	QARNN-1	1.00	0.99	0.98	1.00	0.99	0.89	1.27	0.31	0.27	1.00	1.00	0.57	1.20	0.45	0.34	0	0
	QARNN-2	0.93	0.78	0.80	0.93	0.78	0.60	1.20	0.45	0.61	0.93	0.78	0.79	0.93	0.80	0.55	0	0
5%	Riskmetric	3.00	0.03	0	5.60	0.55	0.78	3.20	0.05	0	5.40	0.69	0.42	1.13	0	0	3	3
	Garch-EVT	3.80	0.03	0.06	4.60	0.50	0.77	4.20	0.15	0.34	5.54	0.35	0.35	4.55	0.42	0.30	1	0
	CAViaR1	4.86	0.80	0.57	4.67	0.55	0.33	4.60	0.47	0.30	4.86	0.80	0.51	4.60	0.48	0.04	0	1
	CAViaR2	5.70	0.22	0.51	4.74	0.64	0.58	4.87	0.82	0.91	4.80	0.72	0.25	4.47	0.34	0.17	0	0
	CAViaR3	5.84	0.15	0.03	4.55	0.42	0.22	4.67	0.55	0.33	4.06	0.09	0.04	4.87	0.82	0	0	3
	CAViaR4	4.47	0.34	0.01	4.55	0.42	0.30	4.93	0.90	0.54	6.07	0.07	0.02	0.73	0	0	1	3
	QRNN	4.93	0.90	0.55	5.07	0.90	0.55	4.86	0.80	0.77	4.80	0.72	0.51	4.34	0.23	0.51	0	0
	APARCH	4.47	0.34	0.11	4.55	0.42	0.30	6.80	0	0	6.47	0.01	0	6.20	0.23	0	2	3
	PCC	2.80	0.01	0	6.00	0.32	0.42	5.40	0.69	0.41	5.80	0.43	0.70	0	0	0	2	2
	QARNN-1	4.94	0.91	0.83	5.27	0.63	0.94	4.60	0.47	0.97	5.00	1.00	0.63	5.27	0.63	0.94	0	0
	QARNN-2	5.00	0.99	0.88	4.74	0.64	0.94	4.86	0.80	0.97	5.00	1.00	0.89	5.07	0.90	0.56	0	0
10%	Riskmetric	6.67	0	0	8.73	0.10	0.23	7.67	0	0.01	10.27	0.72	0.50	1.40	0	0	3	3
	Garch-EVT	5.40	0	0	8.80	0.36	0.66	9.20	0.55	0.26	9.00	0.45	0.39	8.80	0.36	0.66	1	1
	CAViaR1	10.80	0.31	0.26	9.87	0.87	0.33	9.67	0.67	0.44	9.87	0.87	0.65	9.07	0.23	0.02	0	1
	CAViaR2	9.13	0.26	0.21	9.87	0.87	0.45	9.15	0.27	0.11	9.93	0.93	0.70	9.54	0.55	0.08	0	0
	CAViaR3	11.90	0.02	0	9.53	0.54	0.29	9.60	0.60	0.31	9.41	0.44	0.38	10.07	0.93	0.34	1	1
	CAViaR4	9.00	0.19	0.33	9.87	0.87	0.45	11.1	0.16	0	9.03	0.20	0.52	3.54	0	0	1	2
	QRNN	9.94	0.94	0.99	9.93	0.93	0.54	9.60	0.60	0.41	9.93	0.93	0.70	10.14	0.86	0.77	0	0
	APARCH	12.70	0	0	9.27	0.34	0.50	13.60	0	0	13.20	0	0	6.80	0	0	4	4
	PCC	6.40	0.02	0.02	9.00	0.45	0.57	11.40	0.31	0.47	11.40	0.31	0.58	0	0	0	2	2
	QARNN-1	9.21	0.30	0.57	10.10	0.89	0.97	9.53	0.54	0.66	10.01	0.99	0.70	10.47	0.54	0.56	0	0
	QARNN-2	9.21	0.30	0.51	10.00	0.99	0.51	10.01	0.99	0.70	9.92	0.92	0.63	9.47	0.50	0.72	0	0

NOTE: (1)norm, snorm, std, sstd denote the generated data with random error η_t as standard normal distribution, skewed normal distribution, Student-t distribution and skewed Student-t distribution, respectively. (2) CAViaR-1, CAViaR-2, CAViaR-3, and CAViaR-4 denote CAViaR Symmetric Absolute Value, CAViaR-Asymmetric Slope, CAViaR-Indirect GARCH(1,1) model and CAViaR-Adaptive; (3) F denotes proportion of failures while P1 and P2 stand for p-values obtained via uc.LR test and cc.LR test separately; (4)NS1 and NS2 represent a count for the number of uc.LR test and cc.LR test separately for which the null is rejected at 5% significance level.

It is apparent that both Riskmetric model and APARCH(1,1) model show relatively poor performance compared to the others at all the chosen quantiles. It may be noticed that CAViaR-Indirect GARCH(1,1) is performing slightly better than our QARNN model yet only with s1.std at $\tau = 0.01$ and s2.norm at $\tau = 0.1$. However, under a stricter VaR conditional coverage test, 90% VaR and 95% VaR derived from CAViaR-Indirect GARCH(1,1) model and CAViaR-Adaptive model are generally poor, with 3 NSs at $\tau = 5\%$ and 2 NSs at $\tau = 10\%$. The state of art approach PCC model performs inferior than our QARNN model under all three quantiles, with 1NS, 2NSs and 2NSs, respectively. Moreover, although other existing models, such as QRNN, provide relatively significant estimates, the simulation results indicate that these models are still inferior to our QARNN model. Table 3 summarizes the best model for VaR evaluation on both simulation data. It is obvious that our QARNN model is recommended most often as the best model for different cases.

Table 3: **The best models for VaR evaluation on simulation data.**

Test	τ	Confidence level (%)	Best model				
			s1.norm	s1.snorm	s1.std	s1.sstd	s2.norm
uc.LR	0.01	99	QARNN-1	QARNN-1	CAViaR-3	QARNN-1	QARNN-2
	0.05	95	QARNN-2	QRNN	CAViaR-4	QARNN-2	QARNN-2
	0.10	90	QRNN	QARNN-2	QARNN-2	QARNN-1	CAViaR-3
cc.LR	0.01	99	QARNN-1	QARNN-1	QARNN-2	QRNN	QARNN-2
	0.05	95	QARNN-2	QARNN-1	QARNN-1	QARNN-2	QARNN-1
	0.10	90	QRNN	QARNN-1	QARNN-2	QARNN-1	QRNN

The superiority of the proposed QARNN method is demonstrated in Table 4 which summarizes the simulation results for three representative values of τs: 0.01, 0.1, and 0.5. The average value of evaluation indices for QARNN is always less than those for the other methods, except for the values of MAE with QRNN at $\tau = 0.01$, which demonstrate that our method outperforms the others. The bold face results show that QARNN is the optimal method almost for simulation 2 at three different quantiles.

4.3. Real Applications

4.3.1. Real World Data

We illustrate the efficacy of our QARNN model on three important stock market indices: the HSI, the S&P500 and the FTSE100. Our application analysis uses daily observations of the above-mentioned three stock indices from 1st January 2008 to 31st December 2013. After merged the time-period, 1431 log returns are delivered,

Table 4: **Average value of the evaluation indices for 500 test data.**

τ	Indices	Riskmetric	Garch-EVT	CAViaR1	CAViaR2	CAViaR3	CAViaR4	QRNN	APARCH	PCC	QARNN-1	QARNN-2
1%	Risk	19.32	0.35	17.57	8.46	20.48	42.47	12.07	22.46	47.05	**0.08**	0.13
	RMSE	8.11	3.65	4.23	2.92	4.56	6.88	3.53	6.24	8.13	1.18	**1.16**
	MAE	5.91	2.89	1.55	1.30	1.59	3.17	**0.79**	3.14	3.38	1.72	1.16
5%	Risk	12.85	0.41	2.22	0.06	2.44	244.12	4.70	15.89	27.26	3.53	**0.26**
	RMSE	6.22	2.85	1.19	0.64	1.99	16.32	2.48	5.20	6.13	0.75	**0.60**
	MAE	4.56	2.32	1.10	0.45	1.12	8.86	0.64	2.71	2.46	0.99	**0.41**
10%	Risk	9.00	0.71	5.19	0.42	6.41	15.87	1.80	11.57	16.68	1.07	**0.19**
	RMSE	5.17	2.54	2.41	4.15	2.69	4.26	1.45	4.63	4.86	1.38	**0.69**
	MAE	3.90	2.09	0.64	1.65	0.81	1.54	0.41	2.56	1.89	1.51	**0.19**

NOTE: CAViaR-1, CAViaR-2, CAViaR-3, and CAViaR-4 denote CAViaR Symmetric Absolute Value, CAViaR-Asymmetric Slope, CAViaR-Indirect GARCH(1,1) model and CAViaR-Adaptive.

which are defined as $r_t = \log \times (\ln p_t - \ln p_{t-1})$. The first 954 samples are used as training data for model estimation and the remaining 477 samples are left as test data for the out-of-sample evaluation. Table 5 collects the summary statistics of these daily log returns. We find that all returns have mean close to 0 while standard deviations are greater than 1. All of them are negative skewed except for the return of HSI.

Table 5: **Summary statistics of daily log returns.**

	Mean	Median	Min	Max	Std. Dev	Skewness	Kurtosis
HSI	-0.011	0.000	-13.582	13.407	1.843	0.104	8.154
S&P500	0.014	0.078	-9.470	10.957	1.546	-0.283	8.179
FTSE100	0.000	0.000	-9.265	9.384	1.424	-0.086	6.927

4.3.2. Performance of QARNN Model

To further illustrate the performance of our QARNN model for real world data, we also consider the ARMA-APARCH model of Wurtz et al. (2006), the CAViaR model of Engle & Manganelli (2004), and the QRNN model of Cannon (2011) and make thorough and specific comparisons. To be convenient for explanation, we modify the proposed model with predictor $|y_{t-j}|$ rather than y_{t-j} for $j = 1, 2, \cdots, q$, which is in accordance with CAViaR settings. Moreover, the choice of penalty parameter λ is implemented using AIC and GACV, where λ is fixed and takes value from 0, 0.001, 0.01, 0.1, 1, 10, 100 and 1000. We estimate the QARNN models by minimizing AIC and GACV, corresponding to each λ. Then we compare the index Risk in equation (37) using the predicted quantile obtained from each QARNN model. The results show that $\lambda = 1$, which minimizes the index Risk, is the optimal choice.

Before conducting VaR evaluation, the optimal lags of predictors and hidden

nodes in QARNN model determined by AIC and GACV are reported in Table 6. The results show that the optimal choice of lags is either $p = 2, q = 1, k = 3$ or $p = 1, q = 1, k = 3$, which also indicates that we do not need a very complicated neural network structure in real applications. To save space, we only present VaR estimates at confidence levels 99%, 95%, and 90% via the QARNN model for $\tau = 1\%, 5\%, 10\%$ and $p = 1, q = 1, k = 3$ in Fig. 2. It is clear that the volatility of estimated VaR is similar to stock indices returns. We also apply both the uc.LR test and the cc.LR test to assess the performance of models in VaR backtesting and report the results in Table 7.

Interestingly, relative poor performances are measured by CAViaR models. Only CAViaR-Asymmetric Slope performs well without any significant for both uc.LR test and cc.LR test under 5% quantile. It is also worth noting that at the extreme quantile $\tau = 0.01$, our QARNN model is the only one that performs well, whereas others yield higher or lower values of NS. PCC model is recommended to use only for stock HSI under 1% quantile, however, the rest poor performances show that our QARNN model prevails over the other state of arts VaR evaluation methods. The results, summarized in Table 8, show that the QARNN model outperforms the other models in most cases. Moreover, it is not surprisingly that the QARNN model performs better for real world data than for simulation data, with 16/18 recommendations for real stock indices versus 22/30 for numerical experiments. The reason may be that the auto-correlation relationship is common and complicated in the real stock market, and can be successfully resolved by the QARNN model instead of other more restrictive models.

Table 6: **The optimal parameters in QARNN model determined by AIC and GACV for stock indices.**

Stock index	$\tau = 1\%$			$\tau = 5\%$			$\tau = 10\%$		
	p, q, K	GACV	AIC	p, q, K	GACV	AIC	p, q, K	GACV	AIC
HSI	(1,1,3)	0.269	-1.092	(1,1,3)	0.738	-0.400	(2,1,3)	1.091	0.087
S&P500	(1,1,3)	0.220	-1.318	(1,1,3)	0.571	-0.711	(1,1,3)	0.770	-0.261
FTSE100	(1,1,3)	0.177	-1.500	(2,1,3)	0.447	-0.874	(2,1,3)	0.658	-0.419

Table 7: **Out-of-sample VaR backtesting on each stock index.**

τ	Models	HSI			S&P 500			FTSE100			NS1	NS2
		F	P1	P2	F	P1	P2	F	P1	P2		
1%	Riskmetric	1.87	0	0	2.60	0	0	2.47	0	0	3	3
	Garch-EVT	0.80	0.42	0.15	0.20	0.03	0	0.04	0.13	0	1	2
	CAViaR1	1.38	0.16	0.42	1.67	0.02	0	1.38	0.16	0.55	1	1
	CAViaR2	1.00	0.99	0.84	1.52	0.06	0	0.89	0.66	0.23	0	1
	CAViaR3	0.88	0.63	0.02	1.52	0.06	0.04	1.25	0.35	0.40	0	2
	CAViaR4	0.77	0.35	0.56	8.12	0	0	0.89	0.66	0.72	1	1
	QRNN	0.78	0.37	0.22	1.36	0.18	0.02	0.89	0.66	0.23	0	1
	APARCH	1.72	0.01	0.34	3.53	0	0	1.55	0.05	0.11	2	1
	PCC	1	1	0.54	0	0	0	0.02	0.03	0.03	2	2
	QARNN-1	0.87	0.60	0.22	0.87	0.60	0.56	1.00	1.00	0.83	0	0
	QARNN-2	0.80	0.42	0.15	0.73	0.28	0.40	1.27	0.32	0.56	0	0
5%	Riskmetric	6.14	0.05	0.14	6.54	0.01	0.01	6.74	0	0	3	2
	Garch-EVT	3.80	0.12	0	2.80	0.01	0	3.60	0.13	0	1	3
	CAViaR1	5.97	0.09	0.04	6.49	0.01	0.01	4.20	0.14	0.12	1	2
	CAViaR2	4.43	0.30	0.26	4.54	0.41	0.34	4.54	0.41	0.17	0	0
	CAViaR3	5.84	0.15	0.11	6.75	0	0	5.97	0.09	0.12	1	1
	CAViaR4	6.09	0.06	0	7.31	0	0	6.22	0.04	0	2	3
	QRNN	5.12	0.83	0.52	5.24	0.67	0.55	5.24	0.67	0.52	0	0
	APARCH	5.93	0.11	0.10	6.53	0.01	0	3.51	0.01	0	2	2
	PCC	3.60	0.13	0.05	3.00	0.03	0	4.20	0.40	0.23	1	1
	QARNN-1	5.12	0.83	0.74	4.80	0.73	0.91	5.27	0.63	0.82	0	0
	QARNN-2	5.27	0.63	0.89	4.90	0.86	0.84	5.20	0.72	0.17	0	0
10%	Riskmetric	10.41	0.60	0.37	9.41	0.44	0.44	10.74	0.34	0.05	0	0
	Garch-EVT	7.00	0.02	0	6.00	0	0.01	6.80	0.01	0	3	3
	CAViaR1	8.52	0.05	0	12.20	0.01	0	8.06	0.01	0	2	3
	CAViaR2	8.97	0.18	0	12.20	0.01	0	12.30	0	0	2	3
	CAViaR3	12.30	0	0	8.17	0.02	0	12.61	0	0	3	3
	CAViaR4	20.30	0	0	21.00	0	0	21.20	0	0	3	3
	QRNN	9.18	0.28	0.11	9.66	0.66	0.35	9.67	0.67	0.13	0	0
	APARCH	9.21	0.30	0.47	8.57	0.06	0.04	8.77	0.11	0.18	0	1
	PCC	7.40	0.04	0.05	7.00	0.02	0.06	7.80	0.09	0.19	2	1
	QARNN-1	9.41	0.44	0.52	10.10	0.90	0.13	9.61	0.61	0.43	0	0
	QARNN-2	10.40	0.61	0.84	9.27	0.35	0.64	10.00	0.95	0.51	0	0

NOTE: The same as Table 2.

Table 8: **The best models for VaR evaluation on stock indices.**

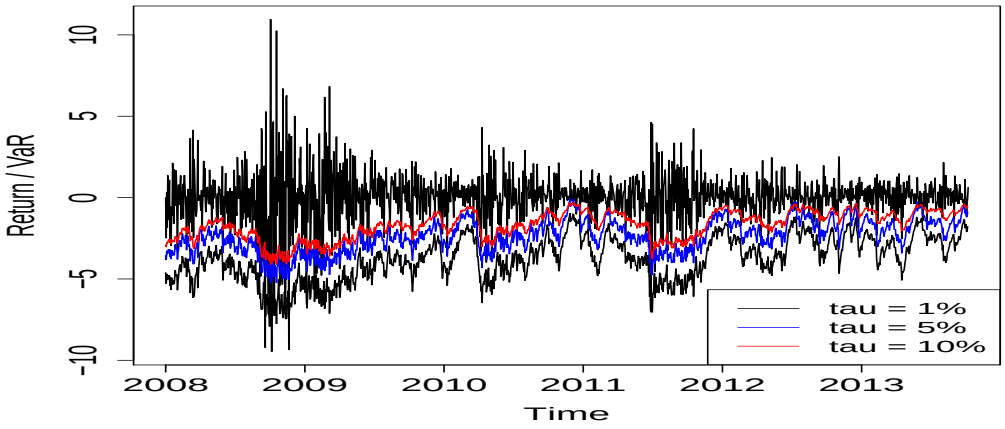
Test	τ	Confidence level (%)	Best model		
			HSI	S&P500	FTSE100
uc.LR	0.01	99	PCC	QARNN-1	QARNN-1
	0.05	95	QARNN-1	QARNN-2	QARNN-2
	0.10	90	QARNN-2	QARNN-1	QARNN-2
cc.LR	0.01	99	CAViaR2	QARNN-1	QARNN-1
	0.05	95	QARNN-2	QARNN-1	QARNN-1
	0.10	90	QARNN-2	QARNN-2	QARNN-2

5. Conclusions

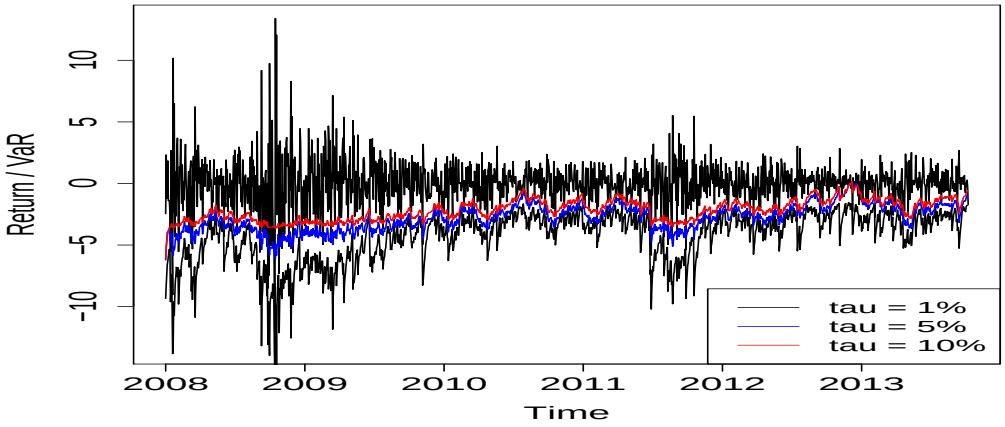
In this article, we reconsider the QAR model based on neural networks and develop a novel nonlinear quantile autoregression model QARNN. The QARNN generalizes existing models and is very flexible at describing complicated data structures. An appealing feature of the QARNN model is that the autoregressive quantiles are used as predictors recursively and can be estimated directly. To illustrate the efficacy of the proposed model we conduct Monte Carlo simulation studies and extensive tests on different stock indices. Numerical results show that the QARNN model is able to explore nonlinearity in financial time series and performs better in VaR evaluation than some competing models, including RiskMetric, GARCH-EVT, ARMA-APARCH, CAViaR, PCC and QRNN.

A noteworthy issue in QARNN modelling lies in penalization of weights in both hidden layer and output layer. We use a quadratic or L2-norm penalty terms to address the overfitting problem. Alternative penalty terms such as L1-norm regularization terms can also be implemented for variable selection, which will increase the interpretability of our model for the case of large numbers of variables.

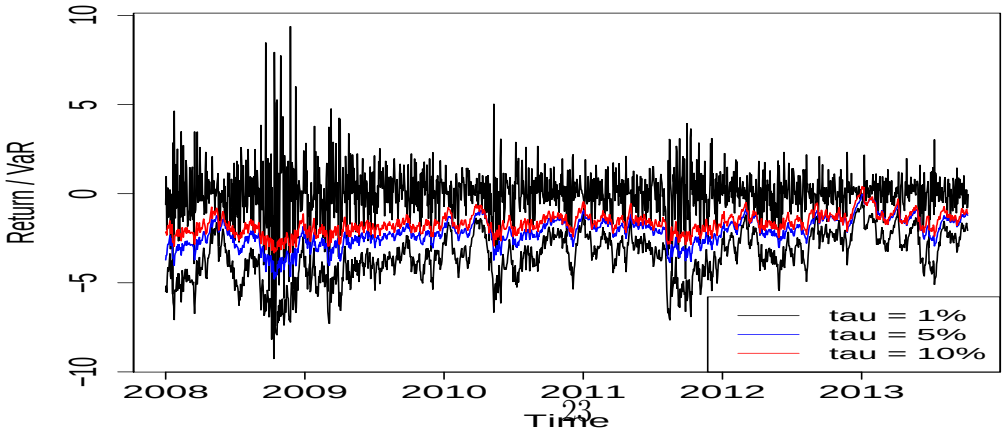
The current version of QRNN or QARNN is for continuous quantile regression analysis, and we plan to develop a classification version of QARNN by combining binary quantile regression of Kordas (2006) with ANN techniques. A more thorough comparison of the QRNN or QARNN model to other classification methods on different benchmark data sets is also desired. But how to design a QRNN or QARNN model for multi-class classification is still a challenging task.



(a) S&P 500



(b) HSI



(c) FTSE 100

Figure 2: **Stock indices returns plot superimposed by VaR estimates via QARNN model with $p = 1, q = 1, k = 3$ at $\tau = 1\%$, 5% and 10%.**

Acknowledgement

The authors would like to thank the Editor-in-Chief and the anonymous reviewers and referees for their useful suggestions and comments that led to a considerably improved version of the article. Any remaining errors are the authors' own. The authors also gratefully acknowledge financial support from the National Natural Science Foundation of PR China (71490725, 71071087), the National Social Science Foundation of PR China (15BJY008), the Humanity and Social Science Foundation of Ministry of Education of PR China (14YJA790015) and the Science and Technology Development Plan Project of Shandong Province (2014GGX101055).

Appendix

We supplement sufficient detail of the QARNN model from the following aspects: design and implementation, Domain of use and execution, Quality, reliability and limitations, and the significance of the model.

Design and implementation of QARNN

Inspired by the QRNN of Cannon (2011), which implements a flexible nonlinear quantile regression for cross-section data without prior specification of the form of the relationships, we consider a nonlinear quantile regression in the context of time series and develop the QARNN model. The relationships among several popular regression methods are summarized in Fig. 3.

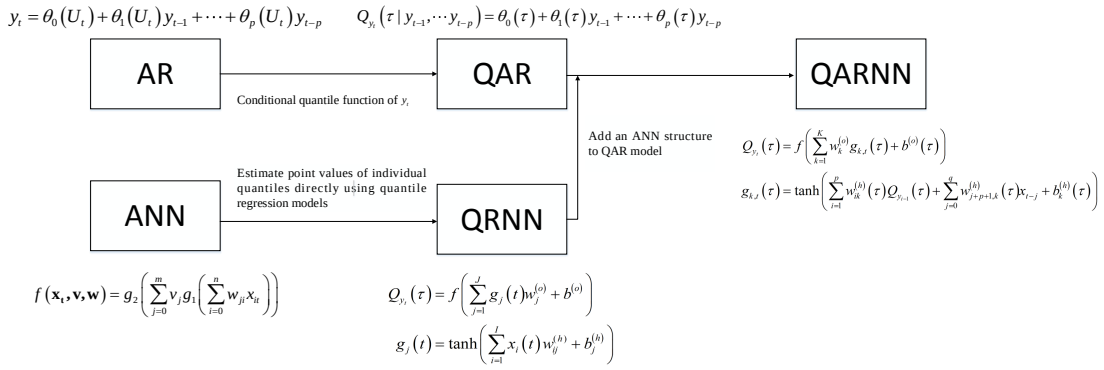


Figure 3: **The relationships among several popular regression methods**

It is worth to note that our QARNN method is different from Cannon (2011). In Cannon (2011), the QRNN model they developed only take explanatory variables as

inputs, however, our QARNN model considers the nonlinear-in-parameters effects not only of exogenous variables but also of the lagged endogenous variables. Therefore, the QARNN model is flexible and can implement a nonlinear quantile autoregression for time series data and estimate nonlinear relationships without the need to specify a precise functional form.

In CAViaR model, the impacts of autoregressive terms and exogenous covariates on the conditional quantile of response are independent with specified form in advance, which is perhaps not suitable for actual data. We extend CAViaR model to a general dependent structure as

$$Q_{y_t}(\tau|\mathbb{F}_t) = f(Q_{y_{t-1}}(\tau), \cdots, Q_{y_{t-p}}(\tau), \boldsymbol{x}_t, \boldsymbol{x}_{t-1}, \cdots, \boldsymbol{x}_{t-q}; \hat{\boldsymbol{\theta}}(\tau)), \tag{45}$$

where f is an arbitrary nonlinear function and we consider the use of ANN to simulate the nonlinear structure in f. As the recursive terms need to be initialized for training the neural network at the beginning, we compute the empirical quantile correspond to each τ using the first m observations, specifically, m is often chosen as $1/10$ of the sample size. Given predictors $Q_{y_{t-1}}(\tau), \cdots, Q_{y_{t-p}}(\tau), \boldsymbol{x}_t, \boldsymbol{x}_{t-1}, \cdots, \boldsymbol{x}_{t-q}$ and a predicted $Q_{y_t}(\tau)$, outputs from a QARNN, are calculated as follows. First, an output from the k-th hidden layer node $g_{k,t}(\tau)$ is given by applying a sigmoid transfer function to the inner product between the predictors and the hidden layer weights $w_{pk}^{(h)}$ plus the hidden layer bias $b_k^{(h)}$

$$g_{k,t}(\tau) = f^{(h)}\left(\sum_{i=1}^{p} w_{ik}^{(h)}(\tau) Q_{y_{i-1}}(\tau) + \sum_{j=0}^{q} w_{j+p+1,k}^{(h)}(\tau)\boldsymbol{x}_{t-j} + b_k^{(h)}(\tau)\right), \tag{46}$$

where $\boldsymbol{w}^{(h)} = (w_{1k}^{(h)}, w_{2k}^{(h)}, \cdots, w_{p+q+1,k}^{(h)})$ is a weight vector of hidden layer, $\boldsymbol{b}^{(h)}(\tau) = (b_1^{(h)}(\tau), b_2^{(h)}(\tau), \cdots, b_K^{(h)})(\tau)$ is a bias vector of hidden layer, and $f^{(h)}$ denotes a sigmoid transfer function such as the hyperbolic tangent. Second, an estimate of the conditional τth conditional quantile of response is then given by

$$Q_{y_t}(\tau) = f^{(o)}\left(\sum_{k=1}^{K} w_k^{(o)} g_{k,t}(\tau) + b^{(o)}(\tau)\right) \tag{47}$$

where $w_k^{(o)}$ are the output layer weights, $b^{(o)}(\tau)$ is the output layer bias, and $f^{(o)}$ is the output layer transfer function such as identity function.

The designed structure is flexible, which considers the nonlinear-in-parameters effects of both exogenous variables and the lagged endogenous variables. The modeling process mainly includes the model estimation and model selection. In the model

estimation, we substitute the approximate check function in equation (16) for the original one in equation (1), as its derivative is invalid at the origin. The substitution makes sure that the standard gradient optimization algorithm can be implemented to estimate the model. In the model selection, we add the penalty term, which controls the trade-off between the error and the weight decay term, to avoid the over-fitting issue during the estimating process. Moreover, we adopt AIC and GACV criteria to perform parameters tuning.

Domain of use and execution

In our particular paper, we account for the implementation only with VaR evaluation, however, the proposed approach can be extended to various areas of application similar to ANN or QRNN model, for example, environmental modelling (Nagy et al., 2002, Shu & Burn, 2004 Cannon, 2011), management (Feng et al., 2010), and survival analysis (Eleuteri et al., 2003), etc.

We implement the proposed QARNN in R. Under the computation environment, the running time for obtaining results of Monte Carlo simulation studies and real world applications are summarized in Table 9. The model can also be executed in Matlab, SAS, etc. All numerical experiments are carried out on an Intel(R) Core(TM) i7-4510U CPU (2.60 GHz) processors and 8 GB RAM.

Table 9: **Running time of QARNN models (in seconds)**

Data	QARNN-1	QARNN-2
s1.norm	77.92	131.43
s1.std	85.27	122.73
s1.snorm	97.53	144.79
s1.sstd	66.86	121.93
s2.norm	79.80	61.62
HSI	76.31	79.29
GSPC	106.31	112.35
FTSE	81.40	80.42

Quality, reliability and limitation

Since quality and reliability are important to decide whether the proposed model under examination is acceptable, we have utilized several ways to elaborate whether the models chosen are consistent with whatever data are available in Section 4.2.3. According to the results of all the simulation data, we could tell that our QARNN method outperforms the others by both out-of-sample VaR backtesting and the evaluation indices, which show both quality and reliability of the model.

It is worthy noting that, similar to QAR in Koenker & Xiao (2006), the issues of identifiability and possible misspecification of models suggest that extra care should be made in making this kind of links. For example, in terms of the scheme of conditional quantile inference, we compute the empirical quantile correspond to each τ using the first m observations (m is often chosen as 1/10 of the sample size) to initialize the recursive terms. Although the reliability has been proved through Monte Carlo results, further studies of how robust the fitting techniques are to the model misspecification are still needed.

Significance

The QARNN model is proposed that combines an ANN with the QAR method for time series data. It has two advantages. First, the QARNN model can be used to explore potential nonlinear relationships by taking advantage of the powerful nonlinear processing capacity of ANN. The main advantages of nonlinear models are parsimony, interpretability, and prediction (Bates & Watts, 2007). In general, nonlinear models are capable of accommodating a vast variety of mean functions, although each individual nonlinear model can be less flexible than linear models in terms of the variety of data they can describe; however, nonlinear models appropriate for a given application can be more easily interpretable. Moreover, predictions of nonlinear models tend to be more robust that competing polynomials, especially outside the range of observed data, which make the proposed model is able to achieve high prediction accuracy. The second advantage of the QARNN model is that it provides more information for decision-making by using the ability of QAR to discover the entire conditional distribution of time series. For example, the model consists of a process behavior and also the disturbance term/noise showing drifting characteristics, therefore one can determine forecasting of the outputs easily and accurately by considering the process behavior, disturbances and outputs.

References

Abad, P., Benito, S., & Lopez, C. (2014). A comprehensive review of value at risk methodologies. *The Spanish Review of Financial Economics*, *12*, 15–32.

Ali, G. (2013). Egarch, gjr-garch, tgarch, avgarch, ngarch, igarch and aparch models for pathogens at marine recreational sites. *Journal of Statistical and Econometric Methods*, *2*, 57–73.

Allen, D. E., Singh, A. K., & Powell, R. J. (2013). EVT and tail-risk modelling: Evidence from market indices and volatility series. *North American Journal of Economics and Finance*, *26*, 355–369.

Auśin, M., Galeano, P., & Ghosh, P. (2014). A semiparametric bayesian approach to the analysis of financial time series with applications to value at risk estimation. *European Journal of Operational Research*, *232*, 350–358.

Bates, D., & Watts, D. (2007). *Nonlinear Regression Analysis and Its Applications*. Wiley Series in Probability and Statistics. Wiley.

Bollerslev, T. (1986). Generalized autoregressive conditional heteroskedasticity. *Journal of econometrics*, *31*, 307–327.

Cannon, A. J. (2011). Quantile regression neural networks: Implementation in r and application to precipitation downscaling. *Computers & Geosciences*, *37*, 1277–1284.

Chen, C., Gerlach, R., Hwang, B., & McAleer, M. (2012). Forecasting value-at-risk using nonlinear regression quantiles and the intra-day range. *International Journal of Forecasting*, *28*, 557–574.

Chen, X., Konenker, R., & Xiao, Z. (2009). Copula-based nonlinear quantile autoregression. *The Econometrics Journal*, *12*, S50–S67.

Christoffersen, P. F. (1998). Evaluating interval forecasts. *International Economic Review*, *39*, 841–862.

Ding, Z., C.W.J., G., & R.F., E. (1993). A long memory property of stock market returns and a new model. *Journal of Empirical Finance*, *1*, 83–106.

Eleuteri, A., Tagliaferri, R., Milano, L., Placido, S., & Laurentiis, M. (2003). A novel neural network-based survival analysis model. *Nerual Networks*, *16*, 855–864.

Engle, R. F. (1982). Autoregressive conditional heteroscedasticity with estimates of the variance of united kingdom inflation. *Econometrica*, *50*, 987–1007.

Engle, R. F., & Manganelli, S. (2004). Caviar: conditional autoregressive value at risk by regression quantiles. *Journal of Business and Economic Statistics*, *22*, 367–381.

Feng, Y., Li, R., Sudjianto, A., & Zhang, Y. (2010). Robust neural network with applications to credit portfolio data analysis. *Statistics and its interface*, *3*, 437–444.

Galvao Jr, A., Montes-Rojas, G., & Olmo, J. (2011). Threshold quantile autoregressive models. *Journal of Time Series Analysis*, *32*, 253–261.

Galvao Jr, A., Montes-Rojas, G., & Park, S. (2013). Quantile autoregressive distributed lag model with an application to house price returns. *Oxford B Econ Stat*, *75*, 307–321.

Ghouch, A. E., & Genton, M. G. (2009). Local polynomial quantile regression with parametric features. *Journal of the American Statistical Association*, *104*, 1416–1429.

Ghourabi, M., Francq, C., & Telmoudi, F. (2015). Consistent estimation of the value at risk when the error distribution of the volatility model is misspecified. *Journal of Time Series Analysis*, *36*, 198–205.

Gouriéroux, C., & Jasiak, J. (2008). Dynamic quantile models. *Journal of Econometrics*, *147*, 198–205.

Jeon, J., & Taylor, J. (2013). Using caviar models with implied volatility for value-at-risk estimation. *Journal of Forecasting*, *32*, 62–74.

Jiang, R., Qian, W.-M., & Li, J.-R. (2014). Testing in linear composite quantile regression models. *Computational Statistics*, *29*, 1381–1402.

Koenker, R. (2005). *Quantile regression*. New York: Cambridge University Press.

Koenker, R. (2011). Additive models for quantile regression: model selection and confidence bandaids. *Brazilian Journal of Probability and Statistics*, *25*, 239–262.

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

Koenker, R., & Mizera, I. (2004). Penalized triograms: total variation regularization for bivariate smoothing. *Journal of the Royal Statistical Society (Series B)*, *66*, 145–163.

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

Koenker, R., & Xiao, Z. (2006). Quantile autoregression. *Journal of the American Statistical Association*, *101*, 980–990.

Kordas, G. (2006). Smoothed binary regression quantiles. *Journal of Applied Econometrics*, *21*, 387–407.

Kupiec, P. (2006). Techniques for verifying the accuracy of risk measurement models. *Journal of Derivatives*, *3*, 73–84.

McNeil, A. J., & Frey, R. (2000). Estimation of tail-related risk measures for heteroscedastic financial time series: An extreme value approach. *Journal of Empirical Finance*, *7*, 271–300.

Morgan, J. (1996). *Riskmetrics: technical document*. Technical Report Morgan Guaranty Trust Company.

Nagy, H., Watanabe, K., & Hirano, M. (2002). Prediction of sediment load concentration in rivers using artificial neural network model. *Journal of Empirical Finance*, *128*, 588–595.

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

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

Righi, M., & Ceretta, P. (2015). Forecasting value at risk and expected shortfall based on serial pair-copula constructions. *Expert Systems with Applications*, *42*, 6380–6390.

Saravanan, N., Siddabattuni, V. K., & Ramachandran, K. (2010). Fault diagnosis of spur bevel gear box using artificial neural network (ann), and proximal support vector machine (psvm). *Applied Soft Computing*, *10*, 344–360.

Shu, C., & Burn, D. H. (2004). Artificial neural network ensembles and their application in pooled flood frequency analysis. *Water Resources Research*, *40*. doi:10.1029/2003WR002816. W09301.

Taylor, J. W. (1999). A quantile regression approach to estimating the distribution of multiperiod returns. *The Journal of Derivatives*, *7*, 64–78.

Taylor, J. W. (2000). A quantile regression neural network approach to estimating the conditional density of multiperiod returns. *Journal of Forecasting*, *19*, 299–311.

Wurtz, D., Chalabi, Y., & Luksan, L. (2006). Parameter estimation of arma models with garch/aparch errors an r and splus software implementation. *Journal of Statistical Software*, *55*, 28–33.

Yeh, I. C. (2014). Estimating distribution of concrete strength using quantile regression neural networks. *Applied Mechanics and Materials*, *584-586*, 5441–5451.

Yu, K., & Jones, M. C. (1998). Local linear quantile regression. *Journal of the American Statistical Association*, *93*, 228–237.

Yu, K., Lu, Z., & Stander, J. (2003). Quantile regression: applications and current research areas. *Journal of the Royal Statistical Society: Series D*, *52*, 331–350.

Yuan, M. (2006). Gacv for quantile smoothing splines. *Computational Statistics and Data Analysis*, *50*, 813–829.