Master's Paper of the Department of Statistics, the University of Chicago

Estimating Conditional Value-at-Risk Using Kernel Conditional Quantile Estimation

Gulishana Adilijiang

Advisor: Professor Wei Biao Wu

Approved Wei Biao Wu

Date May 15, 2020

May-11, 2020

Abstract

Value-at-Risk (VaR) is a measure of the risk of loss for investments. And conditional Value-at-Risk (cVaR) is becoming more and more popular, since its measurement is conditioning on the information variables, such as the prices (or returns) of portfolios. In this paper, we estimate the cVaR using the kernel conditional quantile estimation method, which has been demonstrated to be applicable to short-range and long-range-dependent processes. To evaluate the performance of this estimator, simulation studies are carried out for nonlinear TAR, ARCH and SV models that generate weakly dependent data that are widely used in modeling financial time series. Also, the simulated ARCH and SV models have a heavier tail which is a common aspect of financial data. In the end, the estimation method is also examined by real data analysis implemented on the daily and monthly returns of several stocks and indexes.

Keywords: conditional Value-at-Risk, kernel contidional quantile estimation, bandwidth selection, bias correction, nonparametric kernel smoothing

Acknowledgement

First of all, I would like to express my sincere appreciation to my advisor, Professor Wu, who has carefully instructed my master thesis research and spent a lot of time to help me solve my problems. Second, I want to thank Dr.Wang for her kindly help through my whole master study in the Department of Statistics. In addition, I really appreciate our department for financially supporting my master study. Finally, I want to thank my husband for his support to me during these challenging two years. In the end, I want to thank my dear daughter for coming to our lives in the first year of master study and being so adorable and sweet.

Contents

1 Introduction

1.1 Conditional Value-at-Risk

In financial risk measurement and management, the investors use Value-at-Risk (VaR) to evaluate the risk of the market, assets, investment portfolios, et al. VaR is widely applied to any kind of financial instruments. It was invented by J.P. Morgan. During the late 1980s, he developed a firm-wide VaR system. According to Morgan (1996), VaR basically answers the most concerned question of the investors: "how much can I lose with probability p over a pre-set horizon?" By definition, the (1-p)-th VaR is the loss of a portfolio (measured in price or return) that is exceeded with a given probability p in a given time period (usually a day, a month or a year). Typically either $p = 5\%$ or 1%. For example, if a daily 95% VaR of a stock return equals to 8%, then it means that there is a 5% chance that this stock will lose 8% or more on a given day. According to Thomas and Neil (1999), VaR is usually estimated with three common methods, which are historical simulation, Monte Carlo simulation, and the variance-covariance approach.

According to Carlo and Dirk (2002), a natural coherent alternative to VaR is the Expected Shortfall (ES), which is also known as the Conditional Value-at-Risk (CVaR). CVaR is the expected loss of the tail risk where the losses are equal to or larger than the estimated VaR value. However, in this paper, the conditional Value-at-Risk (cVaR) we talk about is different from this CVaR defined above. The cVaR is the estimation of VaR conditioning on a state process or the information variables, such as the prices (or returns) of portfolios. cVaR is becoming more and more popular since it links the VaR measurement with the dependent information variables. The methods to estimate the cVaR includes parametric and nonparametric approaches. Chernozhukov and Umantsev (2001) established parametric conditional quantile regression models for estimating cVaR. Other more sophisticate parametric models are also developed later. However, though the parametric model is better in interpreting the estimation results, it also requires some assumptions of the underlying models, which may not be the true model for the data. In contrast, the nonparametric approaches can deal with different types of data and do not require such model assumptions. Yu and Jones (1998) introduced kernel-based conditional quantile regression methods for the i.i.d. cases. In this paper, we develop a kernel conditional quantile regression method which can be applied to more complex time series financial models which have dependence structures in the data.

1.2 Mathematical Models and Definitions

In this paper, there are two types of model settings with dependence structures that will be used for estimating conditional quantiles.

Model 1:

$$Z_i = G(X_i, \varepsilon_i)$$

where ε_i are i.i.d., $\varepsilon_i \perp X_j$ $(j \leq i-1)$, and (X_i, Z_i) is stationary. Usually the sequence $\{X_t\}_{t\in\mathbb{Z}}$ has a dependence structure: $X_i = R(\eta_i, \eta_{i-1}, \cdots)$, where η_i are i.i.d., and R is a measurable function such as a linear or nonlinear function such that X_i is well defined. An example of Model 1 is the stochastic volatility (SV) model, which will be used in the simulation study of this paper.

Model 2:

$$Z_i = G(Z_{i-1}, \varepsilon_i)$$

where ε_i are i.i.d., and $\varepsilon_i \perp Z_{i-1}$. In this case, the sequence $\{Z_t\}_{t\in\mathbb{Z}}$ itself has a dependence structure. So we can set $X_i = Z_{i-1}$, then the Model 2 will become a similar regression problem with Model 1. Examples of Model 2 include the threshold autoregressive (TAR) model and the autoregressive models with conditional heteroscedasticity (ARCH) model, which will also be used in the simulation study of this paper.

Based on these two models, we then can define the mathematical expressions of the conditional quantile. First, we define the following distribution and density functions:

(a) cdf and pdf of X_t: $F(x) = \mathbb{P}(X_i \leq x)$, $f(x) = F'(x)$

(b) joint density $f(\xi, x) = f(\xi|x)f(x)$

(c) conditional cdf and pdf of $Z_t|X_t$: $F(\xi|x) = \mathbb{P}(Z_i \leq \xi|X_i = x)$, $f(\xi|x) = \frac{\partial F(\xi|x)}{\partial \xi}$

Therefore, let $\Omega^t = (..., X_{t-1}, X_t)$ be the information available up to time t. Then the (1-p)-th cVaR, which is the p-th conditional quantile of X_{t+1} (i.e. Z_t) given Ω^t, equals to:

$$\xi_p(x) = \inf_{\xi}\{\xi : F(\xi|x) = \mathbb{P}(Z_t \leq \xi|X_t = x) \geq p\} \quad (0 < p < 1)$$

1.3 Organization of the paper

In this paper, we first derive the step-by-step algorithm for kernel conditional quantile estimation method in section 2. And then in section 3, we implement this algorithm on three different types of models to see the behaviors and performances of the estimator. Finally, in section 4, we bring out real data analysis using this algorithm on three categories of stock return data to estimate their conditional VaR for $p = 1\%$ and 5%.

2 Methodology and Theory

2.1 Kernel Conditional Quantile Estimation

2.1.1 Kernel Function

Kernel functions are widely used in nonparametric statistics for kernel density estimations and kernel regressions. According to the definition, a kernel is a symmetric function $K : \mathbb{R} \to \mathbb{R}$ that satisfies the following requirements:

(a) $K(-u) = K(u)$ for all $u \in \mathbb{R}$

(b) $\int uK(u)du = 0$ and $\int K(u)du = 1$

(c) $\int u^2|K(u)|du < \infty$

There are some specific values for a given kernel function. In this paper, we define them as:

$$\kappa = \int K^2(u)du \ , \quad \varrho = \frac{1}{2}\int u^2 K(u)du \ , \quad K_0 = \sup_u |K(u)| < \infty \tag{1}$$

where κ and τ are used in defining the efficiency of a kernel, MSE (mean squared error) of the kernel estimator, and bandwidth selection.

The most commonly used kernel function is the Gaussian kernel function, i.e. $K(x) = \phi(x)$, where ϕ is the standard normal density function. We define $K_h(u) = K(u/h)$, where $h > 0$ is a smoothing parameter called the bandwidth. In our paper, we also use the Gaussian kernel in the estimations. Thus for Gaussian kernel, we can get the exact values in (1): $\kappa = \frac{1}{2\sqrt{\pi}}$, $\varrho = \frac{1}{2}$, $K_0 = \frac{1}{\sqrt{2\pi}}$.

2.1.2 Kernel Density Estimation

First, we consider the kernel density estimation method. Suppose $(X_1, X_2, \cdots, X_n)$ are the samples drawn from an unknown distribution with true density function $f(x)$. Then the kernel density estimate of $f(x)$ is

$$\hat{f}_h(x) = \frac{1}{n}\sum_{i=1}^{n}\frac{1}{h}K_h(x - X_i) = \frac{1}{nh}\sum_{i=1}^{n} K\left(\frac{x - X_i}{h}\right)$$

where the bandwidth h satisfies that $h \to 0$ and $nh \to \infty$.

2.1.3 General Kernel Regression

Next, we consider the general kernel regression method. Suppose $\{(X_1,Y_1),(X_2,Y_2),\cdots,(X_n,Y_n)\}$ are the samples drawn from an unknown joint distribution with true joint density function $f(x,y)$. Then the kernel density estimate of $f(x,y)$ is

$$\hat{f}_{h_x,h_y}(x,y)=\frac{1}{n}\sum_{i=1}^{n}\frac{1}{h_x}K_{h_x}(x-X_i)\frac{1}{h_y}K_{h_y}(y-Y_i)=\frac{1}{nh_xh_y}\sum_{i=1}^{n}K_{h_x}(x-X_i)K_{h_y}(y-Y_i)$$

Since the conditional expectation of Y given $X=x$ is

$$E(Y|X=x)=\int yf(y|x)dy=\int y\frac{f(x,y)}{f(x)}dy=\frac{\int yf(x,y)dy}{f(x)}$$

Thus the kernel regression estimate of $E(Y|X=x)$ is

$$\begin{aligned}\hat{E}(Y|X=x)&=\frac{\int y\hat{f}_{h_x,h_y}(x,y)dy}{\hat{f}_{h_x}(x)}=\frac{\int y\frac{1}{nh_xh_y}\sum_{i=1}^{n}K_{h_x}(x-X_i)K_{h_y}(y-Y_i)dy}{\frac{1}{nh_x}\sum_{i=1}^{n}K_{h_x}(x-X_i)}\\&=\frac{\sum_{i=1}^{n}K_{h_x}(x-X_i)\int y\frac{1}{h_y}K_{h_y}(y-Y_i)dy}{\sum_{i=1}^{n}K_{h_x}(x-X_i)}=\frac{\sum_{i=1}^{n}K_{h_x}(x-X_i)\ Y_i}{\sum_{i=1}^{n}K_{h_x}(x-X_i)}\end{aligned}$$

This is called the Nadaraya–Watson estimator of the kernel regression method.

2.1.4 Kernel Conditional Distribution Estimation

Then, we use the results of kernel regression to derive the kernel estimation of the conditional distribution $F(\xi|x)$. Note that $F(\xi|x)=\mathbb{P}(Z\le\xi|X=x)=E(\mathbf{1}_{Z\le\xi}|X=x)$, which can be seen as the conditional expectation of $\mathbf{1}_{Z_i\le\xi}$ given $X_i=x$. Therefore, we can convert a problem of estimating the conditional distribution to one of estimating the conditional expectation using the kernel regression method. Substituting the Y_i with $\mathbf{1}_{Z_i\le\xi}$ in the above formula, we can get the Nadaraya–Watson estimator of the conditional distribution $F(\xi|x)$, which is

$$\begin{aligned}F_n(\xi|x)=\hat{E}(\mathbf{1}_{Z\le\xi}|X=x)&=\frac{\sum_{i=1}^{n}K_h(x-X_i)\ \mathbf{1}_{Z_i\le\xi}}{\sum_{i=1}^{n}K_h(x-X_i)}\\&=\sum_{i=1}^{n}\frac{K_h(x-X_i)}{\sum_{i=1}^{n}K_h(x-X_i)}\mathbf{1}_{Z_i\le\xi}=\sum_{i=1}^{n}w_i(x)\ \mathbf{1}_{Z_i\le\xi}\end{aligned}\tag{2}$$

which is a weighted average of $\mathbf{1}_{Z_1 \le \xi}, \mathbf{1}_{Z_2 \le \xi}, \cdots, \mathbf{1}_{Z_n \le \xi}$, and $w_i(x) = \frac{K_h(x-X_i)}{\sum_{i=1}^n K_h(x-X_i)}$ is the weight that satisfies $\sum_{i=1}^n w_i(x) = 1$.

2.1.5 Kernel Conditional Quantile Estimation

Recall that our purpose is to get the kernel conditional quantile estimation of $\xi_p(x)$. We can estimate it by the sample conditional quantile, which is

$$\hat{\xi}_p(x) = \inf_{\xi} \{\xi : F_n(\xi|x) = \sum_{i=1}^n w_i(x)\ \mathbf{1}_{Z_i \le \xi}\ \ge p\}$$

By definition, $p = F(\xi_p(x)|x)$, and from equation (2), we can get that $\hat{\xi}_p(x)$ satisfies:

$$\begin{aligned} |F_n(\hat{\xi}_p(x)|x) - F(\xi_p(x)|x)| = |F_n(\hat{\xi}_p(x)|x) - p| &= \frac{\sum_{i=1}^n K_h(x - X_i)\ (\mathbf{1}_{Z_i \le \hat{\xi}_p(x)} - p)}{\sum_{i=1}^n K_h(x - X_i)} \\ &\le \frac{\sup_u |K(u)|}{\sum_{i=1}^n K_h(x - X_i)} = \frac{K_0}{\sum_{i=1}^n K_h(x - X_i)} \end{aligned}$$

Therefore, we can set up a grid values of $x \in [\min(X_i), \max(X_i)]$, then for each $X = x$, we can perform a grid search within a sequence of quantile values $\xi's$ to find an optimal quantile value $\hat{\xi}_p(x)$ such that $F_n(\hat{\xi}_p(x)|x)$ is the closest to the given percentile p among them, i.e. $|F_n(\hat{\xi}_p(x)|x) - p|$ is the smallest.

Note that the sequence of quantile values $\xi's$ to be searched can be just equal to the sample values of $Z_1, Z_2, \cdots, Z_n$. That is to say, instead of setting up equally spaced grid values of $\xi \in [\min(Z), \max(Z)]$, we only need to search $\xi \in \{Z_1, Z_2, \cdots, Z_n\}$. Suppose $Z^{(1)} \le Z^{(2)} \le \cdots \le Z^{(n)}$ are the ordered observations, then for all the values of $\xi^* \in [Z^{(1)}, Z^{(2)})$, the vector $\mathbf{1}_{Z \le \xi^*} = (1, 0, \cdots, 0)^T$ is the same for these ξ^* values, thus will get the same $F_n(\xi^*|x)$ value. And within these ξ^* values, the smallest value $Z^{(1)}$ would be chosen as a candidate due to the definition for $\hat{\xi}_p(x)$.

2.2 Theories and Properties of the Estimate

To select the optimal smoothing parameter, bandwidth h, and further improve the performance of the estimator by reducing the estimation biases, we need to explore the properties and asymptotic distribution of the estimator first. Here we introduce the relative theories and properties regarding the kernel conditional quantile estimator $\hat{\xi}_p(x)$.

Recall that in chapter 1 we introduced two mathematical models, and both models can be written in the form of $Z_i = G(X_i, \varepsilon_i)$, where the process $\{X_i\}$ has a dependence structure. So in this section we will first discuss the properties and theories of the estimator for the short-rang-dependent processes, and then extend it to the long-range-dependent linear processes. For detailed proofs of the theories, please see the paper of Wu, et al. (2007).

Bahadur (1966) first established an approximation of sample quantiles by empirical distribution functions. Wu (2005b) further established the Bahadur representation of sample quantiles for linear and some widely used nonlinear processes. Here we also want to obtain a Bahadur representation for our sample conditional quantile, i.e. the kernel conditional quantile estimator. Bhattacharya and Gangopadhyay (1990) and Chaudhuri (1991) have provided Bahadur representations for kernel conditional quantile regression under iid assumption. However, we need a form for short-range-dependent processes. First, let's define

$$P_n(\xi) = \sum_{i=1}^{n} K_h(x - X_i)\ (\mathbf{1}_{Z_i \le \xi} - p), \quad \text{and} \quad Q_n(\xi) = \mathbb{E}[P_n(\xi)]$$

Then we can get a bound for the approximation error of $\hat{\xi}_p(x) - \xi_p(x)$ by $P_n(\xi) - Q_n(\xi)$, which is shown in *Theorem 1.* This approximation is just the Bahadur representation for our kernel conditional quantile estimator. In this case, the estimator is approximated by a linear form so that it is easier to study its asymptotic behaviors. However, the Theorem 1 needs some appropriate conditions, which are shown in *Condition 1* and *Condition 2.* Condition 1 shows that the process X_t is a short-range dependent (i.e. weakly dependent) process. And condition 2 provides the mild regularity conditions.

Condition 1 (Short-Range Dependence):

Let $f_\eta(x|\mathcal{F}_i)$ be the conditional density of X_{i+1} given $\mathcal{F}_i = (..., \eta_{i-1}, \eta_i)$. Assume that $\sup_x f_\eta(x|\mathcal{F}_0) \le C_0$ almost surely for some $C_0 < \infty$, and that $\sum_{k=0}^{\infty} \omega_k < \infty$ where $\omega_k = \sup_{x\in\mathbb{R}} \|\mathcal{P}_0 f_\eta(x|\mathcal{F}_k)\|$

Here $\mathcal{P}_i$ is the projection operator: $\mathcal{P}_i \cdot = \mathbb{E}(\cdot|\mathcal{F}_i) - \mathbb{E}(\cdot|\mathcal{F}_{i-1})$, and $\|Z\| = [\mathbb{E}(|Z|^2)]^{1/2}$

Condition 2 (Mild Regularity Conditions):

1. (a) $f(x) > 0$ and $\sup_x f(x) < \infty$
   (b) The jth order derivatives $f^{(j)}(x),\ j = 1, ..., 4$, exist on $\mathbb{R}$ and are bounded.
2. (a) $f(\xi, x) > 0$, where $F(\xi|x) = p$
   (b) All first order partial derivatives of $f(\cdot, \cdot)$ exist and are bounded.

Theorem 1 (Bahadur Representation for Short-Range-Dependent Processes):

Let $r_n = (nh)^{-1/2} + h^2$, $\mu_n = r_n^2 + [(r_n \log n)/(nh)]^{1/2}$, $B(\xi, x) = F(\xi|x)f(x)$, $A(\xi, x) = \partial^2 B(\xi, x)/\partial x^2$, $C(\xi, x) = \varrho[pf''(x) - A(\xi, x)]/f(\xi, x)$. Assume that Conditions 1 and 2 hold and that $r_n(\log n)^2 \to 0$. Then

$$\hat{\xi}_p(x) - \xi_p(x) = \frac{Q_n(\xi_p(x)) - P_n(\xi_p(x))}{nhf(\xi_p(x), x)} + h^2 C(\xi_p(x), x) + O_{\mathbb{P}}(\mu_n)$$

Based on Theorem 1, we can derive the asymptotic distribution of $hat\xi_p(x) - \xi_p(x)$, which leads to the *Corollary 1* that can be applied to construct confidence intervals for $\xi_p(x)$.

Corollary 1 (Central Limit Theorem):

Assume $nh^9 \to 0$ and $h \to 0$. Under conditions of Theorem 1, we have

$$\hat{\xi}_p(x) - \xi_p(x) - h^2 C(\xi_p(x), x) \ \Rightarrow \ N\left(0, \ \frac{\kappa p(1-p) f(x)}{nhf^2(\xi_p(x), x)}\right)$$

Note that the Bahadur representation in Theorem 1 holds for both linear and non-linear short-range-dependent processes. Furthermore, Wu, et al. (2007) also showed in *Theorem 2* that for long-range-dependent linear processes, the Theorem 1 holds with $O_{\mathbb{P}}(\mu_n')$.

2.3 Bandwidth Selection

To compute the conditional distribution $F_n(\xi|x)$ in equation (2), first we need to calculate the the weights $w_i(x) = \frac{K_h(x - X_i)}{\sum_{i=1}^n K_h(x - X_i)}$. So it is crucial to select the smoothing parameter, i.e. the bandwidth h, for the the kernel density function $K_h(x - X_i) = K\left(\frac{x - X_i}{h}\right) = \phi\left(\frac{x - X_i}{h}\right)$. By Corollary 1 (CLT), we can get the asymptotic MSE (mean squared error) of the kernel conditional quantile estimator $\hat{\xi}_p(x)$, that is

$$\begin{aligned} MSE = E\left(\hat{\xi}_p(x) - \xi_p(x)\right)^2 &= [E(\hat{\xi}_p(x) - \xi_p(x))]^2 + Var(\hat{\xi}_p(x) - \xi_p(x)) \\ &\approx h^4 C^2(\xi_p(x), x) + \frac{\kappa p(1-p) f(x)}{nhf^2(\xi_p(x), x)} \end{aligned}$$

Then we can take the derivative of MSE regarding the bandwidth h and set it to zero, i.e. $\frac{\partial MSE}{\partial h} = 0$, to get the optimal bandwidth h that minimizes the MSE. So we can get the optimal bandwidth h_p for a given percentile p:

$$h_p^5 = \frac{\kappa p(1-p) f(x)}{4nf^2(\xi_p(x), x) \ C^2(\xi_p(x), x)}$$

Since $p = F(\xi_p(x)|x)$ and $\frac{\partial F(\xi_p(x)|x)}{\partial \xi} = f(\xi_p(x)|x)$, we can take first and second derivatives:

$$\begin{cases} 0 &= \frac{\partial F(\xi_p(x)|x)}{\partial x} + f(\xi_p(x)|x)\xi_p'(x) \\ 0 &= \frac{\partial^2 F(\xi_p(x)|x)}{\partial x^2} + 2\frac{\partial f(\xi_p(x)|x)}{\partial x}\xi_p'(x) + f(\xi_p(x)|x)\xi_p''(x) \end{cases}$$

Then by definition of $C(\xi_p(x), x)$, we can derive that:

$$C(\xi_p(x), x) = -\xi_p''(x) - 2\left[\frac{\partial f(\xi_p(x)|x)}{\partial x}\frac{1}{f(\xi_p(x)|x)} + \frac{f'(x)}{f(x)}\right]\xi_p'(x)$$

We can assume that the leading bias term of $C(\xi_p(x), x)$ is dominated by $-\xi_p''(x)$, which may provide us an applicable bandwidth selection rule. Because the curvature of quantiles will be similar, so we have $\xi_{p_1}''(x)^2 \approx \xi_{p_2}''(x)^2$. Then by the approximation of $f(\xi_p(x)|x) \approx \sigma_x^{-1}\phi(\Phi^{-1}(p))$, we can get the ratio between two bandwidths for different percentiles of p:

$$\frac{h_{p_1}^5}{h_{p_2}^5} = \frac{p_1(1-p_1)\ \xi_{p_2}''(x)^2\ f^2(\xi_{p_2}(x)|x)}{p_2(1-p_2)\ \xi_{p_1}''(x)^2\ f^2(\xi_{p_1}(x)|x)} \approx \frac{p_1(1-p_1)\ \phi(\Phi^{-1}(p_2))^2}{p_2(1-p_2)\ \phi(\Phi^{-1}(p_1))^2}$$

If we set $p_2 = 0.5$, then we can get the optimal bandwidth for a given percentile p, which is

$$h_p^5 = \frac{p(1-p)\ \phi(\Phi^{-1}(0.5))^2}{0.5^2\ \phi(\Phi^{-1}(p))^2}h_{0.5}^5 = \frac{p(1-p)/2\pi}{0.5^2\ \phi(\Phi^{-1}(p))^2}h_{0.5}^5 = \frac{2p(1-p)}{\pi\phi(\Phi^{-1}(p))^2}h_{0.5}^5$$

Because the median value of a distribution is usually close to its mean value, thus the optimal bandwidth for median $p = 0.5$ can be approximated by the opitmal bandwidth for the general kernel (mean) regression that is mentioned in section 2.1.3. Therefore, as long as we can get an optimal bandwidth $h_{0.5}$ for general kernel (mean) regression using the same data, then we can derive the optimal bandwidth h_p for kernel conditional quantile estimation for other percentiles. On the other hand, we can observe that $h_p > h_{0.5}$ for all $p \neq 0.5$, since $\frac{2p(1-p)}{\pi\phi(\Phi^{-1}(p))^2} \geq 1$ and equals to one when $p = 0.5$. This shows that the optimal bandwidth for estimating the conditional quantile for $p \neq 0.5$ is larger than that for the median ($p = 0.5$) quantile, so it needs more smoothness than the median quantile.

However, the bandwidth selection for kernel density estimation or general kernel regression is another big topic that has been extensively studied on using different methods. For example, the optimal bandwidth can be selected using the leave-one-out likelihood cross-validation method. In this paper, we use the direct plug-in method to select the bandwidth of a local linear Gaussian kernel regression estimate, as described by Ruppert, Sheather and Wand (1995). This method is implemented by dpill() function in R package **KernSmooth**.

2.4 Bias Correction

Because when estimating VaR, we are estimating the extreme quantiles such as $p = 1\%$ and 5%. This will introduce bias into the estimator. On the other hand, the kernel method itself would have estimation bias. Note that by Corollary 1 (CLT), we have

$$bias = E[\hat{\xi}_p(x) - \xi_p(x)] \approx h^2 C(\xi_p(x), x)$$

To eliminate this bias term, we can introduce a jack-knife-type bias correction scheme:

$$\tilde{\xi}_p(x) = 2 \times \hat{\xi}_{p,h}(x) - \hat{\xi}_{p,\sqrt{2}h}(x)$$

since $2 \times h^2 C(\xi_p(x), x) - (\sqrt{2}h)^2 C(\xi_p(x), x) = 0$.

2.5 Second Smoothing

Recall that we search for optimal $\hat{\xi}_p(x)$ within $\xi \in \{Z_1, Z_2, \cdots, Z_n\}$. Due to the finite sample sizes and the discrete values of the samples, we will get an piecewise-constant-function like estimation line, thus it needs a second smoothing. Here we also choose kernel-based methods for smoothing the estimated quantile line bacause the optimal bandwidth h_p for the given percentile p has been computed, so it can be easily used in the kernel-based smoothing.

Generally, kernel-based smoothing methods includes kernel regression smoothing and local polynomial smoothing. The kernel regression smoothing is just the Nadaraya-Watson kernel regression estimate that we have mentioned before in section 2.1.3.

On the other hand, according to Wand and Jones (1995), the local polynomial smoothing uses weighted least squares with weights $w_i(x) = K_h(x - X_i)$. For each x, find $\hat{\beta}_0(x)$ and $\hat{\beta}_1(x)$ that minimizes the weighted residual sum of squares (RSS):

$$RSS = \sum_{i=1}^{n} K_h(x - X_i)(Y_i - \beta_0 - \beta_1 X_i)^2 = \sum_{i=1}^{n} w_i(x)(Y_i - \beta_0 - \beta_1 X_i)^2$$

And then we get the estimate $\hat{g}(x) = \hat{\beta}_0(x) + \hat{\beta}_1(x) \cdot x$ for x. So it needs to do this estimation for each value of x, which takes a longer time than the kernel regression smoothing method.

In this paper, we will use both of these two kernel-based smoothing methods to compare their performances.

2.6 Algorithm Summary

Here is a brief summary of our kernel conditional quantile estimation algorithm for estimating $\xi_p(x)$:

(1) **Kernel**: select a symmetric kernel function K (we use Gaussian kernel).

(2) **Bandwidth Selection**: select optimal bandwidth $h_{0.5}$ by kernel (mean) regression estimation method.

(3) **Bandwidth Calculation**: calculate the optimal bandwidth h_p for a given percentile p.

(4) **Kernel Conditional Quantile Estimation** to get $\hat{\xi}_{p,h_p}(x)$:

- Set up a grid values of $x \in [\min(X_i), \max(X_i)]$.
- For each $X = x$, among all $\xi \in \{Z_1, Z_2, \cdots, Z_n\}$, find an optimal $\hat{\xi}_p(x)$ such that $|F_n(\hat{\xi}_p(x)|x) - p|$ is the smallest.

(5) **Bias Correction**: Repeat the estimation one more time with bandwidth $\sqrt{2}h_p$ instead of h_p to get $\hat{\xi}_{p,\sqrt{2}h_p}(x)$, and get the jack-knife-type bias corrected $\tilde{\xi}_p(x)$.

(6) **Second Smoothing**: apply the kernel regression smoothing or the local polynomial smoothing method to get the final smoothed estimator $\xi^*_p(x)$.

3 Simulation Study

3.1 TAR Model

The TAR (threshold autoregressive) model is proposed by Tong and Lim (1980) for describing periodic time series. The TAR model is one of Model 2-type model where the sequence $\{Z_t\}$ has its own dependence structure. The model we used to generate the simulation data is

$$Z_t = \begin{cases} a\ Z_{t-1} + \varepsilon_t, & Z_{t-1} \geq \alpha \\ b\ Z_{t-1} + \varepsilon_t, & Z_{t-1} < \alpha \end{cases} = (a \cdot \mathbf{1}_{Z_{t-1} \geq \alpha} + b \cdot \mathbf{1}_{Z_{t-1} < \alpha})\ Z_{t-1} + \varepsilon_t, \qquad t \in \mathbb{Z}$$

where $\varepsilon_t \sim N(0,1)$, $a = 0.8$, $b = 1.2$, and $\alpha = 1$, whose selections provide us a relatively sharp change of $E(Z_t|Z_{t-1})$ at the threshold value α. Then we set $X_t = Z_{t-1}$, thus the conditional distribution of Z_t given $X_t = x$ is $N(ax \cdot \mathbf{1}_{x \geq \alpha} + bx \cdot \mathbf{1}_{x<\alpha}, 1)$. A burn-in size of 100 data points is used to eliminate the effect of the initial value. Finally we generate a sequence of 500 observations of Z_t, which is shown in Figure 1. According to the distribution of $Z_t|X_t = x$, we can derive the theoretical value of the p-th conditional quantile of $Z_t|X_t = x$, which is

$$\xi_p(x) = ax \cdot \mathbf{1}_{x \geq \alpha} + bx \cdot \mathbf{1}_{x<\alpha} + \Phi^{-1}(p)$$

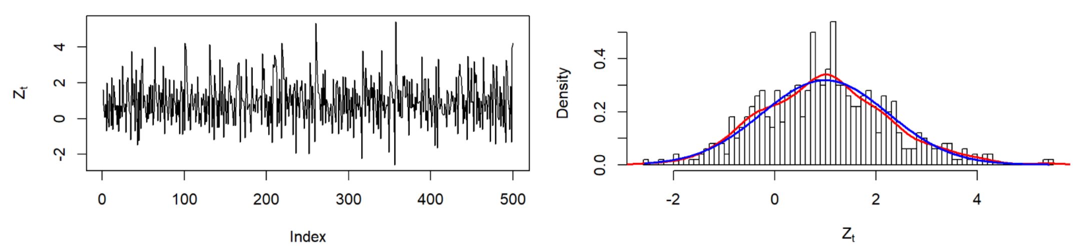

Figure 1: The sequence (left) and distribution (right) of Z_t

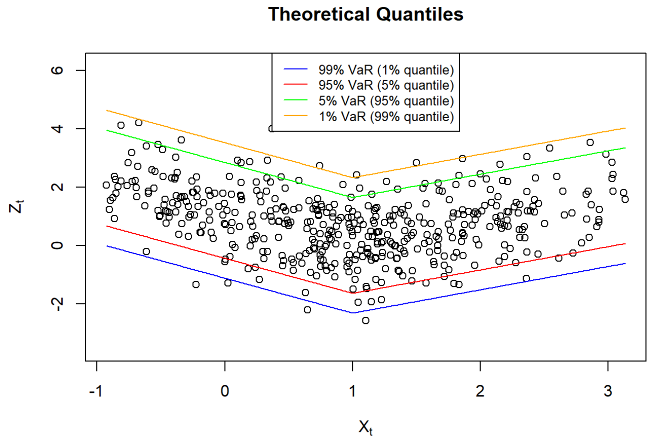

Figure 2: The theoretical quantiles $\xi_p(x)$

The theoretical p-th conditional quantiles of $Z_t|X_t$ for $p = 1\%, 5\%, 95\%$, and 99% in Figure 2 show that the mean of $Z_t|X_t$ changes as the value of X_t changes, but the variance (or volatility) roughly keeps the same as a constant.

Then we look at the results of the original estimation and the smoothed estimations in Figure 3. We can see that the kernel conditional quantile estimations for $p = 1\%$ and 5% are all generally close to their corresponding theoretical values, while the estimation of the 5% quantile (95% VaR) is slightly better than that of the 1% quantile (99% VaR). Among the three second smoothing methods, the kernel regression smoothing is the worst since its relatively worse behaviour at the edges of the X range, which is a disadvantage of the kernel regression smoothing method. And the local polynomial smoothing with degree 1 is better than that with degree 2 because local linear regression better estimates the majority linear parts the theoretical quantiles. In addition, the local quaduatic regression smoothing has a crossing point at the right edge of the plot, which also makes it a worse estimation than the linear one.

To numerically measure the performance of the estimation results, we use the Discrete Mean Absolute Error (DMAE), which is defined as:

$$DMAE(p) = \frac{1}{N} \sum_{i=1}^{N} \left| \tilde{\xi}_p(x_i) - \xi_p(x_i) \right|$$

where $N = 1000$ is the number of grid values of $x \in [\min(X_i), \max(X_i)]$ in our algorithm. Then, we repeat such simulation for $S = 100$ times, and take the average DMAE(p) as the final result:

$$aDMAE(p) = \frac{1}{S} \sum_{j=1}^{S} DMAE(p)_j$$

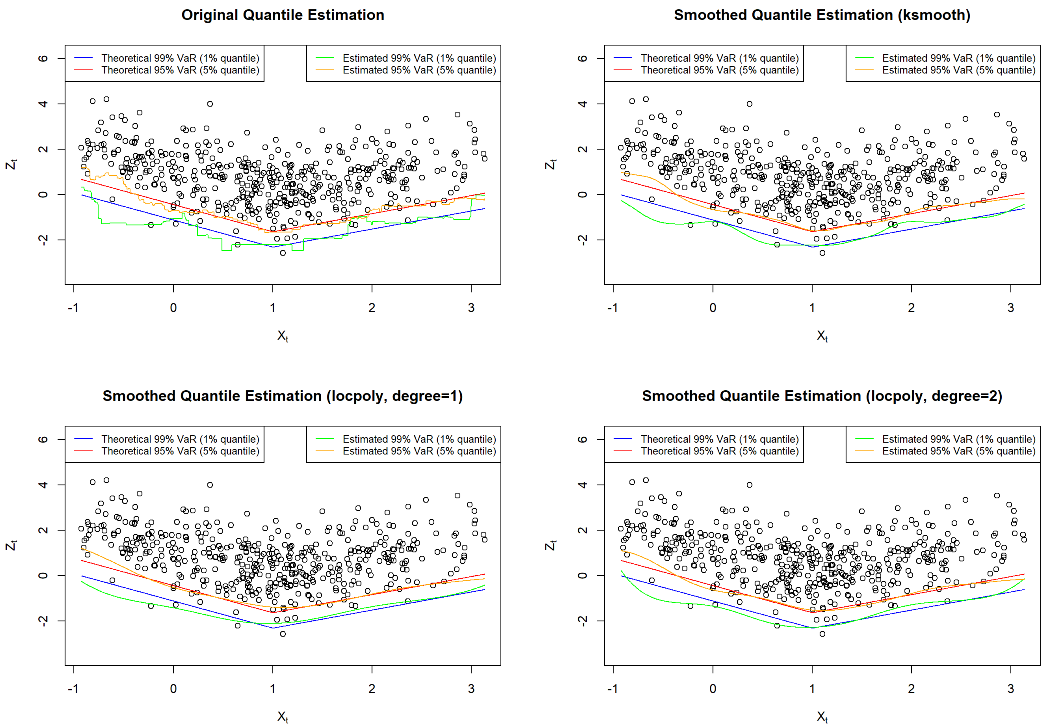

Figure 3: The original and smoothed conditional quantile estimations with different smoothing methods (upright: kernel regression smoothing, bottom: linear (left) and quadratic (right) local polynomial smoothings)

Table 1 shows the average DMAE(p) for TAR model. The red numbers indicate the lowest ones for each case, while the blue numbers indicate the second lowest ones. We can see that the local polynomial smoothing with degree 1 is the best among the three smoothing methods, and local polynomial smoothing with degree 2 is the second. This result confirmed our findings in the estimation figures. Also, with sample size increases, the estimation error gets lower. Here we choose 250 as a base number, since it is the similar number of business days in one year. While estimating VaR, one would usually use 500, about 2-year, data points to estimate the historical quantiles. On the other hand, the estimation result is better for 5% quantile than for 1% quantile.

Table 2 shows the ranges of the sample values and the theoretical values for each simulation study group. We can see that, comparing these numbers, the average DMAE(p) is relatively small. Therefore, the kernel conditional quantile estimation is reasonably well with relatively low errors.

Table 1: aDMAE(p) for TAR Model

		ksmooth	**locpoly(d=1)**	**locpoly(d=2)**
n=250	p=0.01	0.3839	0.3367	0.3770
n=250	p=0.05	0.2839	0.2444	0.2775
n=500	p=0.01	0.3314	0.2825	0.3218
n=500	p=0.05	0.2172	0.1873	0.2114
n=1000	p=0.01	0.2517	0.2124	0.2439
n=1000	p=0.05	0.1652	0.1420	0.1611

Table 2: Range of Z_t and $\xi_p(x)$ of TAR Model

		range of $\{Z_t\}_{t=1}^n$	**range of** $\{\xi_p(x_i)\}_{i=1}^N$
n=250	p=0.01	[-3.2757, 5.5025]	[-2.3263, 0.4833]
n=250	p=0.05	[-3.2757, 5.5025]	[-1.6448, 1.1648]
n=500	p=0.01	[-3.5225, 5.6298]	[-2.3263, 0.3372]
n=500	p=0.05	[-3.5225, 5.6298]	[-1.6448, 1.0187]
n=1000	p=0.01	[-4.0900, 6.1246]	[-2.3263, 0.2689]
n=1000	p=0.05	[-4.0900, 6.1246]	[-1.6449, 0.9504]

3.2 ARCH Model

Robert (1982) introduced the ARCH (autoregressive model with conditional heteroscedasticity) models to analyze time-varying volatility, where the conditional variance of returns at time t is a linear function of the past squared observations. The ARCH model is also one of Model 2-type model as the TAR model is, where the sequence $\{Z_t\}$ has its own dependence structure. The model we used to generate the simulation data is

$$Z_t = \varepsilon_t \sqrt{\omega + \alpha\ Z_{t-1}^2}, \qquad t \in \mathbb{Z}$$

where $\varepsilon_t \sim N(0,1)$, $\omega = 0.4$, and $\alpha = 0.9$, whose selections provide us a distribution with heavier tails than a normal distribution. Then we set $X_t = Z_{t-1}$, thus the conditional distribution of Z_t given $X_t = x$ is $N(0, \omega + \alpha x^2)$. A burn-in size of 100 data points is used to eliminate the effect of the initial value. Finally we generate a sequence of 500 observations of Z_t, which is shown in Figure 4. The heavy tails are also presented in the following distribution of the data Z_t. According to the distribution of $Z_t|X_t = x$, we can derive the theoretical value of the p-th conditional quantile of $Z_t|X_t = x$, which is

$$\xi_p(x) = \Phi^{-1}(p)\ \sqrt{\omega + \alpha x^2}$$

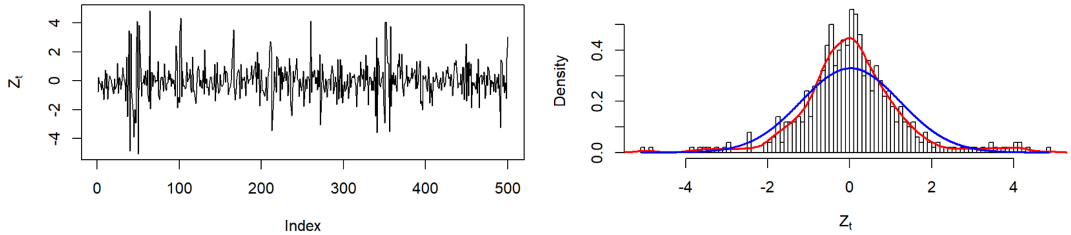

Figure 4: The sequence (left) and distribution (right) of Z_t

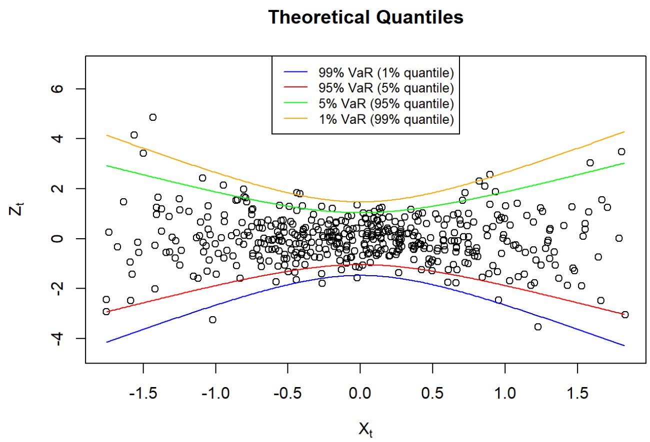

Figure 5: The theoretical quantiles $\xi_p(x)$

The theoretical p-th conditional quantiles of $Z_t|X_t$ for $p = 1\%, 5\%, 95\%$, and 99% in Figure 5 show that the mean of $Z_t|X_t$ roughly keeps the same at zero while its variance (or volatility) changes as the value of X_t changes.

Then we look at the results of the original estimation and the smoothed estimations in Figure 6. We can see that the kernel conditional quantile estimations for $p = 1\%$ and 5% are all also close to their corresponding theoretical values, while the estimation of the 5% quantile (95% VaR) is better than that of the 1% quantile (99% VaR). Among the three second smoothing methods, the kernel regression smoothing is worse in the linear parts and the edges of the X range, while the local polynomial smoothing with degree 1 is worse in the curved middle part. And the local polynomial smoothing with degree 2 is better in most parts but has crossing point at the right edge of the plot. As a result, none of the three methods seems to be significantly better than the other two ones.

We also numerically measure the performance of the estimation results with the average DMAE(p). Table 3 shows the average DMAE(p) for ARCH model. The red numbers indicate the lowest ones for each case, while the blue numbers indicate the second lowest ones. We can see that the local polynomial smoothing with degree 2 is the best among the three smoothing methods, and the other two methods have similar results. Also, with sample size increases, the estimation error gets lower, and the estimation result is better for 5% quantile than for 1% quantile. Table 4 shows the ranges of the sample values and the theoretical values for each simulation study group. We can see that, comparing these numbers, the average DMAE(p) is relatively small. Therefore, the kernel conditional quantile estimation is reasonably well with relatively low errors.

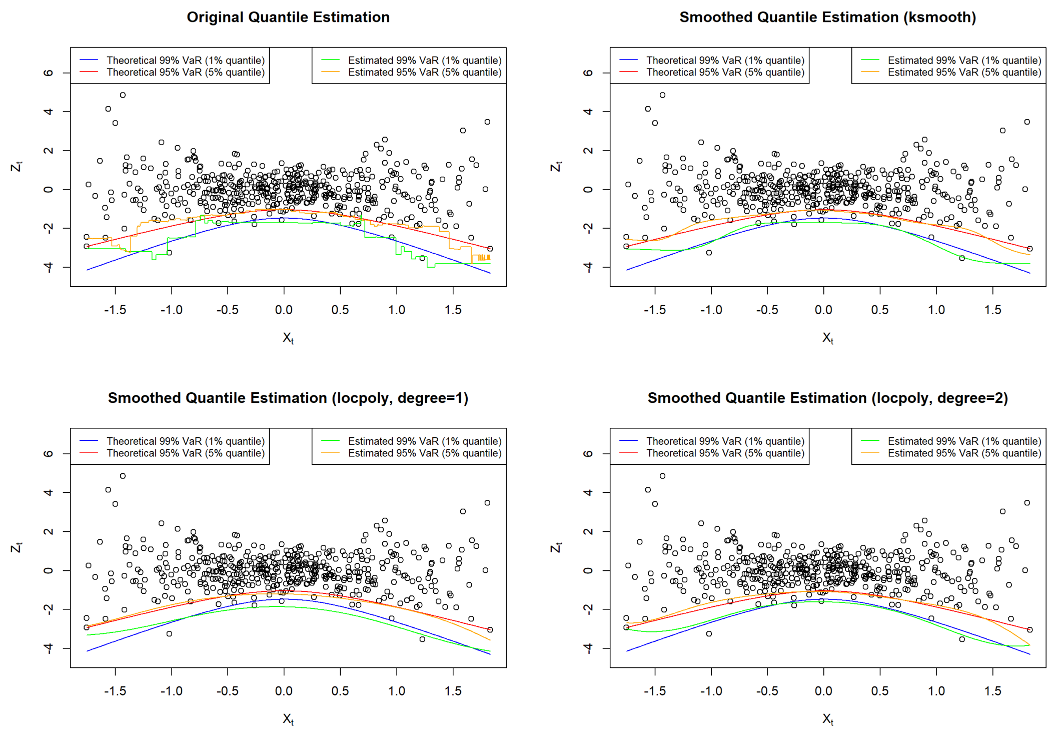

Figure 6: The original and smoothed conditional quantile estimations with different smoothing methods (upright: kernel regression smoothing, bottom: linear (left) and quadratic (right) local polynomial smoothings)

Table 3: aDMAE(p) for ARCH Model

		ksmooth	**locpoly(d=1)**	**locpoly(d=2)**
n=250	p=0.01	0.6185	0.6202	0.6032
n=250	p=0.05	0.4270	0.4199	0.4188
n=500	p=0.01	0.4984	0.4964	0.4789
n=500	p=0.05	0.3388	0.3320	0.3344
n=1000	p=0.01	0.4007	0.4216	0.3849
n=1000	p=0.05	0.2415	0.2420	0.2359

Table 4: Range of Z_t and $\xi_p(x)$ of ARCH Model

		range of $\{Z_t\}_{t=1}^{n}$	**range of** $\{\xi_p(x_i)\}_{i=1}^{N}$
n=250	p=0.01	[-7.0104, 6.6241]	[-8.6332, -1.4713]
n=250	p=0.05	[-7.0104, 6.6241]	[-6.1041, -1.0403]
n=500	p=0.01	[-8.2047, 7.5443]	[-9.3979, -1.4713]
n=500	p=0.05	[-8.2047, 7.5443]	[-6.6449, -1.0403]
n=1000	p=0.01	[-7.7339, 6.5270]	[-6.3188, -1.4713]
n=1000	p=0.05	[-7.7339, 6.5270]	[-4.4677, -1.0403]

3.3 SV Model

According to Kim, Neil and Siddhartha (1998), SV (stochastic volatility) models treat the volatility (i.e. variance) of a return on an asset, such as an option to buy a security, as a latent stochastic process in discrete time. Different from the previous two models, the SV model is one of Model 1-type model, where the sequence $\{X_t\}$ has a own dependence structure and Z_t is a function of X_t. The model we used to generate the simulation data is

$$Z_t = \varepsilon_t \exp(0.5X_t), \quad X_t = a + b\ X_{t-1} + c\ \eta_t, \qquad t \in \mathbb{Z}$$

where $\varepsilon_t \sim N(0,1)$, $\eta_t \sim N(0,1)$, $a = 0.2$, $b = 0.6$, and $c = 0.9$, whose selections provide us a distribution with heavier tails than a normal distribution. Then the conditional distribution of Z_t given $X_t = x$ is $N(0, \exp(x))$. A burn-in size of 100 data points is used to eliminate the effect of the initial value. Finally we generate a sequence of 500 observations of Z_t, which is shown in Figure 7. The heavy tails are also presented in the following distribution of the data Z_t. According to the distribution of $Z_t|X_t = x$, we can derive the theoretical value of the p-th conditional quantile of $Z_t|X_t = x$, which is

$$\xi_p(x) = \Phi^{-1}(p)\ \exp(0.5x)$$

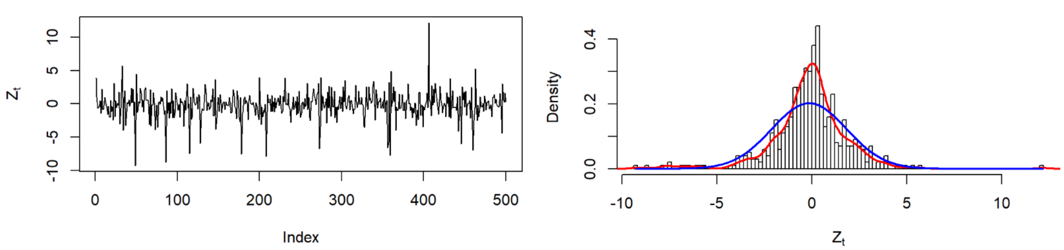

Figure 7: The sequence (left) and distribution (right) of Z_t

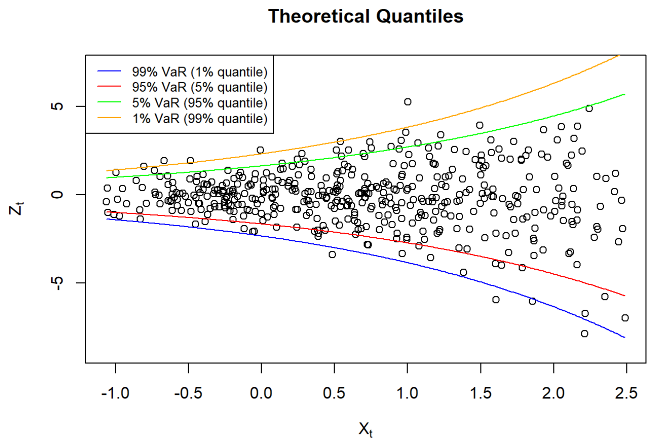

Figure 8: The theoretical quantiles $\xi_p(x)$

The theoretical p-th conditional quantiles of $Z_t|X_t$ for $p = 1\%, 5\%, 95\%$, and 99% in Figure 8 show that the mean of $Z_t|X_t$ roughly keeps the same at zero while its variance (or volatility) changes as the value of X_t changes.

Then we look at the results of the original estimation and the smoothed estimations in Figure 9. We can see that the kernel conditional quantile estimations for $p = 1\%$ and 5% are close to their corresponding theoretical values when the volatility is lower on the left-hand side, but it gradually becomes away from the theoretical values when the volatility is larger on the right-hand side. Among the three second smoothing methods, the kernel regression smoothing is still the worst. And the local polynomial smoothing with degree 1 is slightly better than that with degree 2.

We also numerically measure the performance of the estimation results with the average DMAE(p). Table 5 shows the average DMAE(p) for SV model. The red numbers indicate the lowest ones for each case, while the blue numbers indicate the second lowest ones.We can see that the local polynomial smoothing with degree 1 is the best among the three smoothing methods, and local polynomial smoothing with degree 2 is the second. This result confirmed our findings in the estimation figures. Also, with sample size increases, the estimation error gets lower, and the estimation result is better for 5% quantile than for 1% quantile. Table 6 shows the ranges of the sample values and the theoretical values for each simulation study group. We can see that, comparing these numbers, the average DMAE(p) is relatively small. Therefore, the kernel conditional quantile estimation is reasonably well with relatively low errors.

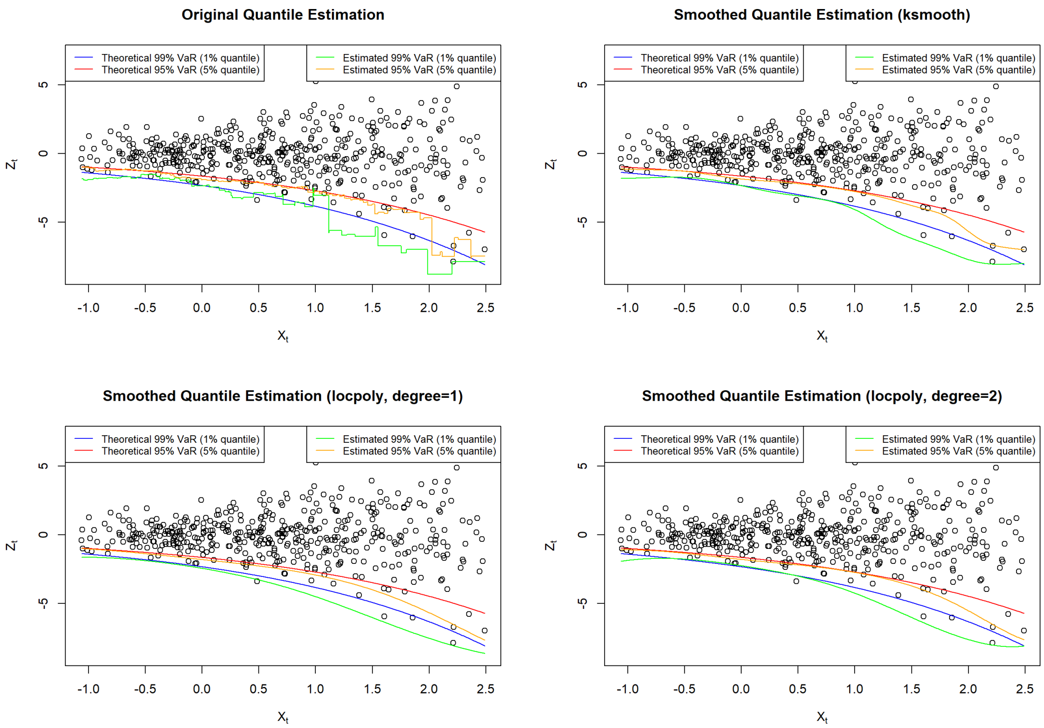

Figure 9: The original and smoothed conditional quantile estimations with different smoothing methods (upright: kernel regression smoothing, bottom: linear (left) and quadratic (right) local polynomial smoothings)

Table 5: aDMAE(p) for SV Model

		ksmooth	**locpoly(d=1)**	**locpoly(d=2)**
n=250	p=0.01	0.6567	0.5928	0.6314
n=250	p=0.05	0.4175	0.3833	0.4103
n=500	p=0.01	0.5360	0.4893	0.5136
n=500	p=0.05	0.3153	0.2835	0.3064
n=1000	p=0.01	0.3939	0.3751	0.3764
n=1000	p=0.05	0.2173	0.1980	0.2100

Table 6: Range of Z_t and $\xi_p(x)$ of SV Model

		range of $\{Z_t\}_{t=1}^n$	**range of** $\{\xi_p(x_i)\}_{i=1}^N$
n=250	p=0.01	[-11.5287, 10.0761]	[-9.0471, -0.8967]
n=250	p=0.05	[-11.5287, 10.0761]	[-6.3968, -0.6340]
n=500	p=0.01	[-9.6945, 11.2705]	[-9.4966, -1.0031]
n=500	p=0.05	[-9.6945, 11.2705]	[-6.7146, -0.7093]
n=1000	p=0.01	[-10.9458, 11.9877]	[-8.6332, -1.0262]
n=1000	p=0.05	[-10.9458, 11.9877]	[-6.1041, -0.7256]

4 Real Data Analysis

In this section, we carry out real data analysis using the kernel conditional quantile estimation method on several stock or index data. The data are all downloaded from Yahoo! Finance. To cover different types of stocks, we select the stocks from three categories, including general indexes, tech stocks and commodity stocks. The selected stock/index names with their corresponding company names are shown in Table 7.

Table 7: Stock/Index Names

type	**stock name**	**company/index name**
index	SPY	S&P 500 Index
index	QQQ	Invesco QQQ Trust, Nasdaq 100 Index
tech	GOOG	Alphabet Inc.
tech	AAPL	Apple Inc.
commodity	MCD	McDonald's Corporation
commodity	TGT	Target Corporation

(1) The SPY (S&P 500) is a stock market index that tracks the stocks of 500 large-cap U.S. companies. It represents the stock market's performance by reporting the risks and returns of the biggest companies. And QQQ tracks the Nasdaq 100 Index. Its focus is on large international and U.S. companies in the technology, health care, industrial, consumer discretionary, and telecommunications sectors. These two indexes can basically reflect the overall market in a general way.

(2) Google and Apple are selected being representatives of technology companies, whose stocks would usually fluctuate more frequently and/or dramatically than other industries' stocks.

(3) McDonald's and Target are selected for commodity companies, whose stocks would generally fluctuate less than the technology companies' stocks.

Both the daily and monthly stock data are downloaded for each stock. The daily stock data are all from 2017-01-01 to 2020-05-01, while the monthly data for each stock is from the first recorded date of that stock on Yahoo! Finance. For each stock data, first the daily/-monthly stock return is calculated. And then the cVaR estimation using kernel conditional quantiel estimation is carried out for both daily and monthly data. Estimations of conditional quantiles for $p = 1\%$ (cVaR 99) and 5% (cVaR 95) are implemented on daily stock data for three different time periods: 1-year(2019), 2-year(2018-2019), over 3-year(2017 to now, i.e. 2020-05-01), respectively. And estimations for monthly stock return data is applied to the whole monthly dataset due to its less number of data points than the daily data. In addition, we use linear and quadratic local polynomial smoothings as the second smoothing methods because they are performing better than the kernel regression smoothing methods in the simulation study. Here we choose one stock in each category to show the estimation results, which are QQQ, AAPL, MCD. The estimation results of the other three stocks are shown in the Appendix.

Figure 10 shows the sequences and estimation results of QQQ index. The fluctuations of the sequences reflect the general trends of the economy. The monthly data of QQQ is from March 1999, thus we have n = 256 data points for monthly data. It shows that the estimation of the conditional quantiles of monthly data is generally fine except for the crossings of the two quantiles at both edges of the X. For daily data, the esimations of three time periods show different shapes. Note that, according to the definitions of the cVaR, different time periods should indeed generate different cVaR values. It shows that, the estimation line is the best when using over 3 years' daily stock return data, because the plots for the other two time periods all have the crossing points of two quantiles. And, in general, the local polynomial smoothing with degree 1 is better than that of degree 2.

Figure 11 shows the sequences and estimation results of AAPL. The fluctuations of the sequences is more than that of QQQ. The monthly data of AAPL is from December 1980, thus we have n = 475 data points for monthly data. It shows that the estimation of the conditional quantiles of monthly data is generally fine for the 5% quantile, but the estimation for 1% quantile is dragged down by an outlier-like data point at the bottom of the plot. For daily data, it shows that, the estimation line is the best when using over 3 years' daily stock return data, because the plots for the other two time periods all have the crossing points of two quantiles. And, in general, the local polynomial smoothing with degree 1 is better than that of degree 2. On the other hand, we can see that there is an outlier-like data point in the plot for 1-year period which has also dragged the 1% quantile line down. However, this data point does not affect much with the 1% quantile lines in the other two plots due to larger number of data. This phenomenon shows that, our kernel conditional quantile estimation method would be affected by the outliers and extreme values. It is best to first detect outliers and influential points and then remove them from the data to make the estimation more accurate and robust.

Figure 12 shows the sequences and estimation results of MCD. The fluctuations of the sequences is the least among all the three stocks. The monthly data of MCD is from July 1966, thus we have n = 648 data points for monthly data. It shows that the estimation of the conditional quantiles of monthly data is generally fine. For daily data, it shows that, all the estimations for three time periods have crossing points at the edges of X range, though the plots for 2-year and 3-year are slightly better than that for 1-year. Based on the previous two figures for QQQ and AAPL and the estimation for MCD, we can find that the better estimation of cVaR for daily stock return should require the number of observations to be at least 500.

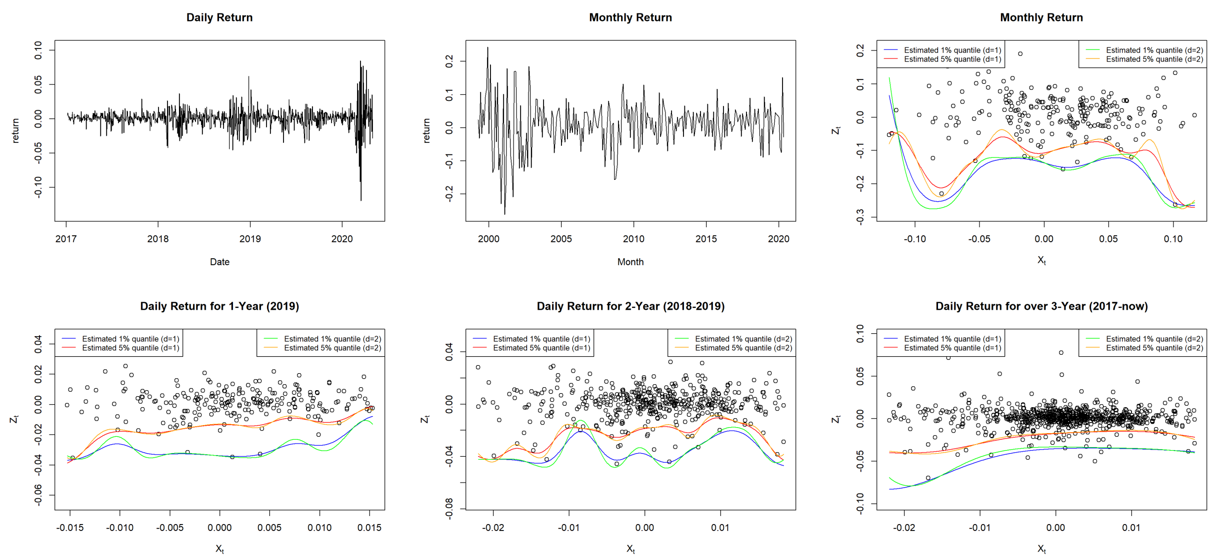

Figure 10: Daily and monthly data of QQQ and the estimations of the conditional quantiles

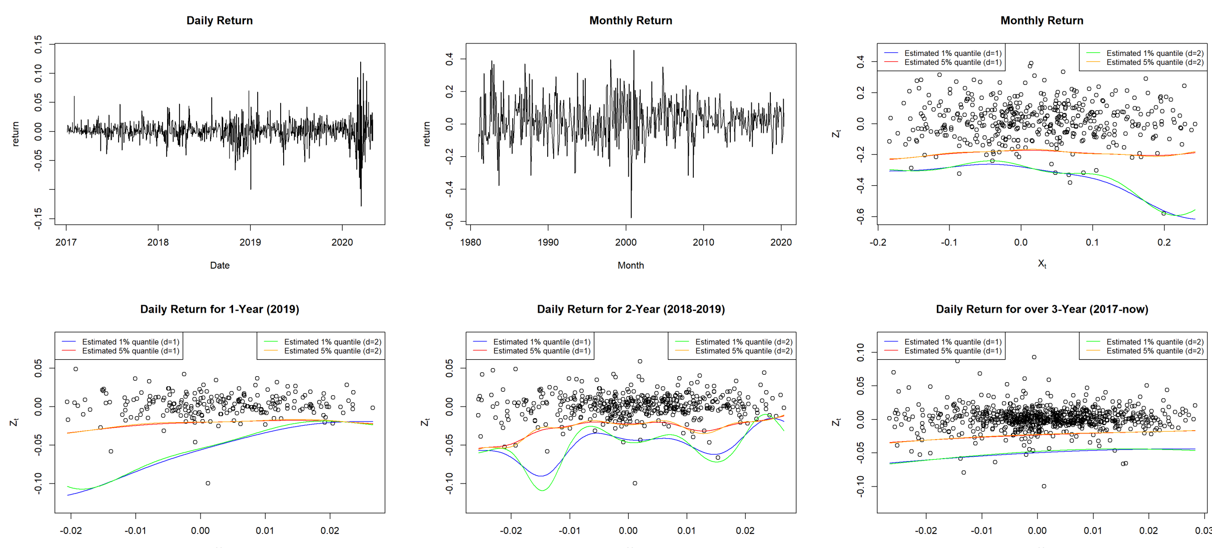

Figure 11: Daily and monthly data of AAPL and the estimations of the conditional quantiles

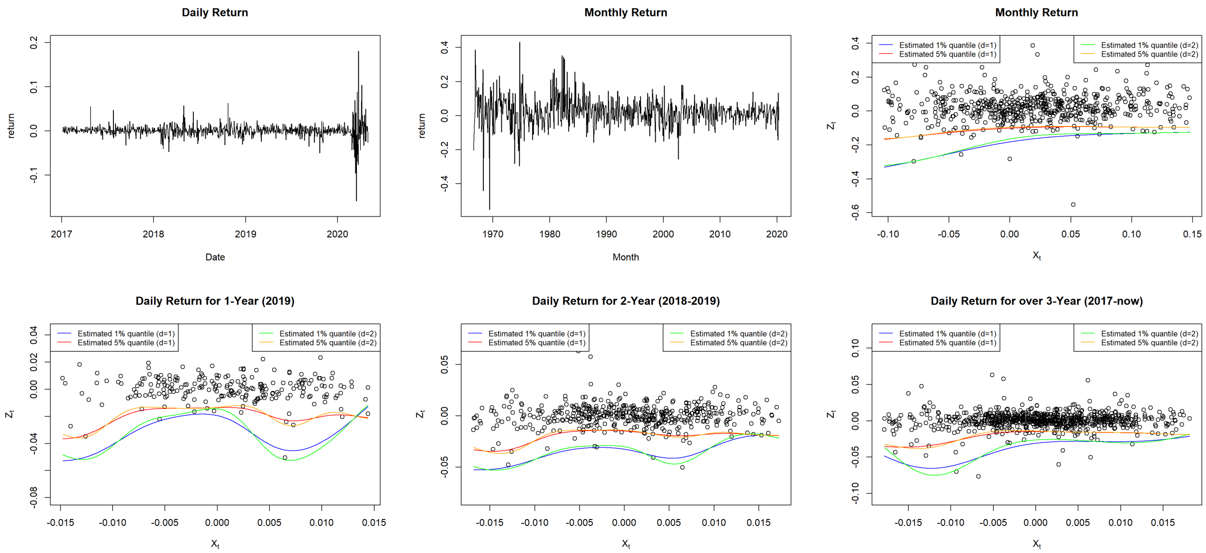

Figure 12: Daily and monthly data of MCD and the estimations of the conditional quantiles

5 Conclusion

In this paper, we mainly worked on the algorithm of kernel conditional quantile estimation, which is a good kernel-based method for estimating conditional VaR (cVaR).

- We first studied the properties of the estimator and derived the optimal bandwidth selection rule for a given percentile p. Then, we used bias correction scheme and a second smoothing to further improve the estimation.

- Later, we did simulations on three sets of weakly dependent data that are widely used in financial data analysis. The three models we used (TAR, ARCH and SV) have different model types and properties of distributions, which together show a relatively good picture of different aspects of financial data. The estimation results show that our estimation algorithm indeed do a reasonably well estimation of the extreme 1% and 5% quantiles. And the local polynomial smoothing works generally better than the kernel regression smoothing as a second smoothing method. Furthermore, the local polynomial smoothing with degree 1 is better than that of degree 2.

- In the end, we performed real data analysis on the daily and monthly stock return data that are downloaded from Yahoo! Finance. We find that the local polynomial smoothing with degree 1 is generally better than that of degree 2. And it is better to use at least 500 observations to perform the estimation algorithm.

In short, the estimation of cVaR through our kernel conditional quantile estimation algorithm works reasonably well for complex time series without making any parametric model assumptions. This is a promising feature in situations where the underlying mechanisms are unknown.

A Appendix

A.1 Codes for Estimation

The following codes show the main functions for the kernel conditional quantile estimation algorithm.

```
# functions for estimating conditional quantile
kernel_con_q_est = function(X,Z,p,h){

    # generate grids for X & Z(quantile) directions
    xx = seq(min(X), max(X), length.out=1000)
    #xx = X[order(X)]
    qq = Z[order(Z)]

    # calculate the weight wi(x) for each x in the xx
    kernel_m = outer(xx,X,"-")
    kernel_d = dnorm(kernel_m/h)  # Gaussian Kernel with bandwidth h
    W = kernel_d / rowSums(kernel_d)

    # get quantile q(x) for each x in the xx s.t. Fn(x) closest to p
    qq_est = rep(NA,length(xx))
    for (i in 1:length(xx)) {
        Fns = rep(NA,length(qq))
        for (j in 1:length(qq)) {
            Fns[j] = sum( W[i,]*(Z<=qq[j]) )
        }
        index = which.min(abs(Fns-p))
        qq_est[i] = qq[index]
    }
    return(list(xx=xx, qq_est=qq_est))
}

library(KernSmooth)
estimate_con_q = function(X,Z,p,h0.5=NULL,dpill_trim){
    # get h0.5 if not given
    if (is.null(h0.5)) {
        h0.5 = dpill(X,Z, trim=dpill_trim,
                        blockmax=ceiling(length(X)/10), divisor=10)
    }
    # calculate optimal h
    opt_h = (2/pi * p*(1-p) * dnorm(qnorm(p))^(-2) * h0.5^5) ^ (1/5)

    # estimate conditional quantile using opt_h
    qq_est1 = kernel_con_q_est(X,Z,p,opt_h)

    # estimate conditional quantile using sqrt(2)*opt_h
```

```
    qq_est2 = kernel_con_q_est(X,Z,p,sqrt(2)*opt_h)

    # bias correction
    qq_est = 2*qq_est1$qq_est - qq_est2$qq_est

    return(list(xx=qq_est1$xx,qq_est=qq_est,h0.5=h0.5,opt_h=opt_h))
}
```

A.2 Codes for Simulation

The following codes show the main functions for simulations.

```
# function for generating sample {Xt,Zt} t=1,...,n for TAR model
func = function(x){  ifelse(x>=1,0.8,1.2) * abs(x-1)  }
simulate_data = function(n_sample, cutoff=0.05){
    # generate sample {Zt}
    n_sample = n_sample + 1
    burn_in = 100
    n = n_sample + burn_in
    Z = rep(NA,n)
    Z[1] = 0
    for (i in 2:n) {
        Z[i] = func(Z[i-1]) + rnorm(1,0,1)
    }
    Z = Z[-(1:burn_in)]

    # then generate {Xt,Zt} t=1,...,n, where Xt = Zt-1
    X = Z[1:(n_sample-1)]
    Z = Z[2:n_sample]

    # cutoff the tail extreme values of each end
    cuts = as.vector(quantile(X, probs=c(cutoff,1-cutoff)))
    index = which((X>cuts[1]) & (X<cuts[2]))
    X_cut = X[index]
    Z_cut = Z[index]

    return(list(X=X, Z=Z, X_cut=X_cut, Z_cut=Z_cut, cuts=cuts))
}

# function for generating sample {Xt,Zt} t=1,...,n for ARCH model
sigma_func = function(x){   sqrt(0.4 + 0.9 * x^2)  }
simulate_data = function(n_sample, cutoff=0.05){
    # generate sample {Zt}
    n_sample = n_sample + 1
    burn_in = 100
    n = n_sample + burn_in
```

```
    Z = rep(NA,n)
    Z[1] = 0
    for (i in 2:n) {
        Z[i] = rnorm(1,0,1) * sigma_func(Z[i-1])
    }
    Z = Z[-(1:burn_in)]

    # then generate {Xt,Zt} t=1,...,n, where Xt = Zt-1
    X = Z[1:(n_sample-1)]
    Z = Z[2:n_sample]

    # cutoff the tail extreme values of each end
    cuts = as.vector(quantile(X, probs=c(cutoff,1-cutoff)))
    index = which((X>cuts[1]) & (X<cuts[2]))
    X_cut = X[index]
    Z_cut = Z[index]

    return(list(X=X, Z=Z, X_cut=X_cut, Z_cut=Z_cut, cuts=cuts))
}

# function for generating sample {Xt,Zt} t=1,...,n for SV model
simulate_data = function(n_sample, cutoff=0.05){
    # generate sample {Zt}
    n_sample = n_sample + 1
    burn_in = 100
    n = n_sample + burn_in
    Z = X = rep(NA,n)
    X[1] = 1
    for (i in 2:n) {
        X[i] = 0.2 + 0.6*X[i-1] + 0.9*rnorm(1,0,1)
        Z[i] = exp(0.5*X[i]) * rnorm(1,0,1)
    }
    X = X[-(1:(burn_in+1))]
    Z = Z[-(1:(burn_in+1))]

    # cutoff the tail extreme values of each end
    cuts = as.vector(quantile(X, probs=c(cutoff,1-cutoff)))
    index = which((X>cuts[1]) & (X<cuts[2]))
    X_cut = X[index]
    Z_cut = Z[index]

    return(list(X=X, Z=Z, X_cut=X_cut, Z_cut=Z_cut, cuts=cuts))
}
```

A.3 Figures for Stocks

The following figures show the estimation results for the other three stock data: SPY, GOOG, and TGT.

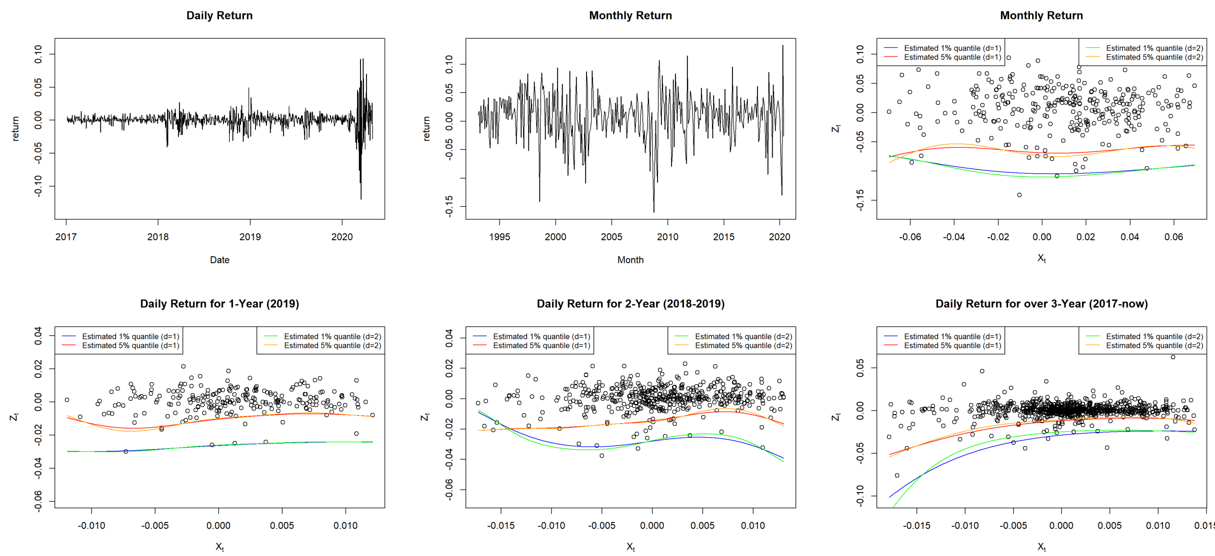

Figure 13: Daily and monthly data of SPY and the estimations of the conditional quantiles

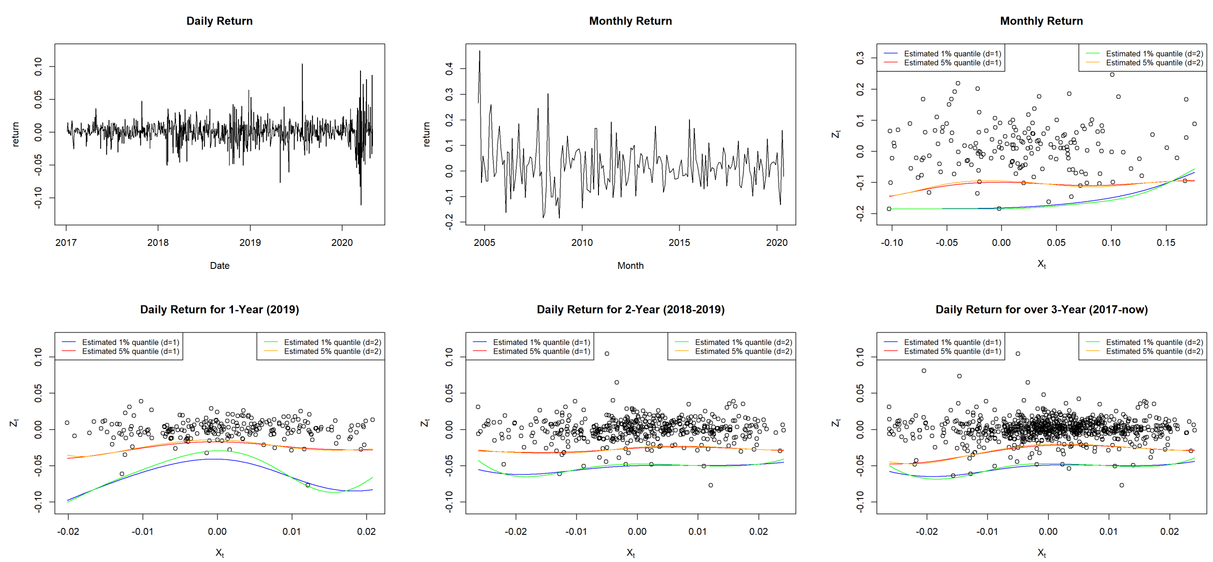

Figure 14: Daily and monthly data of GOOG and the estimations of the conditional quantiles

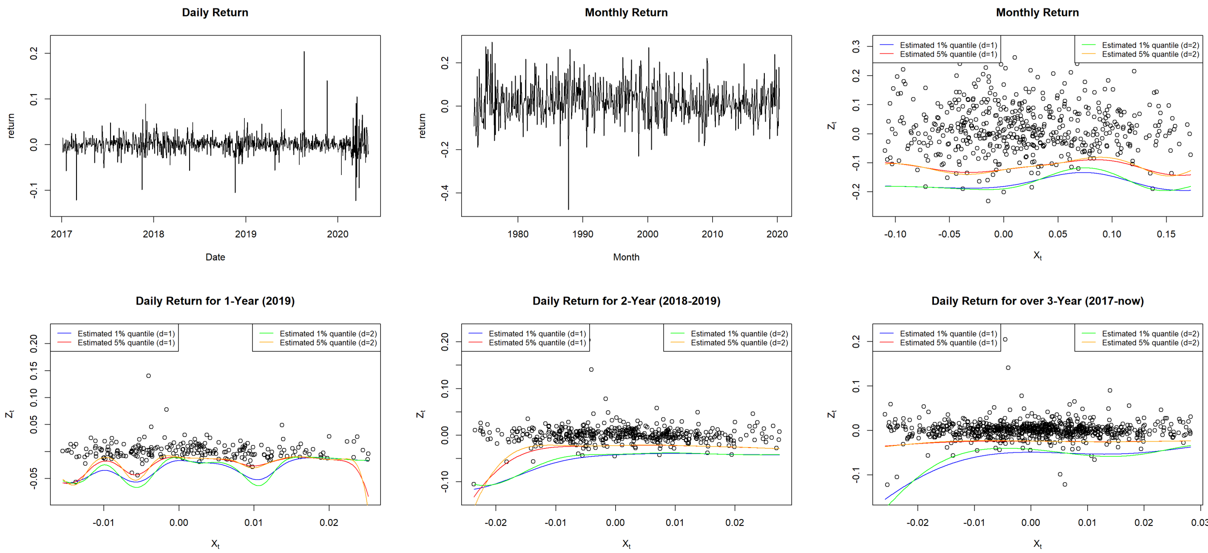

Figure 15: Daily and monthly data of TGT and the estimations of the conditional quantiles

References

[1] Bahadur, R. R. (1966). A note on quantiles in large samples. *Annals of Mathematical Statistics* 37: 577–580.

[2] Bhattacharya P. K., and Gangopadhyay A. K. (1990). Kernel and nearestneighbor estimation of a conditional quantile. *Annals of Statistics* 18: 1400–1415.

[3] Carlo A., and Dirk T. (2002). Expected Shortfall: a natural coherent alternative to Value at Risk. *Economic Notes* 31(2): 379–388.

[4] Chaudhuri, P. (1991). Nonparametric estimates of regression quantiles and their local Bahadur representation. *Annals of Statistics* 19: 760–777.

[5] Chernozhukov, V., and Umantsev, L. (2001). Conditional Value-at-Risk: aspects of modeling and estimation. *Empirical Economics* 26: 271–292.

[6] Engle, R. F. (1982). Autoregressive conditional heteroscedasticity with estimated of the variance of U.K. inflation. *Econometrica* 50: 987-1008.

[7] Kim, S., Neil S., and Siddhartha C. (1998). Stochastic Volatility: Likelihood Inference and Comparison with ARCH Models. *Review of Economic Studies* 65: 361–93.

[8] Morgan, J. P. (1996). *Value at Risk, RiskMetrics-Technical Document*. New York: Morgan Guaranty.

[9] Ruppert, D., Sheather, S. J., and Wand, M. P. (1995). An effective bandwidth selector for local least squares regression. *Journal of the American Statistical Association* 90: 1257-1270.

[10] Thomas J. L., and Neil D.P. (1999). *Risk Measurement: An Introduction to Value at Risk*. University of Illinois at Urbana-Champaign.

[11] Tong, H. and Lim, K. S. (1980). Threshold autoregression, limit cycles and cyclical data (with discussion). *Journal of the Royal Statistical Society* (B) 42: 245–292.

[12] Wand, M. P. and Jones, M. C. (1995). *Kernel Smoothing*. Chapman and Hall, London.

[13] Wu, W. B. (2005b). On the Bahadur representation of sample quantiles for stationary sequences. *Annals of Statistics* 33: 1934–1963.

[14] Wu, W. B., Keming Y., Gautam M. (2007). Kernel conditional quantile estimation for stationary processes with application to conditional value-at-risk. *Journal of Financial Econometrics* 6(2): 253–270.

[15] Yu, K., and M. C. Jones. (1998). Local linear quantile regression. *Journal of American Statistical Association* 93: 228–238.