

1 Simulation Study

We compare the performance of our quantile trend filtering method with the three previously published methods using designs proposed by Racine and Li (2017). The methods compared are

- **npqw**: Racine and Li (2017) constrain the response to follow a smooth location scale model of the form $Y_i = a(X_i) + b(X_i)\epsilon_i$. They estimate the τ_{th} conditional quantile given $X_i = x$ using a kernel estimator

$$q_\tau(x) = \frac{\sum_{i=1}^n \Phi_{(Y_i, b(X_i))}^{-1}(\delta_0) K_h(X_i, x)}{\sum_{i=1}^n K_h(X_i, x)} \quad (1)$$

defining $\Phi_{(Y_i, b(X_i))}^{-1}(\delta_0)$ as the quantile function of the Normal distribution with mean Y_i and standard deviation $b(X_i)$ evaluated at τ . δ_0 is a function of τ and chosen empirically, h is a tuning parameter and K is a kernel function. Code was obtained from the author for the **quantile-ll** method.

- **qsreg**: Oh et al. (2011) proposed a pseudo-data algorithm for a quantile spline estimator of the form

$$\sum_i \rho_\tau(y_i - g(x_i)) + \lambda \int (g''(x))^2 dx. \quad (2)$$

If $\rho_\tau(\cdot)$ were differentiable, the solution to this equation would take a form similar to that of the squared loss smoothing spline with weights equal to $\frac{\rho'_\tau(y_i - g(x_i))}{2(y_i - g(x_i))}$. Relying on this idea, Nychka proposed to solve the problem by iteratively solving the weighted smoothing spline. To address the non-differentiability they propose an approximation

$$\rho_{\tau, \delta}(u) = [\tau I(u > 0) + (1 - \alpha) I(u < 0)] u^2 / \delta \quad (3)$$

The function **qsreg** in the **fields** R package was used. The smoothing parameter is chosen automatically using generalized cross validation on the pseudo data.

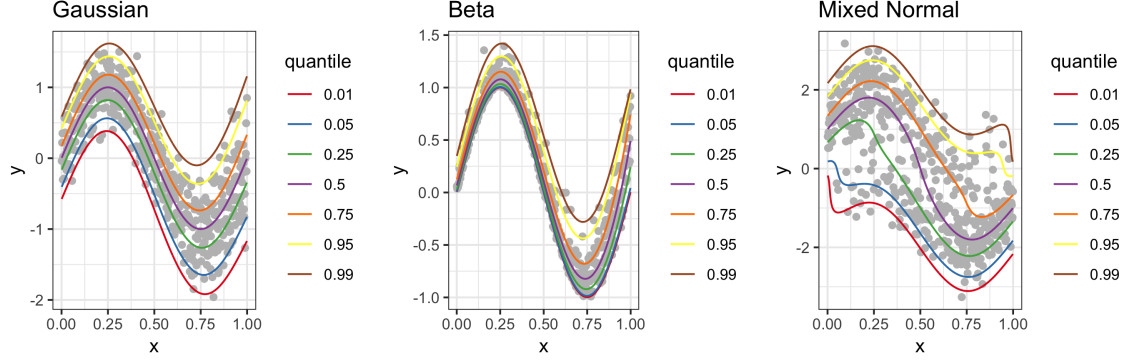
- **rqss**: Koenker et al. (1994) Koenker proposed smoothing splines using trend filtering with the second order differencing matrix which results in linear splines. The function **rqss** in the **quantreg** package implements this method. The smoothing parameter λ is chosen using a grid search and minimizing

$$SIC(p_\lambda) = \log[n^{-1} \sum \rho_\tau(y_i - \hat{g}(x_i))] + \frac{1}{2n} p_\lambda \log n \quad (4)$$

where $p_\lambda = \sum I(y_i = \hat{g}_i(x_i))$, which can be thought of as active knots.

- **detrendr_SIC**: Our method where we minimize $\sum_i \rho_\tau(y_i - \theta_i) + \lambda \|D\theta\|_1$ and λ is chosen using SIC

Figure 1: Simulated data with true quantiles $\tau \in \{0.01, 0.05, 0.25, 0.5, .75, 0.95, 0.99\}$



from above. A single value of λ was chosen by scaling and summing SIC values across all quantiles.

- **detrendr_valid:** Our method where lambda is chosen by leaving out every 5th observation as a validation data set and evaluating the check loss function on the validation data.

Three simulation designs from Racine and Li (2017) were considered. For all designs X_i was generated as a uniformly spaced sequence in $[0, 1]$ and the response Y was generated as

$$Y_i = \sin(2\pi x_i) + \epsilon_i(x_i)$$

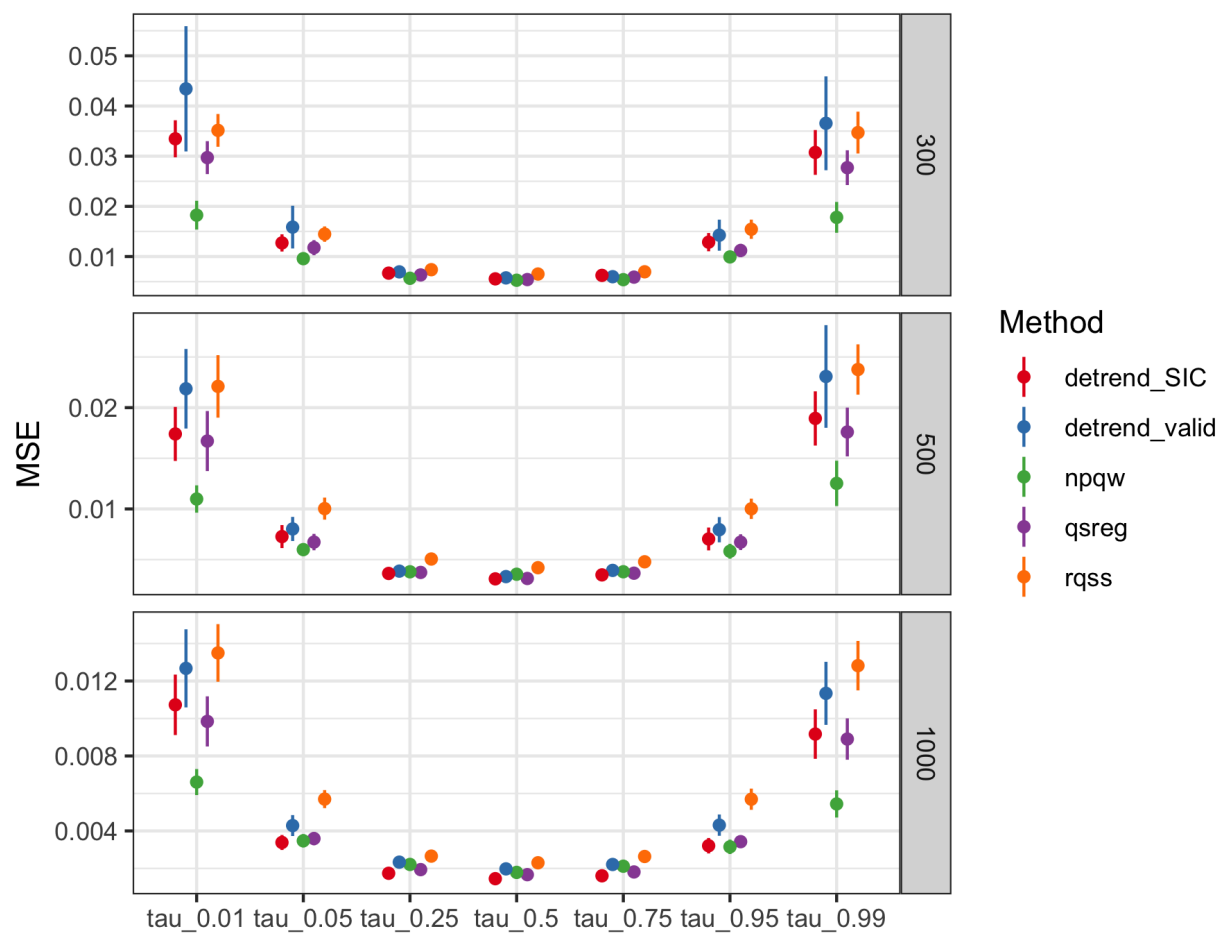
The three error distributions considered were

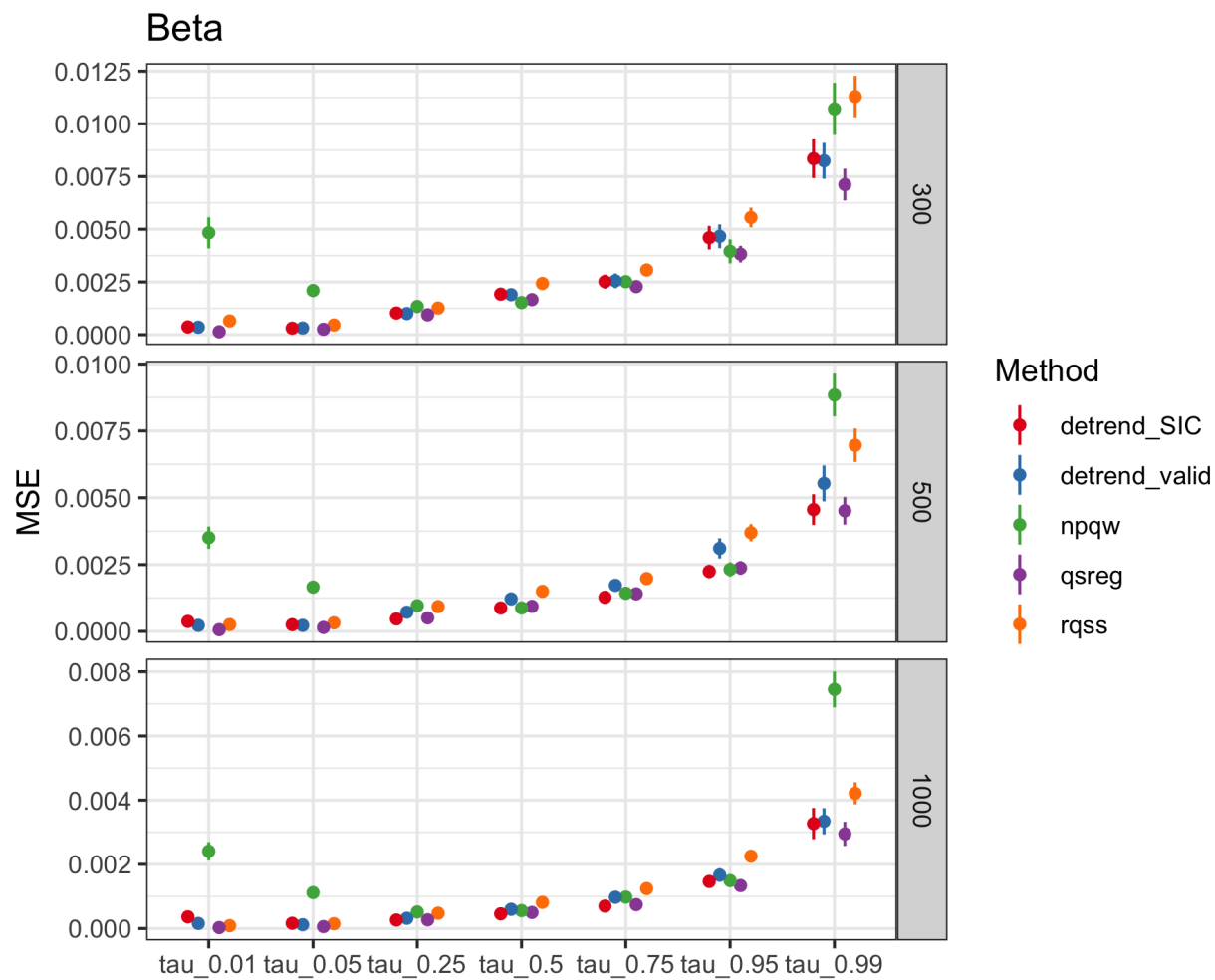
- Gaussian: $\epsilon_i(x_i) \sim N\left(0, \left(\frac{1+x_i^2}{4}\right)^2\right)$
- Beta: $\epsilon_i \sim \text{Beta}(1, 11 - 10x_i)$
- Mixed normal: ϵ_i is simulated from a mixture of $N(-1, 1)$ and $N(1, 1)$ with mixing probability x_i .

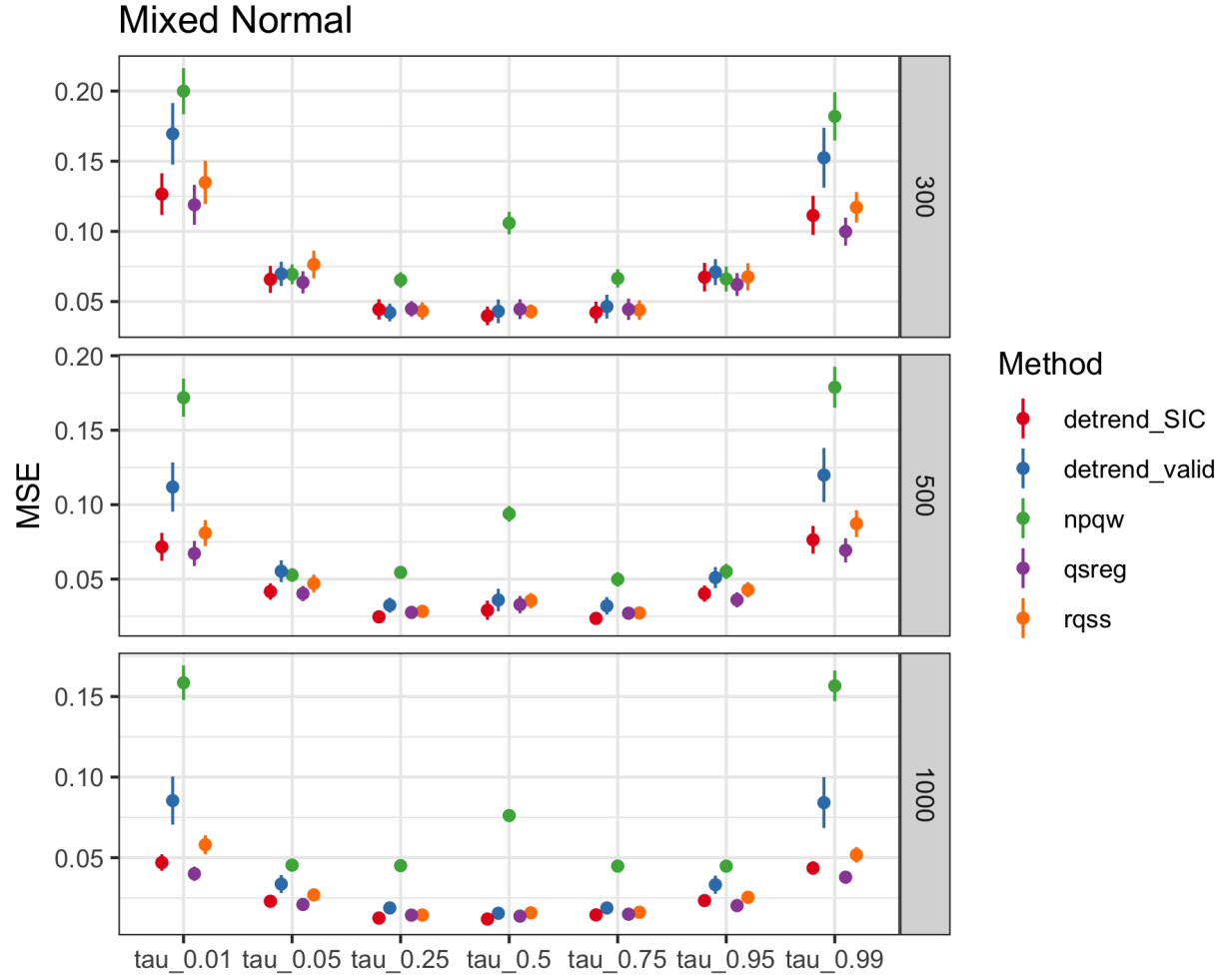
100 datasets were generated of sizes 300, 500 and 1000. The MSE was calculated as $\frac{1}{n} \sum_i (\hat{q}_\tau(x_i) - q_\tau(x_i))^2$.

The plots below show the mean MSE \pm twice the standard error by method, quantile level and sample size.

Gaussian







References

- Koenker, R., Ng, P., and Portnoy, S. (1994), “Quantile smoothing splines,” *Biometrika*, 81, 673–680.
- Oh, H.-S., Lee, T. C. M., and Nychka, D. W. (2011), “Fast Nonparametric Quantile Regression With Arbitrary Smoothing Methods,” *Journal of Computational and Graphical Statistics*, 20, 510–526.
- Racine, J. S. and Li, K. (2017), “Nonparametric conditional quantile estimation: A locally weighted quantile kernel approach,” *Journal of Econometrics*, 201, 72–94.