BSquare: An R package for Bayesian simultaneous quantile regression

Luke B Smith and **Brian J Reich**

North Carolina State University

May 21, 2013

BSquare in an R package to conduct Bayesian quantile regression for continuous, discrete, and censored data. Quantile regression provides a comprehensive analysis of the relationship between covariates and a response. In quantile regression, by specifying different covariate effects at different quantile levels we allow covariates to affect not only the center of the distribution, but also its spread and the magnitude of extreme events. Unlike most approaches to quantile regression, such as those implemented in package quantreg and bayesQR, BSquare analyzes all quantile levels simultaneously. Therefore, this approach can borrow strength across nearby quantile levels to reduce uncertainty in the estimated quantile function, which can be advantageous when the covariate effects are similar across quantile levels. BSquare takes a model-based approach to quantile regression, which we briefly review in Section 1; for thorough descriptions of the model and its properties we refer to Reich et al. (2011); Reich (2012); Reich and Smith (2013); Smith et al. (2013). We then illustrate the use of the package for continuous (Section 2), survival (Section 3), and discrete (Section 4) data. We conclude with an example using splines (Section 5).

1 Simultaneous quantile regression

We begin describing the model assuming a continuous response, $Y \in \mathcal{R}$. Denote the covariates as $\mathbf{X} = (X_1, ..., X_p)$, where $X_1 = 1$ for the intercept and it assumed that all covariates are scaled so that $X_j \in [-1, 1]$. Our objective is to model the quantile function of Y as a function of the covariate $\mathbf{X}$. The quantile function, denoted $q(\tau|\mathbf{X})$, is defined as the function satisfying Prob $[Y < q(\tau|\mathbf{X})] = \tau \in [0, 1]$. For example, with $\tau = 0.5$, $q(0.5|\mathbf{X})$ is the median. Linear quantile regression assumes that the τ^{th} quantile is a linear combination of the covariates, $q(\tau|\mathbf{X}) = \sum_{j=1}^{p} X_j \beta_j(\tau)$.

We view $\beta_j(\tau)$ as a continuous function over quantile level τ, which determines the effect of the j^{th} covariate at quantile level τ. For example, if $\beta_j(\tau) = 0$ for $\tau < 0.5$ and $\beta_j(\tau) > 0$ for $\tau > 0.5$, then X_j has no effect on the lower tail, and a positive effect on the upper tail. Each quantile function is modeled as a linear combination of L basis functions,

$$\beta_j(\tau) = \alpha_{0j} + \sum_{l=1}^{L} B_l(\tau)\alpha_{lj},$$

where $B_l(\tau)$ are fixed basis function and α_{lj} are unknown regression coefficients that determine the shape of the quantile function. The basis functions are taken to be functions of a parametric quantile function, q_0, which has the effect of centering the model on a parametric quantile function. Reich and Smith (2013) propose a prior for $\boldsymbol{\alpha} = \{\alpha_{lj}\}$ which ensures that the quantile function corresponds to a valid density function, denoted $f(y|\mathbf{X}, \boldsymbol{\alpha})$. Given the density, the likelihood is simply the product of densities across observations, and standard MCMC algorithms can be used to generate posterior samples for $\beta_j(\tau)$.

This approach extends to survival and discrete data. A response censored on the interval $[a, b]$ has likelihood $F(b|\mathbf{X}, \boldsymbol{\alpha}) - F(a|\mathbf{X}, \boldsymbol{\alpha})$, where F is the distribution function corresponding to f.

This can be used to specify the likelihood for various types of censored survival data:

- **Left-censored survival data**: $a = -\infty$ and b is time of first observation
- **Right-censored survival data**: a equals the follow-up time and $b = \infty$
- **Interval-censored survival data**: a and b are the endpoints of the censoring interval

Also, for a discrete response, say $Y \in \mathcal{Z}$, we can use the quantile regression model for an underlying continuous response Y^*, which is assumed to be rounded off to Y. Then

- **Integer responses**: $a = Y - 0.5$ and $b = Y + 0.5$

and the quantile functions $\beta_j(\tau)$ are interpreted as corresponding to the underlying continuous response.

2 Continuous example

To illustrate the package, we use the New York air quality data in the R dataset airquality. The response is ozone concentration, and we take solar radiation as the single predictor. We remove missing observations, and standardize solar radiation to the interval $(-0.9, 0.9)$.

```
>  data(airquality)
>  ozone=airquality[,1]
>  solar=airquality[,2]

> #Remove missing observations
>  missing=is.na(ozone) | is.na(solar)
>  ozone=ozone[!missing]
>  solar=solar[!missing]

> #Create design matrix.  First column must be all ones, others must be between -1 and 1
>  solar_std = 1.8 * (solar - min(solar))/(max(solar)-min(solar)) - 0.9
>  X = cbind(1,solar_std)
```

The data are plotted in Figure 1

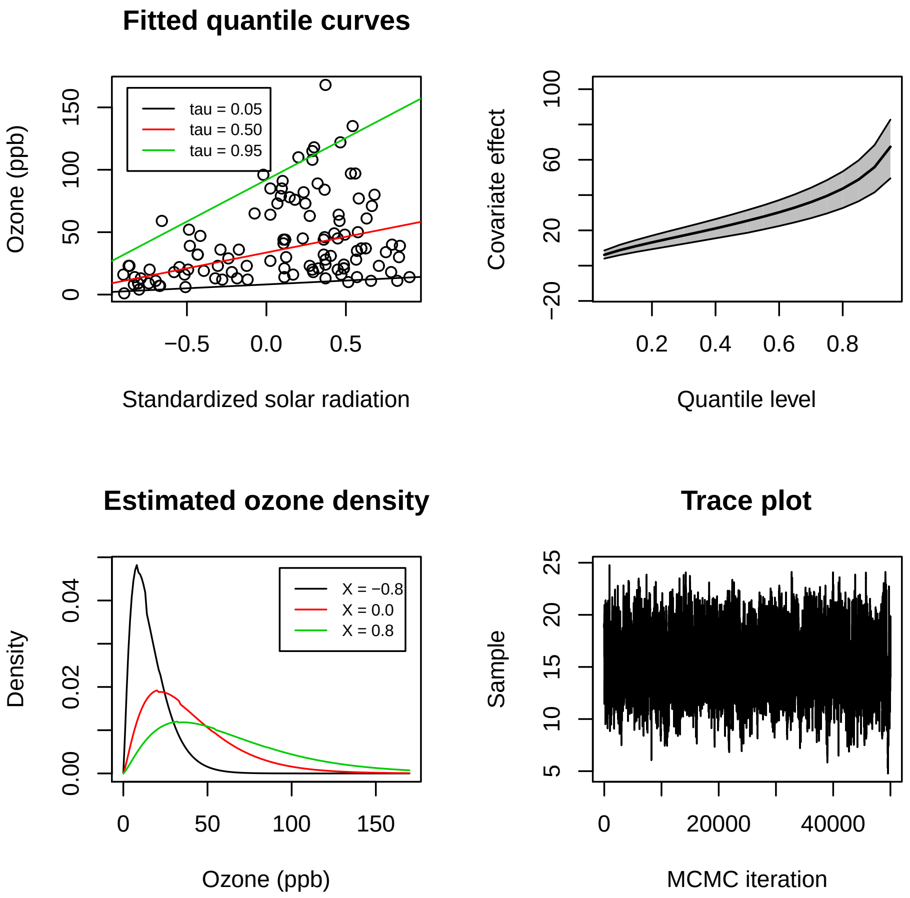
Since ozone concentration cannot be negative, we select the gamma as the centering distribution. The code below fits the model with $L = 4$ basis functions. The samples of $\beta_j(\tau)$ are stored for user-specified quantile levels tau.

```
> library(BSquare)
> tau=seq(0.05,0.95,0.05)
> fit<-qreg(X,ozone,L=4,base="gamma",tau=tau)
> beta<-fit$q
> betahat<-apply(beta,2:3,mean)
```

The array beta has the posterior samples of $\beta_j(\tau)$; beta[i,k,j] is the sample from MCMC iteration i for quantile level tau[k]. betahat is the posterior mean, which is used as the point estimate.

The code below generates Figure 1, which summarizes the relationship between solar radiation and ozone, and checks for convergence of the MCMC sampler. The top left panel plots the data along with the posterior mean of the regression lines for $\tau = 0.05, 0.50$, and 0.95.

Figure 1: Results for the ozone data.



```
> plot(solar_std,ozone,
>      xlab="Standardized solar radiation",
>      ylab="Ozone (ppb)",
>      main="Fitted quantile curves")
> lines(c(-1,1),betahat[1,1] +c(-1,1)*betahat[1,2], col=1)
> lines(c(-1,1),betahat[10,1]+c(-1,1)*betahat[10,2],col=2)
> lines(c(-1,1),betahat[19,1]+c(-1,1)*betahat[19,2],col=3)
> legend("topleft",c("tau = 0.05","tau = 0.50","tau = 0.95"),
>        lty=1,col=1:3,inset=0.05)
```

This shows an increasing trend in solar radiation for all quantile levels, but a much steeper slope for $\tau = 0.95$. Therefore, it appears that solar radiation is a stronger predictor of extreme ozone events than the median and other quantile levels.

The top right panel shows the posterior distribution of $\beta_1(\tau)$ for each quantile level, `tau`.

```
> qr_plot(fit,2)
```

To connect these two plots, we note that the posterior for quantile level 0.05 (0.5, 0.95) in this panel is the posterior of the slope of the black (red, green) line in the top left panel. Again, we see larger slopes for higher quantile levels. The posterior distribution excludes zero for all quantile levels, indicating that solar radiation has a positive effect on ozone at all quantile levels.

The bottom left panel shows the estimated density of ozone given three values of the standardized solar radiation.

```
> y=seq(0,170,1)
> d1=dqreg(fit,y,X=c(1,-.8))
> d2=dqreg(fit,y,X=c(1,0))
> d3=dqreg(fit,y,X=c(1,.8))
> plot(y,d1,type="l",
>      ylab="Density",xlab="Ozone (ppb)",main="Estimated ozone density")
> lines(y,d2,col=2)
> lines(y,d3,col=3)
> legend("topright",c("X = -0.8","X = 0.0","X = 0.8"),
>        lty=1,col=1:3,inset=0.05)
```

For all three solar radiations, the density is skewed and has lower bound zero. As solar radiation increases, the mean and variance of the ozone distribution increase. Note that while these densities appear fairly smooth, the density for this model is actually discontinuous (see Reich and Smith, 2013).

The bottom right panel plots the MCMC samples for $\beta_2(\tau)$ at quantile level `tau`[5].

```
> plot(fit$q[,5,2],type="l",
>      xlab="MCMC iteration",ylab="Sample",main="Trace plot")
```

Ideally, this trace plot will show no temporal trend and will resemble a random sample from the posterior distribution. The user should inspect the trace plot and autocorrelation of the chain for several parameters to ensure convergence (for more discussion of MCMC convergence, see, e.g, Carlin and Louis, 2009). If convergence is not satisfactory, then the number of iterations can be increased, and/or the number of covariates or basis functions should be reduced.

For model comparison, `BSquare` uses the log pseudo-marginal likelihood (LPML) statistic (Carlin and Louis, 2009). Models with larger LPML are preferred.

```
> qreg(X,ozone,L=1,base="gamma")$LPML
[1] -505.1139
> qreg(X,ozone,L=4,base="gamma")$LPML
[1] -505.5292
> qreg(X,ozone,L=1,base="Gaussian")$LPML
[1] -527.9075
> qreg(X,ozone,L=4,base="Gaussian")$LPML
[1] -510.3438
```

For the ozone data, the gamma base distribution is clearly preferred to the Gaussian base distribution.

3 Survival data

`BSquare` can also accommodate survival data. The type of censoring is encoded with the vector `status`, and for interval censored data the vectors of interval endpoints `Y_low[i]` and `Y_high[i]` are required inputs. For observation i,

$$\texttt{status[i]} = \begin{cases} 0 & \text{observation } i \text{ is not censored and is equal to } \texttt{Y[i]} \\ 1 & \text{observation } i \text{ is left-censored on the interval } (\text{-}\infty,\texttt{Y[i]}) \\ 2 & \text{observation } i \text{ is right-censored on the interval } (\texttt{Y[i]},\infty) \\ 3 & \text{observation } i \text{ is censored on the interval } (\texttt{Y\_low[i], Y\_high[i]}) \end{cases}$$

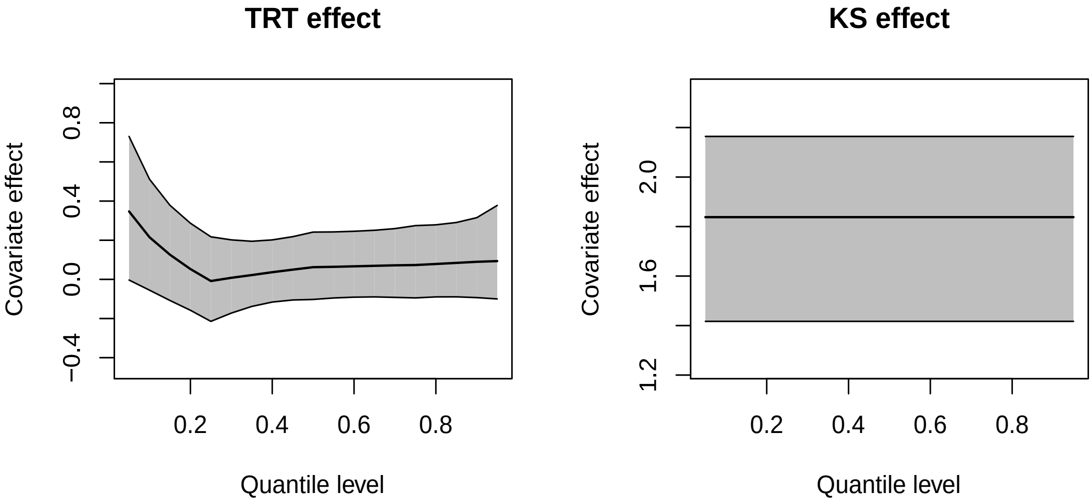
For survival data analysis, we illustrate the package using the `veteran` dataset in the `survival` package. The primary objective is to access the effect for treatment on survival while accounting for several other variables including cell type, age, and Karnofsky performance score. The data are loaded using the code

```
> library(survival)
> data(veteran)

> trt<-ifelse(veteran[,1]==2,-1,1)
> celltype2<-ifelse(veteran[,2]=="smallcell",1,-1)
> celltype3<-ifelse(veteran[,2]=="adeno",1,-1)
> celltype4<-ifelse(veteran[,2]=="large",1,-1)
> time<-veteran[,3]
> logtime<-log(time)
> event<-veteran[,4]
> karno<-veteran[,5]
> karno <- 1.8 * (karno - min(karno))/(max(karno)-min(karno)) - 0.9
> age<-veteran[,7]
> age <- 1.8 * (age - min(age))/(max(age)-min(age)) - 0.9

> coxph(Surv(time,event)~trt+karno+celltype2+celltype3+celltype4+age)
>              coef exp(coef) se(coef)      z       p
> trt       -0.152     0.859    0.103 -1.474 1.4e-01
> karno     -1.616     0.199    0.267 -6.043 1.5e-09
> celltype2  0.428     1.534    0.136  3.156 1.6e-03
```

Figure 2: Survival analysis results.



```
> celltype3  0.589      1.803    0.148  3.977 7.0e-05
> celltype4  0.201      1.223    0.141  1.424 1.5e-01
> age       -0.232      0.793    0.241 -0.965 3.3e-01
```

We fit the quantile model to log survival with $L = 4$ Gaussian basis functions. With 6 predictors, the number of parameters is fairly large if each predictor has L basis functions. Since the objective is to study treatment effects while accounting for the other factors, we hold their effects constant across quantile level, i.e., $\beta_j(\tau) = \beta_j$ for all τ. This has the effect of assuming these predictors are simply location-shift parameters, as in the common AFT model. This is specified using `varying_effect=2` option, which states that only the first two covariates (the intercept and treatment in this case) should have non-constant quantile functions.

```
> status <- ifelse(event==1,0,2)
> X      <- cbind(1,trt,karno,celltype2,celltype3,celltype4,age)
> fit    <- qreg(X,logtime,status=status,L=4,varying_effect=2)

> par(mfrow=c(1,2))
> qr_plot(fit,index=2,main="TRT effect")
> qr_plot(fit,index=3,main="KS effect")
```

The results are plotted in Figure 2. Treatment appears to have a positive effect on lower quantiles, but no effect for other quantiles.

4 Discrete data

To demonstrate the analysis of discrete data, we use the `R` data set `discoveries` which has the number of "great" discoveries in each year from 1860 to 1959 (Figure 3). We fit a quadratic function of time for each quantile level. The data are loaded as processed using

```
> #Load the data
>   data(discoveries)
>   Y=as.vector(discoveries)
>   year=1860:1959
>
> #Prep the data
>   LOWER=Y-0.5
>   UPPER=Y+0.5
>   x1=seq(-1,1,length=100)
>   x2=x1^2
>   X=cbind(1,x1,x2)
```

We fit the model with $L = 4$ basis functions and a gamma centering distribution for the underlying continuous process. The MCMC algorithm is called via

```
>  status<-rep(3,100)
>  fit<-qreg(X=X,Y_low=LOWER,Y_high=UPPER,L=4,status=status,base="gamma")
```

The fitted quantiles in Figure 3 are plotted using the code

```
> #Extract samples and compute posterior of the 50th and 95th quantile by year
> Q50<-fit$q[,10,]%*%t(X)
> Q95<-fit$q[,19,]%*%t(X)

> #Plot the data and fitted quantiles
> plot(year,Y,
>      xlab="Year",
>      ylab="Great discoveries")
>
> lines(year,apply(Q50,2,quantile,0.05),col=2,lty=2)
> lines(year,apply(Q50,2,quantile,0.50),col=2,lty=1)
> lines(year,apply(Q50,2,quantile,0.95),col=2,lty=2)
>
> lines(year,apply(Q95,2,quantile,0.05),col=3,lty=2)
> lines(year,apply(Q95,2,quantile,0.50),col=3,lty=1)
> lines(year,apply(Q95,2,quantile,0.95),col=3,lty=2)
>
> legend("topright",c("tau = 0.50","tau = 0.95"),lty=1,col=2:3,inset=0.05)
```

Here we see, sadly, a decline in the number of great discoveries beginning around the turn of the century, especially for the 95th percentile.

5 An example using spline basis functions

While parametric basis functions provide substantial flexibility, the researcher may not want to make parametric assumptions about the distribution, even locally. In this section, rather than assuming a parametric form for the basis functions $B_l(\tau)$ we model the quantile function as a linear combination of Integrated M-splines (I-splines). I-splines are monotonic in the quantile level, so any positive linear combination of I-splines results in a valid quantile process.

Figure 3: Results for the discoveries data. The solid lines are posterior medians, and the dashed lines give posterior 90% intervals

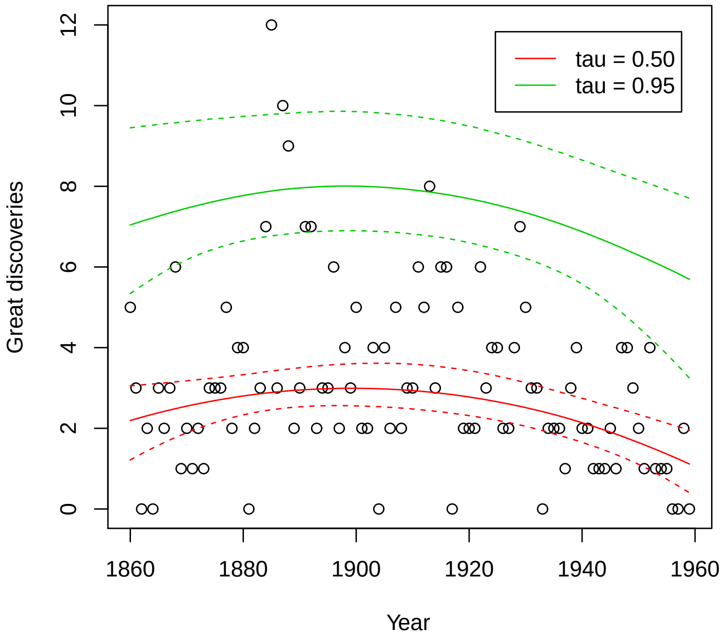


Figure 4: Spline basis function with internal knots 0.1, 0.5, and 0.9.

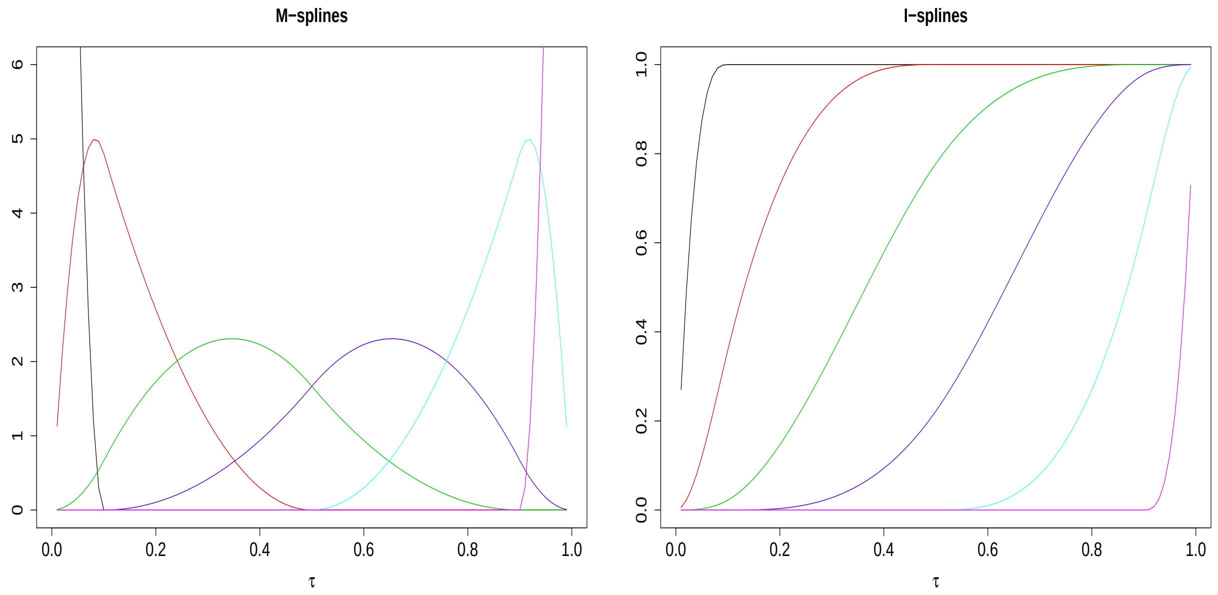


Figure 4 shows plots of the I-splines corresponding to knots at 0.1, 0.5, and 0.9. I-splines are similar to the parametric basis functions in that they are increasing along only a subset of the quantile domain. However, I-splines are continuous and differentiable, providing a smoother transition across the knots and resulting in a continuous density everywhere.

Common nonparametric maxims apply to knot selection in the spline fit. Increasing the number of knots adds flexibility and decreases the smoothness of the fit, then increasing the number of basis functions tends to increase variance and decrease bias. If the knots are roughly evenly spaced, then the number of knots is more important than the knot locations.

For an individual predictor we shrink the non-constant basis functions to a common mean to jointly inform the distribution. This mean is assumed to be 0 with variance `mu_var`. The hyperparameters `sig_a` and `sig_b` are the shape and scale for the precision parameter that correlates the basis functions corresponding to an individual effect. The parameters associated with the constant basis functions are given independent mean 0, variance `cbf_var` Gaussian priors. To enhance the flexibility of I-splines in the tails we assume a parametric form for the distribution for extreme quantiles. Observations below the quantile function evaluated at `q_high` are modeled using I-splines, while observations above the threshold are modeled with either a Pareto or exponential distribution.

We similarly fit a parametric tail for observations below `q_low`. This permits deep tail inference while ensuring a few outliers deep in the tails do not distort the ends of the quantile function. Distributions whose tails decay slowly (e.g. t-distribution) should be modeled using a Pareto tail, while distributions with lighter tails (e.g. Gaussian) should use exponential tails. In both distributions the scale parameter is determined by the density of the quantile function at the threshold. The shape parameter for the Pareto distribution is given a log-normal prior. The default hyperparameters are chosen such that roughly 90% of the mass is below 0.5 and another 9% is below 1. If there are few observations beyond the thresholds, tail type may not be identifiable from the data. For small sample sizes inference on the inner quantiles is robust to tail selection.

The I-spline model can accommodate censored data. If all of the data are censored beyond a certain threshold, it is difficult to conduct inference beyond the threshold. In this scenario we recommend parsimonious models and not placing knots beyond where the data are censored.

Likelihood calculations for the parametric basis functions are faster than their spline counterparts due to the lack of a closed form relationship between the density and the quantile function for the splines. Diagnosing MCMC convergence can be difficult. The basis coefficients are truncated to ensure the quantile function is monotonic, therefore, trace plots of the posterior effects can exhibit drastic jumps in a single iteration. When examining trace plots it is important to distinguish between a large jump at one iteration in the middle of a stationary chain and a chain that is still exploring the posterior parameter space. These jumps can indicate overfitting of the quantile function. Finally, the tails are harder to estimate than the middle of the distribution. In some applications there is not enough data to identify tail effects, so trace plots of effects near or in the tails may never appear to converge.

We illustrate the spline method using the air quality data from Section 2.

```
> library(BSquare)
> data(airquality)
> ozone<-airquality[,1]
> solar<-airquality[,2]
>
> #Remove missing observations
> missing<-is.na(ozone+solar)
> ozone<-ozone[!missing]
> solar<-solar[!missing]
> solar_std<-1.8*(solar - min(solar))/(max(solar)-min(solar)) - 0.9
>
> X<-cbind(1,solar_std)
> Y<-ozone
```

We compare 4 fits. The first 2 fits use Pareto tails with different knots.

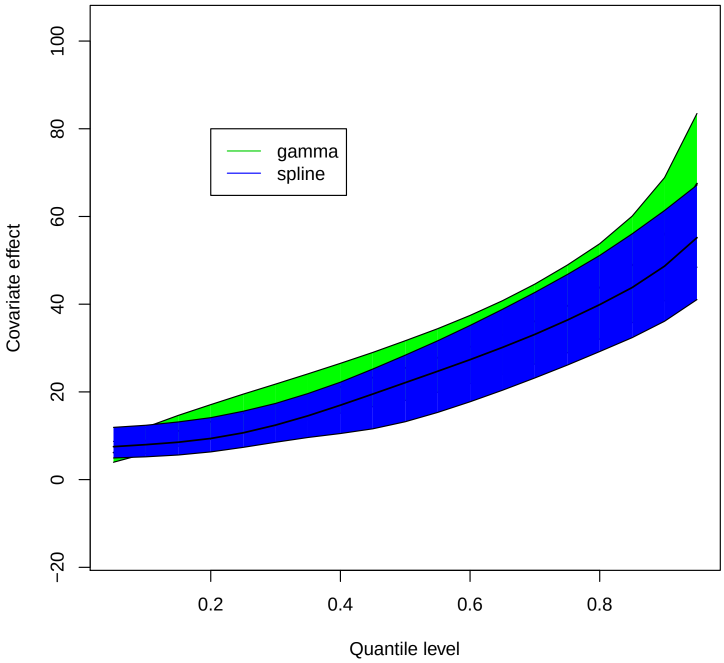
```
> out1<-qreg_spline(X,Y=Y,knots_inter=c(.5),burn = 10000, iters = 50000)
> out2<-qreg_spline(X,Y=Y,knots_inter=c(.2,.5,.8),burn = 10000, iters = 50000)
```

The first places a knot only at the median, while the second places knots at the median and the 20^{th} and 80^{th} percentiles. The next 2 fits use exponential tails.

```
> out3<-qreg_spline(X,Y=Y,knots_inter=c(.5),Pareto=F,burn = 10000, iters = 50000)
> out4<-qreg_spline(X,Y=Y,knots_inter=c(.2,.5,.8),Pareto=F,burn = 10000, iters = 50000)
```

```
> round(out1$LPML)
[1] -508
> round(out2$LPML)
[1] -507
> round(out3$LPML)
[1] -517
```

Figure 5: Solar Spline Effect



```
> round(out4$LPML)
[1] -511
```

The model with Pareto tail and 3 knots is preferred over the others. The gamma model of Section 2 is selected as the best overall. Qualitatively solar radiation has the same effect using the parametric and non-parametric basis functions. The gamma fit has a slightly stronger effect across quantile level.

```
> out_best<- qreg(X,ozone,L=4,base="gamma",tau=tau)
> qr_plot(out_best,2,col="green")
> qr_plot(out2,2,add=T,col="blue")
> legend(x=.2,y=80,legend=c("gamma","spline"),lty=c(1,1),col=c(3,4))
```

The strength of the solar radiation effect increases with the quantile level, as does the uncertainty, as seen in Figure 5.

Two extensions may be of interest to the reader. First, if modeling multiple quantile functions that are correlated across time or space is desirous, code is available upon request. Second, likelihood calculations increase linearly in sample size, which can be prohibitive for large datasets (e.g. $n \geq 5000$.) Fortunately the likelihood calculation is embarrassingly parallel, and code for graphics processing units is available from the authors on request.

6 Acknowledgements

This work was partially supported by the NIH (5R01ES014843-02) and US EPA (R835228).

References

Carlin, B. P. and Louis, T. A. (2009) *Bayesian methods for data analysis.* CRC Press.

Reich, B. J. (2012) Spatiotemporal quantile regression for detecting distributional changes in environmental processes. *Journal of the Royal Statistical Society: Series C*, **64**, 535–553.

Reich, B. J., Fuentes, M. and Dunson, D. B. (2011) Bayesian spatial quantile regression. *Journal of the American Statistical Association*, **106**, 6–20.

Reich, B. J. and Smith, L. B. (2013) Bayesian quantile regression for censored data. *Biometrics.* In press.

Smith, L. B., Fuentes, M., Herring, A. H. and Reich, B. J. (2013) Bayesian quantile regression for discrete data. Submitted.