A brief vignette on joint quantile regression for spatial data

Xu Chen

1 $PM_{2.5}$ concentration dataset

The dataset[1] (Paciorek 2013) contains averaged daily $PM_{2.5}$ concentrations from 2001 to 2002 monitored at 339 stations in the northeastern United States. We load the data and split it into a training set (80%) and a test set (20%).

```
# load data
pm <- read.csv('simulation/real/pm.csv')

# assign numbers of observations in training set and test set
n <- nrow(pm)
n.train <- 270
n.test <- n - n.train

# random split the dataset into a training set and a test set
set.seed(123)
train.ix <- sample(n, n.train)
test.ix <- (1:n)[-train.ix]
```

Many geographical and meteorological covariates are recorded. For illustration, the following 5 covariates are included in our analysis: (i) logarithm of population density at county level (LCYPOP00), (ii) logarithm of distance to A1 class roads (primary roads, typically interstates) (LDISTA1), (iii) proportion of urban land use within 1 kilometer of the station location (NCLD_URB), (iv) logarithm of PM 2.5 emissions with in a 10 kilometer buffer (LS25_E10), and, (v) elevation of the station (ELEV_NED).

```
# extract response and predictors
y.train <- pm[train.ix,]$pm
x.train <- pm[train.ix, c('LCYPOP00', 'LDISTA1', 'NLCD_URB', 'LS25_E10', 'ELEV_NED')]

y.test <- pm[test.ix,]$pm
x.test <- pm[test.ix, c('LCYPOP00', 'LDISTA1', 'NLCD_URB', 'LS25_E10', 'ELEV_NED')]
```

We calculate the Euclidean distances between monitoring stations.

```
# get coordinates of monitoring stations
coords <- pm[,c('x','y')]

# calculate the Euclidean distances between monitoring stations in the training set
distance <- as.vector(dist(coords[train.ix,], method = 'euclidean'))
```

2 Joint quantile regression analysis of $PM_{2.5}$ concentration

Joint spatial quantile regression with Gaussian copula process

We first load the necessary functions for conducting the analysis.

[1]The dataset is available at https://www.stat.berkeley.edu/users/paciorek/code/ejs/.

```r
# load MCMC sampler written in C
dyn.load('jsqr/jsqrgp/jsqrgp.so')

# load utility functions
source('jsqr/jsqrgp/utility.R')

# load the main function
source('jsqr/jsqrgp/jsqrgp.R')
```

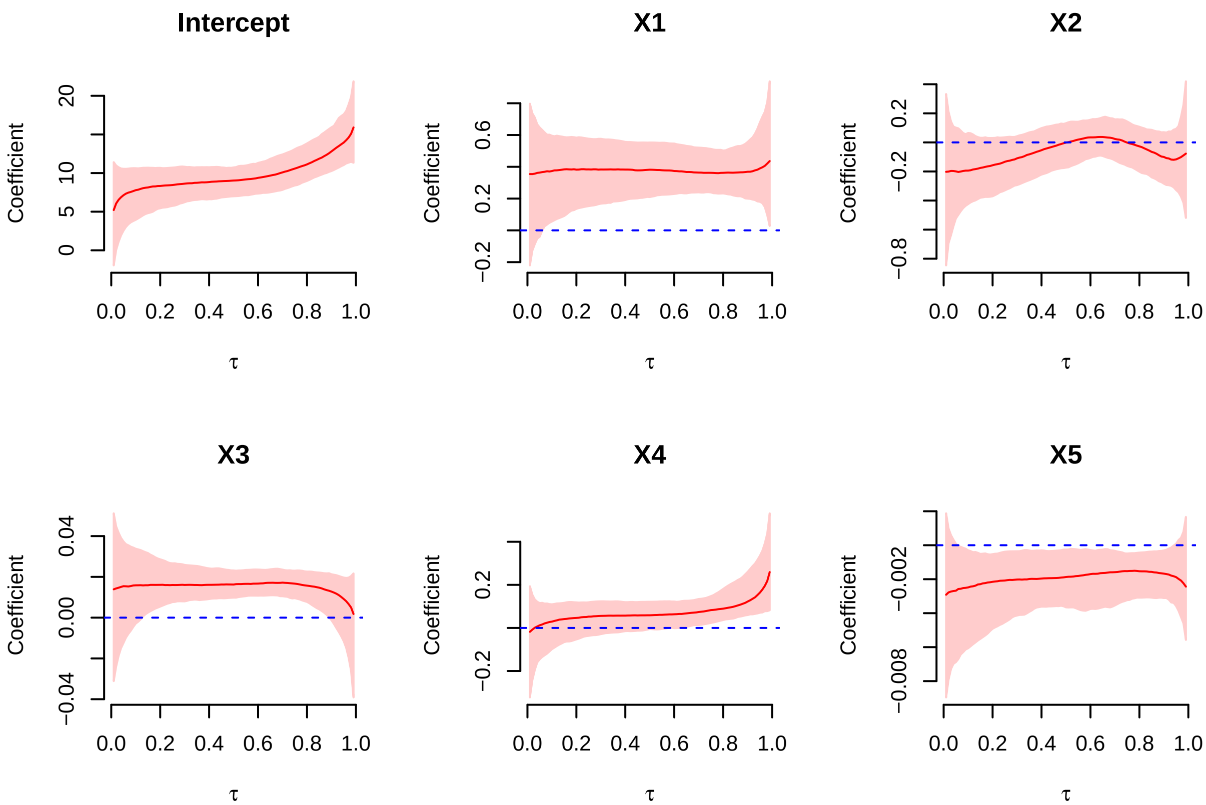
We fit the joint spatial quantile regression (JSQR) model with Mat'ern correlation function and logistic bsae distribution. The number of discrete values of decay parameter (ϕ) is set to be 10. The parameter values are initialized according to priors. The MCMC algorithm is run for 20,000 iterations with first 10,000 as burnin. The target acceptance ratio is 15%. We monitor the log posterior value every 2,000 iterations.

```r
# fit joint spatial quantile regression model
jsqrgp.fit <- qrjointgp(x = x.train, y = y.train, par = 'prior', distance = distance,
                        location = coords[train.ix,], nphi = 10, kernel = 'matern',
                        acpt.target = 0.15, acpt.hptarget = 0.15, kappa = 2, nsamp = 2e3,
                        thin = 10, fbase = 'logistic')
```

```
## Initial lp = -601.501
## iter = 2000, lp = -440.348
## iter = 4000, lp = -418.993
## iter = 6000, lp = -439.563
## iter = 8000, lp = -432.194
## iter = 10000, lp = -434.894
## iter = 12000, lp = -430.472
## iter = 14000, lp = -423.425
## iter = 16000, lp = -439.367
## iter = 18000, lp = -426.448
## iter = 20000, lp = -431.837
## time = 194.929 elapsed time: 196 seconds
```

After the model fitting, we plot the estimated regression coefficients across quantile levels from 0 to 1. The point estimates (red curve) are given by posterior medians. 95% credible intervals are also presented.

```r
# plot the regression coefficients across quantile levels from 0 to 1
coef.qrjointgp(jsqrgp.fit, burn.perc = 0.5, nmc = 5e2, plot = TRUE)
```



The Watanabe-Akaike information criteria (WAIC) (Watanabe 2010, Gelman, Hwang, and Vehtari (2014)) can be calculated as follows.

```
# calculate two versions of WAIC (the latter one should be more reliable)
waicgp(jsqrgp.fit, burn.perc = 0.5, nmc = 500)
```

```
## WAIC.1 = 793.8 , WAIC.2 = 808.15
```

```
##   WAIC1   WAIC2
## 793.803 808.150
```

We then showcase how to perform spatial interpolation with our model.

```
# the quantile levels at which we examine the prediction performances of the model
tau.grid <- c(0.01,0.05,seq(0.1,0.9,0.1),0.95,0.99)

# calculate predicted conditional quantiles at specified quantile levels
qt.pred <- predict.qrjointgp(jsqrgp.fit, x.test = x.test, location.test = coords[test.ix,],
                             tau.test = tau.grid, burn.perc = 0.5, nmc = 5e2)
```

The check loss is computed to evaluate the prediction performance of the model.

```
# specify check loss function
check.loss <- function(r, tau){return(r*(tau - (r<0)))}

# calculate residuals
jsqrgp.test.err <- apply(qt.pred, 2, function(u) y.test - u)

# calculate check loss
jsqrgp.test.loss <- matrix(NA, n.test, length(tau.grid))
for(i in 1:length(tau.grid)){jsqrgp.test.loss[,i] <- check.loss(jsqrgp.test.err[,i], tau.grid[i])}
```

Joint spatial quantile regression with t copula process

We could alternatively adopt t copula process to carry out the analysis as follows. The code is similar to that of Gaussian copula process.

```
dyn.load('jsqr/jsqrtp/jsqrtp.so')
source('jsqr/jsqrtp/utility.R')
source('jsqr/jsqrtp/jsqrtp.R')
```

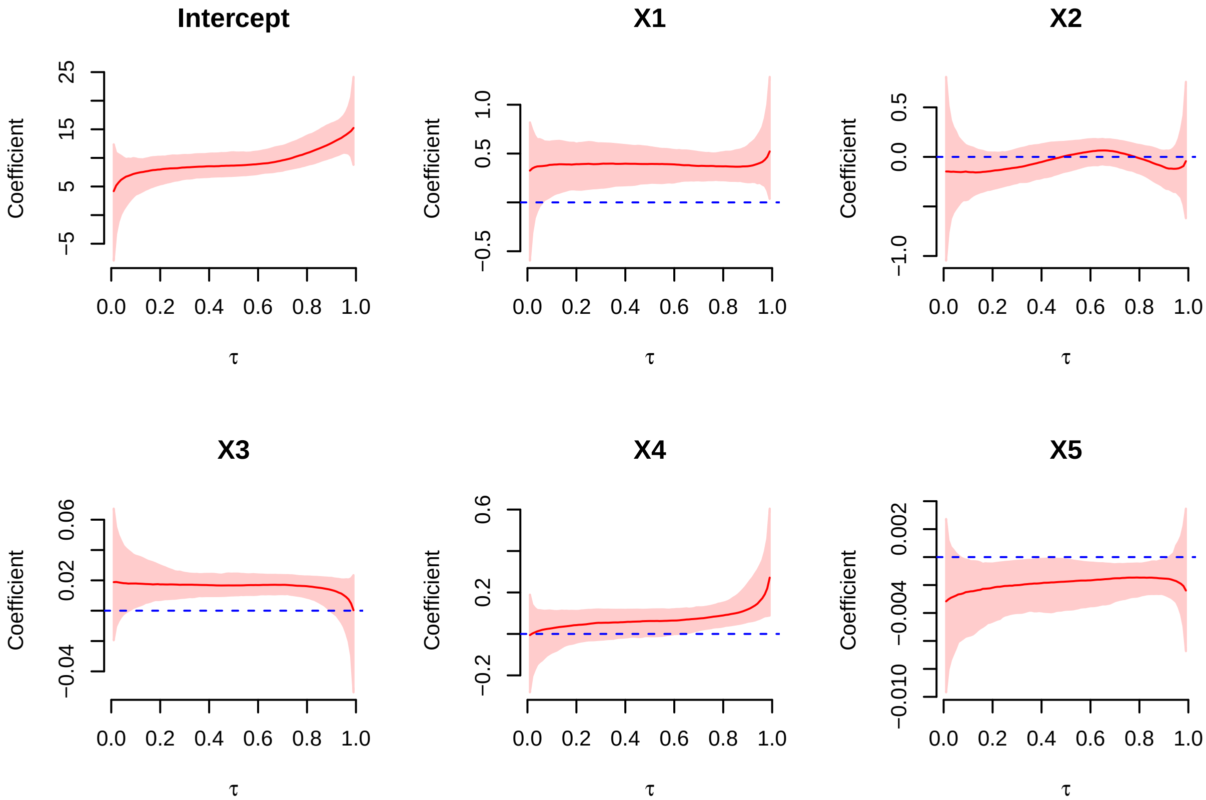
Instead of using logistic base distribution, we employ a t base to account for potential heavy tails of response.

```
jsqrtp.fit <- qrjointtp(x = x.train, y = y.train, par = 'prior', distance = distance,
                         location = coords[train.ix,], nphi = 10, kernel = 'matern',
                         acpt.target = 0.15, acpt.hptarget = 0.15, kappa = 2, nsamp = 2e3,
                         thin = 10, fbase = 't')
```

```
## Initial lp = -624.545
## iter = 2000, lp = -437.108
## iter = 4000, lp = -422.18
## iter = 6000, lp = -439.383
## iter = 8000, lp = -433.906
## iter = 10000, lp = -418.954
## iter = 12000, lp = -438.707
## iter = 14000, lp = -433.025
## iter = 16000, lp = -423.972
## iter = 18000, lp = -442.299
## iter = 20000, lp = -425.223
## time = 480.627 elapsed time: 481 seconds
```

The plot of regression coefficients and WAIC are similar to those of the model with Gaussian copula process.

```
coef.qrjointtp(jsqrtp.fit, burn.perc = 0.5, nmc = 5e2, plot = TRUE)
```




```
waictp(jsqrtp.fit, burn.perc = 0.5, nmc = 500)
```

```
## WAIC.1 = 793.58 , WAIC.2 = 807.61

##    WAIC1    WAIC2
## 793.5762 807.6082
```

We also calculate the predicted conditional quantiles given by JSQR with t copula process

```
qt.pred <- predict.qrjointtp(jsqrtp.fit, x.test = x.test, location.test = coords[test.ix,],
                             tau.test = tau.grid, burn.perc = 0.5, nmc = 5e2)

jsqrtp.test.err <- apply(qt.pred, 2, function(u) y.test - u)
jsqrtp.test.loss <- matrix(NA, n.test, length(tau.grid))

for(i in 1:length(tau.grid)){jsqrtp.test.loss[,i] <- check.loss(jsqrtp.test.err[,i], tau.grid[i])}
```

Comparison of prediction performances

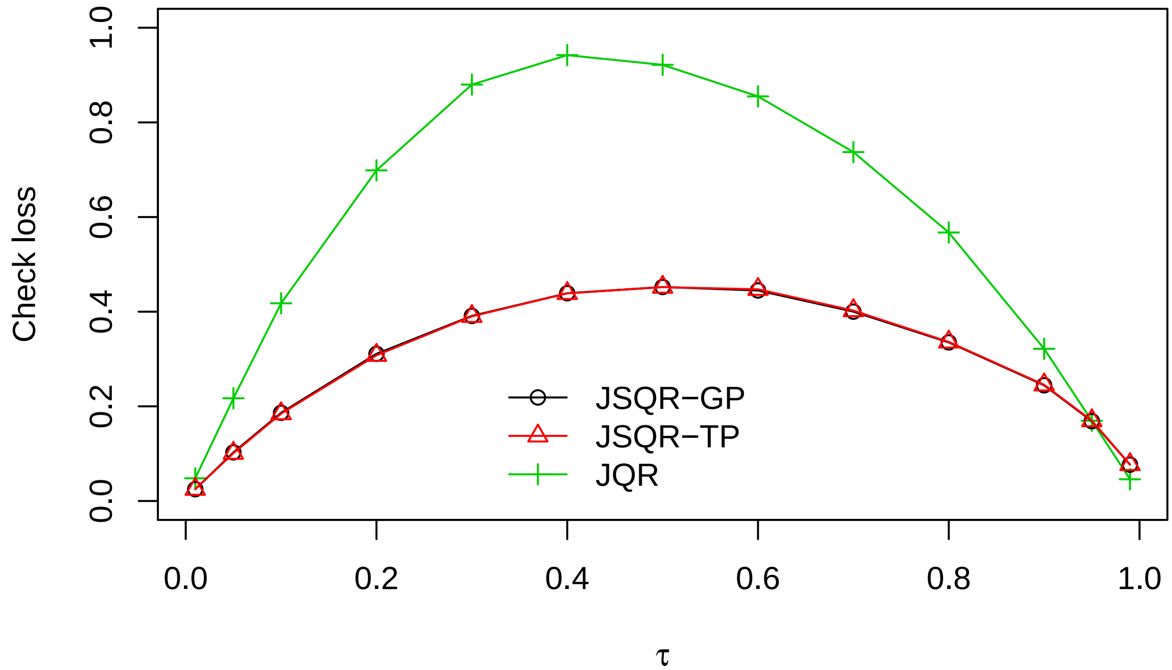
We illustrate the prediction performances of JSQR model with Gaussian and t copula processes fitted above, benchmarked by the joint quantile regression (JQR) model (Yang and Tokdar 2017) without adjusting for spatial dependence. The implementation of JQR is hided.

The check losses are presented in the figure below. JSQR-GP and JSQR-TP provide similar prediction performances which are clearly superior than that given by JQR.

```
# plot check losses of predicted conditional quantiles given by three models
plot(tau.grid, apply(jqr.test.loss,2,mean), type = 'o', pch = 3, col = 3,
     xlab = expression(tau), ylab = 'Check loss', ylim = c(0,1))
points(tau.grid, apply(jsqrgp.test.loss,2,mean), type = 'o', pch = 1)
```

```r
points(tau.grid, apply(jsqrtp.test.loss,2,mean), type = 'o', pch = 2, col = 2)

# add legend
legend(x = 0.33, y = 0.3, legend = c('JSQR-GP', 'JSQR-TP', "JQR"), lty = 1,
       pch=c(1,2,3), col=c(1,2,3), bty = 'n')
```



References

Gelman, Andrew, Jessica Hwang, and Aki Vehtari. 2014. "Understanding Predictive Information Criteria for Bayesian Models." *Statistics and Computing* 24 (6). Springer: 997–1016.

Paciorek, Christopher J. 2013. "Spatial Models for Point and Areal Data Using Markov Random Fields on a Fine Grid." *Electronic Journal of Statistics* 7: 946–72.

Watanabe, Sumio. 2010. "Asymptotic Equivalence of Bayes Cross Validation and Widely Applicable Information Criterion in Singular Learning Theory." *Journal of Machine Learning Research* 11 (Dec): 3571–94.

Yang, Yun, and Surya T Tokdar. 2017. "Joint Estimation of Quantile Planes over Arbitrary Predictor Spaces." *Journal of the American Statistical Association* 112 (519). Taylor & Francis: 1107–20.