A Vignette on Model-Based Quantile Regression: Analyzing Excess-Zero Response

Erika Cunningham, Surya T Tokdar and James S Clark
Duke University

2019-01-26

Abstract

Quantile regression is widely seen as an ideal tool to understand complex predictor-response relations. Its biggest promise rests in its ability to quantify whether and how predictor effects vary across response quantile levels. But this promise has not been fully met due to a lack of statistical estimation methods that perform a rigorous, joint analysis of all quantile levels. This gap has been recently bridged by Yang and Tokdar (2017). Here we demonstrate how their joint quantile regression method, as encoded in the R package qrjoint, offers a comprehensive and model-based regression analysis framework. This chapter is an R vignette where we illustrate how to fit models, interpret coefficients, improve and compare models, and obtain predictions under this framework. Our case study is an application to ecology where we analyze how the abundance of red maple trees depends on geological and geographical features of the location. A complete absence of the species contributes excess zeros in the response data. We treat such excess zeros as left censoring in the spirit of a Tobit regression analysis. By utilizing the generative nature of the joint quantile regression model, we not only adjust for censoring but also treat it as an object of independent scientific interest.

Keywords: joint quantile regression, nonparametric regression, Tobit Regression, censoring, excess zero, semi-continuous, qrjoint, R Vignette.

Contents

1 Introduction

Four decades ago, Roger Koenker and Gib Bassett showed how to formalize statistical inference using quantile regression (Koenker and Bassett, 1978). Today quantile regression is widely recognized as a fundamental statistical tool for analyzing complex predictor-response relationships, with a growing list of applications in ecology, economics, education, public health, climatology, and so on (Burgette et al., 2011; Elsner et al., 2008; Dunham et al., 2002; Abrevaya, 2002). In quantile regression (QR), one replaces the standard regression equation of the mean $E[Y \mid X] = \beta_0 + X^T\beta$ with an equation for a quantile $Q_Y(\tau \mid X) = \beta_{0\tau} + X^T\beta_\tau$, where $\tau \in (0, 1)$ is a quantile level of interest and $Q(\tau)$ denotes the $100\tau^{th}$ percentile. A choice of $\tau = 0.5$ results in the familiar median regression, a robust alternative to mean regression when one suspects the response distribution to be heavy tailed. But the real strength of QR lies in the possibility of analyzing any quantile level of interest, and perhaps more importantly, contrasting many such analyses against each other with fascinating consequences.

This strength of QR has also been its liability. Most modern scientific applications of QR involve a synthesis of estimates obtained at several quantile levels. Estimates and p-values are pooled together to build a composite picture of how predictors influence the response and to analyze how this influence varies from the center of the response distribution to its tails. But such a synthesis is flawed! The composite picture is not based on a single statistical model of the data. Instead, for each single quantile level in the ensemble, a new model has been fitted, without sharing any information with models fitted at the other τ values. It is entirely possible that the quantile lines estimated at different quantile levels cross each other, thus violating basic laws of probability. Additionally, due to a lack of information borrowing, estimated standard errors and p-values may fluctuate wildly as functions of τ (Tokdar and Kadane, 2012). This, at best, creates confusion and, at worst, may encourage selective reporting!

A composite QR analysis can be formalized with the simultaneous equations

$$Q_Y(\tau \mid X) = \beta_0(\tau) + X^T\beta(\tau), \quad \tau \in (0, 1), \tag{1}$$

where $\beta_0(\tau)$ and $\beta(\tau) = (\beta_1(\tau), \ldots, \beta_p(\tau))^T$, are unknown intercept and slope curves. Because quantiles are linearly ordered in their levels, estimation of β_0 and β must be carried out under the "non-crossing" constraint: $\beta_0(\tau_1) + x^T\beta(\tau_1) < \beta_0(\tau_2) + x^T\beta(\tau_2)$ for every $0 < \tau_1 < \tau_2 < 1$, and, every $x \in \mathcal{X}$, where $\mathcal{X}$ is the domain of the predictor vector X. A largely under-appreciated, simple observation is that the simultaneous QR equations and the non-crossing constraint together offer a fully generative probability model for the response

$$Y = \beta_0(U) + X^T\beta(U), \quad U \mid X \sim Unif(0, 1), \tag{2}$$

opening up the possibility of obtaining proper statistical inference on the intercept and slope curves by means of a joint analysis.

Yang and Tokdar (2017) offer an estimation framework for the joint QR model (2), subject to the non-crossing constraint, by introducing a bijective map of the intercept and slope curves to a new parameter ensemble

consisting of scalars, vectors and curves, all but one of which are constraint free. The likelihood score, as a function of the new parameter ensemble, can be efficiently computed through numerical approximation methods. Parameter estimation can then proceed according to either a penalized likelihood or a Bayesian approach. An instance of the latter, where curve valued parameters are assigned Gaussian process priors, is further investigated by Yang and Tokdar (2017) who establish that the resulting estimation method is consistent and robust to moderate amount of model-misspecification.

To the best of our knowledge, Yang and Tokdar (2017) provide the only estimation framework that supports quantile regression as a model-based inference and prediction technique in its full generality. Their reparameterization technique applies to any predictor dimension and to any arbitrarily shaped predictor domain $\mathcal{X}$ that is convex and bounded. Both issues have proven major vexing points to the earlier attempts at a joint QR analysis, e.g. He (1997); Dunson and Taylor (2005); Bondell et al. (2010); Reich et al. (2011); Tokdar and Kadane (2012); Feng et al. (2015).

In this chapter we demonstrate that the joint quantile regression method of Yang and Tokdar (2017), as implemented in the `R` package[1] `qrjoint`, offers a comprehensive, model-based, regression analysis toolbox. We demonstrate how to fit models, interpret their coefficients, improve and compare models, and obtain predictions under the joint quantile regression setup. Taking this modeling one step further, we show how utilizing the censored-data options built into the `qrjoint` package can yield a interpretable yet distributionally-flexible model for non-negative, continuous data with excess-zeroes. This latter extension fully exploits the generative model interpretation (2) of joint quantile regression.

2 Excess-Zero Regression Analysis

Zero-inflation, or the frequent occurrence of zeroes, is common in ecological data. For instance, when counting the number of species in a region, some regions may not have any of the target species, resulting in "zero" records. Another example, one that will serve as case study here, involves measuring the basal area of trees within a site. When trees are present, basal area is measured as a continuous, positive number, but when trees are not present, a zero is recorded.

Tobit regression (Tobin, 1958) is commonly used to model censored data but can also be used to model data with excess boundary zeroes. To do so, it uses a latent construct, namely $y_i^* = \beta_0 + \beta x_i + \epsilon_i,\ \epsilon_i \sim N(0, \sigma^2)$ with observables $y_i = max\{y_i^*, 0\}$. Under this assumption of normality, the mean $\beta_0 + X\beta$ and variance σ^2 fully specify the response distribution. If the latent Tobit model is framed in terms of a joint quantile regression it would be written as $Q_{Y*}(\tau|X) = \beta_0(\tau) + X\beta$ where $\beta_0(\tau) = \sigma\Phi^{-1}(\tau)$. That is, the normality is captured in the τ-functional intercept by the normal inverse CDF, and all remaining variability in the response quantiles is explained by τ-constant slopes and the design matrix X.

Joint quantile regression is also capable of both capturing the probability of atomic zero-measurements and modeling the remaining positive, continuous response distribution. Like the Tobit model, it captures the zeroes via a censored-data or latent-truth construct; however, unlike Tobit, it is not limited by an assumption of normality. In fact, it makes no assumption about the distributional form of the response distribution and has only two other modeling assumptions: 1) data can be explained as linear combinations of covariates expressed in the design matrix X, which incorporates any desired interactions or non-linearities (e.g. via splines); and 2) observations are independent of each other.

Other quantile regression methods (Powell, 1986; Portnoy, 2003) are capable of distribution-free estimation in the presence of excess-zeroes; however, these other methods estimate regression quantiles independently and, lacking a comprehensive model specification to capture dependence between regression quantiles, they only make adjustments for and do not actually model the probability of atomic-zero.

We demonstrate how to use the `qrjoint` package on tree basal area data from the U.S. Forest Service. Tobit regression models are included, both as a stepping-stone to understanding censored joint quantile regression and as a foil to the more flexible joint quantile regression.

[1] https://CRAN.R-project.org/package=qrjoint

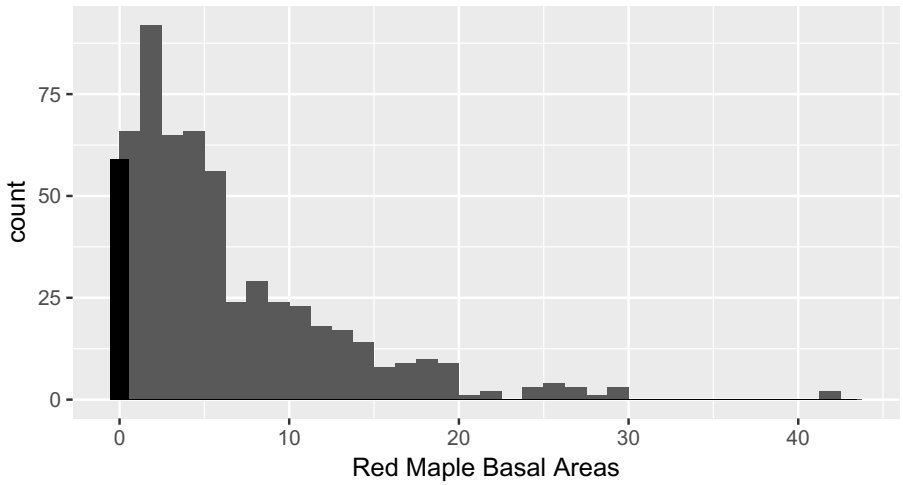

Figure 1: Red maple basal areas for 608 sites in Masachusettes, Connecticut, and Rhode Island. Those with no red maple trees, i.e. baRedMaple of zero, are displayed in black.

3 Case Study Data and Objective

The U.S. Forest Service tracks the biomass of hundreds of species of trees on thousands of plots of land throughout the United States. We consider a subset of data from the Forest Inventory Analysis composed of 608 unmanaged and forested sites in Massachusetts, Connecticut, and Rhode Island[2].

```
library(qrjoint)   # For joint quantile regression fitting
library(ggplot2)   # For plotting results
library(gridExtra) # For arranging side-by-side plots
data(redmaple)
dat <- redmaple
```

While tree counts and cumulative basal area (ft^2/acre) are recorded on hundreds of species, we focus on basal area for a single species, the red maple tree (*Acer rubrum*). Red maple is common among the 608 sites with 59 sites (9.7%) having no red maple trees (i.e. basal area equals zero) and the remaining sites having median basal area of 4.7 ft^2/acre. A histogram of all basal areas from the sample can be seen in Figure 1.

In addition to basal area, several covariates are available for each site:

- `elev.` Elevation of site, measured in feet
- `slope.` Slope of site, measured in degrees
- `aspect.` Aspect of site, measured in degrees proceeding from North clockwise around a compass. For sites with zero or near-zero slopes, aspect is recorded as 0. North is recorded as 360.
- `region.` EPA Level-III geographical region

The first three covariates are continuous measures, and the fourth, `region`, is categorical. We desire to build a model to understand the relationships between these explanatory variables and red maple basal areas. More specifically, we would like to gain direct inference not only on the how the predictors affect the mean or median response but also on how they affect the upper and lower quantiles of the response distribution.

[2]http://apps.fs.fed.us/fiadb-downloads/datamart.html

4 Fitting Single-covariate Basal Area Models

For pedagogical reasons, we start with a model that uses a single covariate, elevation (`elev`), to predict red maple basal area and compare to the more widely-recognized Tobit model. R's `AER` package is used to obtain maximum likelihood estimates for the Tobit model. Note that this `tobit` function sets the left limit of the censored dependent variable to zero by default.

```
library(AER)      # for Tobit regression fit
fit.tb1 <- tobit(baRedMaple ~ elev, data = dat)
summary(fit.tb1)
#>
#> Call:
#> tobit(formula = baRedMaple ~ elev, data = dat)
#>
#> Observations:
#>          Total  Left-censored     Uncensored Right-censored
#>            608             59            549               0
#>
#> Coefficients:
#>              Estimate Std. Error z value Pr(>|z|)
#> (Intercept)  6.321077   0.462592  13.664   <2e-16 ***
#> elev        -0.003316   0.001864  -1.779   0.0752 .
#> Log(scale)   1.911352   0.030748  62.162   <2e-16 ***
#> ---
#> Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
#>
#> Scale: 6.762
#>
#> Gaussian distribution
#> Number of Newton-Raphson Iterations: 2
#> Log-likelihood: -1889 on 3 Df
#> Wald-statistic: 3.165 on 1 Df, p-value: 0.075246
```

4.1 Joint Quantile Regression Call

The `qrjoint` package contains an eponymous function which performs a Bayesian parameter estimation of the generative model (2). Posterior computation is done by with the help of Markov chain Monte Carlo (MCMC) over an unconstrained parameter space that offers a complete reparameterization of the original model. Likelihood score calculation is done by discretizing the quantile levels to a finite, dense grid of τ values. The function-valued parameters of the model, which are assigned independent Gaussian process priors, are approximated by closely related finite rank predictive processes Tokdar (2007); Banerjee et al. (2008). See Yang and Tokdar (2017) for more technical details.

The `qrjoint` function uses a data-formula specification similar to the `lm` function from the `stats` package to build the design matrix X. The function performs all necessary data centering so that inference may proceed anywhere within the convex hull of the data predictor space. The default `incr=0.01` provides estimates over a τ-grid at 0.01 resolution, i.e. 0.01, 0.02, 0.03, ..., 0.98, 0.99, with slightly more dense grids in the tails; the same grid is used in likelihood score computation. This resolution is sufficient for our needs. Also sufficient is the default `nknots=6`, which dictates the number of knots used in the finite rank predictive process approximation. One may consider increasing `nknots` to allow for more waviness or multimodality of the response distribution. While the likelihood computation scales well in `nknots`, the overall MCMC may take much longer to mix when a larger `nknots` is used. The total number of parameters (after reparameterization and discretization) is `(p+1)*(nknots+1) + 3`, where `p` is the number of predictors (excluding intercept).

Several non-default options are employed in the code that follows for the basal-area model. We explain our use of them here:

- *Excess Zero as Censoring.* We repurpose the censoring argument to identify observations that are truncated at zero. Within the vector, `cens=2` indicates left censoring or left truncation and `cens=0` indicates uncensored observations.
- *MCMC Initialization.* The `par="RQ"` option allows us to initialize our regression coefficients in the MCMC chain to be close to the traditional (τ independently estimated) quantile regression estimates.
- *"Base" Distribution.* The `fbase` argument specifies a prior guess for the shape of the distribution at the center of the covariate space. That prior guess will be deformed to match the actual shape of the distribution; however, the estimated tails are designed to retain the decay behavior of the prior guess. The options when modeling on the full real line, albeit truncated to the non-negative reals, are `"logistic"` or `"t"`. We use the `"logistic"` option because 1) we are primarily concerned with estimation in the distribution's bulk and do not desire to guarantee t-like, power decay in the tails and 2) because it runs slightly faster than the "t" option.
- *MCMC Sampling and Thinning.* The `nsamp` argument tells us how many total samples to retain, while `thin` designates how often to retain the MCMC sample. As the output objects can get large and the MCMC chains can exhibit some autocorrelation, we choose to retain every 20^{th} sample. After running `nsamp` * `thin` = 500 * 20 = 10000 total observations, of which only 500 will be retained and displayed, we pause to assess the state of the MCMC chain.

Even this simple model may take a minute or two to run.

```
set.seed(11111)
fit.qrj1 <- qrjoint(baRedMaple ~ elev, data=dat, cens=ifelse(dat$baRedMaple==0,2,0),
            par="RQ", fbase="logistic", nsamp = 500, thin = 20)
#> Initial lp = -3226.81
#> iter = 1000, lp = -1783.68 acpt = 0.21 0.12 0.16 0.16 0.07
#> iter = 2000, lp = -1784.07 acpt = 0.13 0.24 0.14 0.11 0.28
#> iter = 3000, lp = -1782.49 acpt = 0.13 0.15 0.16 0.18 0.22
#> iter = 4000, lp = -1782.23 acpt = 0.12 0.13 0.13 0.11 0.19
#> iter = 5000, lp = -1778.64 acpt = 0.11 0.14 0.16 0.13 0.14
#> iter = 6000, lp = -1778.55 acpt = 0.17 0.15 0.13 0.19 0.16
#> iter = 7000, lp = -1784.54 acpt = 0.16 0.17 0.13 0.12 0.16
#> iter = 8000, lp = -1784.02 acpt = 0.12 0.15 0.16 0.11 0.15
#> iter = 9000, lp = -1781.3 acpt = 0.13 0.14 0.15 0.16 0.13
#> iter = 10000, lp = -1782.77 acpt = 0.16 0.12 0.15 0.15 0.17
#> elapsed time: 49 seconds
```

4.2 MCMC Progress and Convergence Assessment

The output prints, on the fly, the log posterior value at initialization and subsequently prints updates to the log posterior after each 10% of total iterations completed. The MCMC calculation utilizes a blocked adaptive Metropolis sampler (Andrieu and Thoms, 2008) that places the model parameters into `p+4` overlapping groups. At each update, acceptance rates for each block of the adaptive metropolis sampler are also printed. Having not changed the default `acpt.target` option, we are looking for each block to approach the default acceptance target of 0.15, which they are beginning to. The final line of output gives the total run time.

The `summary` function provides insight into the convergence of the MCMC sampler. The `more.details=TRUE` option gives additional diagnostic plots. The suite of plots created by the `summary` call are shown in Figure 2.

```
summary(fit.qrj1, more.details=TRUE)
#> WAIC.1 = 3550.2 , WAIC.2 = 3550.29
```

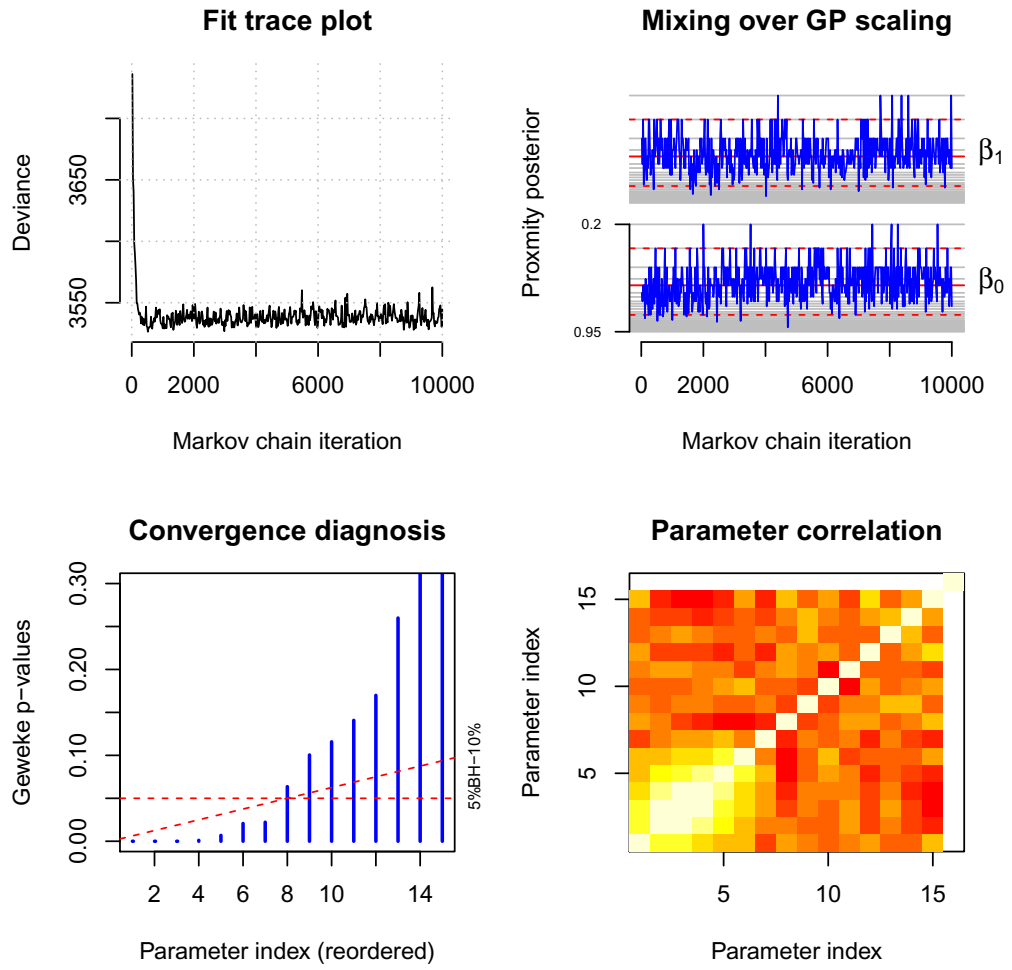

Figure 2: MCMC diagnostics for qrjoint model fit

In Figure 2, the "Fit trace plot" shows that the chain has moved away from its initial values and may be coming closer to a stable state. Here and in the subplot labeled "Mixing over GP scaling", we are looking for "fuzzy caterpillar" plots indicating good mixing, as is typical in evaluating MCMC trace plots. The GP scaling plot shows, for each β_j curve, how much correlation exists between its values at quantile levels 0.1 apart. These proximity parameters are sampled from a discrete set of values over a fixed range, so if we see posterior mass building at either an upper or lower boundary we may need to adjust the `hyper` parameters for `lam` to cover a better range of values. The red lines show prior 95% credible intervals on the proximity parameters.

The "Convergence diagnosis" subplot displays p-values from Geweke tests for convergence. The diagonal line represents a Benjamini-Hochberg adjustment for multiple-testing across parameters (controlling false discovery rate at 10%). Seeing parameters with p-values below the diagonal blue line, as we do here, is one indication that the MCMC chain needs to run longer. The "Parameter correlation" subplot gives a heat-map of the correlation among model parameters.

We use the `update` function to add an additional 500 draws to our sample. The sampler maintains the thinning rate (every 20^{th} observation) specified in the original `qrjoint` call.

```
fit.qrj1 <- update(fit.qrj1, nadd=500)
```

```
summary(fit.qrj1, more.details=TRUE)
#> WAIC.1 = 3550.71 , WAIC.2 = 3550.75
```

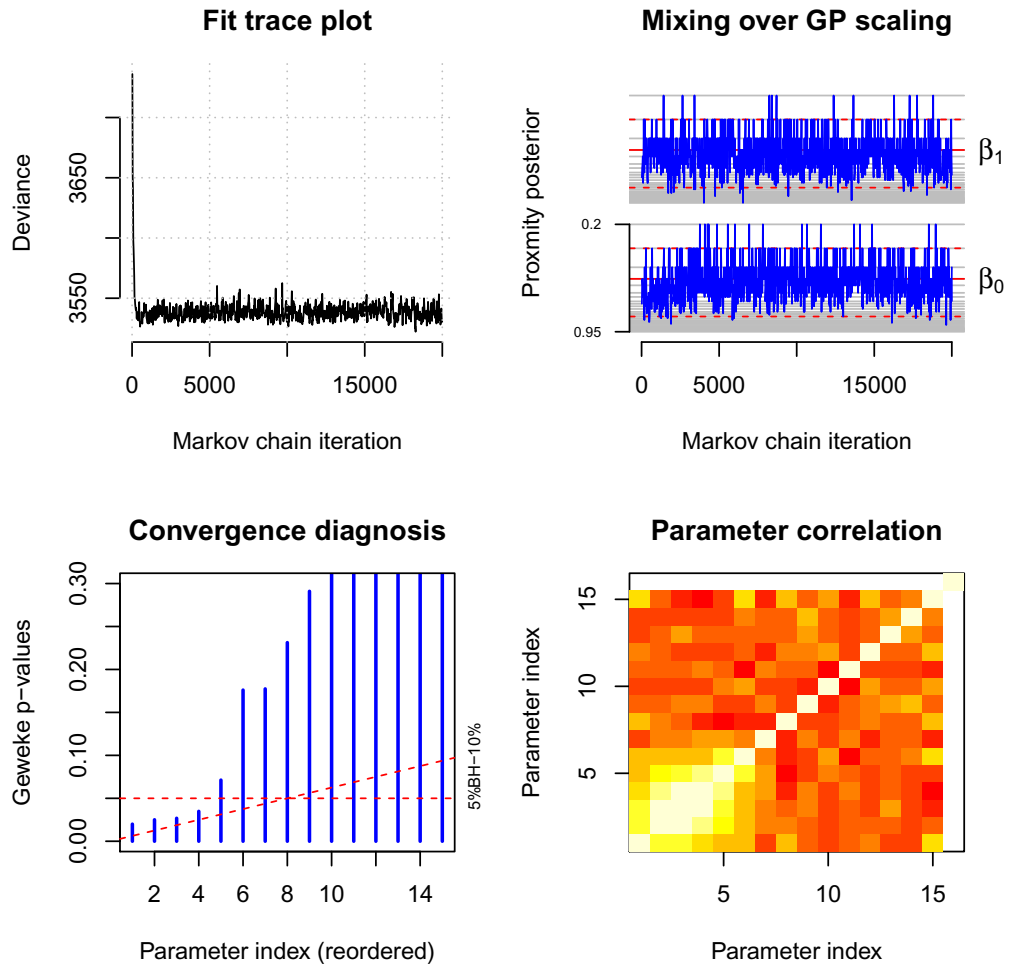

Figure 3: Updated MCMC diagnostics for qrjoint model fit

The MCMC diagnostic plots run on the extended chain, shown in Figure 3, look better. The summary function prints two versions of the Watanabe Akaike Information Criterion, which can be used to compare models (lower WAIC indicates a better fit).

It is possible to run multiple MCMC chains and assess convergence with associated multi-chain diagnostics, e.g. Gelman and Rubin, although we do not do so here. In the `qrjoint` call, setting `par` equal to a numeric vector of length equal to the total number of model parameters can override `par`'s supported options and directly specify desired MCMC starting values.

To recap, 20000 total MCMC iterations have been run using the `qrjoint` and `update` functions, and 1000 of those samples have been retained. We will use the auxiliary functions' default burn-in rates of `burn.perc=0.5` to obtain posterior summaries (medians, 95% credible intervals, etc.) from the second set of 500 retained samples.

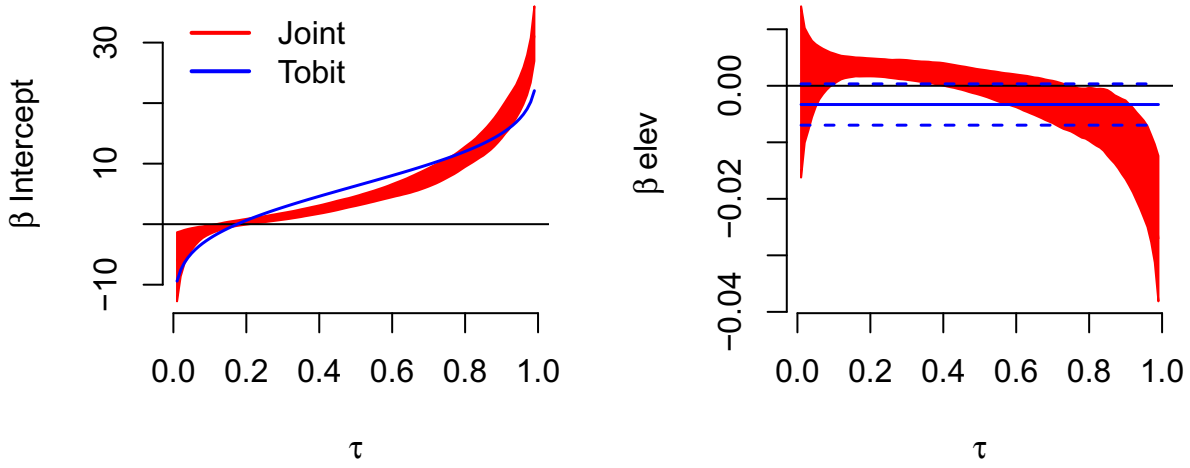

Figure 4: Coefficient estimates and 95-percent intervals across quantile levels for simple basal area model

5 Interpreting Quantile Regressions

5.1 Coefficient Plots

The `coef` function returns posterior samples for intercept and slope parameters at all quantile levels matching the τ-grid used in model fitting. It also returns, as estimates, posterior medians and the end points of the 95% posterior credible intervals of those parameters. By default, the `coef` function also plots the regression coefficients across τ. We suppress plotting in favor of constructing our own plots that also contain the estimated Tobit parameters (Figure 4).

```
tau <- round(fit.qrj1$tau.g[fit.qrj1$reg.ix],2)
coef.qrj1 <- coef(fit.qrj1, nmc = 500, plot = FALSE)
beta.qrj1 <- coef.qrj1$beta.est
finite <- !(tau%in%c(0,1))
p <- dim(beta.qrj1)[2]

beta.tb1 <- array(NA, dim(beta.qrj1), dimnames=dimnames(beta.qrj1))
beta.tb1[,"Intercept","b.med"] <- qnorm(tau, fit.tb1$coef[1], fit.tb1$scale)
for (i in 2:p){
  beta.tb1[,i,"b.lo"]  <- confint(fit.tb1)[i,"2.5 %"]
  beta.tb1[,i,"b.med"] <- fit.tb1$coef[i]
  beta.tb1[,i,"b.hi"]  <- confint(fit.tb1)[i,"97.5 %"]
}

varname <- dimnames(coef.qrj1$beta.samp)$beta
par(mfrow=c(1,2))
for(i in 1:p){
  getBands(coef.qrj1$beta.samp[,i,], xlab=bquote(tau), ylab=bquote(beta~.(varname[i])), bty='n')
  abline(h=0)
  matlines(tau[finite], beta.tb1[finite,i,], col="blue", lty=c(2,1,2), lwd=1.5)
  if(i==1) legend("topleft", c("Joint","Tobit"), col=c("red","blue"), lty=1, lwd=2, bty='n')
}
par(mfrow=c(1,1))
```

The estimates and intervals at $\tau = 0.5$ correspond to median regressions, whereas those at $\tau = 0.8$ correspond to the 80^{th} percentile regression, and so on. When looking at these plots, three types of comparisons are

useful. We illustrate by interpreting the `elev` plot.

Comparisons to zero. Tobit's slope estimates are constant for all parts of the response distribution. Because the 95% confidence interval bands contain zero, the Tobit regression might lead us to conclude that `elev` is not linearly related to `baRedMaple`. The 95% intervals from the `qrjoint` fit, however, *do* have non-zero coefficients. The positive bands in the τ-region of (0.1, 0.4) means that an increase in elevation is associated with increased basal areas, but only for those low-to-mid quantile levels. When we consider the upper quantiles, i.e. $\tau > 0.8$, an increase in elevation is actually associated with a decrease in red maple basal areas. These interpretations are similar to traditional interpretations of a regression model; however, here we are able to make inferential claims for all parts of the response distribution and not just for the mean or median.

Comparisons between quantile levels, τ. The increasingly negative slopes for `elev` across τ in the joint quantile regressions illustrate a differential effect of elevation on basal area at different places in the response distribution. The lower quantile levels have positive slopes, whereas the upper quantile levels have negative slopes. This likely reflects a fanning of the data with larger variance at small `elev` values and smaller variance at large `elev` values. In this way, quantile regression can capture heterogeneity of variance. Tobit, with its flat slopes across τ, is not capable of capturing heterogeneity of variance or other types of differential effects.

Comparisons between methods. Finally, a visual comparison of interval estimates between methods at any given τ shows overlap or concordance between the joint quantile regression and the Tobit regression in parts the lowest decile and for τ in (0.4,0.95).

5.2 Quantile Line Plots

It may be instructive in this single-variate case to plot the regression lines for a few τ values.

```
# Retrieve subset of tau-fitted lines
tau.use <- round(c(0.01, 0.05, seq(0.1,0.9,by=.1), 0.95,0.99),2)
tau.factor <- factor(tau.use, levels=rev(tau.use))
beta1 <- beta.qrj1[which(tau %in%tau.use),,"b.med"]
beta1 <- data.frame(Level=factor(rownames(beta1),levels=levels(tau.factor)), beta1)

# Use quantile-fitted coefficients to add ablines to scatterplot of data
p.dat <- ggplot() + geom_point(data=dat, aes(x=elev, y=baRedMaple), col="#999999") +
     ylab("Red Maple Basal Area")
p.dat + geom_abline(data=beta1, aes(slope=elev, intercept=Intercept, col=Level))
```

Figure 5 shows that approximately 1% of the observations lie above the 0.99 quantile line, about 50% of the observations lie above the 0.5 line, etc. Note that the regression lines do not cross within the range of the `elev` covariate, i.e. they obey the monotonicity constraint. Here the heterogeneity of variance across `elev` is visible, and it makes sense that regression slopes generally progress from positive to negative slopes as τ, the quantile level, is increased. Some lines extend below zero because the plotted lines are estimates for the latent or non-truncated model.

The above comparisons were made for didactic purposes. Prior to interpreting coefficients, an assessment of the model assumptions is warranted.

6 Assessing Model Assumptions and Making Improvements

After the assumption of independence, joint quantile regression has only one other assumption: all effects can be explained as linear combinations of the design matrix X. The Tobit model additionally assumes that the latent responses are normally distributed with constant variance across all observations. A first instinct may be to turn to "residual" diagnostics for evaluation of these model assumptions, where residuals are traditionally defined as the difference between an observed value and its predicted mean. Diagnostics

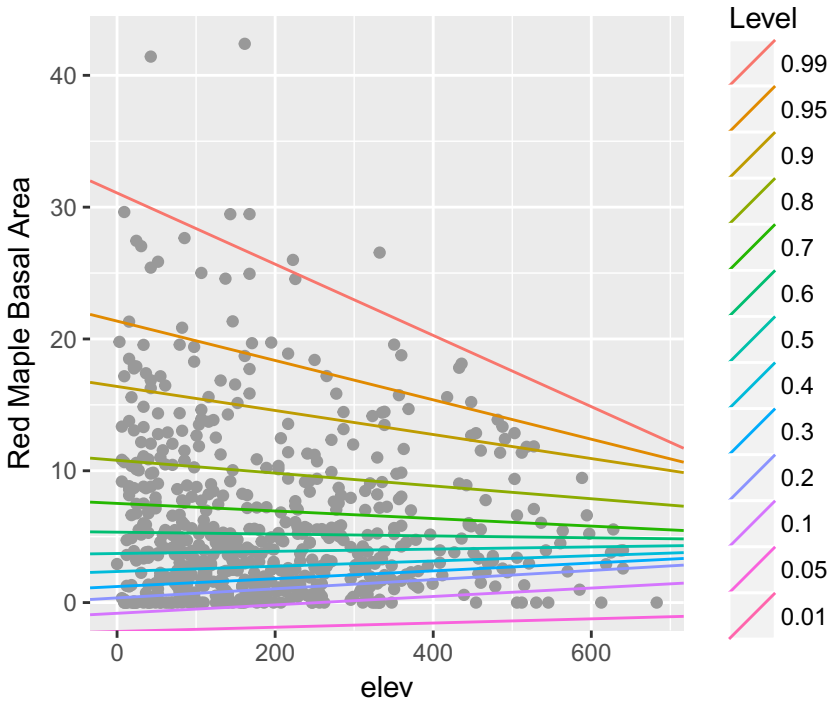

Figure 5: Quantile regression lines

based on residuals may be sufficient when the (assumed) response distribution can be summarized by its mean, as in the case of the Tobit model; however, they are insufficient for joint quantile regression, which outputs a conditional prediction that is an *entire response distribution*.

With estimated quantile functions available for the Tobit model *and* the joint quantile regression model, diagnostics based on the probability integral transform are possible for both models. If the models are appropriate, the estimated quantile levels for the observations, $\hat{\tau}_{y_i} = \hat{Q}^{-1}(y_i|x_i)$, should follow a uniform distribution.

6.1 Obtaining Estimated Quantile Levels

Under the Tobit model, $\hat{\tau}_{y_i}$ for $y_i > 0$ is estimated using the normal CDF, $\Phi((y_i - x_i\hat{\beta})/\hat{\sigma})$. Under the joint quantile regression model, $\hat{\tau}_{y_i} = \hat{Q}^{-1}(y_i|x_i)$. The `summary` function carries out this inverse calculation by interpolating the estimated quantile lines between τ-grid points. For either model, if $y_i = 0$ then $\hat{\tau}_{y_i}$ can be taken as a random draw from $Unif(0, \hat{Q}^{-1}(0|x_i))$. Estimated quantile levels for each observation and each draw of the MCMC sampler can be retrieved from the `summary` function for `qrjoint`. The function that follows obtains the quantile levels, corrects them under censoring (or in this case under zero-truncation), and summarizes them across posterior draws. We demonstrate, using the "Summary" option.

```
# Function to summarize posterior draws of tau_Y. When fit includes censored values,
# left-censored tau_Y are replaced with draw from Unif(0, tau_Y) and right censored tau_Y
# are replaced with draw from Unif(tau_Y,1)
#
# Inputs
# fit: qrjoint fit object
# plot: If TRUE, plot produces a histogram and qq-plot comparing to uniform distribution
# mcmc: Character string describing how to summarize over posterior draws. Options are
#       "Summary" - takes mean of tau over posterior draws
```

```
#       "One" - returns tau at a single random MCMC iteration
#       "Many" - returns tau at all MCMC iterations

modfit.qrjoint  <- function(fit, burn.perc=0.5, mcmc="Summary"){
  invisible(capture.output(ql <- summary(fit, plot.dev=FALSE)$ql))
  cens <- fit$cens; nsamp <- ncol(ql)
  ql <- ql[, 1:nsamp > nsamp * burn.perc]
  if(mcmc=="Summary") {
    MCMC <- apply(ql, 1, mean)
    if(!is.null(cens)){
      MCMC[cens==2] <- runif(sum(cens==2),0, MCMC[cens==2])
      MCMC[cens==1] <- runif(sum(cens==1), MCMC[cens==1], 1)
    }
  }
  if(mcmc=="One"){
    MCMC <- ql[,sample(1:ncol(ql),1)]
    if(!is.null(cens)){
      MCMC[cens==2] <- runif(sum(cens==2),0, MCMC[cens==2])
      MCMC[cens==1] <- runif(sum(cens==1), MCMC[cens==1], 1)
    }
  }
  if(mcmc=="Many")    {
    if(!is.null(cens)){
      ql[cens==2,] <- runif(length(ql[cens==2,]),0, ql[cens==2,])
      ql[cens==1,] <- runif(length(ql[cens==1,]), ql[cens==1,], 1)
    }
    MCMC <- ql
  }
  invisible(MCMC)
}

set.seed(22222) # Censoring corrections perform stochastic operation
dat$pfit.qrj1 <- modfit.qrjoint(fit.qrj1, mcmc="Summary")
dat$pfit.tb1 <- pnorm(dat$baRedMaple, mean=predict(fit.tb1), sd=summary(fit.tb1)$scale)
dat$pfit.tb1[dat$baRedMaple==0] <- runif(sum(dat$baRedMaple==0),0,dat$pfit.tb1[dat$baRedMaple==0])
```

We store them in the data frame containing the original data for convenience when assessing assumptions of linearity.

6.2 Assessing Overall Fit

A PP-plot may be used to compare the estimated quantile levels to their equivalent uniform probabilities.

```
p.qqtb1 <- ggplot() + geom_qq(aes(sample=dat$pfit.tb1), distribution=stats::qunif) +
  ylab("actual") + ggtitle("Tobit Model") + geom_abline(intercept=0, slope=1)

p.qqqrj1 <- ggplot() + geom_qq(aes(sample=dat$pfit.qrj1), distribution=stats::qunif) +
  ylab("actual") + ggtitle("Joint QR Model") + geom_abline(intercept=0, slope=1)

grid.arrange(p.qqtb1, p.qqqrj1, ncol=2)
```

In Figure 6, the joint quantile regression lies close to the forty five degree line, showing similarity to a uniform distribution and indicating good aggregate model fit, while the Tobit model is decidedly non-uniform, indicating a poor fit.

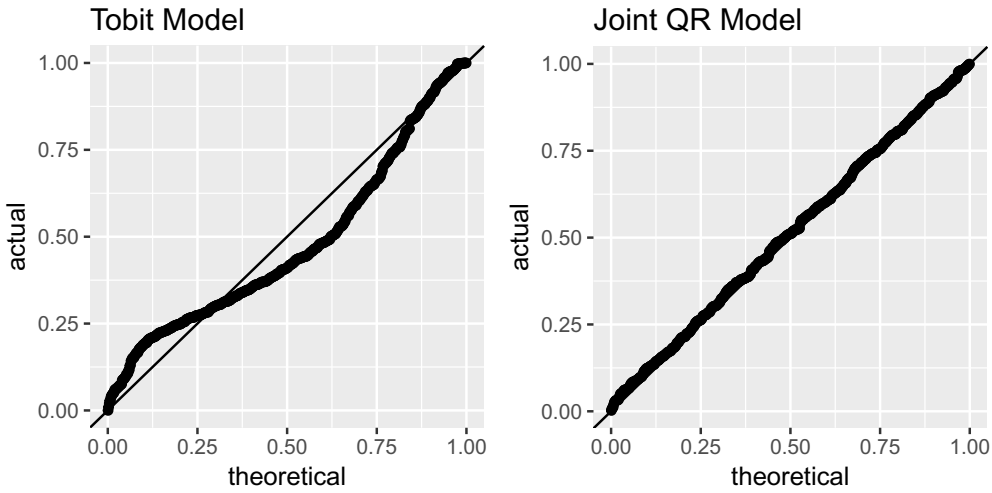

Figure 6: Estimated quantile-level plots for assessing overall model fit

6.3 Assessing Linearity

The estimated quantile levels, $\hat{\tau}_{y_i}$, also provide a way to diagnose design matrix misspecification and assess the assumption of linearity. At any given X, we expect the estimated quantile levels to be uniform. Therefore plotting a covariate against $\hat{\tau}_{y_i}$ and looking at any vertical cross section, the estimated data quantile levels should be uniform within the swath.

We illustrate two options for quantile-level diagnostic plots using the joint quantile regression model. Scatter plots with mean-trend gam/loess lines are illustrative for diagnosing potential non-linearities; trend lines should lie close to a horizontal line at the constant value of 1/2. Violin plots, which cut the continuous variables into quantile bins (here deciles) and then display kernel density estimates within each bin, can also assist in diagnosing non-linearity. These violin plots should look uniform or blocky within each bin.

```
library(dplyr)      # for binning into deciles

qlplot <- function(data, x, y, plot=TRUE){
  if(is.numeric(data[,deparse(x)])){
    data$bin <- factor(ntile(data[,deparse(x)],10))   # bin numeric
    p.s <- ggplot(data, aes_q(x, y)) + geom_point() + geom_smooth(se=F, method="loess")
    p.v <- ggplot(data, aes_q(quote(bin), y)) + geom_violin() + xlab(paste("Decile bins of",x))
  } else{
    p.s <- ggplot(data, aes_q(x, y)) + geom_point()
    p.v <- ggplot(data, aes_q(x, y)) + geom_violin()}

  yax <- list(ylim(0,1), ylab("Estimated quantile level"))
  if(plot) {grid.arrange(p.s + yax, p.v + yax, ncol=2)} else{
    invisible(arrangeGrob(p.s + yax, p.v + yax, ncol=2))
  }
}

qlplot(dat, x=quote(elev), y=quote(pfit.qrj1))
```

As seen in Figure 7, the estimated data quantile levels plotted against `elev` show slight non-linearity; for low elevations the mean trend line bows upward, away from zero, and the violin plot is somewhat top-heavy. The bowing downward at high elevations is likely driven by a few outlying-in-x elevation values and not a systematic departure from uniformity. The mostly-uniform densities in decile bins 9 and 10 help to confirm

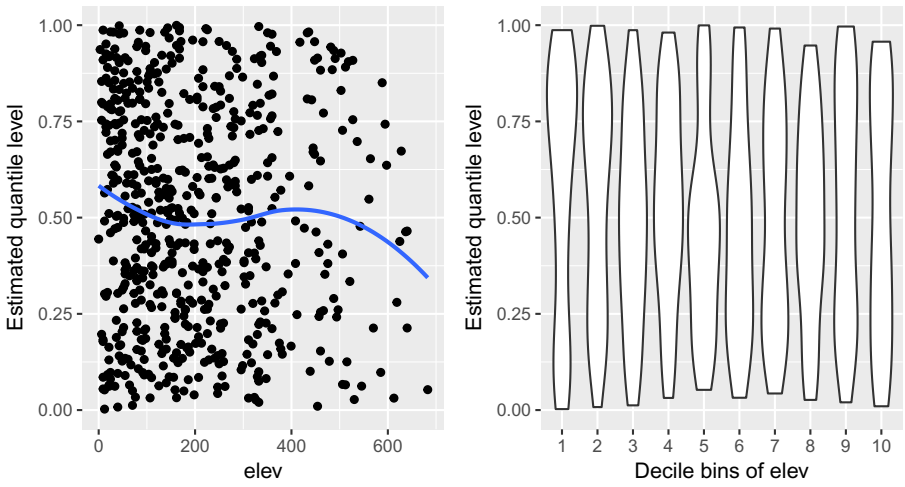

Figure 7: Covariate by estimated data quantile-levels plots for assessing linearity assumption

this.

While the information in these two plots is similar, the violin plots can be helpful for consolidating sparse regions of the covariate space, e.g. with outlying `elev` values, or for spreading out dense regions. Overall, the mean trend-line for `elev` does not depart egregiously from 0.5. Lacking some physically-justified motive for a non-linear elevation effect, some might reasonably elect to keep this covariate in its linear form. For illustration, we modify the design matrix by including a third-order b-spline for `elev`.

6.4 Model Improvement

Using the quantile-level diagnostic plots of the previous section, we deduced that elevation's effect on red maple basal areas may not be linear and that our model might be improved by a b-spline transformation. As was mentioned previously, the linear model can be compared to the spline model using WAIC, which is calculated in `qrjoint`'s auxiliary `summary` function.

```
library(splines)   # for b-splines
set.seed(33333)
fit.qrj2 <- qrjoint(baRedMaple ~ ns(elev,3), data=dat, cens=ifelse(dat$baRedMaple==0,2,0),
        par="RQ", fbase="logistic", nsamp = 2000, thin = 20)
```

```
summary(fit.qrj2, plot.dev=FALSE)
#> WAIC.1 = 3531.35 , WAIC.2 = 3530.85
```

The spline-model run-time is longer both because of the increase in number of predictors and because more iterations were needed to reach convergence; however, it seems to pay off. The WAIC has decreased from ≈ 3550 to ≈ 3531, indicating an improved fit with the `elev` spline. Also, the quantile-level plots have less bowing near zero as can be seen in Figure 8. If the the non-linear effect of `elev` were of specific interest, additional degrees of freedom could be added to the b-spline basis until the practitioner is satisfied with the uniformity of quantile-level plots or until WAIC indicates over-fitting. As our specific interest lies in the multiple regression model, we leave off further model modifications for now.

```
set.seed(44444)
dat$pfit.qrj2 <- modfit.qrjoint(fit.qrj2, mcmc="Summary")
qlplot(dat, x=quote(elev), y=quote(pfit.qrj2))
```

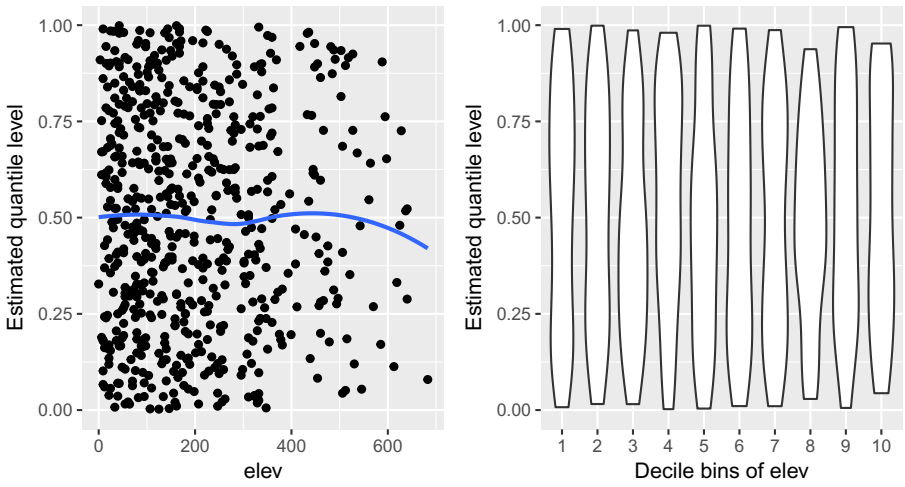

Figure 8: Covariate by estimated data quantile-levels plots after including b-spline

7 Prediction and Interpreting Predicted Responses

7.1 Quantiles for Positive Reals

We may desire to apply our fitted model to obtain predictions for a new data set. When doing this, it is important to remember that the joint quantile regression fit is only guaranteed to provide non-crossing quantile planes within the convex hull of the data upon which the model is fit. The code that follows uses the `predict` function to produce quantile line plots similar to those originally made using `coef`.

```
# Define new dataset and perform prediction
pred.grid <- seq(min(dat$elev),max(dat$elev), length=50)
dat.new <- data.frame(elev=pred.grid, baRedMaple=999)
pred1 <- predict(fit.qrj1, newdata=dat.new)
dimnames(pred1) <- list(elev=pred.grid, Level=tau)

library(reshape)      # for melting from wide to long for ggplot
pred1.long <- melt(pred1[,tau %in%tau.use])
pred1.long$Lev <- factor(pred1.long$Level,levels=levels(tau.factor))
p.dat + geom_line(data=pred1.long, aes(x=elev, y=value, col=Lev, group=Lev)) +
    coord_cartesian(ylim=c(0,43)) + labs(col="Level")
```

Alternately, we could build a new X matrix and get predictions through the matrix multiplication $X\beta$. This method is preferred when predicting on the spline model because it guarantees that the b-splines bases over the new data are the same bases upon which the regression is fit.

```
# Get beta from coef, X from predict on spline object, prediction from matrix mult
library(splines)
beta2 <- coef(fit.qrj2)$beta.est
splines <- ns(dat$elev,3)
Xnew2 <- cbind(1,predict(splines, dat.new$elev))
pred2 <- Xnew2%*%t(beta2[,,"b.med",drop=TRUE])
dimnames(pred2) <- list(elev=pred.grid, Level=tau)
pred2.long <- melt(pred2[,tau %in%tau.use])

# Plot quantile lines using continuous gradient for tau
our.palette <- hcl(h=seq(375,55,length=9), l=65, c=100)
```

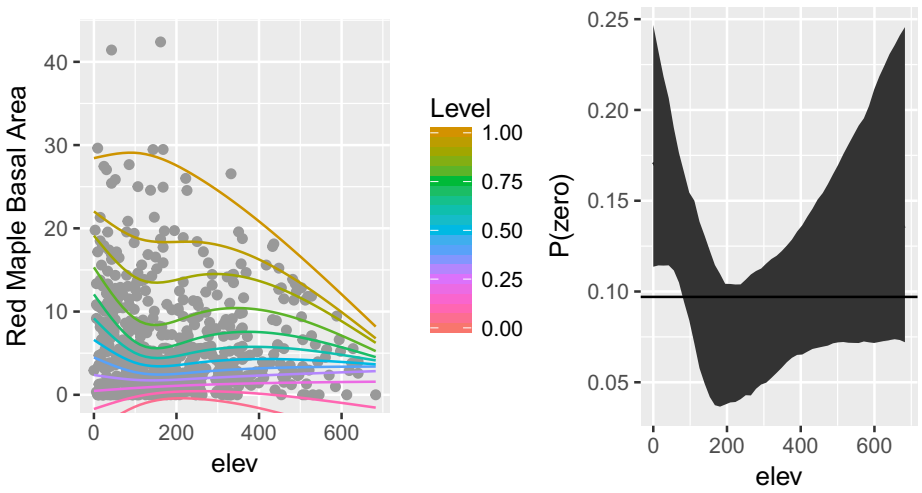

Figure 9: Predictions for elevation spline model. Left: Quantile lines. Right: Zero probabilities.

```
s.lev <- scale_color_gradientn(limits=c(0,1), colors=our.palette)
p.quants <- p.dat + coord_cartesian(ylim=c(0,43)) + s.lev +
     geom_line(data=pred2.long, aes(x=elev, y=value, col=Level, group=Level))
```

The first plot of Figure 9, created from the code above, shows that even when there is more than one predictor, e.g. in this case three b-spline bases, the quantile planes do not cross.

7.2 Probability of Zero

We may also desire to know the probability of having no red maple trees at a given site. These probabilities are equivalent to the probability of censoring (truncation) and can be obtained under the joint modeling context. Using the conditional quantile predictions for some x at every MCMC draw, we obtain $\tau_0 = Q^{-1}(0|x))$, the quantile level corresponding to when the conditional quantile equals zero, by linear interpolation between estimated grid points. Summarizing these values across draws produces posterior intervals for the probability of having zero basal area at a site with given covariates, P(zero). In this case with all predictors derived from a single covariate, we are able to aggregate into one plot the effects of the three `elev` b-splines on the probability of zero using the code that follows.

```
  nsamp <- 500
  coef2 <- coef(fit.qrj2, reduce=FALSE, nmc=nsamp)$beta.samp
  tauplus <- round(fit.qrj2$tau.g,8)

  # Probability of zero with error bars across elev
  Q2 <- sapply(1:nsamp, function(f) Xnew2%*%t(coef2[,,f]), simplify="array")
  tau0 <- apply(Q2,c(1,3), function(f) approx(f, tauplus,0)$y)
  prob0 <- as.data.frame(t(apply(tau0, 1, quantile, prob=c(0.025, 0.5, 0.975))))
  prob0$elev <- pred.grid
  p.p0 <- ggplot(prob0, aes(x=elev)) + geom_line(aes(y=`50%`)) +
    geom_ribbon(aes(ymin=`2.5%`, ymax=`97.5%`), alpha=0.3) + ylab("P(zero)") +
    geom_hline(yintercept=mean(dat$baRedMaple==0))

grid.arrange(p.quants, p.p0, ncol=2)
```

In the output plot (second plot of Figure 9), we compare the zero probability bands to the sample prevalence of zeroes, displayed as a black horizontal line. Bands that are fully above (or fully below) the sample prevalence

indicate an increase (or decrease) in zero-probability for values in that range of `elev`. Here we conclude that lower elevations are less likely to have red maple trees.

While estimating a given quantile or probability of zero is straightforward for any single observation's set of covariates, creating multiple-regression analogs of the plots in Figure 9 using two or more covariates is more difficult. The functional-β-coefficient plots and the quantile-level diagnostic plots translate seamlessly from a single-covariate setting to a multiple-regression setting.

8 Fitting Multiple-Regression Basal Area Models

8.1 Model Terms, Transformations, and Interactions

The four covariates that we are interested in exploring simultaneously in a multiple, quantile regression setting are `elev`, `region`, `aspect`, and `slope`. Before starting, we compile a list of notes and questions to address during modeling:

- *Dealing with Directional Covariates.* The covariate `aspect` is radial or wrapping in nature, with values 360 and 1 being adjacent degree measurements. A common way to treat radial data is to include both `cos` and `sin` bases. This transformation makes `aspect` less unit-interpretable (i.e. slope can no longer be interpreted as "a one unit increase in degrees corresponds to a x unit increase in basal area...") but more interpretable in terms of cardinal directionality. For these data, a `cos` transformation measures southerliness-to-northerliness (-1 to 1 respectively), while a `sin` transformation measures westerliness-to-easterliness (-1 to 1 respectively). Depending on sun and shade tolerance, some trees prefer north or south, east or west facing slopes. Do red maple trees?
- *Partially Deterministic Relations Between `aspect` and `slope`.* On a related note, a site cannot face a direction unless it is sloped. The `aspect` covariate records "0" for many sites that have zero or near-zero slopes. To prevent these values from influencing the directional effect of `aspect`, an indicator value can be added to let flat sites have their own adjusting intercept.
- *Interaction Effect.* One may well suspect some interaction between `slope` and `aspect`. For instance, the east-westerly effect on red maple basal areas may be different for moderately sloped sites than for steeply sloped sites. Is an interaction necessary for describing the quantiles of red maple basal areas?
- *Categorical Covariate Encoding.* The EPA Level-III `region` variable transcends state boundaries to categorize sites into roughly similar geophysical regions. In these data, there are only three regions: the Atlantic Coastal Pine Barrens (13 sites), the Northeastern Coastal Zone (393 sites), and the Northeastern Highlands (202 sites). These regions may stand as rough proxy for soil covariates such as sand, rock, or clay composition, which are not included in the data but which one might imagine be related to tree growth for a given species. By default, `R` will use the Atlantic Coastal Pine Barrens as a reference category and code the other two regions using indicator variables.
- *Dependence Between Region and Elevation.* Finally, the variables `region` and `elev` are highly related (see Figure 10), having different-though-overlapping ranges of elevation per region. It would be interesting to know if both variables are needed in the regression model or if the effect of `region` on red maple basal areas subsumes the need for an `elev` effect or vice versa.

We create a model that includes all covariates along with the necessary transformations and interactions to test for the effects listed.

```
set.seed(55555)
fit.qrj3 <- qrjoint(baRedMaple ~  slope*(I(cos(aspect*pi/180)) + I(sin(aspect*pi/180))) +
            I(aspect==0) + region + elev, data=dat, cens=ifelse(dat$baRedMaple==0,2,0),
            par="RQ", fbase="logistic", nsamp = 2000, thin = 20)
```

```
summary(fit.qrj3, more.details=TRUE)
#> WAIC.1 = 3455.15 , WAIC.2 = 3455.95
```

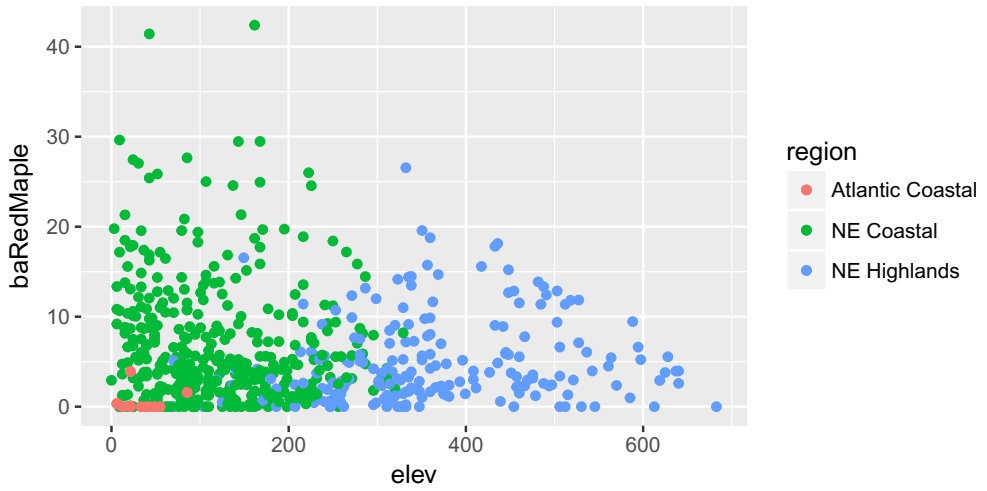

Figure 10: Region may share some of the same effect on basal area as elevation.

The plots of Figure 11 give us some confidence that the MCMC sampler is converging and that we are able to continue with our model diagnostics.

```
dat$pfit.qrj3 <- modfit.qrjoint(fit.qrj3, mcmc="Summary")

p.ql1 <- qlplot(dat, x=quote(slope), y=quote(pfit.qrj3), plot=FALSE)
p.ql2 <- qlplot(dat, x=quote(elev), y=quote(pfit.qrj3), plot=FALSE)
p.ql3 <- qlplot(dat, x=quote(aspect), y=quote(pfit.qrj3), plot=FALSE)
p.ql4 <- qlplot(dat, x=quote(region), y=quote(pfit.qrj3), plot=FALSE)

grid.arrange(p.ql1, p.ql2, p.ql3, p.ql4, ncol=1)
```

8.2 Assessing Model Assumptions

The uniformity across quantile-level plots (Figure 12) is sufficient that we feel confident in interpreting these regression parameters; however, there may yet be some room for improvement. The elevation variable is exhibiting similar bowing in its mean trend line as that seen in the single-variate, elevation regression. Perhaps the model could be aided by reintroducing the b-spline for elevation? We register this modification for future model iterations but first take a look at the coefficients from the model.

```
coef(fit.qrj3, nmc = 500, plot=TRUE, show.intercept=FALSE)
```

8.3 Interpreting Coefficients

The coefficients plots are show in Figure 13. A description of the effects follows:

- The reference intercept distribution (not plotted) corresponds to a Atlantic-Coastal-region, at-sea-level site that has some directional aspect, yet is supposedly flat. Since this site only exists in theory, the intercept is not worth interpreting.
- The indicator variable for `aspect==0` is non-zero for most τ and thereby performs an adjustment to the reference intercept distribution. Without an interpretable reference intercept, this adjusted intercept distribution will not be interpretable either.
- The two categorical `region` indicators have 95%-interval bands fully above zero for all τ; we can say that a Northeastern region site has basal areas about 6 ft^2/acre greater than an Atlantic Coastal site with otherwise equivalent covariates.

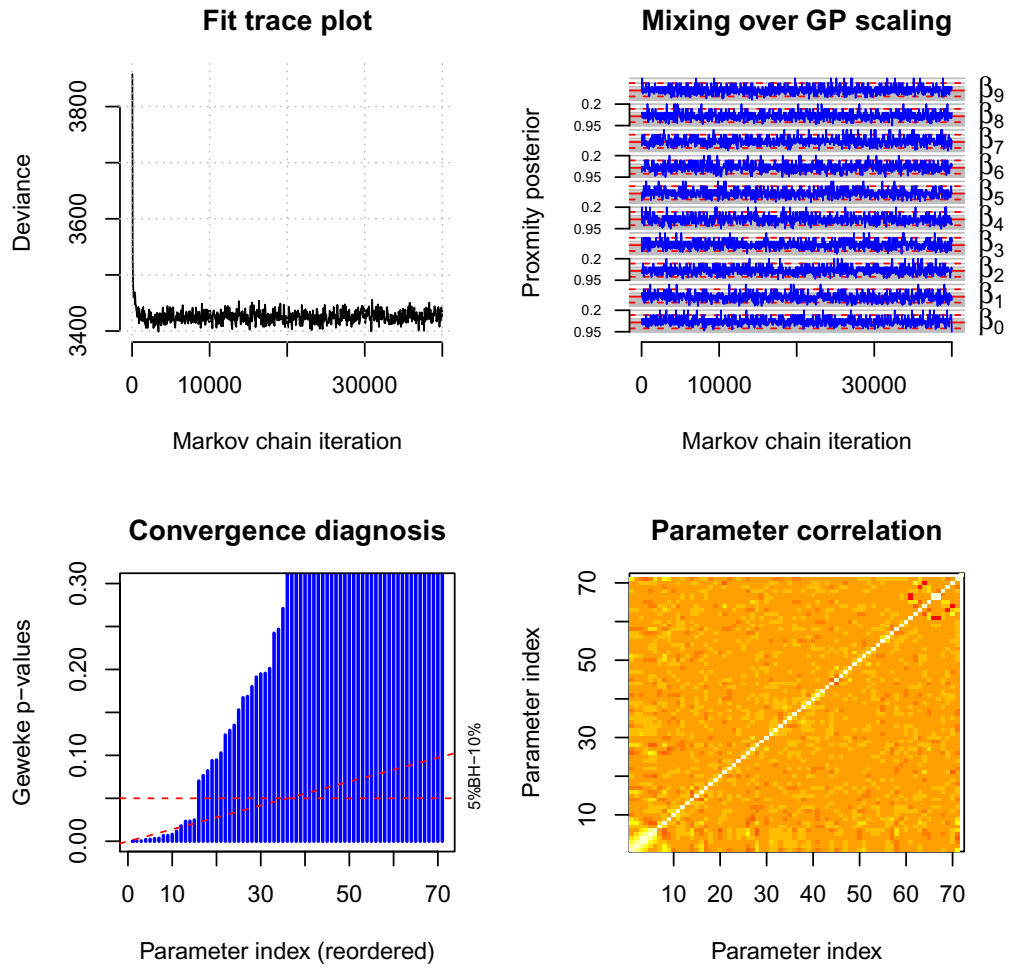

Figure 11: MCMC diagnostics for full qrjoint model, fit.qrj3

- It seems that both `region` and `elev` have effects, because the slope function of `elev` is constant above zero for all but the highest and lowest τ. For quantile levels with bands fully above zero, we would conclude that increased elevation corresponds to greater red maple basal areas, at least within the range of `elev` considered (0 ft to 682 ft). Although a coefficient of 0.004 seems small, the cumulative effect over the sample range could amount to a 0.004*682 = 2.7 difference in basal areas. Contrasting this positive, τ-flat effect to the effect found in the linear-elevation model, the quantiles in the upper portion of the response distribution must be explained by some newly-included variable because we no longer see negative coefficients for large τ.
- The coefficient for `slope` shows a differential effect, growing increasingly negative as τ increases. One can conclude that steeper sites have smaller red maple basal areas (bands below zero for τ greater than about 0.3). The decreasing differential effect also points to a decrease in variance of basal area as `slope` increases, i.e. heterogeneity of variance.
- Neither the marginal `cos` nor `sin` effect for `aspect` is significantly different than zero. Perhaps we should have anticipated this since these terms correspond to cardinal-direction effects when `slope==0`, and as we said previously, `aspect` only has meaning when `slope` is non-zero. In future model iterations, leaving these marginal variables out will have the effect of fixing them equal to zero.
- The `slope:sin(aspect)` term has a negative coefficient, pointing to an interaction between `slope` and the West-East variable for quantile levels $\tau > .3$. We would like to understand this interaction better.

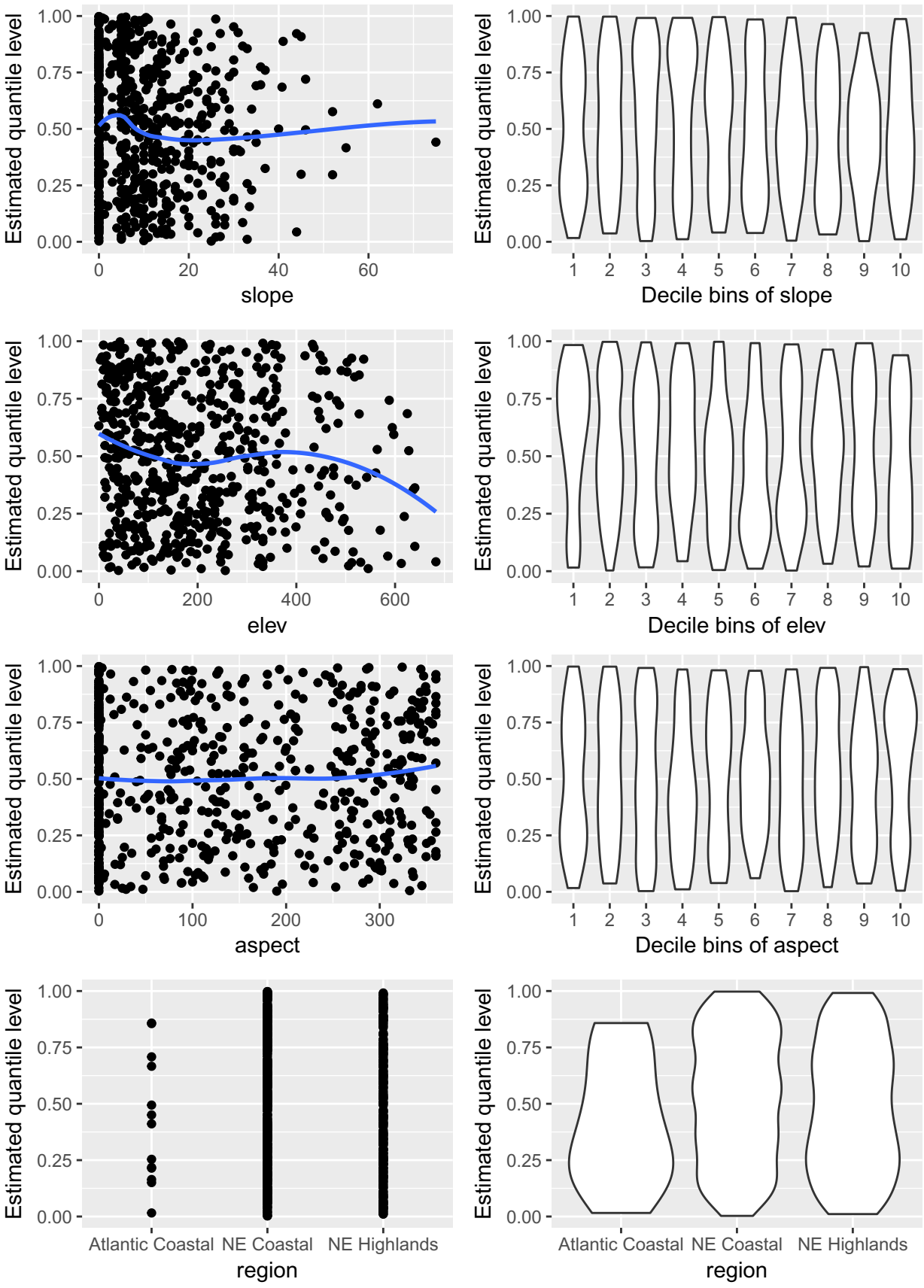

Figure 12: Quantile-level diagnostic plots from full regression model

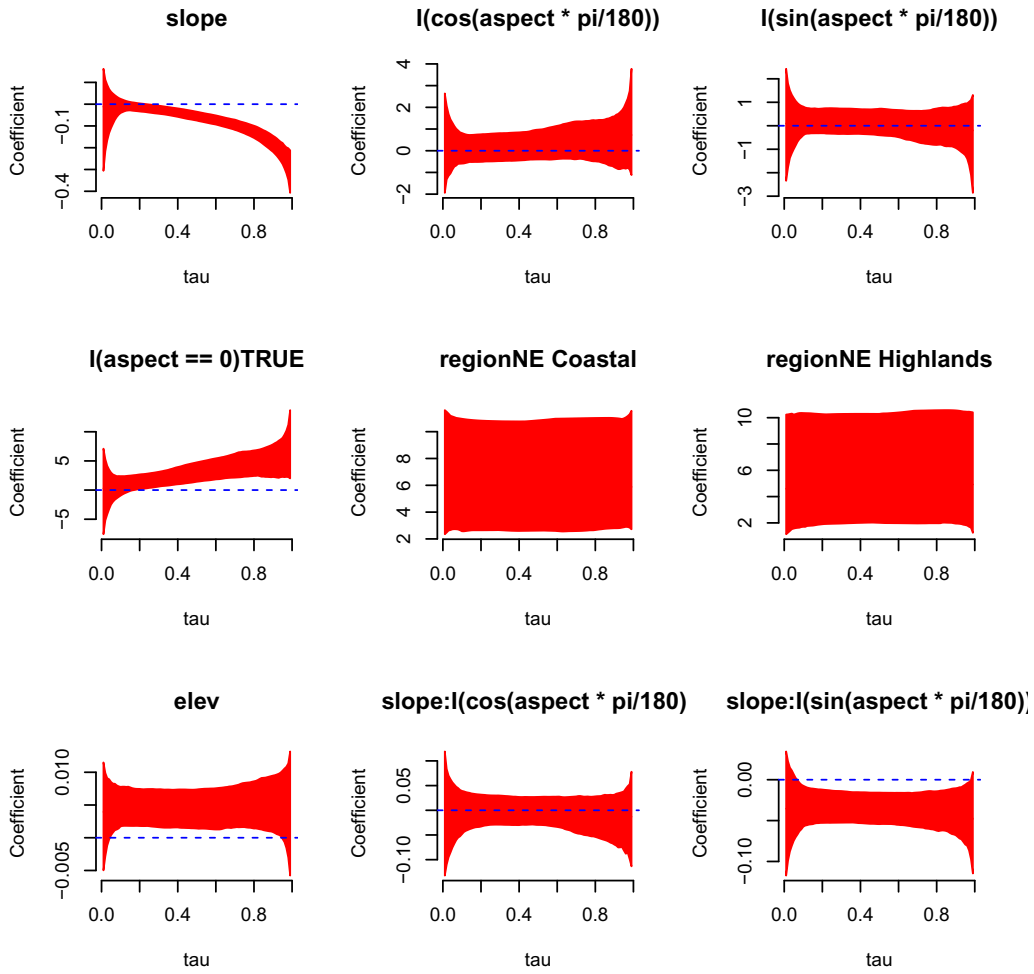

Figure 13: Joint quantile regression coefficients from full regression model

8.4 Understanding Marginal and Interaction Effects

To understand the direction and magnitude of an interaction effect we `predict` over a dataset in which the interacting covariates have been varied over some interpretable range (being careful not to extrapolate out of the convex hull created by the original data) and the remaining covariates have been fixed. Plots are then made of the predicted quantiles across the varied covariates for select τ to visualize their interaction effect. This technique can also be used to plot a single variable's marginal effect, but it is especially helpful when trying to understand interactions.

To tease out the `slope`-by-`aspect` interaction, we vary `slope` between 0 and 50 degrees and `aspect` between 0 to 360 degrees, while arbitrarily fixing `region=="NE Coastal"` and `elev==median(elev)`.

```
newdat <- expand.grid(slope=seq(0,50,by=5), aspect=seq(0,360,length=9)[-1],
          elev=median(dat$elev), region=factor("NE Coastal", levels=levels(dat$region)),
          baRedMaple=999)
newdat$cos <- round(cos(newdat$aspect*pi/180),2)
newdat$sin <- round(sin(newdat$aspect*pi/180),2)

pred3 <- as.data.frame(predict(newdata=newdat, fit.qrj3)) # default summary is posterior median
tau0 <- apply(pred3, 1,function(f) approx(f, seq(0,1,length=101), 0)$y)
pred <- cbind(newdat, pred3, tau0)
```

```
# Makes nicer radial plots when 0 and 360 in data. Not true meaning of/prediction for aspect=0
pred.north <- pred[pred$aspect==360,]
pred.north$aspect <- 0
pred <- rbind(pred.north, pred)

plotrad <- function(y, ylabel){
  slp <- quote(slope); asp <- quote(aspect); qsin <- quote(sin); qcos <- quote(cos)

  p1 <- ggplot() + geom_line(data=pred, aes_q(x=asp, y, group=slp, col=slp)) +
                   scale_x_continuous(breaks=c(90,180,270,360))
  p2 <- ggplot() + geom_line(data=pred, aes_q(x=asp, y, group=slp, col=slp)) + coord_polar() +
                   scale_x_continuous(breaks=c(45,90,135,180,225,270,305,360),
                                      labels=c("NE","E","SE","S","SW","W","NW","N"))
  p3 <- ggplot() + geom_line(data=subset(pred, cos>=0), aes_q(x=qsin, y, group=slp, col=slp)) +
                   geom_line(data=subset(pred, cos<=0), aes_q(x=qsin, y, group=slp, col=slp))
  p4 <- ggplot() + geom_line(data=subset(pred, sin>=0), aes_q(x=qcos, y, group=slp, col=slp)) +
                   geom_line(data=subset(pred, sin<=0), aes_q(x=qcos, y, group=slp, col=slp))

  tmp <- ggplot_gtable(ggplot_build(p1))    # Only print one guide box
  legend <- tmp$grobs[[which(sapply(tmp$grobs, function(x) x$name) == "guide-box")]]
  addend <- list(theme(legend.position="none"), ylab(ylabel))

  grid.arrange(arrangeGrob(p1 + addend, p2 + addend, ncol=2, widths=c(4,5)),
               arrangeGrob(p3 + addend, p4 + addend, legend, ncol=3, widths=c(4,4,1)))}

plotrad(quote(`0.5`), "Median basal area")
```

Figure 14 shows a suite of median regression plots, each intended to aid in interpreting the `slope` by radial `aspect` interaction. Without adding error bars, which would make these already busy plots even more difficult to interpret, these plots can only suggest the magnitude and direction of the effects on red maple basal areas:

In the first plot, by picking a particular `slope` we generally see that the median basal area is greater for `aspect` near 270 (West) than it is for `aspect` near 90 (East); however, the differential is more pronounced the steeper the `slope` of the site is. Another way to think of the interaction is that there are bigger decreases in median basal areas when comparing an eastward facing 50-degree-sloped site to its mostly-flat counterpart than there are when comparing a westward facing 50-degree-sloped site to *its* mostly-flat counterpart.

This interaction plays out in the second, radial plot by having near-circles (no directional effect) for near-flat slopes, but then relatively bigger basal areas for westerly facing sites as slope increases.

In the third plot, the interaction shows up as differently-sloped lines or "rings" for different `slope` values across the `sin(aspect)` variable.

The fourth plot does not show differently-sloped marginal "rings" across `cos(aspect)` because that interaction effect is nonsignificant.

Here we arbitrarily picked the median quantile for illustration. If we were interested in the interaction effect on the 99^{th} percentile we could use the code below to get a similar set of plots.

```
plotrad(quote(`0.99`), "99th percentile basal area")
```

8.5 Understanding Effects on Probability of Zero

Perhaps more interesting than looking at additional τ-predicted quantiles would be to see the effect of the covariates on the probability of having zero basal area. These can be found as extensions of the marginal or

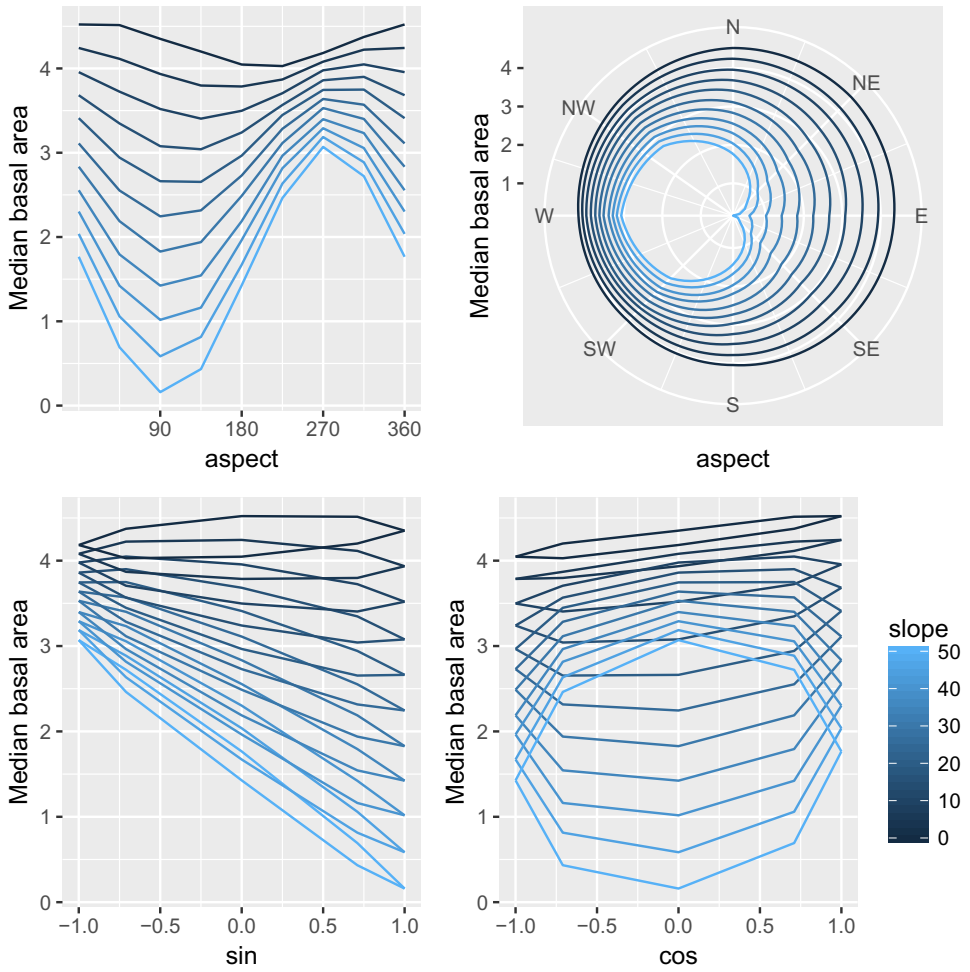

Figure 14: Marginal predicted medians over varying slope and aspect

interaction predictions from the previous section. We illustrate using the `slope` by trig-transformed `aspect` interaction, repurposing our custom function for use on the zero probabilities.

For a "quick-and-dirty" approximation, we find the zero-probabilities by interpolation over the already-summarized `pred3` array. To include error bands around our zero-probabilities, we would need to back up a step and get predicted values for each iteration of the MCMC sampler, interpolate to find the zero probabilities, *then* summarize, as laid out in the "Interpreting Quantile Regressions" section.

```
plotrad(quote(`tau0`), "Prob(0)")
```

Directionally, the plots of Figure 15 seem to tell a similar story to the median interaction plots; eastern-facing sites are more likely to have zero basal areas than similarly-sloped westward-facing sites. We interpret these cautiously though, lacking appropriate error bands to quantify our uncertainty and definitively declare the probabilities different than the sample prevalence of zeroes.

8.6 Further Model Refinement and Comparison

By comparing WAIC, we see that the multiple-regression fit `fit.qrj3` is a better fit than `fit.qrj2` of the simple-quantile regression section. Seeking an even better fit, we make several refinements to the multiple regression model.

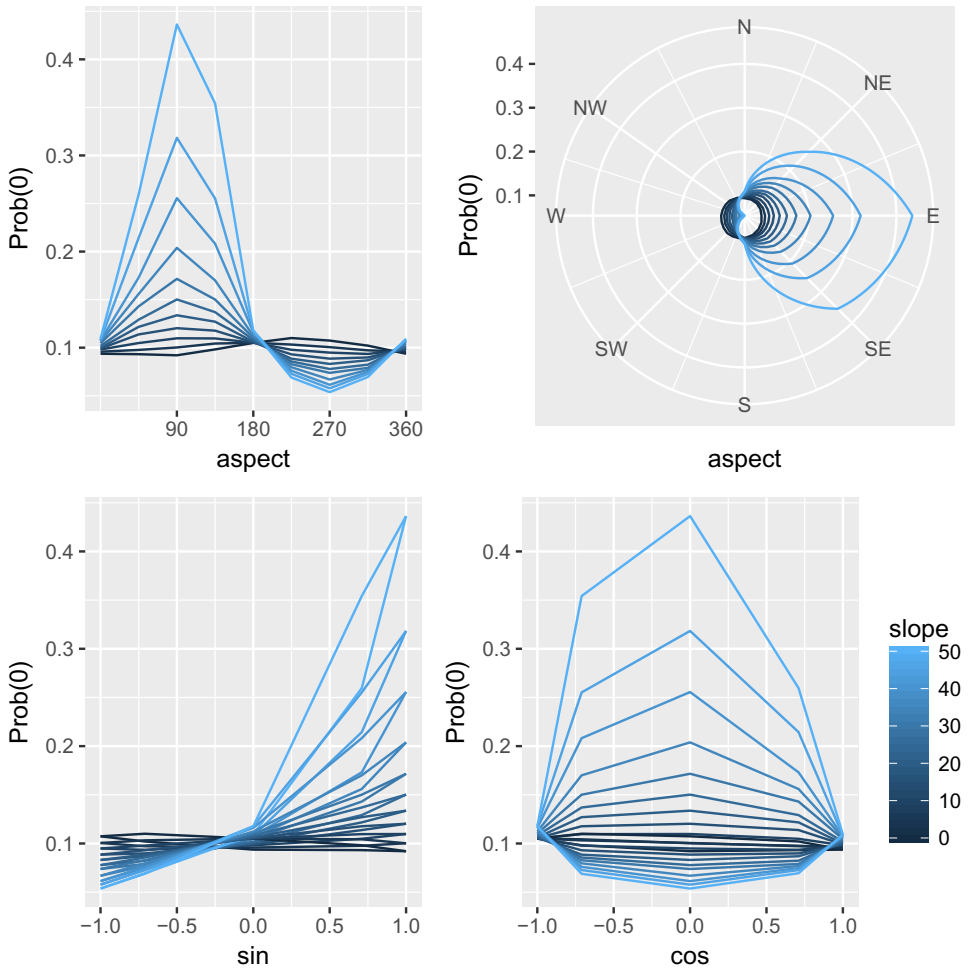

Figure 15: Marginal predicted medians over varying slope and aspect

First, we add the b-spline on elevation that we used in the single-covariate model, hoping to straighten out the quantile levels for that covariate. And second, we drop the marginal transformed `aspect` covariates from the model. Usually, dropping a non-significant main-effect from a model when the interaction is significant and retained is not advocated; however, in this application we have a scientific justification.

For related but slightly different reasons, dropping part of a radial transformation is not usually advocated; however, in this application the covariate that corresponds to North-South direction does not show a significant effect, and the inclusion of the East-West covariate could be justified under its own ecological moorings. We compare models: one that has the non-significant `cos(aspect):slope` effect retained and one that drops it from the model.

```
library(splines)
set.seed(66666)
fit.qrj4 <- qrjoint(baRedMaple ~ slope + slope:I(sin(aspect*pi/180)) + slope:I(cos(aspect*pi/180)) +
            I(aspect==0) + region + ns(elev,3),
            data=dat, cens=ifelse(dat$baRedMaple==0,2,0),
            par="RQ", fbase="logistic", nsamp = 2000, thin = 20)
```

```
summary(fit.qrj4, plot.dev=FALSE)
#> WAIC.1 = 3446.18 , WAIC.2 = 3445.82
```

```
library(splines)
set.seed(77777)
fit.qrj5 <- qrjoint(baRedMaple ~ slope + slope:I(sin(aspect*pi/180)) + I(aspect==0) +
            region + ns(elev,3), data=dat, cens=ifelse(dat$baRedMaple==0,2,0),
            par="RQ", fbase="logistic", nsamp = 2000, thin = 20)
```

```
summary(fit.qrj5, plot.dev=FALSE)
#> WAIC.1 = 3442.22 , WAIC.2 = 3442.57
```

Here we see that the WAIC is improved by the addition of the elevation b-spline and by dropping the marginal `cos` and `sin` effects. Also we see that WAIC improves slightly when `cos(aspect):slope` is also dropped from the model.

These models, fit on ~600 observations and <10 predictors, can each take 10 minutes or so to run, depending on computing resources. Models run on ~2000 observations and a similar number of predictors can take an hour to reach convergence and collect adequate samples from the MCMC chain. We see that model building and comparison for joint quantile regression is possible on moderately sized datasets and yields rich, interpretable, and distributional-free results; however, it requires some time commitment for computing and interpretation. We suggest these methods to the practitioner who is willing to invest their time to achieve such rich, distribution-free results but not to the casual user.

9 Conclusions and Final Remarks

In this chapter we have illustrated how joint quantile regression, as implemented in the `qrjoint` R package, can be used to carry out a model-based regression analysis of a zero-inflated but otherwise-positive continuous response. Joint quantile regression is able to model the continuous response distribution with few distributional assumptions, making it more broadly applicable than Tobit regression. The censoring or latent-variable construct is only appropriate for modeling excess-zero values when the same mechanisms that drive small-response values also drive zero-response values. We believe that to be reasonable in the case of the basal area case study.

We have shown how the quantile planes produced by the joint quantile regression obey the appropriate monotonicity constraints, something that cannot be said for traditional quantile regression methods. We have illustrated via a case study how to interpret the coefficient estimates obtained from a joint quantile regression. We also introduced visual diagnostics based on the probability integral transform that allow us to assess overall model fit and linearity assumptions, pointing us to areas where our design matrix could be refined. These diagnostics are not available for independently estimated quantile regressions, and therefore represent a valuable new tool for model refinement in the quantile regression context.

Additionally, by utilizing the generative nature of our joint quantile regression model, we not only adjust for censoring but also make it a prominent inferential objective. For our case study, observing zero red maple basal area can be a phenomenon of independent scientific interest. The probability of this event, measured as $\tau_0(x) = Q^{-1}(0|x)$ can only be calculated by inverting the quantile function $Q(\tau|x)$ – which necessitates obtaining non-crossing, joint estimation of the function at all quantile levels. Such estimates are hard to obtain from an ensemble of single quantile level quantile regressions.

We have approached this problem primarily from the goal of inference on regression intercept and slopes, viewed as unknown functions of the quantile level. However, we have also illustrated elementary tools for prediction, including how-tos for estimating the probability of zero when data are left-truncated at zero. If prediction were the primary goal, we could train models on a subset of data and compare observed basal areas to the predictions on the held-out data. Comparison between `qrjoint`-fit models can be made on quantile predictions by using the check-loss metric and/or on the probability of zero by maximizing the area under a receiver-operating-characteristic curve, depending on what is the focus of prediction.

References

Abrevaya, J. (2002). The effects of demographics and maternal behavior on the distribution of birth outcomes. In *Economic Applications of Quantile Regression*, pp. 247–257. Springer.

Andrieu, C. and J. Thoms (2008). A tutorial on adaptive mcmc. *Statistics and computing 18*(4), 343–373.

Banerjee, S., A. E. Gelfand, A. O. Finley, and H. Sang (2008). Gaussian predictive process models for large spatial data sets. *Journal of the Royal Statistical Society: Series B (Statistical Methodology) 70*(4), 825–848.

Bondell, H. D., B. J. Reich, and H. Wang (2010). Noncrossing quantile regression curve estimation. *Biometrika 97*(4), 825–838.

Burgette, L. F., J. P. Reiter, and M. L. Miranda (2011). Exploratory quantile regression with many covariates: an application to adverse birth outcomes. *Epidemiology 22*(6), 859–866.

Dunham, J. B., B. S. Cade, and J. W. Terrell (2002). Influences of spatial and temporal variation on fish-habitat relationships defined by regression quantiles. *Transactions of the American Fisheries Society 131*(1), 86–98.

Dunson, D. B. and J. A. Taylor (2005). Approximate Bayesian inference for quantiles. *Nonparametric Statistics 17*(3), 385–400.

Elsner, J. B., J. P. Kossin, and T. H. Jagger (2008). The increasing intensity of the strongest tropical cyclones. *Nature 455*(7209), 92–95.

Feng, Y., Y. Chen, X. He, et al. (2015). Bayesian quantile regression with approximate likelihood. *Bernoulli 21*(2), 832–850.

He, X. (1997). Quantile curves without crossing. *The American Statistician 51*(2), 186–192.

Koenker, R. and G. Bassett (1978). Regression quantiles. *Econometrica: Journal of the Econometric Society 46*(1), 33–50.

Portnoy, S. (2003). Censored regression quantiles. *Journal of the American Statistical Association 98*(464), 1001–1012.

Powell, J. L. (1986). Censored regression quantiles. *Journal of econometrics 32*(1), 143–155.

Reich, B. J., M. Fuentes, and D. B. Dunson (2011). Bayesian spatial quantile regression. *Journal of the American Statistical Association 106*(493), 6–20.

Tobin, J. (1958). Estimation of relationships for limited dependent variables. *Econometrica: journal of the Econometric Society*, 24–36.

Tokdar, S. T. (2007). Towards a faster implementation of density estimation with logistic gaussian process priors. *Journal of Computational and Graphical Statistics 16*(3), 633–655.

Tokdar, S. T. and J. B. Kadane (2012). Simultaneous linear quantile regression: a semiparametric Bayesian approach. *Bayesian Analysis 7*(1), 51–72.

Yang, Y. and S. T. Tokdar (2017). Joint estimation of quantile planes over arbitrary predictor spaces. *Journal of the American Statistical Association 112*(519), 1107–1120.