Appendix to Ordered Beta Regression: A Parsimonious, Well-Fitting Model for Continuous Data with Lower and Upper Bounds

Robert Kubinec

March 7th, 2020

Contents

1 Joint Posterior and Log-likelihood Definition

I can express the model as a log-likelihood for a given distribution of y_i as follows:

$$ll(y_i|K,\beta,\phi) = \sum_{i=1}^{N} \left\{ \begin{array}{ll} \log\ [1 - g(X'\beta - k_1)] & \text{if } y_i = 0 \\ \log\ [g(X'\beta - k_1) - g(X'\beta - k_2)] + \log \text{Beta}(g(X'\beta), \phi) & \text{if } y_i \in (0,1) \\ \log\ g(X'\beta - k_2) & \text{if } y_i = 1 \end{array} \right\} \quad (1)$$

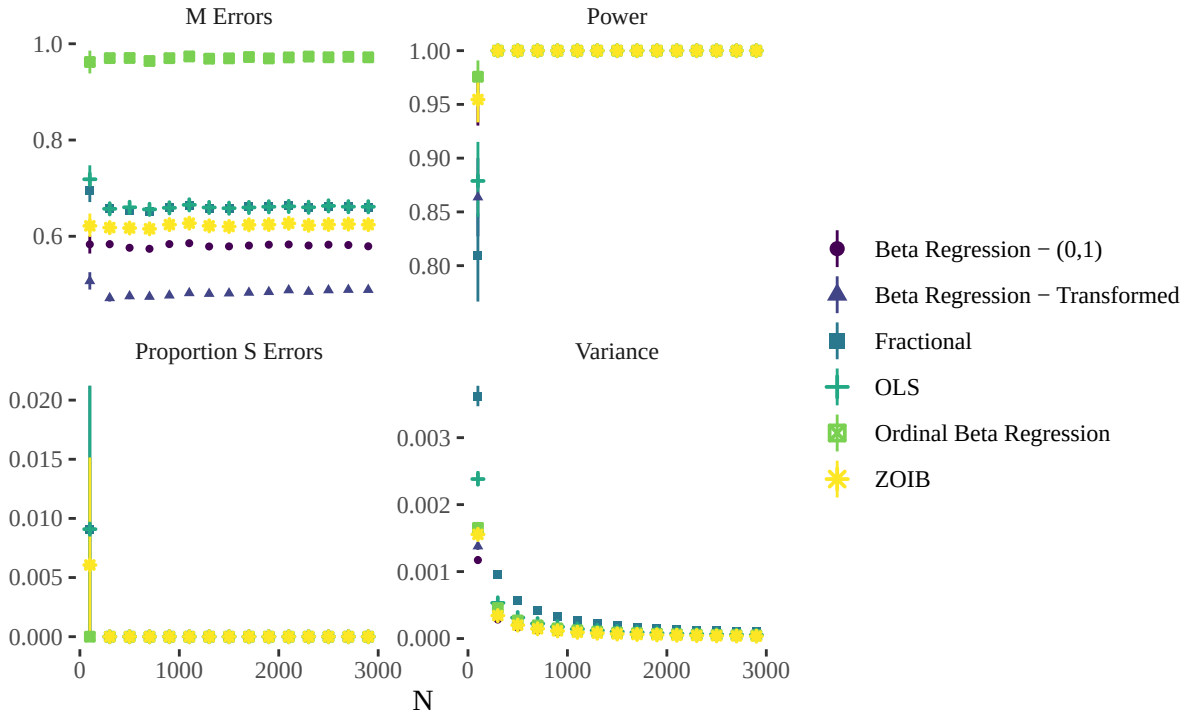
Given this likelihood, I can define a joint log posterior distribution over y given the log-likelihood function and set of parameters:

$$\log p(k_1, k_2, \beta, \phi|y) \propto \sum_{i=1}^{N} \log p(K) + \log p(\beta) + \log p(\phi) + ll(y_i|K, \beta, \phi) \tag{2}$$

where $\propto$ indicates that the posterior is calculated proportional to the normalizing constant, i.e., the denominator in Bayes' formula.

2 Simulation with Fixed Parameter Values

The fixed simulation relative to the more thorough simulation presented in the main text shows that, for this particular set of parameter draws (five covariates with a ρ_x of 0.5, ϕ of 2, k_1 of -3 and k_2 of 2), the ZOIB shows somewhat less variance than ordered beta regression but at the cost of very high M-errors. The average ZOIB coefficient magnitude isless than one-half that of ordered beta regression, which is a worrying level of bias admitted for a small reduction in variance. This simulation also shows that fractional logit regression has relatively high variance, as is also seen in the empirical example. The Beta regression on transformed values and only continuous responses show high M errors and very low variance, suggesting that again that these data-driven fixes can cause severe distortions in estimating marginal effects.



Summary statistics for each value of N calculated via bootstrapping. M Errors and S errors are magnitude of bias and incorrect sign of the estimated marginal effect. Variance refers to estimated posterior variance (uncertainty) of the marginal effect.

3 Models Without Degenerate Responses

Not only is it possible to fit the ordered beta regression mdoel to data without observations at the bounds, but it is advisable to do so if there is even a remote chance that such observations could be observed. For example, it may well be that a certain realization of the data contains observations that just so happened to not reach the bounds and are in the (0.01,0.99) interval. We could imagine this arising in a feeling thermometer/VAS scale where respondents' preferences tend to fairly clustered around the midpoint of the scale. However, a future sample of this same data could end up with observations at the bounds. It would be problematic in this case to fit only a Beta regression to the current data as the estimates would later be incomparable to estimates of future data with observations at the bounds.

While this scenario does not necessarily need to happen, it is enough of a motivation to fit the ordered beta regression model even in situations where there are no observations at the bounds (or perhaps only at one

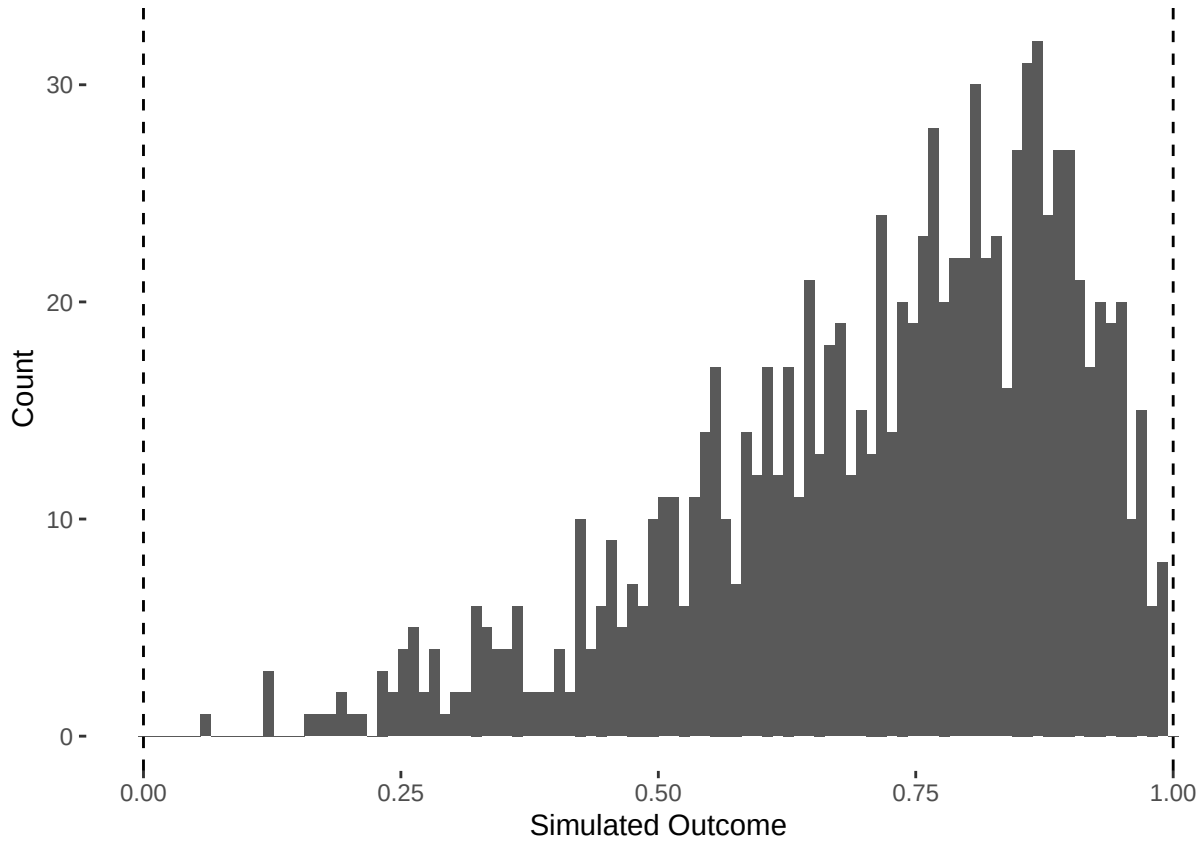
bound). The costs of doing so, both in terms of inference and computation, are quite low. Because the cutpoints were assigned a weakly informative prior, *they are identified without any data.* As a result, if a model is fit without any observations on the bounds, the cutpoints will end up in the far corners of the distribution, say at 0.001 and 0.999, but they will still exist and the posterior predictive distribution can produce them with some small probability. If future data was added to the sample incorporating observations at the bounds, the combined estimates would be interpretable and the cutpoints would adjust to handle the new data.

To demonstrate this, I simulate data from a model with widely spaced cutpoints where I remove any of the few observations that end up at the bounds:

```
N <- 1000


X <- rnorm(N,runif(1,-2,2),1)


X_beta <- -1

eta <- X*X_beta


# ancillary parameter of beta distribution

# high clustering

phi <- 70


# predictor for ordered model

mu1 <- eta

# predictor for beta regression

mu2 <- eta


# wide cutpoints on logit scale

cutpoints <- c(-8,8)
```

```r
# probabilities for three possible categories (0, proportion, 1)
low <- 1-plogis(mu2 - cutpoints[1])
middle <- plogis(mu2-cutpoints[1]) - plogis(mu2-cutpoints[2])
high <- plogis(mu2 - cutpoints[2])

# we'll assume the same eta was used to generate outcomes

out_beta <- rbeta(N,plogis(mu1) * phi, (1 - plogis(mu1)) * phi)

# now determine which one we get for each observation
outcomes <- sapply(1:N, function(i) {
  sample(1:3,size=1,prob=c(low[i],middle[i],high[i]))
})

# now combine binary (0/1) with proportion (beta)

final_out <- sapply(1:length(outcomes),function(i) {
  if(outcomes[i]==1) {
    return(0)
  } else if(outcomes[i]==2) {
    return(out_beta[i])
  } else {
    return(1)
  }
})
```

```r
# remove residual 1/Os

remove_degen <- final_out>0 & final_out<1

final_out <- final_out[remove_degen]

X <- X[remove_degen]


tibble(x=final_out) %>%

ggplot(aes(x=final_out)) +

  geom_histogram(bins=100) +

  geom_vline(xintercept = 0,linetype=2) +

  geom_vline(xintercept = 1,linetype=2) +

  theme(panel.grid=element_blank(),

        panel.background=element_blank()) +

  ylab("Count") +

  xlab("Simulated Outcome")
```



We can then model this distribution as follows:

```
to_bl <- list(N_degen=sum(final_out %in% c(0,1)),
              N_prop=sum(final_out>0 & final_out<1),
              X=1,
              outcome_prop=final_out[final_out>0 & final_out<1],
              outcome_degen=final_out[final_out %in% c(0,1)],
              covar_prop=as.matrix(X),
              covar_degen=as.matrix(X[final_out %in% c(0,1)]),
              N_pred_degen=sum(final_out %in% c(0,1)),
              N_pred_prop=sum(final_out>0 & final_out<1),
              indices_degen=array(dim=0),
              indices_prop=1:(sum(final_out>0 & final_out<1)),
```

```
                run_gen=1)


  fit_model <- ord_beta_mod$sample(data=to_bl,seed=random_seed,
                          refresh=0,
                          chains=1,cores=1,iter_sampling=1000)
```

```
## Running MCMC with 1 chain...
##
## Chain 1 finished in 6.9 seconds.
```

```
  cutpoints <- fit_model$draws("cutpoints") %>% as_draws_matrix


  print(fit_model,c("X_beta","cutpoints"))
```

```
##       variable  mean median   sd  mad    q5   q95 rhat ess_bulk ess_tail
##  X_beta[1]    -0.99  -0.99 0.01 0.01 -1.01 -0.97 1.00      564      555
##  cutpoints[1] -6.86  -6.68 1.22 1.12 -9.05 -5.18 1.00      613      474
##  cutpoints[2]  9.04   8.89 1.29 1.17  7.33 11.40 1.00     1074      567
```

We can see from the model results that our coefficient `X_beta` was estimated without bias (equal to -1). The cutpoints were estimated with a little bit of bias due to the censoring we did on the outcome variable, but are still quite close to the original values. Furthermore, they are estimated at extremes – the lower cutpoint is 0.0011 and the upper cutpoint is 0.9999. As this example indicates, there is no reason not to fit a model with no observations at the bounds. The cutpoints are still identified and the model converges without a problem. Furthermore, we can then still simulate observations at the bounds from the posterior predictive distribution:

```
ppc_dens_overlay(final_out,as_draws_matrix(fit_model$draws("regen_all")))
```