

## **Peer Review 2: Spencer Woody**

April 8, 2016

**Jennifer Starling**

Dear Spencer,

First, I hope you appreciate that this peer review is in Palatino font.

You asked me about sampling multivariate normals from Sparse Matrices, so I will address that first. I did not find a method preferable to the one you used, which was the `sparseMatrix` package. Since we also discussed a potential Rcpp implementation, I investigated whether there is a way to sample sparse multivariate normals in Rcpp.

The short answer is, not natively in Rcpp. You could pass the `sparseMatrix` package's R function to Rcpp, but that method is slow, and there is a lot of overhead. The link below explains a bit more technically.

<http://gallery.rcpp.org/articles/r-function-from-c++/>

I looked through your R code and pdf document, and do not have anything terribly momentous to add. You did fantastic work on this project, and your plots were great. Your code is clean and well-designed. I know that the speed of your code was the one thing you were not entirely satisfied with. So first, my handful of suggestions, and second, some Rcpp goodies for you.

My first suggestion was to plot the demand curves in their regular scale, instead of in log-scale. For statisticians this doesn't matter, but if your audience were economists, they may find it more intuitive to look at the curves the way they are used to seeing them.

Second, I suggest functionalizing your Gibbs sampler, and bringing back your beautiful function header blocks. These will make your code more quickly reusable.

Third, as we chatted about, I suggest working directly with the variables you are sampling in Gibbs, instead of making `.c` copies during each iteration. It may be less pretty to type `theta[k-1]` versus `theta.c`, but if you are working with a large data set, you can avoid making extra copies during each iteration. This should help with good memory management and speed.

Last, I have included some functions that I have found handy in Rcpp. Since Rcpp doesn't have some of the built-in stats functions that R does, these will save you time in recoding many of the samplers we have done in Rcpp. They include functions to sample normals, multivariate normals, and the inverse wishart. I also included a function to replicate R's `seq` function, as this is tricky but comes in handy often. See attached `.cpp` file for these. I have included them in all with export headers so that you can load them directly into R, but you can just wrap them in a namespace to make it easy to have other Rcpp functions call them. (Or you can directly import them into other Rcpp functions without a namespace. I also included a toy R file which calls the functions.

Happy coding!

Jennifer