Brief notes on how to sample from Polyagamma distribution

Yunran Chen

Here I provide brief notes on how to sample from $PG(b, z)$ distribution and how the `rpg` function in R package `BayesLogit` works.

1. Naive way: gamma approximation

For $X \sim PG(b, z)$, we have

$$X = \frac{1}{2\pi^2} \sum_{k=1}^{\infty} \frac{g_k}{(k-0.5)^2 + c^2/(4\pi^2)} \quad (1)$$

where $g_k \sim Ga(b, 1)$.

Using truncated summation (for example, $k = 1 : 200$) to approximate.

Problems:

1. Low efficiency.
2. Risk of using truncated value to approximate infinite sum.

Using accept/reject sampling could provide accurate distribution with high efficiency (high accept rate).

2. Efficient way

Polson *et al.* (2013a [2], 2013b [1]) provided hybrid sampling approach based on Devroye, alternative and saddlepoint sampling method. These three methods are all applying accept/reject sampling to sample $X \sim J^*(1, z/2)$ then get $Y \sim PG(b, z)$ from $Y = X/4$. The following illustration define the target distribution $J^*(b, z)$. The first two are based on partial sum version of accept/reject sampling (when target $f(x)$ can be written as $\sum_{n=0}^{\infty}(-1)^n a_n(x)$ and corresponding partial sum is decreasing sequence, the accept/reject decision can be based on partial sum $S_i(x)$). The goal of Devroye and alternative sampling are to find the decreasing partial sequence.

Devroye: When $b = 1$, $J^*(1, z)$ can be expressed as two kinds of sum of alternative sign sequence a_n^L (inverse-Gaussian) and a_n^R (exponential) decreasing on $(0, t]$ and $[t, \infty)$ respectively. Define a new sequence as piecewise function of these two sequences to make sure the new sequence is decreasing on $x \in (0, \infty)$. The accept/reject sampling derived from this is denoted as Devroye sampling. With $t = 2/\pi$, the accept rate is around 99% and usually only need to calculate S_1 to accept/reject, which is super efficient.

For $J^*(b, z)$ where $b \neq 1$, only a_n^R holds. A direct idea is : For $J^*(n, z)$ where n is integer, we can get $Y \sim J^*(n, z)$ from the sum of n independent samples from $J^*(1, z)$. For $J^*(b, z)$ when b is not integer, we can set $b = [b] + e$, use naive way with $K = 200$ to sample $J^*(e, z)$ then sum up. However, these methods are inefficient.

Alternative sampling is to redefine a a_n^L and a_n^R for $J^*(b, z)$ directly. Numeric calculation is needed to find the optimal separate point $t(b)$, which is a function of shape parameter b and would effect the accept rate. However, with large b, the accept rate is quite low.

Saddlepoint sampling is for relative large shape parameter b. It is based on accept/reject method (not partial sum version). It constructed envelope to find the proposal distribution g.

`rpg` in R package `BayesLogit` is hybrid method based on the above three methods.

1. For $b = 1, 2$, sampling $X_i \sim PG(1, z)$ using Devroye method , and $X = \sum_{i=1}^{b} X_i \sim PG(b, z)$.
2. For $b \in (1, 13)/\{1, 2\}$, apply alternative sampling and sample from $PG(b, z)$ directly.
3. For $b \in [13, 170]$, apply saddlepoint sampling and sample from $PG(b, z)$ directly.
4. For $b > 170$, use normal approximation Bug exists in R package. This method cannot make sure the samples we get is above 0.

3. R package

The R package is run C code from R. To install `BayesLogit`, make sure to follow the three steps. First, install the `gcc`. Second, make sure to set the right path for `gfortran`. Third, follow the instruction in `https://github.com/jwindle/BayesLogit/blob/master/INSTALL` to install.

References

[1] N. G. Polson, J. G. Scott, and J. Windle. Improved pólya-gamma sampling.

[2] N. G. Polson, J. G. Scott, and J. Windle. Bayesian inference for logistic models using pólya–gamma latent variables. *Journal of the American statistical Association*, 108(504):1339–1349, 2013.