A Study of Bootstrap Based Approximations for Posterior Distributions

Alex Zhao

Pennsylvania State University

yazhao@psu.edu

April 16, 2018

Monte Carlo Markov Chains

- MCMC methods are algorithms for sampling from a probability distribution
- By building a Markov chain where the equilibrium distribution is the desired probability distribution, a sample can be drawn simply by running the chain
- Mostly used in Bayesian statistics, generally for cases where there's a need to draw from analytically difficult posterior distributions, usually those with multi-dimensional integrals

Example: Metropolis-Hastings Algorithm

Assume we have $h(x) = c\pi(x), x \in \Omega$ as a function proportional to our posterior, and a proposal $q(x, y) = q(y|x)$ (transition kernel of irreducible Markov Chain)

1. Start with $X_0 = x_0 \in \Omega$. For $n = 0, 1, 2, \ldots$, if $X_n = x$, generate as follows:
2. Propose $y \sim q(.|x)$
3. Accept/reject proposal:
    1. $\alpha(x, y) = \begin{cases} \min\{\frac{h(y)q(x|y)}{h(x)q(y|x)}, 1\} & h(x)g(x, y) > 0 \\ 1 & o/w \end{cases}$
    2. Accept $X_{n+1} = y$ with probability $\alpha(x, y)$ or instead reject and have $X_{n+1} = x$ with probability $1 - \alpha(x, y)$

Issues with MCMC

1. Within the context of certain classes of problems, for example multinomial inverse regression (MNIR), fully Bayesian methods through Monte Carlo marginalization are prohibitively expensive (Taddy 2013)
2. Sometimes the likelihood doesn't have a good conjugate prior as in the case of the negative binomial likelihood model (Pillow and Scott, 2012)
3. Even when MCMC is feasible, sometimes there are simpler or easier ways to get estimates from the posterior distribution
    - Weighted Likelihood Bootstrap (Newton and Raftery 1994)
    - Simple parametric bootstrap (Efron 2011)

Weighted Likelihood Bootstrapping

Regular Likelihood

Given independent data $x_1, \ldots, x_n$, with each x_i having a probability density function of $f_i(x_i;\theta)$, the likelihood function we get is

$$L(\theta) = \prod_{i=1}^{n} f_i(x_i;\theta)$$

Weighted Likelihood

Given independent data $x_1, \ldots, x_n$, with each x_i having a probability density function of $f_i(x_i;\theta)$, the weighted likelihood function we get is

$$\tilde{L}(\theta) = \prod_{i=1}^{n} f_i(x_i;\theta)^{w_{n,i}}$$

Weighted Likelihood Bootstrapping, Cont'd

How are the weights determined?

- "[B]y the statistician." (Newton Raftery 1994)
- Uniform Dirichlet distribution
  1. Generate n samples from $Y_i \sim Exp(\lambda)$
  2. Create $W_{n,i} = Y_i / \bar{Y}$
  3. If need be, get $W_{n,i} \propto Y_i^{\alpha}, \alpha \neq 1$ if needed to be over or underdispersed with respect to Dirichlet
- Many other possible distributions

Raw sample of weighted likelihood bootstrap parameter estimates from repeatedly generating weight vectors and optimizing the weighted likelihood function

WLB Algorithm

1. Start with data $x_1, \ldots, x_n$ with $f_i(x_i; \theta)$
2. For j = 1 to total number of iterations N
    1. Generate weight vector $w_n = (w_{n,1}, \ldots, w_{n,n})$
        1. Generate $y_1, \ldots, y_n \sim Exp(\lambda)$
        2. Create $w_{n,i} = y_i / \bar{y}$, with α if necessary
    2. Optimize $\tilde{L}(\theta) = \prod_{i=1}^{n} f_i(x_i; \theta)^{w_{n,i}}$ to find "maximum likelihood" estimates for θ, $\tilde{\theta}^j$
3. Create an importance weight $\mu_j \propto r(\tilde{\theta}^j) = \pi(\tilde{\theta}^j) L_m(\tilde{\theta}^j) / \hat{g}(\tilde{\theta}^j)$
    - $\pi()$ is a prior on the parameter θ
    - $L_m()$ is the marginal likelihood for θ
    - $\hat{g}$ is the estimate of the joint density of $\tilde{\theta}$ with a normal kernel and Terrell's (1990) method of maximal smoothing
4. Sample from the discrete distribution determined by the weights (Sampling-Importance Resampling)

Parametric Bootstrapping

1. We have a Bayesian prior and want to compute its posterior distribution
2. Even without weighting the individual components of the complete likelihood, it's possible to use bootstrapping to achieve the same kind of estimates as MCMC
3. Sometimes offers an easier path towards calculating posterior distributions

Parametric Bootstrap Example

Assuming $y_i \sim N(a_0, \sigma^2), i = 1, \dots, n$, we want to look at the variability of $\beta = (a_0, \sigma^2)$. We get our bootstrap estimates for β^* from

$$a_0^* \sim N(\hat{a}_0, \frac{\hat{\sigma^2}}{n}), {\sigma^2}^* \sim \hat{\sigma^2} \frac{\chi^2_{n-1}}{n}$$

Bayes Parameter Expected Value

With a prior $\pi(\beta)$, Bayes theorem says given $\hat{\beta}$:

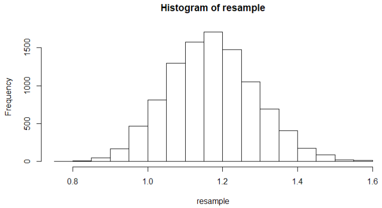
$$E\{\theta|\hat{\beta}\} = \frac{\int_\beta t(\beta)\pi(\beta)g_\beta(\hat{\beta})d\beta}{\int_\beta \pi(\beta)g_\beta(\hat{\beta})d\beta}$$

If we take $R(\beta) = \frac{g_\beta(\hat{\beta})}{g_{\hat{\beta}(\beta^*)}}$, we can replace $g_\beta(\hat{\beta})$ in the expected value with $R(\beta)g_{\hat{\beta}}(\beta^*)$, which allows us to integrate over the bootstrap density.

Parametric Bootstrap Algorithm

1. For data $y_i \sim N(a_0, \sigma^2)$, get the maximum likelihood estimates for $\hat{\beta} = (\hat{\alpha_0}, \hat{\sigma^2})$
2. For bootstrap samples j in 1 to B
   1. Draw bootstrap samples $\beta_1, \beta_2, \ldots, \beta_B$ according to $a_0^* \sim N(\hat{a}_0, \frac{\hat{\sigma^2}}{n}), \sigma^{2^*} \sim \hat{\sigma^2}\frac{\chi^2_{n-1}}{n}$
3. Calculate $R(\beta_j), \pi(\beta_j), t(\beta_j)$ for each bootstrap sample
4. Calculate $\hat{E}\{\theta|\hat{\beta}\} = \frac{\sum_{j=1}^{B} t(\beta_j)\pi(\beta_j)R(\beta_j)}{\sum_{j=1}^{B} \pi(\beta_j)R(\beta_j)}$

With our example: $n = 100, \sigma^2 = 1.25, a_0 = 1$, looking at just our bootstrap of σ^2:



General Properties of the Bootstrap to Note

- Both the weighted likelihood and Efron's parametric bootstrap approach require an importance weighing step in order to get samples from the posterior that's comparable to MCMC methods
- We are using an importance distribution, not trying to draw from the true posterior (since we don't have the true likelihood). Instead this approach is approximating the likelihood with the MLE
- Replacing the likelihood of $\theta|X$ with likelihood of $\theta|\hat{\beta}$ where $\hat{\beta}$ is a sufficient statistic
- Posterior distribution is different from MLE, hence the weighting
- Using the MLE to approximate the likelihood is adding an extra level of approximation in exchange for computational speed
- Because of the importance weighing, these bootstrap approaches can be considered the importance sampling to approximate Bayesian computing's rejection sampling

Areas of Comparison

1. Does parametric or weighted likelihood bootstrapping produce better estimates of parameters or values compared with MCMC approaches like Metropolis Hastings?
2. What is the difference in computational time and efficiency between these two categories of approaches?
3. How do these evaluations differ when it comes non-simulated data? Are there classes of problems where bootstrap can be effective when MCMC cannot, and vice versa?

Neural Models with Negative-Binomial Spiking (Pillow and Scott 2012)

- Neuroscience requires estimating neural spike responses, generally done through Poisson
- A better model is negative-binomial to account over overdispersion, but this is harder to work with analytically
- Instead of using MCMC or a Poly-Gamma distribution to sidestep this analytically intractable posterior, we instead applied parametric bootstrapping?

Multinomial Inverse Regression (Taddy 2013)

MNIR

Consider the text-sentiment contingency table with collapsed token (word) counts $x_y = \sum_{i:y_i=y} x_i$ for each $y \in Y$. Then the multinomial inverse regression model is

$$x_y \sim MN(q_y, m_y), q_{yj} = \frac{\exp(\alpha_j + y\phi_j)}{\sum_{l=1}^{p} \exp(\alpha_l + y\phi_l)}$$

Each MN is a p-dimensional multinomial distribution with size $m_y = \sum_{i:y_i=y} m_i$ and probabilities $q_y = [q_{y1}, \dots, q_{yp}]$

Generally, each coefficient ϕ_j is estimated from LaPlace priors, which is difficult to do through Monte Carlo marginalization. Could bootstrap techniques address this?