Automatic Monte Carlo Methods for Bayesian Inference

Noah Grudowski

Applied Mathematics, Illinois Institute of Technology

Bayesian Statistics

- Model parameters are random
- Prior distribution, π, reflects beliefs about the parameters
- Sampling yields a likelihood function, L
- Posterior density: product of the prior and the likelihood
- Estimates involve integrals, e.g.

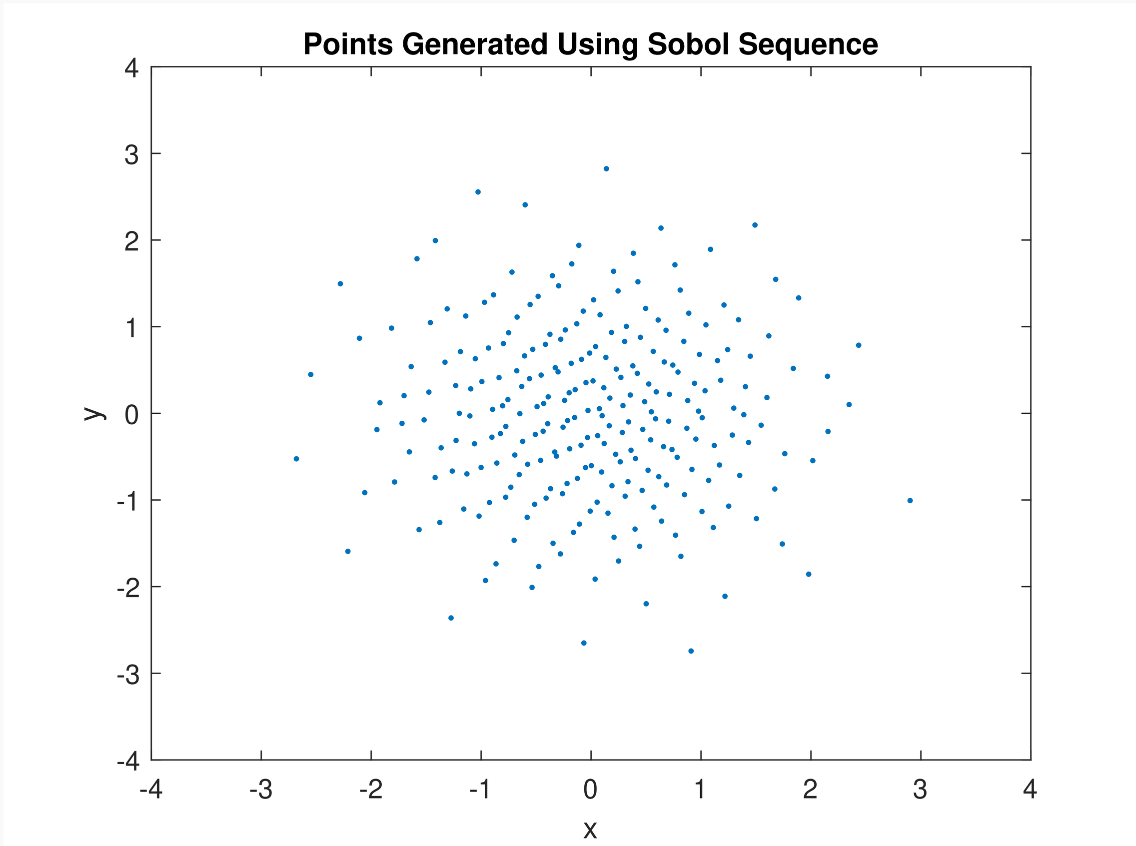
$$\hat{\beta}_j = \frac{\int_{\mathbb{R}^{d+1}} b_j L(\boldsymbol{b})\pi(\boldsymbol{b})d\boldsymbol{b}}{\int_{\mathbb{R}^{d+1}} L(\boldsymbol{b})\pi(\boldsymbol{b})d\boldsymbol{b}} =: \frac{\mu_j}{\boldsymbol{\mu}} \text{ for } j = 0, 1, 2, \ldots, d$$

Quasi-Monte Carlo Cubature

- Approximates integrals:

$$\int_{[0,1]^{d+1}} f(\boldsymbol{x})d\boldsymbol{x} \approx \frac{1}{n}\sum_{i=0}^{n-1} f(x_i)$$

  where $\{x_i\}_{i=1}^{\infty}$ is a low-discrepancy sequence
- A low-discrepancy sequence is a correlated sequence whose empirical distribution matches the target distribution better than an IID sequence



Model Problem

- Bayesian inference is applied to logistic regression.
- $t_i \sim \text{Ber}\left(\frac{\exp\left(\beta_0+\sum_{j=1}^{d}\beta_j s_{ij}\right)}{1+\exp\left(\beta_0+\sum_{j=1}^{d}\beta_j s_{ij}\right)}\right)$, for $i = 1, 2, \ldots, M$
- Standard norm prior for $\boldsymbol{\beta}$

$$\pi(\boldsymbol{b}) = \frac{\exp\left(-\frac{1}{2}\boldsymbol{b}^T\boldsymbol{b}\right)}{\sqrt{(2\pi)^{d+1}}}$$

- Posterior mean estimates of $\boldsymbol{\beta}$

$$\hat{\beta}_j = \frac{\int_{\mathbb{R}^{d+1}} b_j L(\boldsymbol{b})\pi(\boldsymbol{b})d\boldsymbol{b}}{\int_{\mathbb{R}^{d+1}} L(\boldsymbol{b})\pi(\boldsymbol{b})d\boldsymbol{b}} =: \frac{\mu_j}{\boldsymbol{\mu}}, \text{ for } j = 0, 1, 2, \ldots, d$$

A special appreciation and thank you to the IIT College of Science for providing the funding behind this research project.

Choice of Density for Integration

- $\mu_j/\boldsymbol{\mu}$ cannot be calculated analytically
- Rewrite $\boldsymbol{\mu}$ as (similarly for μ_j):

$$\boldsymbol{\mu} = \int_{\mathbb{R}^{d+1}} L(\boldsymbol{b})\pi(\boldsymbol{b})d\boldsymbol{b} = \int_{\mathbb{R}^{d+1}} \frac{L(\boldsymbol{b})\pi(\boldsymbol{b})}{\rho(\boldsymbol{b})}\rho(\boldsymbol{b})d\boldsymbol{b} = \int_{[0,1]^{d+1}} f(\boldsymbol{x})d\boldsymbol{x},$$

  where $f(\boldsymbol{x}) = f(\boldsymbol{R}(\boldsymbol{b})) = \frac{L(\boldsymbol{b})\pi(\boldsymbol{b})}{\rho(\boldsymbol{b})}$

  and $\rho(\boldsymbol{b}) = |d\boldsymbol{R}/d\boldsymbol{b}|$ for some suitable $\boldsymbol{R} : \mathbb{R}^{d+1} \to [0,1]^{d+1}$
- Now, $\mu_j/\boldsymbol{\mu}$ can be approximated by sampling from $\rho(\boldsymbol{b})$ and applying quasi-Monte Carlo cubature
- We made the following choices for $\rho(\boldsymbol{b})$:

  1) $\rho = \pi$
  2) $\rho = \rho_{\text{MLE}}$ = Gaussian approximation to the likelihood
  3) $\rho \propto \pi \cdot \rho_{\text{MLE}}$

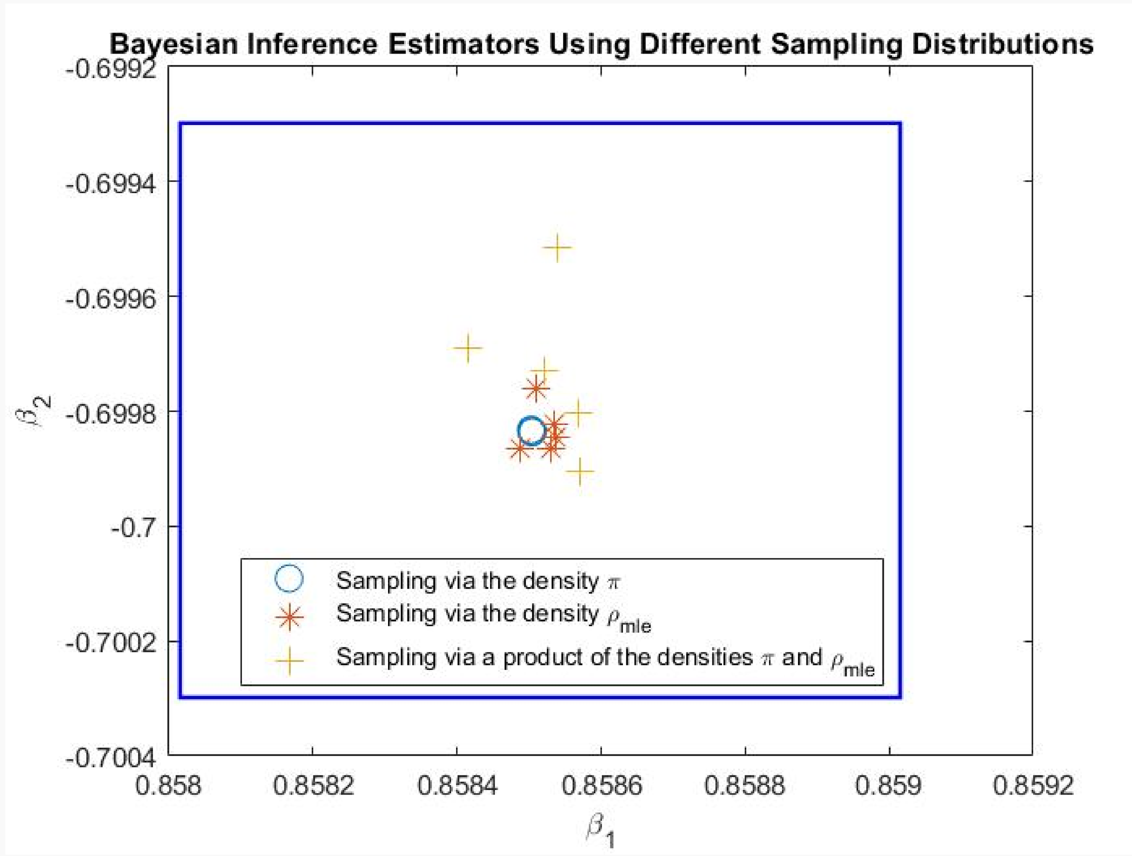
New MATLAB Function

- Developed function `BayesianInference` that automatically estimates the unknown parameters $\hat{\beta}$ using GAIL (Choi Et al, 2015)
- User inputs error tolerance, d, $\rho(\boldsymbol{b})$, and more

Uniform s-Data

```
M = 100; %number of s-Data values
s = (6+2)*rand(1,M)-2; % s-Data uniform in [-2,6]
dim=2; absTol=0.0005; n=5;
densityChoice = [true true true]; %Use all 3 densities
BayesianInference(M, s, dim, absTol,
                  densityChoice,n)
```

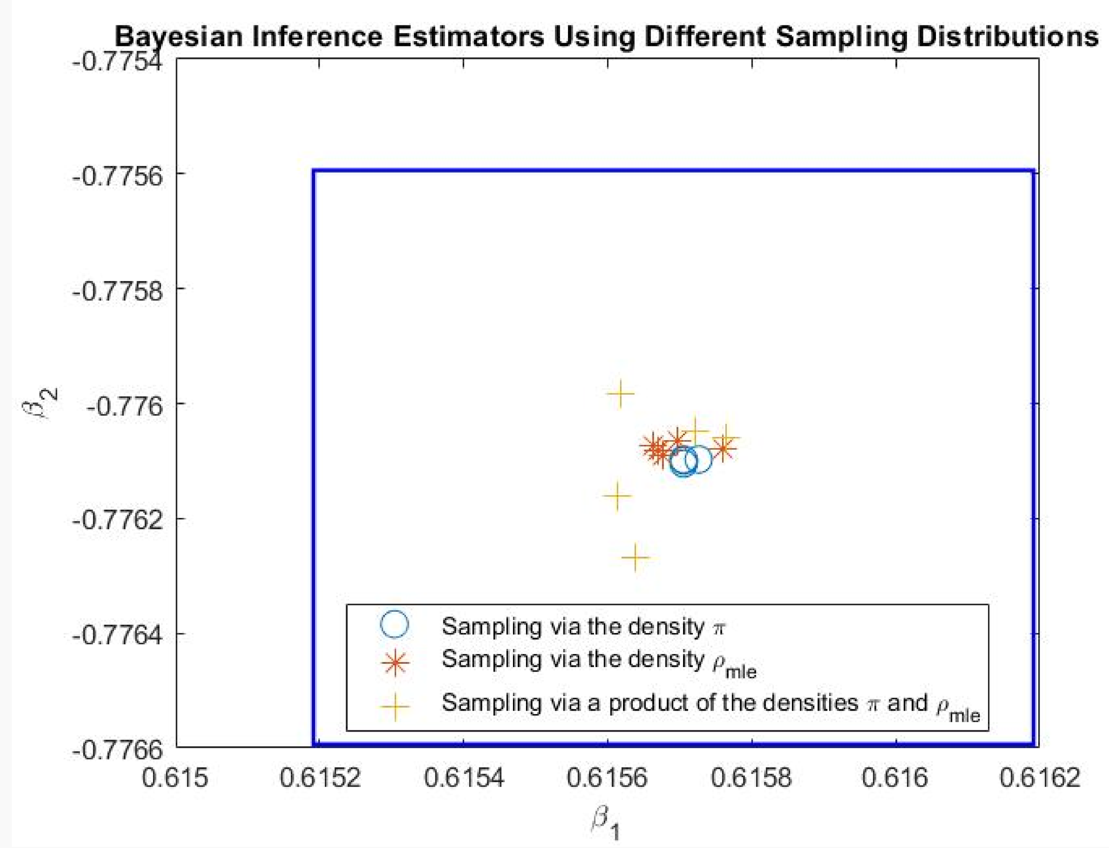
Density	Runtime (seconds)	Samples Taken
$\rho = \pi$	19.9122	655360
$\rho = \rho_{\text{MLE}}$	0.91593	24576
$\rho \propto \pi \cdot \rho_{\text{MLE}}$	0.99225	40960



Normal s-Data

```
M = 100; % number of s-Data values
s = 2.*randn(1,M)+2; % s-Data normal with mean and sd 2
dim=2; absTol=0.0005; n=5;
densityChoice = [true true true]; % Use all 3 densities
BayesianInference(M, s, dim, absTol,
                  densityChoice,n)
```

Density	Runtime (seconds)	Samples Taken
$\rho = \pi$	23.7317	589824
$\rho = \rho_{\text{MLE}}$	0.78905	20480
$\rho \propto \pi \cdot \rho_{\text{MLE}}$	1.0491	20480



Conclusions

- Each density choice was successful in approximating posterior mean estimates within the desired tolerance level
- The choice of density directly affects the time taken
- Further work should be done to improve the run time of `BayesianInference` and generalize the function to solve a greater variety of problems
- These results will be compared in detail to other methods used today, specifically Markov Chain Monte Carlo

References

- Choi, S.C.T., Ding, Y., Hickernell, F.J., Jiang, L., Jimenez Rugama, Ll.A., Tong, X., Zhang,Y., Zhou, X. *GAIL: Guaranteed Automatic Integration Library (version 2.1)*, MATLAB software, 2015.
- Hickernell, F.J., Jimenez Rugama, Ll.A., Li, D. *Contemporary Computational Mathematics — A Celebration of the 80th Birthday of Ian Sloan.* Adaptive Quasi-Monte Carlo Methods for Cubature. pages 597-619, Springer-Verlag, 2018.