MCMC

Jason Qiang Guo, New York University

Bayesian Statistics

Bayesian statistics, subjective view of probability:

- Personal belief in a statement
- Can be affected by different sources of information: personal beliefs, prior experience
- Does not rely on repeated sampling or large n assumptions (but the posterior $\approx$ MLE estimation when n is large)

Run a Bayesian Model

How does belief update?

$$p(\theta|y) \propto p(y|\theta)p(\theta)$$

$$\text{Posterior} = \text{Likelihood} \times \text{Prior}$$

- Make probabilistic assumptions for the prior of parameters ($p(\theta)$) and data (y) and specify a model
- Derive analytically or simulate numerically for the posterior ($p(\theta|y)$) distribution
- Obtain quantities of interest using the quantities summarized from the posterior distribution of parameters

Example: Binomial-Beta Model

The Golden State Warriors play 82 games during a regular NBA season. In the 2015-2016 season, they became the most winning team in NBA regular season history and won 73 games. Suppose our subjective belief that the probability for Warriors to win each game is probability π, and the outcome of a game is independent of the other, how would you estimate π after the regular season?

We have observation Y that follows binomial distribution

$$Y \sim \mathit{Binomial}(n, \pi) \text{ with } n = 82$$

The prior distribution we use here is a beta distribution since the domain of beta distribution is $[0, 1]$

Example: Binomial-Beta Model

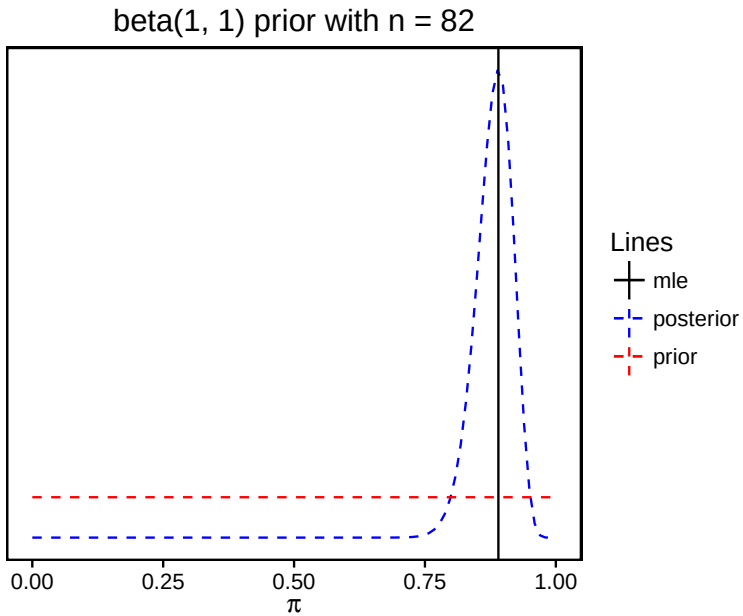
$$
\begin{aligned}
p(\pi|y) &\propto p(y|\pi)p(\pi) \\
&= binomial(n, \pi) \times beta(\alpha, \beta) \\
&= \binom{n}{y} \pi^{y}(1-\pi)^{(n-y)} \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)} \pi^{\alpha-1}(1-\pi)^{\beta-1} \\
&\propto \pi^{Y+\alpha-1}(1-\pi)^{n-y+\beta-1}
\end{aligned}
$$

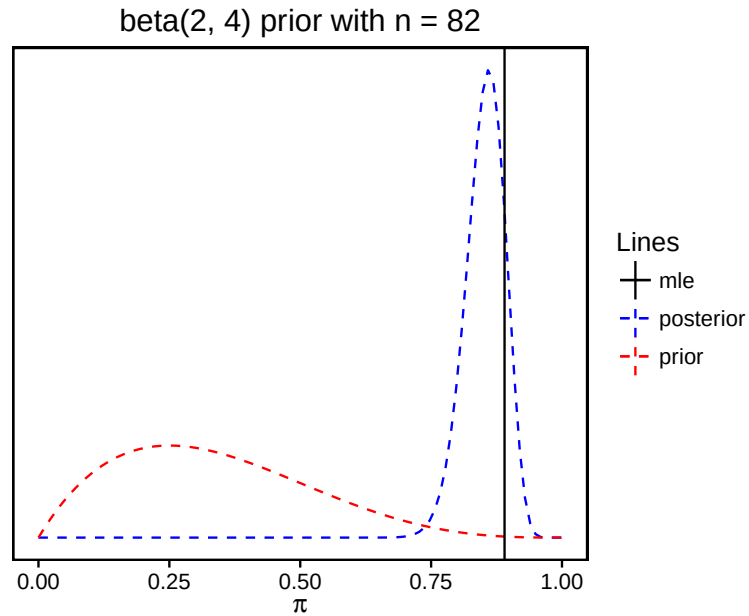
Example: Binomial-Beta Model

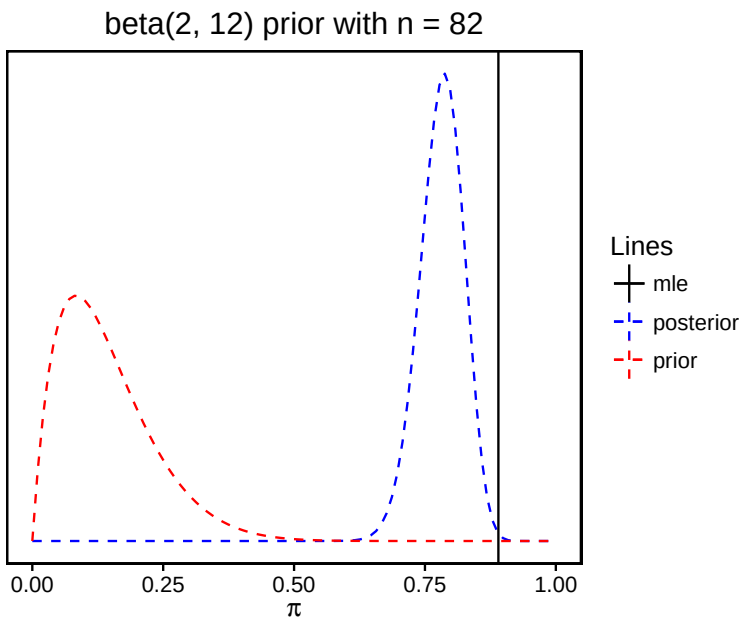
$$
\begin{aligned}
p(\pi|y) &\propto p(y|\pi)p(\pi) \\
&= binomial(n, \pi) \times beta(\alpha, \beta) \\
&= \binom{n}{y} \pi^{y}(1-\pi)^{(n-y)} \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)} \pi^{\alpha-1}(1-\pi)^{\beta-1} \\
&\propto \pi^{Y+\alpha-1}(1-\pi)^{n-y+\beta-1}
\end{aligned}
$$

The posterior distribution is a beta distribution $\text{Beta}(y+\alpha, n-Y+\beta)$, by having a beta prior we add $\alpha-1$ successes and $\beta-1$ failures to the dataset. Since we know the posterior distribution, we can get the following quantities: posterior mean, posterior SD, posterior credible intervals and highest posterior density region

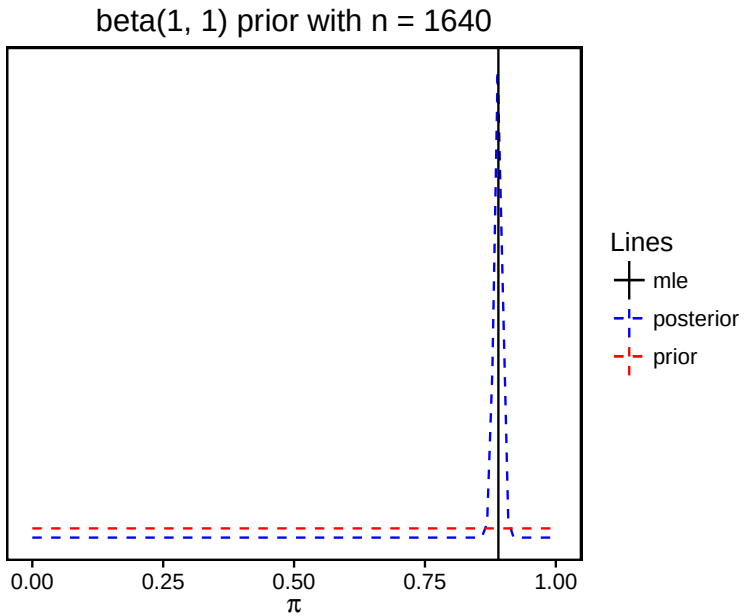
Example: Binomial-Beta Model

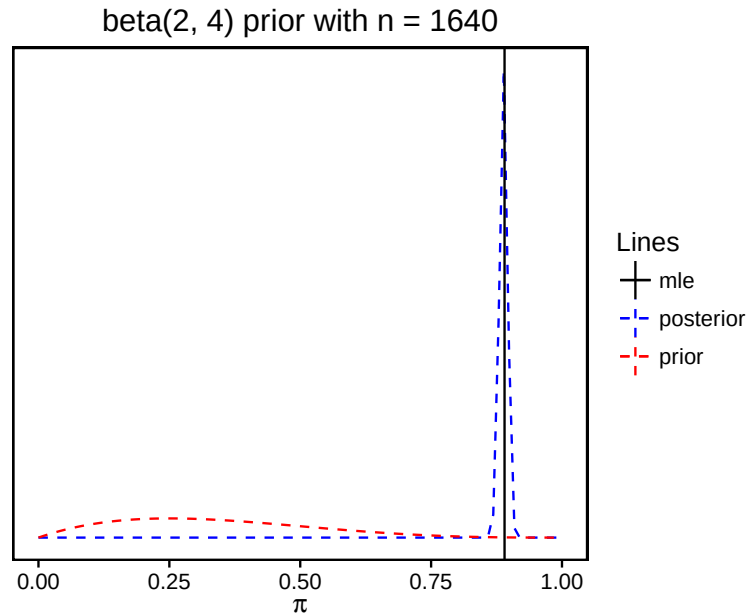


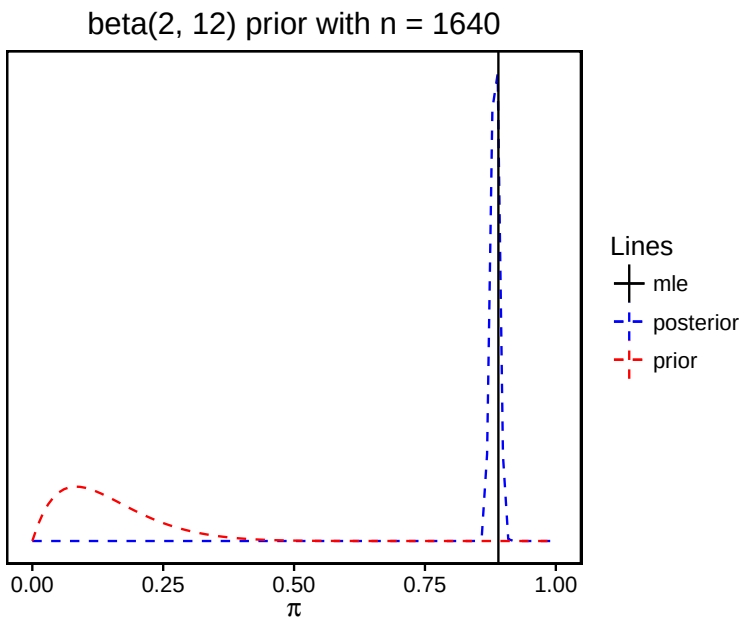




Posterior close to MLE when n increases







Conjugacy

- In the example of binomial-beta model, we can see that prior and posterior follow the same family of probability distribution. We call them conjugate distributions and the prior is called a conjugate prior for the likelihood function.
- Conjugacy is great because we can analytically derive posterior and therefore easily obtain the quantities of interest
- The limitation of conjugacy is that only a small set of problems can make use of conjugacy.

Conjugacy summary

Prior	Data/Likelihood	Posterior
$\theta \sim \text{Beta}$	$r \sim \text{Binomial}(\theta; n)$	$\theta \vert r, n \sim \text{Beta}$
$\mu \vert \sigma^2 \sim N$	$y \sim N(\mu, \sigma^2)$	$\mu \vert y, \sigma^2 \sim N$
		$\mu \vert y \sim t$
$\sigma^2 \sim \text{inverse-Gamma}$	$y \sim N(\mu, \sigma^2)$	$\sigma^2 \vert y \sim \text{inverse-Gamma}$
$\theta \sim \text{Gamma}$	$y \sim \text{Poisson}(\theta)$	$\theta \vert y \sim \text{Gamma}$
$\mathbf{\Sigma} \sim \text{inverse-Wishart}$	$\mathbf{y} \sim N(\boldsymbol{\mu}, \mathbf{\Sigma})$	$\mathbf{\Sigma} \vert \mathbf{y} \sim \text{inverse-Wishart}$
$\boldsymbol{\alpha} \sim \text{Dirichlet}$	$\mathbf{r} \sim \text{Multinomial}(\boldsymbol{\alpha}; n)$	$\boldsymbol{\alpha} \vert \mathbf{r}, n \sim \text{Dirichlet}$

Markov Chain Monte Carlo

- If we do not have conjugacy, then it is MCMC coming to rescue. Markov Chain Monte Carlo (MCMC): a class of algorithms that produce a chain of simulated draws from a distribution where each draw is dependent on the previous draw. And the chain would converge to a stationary distribution (posterior distribution).

Markov Chain Monte Carlo

- If we do not have conjugacy, then it is MCMC coming to rescue. Markov Chain Monte Carlo (MCMC): a class of algorithms that produce a chain of simulated draws from a distribution where each draw is dependent on the previous draw. And the chain would converge to a stationary distribution (posterior distribution).
- Even if the draws are slightly dependent, but the three properties (aperiodic, irreducible and positive recurrent) of Markov Chain ensure that we can use Monte Carlo simulation to calculate our quantities of interest from our draws.

Markov Chain Monte Carlo

Gibbs sampler

- When there are multiple parameters to be estimated for the posterior distribution, sometimes it is easier to consider the joint posterior distribution $p(\theta_1, \theta_2, \theta_3|y)$ instead of simulating each of $p(\theta_1|y)$, $p(\theta_2|y)$ and $p(\theta_3|y)$.
- We can use the Gibbs sampler to sample from the joint distribution if we knew the full conditional distributions for each parameter (this idea comes from $f(x, y) = f(x|y)f(y)$).
- For each parameter, the full conditional distribution is the distribution of the parameter conditional on the known information and all the other parameters: $p(\theta_j|\theta_{-j}, y)$.

Suppose $\boldsymbol{\theta}|\boldsymbol{y} = (\theta_1, \theta_2, \theta_3, \ldots, \theta_d|y)$

Algorithm 1 Gibbs Sampling

for t = 1 to T **do**

sample θ_1^{t+1} from $p(\theta_1|\theta_2^t, \theta_3^t, \ldots, \theta_d^t, y)$

sample θ_2^{t+1} from $p(\theta_2|\theta_1^t, \theta_3^t, \ldots, \theta_d^t, y)$

$\ldots$

sample θ_d^{t+1} from $p(\theta_d|\theta_1^t, \theta_2^t, \ldots, \theta_{d-1}^t, y)$

$\boldsymbol{\theta^{t+1}}|\boldsymbol{y} \leftarrow (\theta_1^{t+1}, \theta_2^{t+1}, \ldots, \theta_d^{t+1}|y)$

end for

Markov Chain Monte Carlo

Gibbs Sampler

Example: You are asked to conduct a normal Bayesian analysis assuming that $(y_1, ..., y_n) \sim N(\mu, \sigma^2)$. Assume that $\mu \sim N(\mu_0, \tau^2)$ and assume that $\sigma^2 \sim$ InverseGamma(a, b). Full conditionals:

- $p(\mu|\sigma^2, y) \propto N(\frac{\frac{\mu_0}{\tau^2} + \frac{n\bar{y}}{\sigma^2}}{\frac{1}{\tau^2} + \frac{n}{\sigma^2}}, \frac{1}{\frac{1}{\tau^2} + \frac{n}{\sigma^2}})$
  - trick here is
    $f(x) = exp(a(y-x)^2 + b(x-z)^2) \propto exp((a+b)[x - \frac{ay+bz}{a+b}]^2)$
- $p(\sigma^2|\mu, y) \propto InverseGamma(\frac{n}{2} + a, \frac{1}{2}\sum(y-\mu)^2 + b)$

Markov Chain Monte Carlo

Metropolis-Hastings Algorithm

If the posterior can not be decomposed into any known full conditional distribution, then Gibbs sampling does not work. In this case we can use Metropolis-Hastings Algorithm, of which the procedure is quite simple.

1. Write down the joint likelihood of posterior distribution $\boldsymbol{p(\theta|y)}$
2. At iteration t, draw a candidate $\boldsymbol{\theta^*}$ from a jumping distribution $\boldsymbol{J_t(\theta^*|\theta^{t-1})}$.
3. Compute an acceptance ratio:

$$r = \frac{\boldsymbol{p(\theta^*|y)/J_t(\theta^*|\theta^{t-1})}}{\boldsymbol{p(\theta^{t-1}|y)/J_t(\theta^{t-1}|\theta^*)}}$$

4. Accept $\boldsymbol{\theta^*}$ as $\boldsymbol{\theta^t}$ with probability $min\{r, 1\}$, if $\boldsymbol{\theta^*}$ is rejected, then keep $\boldsymbol{\theta^{t-1}}$
5. Repeat m times to get m draws of our parameters from the approximate posterior (assuming convergence).

Markov Chain Monte Carlo

Metropolis-Hastings Algorithm Example:

We have $(y_1, \ldots, y_n) \sim N(\boldsymbol{X}\boldsymbol{\beta}, \sigma^2)$. Assume that $\boldsymbol{\beta} \sim N(0, 100)$ and assume that $\sigma^2 \sim \mathit{InverseGamma}(1, 1)$. Under these assumptions, construct a Metropolis algorithm to compute the posterior distribution $\pi(\boldsymbol{\beta}|\boldsymbol{y})$ and $\pi(\sigma^2|\boldsymbol{y})$ using a random walk proposal: $\boldsymbol{\theta^t} \sim N(\theta^{t-1}, s)$.

The joint prior density is:

$$\begin{aligned}
\pi(\boldsymbol{\beta}, \sigma^2) &= \pi(\boldsymbol{\beta})\pi(\sigma^2) \\
&= \left(\prod_{j=1}^{k} \frac{1}{10\sqrt{2\pi}} e^{-\frac{1}{200}\beta_j^2} \right) \left((\sigma^2)^{-2} e^{-\frac{1}{\sigma^2}} \right) \\
&= (\sigma^2)^{-2} e^{-\frac{1}{\sigma^2} - \frac{1}{200}\{\boldsymbol{\beta}'\boldsymbol{\beta}\}}.
\end{aligned}$$

Therefore:

$$\begin{aligned}
\pi(\boldsymbol{\beta}, \sigma^2|\boldsymbol{y}) &\propto \pi(\boldsymbol{\beta}, \sigma^2)\pi(\boldsymbol{y}|\boldsymbol{\beta}, \sigma^2) \\
&= (\sigma^2)^{-2} e^{-\frac{1}{\sigma^2} - \frac{1}{200}\{\boldsymbol{\beta}'\boldsymbol{\beta}\}} (\sigma^2)^{-\frac{n}{2}} e^{-\frac{1}{2\sigma^2}\{(\boldsymbol{y}-\boldsymbol{X}\boldsymbol{\beta})'(\boldsymbol{y}-\boldsymbol{X}\boldsymbol{\beta})\}} \\
&= (\sigma^2)^{-\frac{n}{2}-2} e^{-\frac{1}{\sigma^2} - \frac{1}{200}\{\boldsymbol{\beta}'\boldsymbol{\beta}\} - \frac{1}{2\sigma^2}\{(\boldsymbol{y}-\boldsymbol{X}\boldsymbol{\beta})'(\boldsymbol{y}-\boldsymbol{X}\boldsymbol{\beta})\}}.
\end{aligned}$$

Markov Chain Monte Carlo

We let $e^{\tau} = \sigma^2$ and draw τ's in order to guarantee $\sigma^2 > 0$. We thus have,

$$\pi(\boldsymbol{\beta}, \tau|\boldsymbol{y}) \propto (e^{\tau})^{-\frac{n}{2}-2} e^{-\frac{1}{e^{\tau}} - \frac{1}{200}\{\boldsymbol{\beta}'\boldsymbol{\beta}\} - \frac{1}{2e^{\tau}}\{(\boldsymbol{y}-\boldsymbol{X}\boldsymbol{\beta})'(\boldsymbol{y}-\boldsymbol{X}\boldsymbol{\beta})\}}.$$

Taking the logs, this becomes

$$\ln\{\pi(\boldsymbol{\beta}, \tau|\boldsymbol{y})\} \propto \left(-\frac{n}{2} - 2\right)\tau - \frac{1}{e^{\tau}} - \frac{1}{200}\{\boldsymbol{\beta}'\boldsymbol{\beta}\} - \frac{1}{2e^{\tau}}\{(\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta})'(\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta})\}.$$

Metropolis algorithm is constructed as follows (Jackman 2009, 201f.):

1: sample candidate values $\boldsymbol{\theta}^*$ from proposal density, here $\theta^* \sim N(\theta^{(t-1)}, s)$.

2: compute acceptance ratio $\bar{r}$

$$\bar{r} \leftarrow \ln\{\pi(\boldsymbol{\theta}^*|\boldsymbol{y})\} - \ln\{\pi(\boldsymbol{\theta}^{(t-1)}|\boldsymbol{y})\}$$

to avoid numeric overflow.

3: $\alpha \leftarrow e^{\min(0,\bar{r})}$

4: sample $U \sim Unif(0, 1)$

5: **if** $U \leq \alpha$ **then**

6: $\quad \boldsymbol{\theta}^{(t)} \leftarrow \boldsymbol{\theta}^*$

7: else

8: $\quad \boldsymbol{\theta}^{(t)} \leftarrow \boldsymbol{\theta}^{(t-1)}$

9: **end if**

Markov Chain Monte Carlo

Metropolis-Hastings Algorithm

- Metropolis-Hastings Algorithm always works.
- Usually it takes time to reach convergence and in practice most people throw out a batch of early draws, which we call **burn-in**. This is to make our draws closer to the stationary distribution and less dependent on the starting point.
- But it is unclear how much we should burn-in since our draws are slightly dependent and we don't know exactly when convergence occurs.
- Also, in order to break the dependence between draws in the Markov chain, some have suggested only keeping every dth draw of the chain. This is known as **thinning**.