Bayesian Inference using Markov Chain Monte Carlo (MCMC), Part 1

William Wong

September 14, 2021

Outline

- Markov chain
- Monte Carlo
- Conjugate priors
- PyMC3
- Facebook Prophet

Probability

$$\cancel{\Pr(A) = \Pr(A|B)\,\Pr(B)}$$

Probability (con't)

- The previous equation is generally NOT TRUE.
- The law of total probability $\Pr(A) = \Pr(A|B_1)\Pr(B_1) + \Pr(A|B_2)\Pr(B_2) + \ldots$
- θ: model parameters, X: observations

$$\begin{aligned}
\Pr(\theta|X) &= \frac{\Pr(X|\theta)\Pr(\theta)}{\Pr(X)} \\
\Pr(\theta) &= \text{prior; can make our own choice!} \\
\Pr(X|\theta) &= \text{likelihood; fixed by model} \\
\Pr(\theta|X) &= \text{posterior} \\
\Pr(X) &= \text{evidence; fixed by data}
\end{aligned}$$

- The math gets messy when we rewrite $\Pr(X)$ using the law of total probability.

$$\begin{aligned}
\Pr(\theta|X) &= \frac{\Pr(X|\theta)\Pr(\theta)}{\Pr(X)} \\
&= \frac{\Pr(X|\theta)\Pr(\theta)}{\int \Pr(X|\theta)\Pr(\theta)d\theta}
\end{aligned}$$

- When we update our belief, it gets very expensive because we have to compute an integral.

Conjugate Priors

- Turn the messy integration into multipication
- Can think of conjugate priors as eigenfunctions.

MCMC

- Stata video, part 1 (00:30 to 4:30).
  - coin toss. $\theta = \Pr(\text{head})$
  - posterior $\propto$ prior $\bullet$ likelihood. Generally it is not a multiplication but note that we are using a conjugate prior!
  - $\Pr(\theta|y) = \Pr(\theta)\Pr(y|\theta) = \text{Beta}(1,1) \bullet \text{Binomial}(4,10,\theta)$ // 4 heads out of 10 tosses.
- Stata video, part 2 (00:00 to 5:01).
  1. Markov chain
  2. Monte Carlo
  3. Metropolis-Hastings algorithm

Question — why do we sometimes reject the value of θ in the Metropolis-Hastings algorithm?

PyMC3

- PyMC3 supports Bayesian inference on user-defined probabilistic models.
- To go over a Jupyter notebook.
- Example 1 — Linear regression where the response function is normally distributed, where the mean is a linear function of two predictor variables, X_1 and X_2.

$$Y \sim \mathcal{N}(\mu, \sigma^2)$$
$$\mu = \alpha + \beta_1 X_1 + \beta_2 X_2$$

Priors

$$\alpha \sim \mathcal{N}(0, 100)$$
$$\beta_i \sim \mathcal{N}(0, 100)$$
$$\sigma \sim |\mathcal{N}(0, 1)|$$

PyMC3 (con't)

- Example 2 — Coal mining disasters
  In our model, **the disaster rate** r_t **changes at some time** s.

$$D_t \sim \text{Poisson}(r_t), r_t = \begin{cases} e, & \text{if } t \leq s \\ l, & \text{if } t > s \end{cases}$$
$$s \sim \text{Uniform}(t_l, t_h)$$
$$e \sim \exp(1)$$
$$l \sim \exp(1)$$

  The parameters are defined as follows:
  - D_t: The number of disasters in year t
  - r_t: The rate parameter of the Poisson distribution of disasters in year t.
  - s: The year in which the rate parameter changes (the switchpoint).
  - e: The rate parameter before the switchpoint s.
  - l: The rate parameter after the switchpoint s.
  - t_l, t_h: The lower and upper boundaries of year t.

Facebook Prophet

- Applies Bayesian inference where we choose the prior and the likelihood.
- Finds the MAP (maximum a posteriori) estimate of the parameters via nonlinear regression.
- Uses MCMC (e.g., Gibbs sampling) to sample the posterior probability distribution.
- In principle, we should be able to develop our own model(s) inside the same framework.

Facebook Prophet (con't)

Priors

$$
\begin{aligned}
k &\sim \mathcal{N}(0, 5) \\
m &\sim \mathcal{N}(0, 5) \\
\epsilon &\sim \mathcal{N}(0, 0.5) \\
\delta &\sim \text{Double Exponential}(0, \tau) \\
\beta &\sim \mathcal{N}(0, \sigma)
\end{aligned}
$$

Logistic likelihood

$$
y \sim \mathcal{N}\left(\frac{C}{1 + \exp(-(k + A\delta)(t - (m + A\gamma))) + X\beta}, \sigma\right)
$$

Linear likelihood

$$
y \sim \mathcal{N}((k + A\delta)t + (m + A\gamma) + X\beta, \sigma)
$$

Question — where do the seasonal oscillations come from in the model?

Discussions

- How is MCMC used in Bayesian inference?
- In Part 2, we will go over the Causal Impact package from Google.