MSA101/MVE187 2018 Lecture 9

Petter Mostad

Chalmers University

October 4, 2018

Missing data

- Idea: Simulate the missing data given the parameters, and then simulate the parameters given the missing data: Gibbs sampling idea!
- Example: Censored data, for example in survival analysis: We want to learn about density $f(\cdot \mid \theta)$ from sample where $x_1, \dots, x_k$ are observed values and $c_1, \dots, c_n$ are observations that the corresponding x_i is greater than some a_i. The likelihood becomes

  $$\pi(x_1, \dots, x_k, c_1, \dots, c_n \mid \theta) = \prod_{i=1}^{k} f(x_i \mid \theta) \prod_{i=1}^{n} (1 - F(a_i \mid \theta))$$

  where $F(\cdot \mid \theta)$ is the cumulative density.
- Simulating alternatively the missing data and distribution for the parameters given *all* data may be easier than dealing with the likelihood above.
- Example 7.6 in RC: A Normal$(\theta, 1)$ model with data truncated at a.

Augmented data (or latent variables)

- Idea: Sometimes the model had been much simpler to handle if we had observed certain parameters. So: Pretend that these are missing data!
- Example 7.7 in RC: The model is the multinomial distribution

$$\text{Multinomial}(n; \frac{1}{2} + \frac{\theta}{4}, \frac{1}{4}(1-\theta), \frac{1}{4}(1-\theta), \frac{\theta}{4})$$

- The likelihood for θ has a form which makes analytical computations difficult.
- We extend the data (x_1, x_2, x_3, x_4) with a latent variable z, so that

$$(z, x_1 - z, x_2, x_3, x_4) \sim \mathcal{M}_5(n; \frac{1}{2}, \frac{\theta}{4}, \frac{1}{4}(1-\theta), \frac{1}{4}(1-\theta), \frac{\theta}{4})$$

- What is the posterior probability of θ given the extended data and a Beta prior?
- What is the conditional probability of z given θ and the actual data?
- Example 7.8 in RC: A more complex estension of Example 7.7.

Mixture models

- Assume likelihood has form

$$\pi(x_1, \ldots, x_n \mid \theta) = \prod_{i=1}^{n} \sum_{j=1}^{k} p_j f(x_i \mid \xi_j)$$

  where $\theta = (\xi_1, \ldots, \xi_k)$ are the parameters.
- Analytical calculations difficult with the sums appearing in the likelihood.
- Improved model: Add latent variables $Z = (Z_1, \ldots, Z_n)$, where $Z_i = j$ indicates the distribution x_i comes from:

$$x_i \mid z_i \sim f(x_i \mid \xi_{z_i}) \text{ and } \Pr(z_i = j) = p_j$$

- The full conditional $\pi(Z_i \mid x_i, \theta)$ can be computed as the probabilities that x_i belongs to the various distributions $f(x_i \mid \xi_j)$, when the parameters θ are given: $\Pr(Z_i = j \mid x, \theta) \propto p_j f(x_i \mid \xi_j)$.
- The full conditional $\pi(\theta \mid x_1, \ldots, x_n, Z_1, \ldots, Z_n)$ can be much easier to handle than the original likelihood: No sums occur.

Example

- Assume $x \mid \theta \sim \sum_{j=1}^{k} p_k \,\text{Normal}(x; \mu_i, \sigma^2)$, where $\theta = (\mu_1, \ldots, \mu_k)$ are unknonwn.
- Using a normal (or flat) prior on the μ_i, the posterior for each μ_i given $x_1, \ldots, x_n, z_1, \ldots, z_n$ can be found as a conjugate update using those x_i with $z_i = j$.
- The posterior for each Z_i can be computed by computing normal densities, given the current value of θ.
- Example 7.9 in RC.
- Extension: Also the weights $p = (p_1, \ldots, p_k)$ may be considered unknown, and estimated: Also here, we get a conjugate update if we use a Dirichlet prior!

Hybrid Gibbs Metropolis-Hastings methods

- The Metropolis-Hastings / Gibbs framework is very flexible: Often you can mix and match together many different proposal functions that the algorithm can switch between. As long as you can prove
  1. The target distribution fulfills the detailed balance condition for each (combination of) step(s).
  2. The Markov chain defined by the whole algorithm has a unique stationary distribution.

  you are OK.
- The objective of using hybrid methods is generally to speed up convergence.
- A good strategy may be to intersperse Gibbs sampling steps with Metropolis-Hastings specialized steps that change many variables simultaneously, to "jump" from one area with high likelihood to another.
- Another strategy may be to let the computer select randomly at each step between using a step from one of *k* possible Metropolis-Hastings algorithm for the target distribution. May be easier than figuring out which one has good convergence properties in various situations.

Example 6.5 in RC

- Example 6.5: The likelihood is a mixture:

$$\frac{1}{4}\,\mathrm{Normal}(\mu_1, 1) + \frac{3}{4}\,\mathrm{Normal}(\mu_2, 1)$$

- We simulate 400 data values using $\mu_1 = 0$, and $\mu_2 = 2.5$.
- With a prior for (μ_1, μ_2) that is uniform on $[-2, 5] \times [-2, 5]$ we get a posterior density as in Figure 6.8.
- R-code for log-likelihood function on page 128.
- R-code for simulation from posterior on page 184.
- Result very dependent on "scale" parameter. Can you think of alternative approaches?

The Laplace multivariate normal approximation

It is sometimes useful to consider the following approximation, when we have a density written

$$\pi(\theta) = C \cdot \exp(h(\theta))$$

for some known function h and unknown constant C. If $\hat{\theta}$ is the mode of the density, the second-degree Taylor approximation gives

$$h(\theta) \approx h(\hat{\theta}) + \frac{1}{2}(\theta - \hat{\theta})^t H(\hat{\theta})(\theta - \hat{\theta})$$

where $H(\theta)$ is the Hessian matrix of second derivatives. We get

$$\pi(\theta) \approx C \cdot \exp(h(\hat{\theta})) \exp\left(-\frac{1}{2}(\theta - \hat{\theta})^t((-H(\hat{\theta}))^{-1})^{-1}(\theta - \hat{\theta})\right).$$

This means that $\pi(\theta)$ might be approximated by a multivariate normal distribution with expectation $\hat{\theta}$ and covariance matrix $-H(\hat{\theta})^{-1}$. If we integrate both sides with respect to θ we get

$$C \approx \frac{1}{\exp(h(\hat{\theta}))|2\pi(-H(\hat{\theta}))^{-1}|^{1/2}}.$$

The Langevin algorithm

- Problem: It takes MCMC “too long” to “find” areas with high posterior density.
- Idea: Use not only the density value at $X^{(t)}$ but also the gradient of the density at that point to make a smart proposal Y^t.
- Concrete proposal function

$$Y^t = X^{(t)} + \frac{\sigma^2}{2} \nabla \log f(X^{(t)}) + \sigma \epsilon_t$$

- Nice to implement when formulas for the gradient can be computed analytically.
- BUT: In many cases, the convergence of the Markov chain is not improved: (One can get too easily stuck at a mode). Example 6.7 in RC.

Tempered MCMC

- Problem: The MCMC too easily gets stuck, and then does not reach the areas of high posterior density.
- Idea: Start with a period of "improved searching" before approaching the acutal MCMC formulas.
- The posterior $\exp(h(x))$ is replaced with $\exp\left(\frac{h(x)}{T}\right)$ for some positive "temperature" T: For large T this "evens out" the posterior.
- Making T monotonically sink towards 1 gives an MCMC chain that can jump more easily in the start while simulating from the correct posterior in the end.
- Making T monotonically sink towards 0 gives an MCMC chain that finds a maximum! If T sinks sufficiently slowly, one can prove it finds the *global* optimum with probability 1. *Simulated annealing*.

The slice sampler

- Idea: Do Gibbs sampling from "the area under the density curve".
- More formally, simmulate from the density

$$f(x, u) = I(0 < u < f_x(x))$$

- Works even if the density f_x is known only up to a constant.
- The challenge is to simulate x uniformly on $\{x : f_x(x) > u\}$.
- Example 7.10 in RC.
- Generalization: When $f(x) = \prod_{i=1}^{n} g_i(x)$ we can define the joint density

$$h(x, u_1, \dots, u_n) = \prod_{i=1}^{n} I\left(0 < u_i < g_i(x)\right)$$

- Simulate x uniformly on $\cap_{i=1}^{n}\{x : g_i(x) > u_i\}$.

Example: Logistic regression

(Example 7.11 in RC, but book contains errors)

- Data $(x_1, y_1), \ldots, (x_n, y_n)$; $y_i \sim$ Bernoulli$(p(x_i))$; $p(x_i) = \frac{\exp(a+bx_i)}{1+\exp(a+bx_i)}$
- Using a flat prior, simulate from posterior for (a, b) using slice sampling.
- $\pi(a, b \mid \text{data}) \propto \prod_{i=1}^n \left(\frac{\exp(a+bx_i)}{1+\exp(a+bx_i)}\right)^{y_i} \left(\frac{1}{1+\exp(a+bx_i)}\right)^{1-y_i} = \prod_{i=1}^n \frac{\exp(a+bx_i)^{y_i}}{1+\exp(a+bx_i)}$
- For $i = 1, \ldots, n$, simulate $u_i \sim$ Uniform $\left[0, \frac{\exp(a+bx_i)^{y_i}}{1+\exp(a+bx_i)}\right]$.
- Simulate (a, b) uniformly on set satisfying, for all i, $\frac{\exp(a+bx_i)^{y_i}}{1+\exp(a+bx_i)} > u_i$.
- Corresponds to $a + bx_i > \log(u_i/(1-u_i))$ for i with $y_i = 1$, and $a + bx_i < \log((1-u_i)/u_i)$ for i with $y_i = 0$.
- Extend the Gibbs sampling, simulating for a

$$a \sim \text{Uniform}\left[\max_{y_i=1}\left(\log\frac{u_i}{1-u_i} - bx_i\right), min_{y_i=0}\left(\log\frac{1-u_i}{u_i} - bx_i\right)\right]$$

Logistic regression, cont.

- For b, we need to be more careful, simulating b uniformly in the interval of numbers
  - Greater than $\left(\log\frac{u_i}{1-u_i} - a\right)/x_i$ for i with $y_i = 1$ and $x_i > 0$.
  - Smaller than $\left(\log\frac{u_i}{1-u_i} - a\right)/x_i$ for i with $y_i = 1$ and $x_i < 0$.
  - Smaller than $\left(\log\frac{1-u_i}{u_i} - a\right)/x_i$ for i with $y_i = 0$ and $x_i > 0$.
  - Greater than $\left(\log\frac{1-u_i}{u_i} - a\right)/x_i$ for i with $y_i = 0$ and $x_i < 0$.
- See R code on course home page for implementation.
- NOTE: a and b are highly correlated! Convergence improved by centering data!
- Errors in RC:
  - Confusion beween (a, b) and (α, β)
  - Second and fourth formulas on page 220 are wrong.
  - No need to use a prior for a and b to get this to work; use centering instead.