MSA101/MVE187 2021 Lecture 11
Missing data / augmented data
Hamintonian MCMC

Petter Mostad

Chalmers University

October 4, 2021

Review / overview

- We continue with ways of finding an (approximate) sample from the posterior for complicated models.
- We go back to models with no time structure.
- First subject today: Handling missing data and using augmented data.
- Second subject: Hamiltonian MCMC

Missing data / augmented data

- Assume some data values are *censored*: You don't know them exactly, only that they are (for example) above some threshold. How to deal with this?
- Example application: Survival analysis. You want to know how long people live after some event. But some people are still alive at the end of the study (or they died from other causes).
- We want to learn about density $f(\cdot \mid \theta)$ from sample where $x_1, \ldots, x_k$ are observed values and $c_1, \ldots, c_n$ are observations that the corresponding x_i is greater than some a_i. The likelihood becomes

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

  where $F(\cdot \mid \theta)$ is the cumulative distribution function.
- You may simulate from the posterior for θ using for example random walk MH.
- ALTERNATIVELY: You may *add to the model* variables representing the censored values, and simulate these together with the unknown θ. **Presentation break for R example**.

Handling missing data

- In many classical statistical methods, missing data may present a problem.
- The standard Bayesian answer in such cases: Add to the model random variables representing the unobserved values, and simulate them together with parameters and other variables of interest.
- This solves the problem in theory, but may of course sometimes be difficult in practice.

Example: Augmented data

- Example (7.7. in RC): In a genetics problem, one wants to know how close two genes are on the chromosome, measured by a parameter θ. Given n individuals, the number of individuals x_1, x_2, x_3, x_4 in each of 4 categories will be multinomially distributed accoring to

$$(x_1, x_2, x_3, x_4) \mid \theta \sim \text{Multinomial}\left(n, \frac{1}{2} + \frac{\theta}{4}, \frac{1}{4}(1-\theta), \frac{1}{4}(1-\theta), \frac{\theta}{4}\right)$$

  Given a prior on θ, how do you simulate from the posterior?
- The likelihood for θ makes necessary approximate or numerical simulation:

$$\pi(x_1, \ldots, x_4 \mid \theta) \propto_\theta \left(\frac{1}{2} + \frac{\theta}{4}\right)^{x_1} \left(\frac{1}{4}(1-\theta)\right)^{x_2} \left(\frac{1}{4}(1-\theta)\right)^{x_3} \left(\frac{\theta}{4}\right)^{x_4}.$$

- We extend the data (x_1, x_2, x_3, x_4) with a latent variable z, so that

$$(z, x_1 - z, x_2, x_3, x_4) \mid \theta \sim \text{Multinomial}\left(n; \frac{1}{2}, \frac{\theta}{4}, \frac{1}{4}(1-\theta), \frac{1}{4}(1-\theta), \frac{\theta}{4}\right)$$

- The likelihood becomes

$$\pi(z, x_1, \ldots, x_4 \mid \theta) \propto_\theta \theta^{x_1 - z + x_4}(1-\theta)^{x_2 + x_3}.$$

Example continued

- Note that, with the augmented data (z, x_1, x_2, x_3, x_4), the likelihood has the Beta family of densities as conjugate priors! Assume, for example, $\theta \sim \text{Beta}(\alpha, \beta)$.
- You can now use Gibbs sampling to sample from the distribution $\pi(z, \theta \mid x_1, \ldots, x_4)$:
  - $\theta \mid z, x_1, x_2, x_3, x_4 \sim \text{Beta}(\alpha + x_1 - z + x_4, \beta + x_2 + x_3)$.
  - $z \mid \theta, x_1, x_2, x_3, x_4 \sim \text{Binomial}\left(x_1, \frac{\frac{1}{2}}{\frac{1}{2}+\frac{\theta}{4}}\right)$.
- Exercise: Derive the Binomal distribution for z above.

Example: Augmenting the model to improve convergence

- In some models, the likelihood will have peaks that are far apart. This may cause problems for Random Walk Metropolis Hastings.
- This may occur if the model has symmetries where the likelihood is almost the same after for example a specific simultaneous change of many parameters.
- A possibility is then to add an occational proposal where such simultaneous changes are proposed.
- This may also be described as augmenting the model with an additional parameter keeping track of the changes.
- **Presentation break for R example**

Part 2. Using the target density in the proposal

- We have looked at several ideas for constructing good proposal densities. Somehow, they take into account the properties of the target density.
- Can one construct general methods that "automatically" learns about the target density and makes good proposals based on that?
- Several methods exist that do this; they have varying degrees of success with good convergence.
- We will look at one quite popular and clever method: *Hamiltonian Monte Carlo*.

Hamiltonian Monte Carlo: Idea

We are given a posterior density $\pi(q) \propto_q \exp(-U(q))$ for vectors $q = (q_1, \ldots, q_d)$. We want to find a smart proposal function that utilizes U:

- Look at $U(q)$ as some kind of "potential energy" for a particle that can move between different q's.
- If the particle moves so that it looses potential energy, it gains kinetic energy, i.e., it moves faster.
- If the particle moves in this way, it will move faster in the direction of higher density for $\pi(q)$.
- Idea: As a proposal function, randomly generate a direction and a speed for the particle to move from the current q. Then let the particle move according to dynamics above for time period s.
- Below, we use pairs (p, q) of particle momentum $p = (p_1, \ldots, p_d)$ and particle position q, moving the particle so that the total energy

$$H(p, q) = U(q) + \frac{1}{2} \sum_{i=1}^{d} \frac{p_i^2}{m}$$

is kept constant.

Metropolis Hastings using an ancillary variable

You want to generate approximate sample from $\pi(q) \propto_q \exp(-U(q))$.

- Define

$$\pi(p, q) \propto_{p,q} \exp(-H(p, q)) = \exp\left(-U(q) - g(p)\right)$$

  where $g(p)$ is symmetric: $g(-p) = g(p)$.
- Assume for all real s there is a transformation T_s on pairs (p, q) such that
  - $T_{-s}(T_s(p, q)) = (p, q)$
  - $T_s(\text{signswap}(p, q)) = \text{signswap}(T_{-s}(p, q)))$
  - $H(T_s(p, q)) = H(p, q)$

  where $\text{signswap}(p, q) = (-p, q)$.
- Given (p, q), the following two Metropolis Hastings proposal functions have acceptance probability 1:
  - Keep q, replace p simulaing from $\pi(p) \propto_p \exp(-g(p))$.
  - Propose deterministically $\text{signswap}(T_s(p, q))$ for some s.
- Thus you alternate between the two steps, and keep just the generated q's.
- Ergodicity: Make sure combining the two steps can get you from any q to any other q.

Hamiltonian dynamics

Given a function $H(p, q)$.

- A particle that has "position" q and "momentum" p at time t is said to follow Hamiltonian dynamics if, for $i = 1, \ldots, d$,

$$\frac{dq_i}{dt} = \frac{\partial H}{\partial p_i} \qquad \text{and} \qquad \frac{dp_i}{dt} = -\frac{\partial H}{\partial q_i}.$$

- After a specific time s, a particle with position q and momentum p will have position q^* and momentum p^*. This defines a mapping T_s sending the set of all pairs (p, q) to itself.
- It follows from the definition that $T_{-s}(T_s(p, q)) = (p, q)$.
- The differential equations, and thus solutions, are invariant if we change the signs of both p and s. Thus $T_s(\text{signswap}(p, q)) = \text{signswap}(T_{-s}(p, q))$.
- We have $H(T_s(p, q)) = H(p, q)$ because

$$\frac{dH}{dt} = \sum_{i=1}^{d} \left(\frac{dq_i}{dt} \frac{\partial H}{\partial q_i} + \frac{dp_i}{dt} \frac{\partial H}{\partial p_i} \right) = \sum_{i=1}^{d} \left(\frac{\partial H}{\partial p_i} \frac{\partial H}{\partial q_i} - \frac{\partial H}{\partial q_i} \frac{\partial H}{\partial p_i} \right) = 0.$$

Hamiltonian Monte Carlo

- In practice we use

$$g(p) = \frac{1}{2} \sum_{i=1}^{d} \frac{p_i^2}{m_i}.$$

  for some $m_1, \ldots, m_d$.
- We then get

$$\frac{dq_i}{dt} = \frac{p_i}{m_i} \qquad \text{and} \qquad \frac{dp_i}{dt} = -\frac{\partial U}{\partial q_i}.$$

- Then the p_i are independent, with $p_i \sim \text{Normal}(0, m_i)$.
- Summing up, the algorithm at each step starts with a current q vector. Then
  - Simulate p as above.
  - For some s solve the differential equations to compute $T_s(p, q)$.
  - Keep only the q for the next step.
- It "only" remains to see how such differential equations can be solved.

The Leapfrog algorithm: A numerical approximation of T_s

- For simplicity set all $m_i = 1$ and use vector notation: We need that

$$\frac{dq}{dt} = p \qquad \text{and} \qquad \frac{dp}{dt} = -\nabla U(q)$$

- Let $q_0, q_1, q_2 \ldots, q_n$ be the values of q along the particle path at times $0, \frac{s}{n}1, \frac{s}{n}2, \ldots, \frac{s}{n}n = s$, respectively.
- Let $p_0, p_1, p_2, \ldots, p_n$ be the values of p along the particle path at times $0, \frac{s}{n}(1-\frac{1}{2}), \frac{s}{n}(2-\frac{1}{2}), \ldots, \frac{s}{n}(n-\frac{1}{2})$, respectively.
- Approximate $\frac{dq}{dt} = p$ with

$$\frac{q_{j+1} - q_j}{s/n} = p_{j+1} \qquad j = 0, \ldots, n-1.$$

- Approximate $\frac{dp}{dt} = -\nabla U(q)$ with

$$\frac{p_{j+1} - p_j}{s/n} = -\nabla U(q_i) \qquad j = 1, \ldots, n-1.$$

  while using half stepsize for $j = 0$.
- We get the recursive equations for $j = 0, \ldots, n-1$

$$\begin{aligned} p_{j+1} &= p_j - (s/n)\nabla U(q_j) \\ q_{j+1} &= q_j + (s/n)p_{j+1} \end{aligned}$$

Hamiltonian MCMC: Summary

- Note: n computations of the gradient ∇U must be done: Possible? Time consuming?
- Note: As this is an approximation, we only have that $H(p^*, q^*) \approx H(p, q)$. But this is no problem, as we can compute and use the standard acceptance probability for Metropolis Hastings proposals.
- Note: You must still check that the Markov chain is Ergodic: In practice, that the algorithm can reach any q from any q.
- Can give great fast convergence in the cases where the gradient of the logged density is easily available and computable.
- For more information see for example Neal (2011) "MCMC Using Hamiltonian Dynamics".
- **Presentation break for computations in R**