36-463/663: Hierarchical Linear Models

Taste of MCMC / Bayes for 3 or more "levels"

Brian Junker
132E Baker Hall
brian@stat.cmu.edu

Outline

- Practical Bayes
- Mastery Learning Example
- A brief taste of JAGS and RUBE
- Hierarchical Form of Bayesian Models
- Extending the "Levels" and the "Slogan"
- Mastery Learning – Distribution of "mastery"
- (to be continued…)

Practical Bayesian Statistics

- (posterior) $\propto$ (likelihood) $\times$ (prior)
- We typically want quantities like
  - point estimate: Posterior mean, median, mode
  - uncertainty: SE, IQR, or other measure of 'spread'
  - credible interval (CI)
    - ($\hat{\theta}$ - 2SE, $\hat{\theta}$ + 2SE)
    - ($\theta_{0.025}$, $\theta_{0.975}$)
  - Other aspects of the "shape" of the posterior distribution
- *Aside:* If (prior) $\propto$ 1, then
  - (posterior) $\propto$ (likelihood)
  - posterior mode = mle, posterior SE = $1/I(\theta)^{1/2}$, etc.

Obtaining posterior point estimates, credible intervals

- Easy if we recognize the posterior distribution and we have formulae for means, variances, etc.
- Whether or not we have such formulae, we can get similar information by simulating from the posterior distribution.
- Key idea: $E[g(\theta)|\text{data}] = \int g(\theta) f(\theta|\text{data}) d\theta$

$$\approx \frac{1}{M} \sum_{m=1}^{M} g(\theta_m)$$

where $\theta_1, \theta_2, \ldots, \theta_M$ is a sample from f(θ|data).

Example: Mastery Learning

- Some computer-based tutoring systems declare that you have "mastered" a skill if you can perform it successfully r times.
- The number of times x that you erroneously perform the skill before the r^{th} success is a measure of how likely you are to perform the skill correctly (how well you know the skill).
- The distribution for the number of failures x before the r^{th} success is the negative binomial distribution.

Negative Binomial Distribution

- Let X = x, the number of failures before the r^{th} success. The negative binomial distribution is

$$f(x|p,r) = \binom{x+r-1}{x} p^r (1-p)^x$$

- The conjugate prior distribution is the beta distribution

$$f(p|\alpha,\beta) = \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)} p^{\alpha-1}(1-p)^{\beta-1}$$

Posterior inference for negative binomial and beta prior

- (posterior) $\propto$ (likelihood)$\times$(prior):

$$f(p|X=x) \propto p^r(1-p)^x \times p^{\alpha-1}(1-p)^{\beta-1}$$

- Since this is Beta(p|α+r,β+x),

$$\begin{aligned} E[p|X=x] &= \frac{\alpha+r}{\alpha+r+\beta+x} \\ Var(p|X=x) &= \frac{(\alpha+r)(\beta+x)}{(\alpha+r+\beta+x)^2(\alpha+r+\beta+x+1)} \end{aligned}$$

- and this lets us compute point estimate and approximate 95% CI's.

Some specific cases…

- Let α = 1, β = 1 (so the prior is Unif(0,1)), and suppose we declare mastery for 3 successes
- If X = 4, then

$$\begin{aligned} E[p|X=x] &= \frac{\alpha+r}{\alpha+r+\beta+x} = \frac{1+3}{1+3+1+4} = \frac{4}{9} \\ Var(p|X=x) &= \frac{(1+3)(1+4)}{(1+3+1+4)^2(1+3+1+4+1)} = \frac{20}{810} \end{aligned}$$

  Approx 95% CI: (0.12, 0.76)

- If X = 10, then

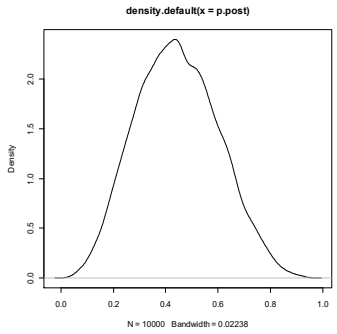
$$\begin{aligned} E[p|X=x] &= \frac{\alpha+r}{\alpha+r+\beta+x} = \frac{1+3}{1+3+1+10} = \frac{4}{15} \\ Var(p|X=x) &= \frac{(1+3)(1+10)}{(1+3+1+10)^2(1+3+1+10+1)} = \frac{44}{3600} \end{aligned}$$

  Approx 95% CI: (0.05, 0.49)

We can get similar answers with simulation

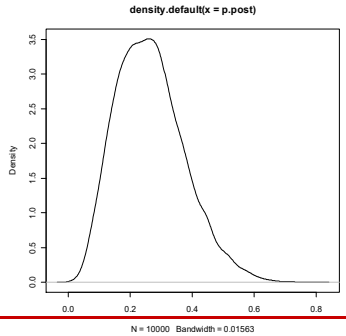
- When X = 4, r = 3, α + r = 4, β + x = 5

```
> p.post <- rbeta(10000,4,5)
> mean(p.post)
[1] 0.4444919
> var(p.post)
[1] 0.02461883
> quantile(p.post,c(0.025,0.975))
     2.5%     97.5%
0.1605145 0.7572414
> plot(density(p.post))
```



- When X = 10, r = 3, α + r = 4, β + x = 11

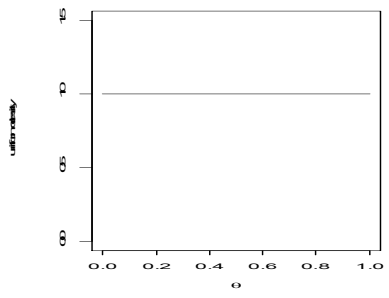
```
> p.post <- rbeta(10000,4,11)
> mean(p.post)
[1] 0.2651332
> var(p.post)
[1] 0.01201078
> quantile(p.post,c(0.025,0.975))
     2.5%     97.5%
0.0846486 0.5099205
> plot(density(p.post))
```



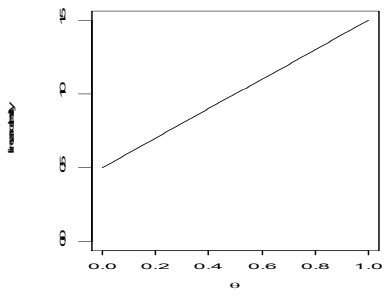
Using a non-conjugate prior distribution

- Instead of a uniform prior, suppose psychological theory says the prior density should increase linearly from 0.5 to 1.5 as θ moves from 0 to 1?

$$f(p) = 1,\ 0 \le p \le 1$$



$$f(p) = p + 0.5,\ 0 \le p \le 1$$



Posterior inference for negative binomial and linear prior

- (posterior) $\propto$ (likelihood)$\times$(prior):

$$f(p|X = x) \propto p^r(1-p)^x \times (p + 0.5)$$

- r = 3, x = 4:

$$f(p|X = x) \propto p^3(1-p)^4(p + 0.5) \text{ ??}$$

- r = 3, x = 10:

$$f(p|X = x) \propto p^3(1-p)^{10}(p + 0.5) \text{ ??}$$

What to do with $p^r(1-p)^x(p + 0.5)$?

- No formulae for means, variances!
- No nice function in R for simulation!
- There are many simulation methods that we can build "from scratch"
  - B. Ripley (1981) *Stochastic Simulation*, Wiley.
  - L. Devroye (1986) *Nonuniform random variate generation*, Springer.
- We will focus on just one method:
  - Markov Chain Monte Carlo (MCMC)
  - Programs like WinBUGS and JAGS automate the simulation/sampling method for us.

We will use this to give a taste of JAGS… Need likelihood and prior:

- Likelihood is easy:
  `x ~ dnegbin(p,r)`
- Prior for p is a little tricky:
  p ~ f(p) = p + 0.5 not a standard density!
- But we can use a trick:
  the *inverse probability transform*

Aside: the Inverse Probability Transform

- *If U ~ Unif(0,1) and F(z) is the CDF of a continuous random variable, then Z = $F^{-1}(U)$ will have F(z) as its CDF* (exercise!!).
- The prior for *p* in our model has pdf *p + 1/2*, so its CDF is *F(p)=(p^2 + p)/2*.
- *$F^{-1}(u) = (2u + 1/4)^{1/2} – 1/2$*
- So if *U* ~ Unif(0,1), then
  $$P = (2U + 1/4)^{1/2} – 1/2$$
  has density *f(p) = p + 1/2*.

JAGS: Need to specify likelihood and prior

- Likelihood is easy:

```
x ~ dnegbin(p,r)
```

- Prior for p is:

```
p <- (u + 1/4)^{1/2} + 1/2
```

where

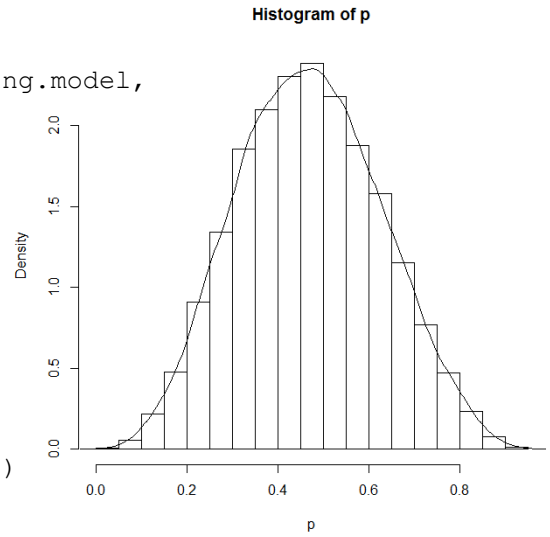
```
u ~ dunif(0,1)
```


The JAGS model

```
mastery.learning.model <- "model {

  # specify the number of successes needed for mastery
  r <- 3

  # specify likelihood
  x ~ dnegbin(p,r)

  # specify prior (using inverse probability xform)
  p <- sqrt(2*u + 0.25) - 0.5
  u ~ dunif(0,1)

}"
```

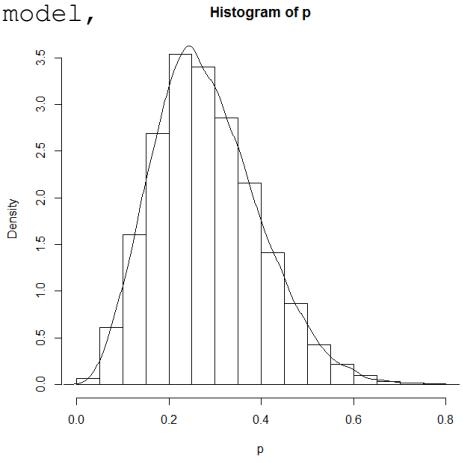
Estimate p when student has 4 failures before 3 successes

```
> library(R2jags)
> library(rube)
> mastery.data <- list(x=4)
> rube.fit <-  rube(mastery.learning.model,
+   mastery.data, mastery.init,
+   parameters.to.save=c("p"),
+   n.iter=20000,n.thin=3)
> # generates about 10,000 samples
> p <- rube.fit$sims.list$p
> hist(p,prob=T)
> lines(density(p))
> mean(p) + c(-2,0,2)*sd(p)
[1] 0.1549460 0.4685989 0.7822518
> quantile(p,c(0.025, 0.50, 0.975))
 2.5%       50%     97.5%
0.1777994 0.4661076 0.7801111
```



Estimate p when student has 10 failures before 3 successes

```
> # library(R2jags)
> # library(rube)
> mastery.data <- list(x=10)
> rube.fit <-  rube(mastery.learning.model,
+   mastery.data, mastery.init,
+   parameters.to.save=c("p"),
+   n.iter=20000,n.thin=3)
> # generates about 10,000 samples
> p <- rube.fit$sims.list$p
> hist(p,prob=T)
> lines(density(p))
> mean(p) + c(-2,0,2)*sd(p)
[1] 0.05594477 0.28247419 0.50900361
> quantile(p,c(0.025, 0.50, 0.975))
 2.5%       50%     97.5%
0.09085802 0.27119223 0.52657339
```



Hierarchical Form of Bayesian Models

- In the past few lectures, we've seen many models that involve a likelihood and a prior
- A somewhat more compact notation is used to describe the models, and we will start using it now
  - Level 1: the likelihood (the distribution of the data)
  - Level 2: the prior (the distribution of the parameter(s))
- Eventually there will be other levels as well!
- Examples on the following pages…

Hierarchical Beta-Binomial model

- Likelihood is binomial:

$$f(x|n,p) = \binom{n}{x} p^x (1-p)^{n-x}$$

- Level 1: x ~ Binom(x|n,p)
- Prior is beta distribution:

$$f(p|\alpha,\beta) = \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)} p^{\alpha-1}(1-p)^{\beta-1}$$

- Level 2: p ~ Beta(p|α,β)
- (posterior) $\propto$ (likelihood)$\times$(prior)
- (posterior) $\propto$ (level 1)$\times$(level 2) = Beta(p|α+x,β+n-x)

Hierarchical Gamma-Exponential Model

- Likelihood is Exponential:

$$f(x_1, \ldots, x_n) = \prod_{i=1}^{n} \lambda e^{-\lambda x_i}$$

- Prior is Gamma:

$$f(\lambda|\alpha, \beta) = \frac{\beta^\alpha}{\Gamma(\alpha)} \lambda^{\alpha-1} e^{-\lambda\beta}$$

- (posterior) $\propto$ (likelihood)$\times$(prior)

- Level 1:

$$x_i \overset{iid}{\sim} Expon(x|\lambda), \quad i = 1, \ldots, n$$

- Level 2:

$$\lambda \sim Gamma(\lambda|\alpha, \beta)$$

- (posterior) $\propto$ (level 1)$\times$(level 2) = Gamma(λ|α+n,β + n $\bar{x}$)

Hierarchical Normal-Normal model

- Likelihood (for mean) is normal:

$$f(x_1, \ldots, x_n|\mu) = \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2\sigma^2}(x_i-\mu)^2}$$

- Prior is normal (for mean):

$$f(\mu) = \frac{1}{\sqrt{2\pi}\tau_0} e^{-\frac{1}{2\tau_0^2}(\mu-\mu_0)^2}$$

- (posterior) $\propto$ (likelihood)$\times$(prior)

- Level 1:

$$x_1, x_2, \ldots, x_n \overset{iid}{\sim} N(\mu, \sigma^2)$$

- Level 2:

$$\mu \sim N(\mu_0, \tau_0^2)$$

- (posterior)$\propto$ (level 1)$\times$(level 2)

$$\mu \sim N(\mu_n, \tau_n^2)$$

Hierarchical Beta-Negative Binomial Model

- Likelihood is Negative-Binomial:

$$f(x|p,r) = \binom{x+r-1}{x} p^r (1-p)^x$$

- Prior is Gamma:

$$f(p|\alpha,\beta) = \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)} p^{\alpha-1}(1-p)^{\beta-1}$$

- (posterior)$\propto$ (likelihood)$\times$(prior)

- Level 1:

$$x \sim NB(x|r,p)$$

- Level 2:

$$p \sim Beta(p|\alpha,\beta)$$

- (posterior)$\propto$ (level 1)$\times$(level 2)

$$p \sim Beta(p|\alpha+r,\beta+x)$$

Hierarchical Linear-Negative Binomial Model

- Likelihood is Negative-Binomial:

$$f(x|p,r) = \binom{x+r-1}{x} p^r (1-p)^x$$

- Prior is Linear Density:

$$f(p) = p + 0.5,\ 0 \le p \le 1$$

- (posterior)$\propto$ (likelihood)$\times$(prior)

- Level 1:

$$x \sim NB(x|r,p)$$

- Level 2:

$$p \sim f(p) = p + 0.5$$

- (posterior)$\propto$ (level 1)$\times$(level 2)

$$f(p|data) \propto p^r (1-p)^x (p+0.5)$$

The “levels” and the “slogan”

- The “levels” and the “slogan” are based on the idea that

$$f(x, \theta) = f(x|\theta) f(\theta)$$

and so

$$\begin{aligned} f(\theta|x) &= \frac{f(x, \theta)}{f(x)} \\ &\propto f(x|\theta) f(\theta) \\ &= \text{(likelihood)} \times \text{(prior)} \\ &= \text{(level 1)} \times \text{(level 2)} \end{aligned}$$

Extending the “levels” and the “slogan”

- Now suppose we have two parameters:

$$\begin{aligned} f(x, \theta_1, \theta_2) &= f(x|\theta_1, \theta_2) f(\theta_1, \theta_2) \\ &= f(x|\theta_1, \theta_2) f(\theta_1|\theta_2) f(\theta_2) \end{aligned}$$

- This means we can write

$$\begin{aligned} f(\theta_1, \theta_2|x) &= \frac{f(x, \theta_1, \theta_2)}{f(x)} \\ &\propto f(x|\theta_1, \theta_2) f(\theta_1|\theta_2) f(\theta_2) \\ &= \text{(likelihood)} \times \text{(prior)} \times \text{(“hyper-prior”)} \\ &= \text{(level 1)} \times \text{(level 2)} \times \text{(level 3)} \end{aligned}$$

- This idea can be extended with more parameters and more levels…

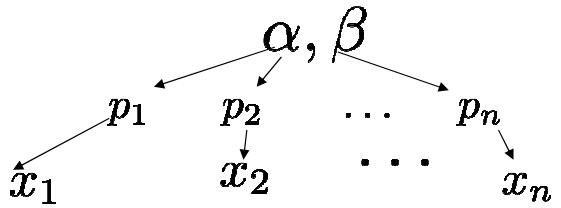
Mastery Learning: Distribution of Masteries

- Last time, we considered one student at a time according to
  - Level 1: $x \sim NB(x|r,p)$
  - Level 2: $p \sim Beta(p|\alpha,\beta)$
- Now, suppose we have a sample of n students, and the number of failures before the r^{th} success for each student is $x_1, x_2, \ldots, x_n$.
  - *We want to know the distribution of p, the probability of success, in the population: i.e. we want to estimate α & β !*
- We will model this as
  - Level 1: $x_i \sim NB(x|r,p_i)$, i=1, …, n
  - Level 2: $p_i \sim Beta(p|\alpha,\beta)$, i = 1, …, n
  - Level 3: $\alpha \sim Gamma(\alpha|a_1,b_1)$, $\beta \sim Gamma(\beta|a_2,b_2)$

Mastery Learning: Distribution of Masteries

- *We want to know the distribution of p, the probability of success, in the population: i.e. we want to estimate α & β !*
  - Level 1: $x_i \sim NB(x|r,p_i)$, i=1, …, n
  - Level 2: $p_i \sim Beta(p|\alpha,\beta)$, i = 1, …, n
  - Level 3: $\alpha \sim Gamma(\alpha|a_1,b_1)$, $\beta \sim Gamma(\alpha|a_2,b_2)$

n+2 parameters!



α,β for population of students

p_i = student prob of right

x_i = errors before r rights

Mastery Learning: Distribution of Masteries

- ***Level 1***: If x_1, x_2, … x_n are the failure counts then

$$f(x_1,\ldots,x_n|p_1,\ldots,p_n,r)=\prod_{i=1}^{n}\binom{r+x_i-1}{x_i}p_i^r(1-p_i)^{x_i}$$

- ***Level 2***: If p_1, p_2, … p_n are the success probabilties,

$$f(p_1,\ldots,p_n|\alpha,\beta)=\prod_{i=1}^{n}\frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}p_i^{\alpha-1}(1-p_i)^{\beta-1}$$

- ***Level 3***:

$$\begin{aligned} f(\alpha|a_1,b_1) &= \frac{b_1^{a_1}}{\Gamma(a_1)}\alpha^{a_1-1}e^{-\alpha b_1} \\ f(\beta|a_2,b_2) &= \frac{b_2^{a_2}}{\Gamma(a_2)}\beta^{a_2-1}e^{-\beta b_2} \end{aligned}$$

Distribution of Mastery probabilities…

- Applying the "slogan" for 3 levels:

$$\begin{aligned} & f(p_1,\ldots,p_n,\alpha,\beta|\text{data}) \\ & \propto \underbrace{f(x_1,\ldots,x_n|p_1,\ldots,p_n,r)}_{\text{(level 1)}} \times \underbrace{f(p_1,\ldots,p_n|\alpha,\beta)}_{\text{(level 2)}} \times \underbrace{f(\alpha|a_1,b_1)f(\beta|a_2,b_2)}_{\text{(level 3)}} \\ & \propto \prod_{i=1}^{n}\binom{r+x_i-1}{x_i}p_i^r(1-p_i)^{x_i} \\ & \quad \times \prod_{i=1}^{n}\frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}p_i^{\alpha-1}(1-p_i)^{\beta-1} \\ & \quad \times \frac{b_1^{a_1}}{\Gamma(a_1)}\alpha^{a_1-1}e^{-\alpha b_1}\frac{b_2^{a_2}}{\Gamma(a_2)}\beta^{a_2-1}e^{-\beta b_2} \end{aligned}$$

We can drop these since they only depend on data

Can't drop these since we want to est α & β…

We need to choose a_1, b_1, a_2, b_2. Then we can drop these

Distribution of Mastery probabilities…

- If we take a_1=1, b_1=1, a_2=1, b_2=1, and drop the constants we can drop, we get

$$f(p_1, \ldots, p_n, \alpha, \beta|\text{data}) \propto \prod_{i=1}^{n} p_i^r (1-p_i)^{x_i} \times \prod_{i=1}^{n} \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)} p_i^{\alpha-1}(1-p_i)^{\beta-1} \times e^{-\alpha} e^{-\beta}$$

- Suppose our data consist of the n=5 values

  $x_1 = 4$, $x_2 = 10$, $x_3=5$, $x_4 = 7$, $x_5 = 3$

  (number of failures before r=3 successes)
- *Problem*: We need a way to sample from the 5+2=7 parameters p_1, …, p_5, α, β!

Solution: Markov-Chain Monte Carlo (MCMC)

- MCMC is very useful for multivariate distributions, e.g. f(θ_1,θ_2, …,θ_K)
- Instead of dreaming up a way to make a draw (simulation) of all K variables at once MCMC takes draws one at a time
- We "pay" for this by not getting independent draws. The draws are the states of a *Markov Chain.*
- The draws will not be "exactly right" right away; the Markov chain has to "burn in" to a stationary distribution; the draws after the "burn-in" are what we want!

 For theory, see Chib & Greenberg, *American Statistician*, 1995, pp. 327-335

Summary

- Practical Bayes
- Mastery Learning Example
- A brief taste of JAGS and RUBE
- Hierarchical Form of Bayesian Models
- Extending the “Levels” and the “Slogan”
- Mastery Learning – Distribution of “mastery”
- (to be continued…)