CS 5135/6035 Learning Probabilistic Models

Lecture 23: Hierarchical Modeling, Application of Gibbs Sampling

Gowtham Atluri

November 26, 2018

Reading Material

- Gelman et al. Bayesian Data Analysis
  - Chapter 5. Hierarchical Models
- Albert et al. Bayesian Computation with R
  - Chapter 7. Hierarchical Modeling

Topics

- Hierarchical Modeling
  - Motivation
  - Differences with traditional approach
  - Advantages
  - Bayesian Setup
- Normal Hierarchical Model
  - A Complete Bayesian Treatment
    - Model Specification to Point-Estimation
  - Gibbs Sampling
    - Determining full-conditionals
- Julia Implementation
  - Generating data
  - Gibbs Sampling
  - Results

Hierarchical Modeling: Motivating Example

- In many scenarios, we are interested in learning about many parameters that are connected in some way
- *Example*
  - We have data related to post-liver-transplant survival from 94 hospitals
    - We know the number of months patients survived at each hospital from 2000 – 2015
  - We are interested in modeling the survival periods post-transplantation
  - One approach is to pool data from all hospitals and model the parameters of the distribution

λ

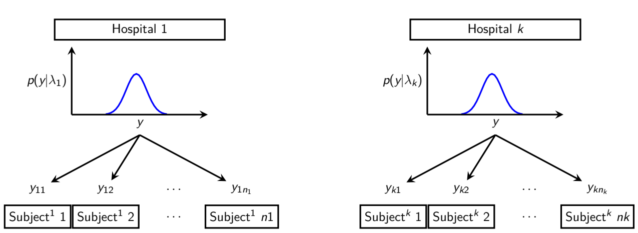
Hospital 1	Hospital 2	...	Hospital 94

  - Not suited for determining which hospital has better survival rates?
  - To address this, we can model the data at each hospital independently
    - Goal is to estimate $\lambda_1, \ldots, \lambda_{94}$

λ_1	λ_2	$\cdots$	λ_{94}
Hospital 1	Hospital 2	$\cdots$	Hospital 94

Hierarchical Modeling: Motivating Example

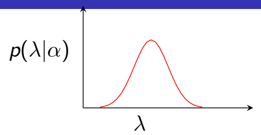
- Let $y_{i1}, y_{i2}, \ldots y_{in_i}$ be the survival periods of n_i subjects at Hospital i.
- Data at each hospital can be modeled independently
  - $y_{ij} \sim p(y|\lambda_i)$, λ_i is the parameter of the *model* at hospital i
    - i is the index of the hospital, j is subject number in hospital i
  - λ_i can be estimated at each hospital separately using MLE or Bayesian approaches



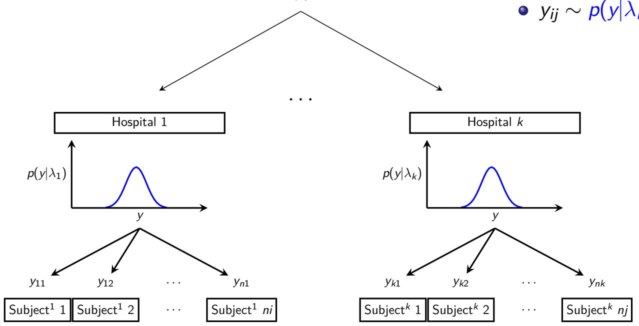
Hierarchical Modeling: Motivating Example

- *It is reasonable to believe that survival is similar across hospitals with some variation*
  - Implies a dependence structure between λs
  - Knowing λ_i affects belief on λ_j
- Specifically, we assume all λs follow a common distribution
  - specific λ_i is sampled from this distribution
  - $\lambda_i \sim p(\lambda|\alpha)$
- Observed data points at each hospital are drawn using the λ_i specific to the hospital
  - $y_{ij} \sim p(y|\lambda_i)$

Hierarchical Modeling



- $\lambda_i \sim p(\lambda)$
- $y_{ij} \sim p(y|\lambda_i)$



Traditional vs. Hierarchical Modeling - Plate Diagrams

At each hospital i

- $y_{ij} \sim p(y|\lambda_i)$

Estimate $\lambda_1, \dots, \lambda_k$, separately

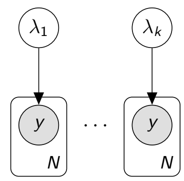
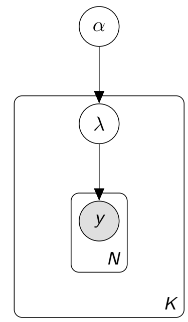


Plate-diag. interpretation:

- Nodes are random vars.
- Arrows show dependency
- Shaded nodes are obs. var.
- Plates for multiple samples

- $\lambda_i \sim p(\lambda|\alpha)$
- $y_{ij} \sim p(y|\lambda_i)$

Estimate $\lambda_1, \dots, \lambda_k$, α



Individual vs. Combined estimation of λ_i's

- Individual estimates λ_i can be highly variable
  - Particularly due to hospitals with a small number of cancer patients
  - There may not be enough samples to accurately estimate *survival rates*
- As individual estimate are poor, it may seem desirable to combine the individual estimates λ_is
  - Treat λ_is as data points and estimate parameter α of the distribution $p(\lambda)$
- Since individual estimates λ_i are already noisy, estimating the parameters of the $p(\lambda)$ is ineffective
- In hierarchical modeling λ_i's and α are estimated simultaneously
  - Overcomes the above limitations with individual modeling

Traditional vs. Hierarchical Modeling - Bayesian Setup

Traditional Model
At each hospital i

- $y_{ij} \sim p(y|\lambda_i)$

Estimate λ_i's

Bayesian setup:

- Likelihood: $p(y_{ij}|\lambda_i)$
- Prior: $p(\lambda_i|\tau)$
- Posterior $p(\lambda_i|y_{ij})$

Prior is on $\lambda_1, \dots, \lambda_k$

Hierarchical Model

- $\lambda_i \sim p(\lambda|\alpha)$
- $y_{ij} \sim p(y|\lambda_i)$

Estimate λ_i's, α

Bayesian setup:

- Likelihood: $\prod_{ij} p(y_{ij}|\lambda_i) p(\lambda_i|\alpha)$
- Prior: $p(\alpha|\phi)$
- Posterior $p(\lambda_1, \dots, \lambda_k, \alpha|y)$

Prior is only on α, not for $\lambda_1, \dots, \lambda_k$

Normal Hierarchical Model

We assume y_{ij} and λ_i follow Gaussian distribution

- λ_i is the mean for hospital i
- variance is σ^2 and is the same for all hospitals

General Version

- $y_{ij} \sim p(y|\lambda_i)$
- $\lambda_i \sim p(\lambda|\alpha)$
- Prior: $p(\alpha|\phi)$
- Likelihood: $\prod_{ij} p(y_{ij}|\lambda_i) p(\lambda_i|\alpha)$

Specific Version: Using Normal distr.

- $y_{ij} \sim \mathcal{N}(\lambda_i, \sigma^2)$
  - where $i = 1, \dots, k$, $j = 1, \dots, n_i$, $n = \sum_{i=1}^k n_i$
- $\lambda_i \sim \mathcal{N}(\mu, \tau^2)$
- (flat) Prior: $p(\mu, \sigma^2, \tau^2) = p(\mu)p(\sigma^2)p(\tau^2) \propto \frac{1}{\sigma^2\tau^2}$

Normal Hierarchical Model

- Generative Model:
  - $y_{ij} \sim \mathcal{N}(\lambda_i, \sigma^2)$
    - where $i = 1, \dots, k$, $j = 1, \dots, n_i$, $n = \sum_{i=1}^k n_i$
  - $\lambda_i \sim \mathcal{N}(\mu, \tau^2)$
- Non-Inf. Prior: $p(\mu, \sigma^2, \tau^2) = p(\mu)p(\sigma^2)p(\tau^2) \propto \frac{1}{\sigma^2\tau^2}$

$$
\begin{aligned}
\text{Posterior } p(\lambda_1, \dots, \lambda_k, \alpha|y) &\propto p(y|\lambda)p(\lambda|\alpha)p(\alpha) \\
&\propto \prod_{ij} p(y_{ij}|\lambda_i)p(\lambda_i|\alpha)p(\alpha) \\
&\propto \prod_{ij} p(y_{ij}|\lambda_i, \sigma^2)p(\lambda_i|\mu, \tau^2)p(\sigma^2, \mu, \tau^2) \\
&\propto \prod_{ij} \mathcal{N}(y_{ij}|\lambda_i, \sigma^2)\mathcal{N}(\lambda_i|\mu, \tau^2)\frac{1}{\sigma^2\tau^2}
\end{aligned}
$$

Gibbs Sampling for Normal Hierarchical Model

$$p(\lambda_1,\dots,\lambda_k,\sigma^2,\mu,\tau^2|y) \propto \prod_{ij}\mathcal{N}(y_{ij}|\lambda_i,\sigma^2)\mathcal{N}(\lambda_i|\mu,\tau^2)\frac{1}{\sigma^2\tau^2}$$

1. Initialize $\lambda_1^{(1)},\dots,\lambda_k^{(1)}$, $\sigma^{2(1)}$, $\mu^{(1)}$, $\tau^{2(1)}$
2. **for** run $=$ 2:n
3. **for** $i = 1,\dots,k$ $\quad\lambda_i^{(run)} \sim p(\lambda_i|,\dots)$ **end**
4. $\sigma^{2(run)} \sim p(\sigma^2|\dots)$
5. $\mu^{(run)} \sim p(\mu|\dots)$
6. $\tau^{2(run)} \sim p(\tau^2|\dots)$
7. **end**

These full conditionals can be written by retaining only the terms in the posterior that has the parameter of interest

Full conditional for λ_i

Full conditional for λ_i is

$$\begin{aligned}p(\lambda_i|\dots) &\propto p(\lambda_1,\dots,\lambda_k,\sigma^2,\mu,\tau^2|y)\\ &\propto \prod_{ij}\mathcal{N}(y_{ij}|\lambda_i,\sigma^2)\mathcal{N}(\lambda_i|\mu,\tau^2)\frac{1}{\sigma^2\tau^2}\\ &\propto \prod_{j=1}^{n_i}\mathcal{N}(y_{ij}|\lambda_i,\sigma^2)\mathcal{N}(\lambda_i|\mu,\tau^2)\end{aligned}$$

- Notice that this not include other $\lambda_{i'}$, for any $i' \neq i$.
  - i.e., λ_i are conditionally independent of each other

Full conditional for λ_i

$$p(\lambda_i|\dots) = \prod_{j=1}^{n_i}\mathcal{N}(y_{ij}|\lambda_i,\sigma^2)\mathcal{N}(\lambda_i|\mu,\tau^2)$$

- We know: product of Gaussians is a Gaussian

$$\begin{aligned}p(\lambda_i|\dots) &= \mathcal{N}(\mu_i,\tau_i^2) \qquad \text{(abusing notation)}\\ \text{where } \tau_i^2 &= [\tau^{-2} + n_i\sigma^{-2}]^{-1}\\ \mu_i &= \tau_i^2[\mu\tau^{-2} + \bar{y}_i n_i\sigma^{-2}]\\ \bar{y}_i &= \frac{1}{n_i}\sum_{j=1}^{n_i} y_{ij}\end{aligned}$$

Full conditional for σ^2

Full conditional for σ^2 is

$$\begin{aligned}p(\sigma^2|\dots) &\propto p(\lambda_1,\dots,\lambda_k,\sigma^2,\mu,\tau^2|y)\\ &\propto \prod_{ij}\mathcal{N}(y_{ij}|\lambda_i,\sigma^2)\mathcal{N}(\lambda_i|\mu,\tau^2)\frac{1}{\sigma^2\tau^2}\\ &\propto \prod_{ij}\mathcal{N}(y_{ij}|\lambda_i,\sigma^2)\frac{1}{\sigma^2}\\ &\propto \prod_{ij}(\sigma^2)^{-1/2}\exp\Big(-\frac{1}{2\sigma^2}(y_{ij}-\lambda_i)^2\Big)\frac{1}{\sigma^2}\\ &\propto (\sigma^2)^{-n/2-1}\exp\Big(-\frac{1}{2}\sum_{i=1}^{k}\sum_{j=1}^{n_i}(y_{ij}-\lambda_i)^2\Big/\sigma^2\Big)\end{aligned}$$

This matches with the kernel of $InverseGamma \propto x^{-(\alpha+1)}exp(-\theta/x)$
$p(\sigma^2|\dots) = InvergeGamma\Big(\alpha = \frac{n}{2}, [\theta = \frac{1}{2}\sum_{i=1}^{k}\sum_{j=1}^{n_i}(y_{ij}-\lambda_i)^2]\Big)$

Full conditional for τ^2

Full conditional for τ^2 is

$$\begin{aligned}p(\tau^2|\dots) &\propto p(\lambda_1,\dots,\lambda_k,\sigma^2,\mu,\tau^2|y)\\ &\propto \prod_{ij}\mathcal{N}(y_{ij}|\lambda_i,\sigma^2)\mathcal{N}(\lambda_i|\mu,\tau^2)\frac{1}{\sigma^2\tau^2}\\ &\propto \prod_{i}\mathcal{N}(\lambda_i|\mu,\tau^2)\frac{1}{\tau^2}\\ &\propto \prod_{i}(\tau^2)^{-1/2}\exp\Big(-\frac{1}{2\tau^2}(\lambda_i-\mu)^2\Big)\frac{1}{\tau^2}\\ &\propto (\tau^2)^{-k/2-1}\exp\Big(-\frac{1}{2\tau^2}\sum_{i}(\lambda_i-\mu)^2\Big)\end{aligned}$$

This matches with the kernel of $InverseGamma \propto x^{-(\alpha+1)}exp(-\theta/x)$
$p(\tau^2|\dots) = InverseGamma\Big(\alpha = k/2, \beta = [\frac{1}{2}\sum_i(\lambda_i-\mu)^2]$

Full conditional for μ

Full conditional for μ is

$$\begin{aligned}p(\mu|\dots) &\propto p(\lambda_1,\dots,\lambda_k,\sigma^2,\mu,\tau^2|y)\\ &\propto \prod_{i}\mathcal{N}(\lambda_i|\mu,\tau^2)\frac{1}{\tau^2}\end{aligned}$$

The marginal posterior posterior for non-informative prior is $\mathcal{N}(\bar{\lambda},\tau^2/k)$
where $\bar{\lambda} = \frac{1}{k}\sum_{i=1}^{k}\lambda_i$

Gibbs Sampling for Normal Hierarchical Model

Use Gibbs sampling to draw samples from the full posterior

1. Initialize $\lambda_1^{(1)}, \dots, \lambda_k^{(1)}$, $\sigma^{2(1)}$, $\mu^{(1)}$, $\tau^{2(1)}$
2. **for** run $=$ 2:n
3. **for** $i = 1, \dots, k$ $\quad \lambda_i^{(run)} \sim p(\lambda_i | , \dots)$ $\quad$ **end**
4. $\sigma^{2(run)} \sim p(\sigma^2 | \dots)$
5. $\mu^{(run)} \sim p(\mu | \dots)$
6. $\tau^{2(run)} \sim p(\tau^2 | \dots)$
7. **end**

- For parameters of survival rates at hospital i
  - compute point-estimates for λ_i, σ^2
- For parameters of survival rate distribution
  - compute point-estimates for μ, τ^2

Julia Implementation

Generating the data

- Generative Model:
  - $y_{ij} \sim \mathcal{N}(\lambda_i, \sigma^2)$
  - $\lambda_i \sim \mathcal{N}(\mu, \tau^2)$

```
k = 100; #number of hospitals
n_k = 1000; # num. subjects/hospital
mu = 5; # mean of p(lambda)
tau = sqrt(1); # std of p(lambda)
dl = Normal(mu,tau);
lambda = rand(dl,100); #generating lambda
sigma = sqrt(0.1);
y = zeros(100,1000);
for i=1:100 #generating observations
    for j=1:1000
        y[i,j] = rand(Normal(lambda[i],sigma));
    end
end
```

Non-Informative Prior: $p(\mu, \sigma^2, \tau^2) = p(\mu)p(\sigma^2)p(\tau^2) \propto \frac{1}{\sigma^2\tau^2}$

Julia implementation: Setting up Gibbs sampling

```
# setup
nruns = 10000;
lambda_est = zeros(100,nruns);
sigma_est = zeros(nruns);
mu_est = zeros(nruns);
tau_est = zeros(nruns);

# initialization
for i=1:100
    lambda_est[i,1] =
      rand(Normal(rand(Uniform(0,10)),rand(Uniform(0,0.1))));
end
sigma_est[1] = rand(Uniform(0,0.1));
mu_est[1] = rand(Normal(rand(Uniform(0,10))));
tau_est[1] = rand(Uniform(0,0.1));
```


Julia implementation: Gibbs sampling

```
for runs=2:nruns
    # estimating lamdda
    for i=1:100
        tau_i = 1/sqrt(1/(tau_est[runs-1]) + 1000/(sigma_est[runs-1]));
        mu_i = (tau_i^2)*(mu_est[runs-1]/(tau_est[runs-1]) +
                                   mean(y[i,:])*1000/(sigma_est[runs-1]));
        lambda_est[i,runs] = rand(Normal(mu_i,sqrt(tau_i)));
    end

    # estimating sigma
    sigma_sum_term = 0;
    for i=1:100
        for j=1:1000
            sigma_sum_term += (y[i,j]-lambda_est[i,runs])^2;
        end
    end
    sigma_est[runs] = rand(InverseGamma(100*1000/2,sigma_sum_term/2));

        #...continued
```


Julia implementation: Gibbs sampling

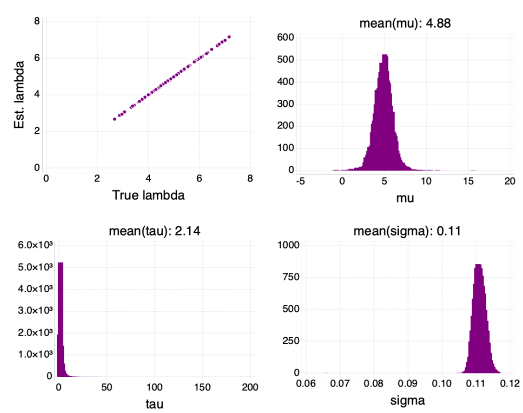
```
    #... continued from previous slide

    # estimating tau
    tau_sum_term = 0;
    for i=1:100
        tau_sum_term += (lambda_est[i,runs] - mu_est[runs-1])^2;
    end

    tau_est[runs] = rand(InverseGamma(100/2,tau_sum_term/2));

    # estimating mu
    mu_est[runs] = rand(Normal(mean(lambda_est[:,runs-1]),
                             sqrt((tau_est[runs])/2)));
end
```


Results



Estimated parameters match precisely with the parameters used for generating data.

Summary

- Hierarchical Modeling allows the use of domain knowledge that connects parameters by the structure of the problem
  - Domain knowledge implies that joint distribution for the parameters should reflect their dependence
- Difference between Traditional modeling vs. Hierarchical modeling
- Bayesian setup for hierarchical modeling
  - Specifying the prior and computing the posterior
- Computing point-estimates for parameters of interest
  - Gibbs sampling
    - No need to select a candidate distribution
    - Need to determine full-conditionals
- Julia implementation
  - Results match precisely with the parameters used to generate the data