Introduction to hierarchical models: Varying intercepts

Patrick Breheny

March 19

Motivation

- Suppose that a certain operation is performed in four hospitals: A, B, C, and D
- Further suppose that the observed mortality rates in A, B, and C are 10%, 19%, and 14%; what would you predict about hospital D?
- It seems unlikely that all hospitals, given that they employ different surgeons, serve different populations of patients, and may have different protocols, have the exact same underlying mortality rate
- However, it also seems natural to think that the hospitals have some similarity to each other and that the mortality rates in A, B, and C tell us something about the mortality rate in D (which is probably somewhere between 10% and 20%)

Identical vs. independent vs. Hierarchical

- To make this more concrete, let θ_i denote the mortality rate in hospital i (or more abstractly, some parameter of interest for unit i)
- We refer to the assumption that $\theta_1 = \theta_2 = \cdots = \theta$ as the "identical parameters" model
- We refer to the other extreme, that $\{\theta_i\}$ are completely unrelated to each other, as the "independent parameters" model
- Bayesian modeling allows a natural compromise between the two extremes: we can assume that the θ_i arise from a common distribution, say, $\theta_i \sim \mathrm{N}(\mu, \sigma^2)$

Hyperparameters and hyperpriors

- The sort of model is different from what we have seen before in the sense that, while there is still observed data $\{y_i\}$ that depend on unobservable parameters $\{\theta_i\}$, the unobservable parameters themselves depend on yet more unobservable parameters (let's call those parameters ϕ)
- Parameters like ϕ, which control the distribution of other parameters (as opposed to controlling the distribution of observable quantities), are known as *hyperparameters*
- Like all unknown quantities in Bayesian statistics, ϕ must be given a prior; the prior of a hyperparameter is known as a *hyperprior*

Hierarchical/multilevel models

- Thus, our Bayesian model involves the parameters $\{\theta_i\}$ arising independently from a common distribution, conditional on the values of the hyperparameters
- Our full prior, then, takes the following form:

$$p(\boldsymbol{\theta}, \phi) = p(\phi) \prod_i p(\theta_i | \phi_i)$$

- Note that our prior is specified in multiple levels, or layers; consequently, this type of model is known as a *hierarchical model* or a *multilevel model*

Remarks

- It is worth noting that similar sorts of models can be proposed in frequentist statistics, and are referred to as *random effects models* or *mixed effects models*
- Hierarchical/multilevel/mixed effects models are most often employed in cases where our data consists of observations that are not independent (if we do not condition on $\{\theta_i\}$, the observations within a hospital are correlated), but they have other uses as well
- For example, when we discussed skeptical (ridge regression) priors for regression models, we arbitrarily specified the value ω; a more natural approach is to specify a distribution for ω and let the data guide our skepticism about the collection $\{\beta_j\}$

Radon data

- To introduce the concepts involved in hierarchical modeling, let's look at data from the State Residential Radon Survey, a study coordinated by the Environmental Protection Agency
- Radon is a naturally occurring radioactive gas that, in high concentrations, is known to cause lung cancer
- Radon concentrations vary considerably from house to house; the purpose of the study was to identify risk factors for houses whose residents might be suffering from dangerously high exposures

Radon data (cont'd)

- Because radon exposure is highly right-skewed, we will take a log transformation, which roughly normalizes the distribution (denote this Y)
- For the sake of simplicity, we will consider only the data from Minnesota (several states were involved in the full study) and two of the potential risk factors:
    - `floor`: Whether the measurement was taken in the house's basement (0) or first floor (1); denote this x
    - `county`

Models

- To illustrate the ways in which hierarchical models differ from other models, we will consider three possible analyses of this data
- "Identical": A simple linear regression model for `floor` treating all counties as identical (our textbook calls this the "pooled" model)
- "Independent": A linear regression model for `floor` in which each county has an independently estimated intercept (our textbook calls this the "unpooled" model)

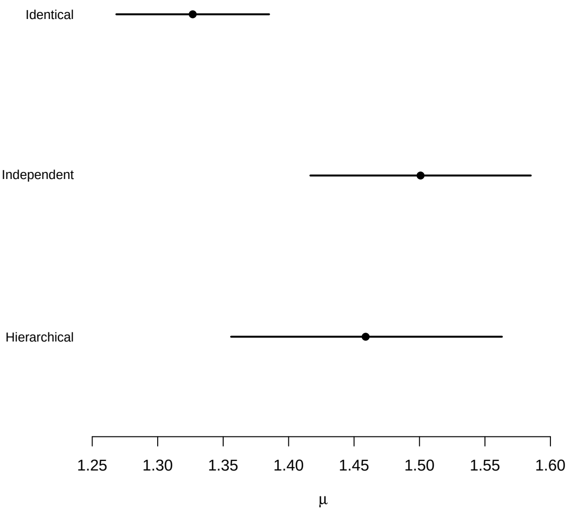
Hierarchical model

- Our third model is hierarchical; letting i index houses and j index counties,

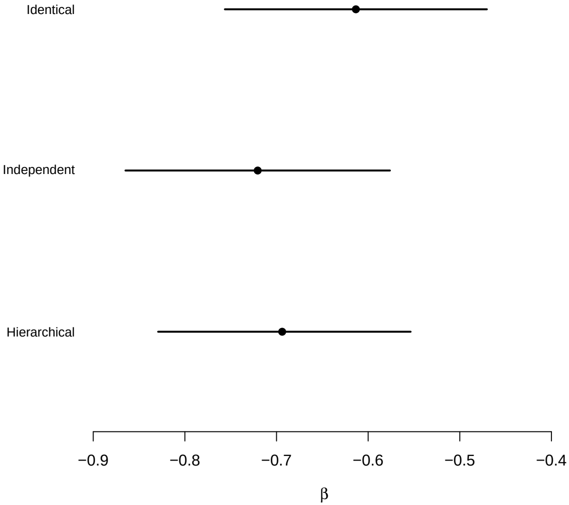
$$\begin{aligned} Y_{ij} &\sim \mathrm{N}(\alpha_j + \beta x_{ij}, \sigma_y^2) \\ \alpha_j &\sim \mathrm{N}(\mu, \sigma_\alpha^2), \end{aligned}$$

  with μ, β, σ_1, and σ_2 given standard uninformative/reference priors
- Note that μ here (a hyperparameter) represents the overall, "population" average intercept, while $\{\alpha_j\}$ are the county-specific intercepts
- Further note that σ_y^2 is the "within-county" variability between houses, while σ_α^2 is the "between-county" variability

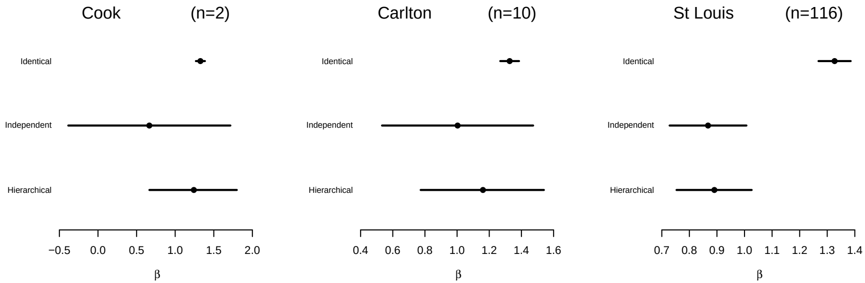
Inference for μ



Inference for β



Inference for α_j

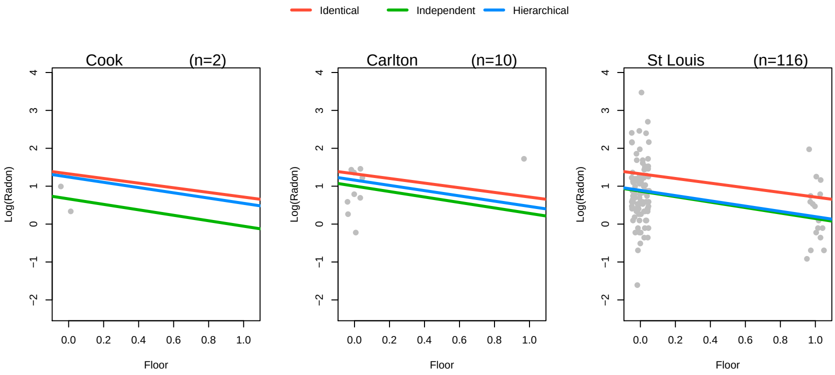


Features of hierarchical models

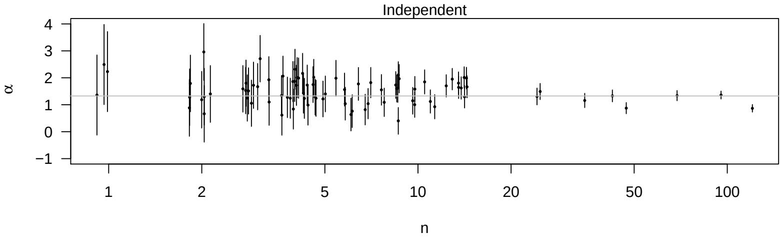
This last plot illustrates three essential features of hierarchical models:

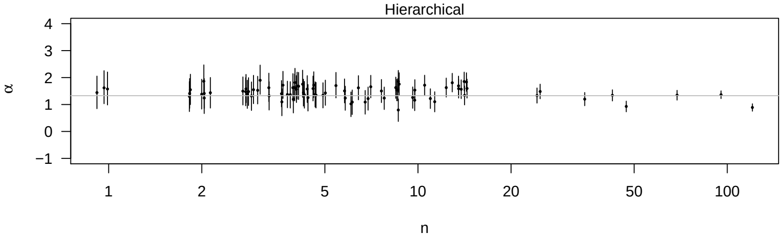
- *Shrinkage*: County-specific means are pulled toward the population mean
- *Smoothing of uncertainty*: Uncertainty about the county-specific means is lower (sometimes much lower) than if these parameters were estimated independently
- *Proportional borrowing of information*: Shrinkage and uncertainty reduction do not occur uniformly – information-poor counties must borrow a great deal of information from the other counties, while information-rich counties do not

Regression line estimates



Inference: All $\{\alpha_j\}$





Inference: Variance components

- Finally, it is worth noting that the within-county variability is quite a bit larger than the between-county variability ($\bar{\sigma}_y = 0.76$, $\bar{\sigma}_\alpha = 0.33$)
- Alternatively, we may express this as a ratio, where $\sigma_\alpha^2/\sigma_y^2 = 0.19$, with a 95% PI of (0.11, 0.32)
- Another common summary measure is the *intraclass correlation*:

  $$\text{ICC} = \frac{\sigma_\alpha^2}{\sigma_\alpha^2 + \sigma_y^2};$$

  if members of a group are unrelated, ICC → 0 (the grouping contains no information); if members of a group are identical, ICC → 1 (the grouping contains all the information)
- Here, the ICC is estimated to be 0.16, with a 95% PI of (0.10, 0.24)

Other quantities of interest

- As with any Bayesian model, it is straightforward to carry out inference for other quantities of interest using Monte Carlo approaches
- For example, if for some reason we were interested in the probability that radon levels were higher in Carlton county than St. Louis county,

```
> mean(a[,9] > a[,73])
[1] 0.9069333
```

- We could obtain the same result (91% posterior probability) by creating a variable `pi <- a[,9] > a[,73]` in BUGS/JAGS; its posterior would then be reported by `print(fit)`

Other quantities of interest (cont'd)

The two are mathematically equivalent, but have some advantages and disadvantages in practice:

- Calculating these quantities in `R` allows interactive exploration and is typically easier to debug
- Calculating these quantities in `R` does not require storing the entire chain for derived quantities
- Calculating these quantities in BUGS/JAGS allows MCMC diagnostics to be run more easily

Prediction/forecasting

- It is worth paying some attention, however, to the issue of *prediction* or *forecasting*
- In principle, this is no different than obtaining the posterior for any other quantity; there are, however, two important differences:
    - Observed quantities are related to parameters stochastically, and therefore are slightly more complicated to draw than other quantities of interest
    - Prediction involves an observed quantity and may, therefore, be subjected to an empirical test of accuracy
- We consider two predictions/forecasts:
    - A radon measurement for a house in an existing county
    - A radon measurement for a house in an new county

Prediction: A house in Carlton county

- Generating posterior predictions involves an additional layer of simulation:

```
nn <- nrow(a)
yy <- rnorm(nn, a[,9], sigma[,1]) ## Carlton county
quantile(yy, c(.025, .5, .975))
      2.5%        50%      97.5%
-0.3393993  1.1640940  2.7190514
```

- Note that we are generating `nn` (here, 15,000) MC draws for this hypothetical new house in Carlton county based on 15,000 draws of α_9 and 15,000 draws of σ_y, the two parameters Y depends on
- Alternatively, we could have created an extra row in our data set with `county` and `floor` specified, but `y` set to `NA`, in which case BUGS/JAGS will do the simulation for you

Prediction: A house in a new county

- This particular data set happens to contain measurements for all the counties in Minnesota, but suppose there was an 86th county for which we had no data
- We can still generate predictions for such a house; it simply involves one more step:

```
> aa <- rnorm(nn, mu, sigma[,2])
> yy <- rnorm(nn, aa, sigma[,1])
> quantile(yy, c(.025, .5, .975))
     2.5%        50%      97.5%
-0.172425  1.465341  3.076426
```

- For all these predictions, an important thing to keep in mind is the *propagation of uncertainty* that Bayesian MCMC methods allow