Lecture 7: Introduction to hierarchical Bayesian Modeling

Zachary Marion

2/14/2018

As before, we need to load some packages and set some options prior to running any models:

```
library(rstan)
library(shinystan)
library(car)
rstan_options(auto_write = TRUE)
options(mc.cores = parallel::detectCores())
source("../utilityFunctions.R")
```

Our first hierarchical model

In our examples so far, we have been using bernoulli and binomial likelihoods with beta priors.

- These beta priors have been parameterized in terms of modes (ω) and concentrations (κ).

$$
\begin{aligned}
y &\sim \text{Binomial}(N, \theta) \\
\theta &\sim \text{Beta}(\alpha, \beta) \\
\alpha &= \omega(\kappa - 2) + 1 \\
\beta &= (1 - \omega)(\kappa - 2) + 1
\end{aligned}
$$

Under this parameterization, each globe toss's probability of turning up W is described by the parameter θ, which depends on a parameter ω describing the typical proportion of water covering the globe.

- This means that when ω is 0.7, θ will be near 0.7.
- κ tells us how close θ is to ω, with larger values of κ generating $\theta's$ closer to ω
  - κ is prior certainty of dependence of θ on ω

Until now both ω and κ were considered as fixed by prior knowledge (i.e., data). Now we consider ω as a parameter to estimate with it's own prior distribution (we will assume κ is still fixed for now) such as

$$p(\omega) = \text{Beta}(\omega|\alpha_\omega, \beta_\omega)$$

where α_ω & β_ω are constants and we believe ω is typically near the mode

$$\frac{\alpha_\omega - 1}{\alpha_\omega + \beta_\omega - 2}.$$

Considering how Bayes's rule applies to this situation,

$$p(\theta, \omega|y) \propto p(y|\theta, \omega)p(\theta, \omega) \tag{1}$$

The likelihood function—$y \sim \text{Binomial}(N, \theta)$—does not involve ω. Therefore, we can rewrite $p(y|\theta, \omega)$ as $p(y|\theta)$. By the definition of conditional probabilities,

$$p(\theta|\omega) = \frac{p(\theta, \omega)}{p(\omega)} \tag{2}$$

we can refactor the joint prior: $p(\theta, \omega) = p(\theta|\omega)p(\omega)$.

Thus the model overall can be rewritten as a hierarchical chain of dependencies

$$\begin{aligned} p(\theta, \omega|y) &\propto p(y|\theta, \omega)p(\theta, \omega) \\ &\propto p(y|\theta)p(\theta|\omega)p(\omega) \end{aligned} \tag{3}$$

Setting up these models is not very hard. It only requires a few tweaks from the models we have already made.

To show what is going on, I am going to make two stan models, `simpleHier1` and `simpleHierNL1`, that are identical, only `simpleHierNL1` will have the likelihood commented out so that we can see the influence of the prior.

```
data {
  int<lower=0> nObs;        // Total number of observations
  int<lower=0> N;
  int<lower=0> obs;   // obs as scalar
  real<lower=2> kappa;      // concentration
  real<lower=0> alpha_omega;    // priors on Omega
  real<lower=0> beta_omega;
}

parameters {
  real<lower=0, upper=1> omega;     // overall prior mode
  real<lower=0, upper=1> theta;     // prob. of water
}

model {
  omega ~ beta(alpha_omega,beta_omega);
  { //  a & b are local parameters and not saved.
```

```
    real alpha;
    real beta;
    alpha = omega * (kappa - 2) +1;
    beta = (1 - omega) * (kappa - 2) + 1;
    theta ~ beta(alpha, beta);
  }

  obs ~ binomial(N, theta);
}
```

Low certainty in ω, high κ

We have three parameters to specify now: α_ω, β_ω, and κ. Initially, we will specify weak priors for α_ω & β_ω and strong priors for κ:

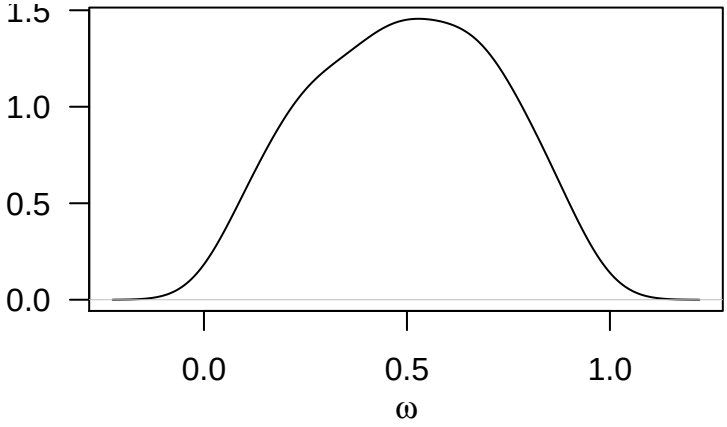
```
alpha_omega <- 2
beta_omega <- 2
kappa <- 100

N <- 6
obs <- 5
nObs <- length(N)
datEx1 <- list(alpha_omega=alpha_omega, beta_omega=beta_omega,
               kappa=kappa, nObs=nObs, N=N, obs=obs)
```

First we will run the model without the likelihood to see the influence of the prior.

```
modExNL1 <- stan(file="07.simpleHierNL1.stan", data=datEx1,
                 iter=2000, chains=4, seed=3, verbose = FALSE)
parNL <- as.matrix(modExNL1, pars=c("theta","omega"))
```

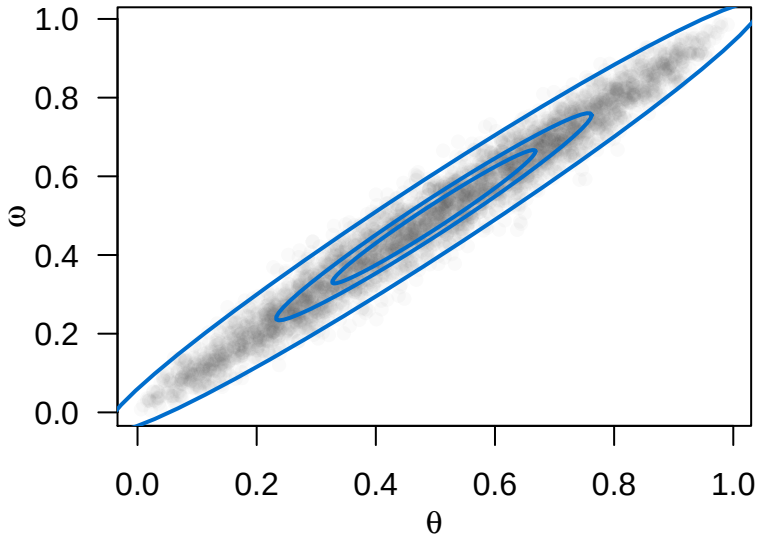
If we look at the marginal posterior probability of ω, we see—not surprisingly given a Beta(2,2) prior—that there is little prior certainty regarding the value of ω.

```
plot(density(parNL[,"omega"], adj=2), main="", xlab="", ylab="",las=1)
mtext(text = expression(paste(omega)), side=1, line = 2)
```



If we plot the joint posterior of ω and θ, however, we see that, by specifying $\kappa = 100$, θ is highly dependent on ω.

```
col <- "#50505008" # specify nice grey color with transparancy
plot(parNL, pch=16, col=col, las=1)
dataEllipse(parNL,level=c(0.25,0.5,0.95), add=TRUE, labels=FALSE,
  plot.points=FALSE, center.pch=FALSE, col=c(col,"#006DCC"))
mtext(text = expression(paste(omega)), side=2, line = 2.2)
mtext(text = expression(paste(theta)), side=1, line = 2)
```

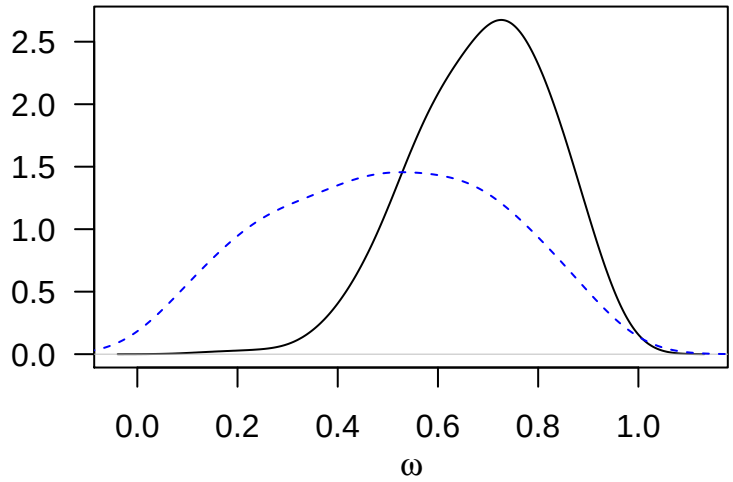


- The ellipses specify the 0.25, 0.5, and 0.95 density contours.

If we run the model with the likelihood included, things look different.

```
modEx1 <- stan(file="07.simpleHier1.stan", data=datEx1, iter=2000,
               chains=4, seed=3, verbose = FALSE)
parL <- as.matrix(modEx1, pars=c("theta","omega"))
```

```
plot(density(parL[,"omega"], adj=2), main="", xlab="", ylab="",las=1)
lines(density(parNL[,"omega"], adj=2),col="blue",lty=2)
mtext(text = expression(paste(omega)), side=1, line = 2)
```

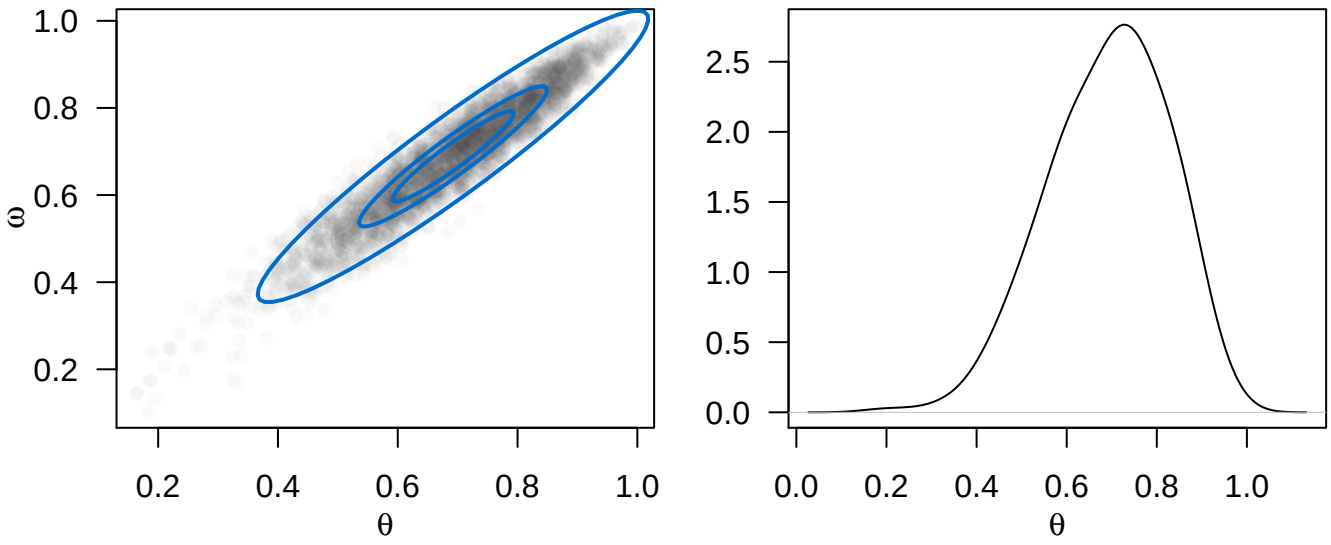


Note that the marginal distribution of $p(\omega|y)$ (black) is now much different than the marginal distribution of the prior $p(\omega)$.

- The low certainty in the value of ω means the data has had a noticable impact on beliefs as to $\omega's$ value.

The high certainty in the dependence of θ on ω means that the joint posterior $p(\theta, \omega)$ has not changed very much in shape. θ has just been shifted along with ω in light of the data.

```
par(mar=c(3,3,0.1,0.5))
par(mfrow=c(1,2))
plot(parL, pch=16, col=col, las=1)
dataEllipse(parL,level=c(0.25,0.5,0.95), add=TRUE, labels=FALSE,
  plot.points=FALSE, center.pch=FALSE, col=c(col,"#006DCC"))
mtext(text = expression(paste(omega)), side=2, line = 2.2)
mtext(text = expression(paste(theta)), side=1, line = 2)

plot(density(parL[,"theta"], adj=2), main="", xlab="", ylab="",las=1)
mtext(text = expression(paste(theta)), side=1, line = 2)
```



High certainty in ω, low κ

In contrast, consider what happens with high certainty in ω & low certainty on the dependence of θ on ω:

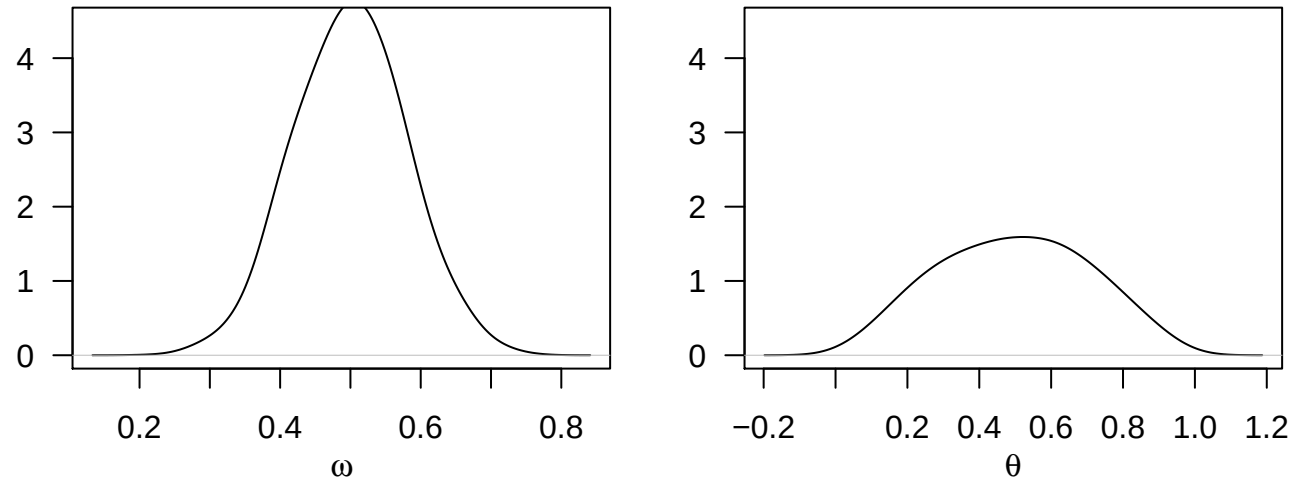
```
alpha_omega <- 20
beta_omega <- 20
kappa <- 5

N <- 6
obs <- 5
datEx2 <- list(alpha_omega=alpha_omega, beta_omega=beta_omega,
               kappa=kappa, N=N, obs=obs)
```

```
modExNL2 <- stan(file="07.simpleHierNL1.stan", data=datEx2,
  iter=2000, chains=4, seed=3, verbose = FALSE)

parNL2 <- as.matrix(modExNL2, pars=c("theta","omega"))
```

Without the data, the marginal probability of ω is tight with a sharp peak at 0.5.

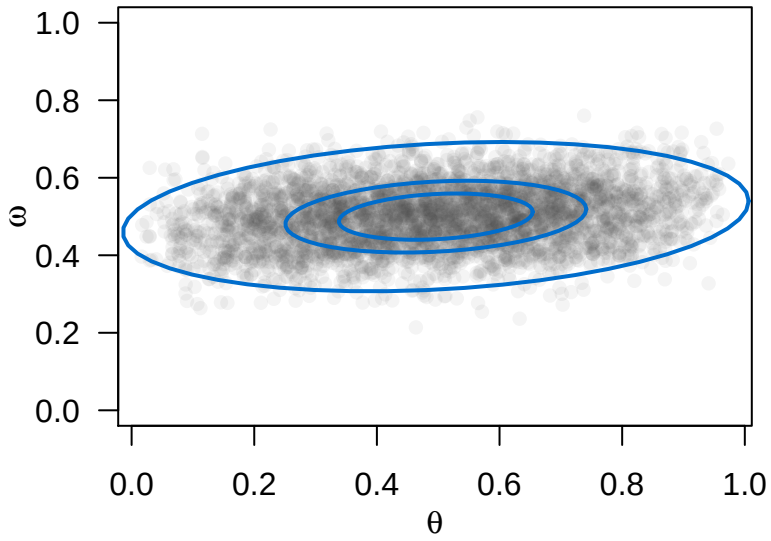
The marginal distribution of θ is fairly broad though because of the low certainty in the dependence of θ on ω.

```
plot(density(parNL2[,"omega"], adj=2), main="", xlab="", ylab="",
     las=1,ylim=c(0,4.5))
mtext(text = expression(paste(omega)), side=1, line = 2)

par(mar=c(3,3,0.1,0.5))
plot(density(parNL2[,"theta"], adj=2), main="", xlab="", ylab="",
     las=1, ylim=c(0,4.5))
mtext(text = expression(paste(theta)), side=1, line = 2)
```



- This is apparent from the joint prior $p(\theta, \omega)$.

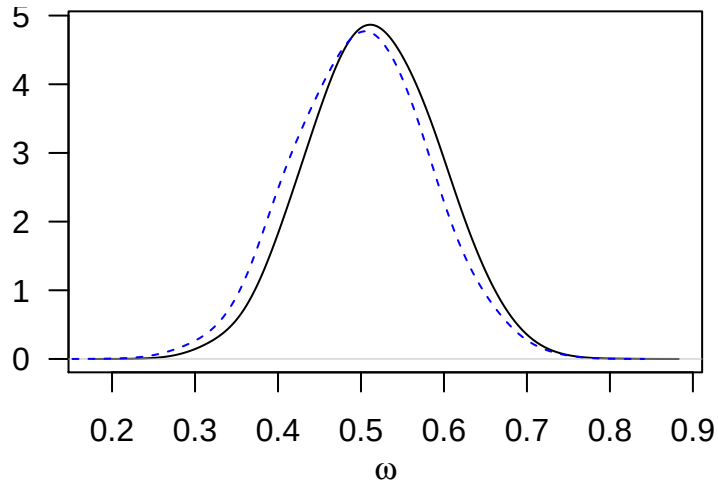
```
plot(parNL2, pch=16, col=col, las=1, ylim=c(0,1))
dataEllipse(parNL2,level=c(0.25,0.5,0.95), add=TRUE, labels=FALSE,
  plot.points=FALSE, center.pch=FALSE, col=c(col,"#006DCC"))
mtext(text = expression(paste(omega)), side=2, line = 2.2)
mtext(text = expression(paste(theta)), side=1, line = 2)
```



Now consider the marginal posterior distribution on ω $p(\omega|y)$ by including likelihood function.

```
modEx2 <- stan(file="07.simpleHier1.stan", data=datEx2,
  iter=2000, chains=4, seed=3, verbose = FALSE)
parL2 <- as.matrix(modEx2, pars=c("theta","omega"))
```

```
plot(density(parL2[,"omega"], adj=2), main="", xlab="", ylab="",las=1)
lines(density(parNL2[,"omega"], adj=2),col="blue",lty=2)
mtext(text = expression(paste(omega)), side=1, line = 2)
```



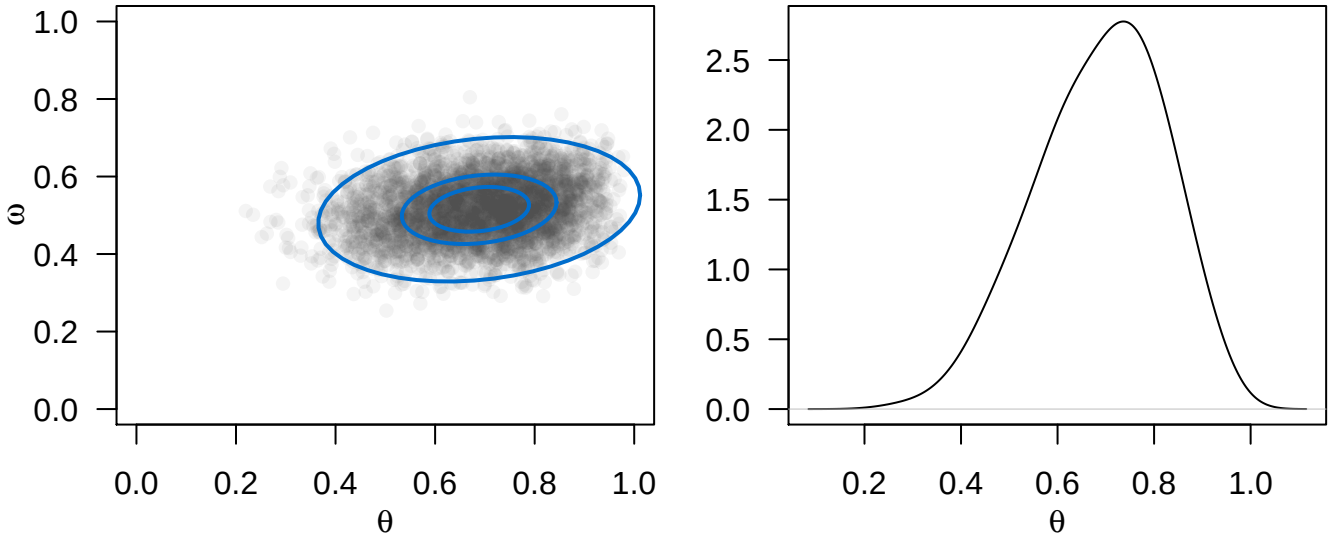
now the marginal distribution of $p(\omega|y)$ (black) is almost identical to the marginal distribution of the prior $p(\omega)$ (blue.

- The high certainty in the value of ω means the data has not affected our beliefs as to $\omega's$ value.

The dependence of θ on ω is quite different now. The low κ means that the posterior of θ is dominated by the data rather than it's dependence on ω.

```
par(mar=c(3,3,0.1,0.5))
par(mfrow=c(1,2))

plot(parL2, pch=16, col=col, las=1, xlim=c(0,1), ylim=c(0,1))
```

```r
dataEllipse(parL2,level=c(0.25,0.5,0.95), add=TRUE, labels=FALSE,
  plot.points=FALSE, center.pch=FALSE, col=c(col,"#006DCC"))
mtext(text = expression(paste(omega)), side=2, line = 2.2)
mtext(text = expression(paste(theta)), side=1, line = 2)

plot(density(parL2[,"theta"], adj=2), main="", xlab="", ylab="",las=1)
mtext(text = expression(paste(theta)), side=1, line = 2)
```



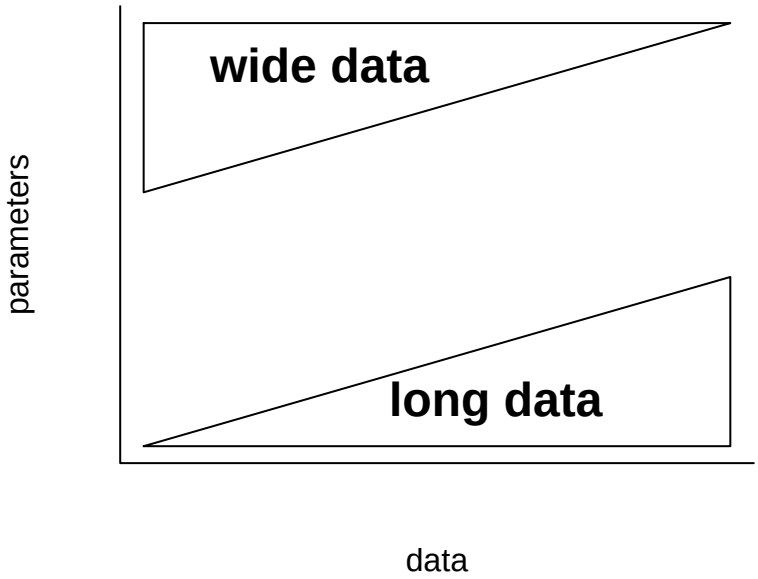
These examples are simple but show the basic concept that hierarchical modeling is Bayesian inference on joint parameter space.

- Shows importance of looking at both the marginal and joint distributions

Practical example of hierarchical models

The previous examples were designed to ease the class into thinking about joint and marginal parameter space, but they don't have much practical utility. So what is the big deal about hierarchical models?

Generally we can place both data and model complexity into a continuum from sparse to dense.



Working with long data isn't a problem, but more often we run into situations where we have more parameters than we do data. Fitting such models can be difficult.

- Want to balance bias (overfitting) and variance (underfitting)

Hierarchical models allow us to model not just individuals but populations through *partial pooling*.

- do this by relaxing *exchangability* (i.e., i.i.d) in the strict sense.
- We can permute any two individuals in the population, and the population still looks the same
  - Not strictly iid but still pretty strong within population

Partial pooling shares information among populations, groups, or clusters with several benefits:

1. **Improved estimates for repeat sampling:** When more than one observation arises from same individual, location, or time, traditional single-level models either over- or underfit.
2. **Improved estimates for sampling imbalance:** When some individuals, locations, or times are sampled more than others, multilevel models implicitly deal with differing uncertainty across groups.
   - Prevents oversampled groups from dominating inference
3. **Estimating variation:** If our questions include variation among individuals or groups, multilevel models are the bee's knees because they explicitly model such variation.
4. **Avoid averaging, retain variation:** Often in traditional analyses groups are collapsed by pre-averaging to make variables. This is naughty because it discards variation, and because there are multiple ways to average.

There are some costs to hierarchical modeling:

1. Hierachical models can be tricky to understand, because estimation occurs at multiple data levels.
2. We have to make more assumptions because now we need to define the distributions from which our population-level parameters arise.

- these distributions should be as biologically intuitive as possible, but...

3. Hierarchical models can be computationally difficult, especially when information or the number of groups is sparse.
    - Scale parameters are especially prone to sampling problems and priors often need to be more informative to keep the MCMC from wandering into regions of high improbability.