Bayesian Linear Mixed Effects Models

Bayesian Inference in the Linear Mixed Effects Model

Recall our model is

$$Y_i = X_i\beta + Z_i b_i + \varepsilon_i,$$

where $b_i \overset{iid}{\sim} N(0, D)$, $\varepsilon_i \overset{iid}{\sim} N(0, R_i)$, and $b_i \perp \varepsilon_i$.

For purposes of prior specification, it will be convenient to express our model as

$$Y_{ij} = X_{ij}\beta_i + \varepsilon_{ij}$$

with

$$\beta_i = \theta + \gamma_i$$

Often we assume $\varepsilon_{ij} \overset{iid}{\sim} N(0, \sigma^2)$, $\gamma_i \overset{iid}{\sim} N(0, \Sigma)$, and we often write $\beta_i \mid \theta \sim N(\theta, \Sigma)$. The parameters θ are fixed effects and the parameters γ_i are random effects.

A conditionally-conjugate prior specification is given by

$\theta \sim N(\mu_0, \Lambda_0)$,

$\Sigma \sim$ inverse-Wishart(η_0, S_0^{-1}),

$\sigma^2 \sim$ inverse-gamma $\left(\frac{\nu_0}{2}, \frac{\nu_0 \sigma_0^2}{2}\right)$

Refresher: Wishart Distribution

- In the univariate case with $y_i \sim N(\mu, \sigma^2)$, an inverse-gamma prior is commonly chosen for the variance σ^2
- This is equivalent to a gamma prior for the precision σ^{-2}
- In the multivariate Gaussian case, we have a covariance matrix Σ instead of a scalar
- Appealing to have a matrix-valued extension of the inverse-gamma that would be conjugate

- One complication is that the covariance Σ must be $\{$*positive definite*$\}$ *and symmetric*
- *Ensures that the diagonal elements of* Σ *(corresponding to the marginal variances* σ_l^2*) are positive*
- *Also, ensures that the correlation coefficients for each pair of variables are between -1 and 1.*
- *Prior for* Σ *must assign probability one to set of positive definite matrices*

Intuition Behind the Wishart: Empirical Covariance Matrices

The *sum of squares* matrix of a collection of multivariate vectors $z_1, \ldots, z_n$ is given by $\sum_{i=1}^{n} z_i z_i' = Z'Z$, where Z is the $n \times p$ matrix whose ith row is z_i'. Note

$$z_i z_i' = \begin{pmatrix} z_{i1}^2 & z_{i1}z_{i2} & \cdots & z_{i1}z_{ip} \\ z_{i2}z_{i1} & z_{i,2}^2 & \cdots & z_{i2}z_{ip} \\ \vdots & & & \vdots \\ z_{ip}z_{i1} & z_{ip}z_{i2} & \cdots & z_{ip}^2 \end{pmatrix}.$$

If the z_i's are from a population with zero mean, we can think of the matrix $\frac{1}{n} z_i z_i'$ as the contribution of z_i to the estimate of the covariance matrix of all the observations. In this case, if we divide $Z'Z$ by n we get a sample covariance, which is an (unbiased – why?) estimator of the population covariance matrix:
$\frac{1}{n}[Z'Z]_{jj} = \frac{1}{n}\sum_{i=1}^{n} z_{ij}^2 = s_{jj} = s_j^2$ and $\frac{1}{n}[Z'Z]_{jk} = \frac{1}{n}\sum_{i=1}^{n} z_{ij} z_{ik} = s_{jk}$

When $n > p$ and the z_i's are linearly independent, $Z'Z$ is positive definite and symmetric! This hints at the following construction of a random covariance matrix for a given positive integer ν_0 and $p \times p$ covariance matrix Φ_0:

- Sample $z_j \overset{iid}{\sim} N_p(0, \Phi_0)$ for $j = 1, \ldots, \nu_0$
- Calculate $\sum_{j=1}^{\nu_0} z_j z_j' = Z'Z$
- Repeat over and over again generating a collection of matrices $Z_1'Z_1, \ldots, Z_S'Z_S$. The population distribution of these sum of squares matrices is a *Wishart distribution* with parameters ν_0 and Φ_0, denoted $\text{Wishart}(\nu_0, \Phi_0)$

Properties of the Wishart

- If the degrees of freedom $\nu_0 > p$, then $Z'Z$ is positive definite with probability 1
- $Z'Z$ is symmetric with probability 1
- $\mathrm{E}(Z'Z) = \nu_0 \Phi_0$
- Hence, Φ_0 is a scaled mean of the Wishart(ν_0, Φ_0)

- A random variable $\Phi \sim \text{Wishart}(\nu_0, \Phi_0)$ has pdf (up to a constant)

$$|\Phi|^{\frac{\nu_0 - p - 1}{2}} e^{-\frac{1}{2}\text{tr}(\Phi_0^{-1}\Phi)},$$

  where $\text{tr}(\cdot)$ is the *trace* function (sum of diagonal elements)
- In univariate case in which $p = 1$, reduces to

$$\phi^{\nu_0/2-1} e^{-\phi\phi_0^{-1}/2} \propto \text{Ga}(\nu_0/2, \phi_0^{-1}/2)$$

- Wishart provides a conditionally-conjugate prior for the precision Σ^{-1} in a multivariate normal model
- The inverse-Wishart is a conditionally conjugate prior for Σ and provides a multivariate generalization of the inverse-gamma

If $\Phi \sim W(\nu_0, \Phi_0)$, then $\Sigma = \Phi^{-1} \sim IW(\nu_0, \Phi_0)$, with

$$\text{pdf} \propto |\Sigma|^{-(\nu_0+p+1)/2} \exp(-tr({\Phi_0}^{-1}\Sigma^{-1})/2)$$

and

$$E[\Sigma^{-1}] = \nu_0 \Phi_0 \quad \text{and} \quad E[\Sigma] = \frac{1}{\nu_0 - p - 1} {\Phi_0}^{-1}.$$

Suppose we choose an inverse-Wishart prior, $\Sigma \sim \mathrm{IW}(\nu_0, S_0^{-1})$

- Up to a norming constant, the pdf is

$$|\Sigma|^{-(\nu_0+p+1)/2} e^{-\mathrm{tr}(S_0\Sigma^{-1})/2}$$

- ν_0= prior dgf, Σ_0 = prior guess for Σ & $S_0 = (\nu_0 - p - 1)\Sigma_0$
- Under this choice, $\mathrm{E}(\Sigma) = \Sigma_0$ & $\nu_0 = p + 2$ would correspond to a vague prior

Back to Priors

Using the prior specification

$\theta \sim N(\mu_0, \Lambda_0)$,

$\Sigma \sim$ inverse-Wishart(η_0, S_0^{-1}), and

$\sigma^2 \sim$ inverse-gamma $\left(\frac{\nu_0}{2}, \frac{\nu_0 \sigma_0^2}{2}\right)$,

a simple Gibbs sampler can be used for posterior computation.

$$\beta_i \mid y_i, X_i, \theta, \Sigma, \sigma^2 \sim N(\mu_{\beta_i}, \Sigma_{\beta_i}),$$

where

$$\Sigma_{\beta_i} = \left(\Sigma^{-1} + \frac{X_i' X_i}{\sigma^2}\right)^{-1}$$

and

$$\mu_{\beta_i} = \left(\Sigma^{-1} + \frac{X_i' X_i}{\sigma^2}\right)^{-1} \left(\Sigma^{-1}\theta + \frac{X_i' y_i}{\sigma^2}\right).$$

$$\theta \mid \beta_1, \ldots, \beta_m, \Sigma \sim N(\mu_\theta, \Lambda_\theta),$$

where $\Lambda_\theta = \left(\Lambda_0^{-1} + m\Sigma^{-1}\right)^{-1}$, $\mu_\theta = \Lambda_\theta \left(\Lambda_0^{-1}\mu_0 + m\Sigma^{-1}\bar{\beta}\right)$, and $\bar{\beta}$ is the vector average $\frac{1}{m}\sum \beta_i$.

$$\Sigma \mid \theta, \beta_1, \ldots, \beta_m \sim \text{inverse-Wishart}\left(\eta_0 + m, [S_0 + S_\theta]^{-1}\right),$$

where

$$S_\theta = \sum_{j=1}^{m} (\beta_j - \theta)(\beta_j - \theta)'$$

$$\sigma^2 \mid \beta_1, \ldots, \beta_m \sim \text{inverse-gamma}\left(\frac{\nu_0 + \sum n_j}{2}, \frac{\nu_0 \sigma_0^2 + SSR}{2}\right),$$

where

$$SSR = \sum_{i=1}^{m} \sum_{j=1}^{n_i} (y_{ij} - x_{ij}\beta_i)^2.$$

Motivation for Other Covariance Priors

While the inverse Wishart is a nice prior for symmetric matrices, computation can be a challenge, expecially if the covariance matrix becomes large.

Why is modeling a covariance matrix difficult?

- number of parameters may be quite large
- matrix constrained to be nonnegative definite

Motivation for Other Covariance Priors

Another down side of the Wishart is that we must use the same df for all elements, though in practice, we may have more information about some components than others.

For example, we may believe in advance that the regression coefficients for one predictor are fairly similar across groups, while we may have little knowledge about similarity of coefficients for another predictor. It is essentially impossible to express these prior beliefs using the inverse Wishart.

A popular alternative approach is to decompose the covariance matrix Σ into a correlation matrix and a diagonal matrix of standard deviations:

$$\Sigma = \begin{pmatrix} \tau_1 & 0 & \cdots & 0 \\ 0 & \tau_2 & \cdots & 0 \\ 0 & \vdots & \cdots & \vdots \\ 0 & \cdots & \cdots & \tau_m \end{pmatrix} \Omega \begin{pmatrix} \tau_1 & 0 & \cdots & 0 \\ 0 & \tau_2 & \cdots & 0 \\ 0 & \vdots & \cdots & \vdots \\ 0 & \cdots & \cdots & \tau_m \end{pmatrix},$$

where $\tau_k = \sqrt{\Sigma_{k,k}}$ and $\Omega_{i,j} = \frac{\Sigma_{i,j}}{\tau_i \tau_j}$.

This separation strategy yields nice interpretations for components, as researchers are often more used to thinking of the standard deviations and correlations than of covariances.

Typically, the priors on τ_k are assumed to be independent of the prior on Ω, though this could be incorporated through a prior on $\Omega \mid \tau$.

In this parameterization, any reasonable prior for scale parameters can be given to the components of the scale vector τ. Popular choices include half-Cauchy or half-normal distributions, but log normal or inverse gamma priors might also be used. This approach is particularly attractive relative to the inverse Wishart, which requires us to use the same df for all elements, though in practice, we may wish to have more flexibility in dealing with tails of individual variance components.

LKJ prior

A nice choice for the correlation matrix is the LKJ prior, which is like an extension of the beta distribution. This prior is

$$\mathsf{LkjCorr}(\Omega \mid \eta) \propto \det(\Omega)^{\eta-1},$$

which for $\eta = 1$ is the joint uniform distribution (note the marginals here are not uniform but favor more mass around 0). For $\eta > 1$, the density concentrates increasing mass around the identiy (favoring lower correlation), and for $\eta < 1$, mass is increasingly spread towards more extreme values.

In-Class Activity!

Plot the LKJ density for a given correlation (unnormalized is ok) for a variety of values of the shape parameter η.

Big hint: you may find this link quite useful along with instructions for installing the rethinking package.[1]

Example: Coffee Robot

We use an example from McElreath's book *Statistical Rethinking* about a coffee robot. While these are simulated data, they provide an interesting application as well as great code should you need to simulate hierarchical data in the future!

Suppose we have a coffee-making robot that moves among cafe's to order coffee and record the wait time. The robot also records the time of the visit because the average wait time in the morning tends to be longer than in the afternoon due to the fact that the cafes are busier in the mornings. The robot learns more efficiently about wait times when it pools information across different cafes.

- We can use varying intercepts to pool information across coffee shops
- Coffee shops vary in average wait times due to a number of factors (e.g., barista skill, number of baristas)
- Coffee shops also vary in differences between morning and afternoon
- Varying intercepts for cafes and slopes for the afternoon effect make for a reasonable model
- In this example we use a mixed model but ignore the longitudinal nature of the data, focusing on the cafe as a grouping factor

Model:

$$y_{ij} = \beta_{0,i} + \beta_{1,i} A_{ij} + \varepsilon_{ij}$$

$$\beta_{0,i} = \alpha_0 + b_{0,i} \quad \beta_{1,i} = \alpha_1 + b_{1,i}$$

$$\varepsilon_i \overset{iid}{\sim} N(0, \sigma^2) \;\perp\; b_i \overset{iid}{\sim} N(0, D), \quad D = \begin{pmatrix} \tau_0 & 0 \\ 0 & \tau_1 \end{pmatrix} \Upsilon \begin{pmatrix} \tau_0 & 0 \\ 0 & \tau_1 \end{pmatrix}$$

Priors:

- $\beta_0 \sim N(0, 10) \qquad \beta_1 \sim N(0, 10)$
- $\sigma \sim \text{Half Cauchy}(0, 1)$
- $\tau_0 \sim \text{Half Cauchy}(0, 1) \qquad \tau_1 \sim \text{Half Cauchy}(0, 1)$
- $\Upsilon = \begin{pmatrix} 1 & \upsilon \\ \upsilon & 1 \end{pmatrix} \sim LKJcorr(2)$

Simulate data

```r
#example from McElreath with thanks to Solomon Kurz for the
library(brms)
```

```
## Warning: package 'brms' was built under R version 3.5.2

## Loading required package: Rcpp

## Warning: package 'Rcpp' was built under R version 3.5.2

## Loading 'brms' package (version 2.9.0). Useful instructi
## can be found by typing help('brms'). A more detailed int
## to the package is available through vignette('brms_overv
```

```r
a       <-  3.5  # average morning wait time
b       <- -1    # average difference afternoon wait time
sigma_a <-  1    # std dev in intercepts
sigma_b <-  0.5  # std dev in slopes
rho     <- -.7   # correlation between intercepts and slope
```

```
library(tidyverse)

sigmas <- c(sigma_a, sigma_b)          # standard deviation
rho    <- matrix(c(1, rho,             # correlation matri
                   rho, 1), nrow = 2)

# now matrix multiply to get covariance matrix
sigma <- diag(sigmas) %*% rho %*% diag(sigmas)

# how many cafes would you like?
n_cafes <- 20

set.seed(13)  # used to replicate example
vary_effects <-
  MASS::mvrnorm(n_cafes, mu, sigma) %>%
  data.frame() %>%
  set_names("a_cafe", "b_cafe")

head(vary_effects)
```

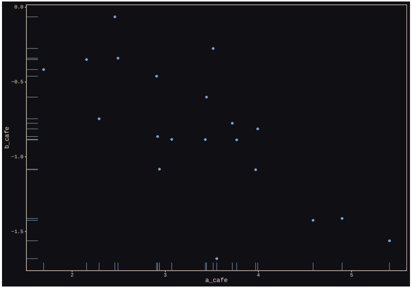
OK, so now we've simulated the cafe-specific intercepts and slopes!

This next block of code adds a pretty set of colors.

```
#ok, McElreath has a thing for colors, so here's his choice
# devtools::install_github("EdwinTh/dutchmasters")
library(dutchmasters)
theme_pearl_earring <-
  theme(text       = element_text(color = "#E8DCCF", family
        strip.text = element_text(color = "#E8DCCF", family
        axis.text  = element_text(color = "#E8DCCF"),
        axis.ticks = element_line(color = "#E8DCCF"),
        line       = element_line(color = "#E8DCCF"),
        plot.background   = element_rect(fill = "#100F14",
        panel.background  = element_rect(fill = "#100F14",
        strip.background  = element_rect(fill = "#100F14",
        panel.grid = element_blank(),
        legend.background = element_rect(fill = "#100F14",
        legend.key        = element_rect(fill = "#100F14",
        axis.line = element_blank())
```

Here we see a negative correlation in our intercepts and slopes (how do we interpret that?). Remember these are the "true" parameters rather than our data.

```
vary_effects %>%
  ggplot(aes(x = a_cafe, y = b_cafe)) +
  geom_point(color = "#80A0C7") +
  geom_rug(color = "#8B9DAF", size = 1/7) +
  theme_pearl_earring
```



```r
n_visits <- 10
sigma    <-  0.5  # std dev within cafes

set.seed(13)  # used to replicate example
d <-
  vary_effects %>%
  mutate(cafe      = 1:n_cafes) %>%
  expand(nesting(cafe, a_cafe, b_cafe), visit = 1:n_visits)
  mutate(afternoon = rep(0:1, times = n() / 2)) %>%
  mutate(mu        = a_cafe + b_cafe * afternoon) %>%
  mutate(wait      = rnorm(n = n(), mean = mu, sd = sigma))
d %>%
  head()
```

```
## # A tibble: 6 x 7
##    cafe a_cafe b_cafe visit afternoon    mu  wait
##   <int>  <dbl>  <dbl> <int>     <int> <dbl> <dbl>
## 1     1   2.92 -0.865     1         0  2.92  3.19
## 2     1   2.92 -0.865     2         1  2.05  1.91
## 3     1   2.92 -0.865     3         0  2.92  3.81
```

First, let's look at that prior for Υ.

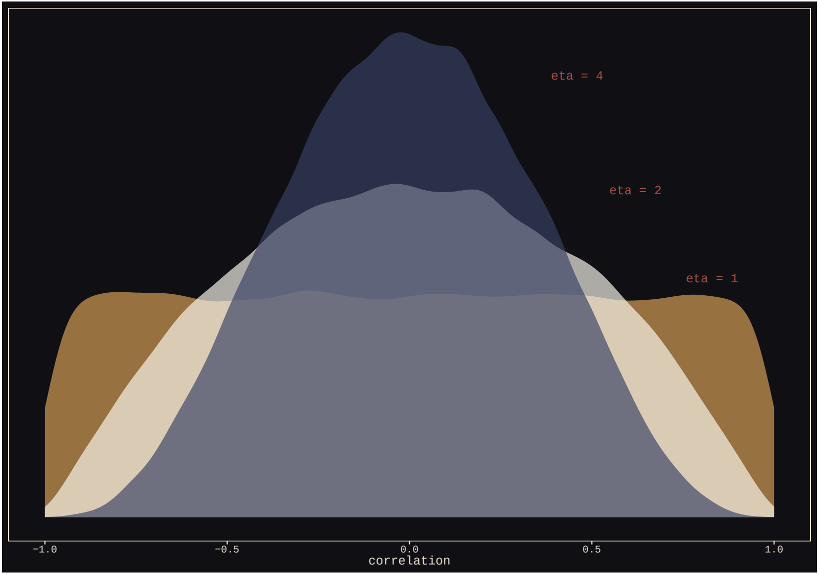
```
library(rethinking)

n_sim <- 1e5

set.seed(13)
r_1 <-
  rlkjcorr(n_sim, K = 2, eta = 1) %>%
  as_tibble()

set.seed(13)
r_2 <-
  rlkjcorr(n_sim, K = 2, eta = 2) %>%
  as_tibble()

set.seed(13)
r_4 <-
  rlkjcorr(n_sim, K = 2, eta = 4) %>%
  as_tibble()
```


eta = 4
eta = 2
eta = 1
-1.0
-0.5
0.0
0.5
1.0
correlation

Now we switch to the brms package and fit the model.

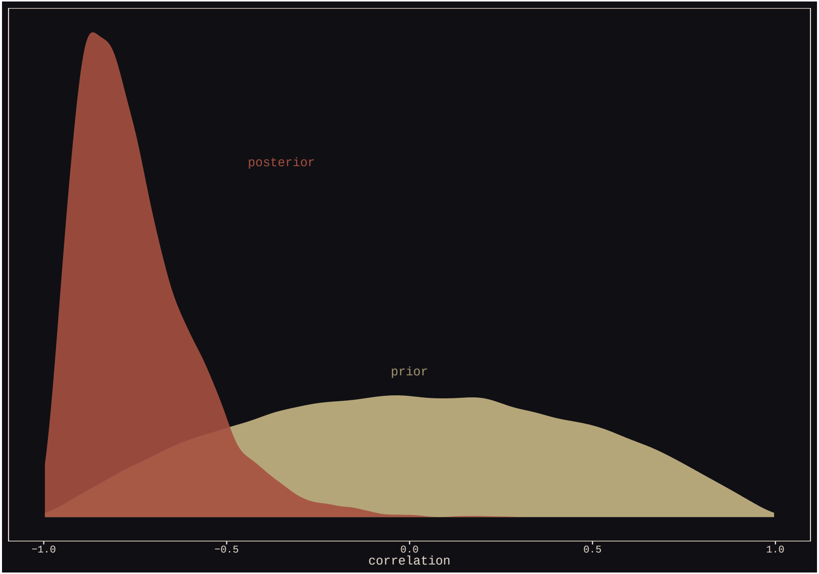
```r
detach(package:rethinking, unload = T)
library(brms)

 b13.1 <-
  brm(data = d, family = gaussian,
      wait ~ 1 + afternoon + (1 + afternoon | cafe),
      prior = c(prior(normal(0, 10), class = Intercept),
                prior(normal(0, 10), class = b),
                prior(cauchy(0, 1), class = sd),
                prior(cauchy(0, 1), class = sigma),
                prior(lkj(2), class = cor)),
      iter = 5000, warmup = 2000, chains = 2, cores = 2,
      seed = 13)
```

Let's compare posterior correlation of random effects to the prior.

```
post <- posterior_samples(b13.1)

post %>%
  ggplot(aes(x = cor_cafe__Intercept__afternoon)) +
  geom_density(data = r_2, aes(x = V2),
               color = "transparent", fill = "#EEDA9D", alp
  geom_density(color = "transparent", fill = "#A65141", alp
  annotate("text", label = "posterior",
           x = -0.35, y = 2.2,
           color = "#A65141", family = "Courier") +
  annotate("text", label = "prior",
           x = 0, y = 0.9,
           color = "#EEDA9D", alpha = 2/3, family = "Couri
  scale_y_continuous(NULL, breaks = NULL) +
  xlab("correlation") +
  theme_pearl_earring
```

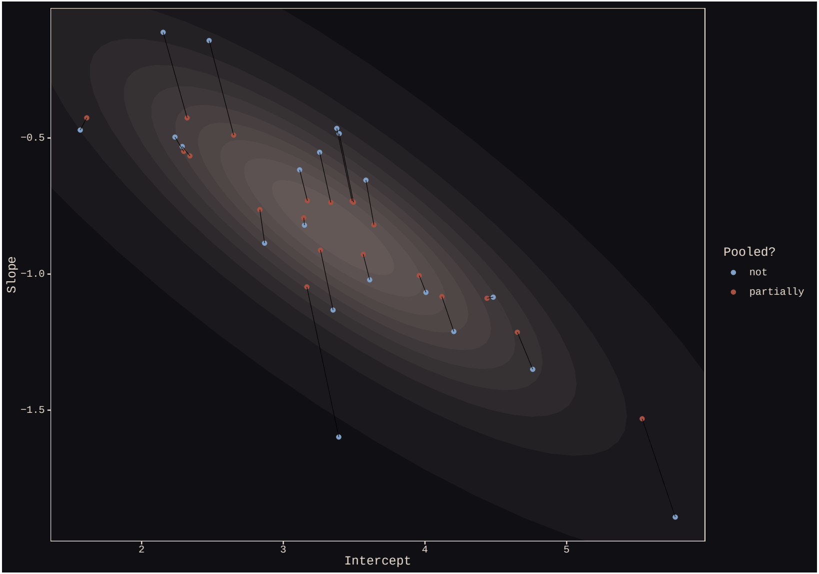

posterior
prior
−1.0
−0.5
0.0
0.5
1.0
correlation

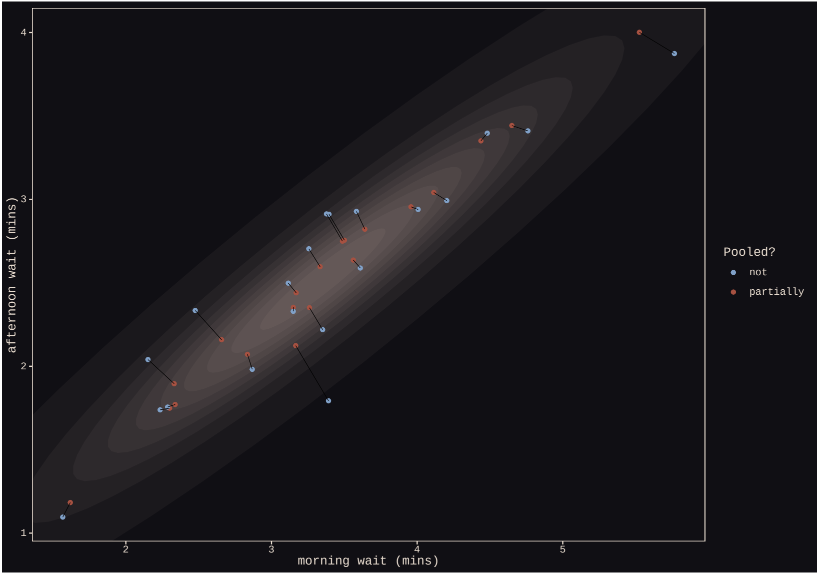
It takes a lot of code to generate the following figures, which illustrate shrinkage in this model. If you're interested, check out my GitHub, or the McElreath book, or Solomon's website.

These figures examine random intercepts vs random slopes as well as the morning and afternoon wait times on the original scale (minutes).

Blue dot: unpooled estimate Red dot: pooled estimate

Note shrinkage is toward the center of the ellipse.


Slope
−0.5
−1.0
−1.5
2
3
4
5
Intercept
Pooled?
not
partially


afternoon wait (mins)
4
3
2
1
2
3
4
5
morning wait (mins)
Pooled?
not
partially