

STA 610L: MODULE 3.7

LOGISTIC MIXED EFFECTS MODEL (PART III)

DR. OLANREWaju MICHAEL AKANDE

SHORT REVIEW: AGGREGATED BINARY OUTCOMES

In the context of logistic regression (and the mixed effect versions), we often observe the binary outcomes for each observation, that is, each $y_i \in \{0, 1\}$.

Of course this is not always the case.

Sometimes, we get an aggregated version, with the outcome summed up by combinations of other variables. For example, suppose we had

response	0	0	1	1	1	0	1	1	0	0	0	1	0	0	1	0	1	1	1	0	0	1	1	0	1
predictor	3	3	2	1	2	3	2	2	2	2	3	1	3	1	1	2	2	2	2	1	3	3	3	1	3

where **predictor** is a factor variable with 3 levels: 1,2,3.

QUICK REVIEW: AGGREGATED BINARY OUTCOMES

The aggregated version of the same data could look then like

predictor	n	successes
1	31	17
2	35	16
3	34	14

QUICK REVIEW: AGGREGATED BINARY OUTCOMES

Recall that if $Y \sim \text{Bin}(n, p)$ (that is, Y is a random variable that follows a binomial distribution with parameters n and p), then Y follows a Bernoulli(p) distribution when $n = 1$.

Alternatively, we also have that if $Z_1, \dots, Z_n \sim \text{Bernoulli}(p)$, then $Y = \sum_i^n Z_i \sim \text{Bin}(n, p)$.

That is, the sum of n "iid" Bernoulli(p) random variables gives a random variable with the $\text{Bin}(n, p)$ distribution.

QUICK REVIEW: AGGREGATED BINARY OUTCOMES

The logistic regression model can be used either for Bernoulli data (as we have done so far) or for data summarized as binomial counts (that is, aggregated counts).

In the aggregated form, the model is a **Binomial logistic regression**:

$$y_i|x_i \sim \text{Bin}(n_i, \pi_i); \quad \log \left(\frac{\pi_i}{1 - \pi_i} \right) = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \dots + \beta_p x_{ip}.$$

QUICK REVIEW: BERNOULLI VERSUS BINOMIAL OUTCOMES

Normally, for individual-level data, we would have

```
##   response predictor
## 1         0         3
## 2         0         3
## 3         1         2
## 4         1         1
## 5         1         2
## 6         0         3
```

```
M1 <- glm(response~predictor,data=Data,family=binomial)
summary(M1)
```

```
##
## Call:
## glm(formula = response ~ predictor, family = binomial, data = Data)
##
## Deviance Residuals:
##   Min       1Q   Median       3Q      Max
## -1.261  -1.105  -1.030   1.251   1.332
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)    0.1942     0.3609   0.538   0.591
## predictor2    -0.3660     0.4954  -0.739   0.460
## predictor3    -0.5508     0.5017  -1.098   0.272
##
## (Dispersion parameter for binomial family taken to be 1)
##
##   Null deviance: 138.27  on 99  degrees of freedom
## Residual deviance: 137.02  on 97  degrees of freedom
## AIC: 143.02
##
## Number of Fisher Scoring iterations: 4
```

QUICK REVIEW: BERNOULLI VERSUS BINOMIAL OUTCOMES

But we could also do the following with the aggregate level data instead

```
M2 <- glm(cbind(successes,n-successes)~predictor,data=Data_agg,family=binomial)
summary(M2)
```

```
##
## Call:
## glm(formula = cbind(successes, n - successes) ~ predictor, family = binomial,
##      data = Data_agg)
##
## Deviance Residuals:
## [1]  0  0  0
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)   0.1942     0.3609   0.538   0.591
## predictor2   -0.3660     0.4954  -0.739   0.460
## predictor3   -0.5508     0.5017  -1.098   0.272
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 1.2524e+00  on 2  degrees of freedom
## Residual deviance: 1.3323e-14  on 0  degrees of freedom
## AIC: 17.868
##
## Number of Fisher Scoring iterations: 2
```

Same results overall! Deviance and AIC are different because of the slightly different likelihood functions.

Note that some glm functions use **n** in the formula instead of **n-successes**.

ANOTHER EXAMPLE: BERKELEY ADMISSIONS

With that in mind, we can move forward to our next example.

We will use this next example to also start to illustrate how to fit Bayesian versions of generalized linear mixed effects models.

However, note that we can fit the frequentist versions of the same models using the `lme4` package.

In fall 1973, the University of California, Berkeley's graduate division admitted 44% of male applicants and 35% of female applicants.

School administrators were concerned about the potential for bias (and lawsuits!) and asked statistics professor Peter Bickel to examine the data more carefully.

We have a subset of the admissions data for 6 departments.

BERKELEY ADMISSIONS

```
library(rethinking)
data(UCBadmit)
d <- UCBadmit
detach(package:rethinking, unload=T)
library(brms)
d <-
  d%>%
  mutate(male=ifelse(applicant.gender=="male",1,0),
         dept_id = rep(1:6, each = 2))
d$successrate=d$admit/d$applications
sum(d$admit[d$male==1])/sum(d$applications[d$male==1])
```

```
## [1] 0.4451877
```

```
sum(d$admit[d$male==0])/sum(d$applications[d$male==0])
```

```
## [1] 0.3035422
```

We see in this subset of departments that roughly 45% of male applicants were admitted, while only 30% of female applicants were admitted.

BERKELEY ADMISSIONS

Because admissions decisions for graduate school are made on a departmental level (not at the school level), it makes sense to examine results of applications by department.

```
d[,c(1,2,3,4,7)]
```

##	dept	applicant.gender	admit	reject	dept_id
## 1	A	male	512	313	1
## 2	A	female	89	19	1
## 3	B	male	353	207	2
## 4	B	female	17	8	2
## 5	C	male	120	205	3
## 6	C	female	202	391	3
## 7	D	male	138	279	4
## 8	D	female	131	244	4
## 9	E	male	53	138	5
## 10	E	female	94	299	5
## 11	F	male	22	351	6
## 12	F	female	24	317	6

Hmm, what's going on here?

BERKELEY ADMISSIONS

Following McElreath's analysis in *Statistical Rethinking*, we start fitting a simple logistic regression model and examine diagnostic measures.

The model for department i and gender j with $n_{admit,ij}$ of n_{ij} applicants admitted is given as:

$$\begin{aligned} n_{admit,ij} &\sim \text{Binomial}(n_{ij}, \pi_{ij}) \\ \text{logit}(\pi_{ij}) &= \alpha + \beta \text{male}_{ij}, \end{aligned}$$

where $\alpha \sim N(0, 10)$ and $\beta \sim N(0, 1)$.

ANOTHER EXAMPLE:

```
adm1 <-  
  brm(data = d, family = binomial,  
    admit | trials(applications) ~ 1 + male ,  
    prior = c(prior(normal(0, 10), class = Intercept),  
      prior(normal(0, 1), class = b)),  
    iter = 2500, warmup = 500, cores = 2, chains = 2,  
    seed = 10)  
summary(adm1)
```

```
## Family: binomial  
## Links: mu = logit  
## Formula: admit | trials(applications) ~ 1 + male  
## Data: d (Number of observations: 12)  
## Draws: 2 chains, each with iter = 2500; warmup = 500; thin = 1;  
## total post-warmup draws = 4000  
##  
## Population-Level Effects:  
##      Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS  
## Intercept    -0.83     0.05   -0.93   -0.73 1.00     2207     2217  
## male          0.61     0.07    0.48    0.73 1.00     2837     2702  
##  
## Draws were sampled using sampling(NUTS). For each parameter, Bulk_ESS  
## and Tail_ESS are effective sample size measures, and Rhat is the potential  
## scale reduction factor on split chains (at convergence, Rhat = 1).
```

Here it appears male applicants have $e^{0.61} = 1.8$ (95% credible interval (1.6, 2.1)) times the odds of admission as female applicants.

ANOTHER EXAMPLE:

We can also put this on the probability scale.

```
post <- posterior_samples(adm1)
```

```
## Warning: Method 'posterior_samples' is deprecated. Please see ?as_draws for  
## recommended alternatives.
```

```
post %>%  
  mutate(p_admit_male = inv_logit_scaled(b_Intercept + b_male),  
         p_admit_female = inv_logit_scaled(b_Intercept),  
         diff_admit = p_admit_male - p_admit_female) %>%  
  summarise(`2.5%` = quantile(diff_admit, probs = .025),  
           `50%` = median(diff_admit),  
           `97.5%` = quantile(diff_admit, probs = .975))
```

```
##           2.5%           50%           97.5%  
## 1 0.1122369 0.1414303 0.1690868
```

Overall it appears the median probability of admission was 14 percentage points higher for males.

MODEL CHECKING

Here we take some posterior predictions and plot against the observed proportions in the data.

Here's the code to do that:

```
library(wesanderson)
library(dutchmasters)
library(ggplot2)
d <-
  d %>%
  mutate(case = factor(1:12))

p <-
  predict(adm1) %>%
  as_tibble() %>%
  bind_cols(d)

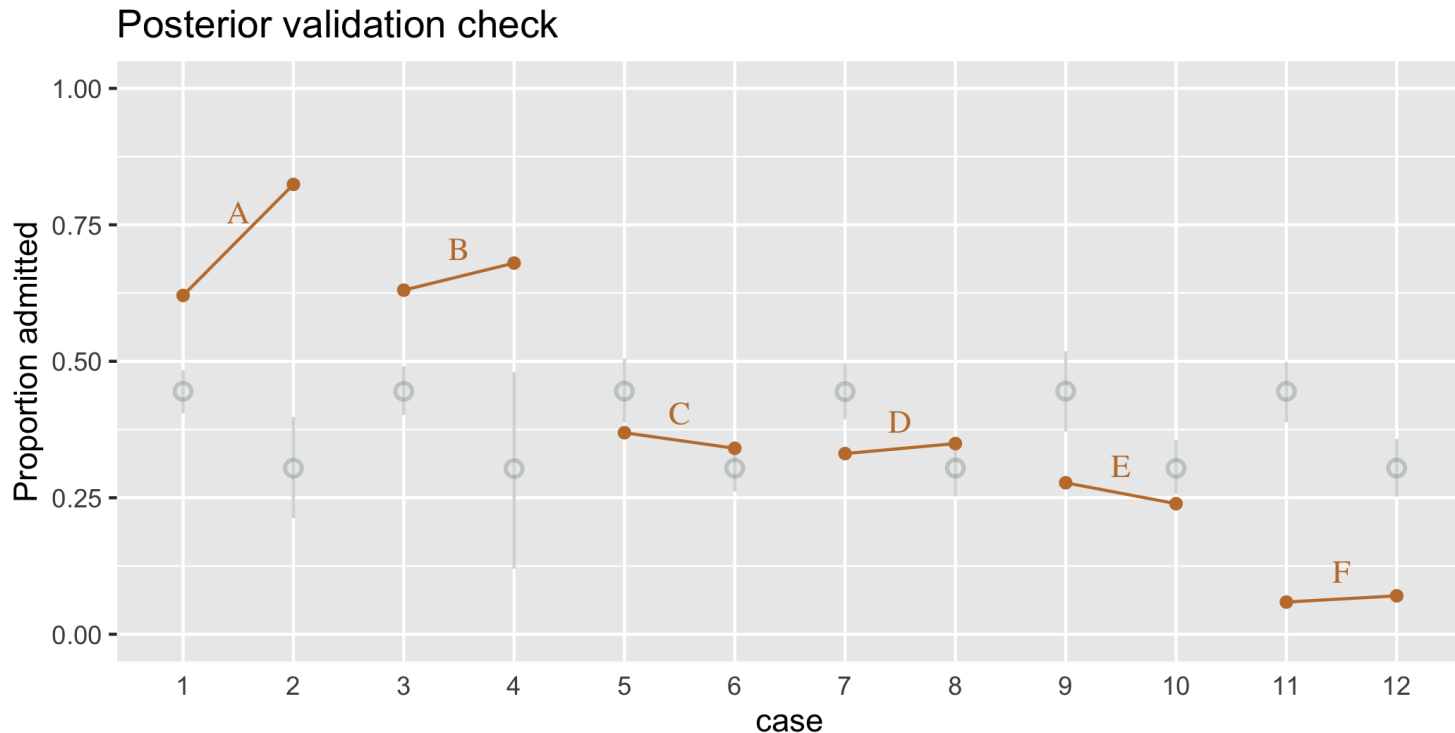
d_text <-
  d %>%
  group_by(dept) %>%
  summarise(case = mean(as.numeric(case)),
            admit = mean(admit / applications) + .05)
```

MODEL CHECKING

..and the rest of the code:

```
ggplot(data = d, aes(x = case, y = admit / applications)) +  
  geom_pointrange(data = p,  
    aes(y = Estimate / applications,  
        ymin = Q2.5 / applications ,  
        ymax = Q97.5 / applications),  
    color = wes_palette("Moonrise2")[1],  
    shape = 1, alpha = 1/3) +  
  geom_point(color = wes_palette("Moonrise2")[2]) +  
  geom_line(aes(group = dept),  
    color = wes_palette("Moonrise2")[2]) +  
  geom_text(data = d_text,  
    aes(y = admit, label = dept),  
    color = wes_palette("Moonrise2")[2],  
    family = "serif") +  
  coord_cartesian(ylim = 0:1) +  
  labs(y = "Proportion admitted",  
    title = "Posterior validation check") +  
  theme(axis.ticks.x = element_blank())
```

MODEL CHECKING



The orange lines connect observed proportions admitted in each department (odd numbers indicate males; even females).

The grey circles indicate point and interval estimates of the model-predicted proportion admitted. Clearly the model fits the data poorly.

VARYING / RANDOM INTERCEPTS

Based on the plot, we have some big departmental differences.

Let's specify department as a random effect in the model.

$$n_{admit,ij} \sim \text{Binomial}(n_{ij}, \pi_{ij})$$

$$\text{logit}(\pi_{ij}) = \alpha_{0i} + \beta \text{male}_{ij}$$

$$\alpha_{0i} \sim N(\alpha, \sigma^2); \quad \sigma^2 \sim \text{HalfCauchy}(0, 1)$$

$$\alpha \sim N(0, 10) \quad \text{and} \quad \beta \sim N(0, 1).$$

VARYING / RANDOM INTERCEPTS

```
adm2 <-  
  brm(data = d, family = binomial,  
    admit | trials(applications) ~ 1 + male + (1 | dept_id),  
    prior = c(prior(normal(0, 10), class = Intercept),  
              prior(normal(0, 1), class = b),  
              prior(cauchy(0, 1), class = sd)),  
    iter = 4500, warmup = 500, chains = 3, cores = 3,  
    seed = 13,  
    control = list(adapt_delta = 0.99))
```

VARYING / RANDOM INTERCEPTS

```
## Compiling Stan program...

## Start sampling

## Inference for Stan model: 7cd9f8491c0709228f6fd6f08ac18ae2.
## 3 chains, each with iter=4500; warmup=500; thin=1;
## post-warmup draws per chain=4000, total post-warmup draws=12000.
##
##               mean se_mean   sd   2.5%    25%    50%    75%   97.5%
## b_Intercept   -0.60    0.01 0.61   -1.81   -0.95   -0.59   -0.24    0.61
## b_male        -0.10    0.00 0.08   -0.26   -0.15   -0.10   -0.04    0.07
## sd_dept_id__Intercept 1.39    0.01 0.54    0.76    1.04    1.26    1.59    2.79
## r_dept_id[1,Intercept] 1.27    0.01 0.61    0.04    0.92    1.27    1.63    2.50
## r_dept_id[2,Intercept] 1.23    0.01 0.61    0.00    0.87    1.22    1.58    2.46
## r_dept_id[3,Intercept] 0.01    0.01 0.61   -1.21   -0.34    0.02    0.37    1.25
## r_dept_id[4,Intercept] -0.02    0.01 0.61   -1.24   -0.37   -0.02    0.34    1.22
## r_dept_id[5,Intercept] -0.46    0.01 0.61   -1.70   -0.82   -0.46   -0.10    0.77
## r_dept_id[6,Intercept] -2.01    0.01 0.62   -3.26   -2.36   -2.00   -1.64   -0.77
## lp__         -62.06    0.05 2.48  -67.82  -63.47  -61.69  -60.27  -58.22
##               n_eff Rhat
## b_Intercept   2125    1
## b_male        4830    1
## sd_dept_id__Intercept 1813    1
## r_dept_id[1,Intercept] 2124    1
## r_dept_id[2,Intercept] 2133    1
## r_dept_id[3,Intercept] 2125    1
## r_dept_id[4,Intercept] 2124    1
## r_dept_id[5,Intercept] 2148    1
## r_dept_id[6,Intercept] 2224    1
## lp__         2701    1
##
## Samples were drawn using NUTS(diag_e) at Mon Oct  4 15:18:28 2021.
## For each parameter, n_eff is a crude measure of effective sample size,
## and Rhat is the potential scale reduction factor on split chains (at
## convergence, Rhat=1).
```

In this model we see no evidence of a difference in admissions probabilities by gender though we do see big departmental variability.

RANDOM SLOPES?

How about random slopes for gender by department?

```
adm3 <-  
  brm(data = d, family = binomial,  
    admit | trials(applications) ~ 1 + male + (1 + male | dept_id),  
    prior = c(prior(normal(0, 10), class = Intercept),  
              prior(normal(0, 1), class = b),  
              prior(cauchy(0, 1), class = sd),  
              prior(lkj(2), class = cor)),  
    iter = 5000, warmup = 1000, chains = 4, cores = 4,  
    seed = 13,  
    control = list(adapt_delta = .99,  
                   max_treedepth = 12))  
adm3$fit
```

RANDOM SLOPES?

```
## Compiling Stan program...

## Start sampling

## Inference for Stan model: 22d21443d46776fcfd494514a0b3ce32.
## 4 chains, each with iter=5000; warmup=1000; thin=1;
## post-warmup draws per chain=4000, total post-warmup draws=16000.
##
##
```

	mean	se_mean	sd	2.5%	25%	50%	75%
## b_Intercept	-0.51	0.01	0.68	-1.84	-0.91	-0.50	-0.11
## b_male	-0.16	0.00	0.22	-0.61	-0.29	-0.15	-0.03
## sd_dept_id__Intercept	1.56	0.01	0.57	0.86	1.17	1.43	1.78
## sd_dept_id__male	0.46	0.00	0.23	0.15	0.31	0.42	0.56
## cor_dept_id__Intercept__male	-0.33	0.00	0.34	-0.86	-0.59	-0.36	-0.10
## r_dept_id[1,Intercept]	1.79	0.01	0.71	0.43	1.36	1.78	2.22
## r_dept_id[2,Intercept]	1.25	0.01	0.72	-0.16	0.80	1.23	1.68
## r_dept_id[3,Intercept]	-0.13	0.01	0.68	-1.47	-0.53	-0.15	0.27
## r_dept_id[4,Intercept]	-0.11	0.01	0.68	-1.44	-0.51	-0.11	0.29
## r_dept_id[5,Intercept]	-0.62	0.01	0.68	-1.96	-1.02	-0.63	-0.21
## r_dept_id[6,Intercept]	-2.09	0.01	0.69	-3.47	-2.50	-2.08	-1.67
## r_dept_id[1,male]	-0.61	0.00	0.31	-1.28	-0.80	-0.59	-0.39
## r_dept_id[2,male]	-0.05	0.00	0.33	-0.71	-0.25	-0.05	0.15
## r_dept_id[3,male]	0.24	0.00	0.24	-0.22	0.08	0.22	0.38
## r_dept_id[4,male]	0.07	0.00	0.24	-0.41	-0.08	0.06	0.21
## r_dept_id[5,male]	0.27	0.00	0.26	-0.21	0.10	0.26	0.43
## r_dept_id[6,male]	0.04	0.00	0.31	-0.58	-0.15	0.04	0.23
## lp__	-65.53	0.07	3.72	-73.90	-67.78	-65.14	-62.84
##	97.5%	n_eff	Rhat				
## b_Intercept	0.83	3751	1				
## b_male	0.27	6301	1				
## sd_dept_id__Intercept	3.03	4867	1				
## sd_dept_id__male	1.01	5224	1				
## cor_dept_id__Intercept__male	0.41	9857	1				
## r_dept_id[1,Intercept]	3.20	3771	1				
## r_dept_id[2,Intercept]	2.68	4215	1				
## r_dept_id[3,Intercept]	1.20	3737	1				
## r_dept_id[4,Intercept]	1.23	3747	1				
## r_dept_id[5,Intercept]	0.72	3820	1				
## r_dept_id[6,Intercept]	-0.72	3962	1				
## r_dept_id[1,male]	-0.06	7500	1				
## r_dept_id[2,male]	0.63	11973	1				
## r_dept_id[3,male]	0.75	7256	1				
## r_dept_id[4,male]	0.56	6909	1				
## r_dept_id[5,male]	0.83	7388	1				
## r_dept_id[6,male]	0.65	10417	1				
## lp__	-59.40	3279	1				
##							

DIAGNOSTICS

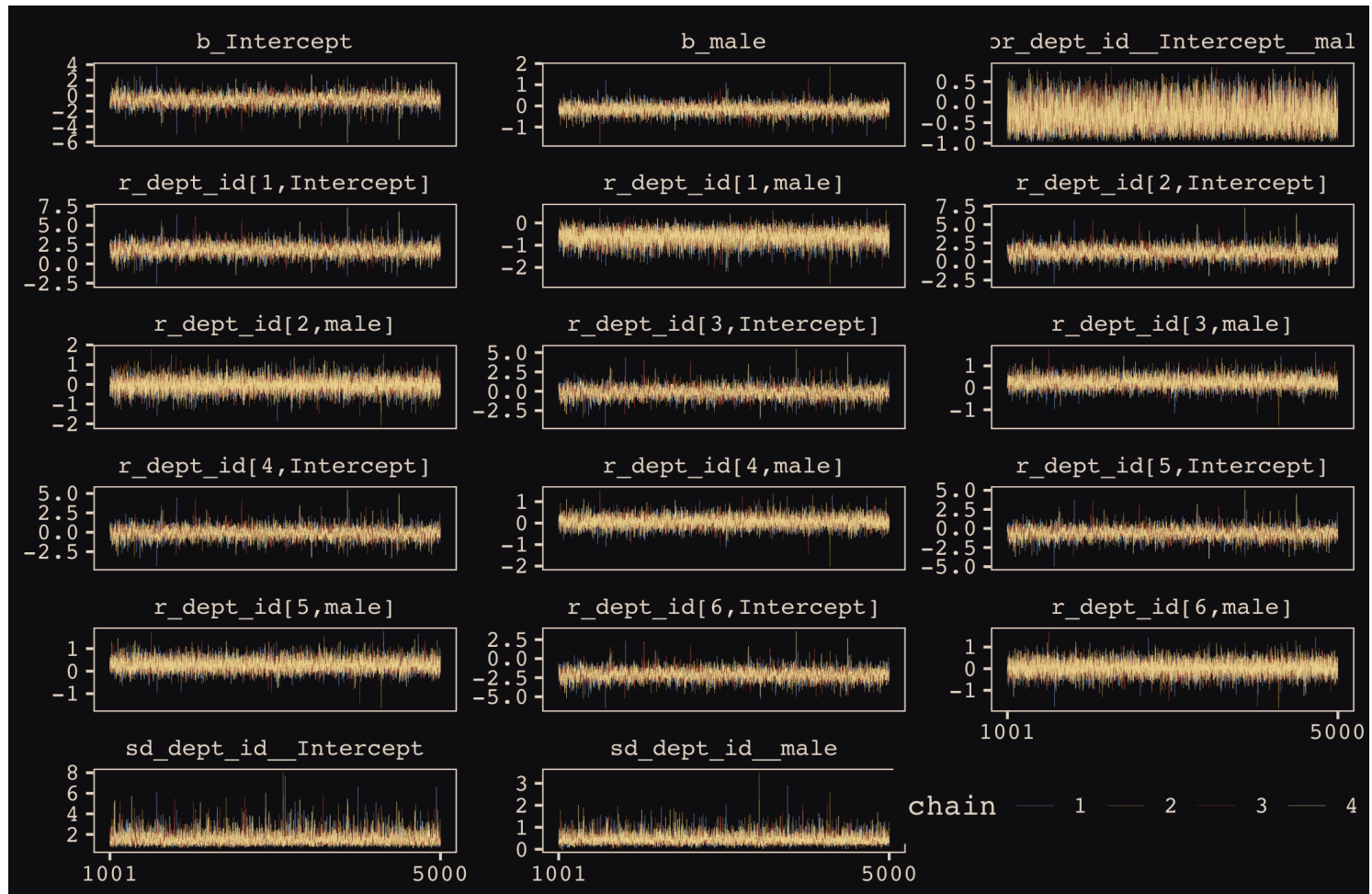
Before we get too excited let's take a quick look at the trace plots.

```
post <- posterior_samples(adm3, add_chain = T)
post <- post[,!is.element(colnames(post),c("lp_"))]

post %>%
  gather(key, value, -chain, -iter) %>%
  mutate(chain = as.character(chain)) %>%
  ggplot(aes(x = iter, y = value, group = chain, color = chain)) +
  geom_line(size = 1/15) +
  scale_color_manual(values = c("#80A0C7", "#B1934A", "#A65141", "#EEDA9D")) +
  scale_x_continuous(NULL, breaks = c(1001, 5000)) +
  ylab(NULL) +
  theme_pearl_earring +
  theme(legend.position = c(.825, .06),
        legend.direction = "horizontal") +
  facet_wrap(~key, ncol = 3, scales = "free_y")
```

DIAGNOSTICS

```
## Warning: Method 'posterior_samples' is deprecated. Please see ?as_draws for  
## recommended alternatives.
```

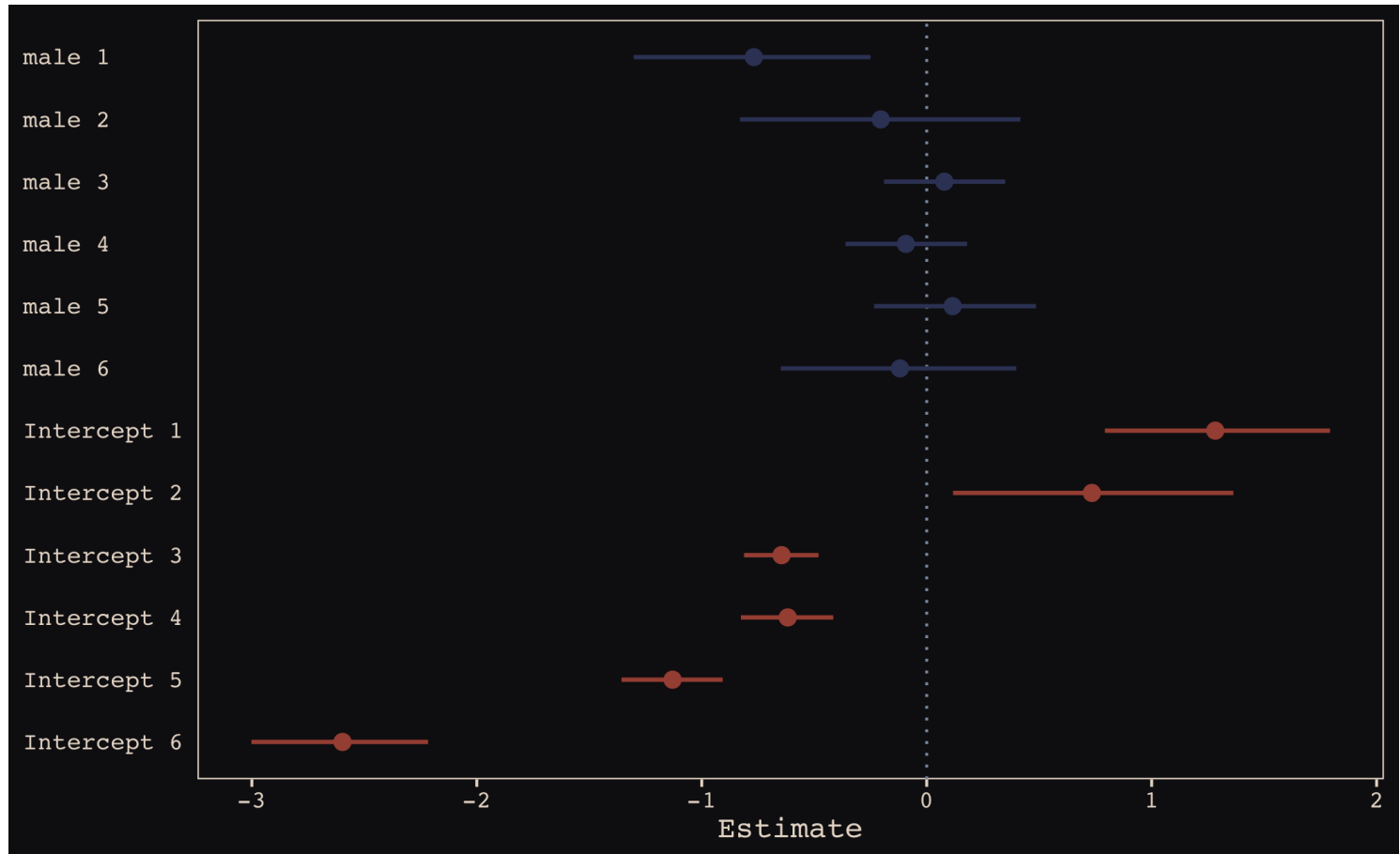


RANDOM EFFECTS

```
rbind(coef(adm3)$dept_id[, , 1],
      coef(adm3)$dept_id[, , 2]) %>%
  as_tibble() %>%
  mutate(param = c(paste("Intercept", 1:6), paste("male", 1:6)),
         reorder = c(6:1, 12:7)) %>%

  # plot
  ggplot(aes(x = reorder(param, reorder))) +
  geom_hline(yintercept = 0, linetype = 3, color = "#8B9DAF") +
  geom_pointrange(aes(ymin = Q2.5, ymax = Q97.5, y = Estimate, color = reorder < 7),
                 shape = 20, size = 3/4) +
  scale_color_manual(values = c("#394165", "#A65141")) +
  xlab(NULL) +
  coord_flip() +
  theme_pearl_earring +
  theme(legend.position = "none",
        axis.ticks.y = element_blank(),
        axis.text.y = element_text(hjust = 0))
```


RANDOM EFFECTS



We see much more variability in the random intercepts than in the random slopes.

WHAT HAPPENED AT BERKELEY?

What happened at Berkeley? It actually doesn't require too much sophisticated modeling.

What we are seeing is just Simpson's paradox.

```
d[,c(1,2,3,4,8)]
```

##	dept	applicant.gender	admit	reject	successrate
## 1	A	male	512	313	0.62060606
## 2	A	female	89	19	0.82407407
## 3	B	male	353	207	0.63035714
## 4	B	female	17	8	0.68000000
## 5	C	male	120	205	0.36923077
## 6	C	female	202	391	0.34064081
## 7	D	male	138	279	0.33093525
## 8	D	female	131	244	0.34933333
## 9	E	male	53	138	0.27748691
## 10	E	female	94	299	0.23918575
## 11	F	male	22	351	0.05898123
## 12	F	female	24	317	0.07038123

WHAT HAPPENED AT BERKELEY?

In the raw data, women had higher acceptance probabilities in 4 of the 6 departments.

However, the departments to which they applied in higher numbers were the departments that had lower overall acceptance rates.

What happened is that women were more likely to apply to departments like English, which have trouble supporting grad students, and they were less likely to apply to STEM departments, which had more plentiful funding for graduate students.

The men, on the other hand, were much more likely to apply to the STEM departments that had higher acceptance rates.

WHAT'S NEXT?

MOVE ON TO THE READINGS FOR THE NEXT MODULE!