E2 prediction

R. Cassano-Coleman

2025-10-02

This notebook analyzes prediction using Bayesian binomial generalized linear mixed effects models (GLMMs).

Set up

```
set.seed(15000)
```

```
data <- read_csv('../data/E1-E2-E4/prediction.csv', show_col_types = FALSE)
```

Convert variables to factors.

```
data %<>%
  mutate(exp_subject_id = as.factor(exp_subject_id),
         response = ifelse(response == 'Correct', TRUE, FALSE),
         scramble = factor(scramble, levels = c('8B', '2B', '1B')),
         Musician = factor(Musician, levels = c('Yes', 'No'))) %>%
  filter(!is.na(response))
```

Set the contrast for condition.

```
contrasts(data$scramble) <- contr.treatment(3)
print(contrasts(data$scramble))
```

```
##    2 3
## 8B 0 0
## 2B 1 0
## 1B 0 1
```

Set the musician/non-musician contrast.

```
contrasts(data$Musician) <- c(-1,1)
print(contrasts(data$Musician))
```

```
##     [,1]
## Yes   -1
## No     1
```

Main analysis

Priors

Priors are expressed in log(odds) space.

Intercept: Given that chance is 50%, we assume that participants will perform somewhere between chance and ceiling. We expect the center of the distribution of accuracy to be somewhere around 75% or 80%. If we use a center of 80% and an SD of 1, 95% of the values fall between 35.1% and 96.7%.

```
prior_intercept <- set_prior('normal(log(0.8 / (1 - 0.8)), 1)', class = 'Intercept')
```

Group: We might expect musicians to do slightly better than non-musicians, on average.

In this range, a difference in 0.25 log odds gives us about a 5% decrease in accuracy.

```
prior_mus <- set_prior('normal(-0.25, 1)', coef = 'Musician1')
```

Scramble: We expect performance to improve as scramble level decreases. If we code 8B as reference level, then we expect 8B > 2B and 8B > 1B.

Since we're keeping the musician slope at SD = 1, we'll keep these (and the interactions) at SD = 1. This seems to be a pretty weak prior.

```
prior_scramble2B <- set_prior('normal(-0.1, 1)', coef = 'scramble2')
prior_scramble1B <- set_prior('normal(-0.2, 1)', coef = 'scramble3')
```

Interaction: We expect no interaction between group and scramble.

```
prior_int2B <- set_prior('normal(0, 1)', coef = 'Musician1:scramble2')
prior_int1B <- set_prior('normal(0, 1)', coef = 'Musician1:scramble3')
```

Random slope for subjects: *Leave this as default for now, may update.*

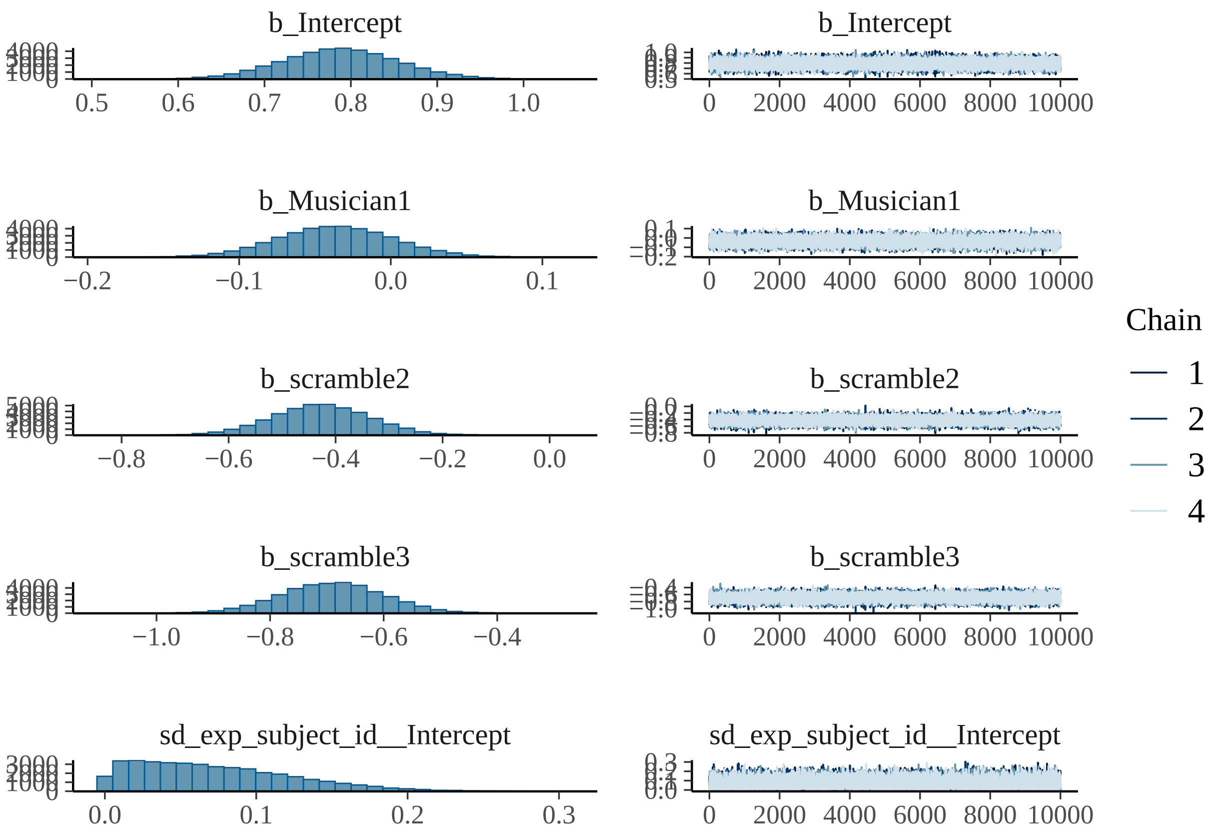
Main model with group and condition

```
get_prior(response ~ Musician + scramble + (1 | exp_subject_id), data = data)
```

```
##                  prior     class      coef          group resp dpar nlpar lb ub
##                 (flat)         b
##                 (flat)         b Musician1
##                 (flat)         b scramble2
##                 (flat)         b scramble3
##   student_t(3, 0, 2.5) Intercept
##   student_t(3, 0, 2.5)        sd                                            0
##   student_t(3, 0, 2.5)        sd           exp_subject_id                   0
##   student_t(3, 0, 2.5)        sd Intercept exp_subject_id                   0
##   student_t(3, 0, 2.5)     sigma                                            0
##        source
##       default
##  (vectorized)
##  (vectorized)
##  (vectorized)
##       default
##       default
##  (vectorized)
##  (vectorized)
##       default
```

```
mus_scram <- brm(response ~ Musician + scramble + (1 | exp_subject_id), data = data,
                 family = bernoulli(),
                 prior = c(prior_intercept, prior_mus,
                           prior_scramble2B, prior_scramble1B),
                 save_pars = save_pars(all = TRUE), iter = 20000, refresh = 0,
                 file = 'models/E2_mus_scram')
```

```
plot(mus_scram)
```




```r
print(summary(mus_scram, robust = TRUE), digits = 4)
```

```
##  Family: bernoulli
##   Links: mu = logit
## Formula: response ~ Musician + scramble + (1 | exp_subject_id)
##    Data: data (Number of observations: 3158)
##   Draws: 4 chains, each with iter = 20000; warmup = 10000; thin = 1;
##          total post-warmup draws = 40000
##
## Multilevel Hyperparameters:
## ~exp_subject_id (Number of levels: 105)
##               Estimate Est.Error l-95% CI u-95% CI   Rhat Bulk_ESS Tail_ESS
## sd(Intercept)   0.0646    0.0540   0.0032   0.1858 1.0002    14982    19427
##
## Regression Coefficients:
##           Estimate Est.Error l-95% CI u-95% CI   Rhat Bulk_ESS Tail_ESS
## Intercept   0.7878    0.0668   0.6574   0.9199 1.0000    49314    27679
## Musician1  -0.0362    0.0381  -0.1101   0.0390 1.0002    61568    29758
## scramble2  -0.4271    0.0907  -0.6059  -0.2488 1.0000    51826    31948
## scramble3  -0.6895    0.0904  -0.8671  -0.5151 1.0000    52026    30879
##
## 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).
```

```r
emm_mus_scram_s <- emmeans(mus_scram, specs = "scramble")
summary(emm_mus_scram_s)
```

```
##  scramble emmean lower.HPD upper.HPD
```

```
##  8B       0.7878    0.6542     0.917
##  2B       0.3610    0.2366     0.484
##  1B       0.0979   -0.0247     0.220
##
## Results are averaged over the levels of: Musician
## Point estimate displayed: median
## Results are given on the logit (not the response) scale.
## HPD interval probability: 0.95
```

```r
contrast(emm_mus_scram_s, method = "pairwise")
```

```
##  contrast estimate lower.HPD upper.HPD
##  8B - 2B     0.427     0.252     0.608
##  8B - 1B     0.690     0.514     0.866
##  2B - 1B     0.263     0.088     0.433
##
## Results are averaged over the levels of: Musician
## Point estimate displayed: median
## Results are given on the log odds ratio (not the response) scale.
## HPD interval probability: 0.95
```

```r
emm_mus_scram_ms <- emmeans(mus_scram, specs = c("Musician", "scramble"))
summary(emm_mus_scram_ms)
```

```
##  Musician scramble emmean lower.HPD upper.HPD
##  Yes      8B       0.8240    0.6738     0.975
##  No       8B       0.7516    0.5971     0.900
##  Yes      2B       0.3971    0.2552     0.546
##  No       2B       0.3252    0.1812     0.468
##  Yes      1B       0.1337   -0.0044     0.278
##  No       1B       0.0618   -0.0813     0.208
##
## Point estimate displayed: median
## Results are given on the logit (not the response) scale.
## HPD interval probability: 0.95
```

```r
contrast(emm_mus_scram_ms, method = "pairwise")
```

```
##  contrast        estimate lower.HPD upper.HPD
##  Yes 8B - No 8B    0.0725   -0.0772     0.221
##  Yes 8B - Yes 2B   0.4271    0.2521     0.608
##  Yes 8B - No 2B    0.4992    0.2762     0.740
##  Yes 8B - Yes 1B   0.6895    0.5142     0.866
##  Yes 8B - No 1B    0.7619    0.5317     0.997
##  No 8B - Yes 2B    0.3549    0.1174     0.586
##  No 8B - No 2B     0.4271    0.2521     0.608
##  No 8B - Yes 1B    0.6170    0.3900     0.849
##  No 8B - No 1B     0.6895    0.5142     0.866
##  Yes 2B - No 2B    0.0725   -0.0772     0.221
##  Yes 2B - Yes 1B   0.2633    0.0880     0.433
##  Yes 2B - No 1B    0.3348    0.1098     0.570
##  No 2B - Yes 1B    0.1917   -0.0344     0.418
##  No 2B - No 1B     0.2633    0.0880     0.433
##  Yes 1B - No 1B    0.0725   -0.0772     0.221
##
## Point estimate displayed: median
```

```
## Results are given on the log odds ratio (not the response) scale.
## HPD interval probability: 0.95
```

Main effects

```r
main_BF <- describe_posterior(mus_scram,
                              estimate = "median", dispersion = TRUE,
                              ci = .95, ci_method = "HDI",
                              test = c("bayes_factor"))
print(main_BF, digits = 5)
```

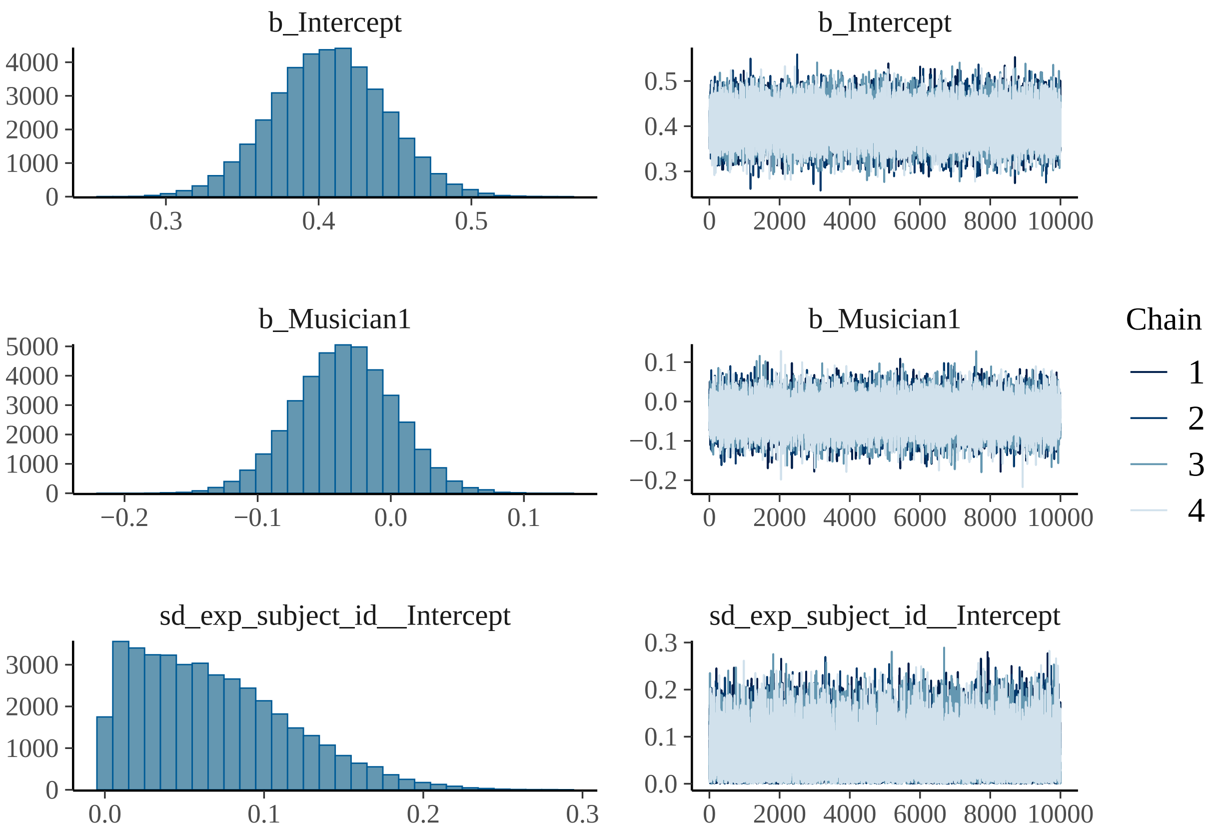
```
## Summary of Posterior Distribution
##
## Parameter   |   Median |     MAD |         95% CI |       BF |  Rhat |         ESS
## ---------------------------------------------------------------------------------
## (Intercept) |  0.78782 | 0.06679 | [ 0.65,  0.92] | 1.49e+11 | 1.000 | 49223.00000
## Musician1   | -0.03623 | 0.03811 | [-0.11,  0.04] |    0.058 | 1.000 | 61475.00000
## scramble2   | -0.42706 | 0.09068 | [-0.61, -0.25] |   393.78 | 1.000 | 51756.00000
## scramble3   | -0.68954 | 0.09045 | [-0.87, -0.51] | 8.71e+06 | 1.000 | 51961.00000
```

Strong evidence against a main effect of group.

To get the main effect of scramble level, fit the "null" model with group only to compare.

```
mus_only <- brm(response ~ Musician + (1 | exp_subject_id), data = data,
                family = bernoulli(),
                prior = c(prior_intercept, prior_mus),
                save_pars = save_pars(all = TRUE), iter = 20000, refresh = 0,
                file = 'models/E2_mus_only')
```

```
plot(mus_only)
```




```
print(summary(mus_only, robust = TRUE), digits = 4)
```

```
## Warning: There were 1 divergent transitions after warmup. Increasing
## adapt_delta above 0.8 may help. See
## http://mc-stan.org/misc/warnings.html#divergent-transitions-after-warmup

##  Family: bernoulli
##   Links: mu = logit
## Formula: response ~ Musician + (1 | exp_subject_id)
##    Data: data (Number of observations: 3158)
##   Draws: 4 chains, each with iter = 20000; warmup = 10000; thin = 1;
##          total post-warmup draws = 40000
##
## Multilevel Hyperparameters:
## ~exp_subject_id (Number of levels: 105)
##               Estimate Est.Error l-95% CI u-95% CI   Rhat Bulk_ESS Tail_ESS
## sd(Intercept)   0.0609    0.0511   0.0029   0.1775 1.0005    15080    20052
##
## Regression Coefficients:
```

```
##           Estimate Est.Error l-95% CI u-95% CI   Rhat Bulk_ESS Tail_ESS
## Intercept   0.4068    0.0370   0.3345   0.4791 1.0002    57241    28589
## Musician1  -0.0343    0.0374  -0.1087   0.0382 1.0001    61174    28656
##
## 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).
```

```
BF_scramble <- bayes_factor(mus_scram, mus_only)
```

```
## Iteration: 1
## Iteration: 2
## Iteration: 3
## Iteration: 4
## Iteration: 5
## Iteration: 1
## Iteration: 2
## Iteration: 3
## Iteration: 4
## Iteration: 5
```

```
print(BF_scramble)
```

```
## Estimated Bayes factor in favor of mus_scram over mus_only: 73896818176.45668
```

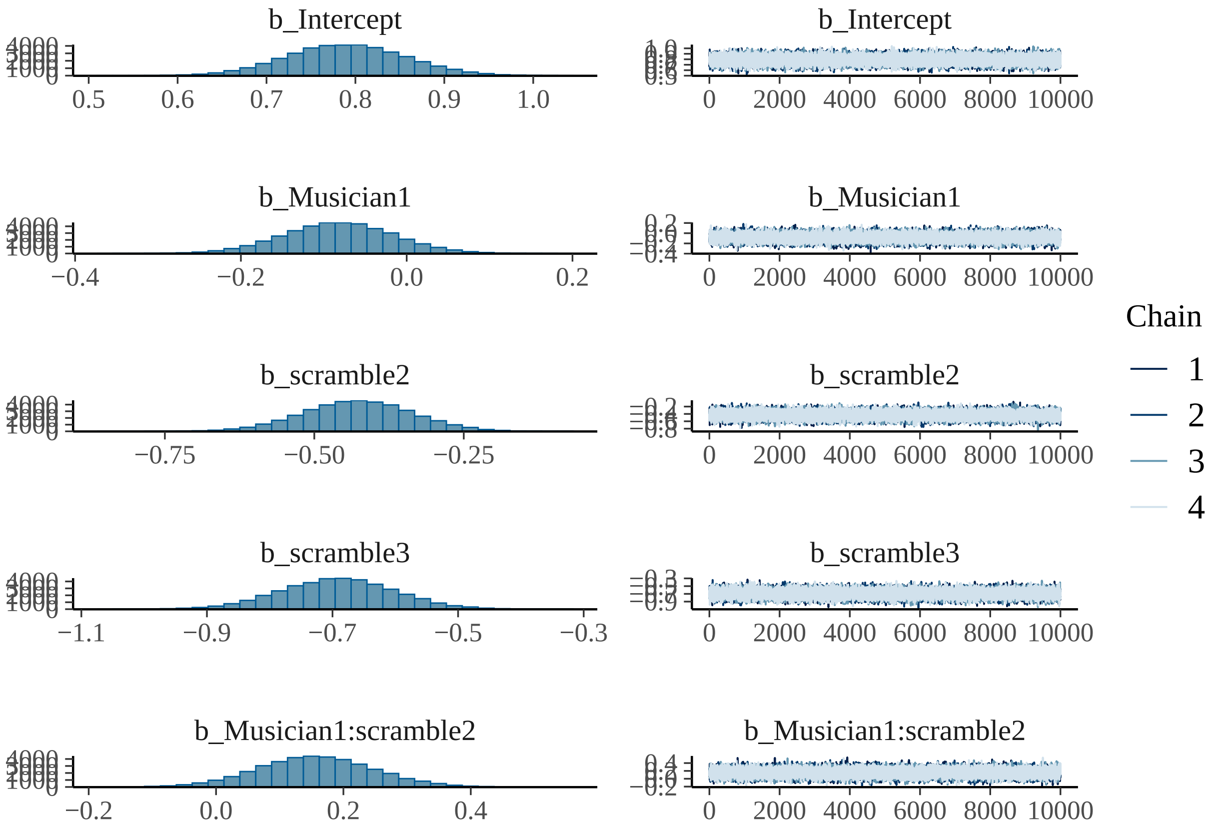
Very strong evidence for a main effect of scramble condition.

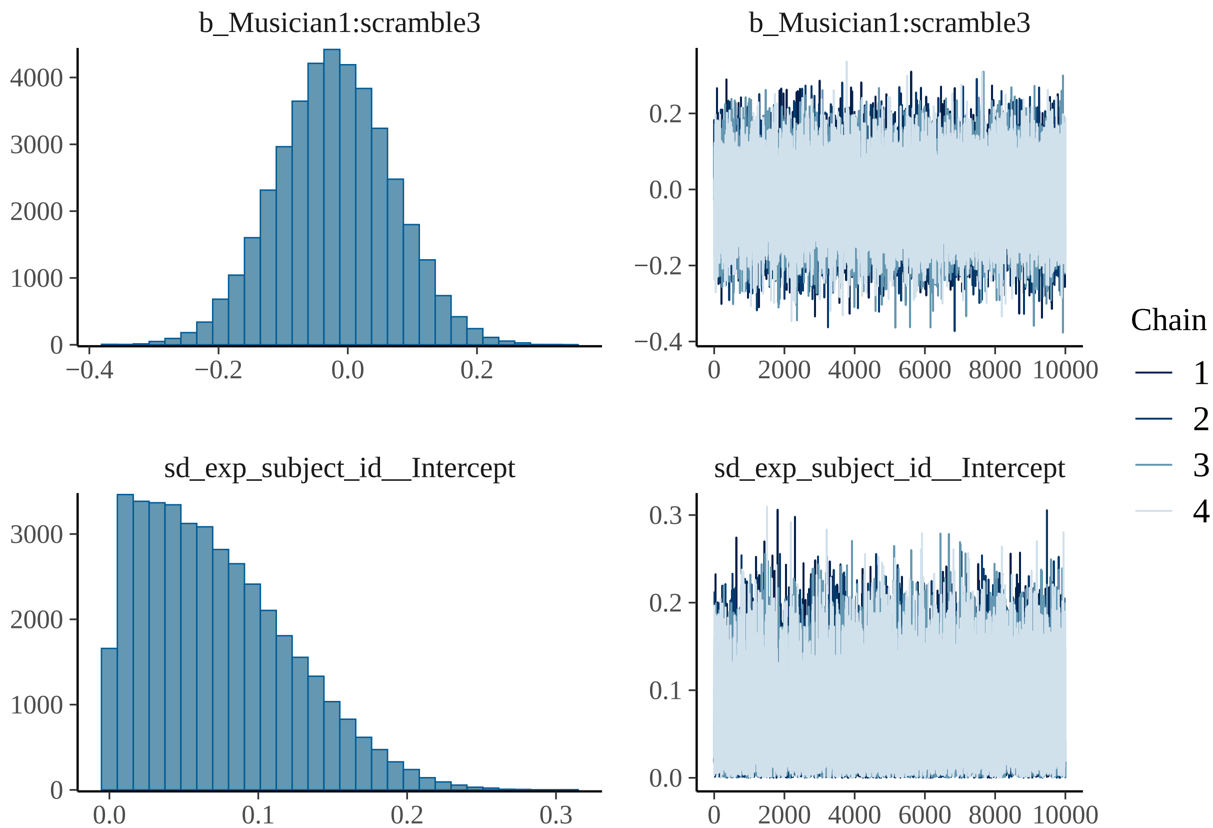
Interaction between group and condition?

Add an interaction between group and condition, and compare the model with the interaction to the one without.

```
mus_scram_int <- brm(response ~ Musician*scramble + (1 | exp_subject_id), data = data,
                  family = bernoulli(),
                  prior = c(prior_intercept, prior_mus,
                            prior_scramble2B, prior_scramble1B,
                            prior_int2B, prior_int1B),
                  save_pars = save_pars(all = TRUE), iter = 20000, refresh = 0,
                  file = 'models/E2_mus_scram_int')
```

```
plot(mus_scram_int)
```






```
print(summary(mus_scram_int, robust = TRUE), digits = 4)
```

```
## Warning: There were 2 divergent transitions after warmup. Increasing
## adapt_delta above 0.8 may help. See
## http://mc-stan.org/misc/warnings.html#divergent-transitions-after-warmup
```

```
##  Family: bernoulli
##   Links: mu = logit
## Formula: response ~ Musician * scramble + (1 | exp_subject_id)
##    Data: data (Number of observations: 3158)
##   Draws: 4 chains, each with iter = 20000; warmup = 10000; thin = 1;
##          total post-warmup draws = 40000
##
## Multilevel Hyperparameters:
## ~exp_subject_id (Number of levels: 105)
##               Estimate Est.Error l-95% CI u-95% CI   Rhat Bulk_ESS Tail_ESS
## sd(Intercept)   0.0644    0.0533   0.0032   0.1852 1.0001    16169    21035
##
## Regression Coefficients:
##                     Estimate Est.Error l-95% CI u-95% CI   Rhat Bulk_ESS
## Intercept             0.7894    0.0669   0.6609   0.9208 1.0004    49515
## Musician1            -0.0820    0.0663  -0.2133   0.0489 1.0001    32682
## scramble2            -0.4273    0.0912  -0.6067  -0.2502 1.0002    52790
## scramble3            -0.6914    0.0906  -0.8700  -0.5163 1.0001    51041
## Musician1:scramble2   0.1561    0.0901  -0.0188   0.3353 1.0000    36495
## Musician1:scramble3  -0.0211    0.0896  -0.1957   0.1545 1.0001    37304
##                     Tail_ESS
## Intercept              29028
```

```
## Musician1              28491
## scramble2              30412
## scramble3              31016
## Musician1:scramble2    31439
## Musician1:scramble3    31277
##
## 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).
```

```r
BF_int <- bayes_factor(mus_scram_int, mus_scram)
```

```
## Iteration: 1
## Iteration: 2
## Iteration: 3
## Iteration: 4
## Iteration: 5
## Iteration: 1
## Iteration: 2
## Iteration: 3
## Iteration: 4
## Iteration: 5
```

```r
print(BF_int)
```

```
## Estimated Bayes factor in favor of mus_scram_int over mus_scram: 0.07697
```

Strong evidence against an interaction between group and condition.

Figure 2B

Create a helper function for the conversion from log odds to probability.

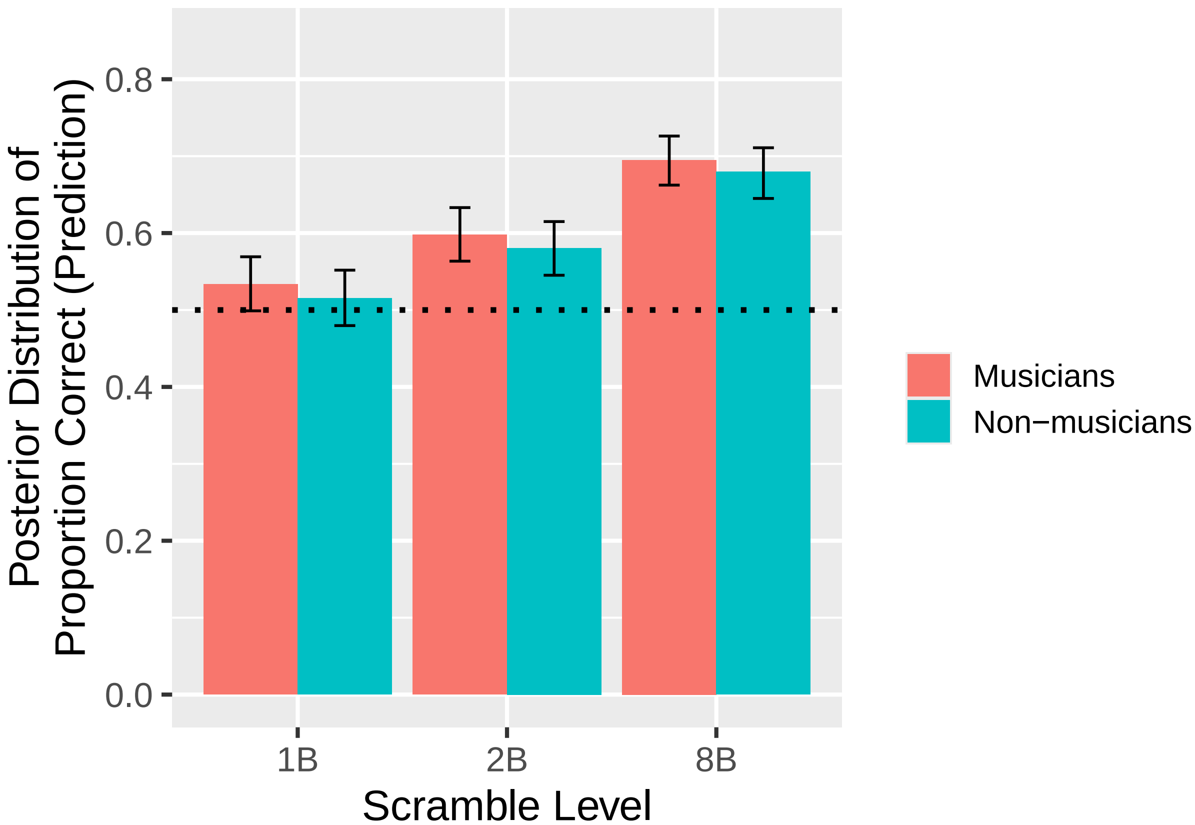
```
calculate_prob_from_logodds <- function(logodds) {
  return(exp(logodds) / (1 + exp(logodds)))
}
```

Visualize with posterior estimates and 95% CrI on the scale of accuracy.

```
posterior_est <- as.data.frame(emm_mus_scram_ms)
```

```
ggplot() +
  geom_col(aes(x = scramble, y = calculate_prob_from_logodds(emmean), fill = Musician),
           data = posterior_est,
           position = "dodge") +
  geom_errorbar(aes(x = scramble,
                    ymin = calculate_prob_from_logodds(lower.HPD),
                    ymax = calculate_prob_from_logodds(upper.HPD),
                    fill = Musician),
                data = posterior_est, position = position_dodge(width = 0.9), width = 0.2) +
  geom_hline(yintercept = 0.5, linetype = "dotted", color = "black", linewidth = 1) +
  theme_gray(base_size = 16) +
  scale_x_discrete(limits = rev) +
  ylim(0, 0.85) +
  xlab('Scramble Level') +
  ylab('Posterior Distribution of\nProportion Correct (Prediction)') +
  scale_fill_discrete(name="", labels=c('Musicians', 'Non-musicians')) +
  theme(legend.text = element_text(size = 12))
```

```
## Warning in geom_errorbar(aes(x = scramble, ymin =
## calculate_prob_from_logodds(lower.HPD), : Ignoring unknown aesthetics: fill
```




```
ggsave('../figures/Fig2B_prediction.png', width = 7, height = 5)
```

1B condition at chance?

There is technically no "right" answer, so performance in the 1B condition should be at chance.

```
data1B <- filter(data, scramble == '1B')
```

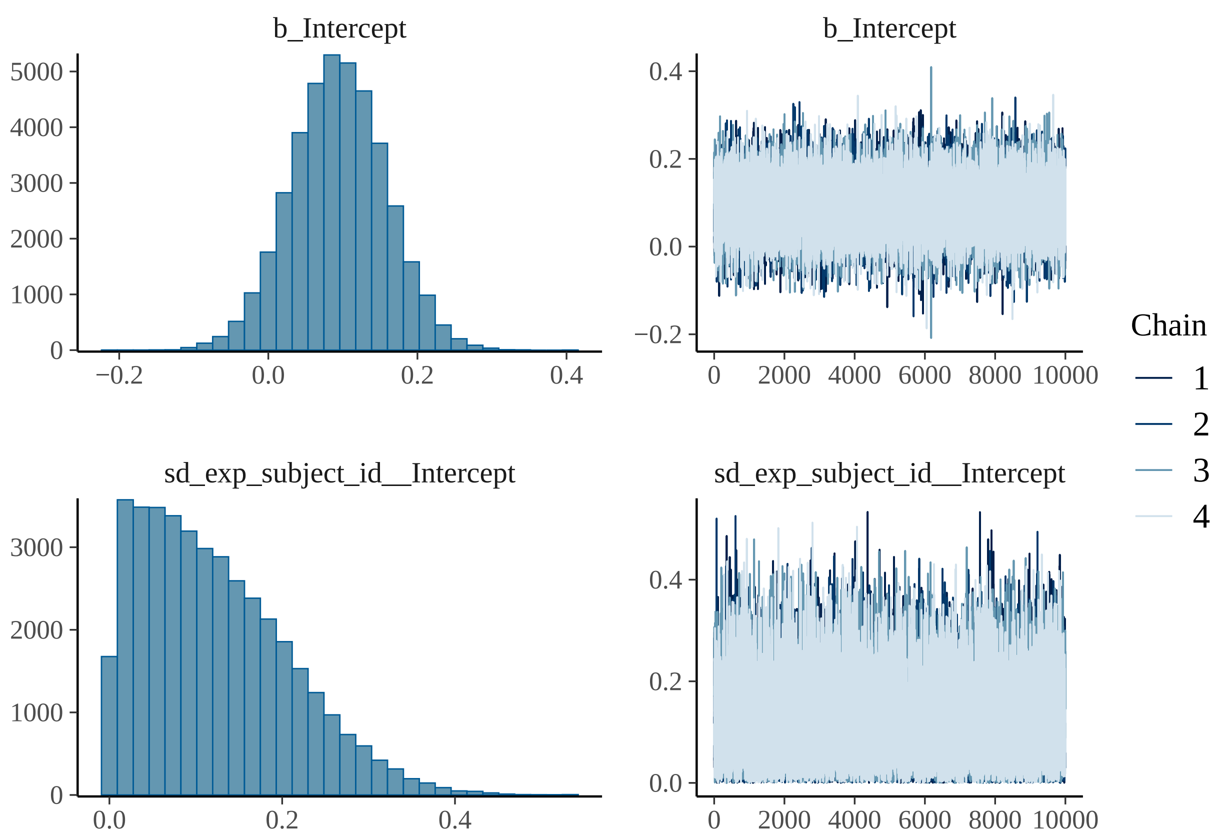
```
get_prior(response ~ 1 + (1 | exp_subject_id), data = data1B)
```

```
##                  prior     class      coef          group resp dpar nlpar lb ub
##  student_t(3, 0, 2.5) Intercept                                                 
##  student_t(3, 0, 2.5)        sd                                            0    
##  student_t(3, 0, 2.5)        sd           exp_subject_id                   0    
##  student_t(3, 0, 2.5)        sd Intercept exp_subject_id                   0    
##  student_t(3, 0, 2.5)     sigma                                            0    
##        source
##       default
##       default
##  (vectorized)
##  (vectorized)
##       default
```

(Leave the default prior for this intercept.)

```
only1B <- brm(response ~ 1 + (1 | exp_subject_id), data = data1B,
              family = bernoulli(),
              save_pars = save_pars(all = TRUE), iter = 20000, refresh = 0,
              file = 'models/E2_only1B')
```

```
plot(only1B)
```



```r
print(summary(only1B, robust = TRUE), digits = 4)
```

```
## Warning: There were 4 divergent transitions after warmup. Increasing
## adapt_delta above 0.8 may help. See
## http://mc-stan.org/misc/warnings.html#divergent-transitions-after-warmup
```

```
##  Family: bernoulli
##   Links: mu = logit
## Formula: response ~ 1 + (1 | exp_subject_id)
##    Data: data1B (Number of observations: 1054)
##   Draws: 4 chains, each with iter = 20000; warmup = 10000; thin = 1;
##          total post-warmup draws = 40000
##
## Multilevel Hyperparameters:
## ~exp_subject_id (Number of levels: 105)
##               Estimate Est.Error l-95% CI u-95% CI   Rhat Bulk_ESS Tail_ESS
## sd(Intercept)   0.1087    0.0906   0.0056   0.3166 1.0000    15060    20106
##
## Regression Coefficients:
##           Estimate Est.Error l-95% CI u-95% CI   Rhat Bulk_ESS Tail_ESS
## Intercept   0.0936    0.0634  -0.0299   0.2180 1.0000    59119    28163
##
## 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).
```

Is intercept different from 0?

```r
bf_pointnull(only1B, null = 0)
```

```
## Sampling priors, please wait...
```

```
## Bayes Factor (Savage-Dickey density ratio)
##
## Parameter   |    BF
## -------------------
## (Intercept) | 0.071
##
## * Evidence Against The Null: 0
```

There is strong evidence that performance in the 1B condition is at chance.

What if we just look at 8B and 2B?

The main thing here is to see if the interaction we see between group and condition (that we see visually) shows up when we take out 1B.

```
data_no1B <- filter(data, scramble != '1B')
```

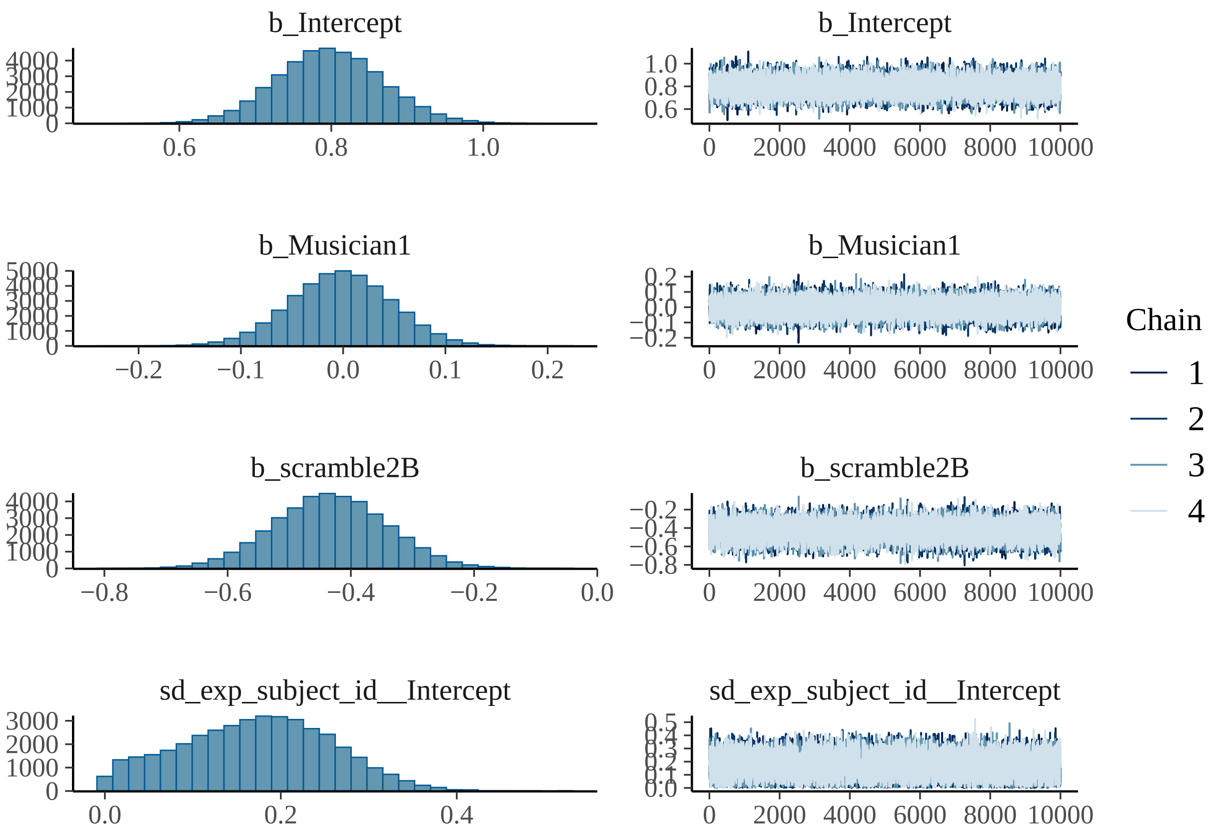
```
get_prior(response ~ Musician + scramble + (1 | exp_subject_id), data = data_no1B)
```

```
## Warning: contrasts dropped from factor scramble due to missing levels

##                 prior     class       coef          group resp dpar nlpar lb ub
##                (flat)         b
##                (flat)         b  Musician1
##                (flat)         b scramble2B
##  student_t(3, 0, 2.5) Intercept
##  student_t(3, 0, 2.5)        sd                                             0
##  student_t(3, 0, 2.5)        sd            exp_subject_id                   0
##  student_t(3, 0, 2.5)        sd  Intercept exp_subject_id                   0
##  student_t(3, 0, 2.5)     sigma                                             0
##        source
##       default
##  (vectorized)
##  (vectorized)
##       default
##       default
##  (vectorized)
##  (vectorized)
##       default
```

```
no1B <- brm(response ~ Musician + scramble + (1 | exp_subject_id), data = data_no1B,
            family = bernoulli(),
            prior = c(
              prior_intercept, prior_mus, set_prior('normal(-0.1, 1)', coef = 'scramble2B')
              ),
            save_pars = save_pars(all = TRUE), iter = 20000, refresh = 0,
            file = 'models/E2_no1B')
```

```
plot(no1B)
```




```r
print(summary(no1B, robust = TRUE), digits = 4)
```

```
## Warning: There were 3 divergent transitions after warmup. Increasing
## adapt_delta above 0.8 may help. See
## http://mc-stan.org/misc/warnings.html#divergent-transitions-after-warmup

##  Family: bernoulli
##   Links: mu = logit
## Formula: response ~ Musician + scramble + (1 | exp_subject_id)
##    Data: data_no1B (Number of observations: 2104)
##   Draws: 4 chains, each with iter = 20000; warmup = 10000; thin = 1;
##          total post-warmup draws = 40000
##
## Multilevel Hyperparameters:
## ~exp_subject_id (Number of levels: 105)
##               Estimate Est.Error l-95% CI u-95% CI   Rhat Bulk_ESS Tail_ESS
## sd(Intercept)   0.1740    0.0913   0.0142   0.3329 1.0002     9842    13005
##
## Regression Coefficients:
##             Estimate Est.Error l-95% CI u-95% CI   Rhat Bulk_ESS Tail_ESS
## Intercept     0.7972    0.0692   0.6631   0.9366 1.0000    42547    25453
## Musician1    -0.0018    0.0496  -0.1001   0.0945 1.0000    43313    28784
## scramble2B   -0.4330    0.0918  -0.6120  -0.2532 1.0000    53118    27256
##
## 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).
```

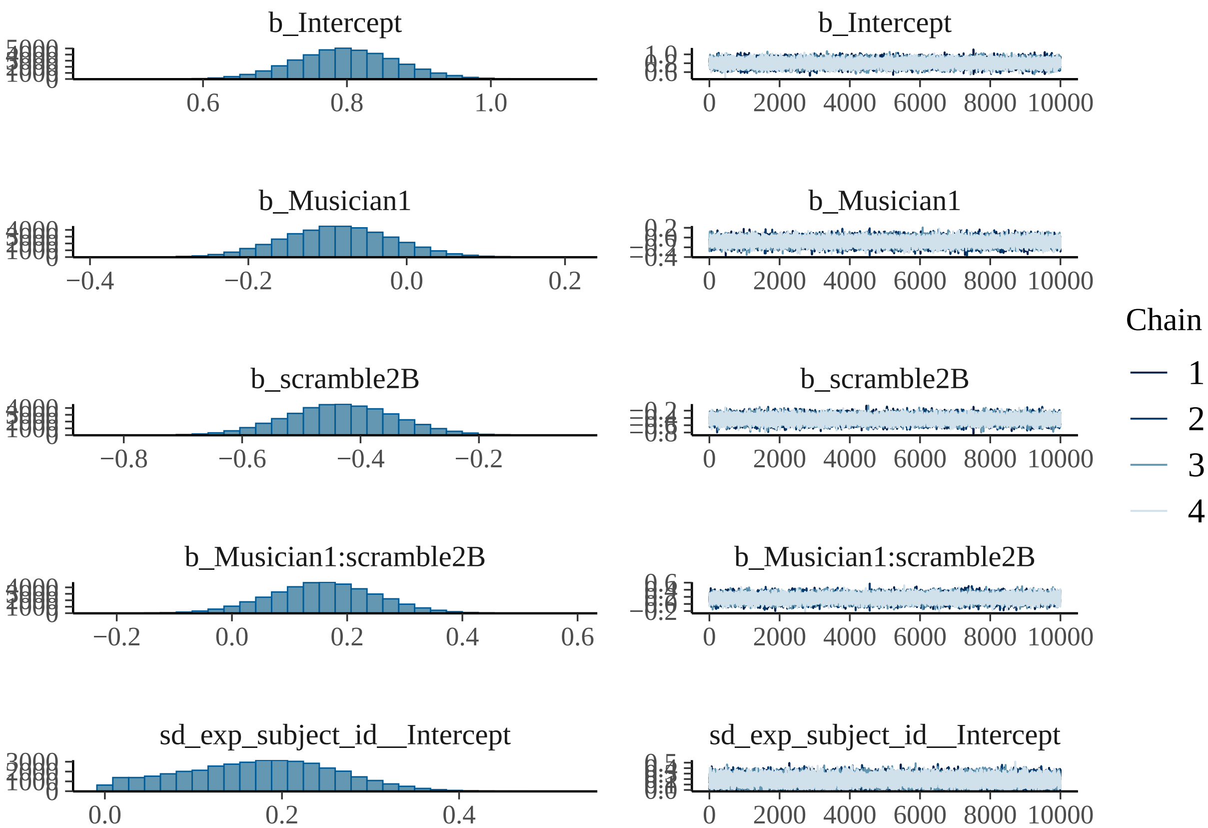
```r
get_prior(response ~ Musician*scramble + (1 | exp_subject_id), data = data_no1B)
```

```
## Warning: contrasts dropped from factor scramble due to missing levels
```

```
##                 prior     class                 coef          group resp dpar
##                (flat)         b
##                (flat)         b            Musician1
##                (flat)         b Musician1:scramble2B
##                (flat)         b           scramble2B
##  student_t(3, 0, 2.5) Intercept
##  student_t(3, 0, 2.5)        sd
##  student_t(3, 0, 2.5)        sd                      exp_subject_id
##  student_t(3, 0, 2.5)        sd            Intercept exp_subject_id
##  student_t(3, 0, 2.5)     sigma
##  nlpar lb ub       source
##                   default
##              (vectorized)
##              (vectorized)
##              (vectorized)
##                   default
##         0         default
##         0    (vectorized)
##         0    (vectorized)
##         0         default
```

```r
no1B_int <- brm(response ~ Musician*scramble + (1 | exp_subject_id), data = data_no1B,
                family = bernoulli(),
                prior = c(
                  prior_intercept, prior_mus,
                  set_prior('normal(-0.1, 1)', coef = 'scramble2B'),
                  set_prior('normal(0, 1)', coef = 'Musician1:scramble2B')
                  ),
                save_pars = save_pars(all = TRUE), iter = 20000, refresh = 0,
                file = 'models/E2_no1B_int')
```

```r
plot(no1B_int)
```




```r
print(summary(no1B_int, robust = TRUE), digits = 4)
```

```
##  Family: bernoulli
##   Links: mu = logit
## Formula: response ~ Musician * scramble + (1 | exp_subject_id)
##    Data: data_no1B (Number of observations: 2104)
##   Draws: 4 chains, each with iter = 20000; warmup = 10000; thin = 1;
##          total post-warmup draws = 40000
##
## Multilevel Hyperparameters:
## ~exp_subject_id (Number of levels: 105)
##               Estimate Est.Error l-95% CI u-95% CI   Rhat Bulk_ESS Tail_ESS
## sd(Intercept)   0.1757    0.0914   0.0141   0.3345 1.0003     9102    11556
##
## Regression Coefficients:
##                        Estimate Est.Error l-95% CI u-95% CI   Rhat Bulk_ESS
## Intercept                0.7982    0.0703   0.6629   0.9394 1.0001    48639
## Musician1               -0.0863    0.0701  -0.2205   0.0489 1.0000    33713
## scramble2B              -0.4327    0.0924  -0.6138  -0.2525 1.0001    61212
## Musician1:scramble2B     0.1595    0.0919  -0.0232   0.3387 1.0001    36039
##                        Tail_ESS
## Intercept                 30385
## Musician1                 30400
## scramble2B                29848
## Musician1:scramble2B      30212
##
## 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).
```

```r
BF_no1B_int <- bayes_factor(no1B_int, no1B)
```

```
## Iteration: 1
## Iteration: 2
## Iteration: 3
## Iteration: 4
## Iteration: 5
## Iteration: 1
## Iteration: 2
## Iteration: 3
## Iteration: 4
## Iteration: 5
```

```r
print(BF_no1B_int)
```

```
## Estimated Bayes factor in favor of no1B_int over no1B: 0.40770
```

Still moderate evidence against an interaction between group and condition.

Years of experience

Keep only the subjects for which we have years of experience data and average accuracy per condition.

```
yrs_exp <- data %>%
  filter(!is.na(yrs_mus_exp)) %>%
  group_by(exp_subject_id, scramble, yrs_mus_exp) %>%
  summarize(count = n(),
            n_correct = sum(response),
            accuracy = n_correct / count)
```

```
## `summarise()` has grouped output by 'exp_subject_id', 'scramble'. You can
## override using the `.groups` argument.
```

Priors

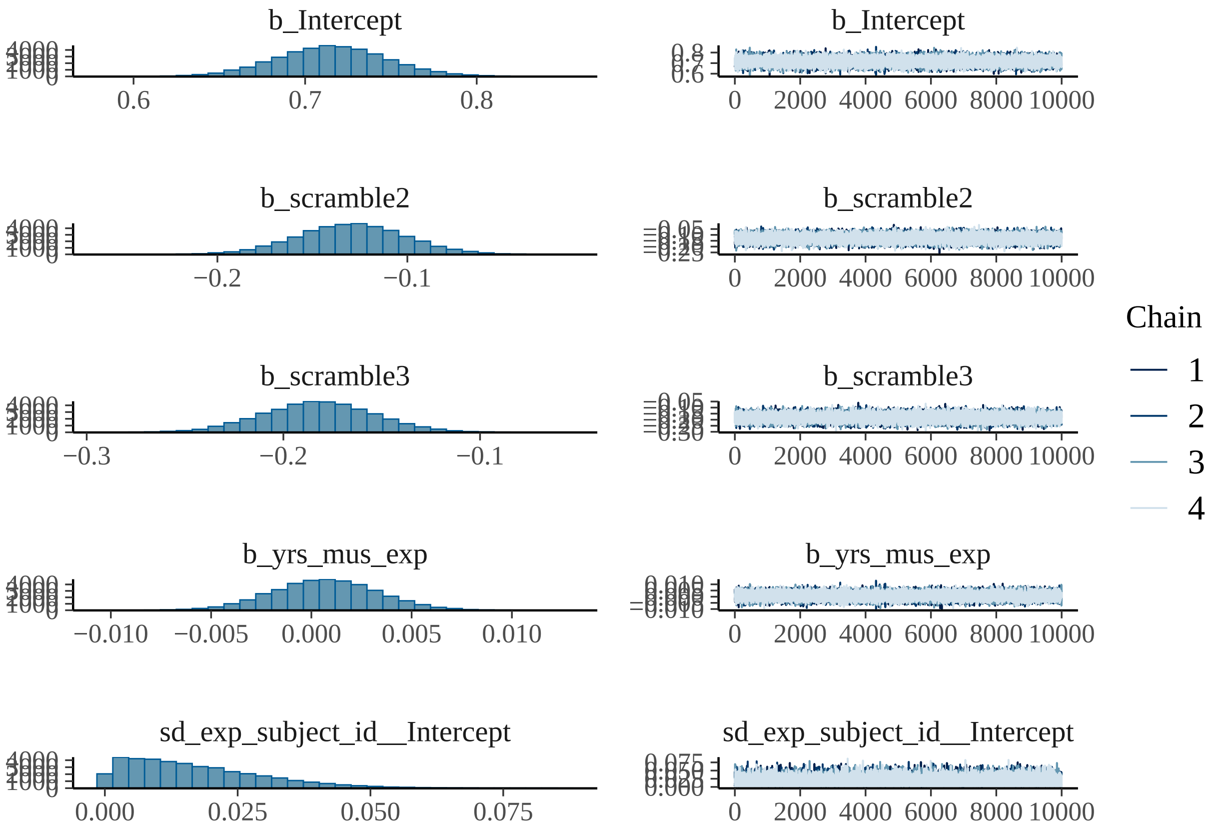
For this analysis, we're operating on the scale of accuracy. Because we don't see ceiling effects (i.e. participants aren't getting too close to perfect accuracy), a linear model is appropriate enough.

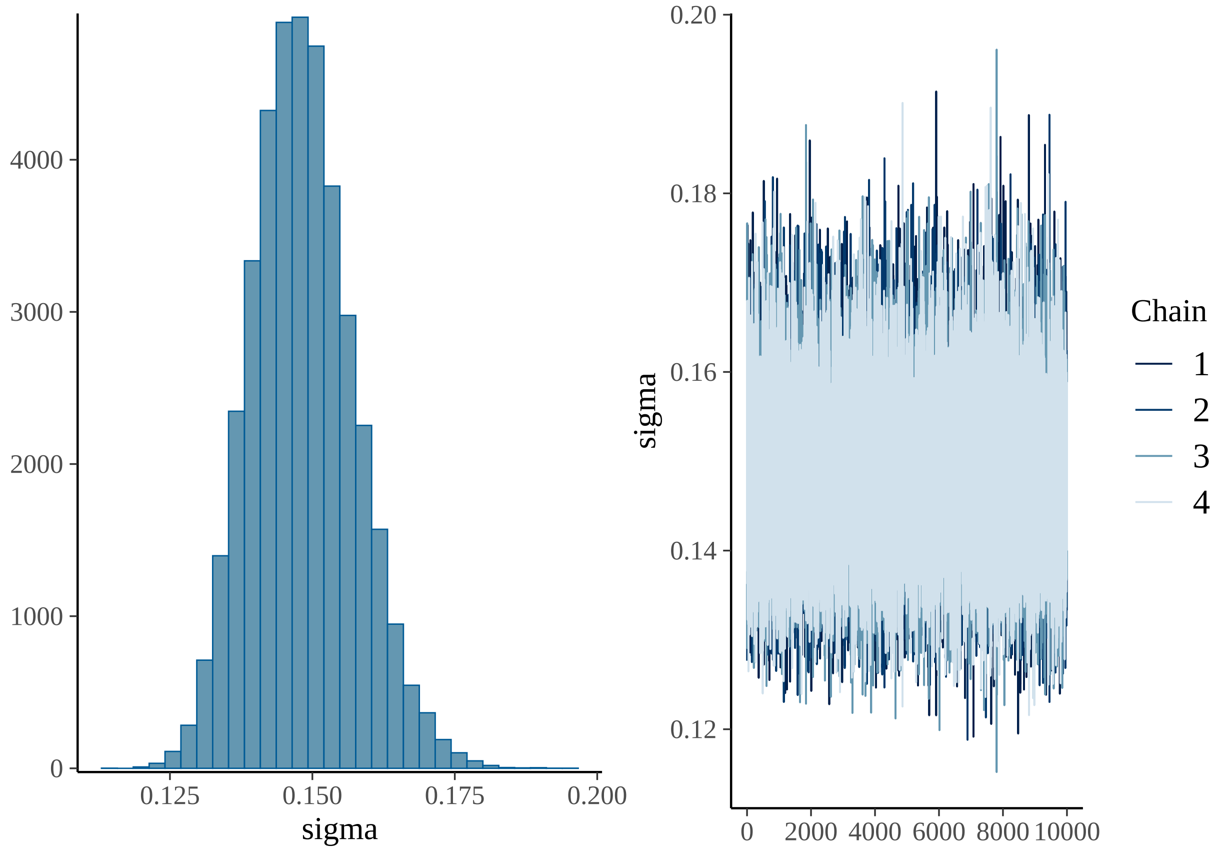
```
these_priors <- c(
  set_prior('normal(0.75, 0.1)', class = 'Intercept'),
  set_prior('normal(-0.1, 0.1)', coef = 'scramble2'),
  set_prior('normal(-0.2, 0.1)', coef = 'scramble3'),
  set_prior('normal(0, 0.1)', coef = 'yrs_mus_exp')
)
```

Main model

```
years_mus_scram <- brm(accuracy ~ scramble + yrs_mus_exp + (1|exp_subject_id), data = yrs_exp,
                       prior = these_priors,
                       save_pars = save_pars(all = TRUE), iter = 20000, refresh = 0,
                       file = 'models/E2_years')
```

```
plot(years_mus_scram)
```






```r
print(summary(years_mus_scram), robust = TRUE, digits = 5)
```

```
## Warning: There were 1 divergent transitions after warmup. Increasing
## adapt_delta above 0.8 may help. See
## http://mc-stan.org/misc/warnings.html#divergent-transitions-after-warmup
```

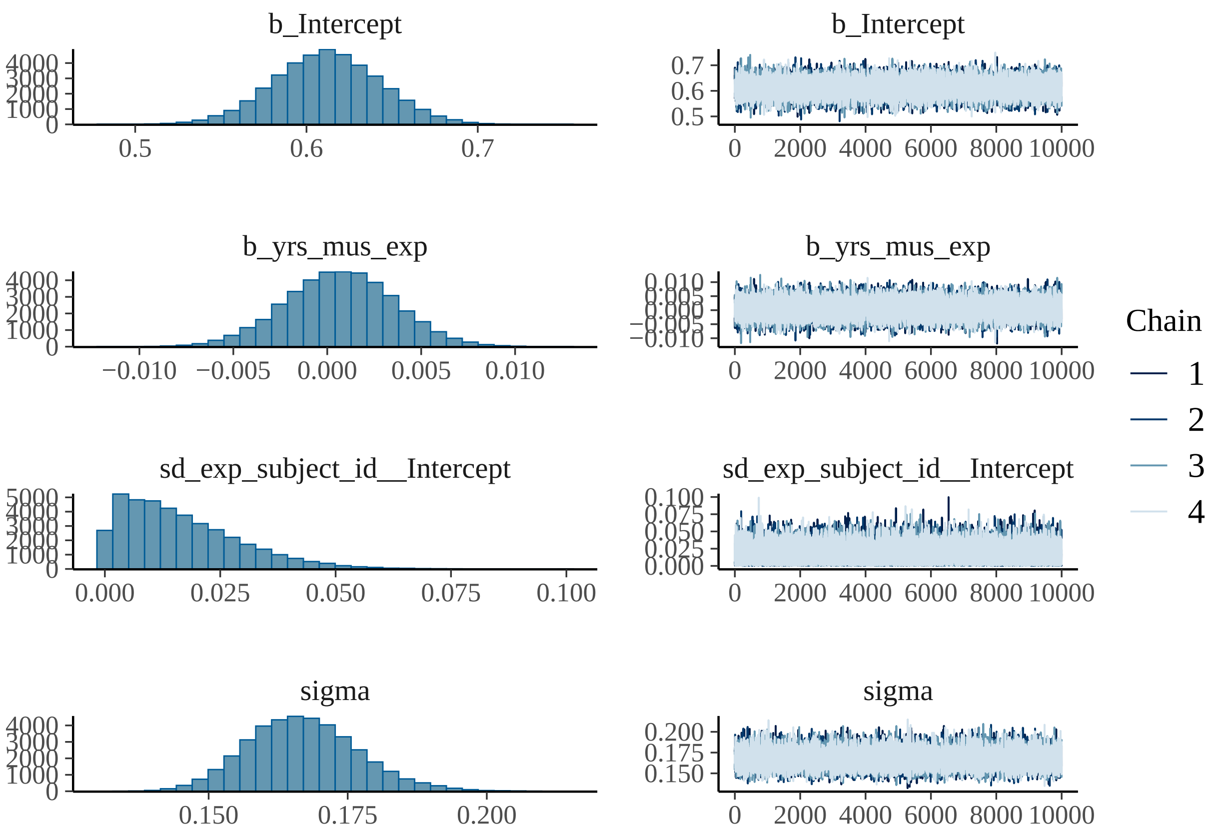
```
##  Family: gaussian
##   Links: mu = identity; sigma = identity
## Formula: accuracy ~ scramble + yrs_mus_exp + (1 | exp_subject_id)
##    Data: yrs_exp (Number of observations: 147)
##   Draws: 4 chains, each with iter = 20000; warmup = 10000; thin = 1;
##          total post-warmup draws = 40000
##
## Multilevel Hyperparameters:
## ~exp_subject_id (Number of levels: 49)
##               Estimate Est.Error l-95% CI u-95% CI    Rhat Bulk_ESS Tail_ESS
## sd(Intercept)  0.01711   0.01270  0.00073  0.04730 1.00002    22978    21010
##
## Regression Coefficients:
##             Estimate Est.Error l-95% CI u-95% CI    Rhat Bulk_ESS Tail_ESS
## Intercept    0.71541   0.03219  0.65259  0.77908 1.00001    70083    29714
## scramble2   -0.12932   0.02834 -0.18505 -0.07350 1.00008    59617    30976
## scramble3   -0.18216   0.02836 -0.23754 -0.12676 1.00000    58559    30749
## yrs_mus_exp  0.00069   0.00262 -0.00440  0.00582 1.00017    73077    29613
##
## Further Distributional Parameters:
##       Estimate Est.Error l-95% CI u-95% CI    Rhat Bulk_ESS Tail_ESS
## sigma  0.14832   0.00898  0.13204  0.16719 1.00004    68996    28545
```

```
##
## 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).
```

Null model (for plotting purposes)

```r
years_mus <- brm(accuracy ~ yrs_mus_exp + (1|exp_subject_id), data = yrs_exp,
                 prior = c(
                   set_prior('normal(0.75, 0.1)', class = 'Intercept'),
                   set_prior('normal(0, 0.1)', coef = 'yrs_mus_exp')),
                 save_pars = save_pars(all = TRUE), iter = 20000, refresh = 0,
                 file = 'models/E2_years_null')
```

```r
plot(years_mus)
```




```r
print(summary(years_mus, robust = TRUE), digits = 4)
```

```
## Warning: There were 2 divergent transitions after warmup. Increasing
## adapt_delta above 0.8 may help. See
## http://mc-stan.org/misc/warnings.html#divergent-transitions-after-warmup
```

```
##  Family: gaussian
##   Links: mu = identity; sigma = identity
## Formula: accuracy ~ yrs_mus_exp + (1 | exp_subject_id)
##    Data: yrs_exp (Number of observations: 147)
##   Draws: 4 chains, each with iter = 20000; warmup = 10000; thin = 1;
##          total post-warmup draws = 40000
##
```

```
## Multilevel Hyperparameters:
## ~exp_subject_id (Number of levels: 49)
##               Estimate Est.Error l-95% CI u-95% CI   Rhat Bulk_ESS Tail_ESS
## sd(Intercept)   0.0140    0.0123   0.0006   0.0467 1.0000    26265    20884
##
## Regression Coefficients:
##             Estimate Est.Error l-95% CI u-95% CI   Rhat Bulk_ESS Tail_ESS
## Intercept     0.6120    0.0311   0.5510   0.6734 1.0001    76248    29682
## yrs_mus_exp   0.0007    0.0029  -0.0051   0.0063 1.0001    73683    29633
##
## Further Distributional Parameters:
##       Estimate Est.Error l-95% CI u-95% CI   Rhat Bulk_ESS Tail_ESS
## sigma   0.1666    0.0098   0.1489   0.1881 1.0001    81109    29066
##
## 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).
```

```r
yrs_BF <- describe_posterior(years_mus_scram,
                             estimate = "median", dispersion = TRUE,
                             ci = .95, ci_method = "HDI",
                             test = c("bayes_factor"))
print(yrs_BF, digits = 4)
```

```
## Summary of Posterior Distribution
##
## Parameter   |  Median |    MAD |          95% CI |       BF |  Rhat |        ESS
## ---------------------------------------------------------------------------------
## (Intercept) |  0.7154 | 0.0319 | [ 0.65,  0.78] | 9.27e+25 | 1.000 | 69789.0000
## scramble2   | -0.1292 | 0.0283 | [-0.18, -0.07] | 1.20e+03 | 1.000 | 60096.0000
## scramble3   | -0.1821 | 0.0281 | [-0.24, -0.13] | 3.44e+04 | 1.000 | 58383.0000
## yrs_mus_exp |  0.0007 | 0.0026 | [ 0.00,  0.01] |    0.027 | 1.000 | 72944.0000
```

```r
yrs_null_BF <- describe_posterior(years_mus,
                                  estimate = "median", dispersion = TRUE,
                                  ci = .95, ci_method = "HDI",
                                  test = c("bayes_factor"))
print(yrs_null_BF, digits = 4)
```

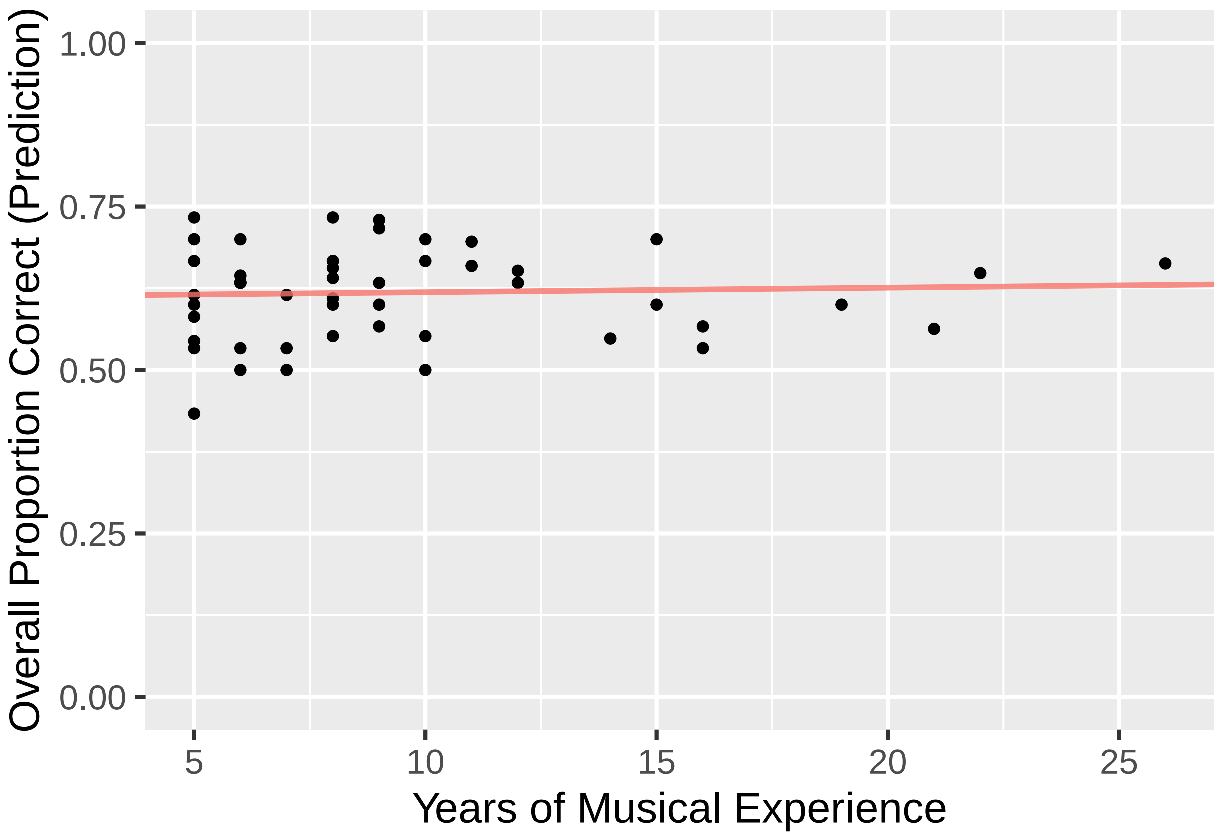
```
## Summary of Posterior Distribution
##
## Parameter   | Median |    MAD |        95% CI |       BF |  Rhat |        ESS
## ------------------------------------------------------------------------------
## (Intercept) | 0.6120 | 0.0311 | [ 0.55, 0.67] | 1.38e+22 | 1.000 | 76085.0000
## yrs_mus_exp | 0.0007 | 0.0029 | [ 0.00, 0.01] |    0.030 | 1.000 | 73809.0000
```

Strong evidence against an effect of years of musical experience.

Figure S1B

```
yrs_exp %>%
  group_by(exp_subject_id, yrs_mus_exp) %>%
  summarize(mean_acc = mean(accuracy)) %>%
  ggplot(aes(yrs_mus_exp, mean_acc)) +
  geom_point() +
  geom_abline(intercept = yrs_null_BF$Median[1], slope = yrs_null_BF$Median[2],
              color = '#F8766D', linewidth = 1, alpha = 0.8) +
  xlab('Years of Musical Experience') +
  ylab('Overall Proportion Correct (Prediction)') +
  scale_x_continuous(breaks = seq(5,30,5)) +
  scale_y_continuous(breaks = seq(0, 1, 0.1)) +
  ylim(0,1) +
  theme_gray(base_size = 16)
```

```
## `summarise()` has grouped output by 'exp_subject_id'. You can override using
## the `.groups` argument.
## Scale for y is already present. Adding another scale for y, which will replace
## the existing scale.
```




```
ggsave('../figures/FigS1B_prediction.png', width = 5, height = 5)
```