

# E2 prediction

R. Cassano-Coleman

2025-06-27

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

## Set up

```
set.seed(15000)

data <- read_csv('../data/prediction.csv')

## Rows: 3210 Columns: 6
## -- Column specification -----
## Delimiter: ","
## chr (3): response, scramble, Musician
## dbl (3): exp_subject_id, Trial_Nr, yrs_mus_exp
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

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
```

# 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.

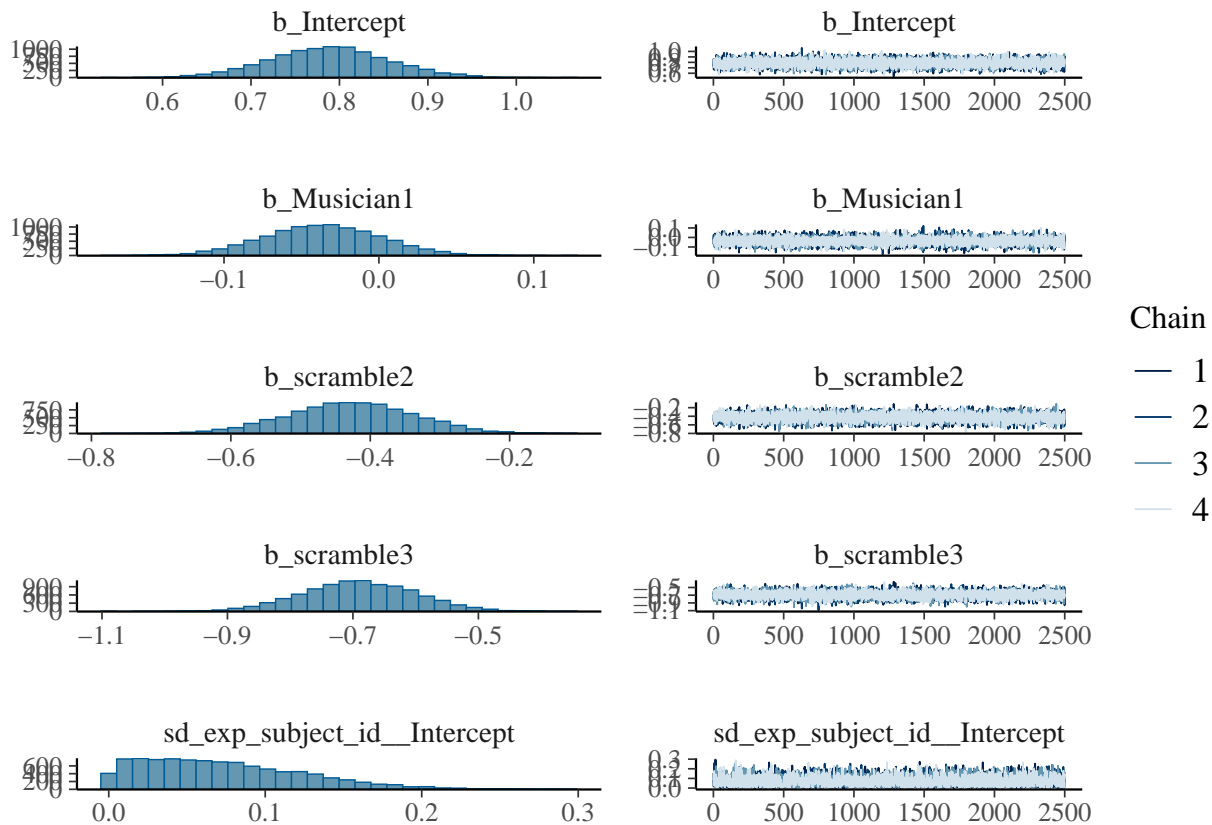
```
prior_int2B <- set_prior('normal(0, 1)', coef = 'Musician1:scramble2')  
prior_int1B <- set_prior('normal(0, 1)', coef = 'Musician1:scramble3')
```

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

## Main model with group and condition

```
mus_scam <- 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 = 5000,
  file = '../models/E2_mus_scam')
```

```
plot(mus_scam)
```



```
print(summary(mus_scam), 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 = 5000; warmup = 2500; thin = 1;
## total post-warmup draws = 10000
##
## Multilevel Hyperparameters:
## ~exp_subject_id (Number of levels: 105)
## Estimate Est.Error 1-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## sd(Intercept) 0.0735 0.0506 0.0029 0.1867 1.0005 3759 4232
##
## Regression Coefficients:
## Estimate Est.Error 1-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## Intercept 0.7882 0.0666 0.6571 0.9209 1.0010 9660 7323
```

```
## Musician1 -0.0362    0.0377 -0.1093    0.0373 1.0001    12143    6990
## scramble2 -0.4278    0.0912 -0.6079   -0.2528 1.0004    10714    7748
## scramble3 -0.6901    0.0902 -0.8667   -0.5139 1.0004    10339    7786
##
## 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).
```

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

```
##  scramble emmean lower.HPD upper.HPD
##  8B      0.7884    0.6555    0.918
##  2B      0.3604    0.2401    0.489
##  1B      0.0969   -0.0217    0.223
##
```

```
## 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
```

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

```
##  contrast estimate lower.HPD upper.HPD
##  8B - 2B      0.427      0.259      0.612
##  8B - 1B      0.691      0.512      0.864
##  2B - 1B      0.262      0.092      0.436
##
```

```
## 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
```

```
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      0.8243    0.6688    0.973
##  No       8B      0.7519    0.6030    0.896
##  Yes      2B      0.3968    0.2488    0.541
##  No       2B      0.3245    0.1835    0.467
##  Yes      1B      0.1328   -0.0103    0.276
##  No       1B      0.0611   -0.0905    0.200
##
```

```
## Point estimate displayed: median
## Results are given on the logit (not the response) scale.
## HPD interval probability: 0.95
```

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

```
##  contrast          estimate lower.HPD upper.HPD
##  Yes 8B - No 8B      0.0725   -0.0739    0.219
##  Yes 8B - Yes 2B     0.4272    0.2588    0.612
##  Yes 8B - No 2B      0.5001    0.2819    0.743
##  Yes 8B - Yes 1B     0.6907    0.5121    0.864
##  Yes 8B - No 1B      0.7626    0.5281    0.999
##  No 8B - Yes 2B      0.3541    0.1297    0.595
```

##	No 8B - No 2B	0.4272	0.2588	0.612
##	No 8B - Yes 1B	0.6162	0.3880	0.842
##	No 8B - No 1B	0.6907	0.5121	0.864
##	Yes 2B - No 2B	0.0725	-0.0739	0.219
##	Yes 2B - Yes 1B	0.2616	0.0920	0.436
##	Yes 2B - No 1B	0.3351	0.0994	0.553
##	No 2B - Yes 1B	0.1905	-0.0276	0.417
##	No 2B - No 1B	0.2616	0.0920	0.436
##	Yes 1B - No 1B	0.0725	-0.0739	0.219

##

## Point estimate displayed: median

## Results are given on the log odds ratio (not the response) scale.

## HPD interval probability: 0.95

## Main effects

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

```
## Warning: Bayes factors might not be precise.
##   For precise Bayes factors, sampling at least 40,000 posterior samples is
##   recommended.
```

```
print(main_BF, digits = 4)
```

```
## Summary of Posterior Distribution
```

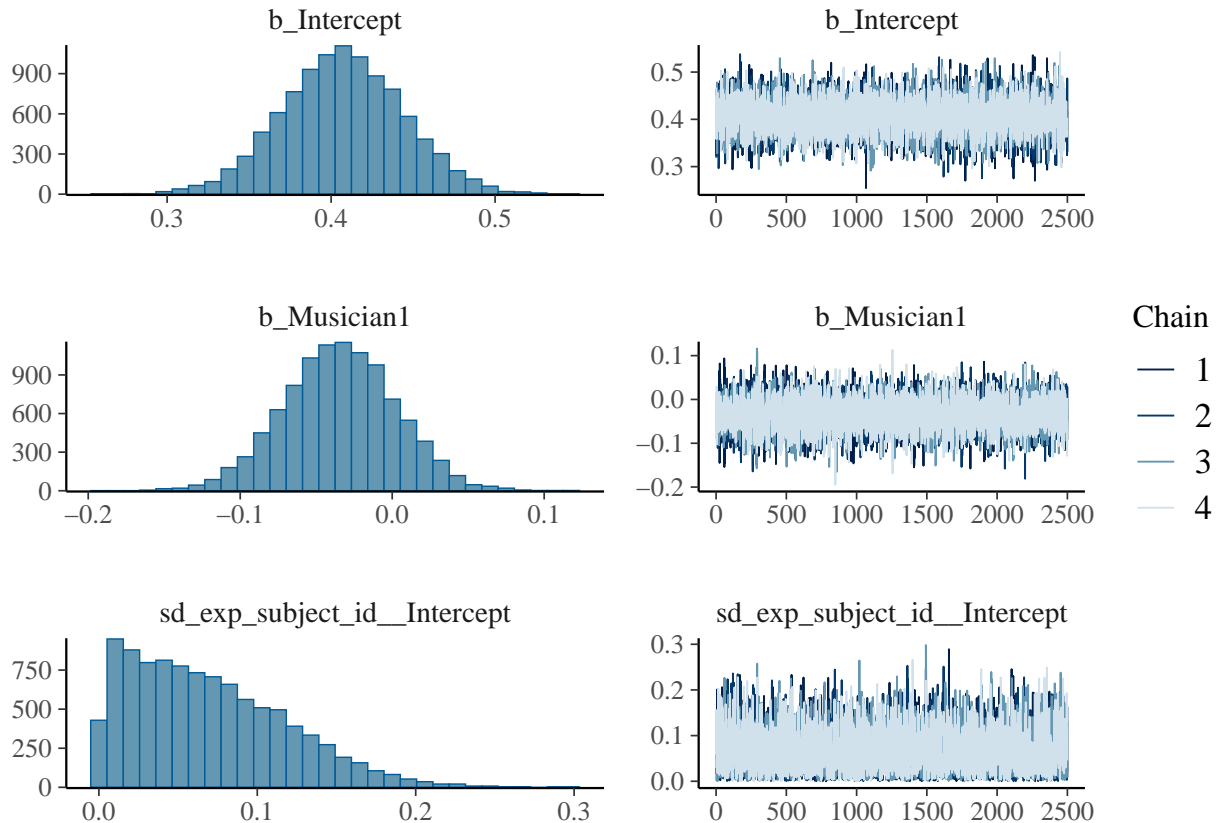
```
##
## Parameter   | Median | MAD | 95% CI | BF | Rhat | ESS
## -----
## (Intercept) | 0.7884 | 0.0658 | [ 0.66, 0.92] | 3.59e+14 | 1.000 | 9602.0000
## Musician1    | -0.0363 | 0.0376 | [-0.11, 0.04] | 0.058 | 1.000 | 12118.0000
## scramble2    | -0.4272 | 0.0912 | [-0.61, -0.26] | 973.08 | 1.000 | 10633.0000
## scramble3    | -0.6907 | 0.0909 | [-0.86, -0.51] | 1.29e+06 | 1.000 | 10298.0000
```

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 = 5000,
  file = '../models/E2_mus_only')
```

```
plot(mus_only)
```



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

```
## Family: bernoulli
## Links: mu = logit
## Formula: response ~ Musician + (1 | exp_subject_id)
## Data: data (Number of observations: 3158)
## Draws: 4 chains, each with iter = 5000; warmup = 2500; thin = 1;
## total post-warmup draws = 10000
##
## Multilevel Hyperparameters:
## ~exp_subject_id (Number of levels: 105)
## Estimate Est.Error 1-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## sd(Intercept) 0.0692 0.0488 0.0033 0.1787 1.0003 4349 5256
##
## Regression Coefficients:
## Estimate Est.Error 1-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## Intercept 0.4070 0.0371 0.3344 0.4802 1.0003 17021 7080
## Musician1 -0.0346 0.0370 -0.1077 0.0366 1.0006 16707 6209
##
```

```

## 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: 6
## Iteration: 7
## Iteration: 1
## Iteration: 2
## Iteration: 3
## Iteration: 4
## Iteration: 5
## Iteration: 6
## Iteration: 7
## Iteration: 8
print(BF_scramble)

## Estimated Bayes factor in favor of mus_scram over mus_only: 76447443676.54962
Very strong evidence for a main effect of scramble condition.

```

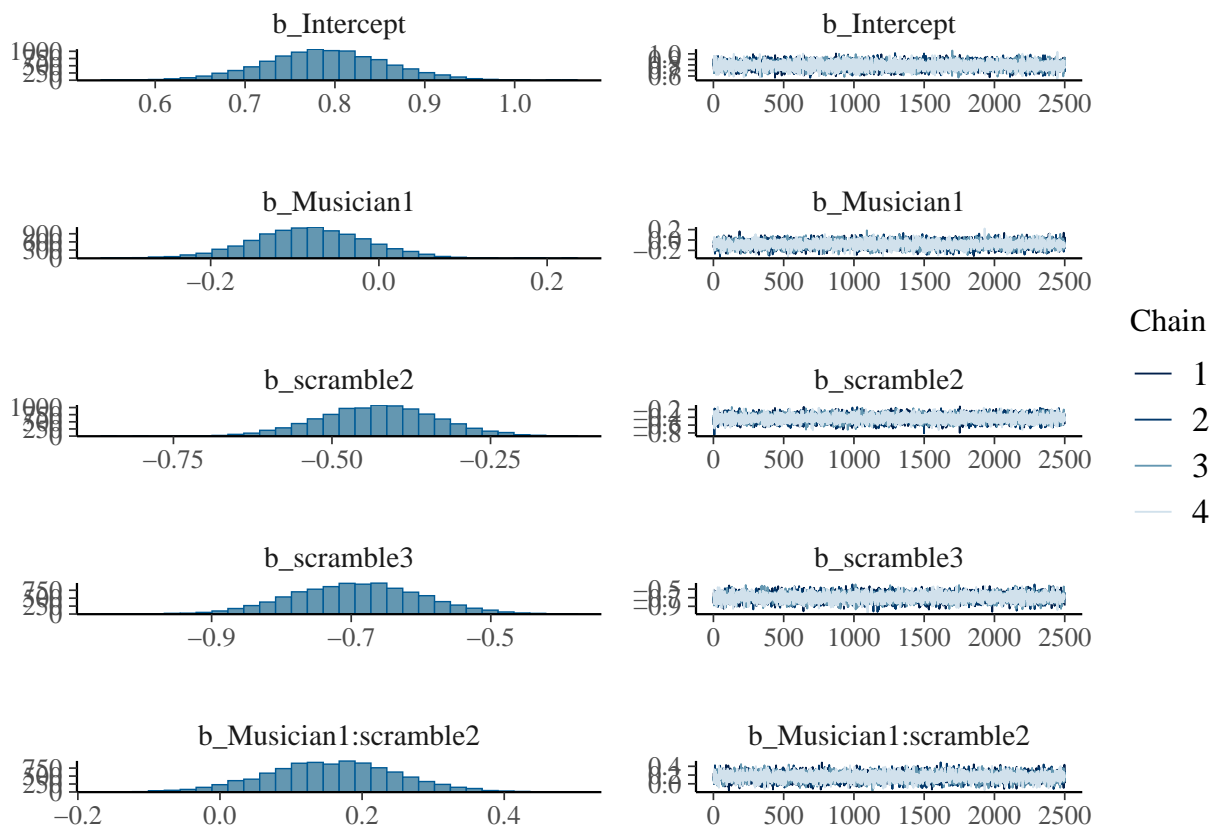


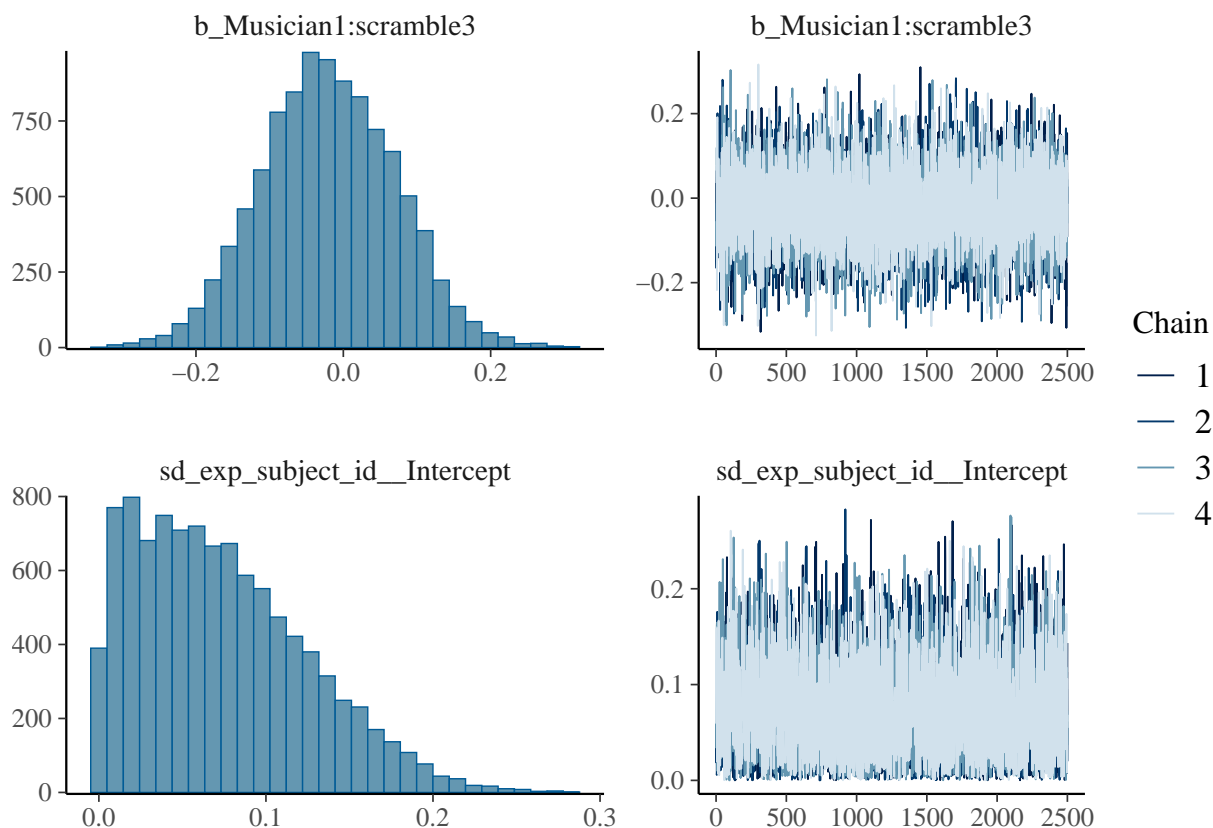
## Interaction between group and condition?

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 = 5000,  
  file = '../models/E2_mus_scram_int')
```

```
plot(mus_scram_int)
```





```
print(summary(mus_scram_int), 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 = 5000; warmup = 2500; thin = 1;
##         total post-warmup draws = 10000
##
## Multilevel Hyperparameters:
## ~exp_subject_id (Number of levels: 105)
##           Estimate Est.Error 1-95% CI u-95% CI   Rhat Bulk_ESS Tail_ESS
## sd(Intercept)   0.0735   0.0508  0.0031  0.1871 1.0012    3226    3899
##
## Regression Coefficients:
##           Estimate Est.Error 1-95% CI u-95% CI   Rhat Bulk_ESS
## Intercept         0.7898   0.0676  0.6572  0.9228 1.0003    10693
## Musician1         -0.0828   0.0669 -0.2154  0.0487 1.0013     6032
## scramble2         -0.4267   0.0921 -0.6069 -0.2457 1.0001    11551
## scramble3         -0.6922   0.0920 -0.8709 -0.5116 1.0000    11489
## Musician1:scramble2  0.1580   0.0919 -0.0231  0.3397 1.0015     7449
## Musician1:scramble3 -0.0202   0.0912 -0.1951  0.1569 1.0008     7739
##
##           Tail_ESS
## Intercept         7100
## Musician1         6849
## scramble2         7427
## scramble3         7794
## Musician1:scramble2 7715
```

```

## Musician1:scramble3      7468
##
## 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_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
## Iteration: 6
print(BF_int)

## Estimated Bayes factor in favor of mus_scram_int over mus_scram: 0.07156
Strong evidence against an interaction between group and condition.

```

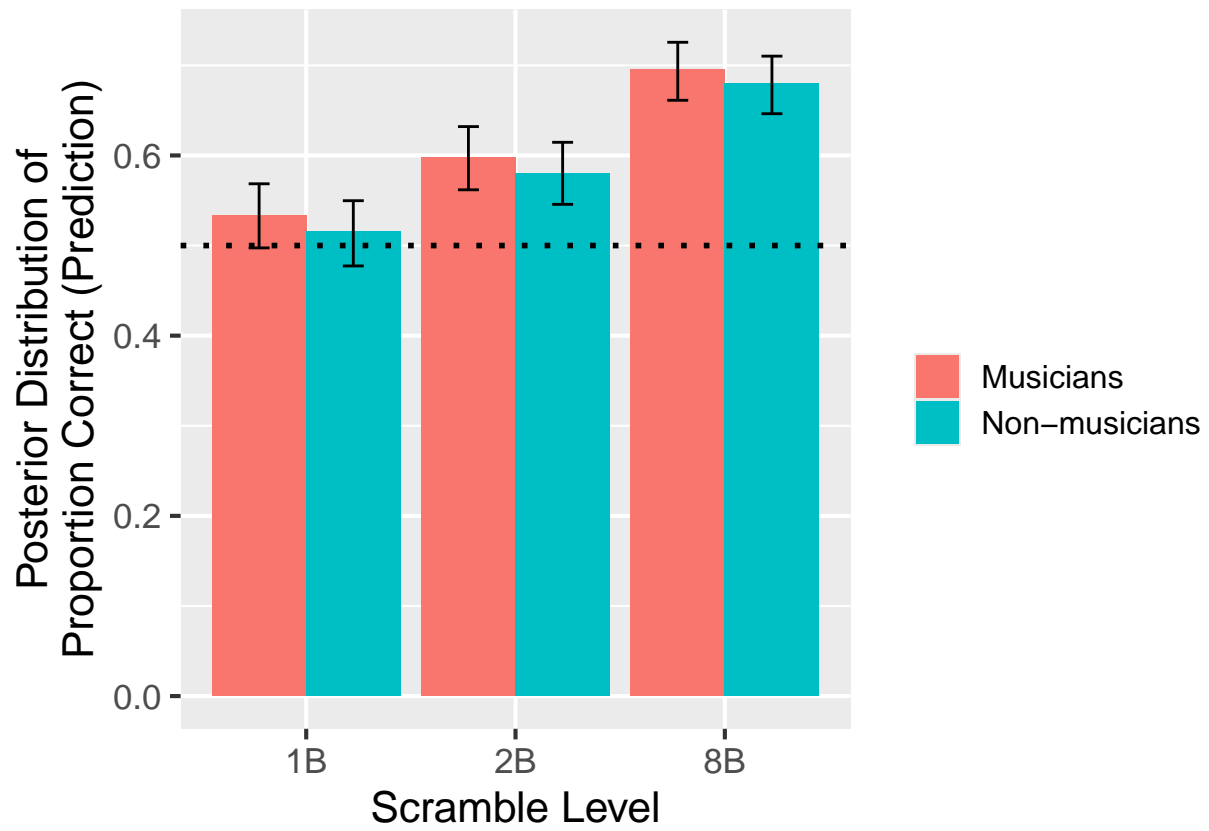
## Figure 2B

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) +  
  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')
```

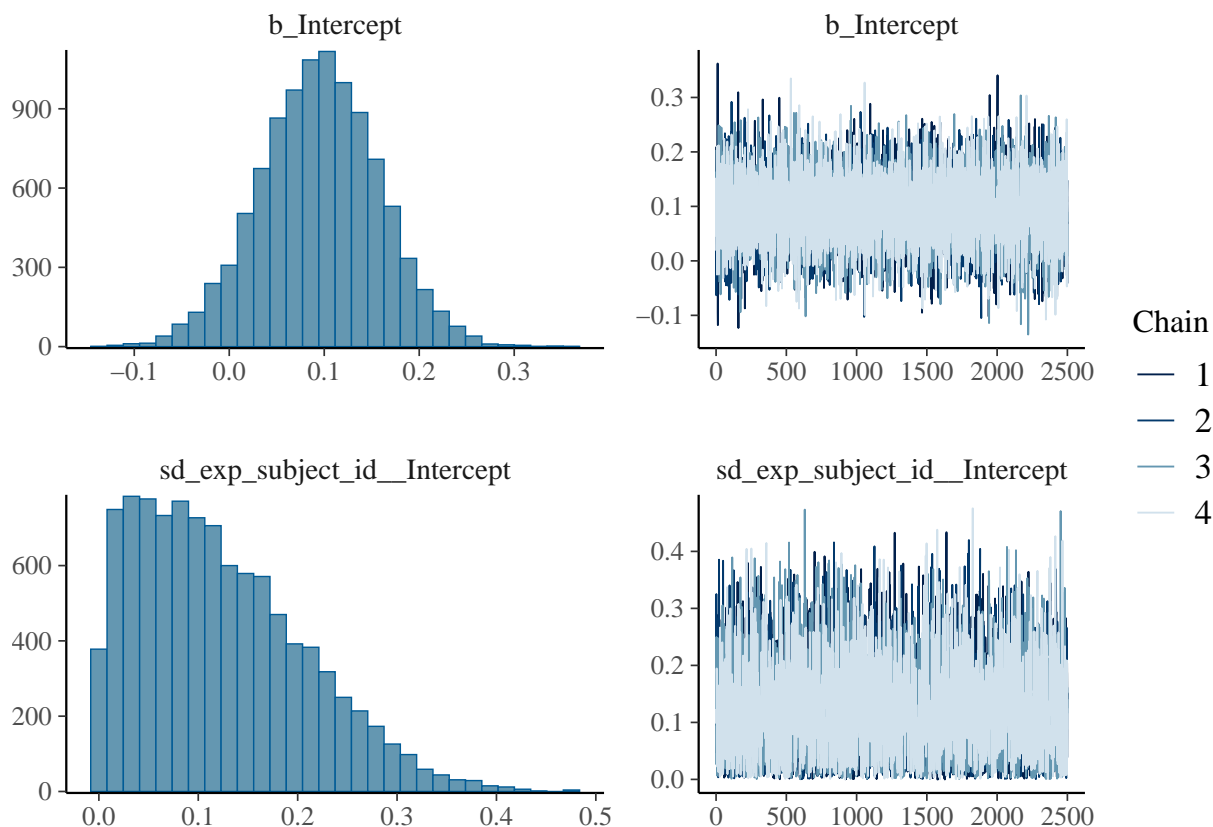
```
get_prior(response ~ 1 + (1 | exp_subject_id), data = data1B)
```

```
##           prior      class      coef      group resp dpar nlpar lb ub
## student_t(3, 0, 2.5) Intercept
## student_t(3, 0, 2.5)      sd
## 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
##      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 = 5000,
  file = '../models/E2_only1B')
```

```
plot(only1B)
```



```
print(summary(only1B), 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 ~ 1 + (1 | exp_subject_id)
## Data: data1B (Number of observations: 1054)
## Draws: 4 chains, each with iter = 5000; warmup = 2500; thin = 1;
## total post-warmup draws = 10000
##
## 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)  0.1220    0.0844   0.0053   0.3105 1.0011    3537    4513
##
## Regression Coefficients:
##           Estimate Est.Error l-95% CI u-95% CI   Rhat Bulk_ESS Tail_ESS
## Intercept    0.0948    0.0627  -0.0301   0.2176 1.0004    14871    6900
##
## 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?

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

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

## Warning: Bayes factors might not be precise.
## For precise Bayes factors, sampling at least 40,000 posterior samples is
## recommended.

## Bayes Factor (Savage-Dickey density ratio)
##
## Parameter | BF
## -----
## (Intercept) | 0.074
##
## * 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)
```

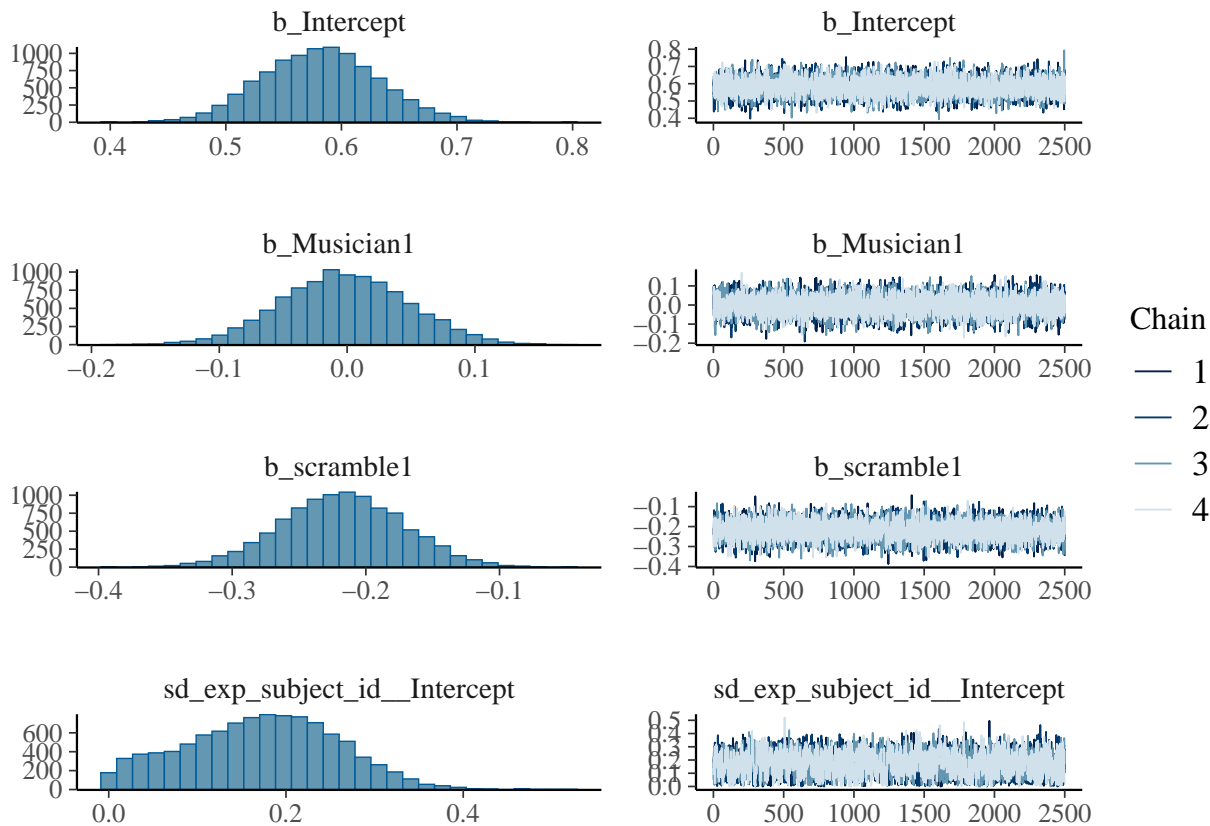
```
## Warning: contrasts dropped from factor scramble due to missing levels
```

```
##           prior      class      coef      group resp dpar nlpar lb ub
##           (flat)         b
##           (flat)         b MusicianNo
##           (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
##      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 = 'scramble1')
  ),
  save_pars = save_pars(all = TRUE), iter = 5000,
  file = '../models/E2_no1B')
```

```
plot(no1B)
```





```
print(summary(no1B), 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 = 5000; warmup = 2500; thin = 1;
## total post-warmup draws = 10000
##
## Multilevel Hyperparameters:
## ~exp_subject_id (Number of levels: 105)
## Estimate Est.Error 1-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## sd(Intercept) 0.1717 0.0854 0.0127 0.3321 1.0010 2032 2408
##
## Regression Coefficients:
## Estimate Est.Error 1-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## Intercept 0.5818 0.0502 0.4864 0.6820 1.0003 9046 6511
## Musician1 -0.0017 0.0496 -0.0979 0.0957 1.0001 8824 6942
## scramble1 -0.2178 0.0457 -0.3088 -0.1284 1.0003 12179 7537
##
## 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).
```

```
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
```

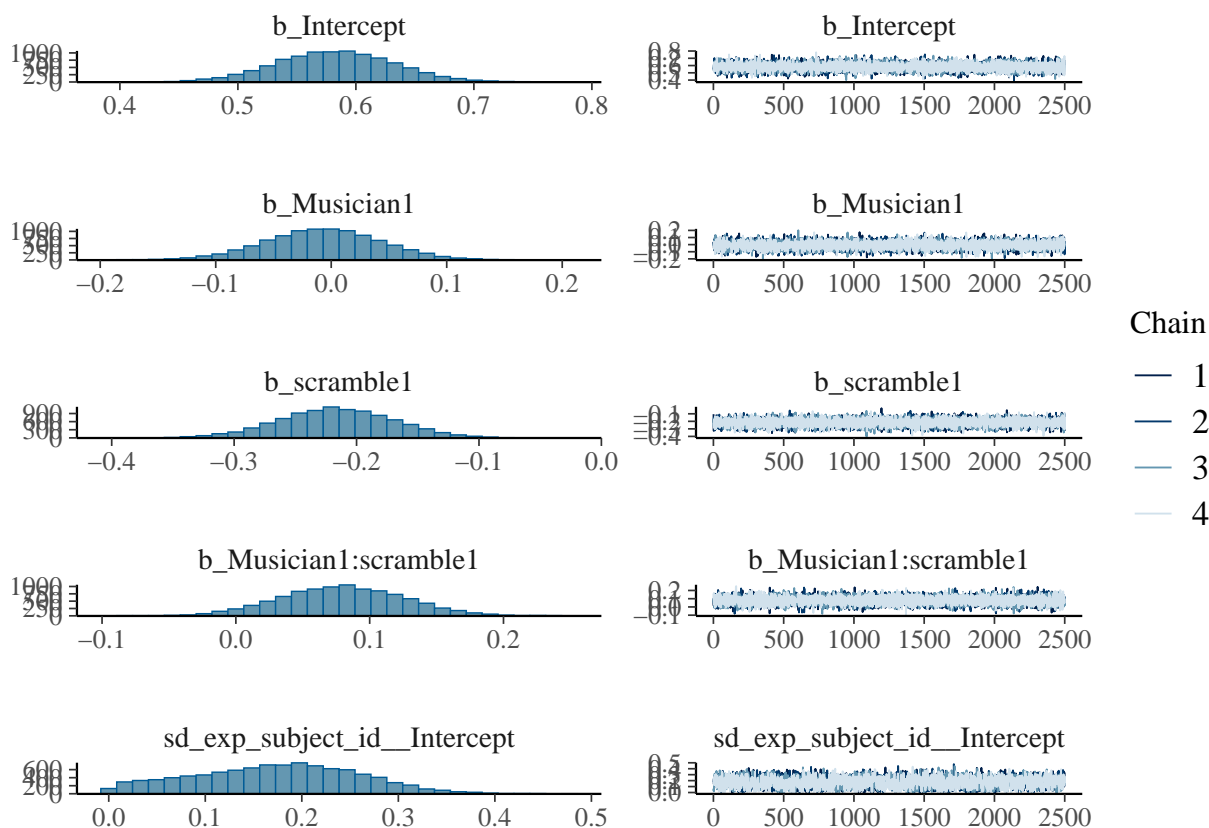
```

##          (flat)          b
##          (flat)          b          MusicianNo
##          (flat)          b MusicianNo: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

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 = 'scramble1'),
    set_prior('normal(0, 1)', coef = 'Musician1:scramble1')
  ),
  save_pars = save_pars(all = TRUE), iter = 5000,
  file = '../models/E2_no1B_int')

plot(no1B_int)

```



```
print(summary(no1B_int), 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 = 5000; warmup = 2500; thin = 1;
## total post-warmup draws = 10000
##
## Multilevel Hyperparameters:
## ~exp_subject_id (Number of levels: 105)
## Estimate Est.Error 1-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## sd(Intercept) 0.1720 0.0844 0.0150 0.3317 1.0016 2138 3201
##
## Regression Coefficients:
## Estimate Est.Error 1-95% CI u-95% CI Rhat Bulk_ESS
## Intercept 0.5825 0.0497 0.4861 0.6799 1.0002 9966
## Musician1 -0.0068 0.0503 -0.1069 0.0901 1.0005 10016
## scramble1 -0.2164 0.0466 -0.3085 -0.1254 1.0002 12902
## Musician1:scramble1 0.0798 0.0460 -0.0093 0.1693 1.0000 13173
## Tail_ESS
## Intercept 7076
## Musician1 7388
## scramble1 7357
## Musician1:scramble1 6636
##
## 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_no1B_int <- bayes_factor(no1B_int, no1B)
```

```
## Iteration: 1
```

```
## Iteration: 2
```

```
## Iteration: 3
```

```
## Iteration: 4
```

```
## Iteration: 5
```

```
## Iteration: 6
```

```
## Iteration: 1
```

```
## Iteration: 2
```

```
## Iteration: 3
```

```
## Iteration: 4
```

```
## Iteration: 5
```

```
## Iteration: 6
```

```
## Iteration: 7
```

```
print(BF_no1B_int)
```

```
## Estimated Bayes factor in favor of no1B_int over no1B: 0.20473
```

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

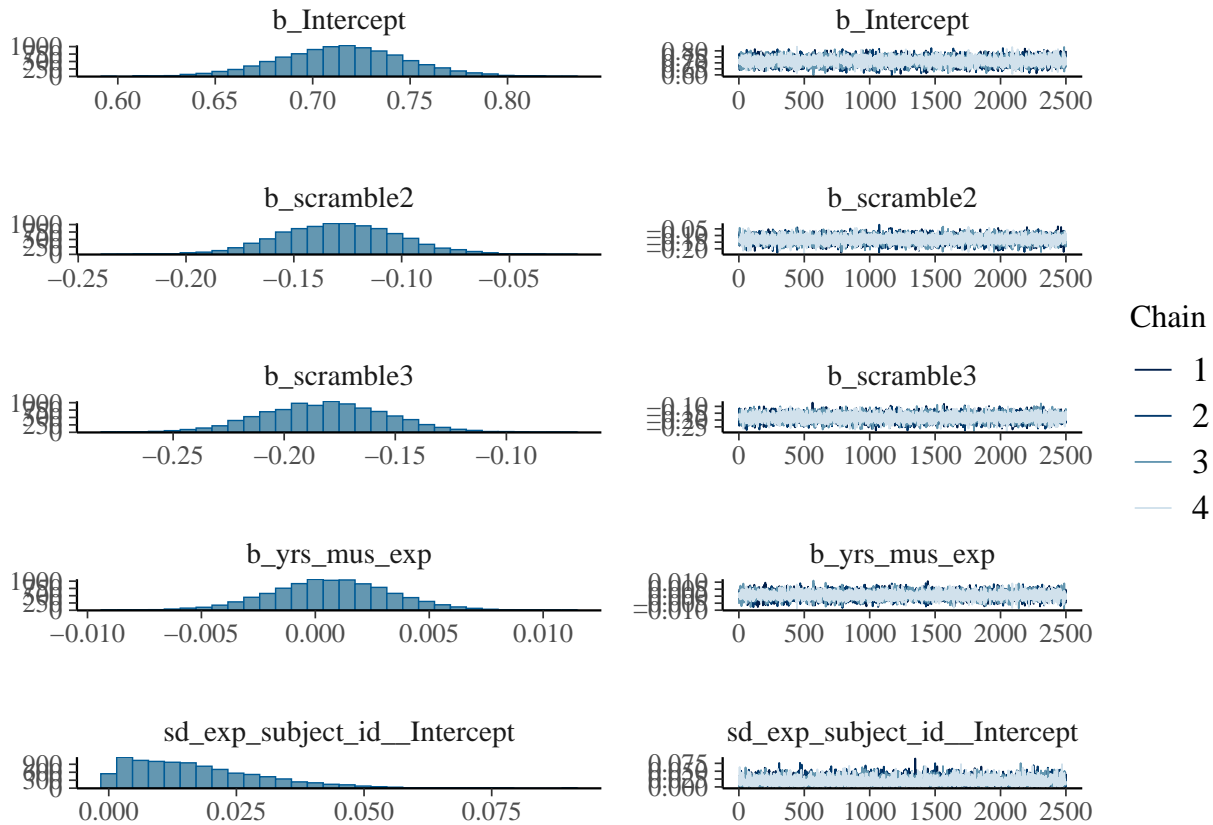
For this analysis, we're operating on the scale of accuracy. Because we don't see ceiling effects (i.e. participants aren't getting too close to perfect accuracy), a linear model is appropriate enough.

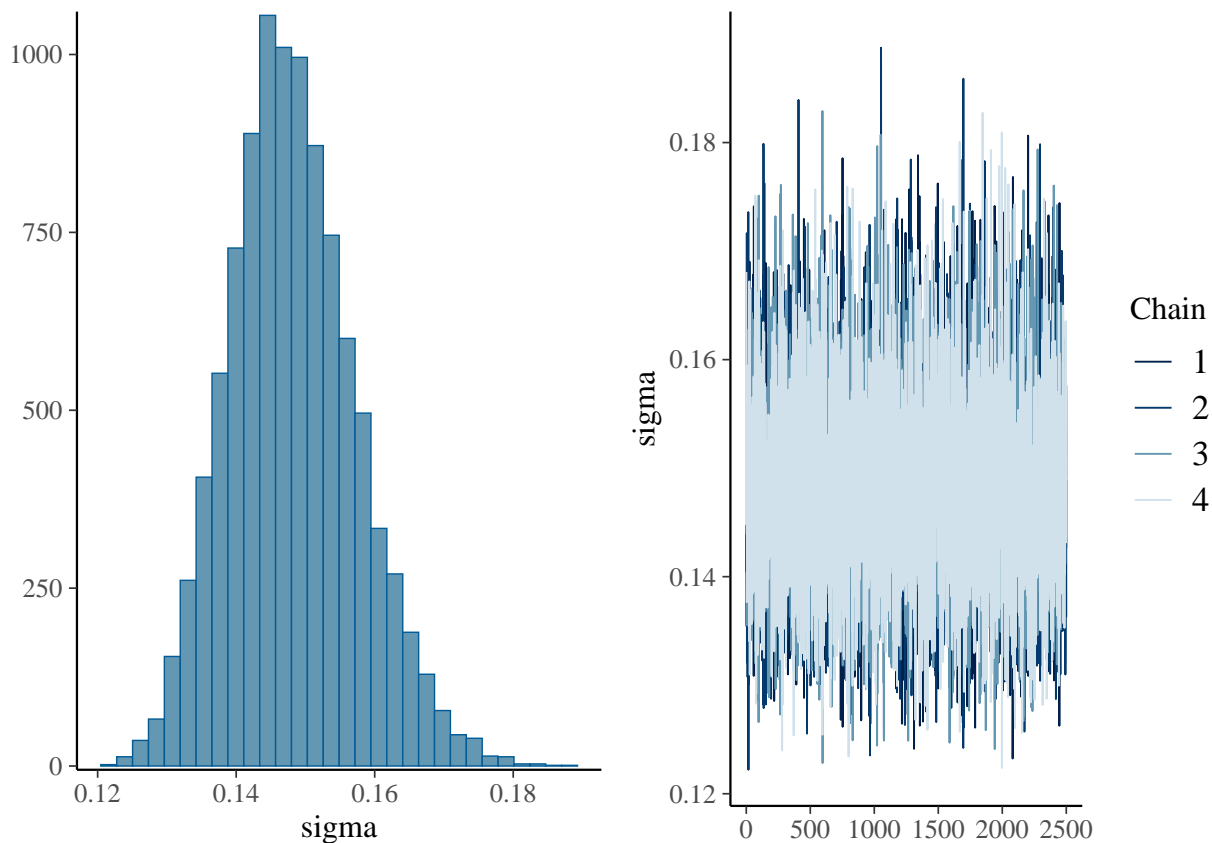
```
these_priors <- c(  
  set_prior('normal(0.75, 0.1)', class = 'Intercept'),  
  set_prior('normal(-0.1, 0.1)', coef = 'scramble2'),  
  set_prior('normal(-0.2, 0.1)', coef = 'scramble3'),  
  set_prior('normal(0, 0.1)', coef = 'yrs_mus_exp')  
)
```

## 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 = 5000,  
  file = '../models/E2_years')
```

```
plot(years_mus_scram)
```





```
print(summary(years_mus_scram), 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

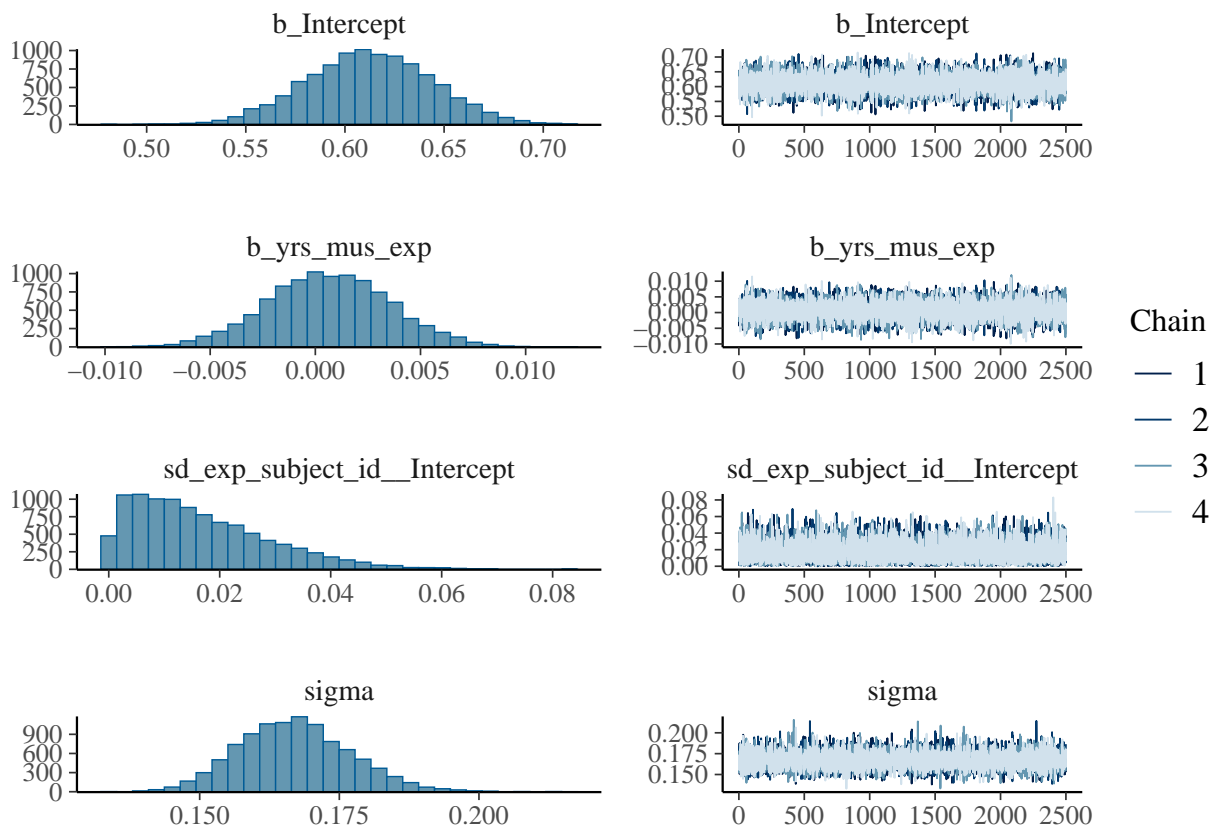
## Family: gaussian
## Links: mu = identity; sigma = identity
## Formula: accuracy ~ scramble + yrs_mus_exp + (1 | exp_subject_id)
## Data: yrs_exp (Number of observations: 147)
## Draws: 4 chains, each with iter = 5000; warmup = 2500; thin = 1;
## total post-warmup draws = 10000
##
## Multilevel Hyperparameters:
## ~exp_subject_id (Number of levels: 49)
## Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## sd(Intercept) 0.0172 0.0127 0.0008 0.0471 1.0006 6212 5659
##
## Regression Coefficients:
## Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## Intercept 0.7151 0.0326 0.6494 0.7797 1.0008 15991 7104
## scramble2 -0.1291 0.0279 -0.1837 -0.0731 1.0002 15253 8207
## scramble3 -0.1817 0.0282 -0.2375 -0.1267 0.9998 14505 7673
## yrs_mus_exp 0.0007 0.0026 -0.0045 0.0059 1.0004 15685 6425
##
## Further Distributional Parameters:
## Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## sigma 0.1482 0.0091 0.1317 0.1675 1.0003 16196 7120
```

```
##
## 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)

```
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 = 5000,
  file = '../models/E2_years_null')
```

```
plot(years_mus)
```



```
print(summary(years_mus), digits = 4)
```

```
## 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 = 5000; warmup = 2500; thin = 1;
## total post-warmup draws = 10000
##
## Multilevel Hyperparameters:
## ~exp_subject_id (Number of levels: 49)
## Estimate Est.Error 1-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
```



```

## sd(Intercept)    0.0167    0.0124    0.0007    0.0458 1.0010    6435    5210
##
## Regression Coefficients:
##      Estimate Est.Error l-95% CI u-95% CI   Rhat Bulk_ESS Tail_ESS
## Intercept      0.6123    0.0316   0.5509   0.6735 1.0004    17550     7165
## yrs_mus_exp    0.0007    0.0029  -0.0051   0.0064 1.0005    17923     6999
##
## Further Distributional Parameters:
##      Estimate Est.Error l-95% CI u-95% CI   Rhat Bulk_ESS Tail_ESS
## sigma   0.1671    0.0100   0.1489   0.1874 1.0012    17877     7495
##
## 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).

```

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

## Warning: Bayes factors might not be precise.  
## For precise Bayes factors, sampling at least 40,000 posterior samples is  
## recommended.

```
print(yrs_BF, digits = 4)
```

## Summary of Posterior Distribution

## Parameter	Median	MAD	95% CI	BF	Rhat	ESS
## (Intercept)	0.7154	0.0323	[ 0.65, 0.78]	2.60e+22	1.000	15954.0000
## scramble2	-0.1290	0.0280	[-0.18, -0.07]	757.55	1.000	15174.0000
## scramble3	-0.1814	0.0284	[-0.24, -0.13]	2.60e+04	1.000	14437.0000
## yrs_mus_exp	0.0007	0.0026	[ 0.00, 0.01]	0.026	1.000	15626.0000

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

## Warning: Bayes factors might not be precise.  
## For precise Bayes factors, sampling at least 40,000 posterior samples is  
## recommended.

```
print(yrs_null_BF, digits = 4)
```

## Summary of Posterior Distribution

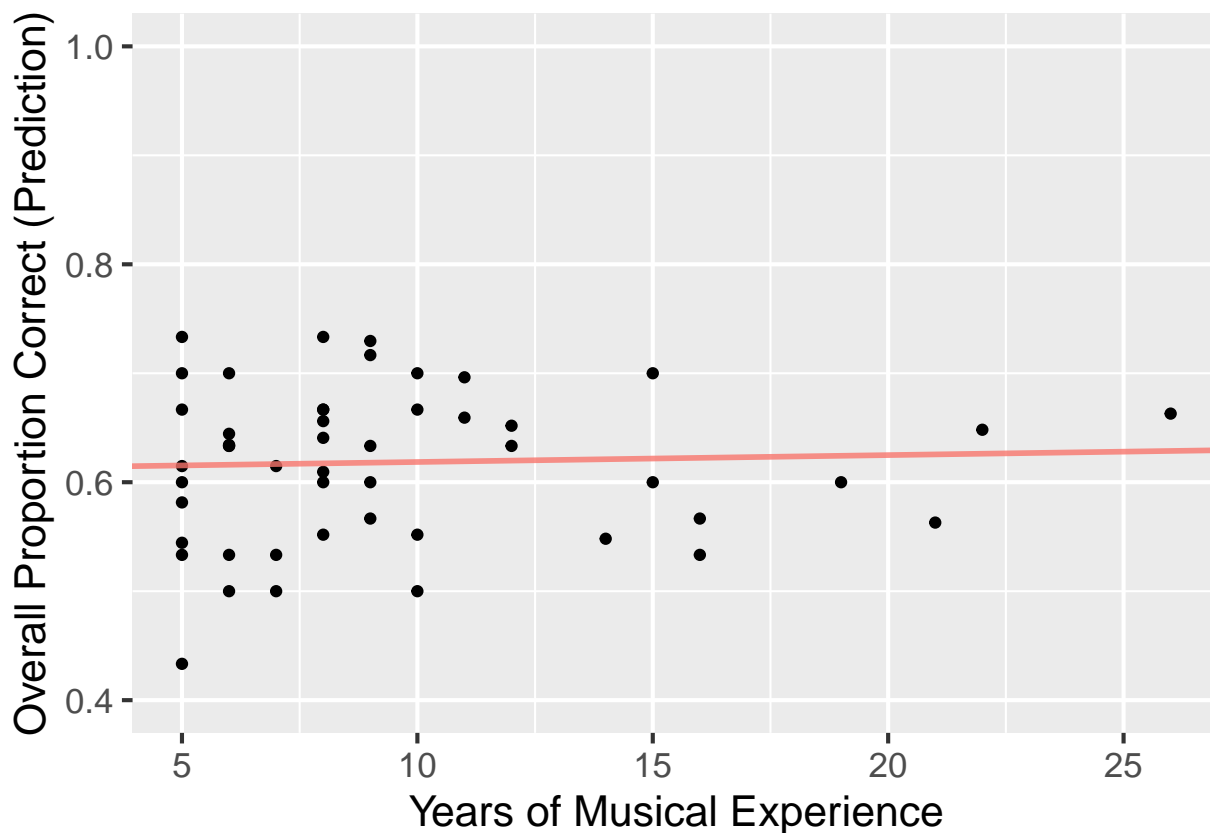
## Parameter	Median	MAD	95% CI	BF	Rhat	ESS
## (Intercept)	0.6122	0.0321	[ 0.55, 0.67]	4.75e+20	1.000	17648.0000
## yrs_mus_exp	0.0006	0.0029	[ 0.00, 0.01]	0.030	1.000	17847.0000

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(0.4, 1, 0.1)) +
  ylim(0.4, 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)
```