

E4 categorization

R. Cassano-Coleman

2025-06-27

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

Set up

```
set.seed(15000)

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

## Rows: 864 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(scramble = ifelse(scramble == 'intact', 'Intact', scramble)) %>%
  mutate(exp_subject_id = as.factor(exp_subject_id),
         response = ifelse(response == 'Correct', TRUE, FALSE),
         scramble = factor(scramble, levels = c('Intact', '8B', '2B', '1B')),
         Musician = factor(Musician, levels = c('Yes', 'No'))) %>%
  filter(!is.na(response))
```

Set the contrast for condition.

```
contrasts(data$scramble) <- contr.treatment(4)
print(contrasts(data$scramble))
```

```
##           2 3 4
## Intact 0 0 0
## 8B      1 0 0
## 2B      0 1 0
## 1B      0 0 1
```

Main analysis

Priors

Priors are expressed in log(odds) space.

Intercept: Given that chance is 25%, we assume that participants will perform somewhere between chance and ceiling. We expect the center of the distribution of accuracy to be somewhere around 60% or 65%. If we use a center of 65% and an SD of 1.5, 95% of the values fall between 8.46% and 97.4%.

```
prior_intercept <- set_prior('normal(log(0.65 / (1 - 0.65)), 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_scramble8B <- set_prior('normal(-0.1, 1)', coef = 'scramble2')
prior_scramble2B <- set_prior('normal(-0.2, 1)', coef = 'scramble3')
prior_scramble1B <- set_prior('normal(-0.3, 1)', coef = 'scramble4')
```

Interaction: We expect no interaction between group and scramble.

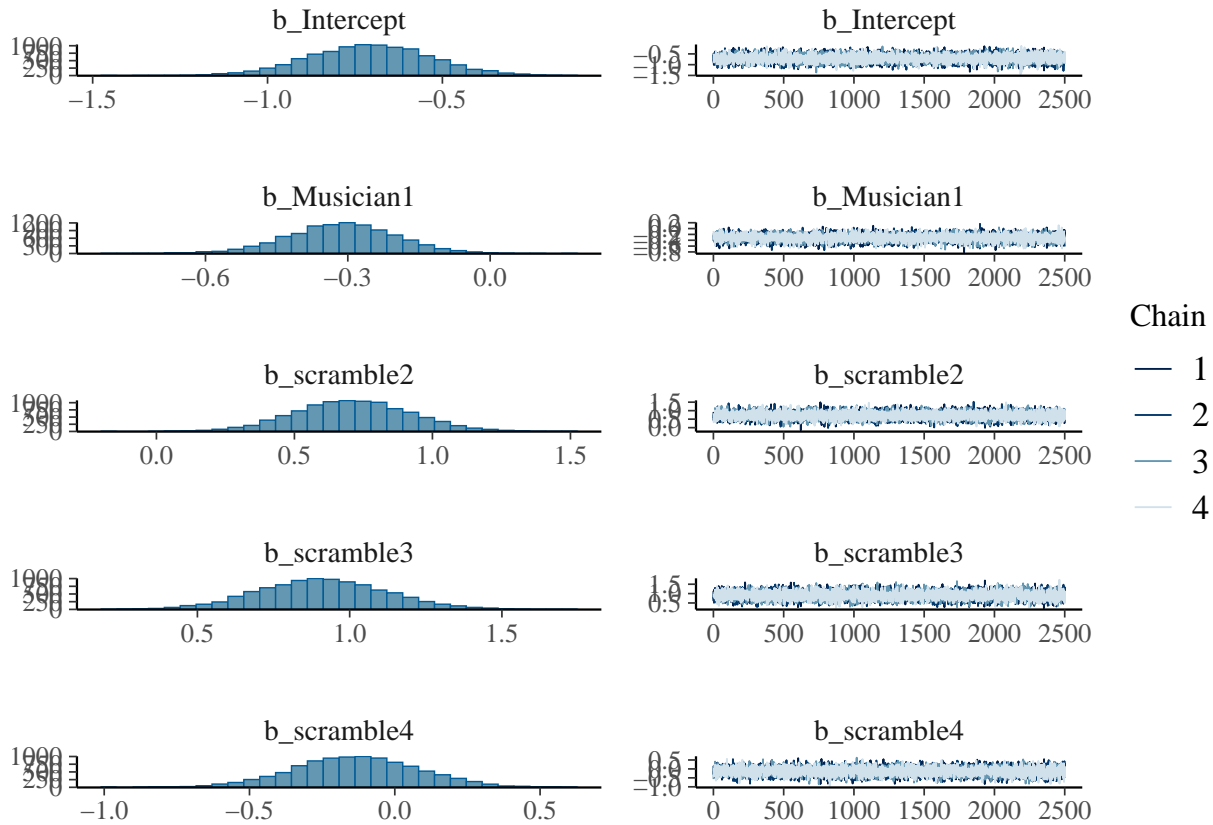
```
prior_int8B <- set_prior('normal(0, 1)', coef = 'Musician1:scramble2')
prior_int2B <- set_prior('normal(0, 1)', coef = 'Musician1:scramble3')
prior_int1B <- set_prior('normal(0, 1)', coef = 'Musician1:scramble4')
```

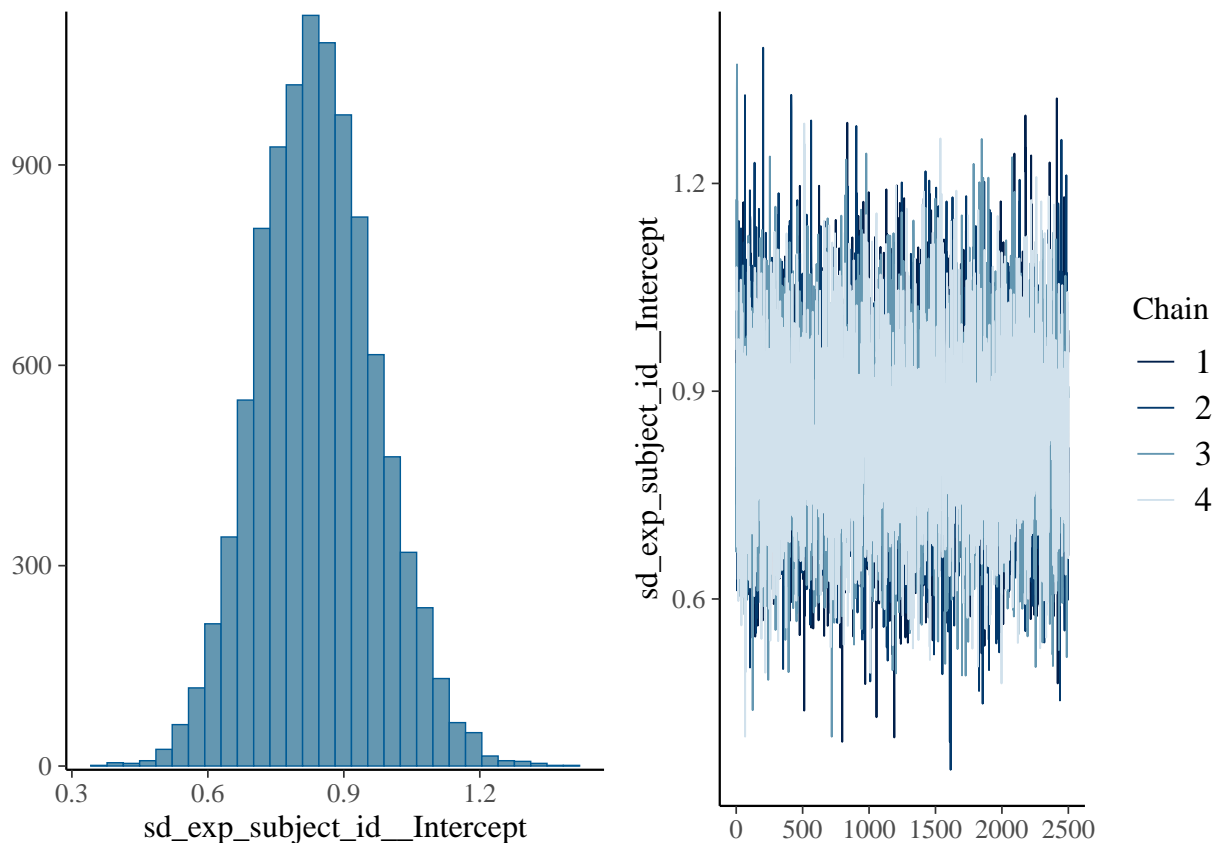
Random slope for subjects: *Leave this as default for now, may update.*

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_scramble8B, prior_scramble2B, prior_scramble1B),  
  save_pars = save_pars(all = TRUE), iter = 5000,  
  file = '../models/E4_mus_scam')
```

```
plot(mus_scam)
```





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

```
## Family: bernoulli
## Links: mu = logit
## Formula: response ~ Musician + scramble + (1 | exp_subject_id)
## Data: data (Number of observations: 825)
## Draws: 4 chains, each with iter = 5000; warmup = 2500; thin = 1;
## total post-warmup draws = 10000
##
## Multilevel Hyperparameters:
## ~exp_subject_id (Number of levels: 106)
## Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## sd(Intercept) 0.8422 0.1293 0.5979 1.1043 1.0001 4297 6368
##
## Regression Coefficients:
## Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## Intercept -0.7071 0.1728 -1.0471 -0.3661 1.0001 11873 7437
## Musician1 -0.3129 0.1146 -0.5409 -0.0930 0.9998 9751 7660
## scramble2 0.7118 0.2097 0.3023 1.1263 1.0000 12921 8289
## scramble3 0.9100 0.2101 0.4988 1.3293 0.9997 12182 7615
## scramble4 -0.1331 0.2153 -0.5599 0.2832 1.0001 13708 8650
##
## 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
##  Intact    -0.70627   -1.048   -0.368
##    8B       0.00344   -0.348    0.331
##    2B       0.20343   -0.134    0.527
##    1B      -0.84006   -1.199   -0.492
##
## 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
##  Intact - 8B    -0.709   -1.099   -0.282
##  Intact - 2B    -0.908   -1.318   -0.489
##  Intact - 1B     0.134   -0.289    0.551
##    8B - 2B     -0.200   -0.600    0.244
##    8B - 1B      0.842    0.402    1.278
##    2B - 1B      1.042    0.610    1.471
##
## 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      Intact   -0.395   -0.7959  0.00399
##  No       Intact   -1.020   -1.4265 -0.60136
##  Yes      8B       0.317   -0.0873  0.72255
##  No       8B      -0.305   -0.7124  0.11702
##  Yes      2B       0.512    0.1266  0.91998
##  No       2B      -0.110   -0.4984  0.30549
##  Yes      1B      -0.527   -0.9215 -0.10908
##  No       1B     -1.149   -1.5819 -0.72118
##
## 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 Intact - No Intact  0.6226    0.186    1.0821
##  Yes Intact - Yes 8B    -0.7089   -1.099   -0.2824
##  Yes Intact - No 8B    -0.0845   -0.695    0.5258
##  Yes Intact - Yes 2B   -0.9077   -1.318   -0.4889
##  Yes Intact - No 2B   -0.2868   -0.911    0.3061
##  Yes Intact - Yes 1B    0.1336   -0.289    0.5509
##  Yes Intact - No 1B    0.7542    0.133    1.3860
```

## No Intact - Yes 8B	-1.3386	-1.921	-0.6880
## No Intact - No 8B	-0.7089	-1.099	-0.2824
## No Intact - Yes 2B	-1.5359	-2.152	-0.9120
## No Intact - No 2B	-0.9077	-1.318	-0.4889
## No Intact - Yes 1B	-0.4936	-1.137	0.0773
## No Intact - No 1B	0.1336	-0.289	0.5509
## Yes 8B - No 8B	0.6226	0.186	1.0821
## Yes 8B - Yes 2B	-0.2003	-0.600	0.2445
## Yes 8B - No 2B	0.4252	-0.180	1.0495
## Yes 8B - Yes 1B	0.8424	0.402	1.2779
## Yes 8B - No 1B	1.4688	0.839	2.0973
## No 8B - Yes 2B	-0.8212	-1.461	-0.2121
## No 8B - No 2B	-0.2003	-0.600	0.2445
## No 8B - Yes 1B	0.2171	-0.404	0.8420
## No 8B - No 1B	0.8424	0.402	1.2779
## Yes 2B - No 2B	0.6226	0.186	1.0821
## Yes 2B - Yes 1B	1.0417	0.610	1.4707
## Yes 2B - No 1B	1.6665	1.026	2.2945
## No 2B - Yes 1B	0.4175	-0.192	1.0378
## No 2B - No 1B	1.0417	0.610	1.4707
## Yes 1B - No 1B	0.6226	0.186	1.0821
##			
## 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
```

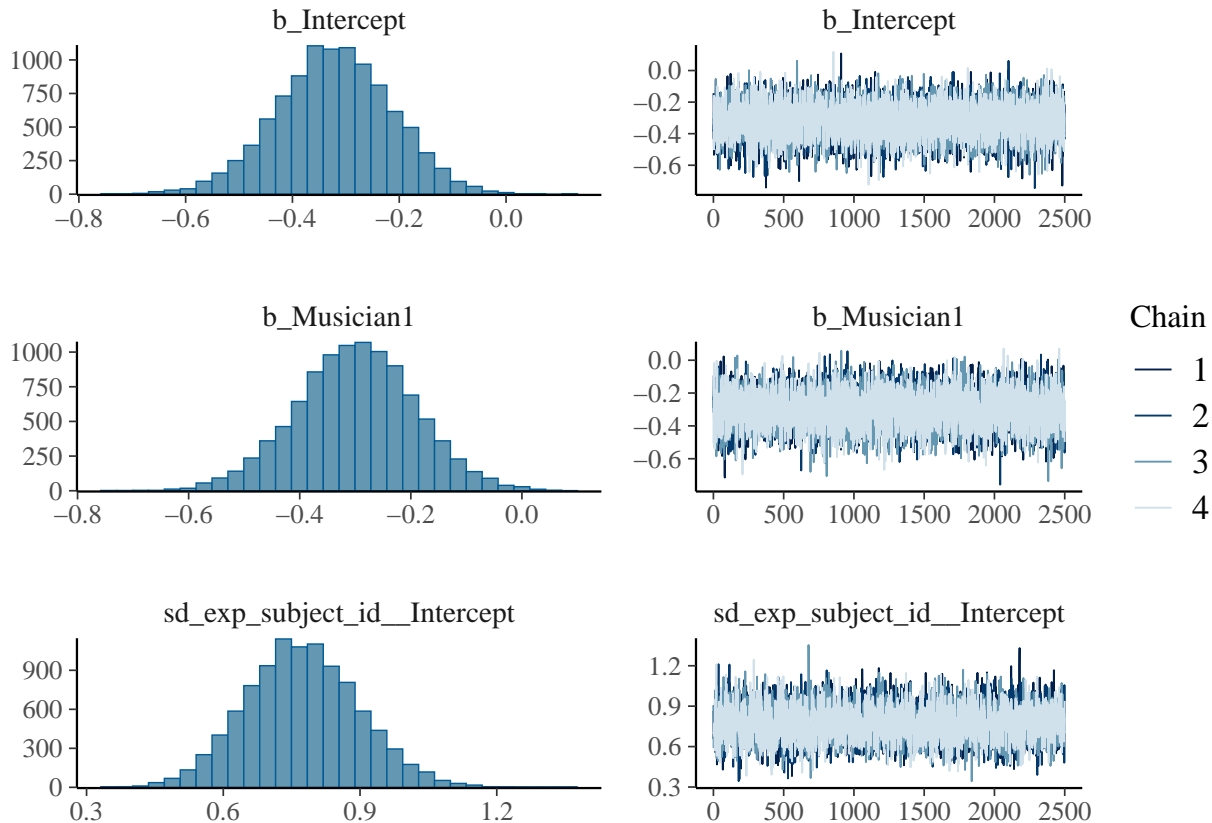
```
##  
## Parameter | Median | MAD | 95% CI | BF | Rhat | ESS  
## -----  
## (Intercept) | -0.7063 | 0.1739 | [-1.05, -0.37] | 179.78 | 1.000 | 11777.0000  
## Musician1 | -0.3113 | 0.1117 | [-0.54, -0.09] | 5.44 | 1.000 | 9652.0000  
## scramble2 | 0.7089 | 0.2080 | [ 0.28, 1.10] | 57.71 | 1.000 | 12850.0000  
## scramble3 | 0.9077 | 0.2106 | [ 0.49, 1.32] | 1.04e+03 | 1.000 | 12108.0000  
## scramble4 | -0.1336 | 0.2154 | [-0.55, 0.29] | 0.252 | 1.000 | 13622.0000
```

Moderate evidence for 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/E4_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: 825)
## Draws: 4 chains, each with iter = 5000; warmup = 2500; thin = 1;
## total post-warmup draws = 10000
##
## Multilevel Hyperparameters:
## ~exp_subject_id (Number of levels: 106)
## Estimate Est.Error 1-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## sd(Intercept) 0.7733 0.1236 0.5389 1.0241 1.0004 4099 5764
##
## Regression Coefficients:
## Estimate Est.Error 1-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## Intercept -0.3219 0.1080 -0.5361 -0.1148 1.0001 9257 6906
## Musician1 -0.2994 0.1073 -0.5152 -0.0878 1.0002 8722 7401
##
```



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

```
print(BF_scramble)
```

```
## Estimated Bayes factor in favor of mus_scram over mus_only: 202477.26749
```

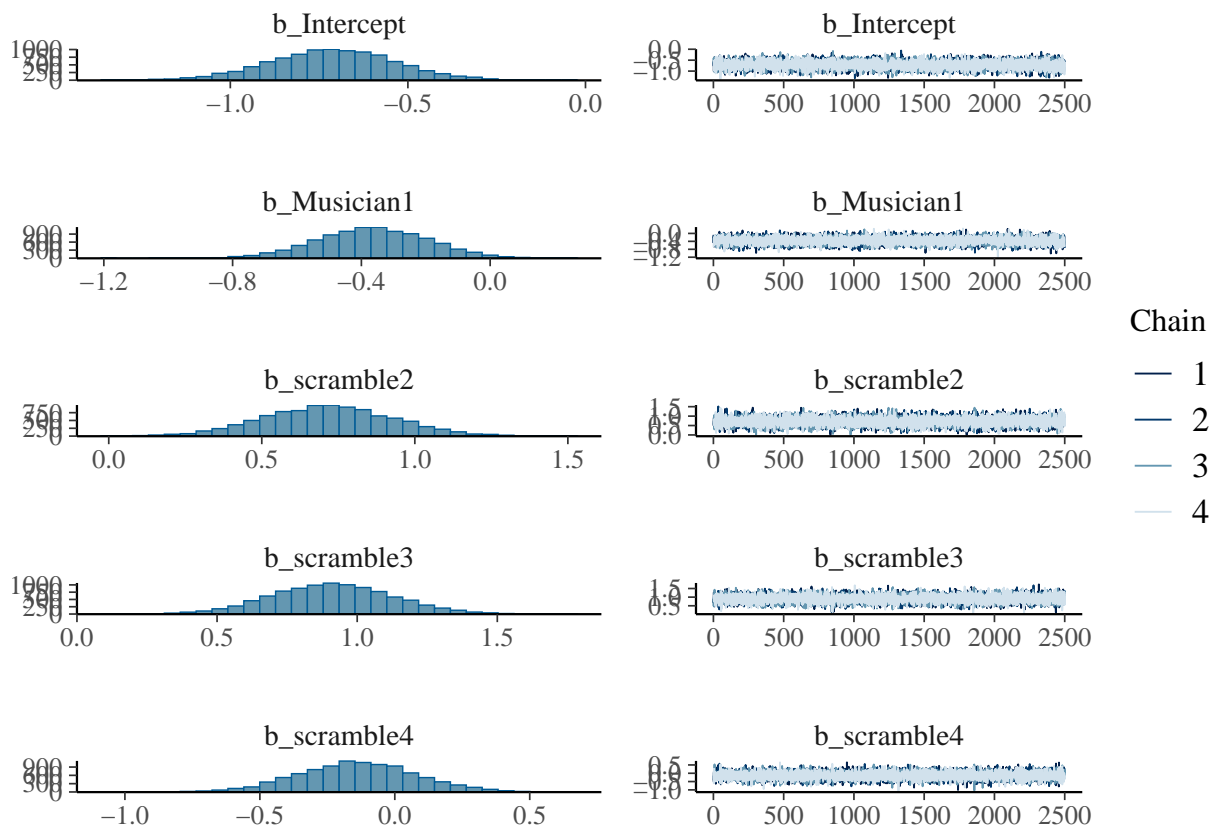
Very strong evidence for a main effect of scramble condition.

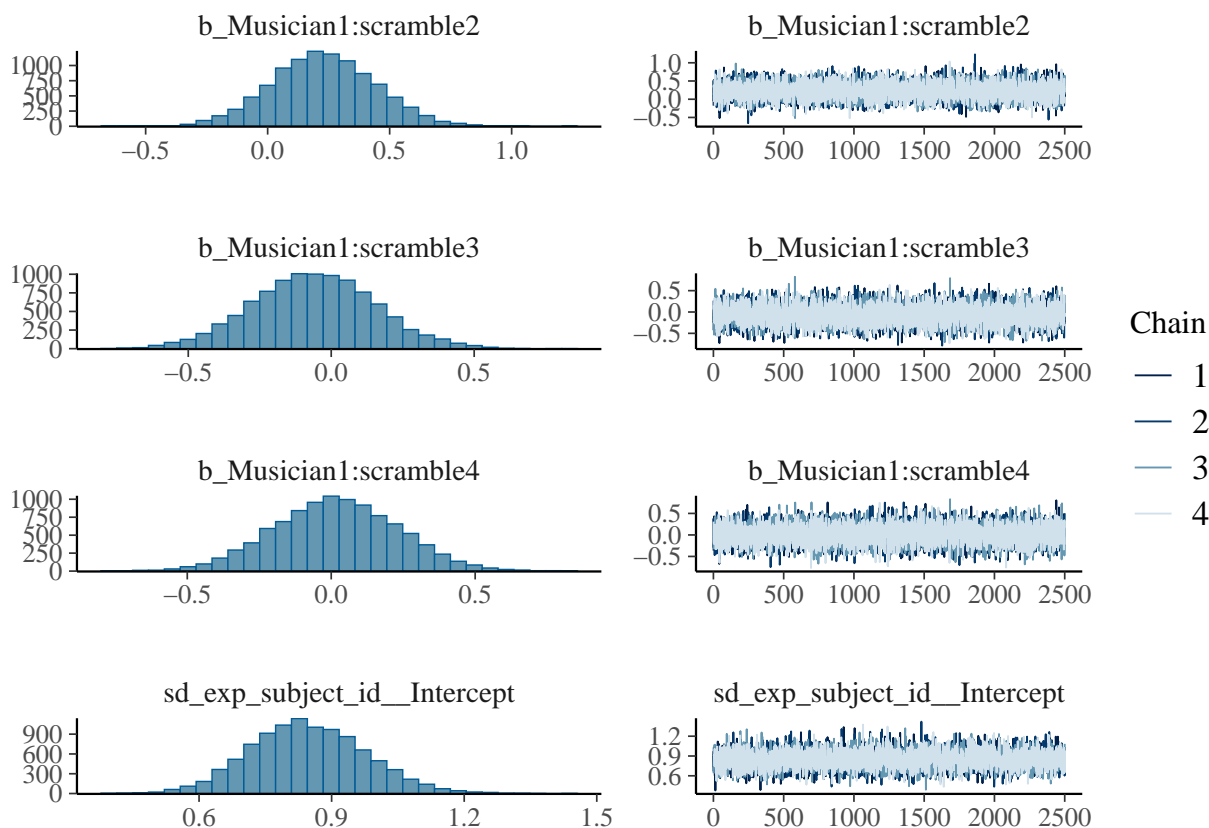
Interaction between group and condition?

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_scramble8B, prior_scramble2B, prior_scramble1B,  
            prior_int8B, prior_int2B, prior_int1B),  
  save_pars = save_pars(all = TRUE), iter = 5000,  
  file = '../models/E4_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: 825)
## Draws: 4 chains, each with iter = 5000; warmup = 2500; thin = 1;
## total post-warmup draws = 10000
##
## Multilevel Hyperparameters:
## ~exp_subject_id (Number of levels: 106)
## Estimate Est.Error 1-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## sd(Intercept) 0.8464 0.1302 0.6004 1.1098 0.9999 4374 7001
##
## Regression Coefficients:
## Estimate Est.Error 1-95% CI u-95% CI Rhat Bulk_ESS
## Intercept -0.7124 0.1765 -1.0630 -0.3682 1.0004 10147
## Musician1 -0.3640 0.1704 -0.6959 -0.0331 1.0001 8745
## scramble2 0.7178 0.2136 0.3057 1.1355 1.0001 13819
## scramble3 0.9115 0.2156 0.4918 1.3367 1.0009 12623
## scramble4 -0.1380 0.2176 -0.5631 0.2900 1.0003 13328
## Musician1:scramble2 0.2294 0.2120 -0.1802 0.6476 0.9999 12137
## Musician1:scramble3 -0.0606 0.2149 -0.4842 0.3681 1.0000 12146
## Musician1:scramble4 0.0167 0.2162 -0.4026 0.4329 1.0001 12339
## Tail_ESS
## Intercept 7950
## Musician1 8647
## scramble2 7953
```

```

## scramble3          7768
## scramble4          8491
## Musician1:scramble2 8211
## Musician1:scramble3 8641
## Musician1:scramble4 8669
##
## 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: 6
## 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: 0.02145
Strong evidence against an interaction between group and condition.

```

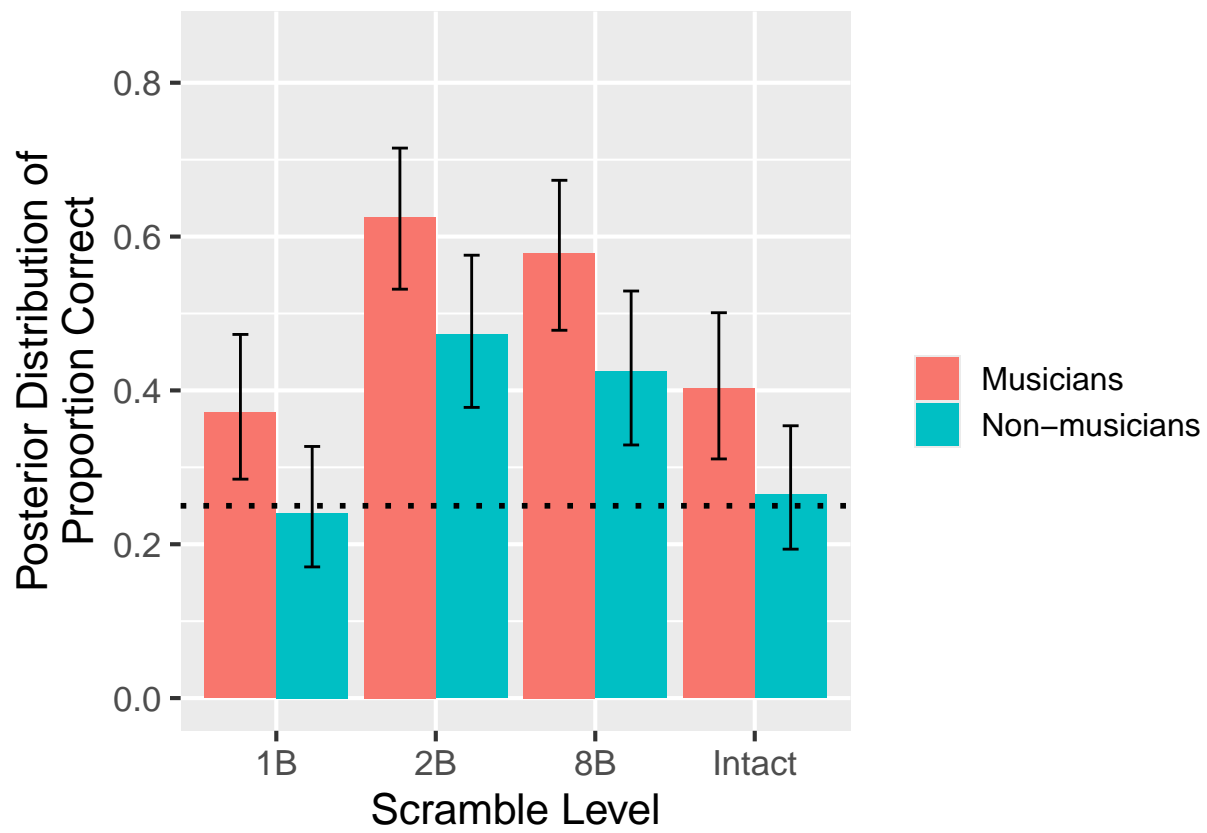
Figure 5

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 = 0.25, 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') +  
  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/fig5_categorization.png', width = 7, height = 5)
```

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

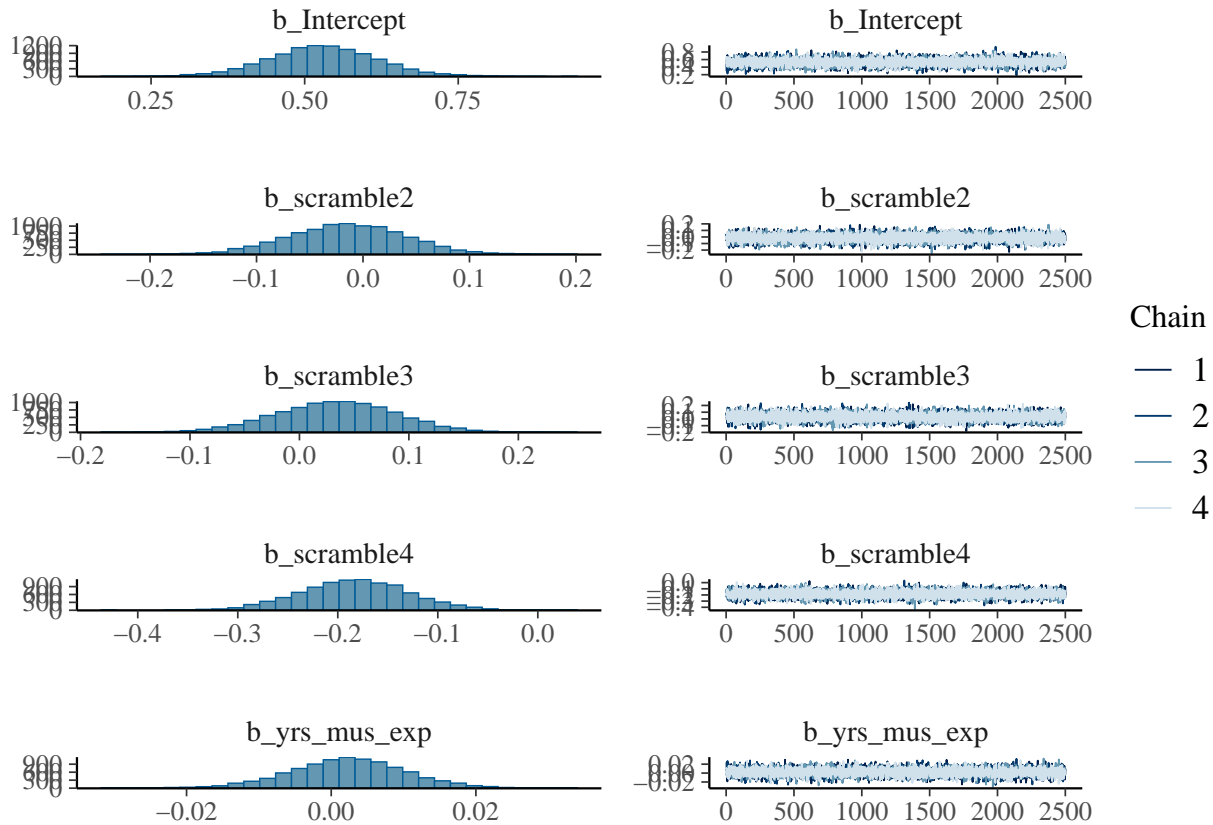
For this analysis, we're operating on the scale of accuracy. Because we don't see ceiling effects (i.e. participants aren't getting too close to perfect accuracy), a linear model is appropriate enough.

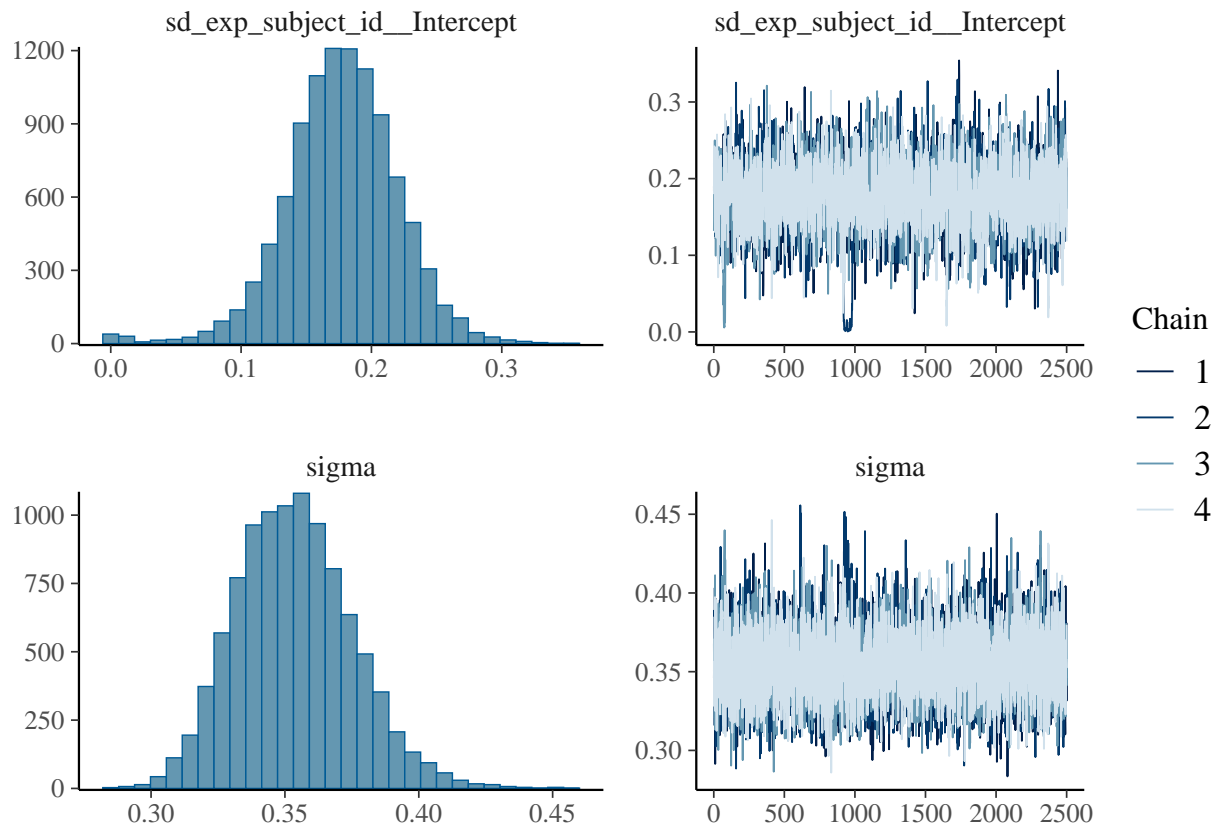
```
these_priors <- c(  
  set_prior('normal(0.625, 0.1)', class = 'Intercept'),  
  set_prior('normal(-0.1, 0.1)', coef = 'scramble2'),  
  set_prior('normal(-0.2, 0.1)', coef = 'scramble3'),  
  set_prior('normal(-0.3, 0.1)', coef = 'scramble4'),  
  set_prior('normal(0, 0.1)', coef = 'yrs_mus_exp')  
)
```

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/E4_years')
```

```
plot(years_mus_scram)
```





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

```
## Family: gaussian
## Links: mu = identity; sigma = identity
## Formula: accuracy ~ scramble + yrs_mus_exp + (1 | exp_subject_id)
## Data: yrs_exp (Number of observations: 203)
## Draws: 4 chains, each with iter = 5000; warmup = 2500; thin = 1;
## total post-warmup draws = 10000
##
## Multilevel Hyperparameters:
## ~exp_subject_id (Number of levels: 51)
##
```

	Estimate	Est.Error	l-95% CI	u-95% CI	Rhat	Bulk_ESS	Tail_ESS
sd(Intercept)	0.1771	0.0431	0.0894	0.2587	1.0023	1778	1018

```
##
## Regression Coefficients:
##
```

	Estimate	Est.Error	l-95% CI	u-95% CI	Rhat	Bulk_ESS	Tail_ESS
Intercept	0.5298	0.0876	0.3554	0.7064	1.0003	10476	7922
scramble2	-0.0150	0.0548	-0.1214	0.0909	0.9998	14722	8491
scramble3	0.0333	0.0552	-0.0760	0.1399	1.0002	14290	8461
scramble4	-0.1831	0.0544	-0.2901	-0.0761	0.9999	14396	7439
yrs_mus_exp	0.0023	0.0080	-0.0138	0.0180	1.0009	9663	7553

```
##
## Further Distributional Parameters:
##
```

	Estimate	Est.Error	l-95% CI	u-95% CI	Rhat	Bulk_ESS	Tail_ESS
sigma	0.3535	0.0219	0.3143	0.3999	1.0011	3858	3243

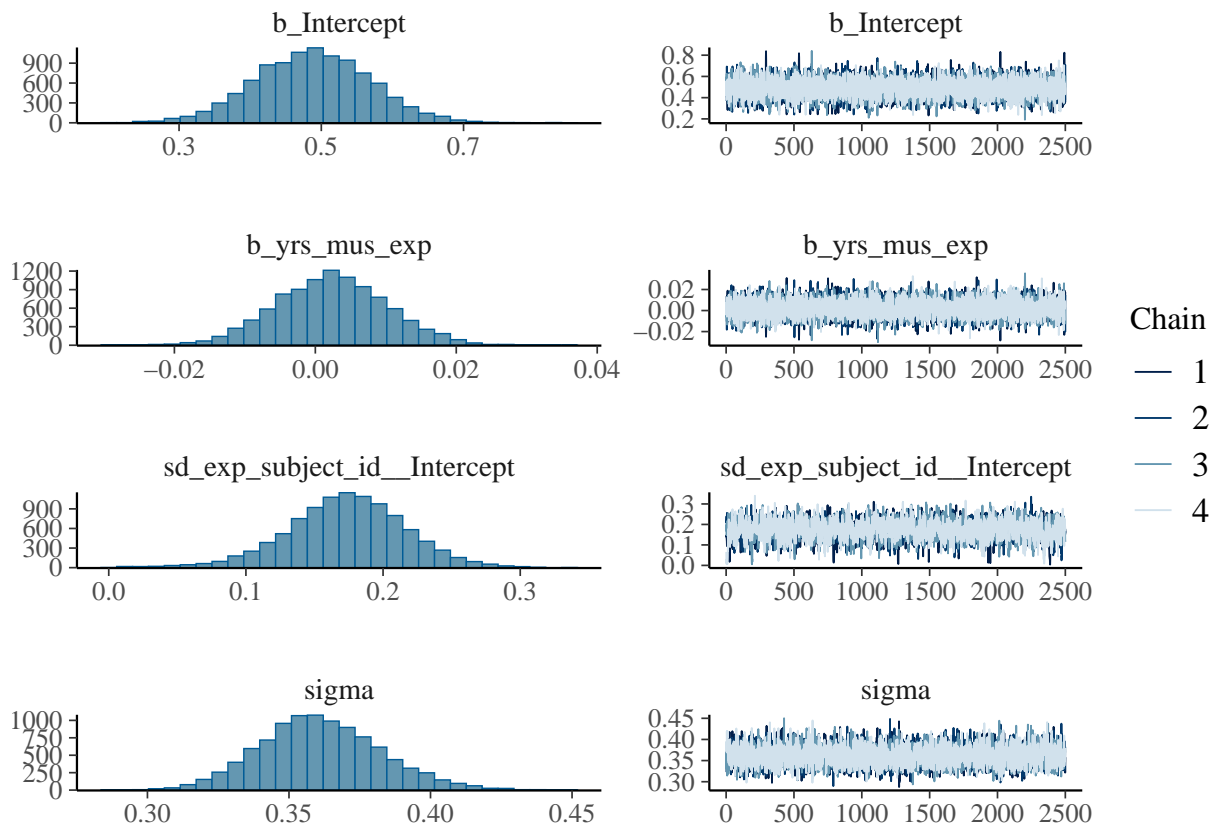
```
##
## Draws were sampled using sampling(NUTS). For each parameter, Bulk_ESS
## and Tail_ESS are effective sample size measures, and Rhat is the potential
```

```
## 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(0.625, 0.1)', class = 'Intercept'),
    set_prior('normal(0, 0.1)', coef = 'yrs_mus_exp')),
  save_pars = save_pars(all = TRUE), iter = 5000,
  file = '../models/E4_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: 203)
## Draws: 4 chains, each with iter = 5000; warmup = 2500; thin = 1;
## total post-warmup draws = 10000
##
## Multilevel Hyperparameters:
## ~exp_subject_id (Number of levels: 51)
## Estimate Est.Error 1-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## sd(Intercept) 0.1731 0.0435 0.0822 0.2559 1.0003 2325 2914
##
## Regression Coefficients:
```

```

##           Estimate Est.Error l-95% CI u-95% CI   Rhat Bulk_ESS Tail_ESS
## Intercept      0.4880   0.0811   0.3296   0.6485 1.0003     9029     7534
## yrs_mus_exp    0.0023   0.0079  -0.0130   0.0180 1.0002     9390     7531
##
## Further Distributional Parameters:
##           Estimate Est.Error l-95% CI u-95% CI   Rhat Bulk_ESS Tail_ESS
## sigma    0.3613    0.0214   0.3218   0.4060 1.0000     6420     5982
##
## 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)	0.5294	0.0859	[0.34, 0.69]	1.82e+04	1.000	10394.0000
## scramble2	-0.0144	0.0550	[-0.12, 0.09]	0.348	1.000	14888.0000
## scramble3	0.0342	0.0551	[-0.07, 0.14]	0.093	1.000	14482.0000
## scramble4	-0.1828	0.0543	[-0.30, -0.08]	1.11	1.000	14173.0000
## yrs_mus_exp	0.0024	0.0077	[-0.01, 0.02]	0.080	1.000	9558.0000

```
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

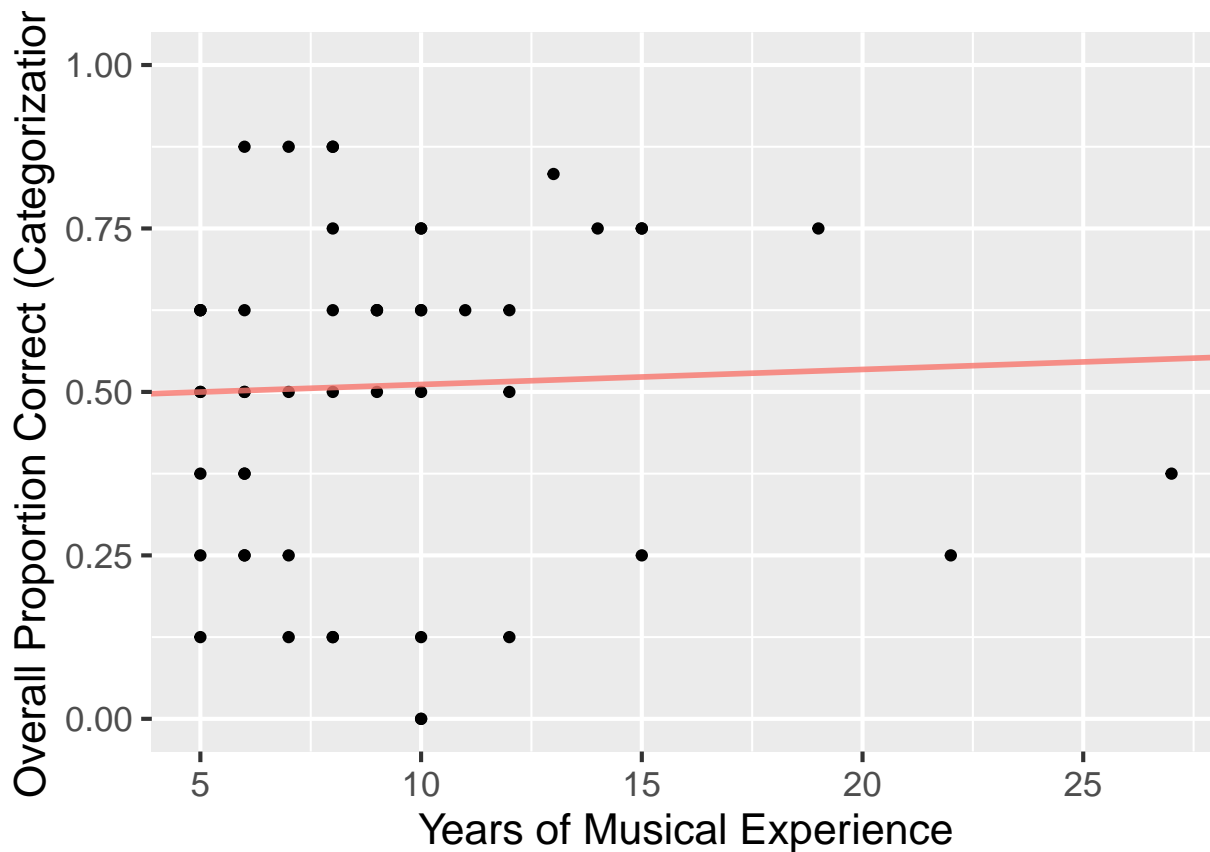
## Parameter	Median	MAD	95% CI	BF	Rhat	ESS
## (Intercept)	0.4883	0.0814	[0.33, 0.65]	1.51e+04	1.000	8987.0000
## yrs_mus_exp	0.0023	0.0079	[-0.01, 0.02]	0.083	1.000	9381.0000

Strong evidence against an effect of years of musical experience.

Figure S1C

```
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 (Categorization)') +
  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/FigS1C_categorization.png', width = 5, height = 5)
```