E1 memory

R. Cassano-Coleman

2025-08-18

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

Set up

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

```
data <- read_csv('../data/E1-E2-E4/memory.csv')
```

```
## Rows: 3180 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.

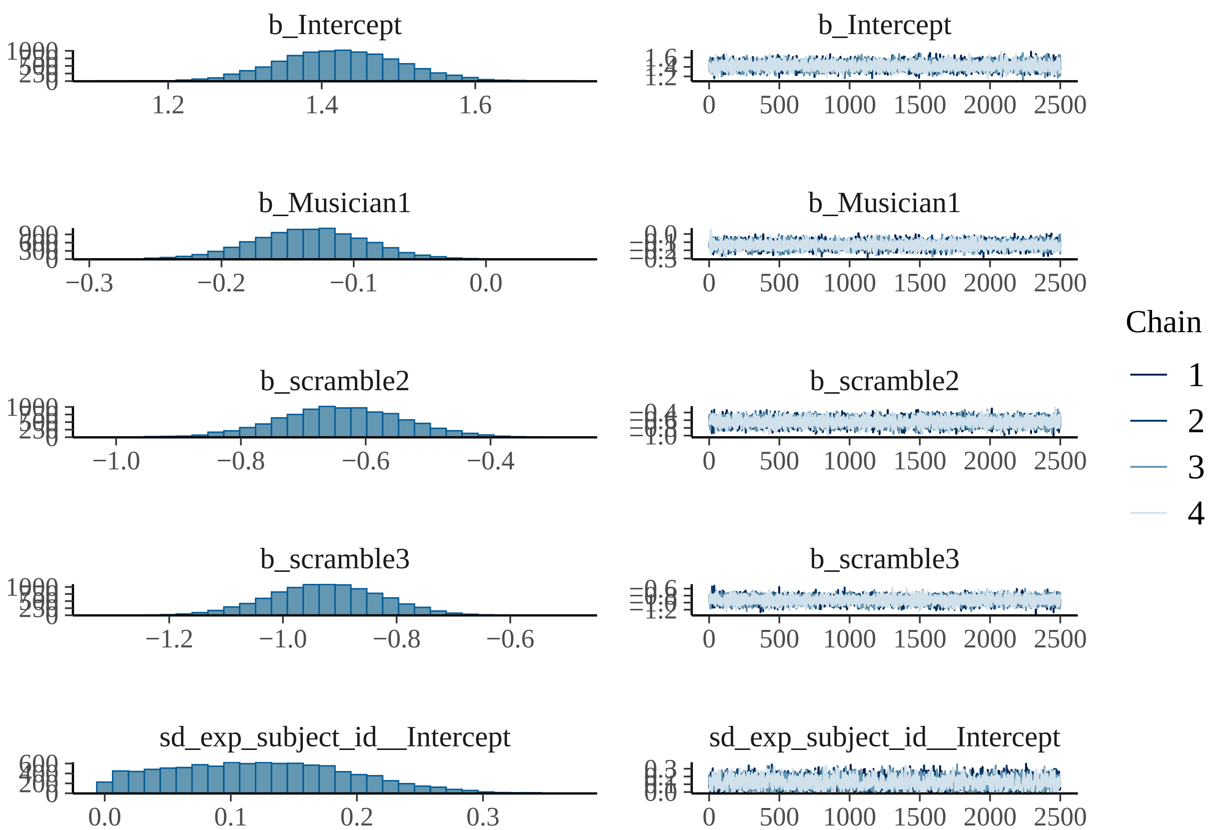
```
prior_int2B <- set_prior('normal(0, 1)', coef = 'Musician1:scramble2')
prior_int1B <- set_prior('normal(0, 1)', coef = 'Musician1:scramble3')
```

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

Main model with group and condition

```r
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 = 5000,
                 file = 'models/E1_mus_scram')
```

```r
plot(mus_scram)
```




```r
print(summary(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: bernoulli
##   Links: mu = logit
## Formula: response ~ Musician + scramble + (1 | exp_subject_id)
##    Data: data (Number of observations: 3153)
##   Draws: 4 chains, each with iter = 5000; warmup = 2500; thin = 1;
##          total post-warmup draws = 10000
##
## Multilevel Hyperparameters:
## ~exp_subject_id (Number of levels: 102)
##               Estimate Est.Error l-95% CI u-95% CI   Rhat Bulk_ESS Tail_ESS
## sd(Intercept)   0.1226    0.0706   0.0069   0.2655 1.0007     2427     4004
##
```

```
## Regression Coefficients:
##           Estimate Est.Error l-95% CI u-95% CI   Rhat Bulk_ESS Tail_ESS
## Intercept   1.4241    0.0790   1.2742   1.5821 1.0001     8603     7224
## Musician1  -0.1336    0.0429  -0.2182  -0.0500 1.0000    10909     6935
## scramble2  -0.6391    0.1007  -0.8379  -0.4433 1.0004     9841     7805
## scramble3  -0.9246    0.0987  -1.1192  -0.7340 1.0006    10169     7911
##
## 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        1.423     1.277     1.584
##  2B        0.785     0.658     0.927
##  1B        0.500     0.376     0.628
##
## 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     0.640     0.442     0.836
##  8B - 1B     0.925     0.732     1.116
##  2B - 1B     0.285     0.110     0.466
##
## 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        1.557     1.388     1.746
##  No       8B        1.289     1.121     1.467
##  Yes      2B        0.919     0.766     1.085
##  No       2B        0.652     0.488     0.804
##  Yes      1B        0.633     0.480     0.783
##  No       1B        0.365     0.208     0.512
##
## 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    0.2662     0.101     0.437
##  Yes 8B - Yes 2B   0.6402     0.442     0.836
##  Yes 8B - No 2B    0.9051     0.655     1.181
```

```
##  Yes 8B - Yes 1B   0.9247     0.732     1.116
##  Yes 8B - No 1B    1.1920     0.940     1.460
##  No 8B - Yes 2B    0.3705     0.117     0.627
##  No 8B - No 2B     0.6402     0.442     0.836
##  No 8B - Yes 1B    0.6575     0.416     0.919
##  No 8B - No 1B     0.9247     0.732     1.116
##  Yes 2B - No 2B    0.2662     0.101     0.437
##  Yes 2B - Yes 1B   0.2845     0.110     0.466
##  Yes 2B - No 1B    0.5528     0.315     0.807
##  No 2B - Yes 1B    0.0201    -0.221     0.265
##  No 2B - No 1B     0.2845     0.110     0.466
##  Yes 1B - No 1B    0.2662     0.101     0.437
##
## 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"))
```

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

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

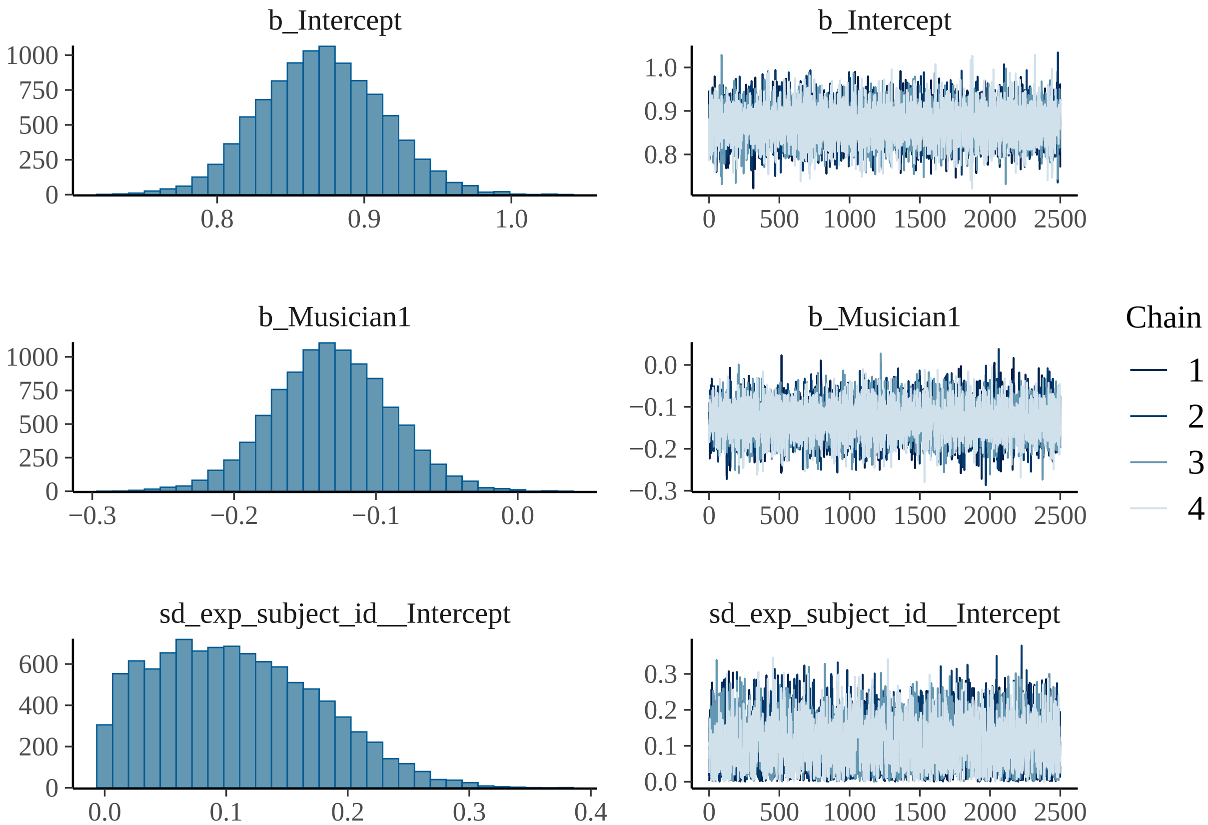
```
## Summary of Posterior Distribution
##
## Parameter   |  Median |    MAD |         95% CI |       BF |  Rhat |        ESS
## --------------------------------------------------------------------------------
## (Intercept) |  1.4226 | 0.0794 | [ 1.28,  1.58] | 1.41e+19 | 1.000 |  8576.0000
## Musician1   | -0.1331 | 0.0428 | [-0.22, -0.05] |     4.49 | 1.000 | 10880.0000
## scramble2   | -0.6402 | 0.1010 | [-0.84, -0.44] | 3.74e+05 | 1.000 |  9833.0000
## scramble3   | -0.9247 | 0.0999 | [-1.12, -0.73] | 9.22e+07 | 1.000 | 10152.0000
```

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/E1_mus_only')
```

```
plot(mus_only)
```




```
print(summary(mus_only), 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 + (1 | exp_subject_id)
##    Data: data (Number of observations: 3153)
##   Draws: 4 chains, each with iter = 5000; warmup = 2500; thin = 1;
##          total post-warmup draws = 10000
##
## Multilevel Hyperparameters:
## ~exp_subject_id (Number of levels: 102)
##               Estimate Est.Error l-95% CI u-95% CI   Rhat Bulk_ESS Tail_ESS
## sd(Intercept)   0.1084    0.0668   0.0057   0.2485 1.0006     2728     5074
##
## Regression Coefficients:
```

```
##           Estimate Est.Error l-95% CI u-95% CI   Rhat Bulk_ESS Tail_ESS
## Intercept   0.8708    0.0412   0.7918   0.9521 1.0006    12542     6331
## Musician1  -0.1314    0.0412  -0.2125  -0.0505 1.0008    12688     6331
##
## 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_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
```

```r
print(BF_scramble)
```

```
## Estimated Bayes factor in favor of mus_scram over mus_only: 477225714046893824.00000
```

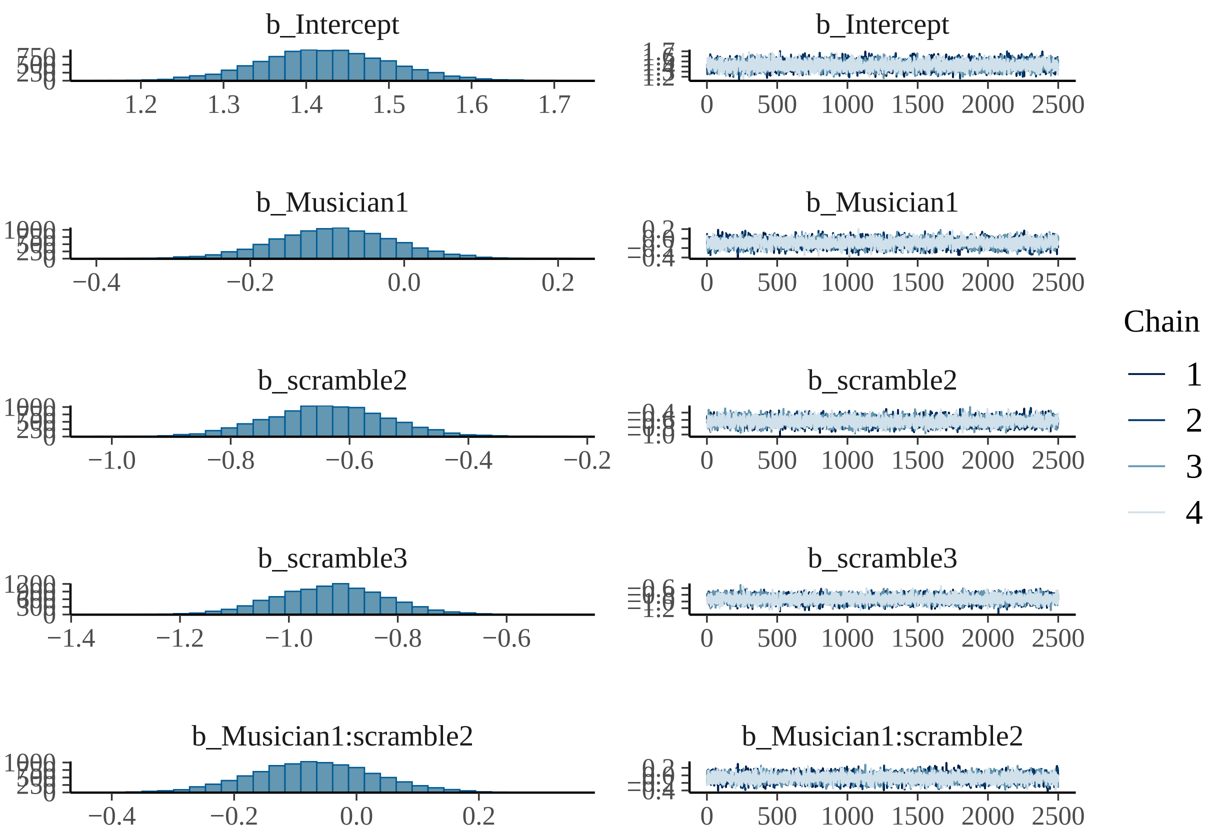
Very strong evidence for a main effect of scramble condition.

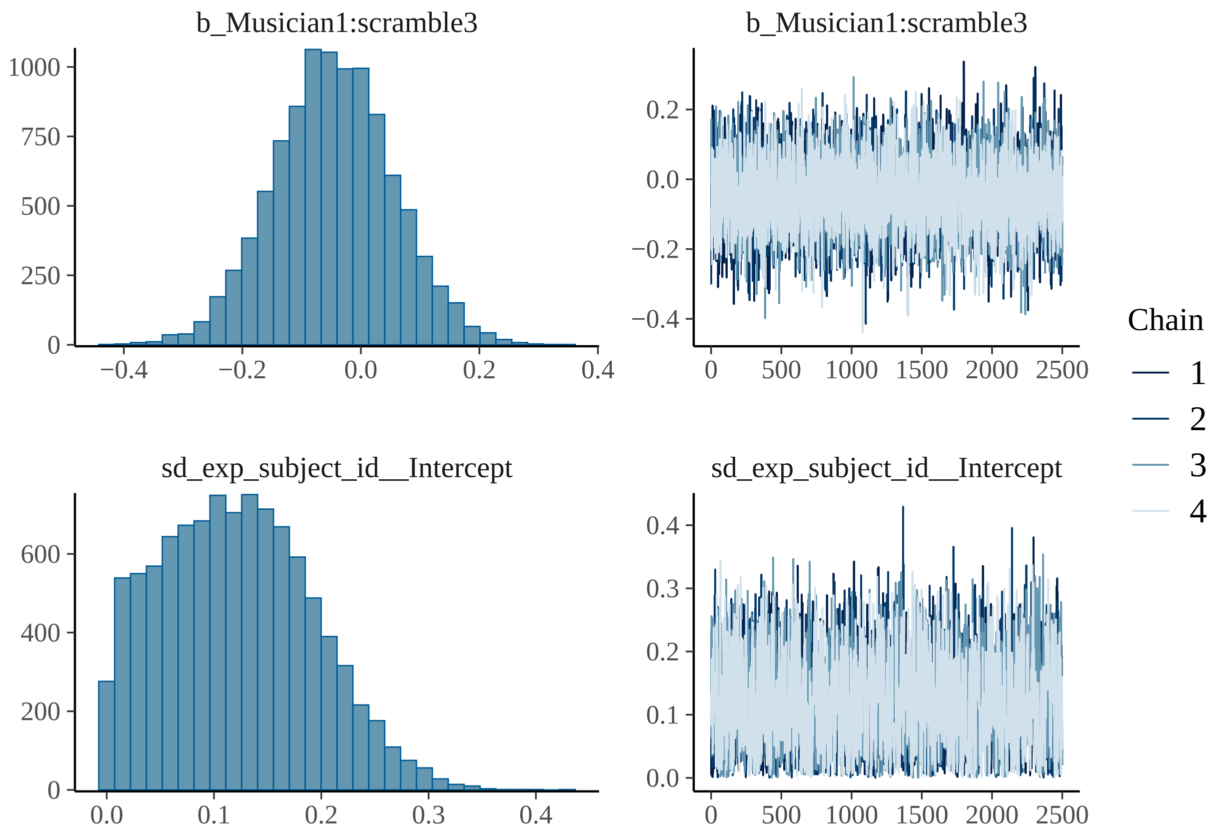
Interaction between group and condition?

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/E1_mus_scram_int')
```

```
plot(mus_scram_int)
```






```r
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: 3153)
##   Draws: 4 chains, each with iter = 5000; warmup = 2500; thin = 1;
##          total post-warmup draws = 10000
##
## Multilevel Hyperparameters:
## ~exp_subject_id (Number of levels: 102)
##               Estimate Est.Error l-95% CI u-95% CI   Rhat Bulk_ESS Tail_ESS
## sd(Intercept)   0.1206    0.0702   0.0068   0.2647 1.0006     2442     3874
##
## Regression Coefficients:
##                     Estimate Est.Error l-95% CI u-95% CI   Rhat Bulk_ESS
## Intercept             1.4234    0.0790   1.2683   1.5812 1.0003     7138
## Musician1            -0.0905    0.0787  -0.2439   0.0638 1.0009     4546
## scramble2            -0.6368    0.1019  -0.8360  -0.4381 1.0006     8478
## scramble3            -0.9228    0.1014  -1.1231  -0.7247 1.0004     8557
## Musician1:scramble2  -0.0694    0.1012  -0.2654   0.1317 1.0010     5442
## Musician1:scramble3  -0.0462    0.1008  -0.2423   0.1542 1.0007     5132
##                     Tail_ESS
## Intercept               6939
## Musician1               6148
## scramble2               7601
## scramble3               7906
## Musician1:scramble2     6896
```

```
## Musician1:scramble3     6924
##
## 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
## Iteration: 6
```

```r
print(BF_int)
```

```
## Estimated Bayes factor in favor of mus_scram_int over mus_scram: 0.01095
```

Strong evidence against an interaction between group and condition.

Figure 2A

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

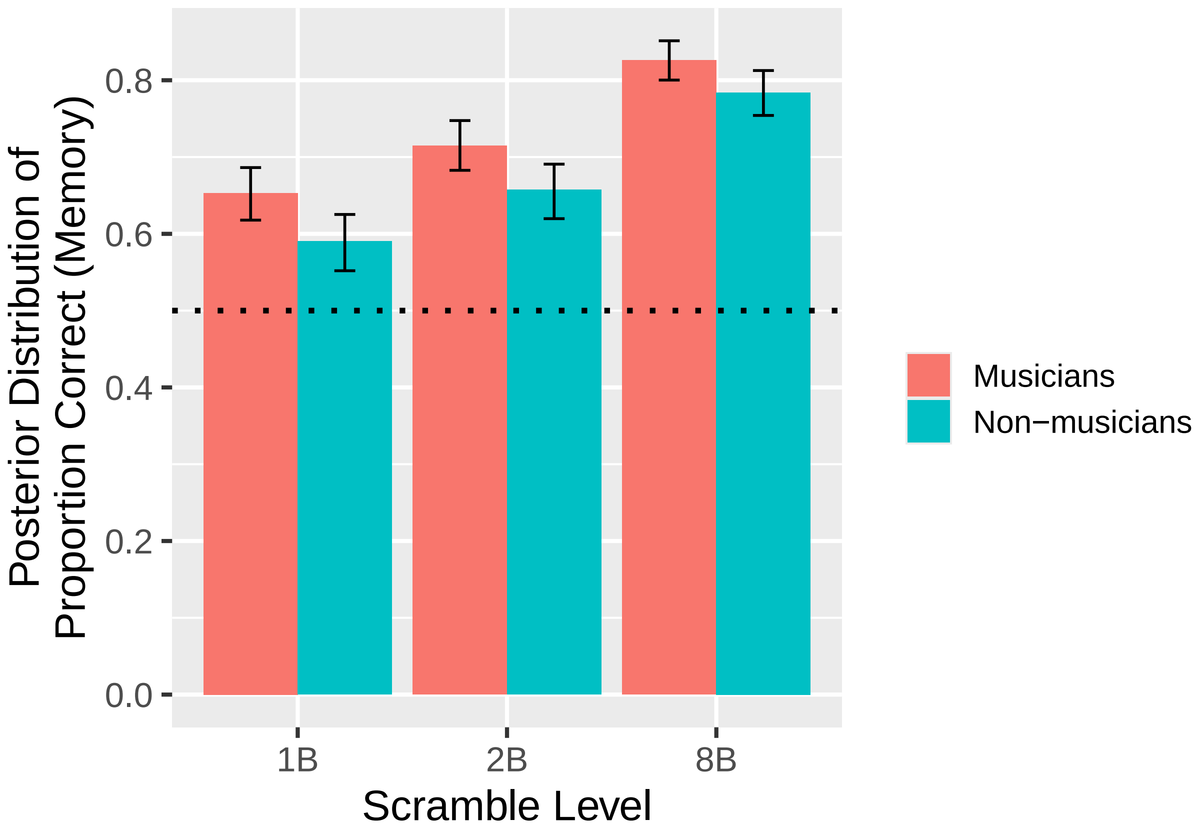
```
calculate_prob_from_logodds <- function(logodds) {
  return(exp(logodds) / (1 + exp(logodds)))
}
```

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 (Memory)') +
  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/Fig2A_memory.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

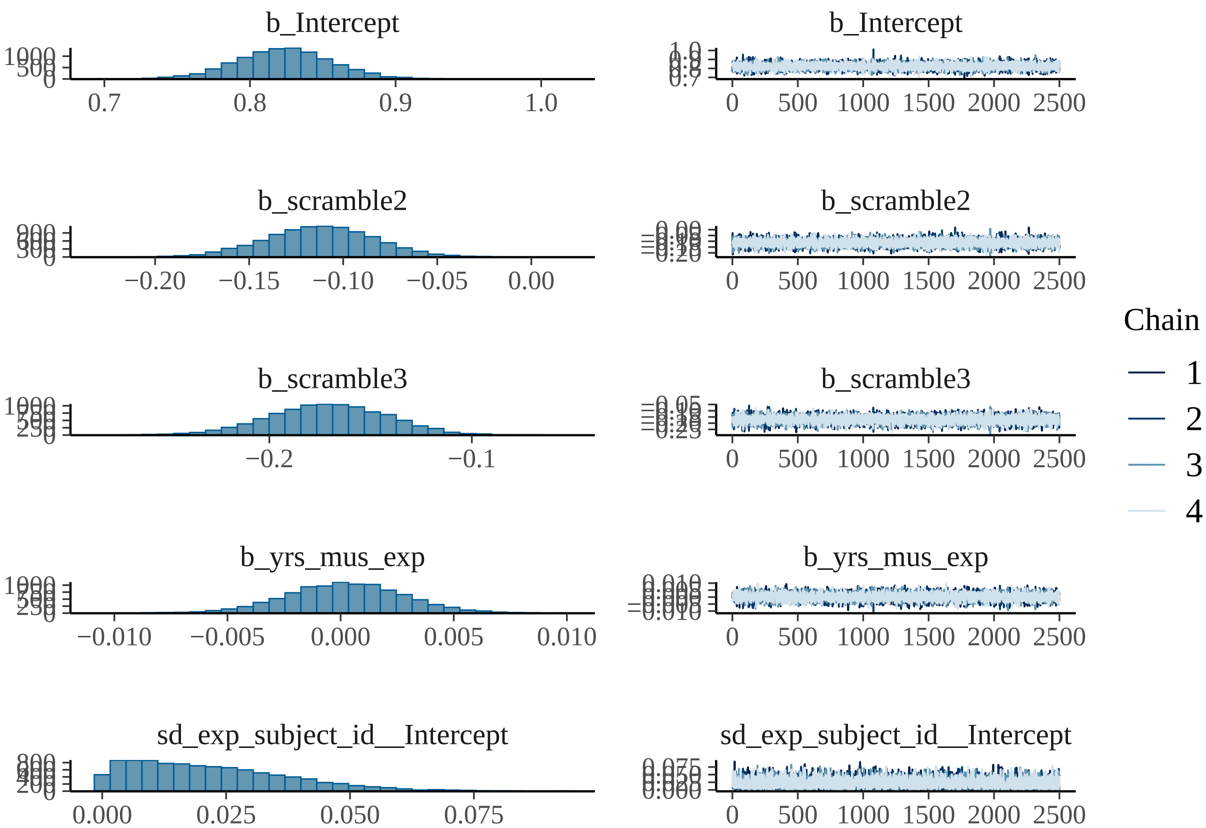
For this analysis, we're operating on the scale of accuracy. Because we don't see ceiling effects (i.e. participants aren't getting too close to perfect accuracy), a linear model is appropriate enough.

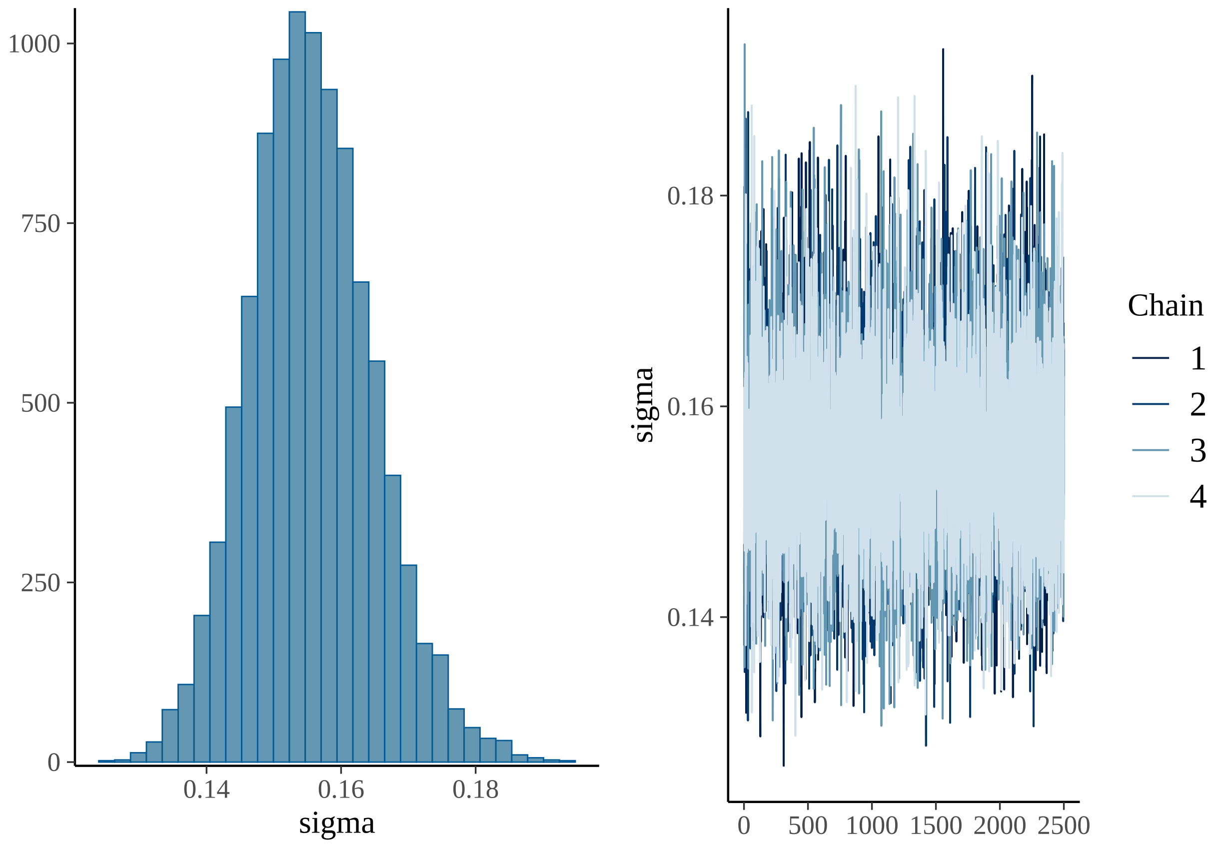
```
these_priors <- c(
  set_prior('normal(0.75, 0.1)', class = 'Intercept'),
  set_prior('normal(-0.1, 0.1)', coef = 'scramble2'),
  set_prior('normal(-0.2, 0.1)', coef = 'scramble3'),
  set_prior('normal(0, 0.1)', coef = 'yrs_mus_exp')
)
```

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/E1_years')
```

```
plot(years_mus_scram)
```






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

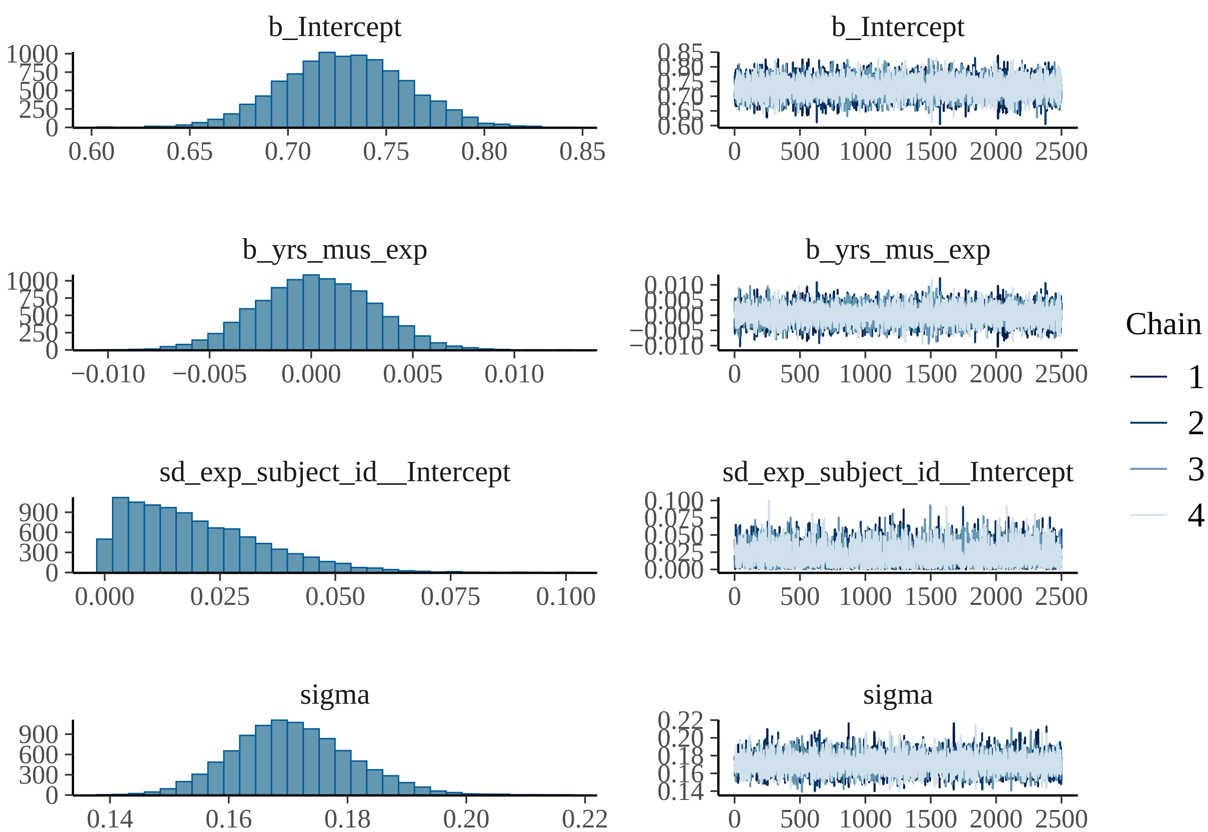
```
##  Family: gaussian
##   Links: mu = identity; sigma = identity
## Formula: accuracy ~ scramble + yrs_mus_exp + (1 | exp_subject_id)
##    Data: yrs_exp (Number of observations: 153)
##   Draws: 4 chains, each with iter = 5000; warmup = 2500; thin = 1;
##          total post-warmup draws = 10000
##
## 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)   0.0220    0.0157   0.0009   0.0579 1.0005     4437     5268
##
## Regression Coefficients:
##             Estimate Est.Error l-95% CI u-95% CI   Rhat Bulk_ESS Tail_ESS
## Intercept     0.8234    0.0328   0.7581   0.8873 1.0001    14119     7938
## scramble2    -0.1122    0.0290  -0.1685  -0.0561 1.0001    13447     7999
## scramble3    -0.1710    0.0291  -0.2290  -0.1162 1.0005    13906     8001
## yrs_mus_exp   0.0002    0.0026  -0.0049   0.0053 0.9999    14935     6904
##
## Further Distributional Parameters:
##       Estimate Est.Error l-95% CI u-95% CI   Rhat Bulk_ESS Tail_ESS
## sigma   0.1556    0.0092   0.1385   0.1751 1.0012    11631     6456
##
## 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 = 5000,
                 file = 'models/E1_years_null')
```

```r
plot(years_mus)
```




```r
print(summary(years_mus), 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: gaussian
##   Links: mu = identity; sigma = identity
## Formula: accuracy ~ yrs_mus_exp + (1 | exp_subject_id)
##    Data: yrs_exp (Number of observations: 153)
##   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)   0.0198    0.0146   0.0008   0.0537 1.0002     5713     5734
##
## Regression Coefficients:
##             Estimate Est.Error l-95% CI u-95% CI   Rhat Bulk_ESS Tail_ESS
## Intercept     0.7289    0.0315   0.6682   0.7898 1.0003    15334     7271
## yrs_mus_exp   0.0002    0.0029  -0.0053   0.0057 1.0008    16567     7221
##
## Further Distributional Parameters:
##       Estimate Est.Error l-95% CI u-95% CI   Rhat Bulk_ESS Tail_ESS
## sigma   0.1708    0.0100   0.1523   0.1915 1.0003    14074     6869
##
## 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"))
```

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

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

```
## Summary of Posterior Distribution
##
## Parameter   |  Median |    MAD |          95% CI |       BF |  Rhat |        ESS
## ----------------------------------------------------------------------------------
## (Intercept) |  0.8233 | 0.0321 | [ 0.76,  0.89] | 3.85e+31 | 1.000 | 13977.0000
## scramble2   | -0.1119 | 0.0287 | [-0.17, -0.06] |    90.86 | 1.000 | 13386.0000
## scramble3   | -0.1705 | 0.0294 | [-0.23, -0.11] | 1.02e+04 | 1.000 | 13855.0000
## yrs_mus_exp |  0.0002 | 0.0026 | [-0.01,  0.01] |    0.025 | 1.000 | 14616.0000
```

```r
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.
```

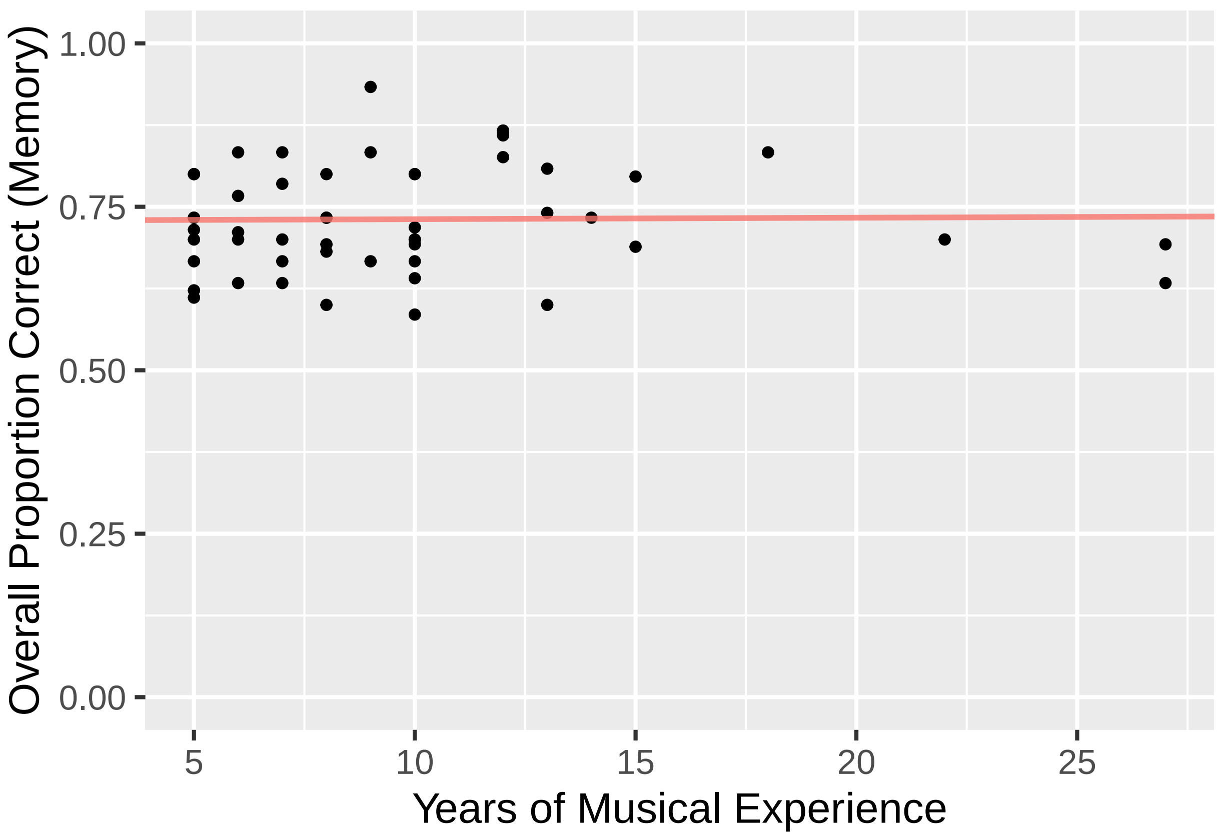
```r
print(yrs_null_BF, digits = 4)
```

```
## Summary of Posterior Distribution
##
## Parameter   | Median |    MAD |        95% CI |       BF |  Rhat |        ESS
## ------------------------------------------------------------------------------
## (Intercept) | 0.7288 | 0.0316 | [ 0.67, 0.79] | 2.05e+21 | 1.000 | 15454.0000
## yrs_mus_exp | 0.0002 | 0.0029 | [-0.01, 0.01] |    0.028 | 1.000 | 16739.0000
```

Figure S1A

```r
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 (Memory)') +
  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.
```




```r
ggsave('../figures/FigS1A_memory.png', width = 5, height = 5)
```