

model2

Lab B Team 2

```
spa3<-read.csv('spa3.csv')

eng3<-read.csv('eng3.csv')

#logistic regression model for english data:
M1<-glm(accuracy~cognate, family=binomial(link="logit"),data=eng3)
summary(M1)

##
## Call:
## glm(formula = accuracy ~ cognate, family = binomial(link = "logit"),
##      data = eng3)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -1.0842  -1.0842  -0.9424   1.2735   1.4322
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.58151    0.08555  -6.797 1.07e-11 ***
## cognate      0.35836    0.11890   3.014 0.00258 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 1600.5  on 1187  degrees of freedom
## Residual deviance: 1591.4  on 1186  degrees of freedom
## AIC: 1595.4
##
## Number of Fisher Scoring iterations: 4

#Intercept and coefficient for cognate:
inv.logit(-0.58)

## [1] 0.3589326

inv.logit(-0.58+0.35)

## [1] 0.4427521

#logistic regression model for spanish data:
M2<-glm(accuracy~cognate, family=binomial(link="logit"),data=spa3)
summary(M2)

##
## Call:
```

```
## glm(formula = accuracy ~ cognate, family = binomial(link = "logit"),
##     data = spa3)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -0.9341  -0.9341  -0.7004   1.4421   1.7466
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -1.28007    0.09953 -12.861  < 2e-16 ***
## cognate      0.67654    0.13143   5.148 2.64e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 1420.0  on 1186  degrees of freedom
## Residual deviance: 1392.9  on 1185  degrees of freedom
## (1 observation deleted due to missingness)
## AIC: 1396.9
##
## Number of Fisher Scoring iterations: 4
```

```
inv.logit(-1.28)
```

```
## [1] 0.2175502
```

```
inv.logit(-1.28+0.67)
```

```
## [1] 0.3520592
```

Add random variable into logistic model

```
M3 <- glmer(accuracy~1+(1|cognate),data=eng3,family=binomial(link="logit"))
print(M3)
```

```
## Generalized linear mixed model fit by maximum likelihood (Laplace
## Approximation) [glmerMod]
## Family: binomial ( logit )
## Formula: accuracy ~ 1 + (1 | cognate)
## Data: eng3
##      AIC      BIC    logLik deviance df.resid
## 1600.4328 1610.5929 -798.2164 1596.4328    1186
## Random effects:
## Groups Name          Std.Dev.
## cognate (Intercept) 0.1585
## Number of obs: 1188, groups: cognate, 2
## Fixed Effects:
## (Intercept)
##      -0.4016
```

```
M4 <- glmer(accuracy~1+(1|cognate),data=spa3,family=binomial(link="logit"))
print(M4)
```

```
## Generalized linear mixed model fit by maximum likelihood (Laplace
## Approximation) [glmerMod]
```

```
## Family: binomial ( logit )
## Formula: accuracy ~ 1 + (1 | cognate)
## Data: spa3
##      AIC      BIC    logLik deviance df.resid
## 1404.1287 1414.2871 -700.0644 1400.1287    1185
## Random effects:
## Groups Name      Std.Dev.
## cognate (Intercept) 0.326
## Number of obs: 1187, groups: cognate, 2
## Fixed Effects:
## (Intercept)
##      -0.94
```

refit the model with subject levels

```
M5 <- glmer(accuracy~cognate+(1|subject)+diff,data=eng3,family=binomial(link="logit"))
print(summary(M5))
```

```
## Generalized linear mixed model fit by maximum likelihood (Laplace
## Approximation) [glmerMod]
## Family: binomial ( logit )
## Formula: accuracy ~ cognate + (1 | subject) + diff
## Data: eng3
##
##      AIC      BIC    logLik deviance df.resid
##    947.9    968.2   -469.9    939.9    1184
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -4.9507 -0.3735 -0.0971  0.3559  5.2318
##
## Random effects:
## Groups Name      Variance Std.Dev.
## subject (Intercept) 6.617    2.572
## Number of obs: 1188, groups: subject, 27
##
## Fixed effects:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  2.08652    0.56691   3.681 0.000233 ***
## cognate      0.74409    0.17385   4.280 1.87e-05 ***
## diff        -0.88052    0.07006 -12.568 < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##      (Intr) cognat
## cognate -0.096
## diff    -0.410 -0.153
```

```
inv.logit(2.08652)
```

```
## [1] 0.8895861
```

```
inv.logit(2.08652+0.74409)
```

```
## [1] 0.9443077
```

```
inv.logit(2.08652-0.8832)
```

```
## [1] 0.7691149
```

Interpret: Intercept: For non-cognate words with same difficulty level, the possibility of answer the words accurately is 88.96%. Cognate: Words with same difficulty level, cognate words tend to have 94.43% higher possibility to be correctly answered. diff: For non-Cognates word, it has 76.91% higher possibility to be correctly answered as the difficulty increasing.

```
M6 <- glmer(accuracy~cognate+(1|subject)+diff,data=spa3,family=binomial(link="logit"))
print(summary(M6))
```

```
## Generalized linear mixed model fit by maximum likelihood (Laplace
## Approximation) [glmerMod]
## Family: binomial ( logit )
## Formula: accuracy ~ cognate + (1 | subject) + diff
## Data: spa3
##
##      AIC      BIC   logLik deviance df.resid
##    965.1    985.4   -478.5    957.1     1183
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -4.0504 -0.4155 -0.1654  0.3529  6.0991
##
## Random effects:
##  Groups Name      Variance Std.Dev.
## subject (Intercept) 4.1      2.025
## Number of obs: 1187, groups: subject, 27
##
## Fixed effects:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  0.84823    0.46092   1.840  0.0657 .
## cognate      1.16385    0.17533   6.638 3.18e-11 ***
## diff        -0.83898    0.06712 -12.500 < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##          (Intr) cognat
## cognate -0.112
## diff    -0.402 -0.256
```

Interpret: Intercept: For non-cognate words with same difficulty level, the possibility of answer the words accurately is 74.17%. Cognate: Words with same difficulty level, cognate words tend to have 76.195% higher possibility to be correctly answered. diff: For non-Cognates word, it has 30.2% higher possibility to be correctly answered as the difficulty increasing.

```
inv.logit(0.84823)
```

```
## [1] 0.7001957
```

```
inv.logit(0.84823+1.16385)
```

```
## [1] 0.8820596
```

```
inv.logit(0.84823-0.83898)
```

```
## [1] 0.5023125
```

Interpret: Intercept: For non-cognate words with same difficulty level, the possibility of answer the words accurately is 70.02%. Cognate: Words with same difficulty level, cognate words tend to have 88.20% higher possibility to be correctly answered. diff: For non-Cognates word, it has 50.23% higher possibility to be correctly answered as the difficulty increasing.

Mixed effect logistic regression

```
#comb <- read.csv("comb.csv",header=T)
#comb <- unite(comb,"accuracy",c("eng.acc","spa.acc"),sep="",remove = F)
#comb<- comb[-115,]
#comb$category <- rep(NA,1143)
#for (i in 1:1143){
  # if (comb$eng.acc[i] == 0 && comb$spa.acc[i] == 0) {
    # comb$category[i] <- 1
  #}
  #if (comb$eng.acc[i] == 0 && comb$spa.acc[i] == 1) {
    # comb$category[i] <- 2
  #}
  ## if (comb$eng.acc[i] == 1 && comb$spa.acc[i] == 0) {
    # comb$category[i] <- 3
  #}
  # if (comb$eng.acc[i] == 1 && comb$spa.acc[i] == 1) {
    # comb$category[i] <- 4
  # }
}

#multinomial model
#M7 <- polr(factor(category)~cognate+diff,data=comb)
#print(summary(M7))
```

Try Multinomial analysis with brms

Note: One NA in spa3: no accuracy for word “knocker” (ID: BUBA46).

```
M7 <- brm(accuracy~cognate+(1|subject)+diff,data=spa3,family=bernoulli,prior = c(set_prior("normal(0,8)

## Warning: Rows containing NAs were excluded from the model.

## Compiling Stan program...

## Trying to compile a simple C file

## Running /Library/Frameworks/R.framework/Resources/bin/R CMD SHLIB foo.c
## clang -I"/Library/Frameworks/R.framework/Resources/include" -DNDEBUG -I"/Library/Frameworks/R.frame
## In file included from <built-in>:1:
## In file included from /Library/Frameworks/R.framework/Versions/4.0/Resources/library/StanHeaders/inc
## In file included from /Library/Frameworks/R.framework/Versions/4.0/Resources/library/RcppEigen/inclu
## In file included from /Library/Frameworks/R.framework/Versions/4.0/Resources/library/RcppEigen/inclu
## /Library/Frameworks/R.framework/Versions/4.0/Resources/library/RcppEigen/include/Eigen/src/Core/util
## namespace Eigen {
## ^
## /Library/Frameworks/R.framework/Versions/4.0/Resources/library/RcppEigen/include/Eigen/src/Core/util
## namespace Eigen {
## ^
```

```

##          ;
## In file included from <built-in>:1:
## In file included from /Library/Frameworks/R.framework/Versions/4.0/Resources/library/StanHeaders/inc.
## In file included from /Library/Frameworks/R.framework/Versions/4.0/Resources/library/RcppEigen/inclu
## /Library/Frameworks/R.framework/Versions/4.0/Resources/library/RcppEigen/include/Eigen/Core:96:10: f
## #include <complex>
##          ~~~~~
## 3 errors generated.
## make: *** [foo.o] Error 1

## Start sampling

##
## SAMPLING FOR MODEL 'fecdb0ea2e3aab30d18291906b136a6d' NOW (CHAIN 1).
## Chain 1:
## Chain 1: Gradient evaluation took 0.000167 seconds
## Chain 1: 1000 transitions using 10 leapfrog steps per transition would take 1.67 seconds.
## Chain 1: Adjust your expectations accordingly!
## Chain 1:
## Chain 1:
## Chain 1: Iteration:    1 / 2000 [  0%] (Warmup)
## Chain 1: Iteration:   200 / 2000 [ 10%] (Warmup)
## Chain 1: Iteration:   400 / 2000 [ 20%] (Warmup)
## Chain 1: Iteration:   600 / 2000 [ 30%] (Warmup)
## Chain 1: Iteration:   800 / 2000 [ 40%] (Warmup)
## Chain 1: Iteration:  1000 / 2000 [ 50%] (Warmup)
## Chain 1: Iteration:  1001 / 2000 [ 50%] (Sampling)
## Chain 1: Iteration:  1200 / 2000 [ 60%] (Sampling)
## Chain 1: Iteration:  1400 / 2000 [ 70%] (Sampling)
## Chain 1: Iteration:  1600 / 2000 [ 80%] (Sampling)
## Chain 1: Iteration:  1800 / 2000 [ 90%] (Sampling)
## Chain 1: Iteration:  2000 / 2000 [100%] (Sampling)
## Chain 1:
## Chain 1: Elapsed Time: 1.85899 seconds (Warm-up)
## Chain 1:                1.4487 seconds (Sampling)
## Chain 1:                3.30769 seconds (Total)
## Chain 1:
##
## SAMPLING FOR MODEL 'fecdb0ea2e3aab30d18291906b136a6d' NOW (CHAIN 2).
## Chain 2:
## Chain 2: Gradient evaluation took 8.1e-05 seconds
## Chain 2: 1000 transitions using 10 leapfrog steps per transition would take 0.81 seconds.
## Chain 2: Adjust your expectations accordingly!
## Chain 2:
## Chain 2:
## Chain 2: Iteration:    1 / 2000 [  0%] (Warmup)
## Chain 2: Iteration:   200 / 2000 [ 10%] (Warmup)
## Chain 2: Iteration:   400 / 2000 [ 20%] (Warmup)
## Chain 2: Iteration:   600 / 2000 [ 30%] (Warmup)
## Chain 2: Iteration:   800 / 2000 [ 40%] (Warmup)
## Chain 2: Iteration:  1000 / 2000 [ 50%] (Warmup)
## Chain 2: Iteration:  1001 / 2000 [ 50%] (Sampling)
## Chain 2: Iteration:  1200 / 2000 [ 60%] (Sampling)
## Chain 2: Iteration:  1400 / 2000 [ 70%] (Sampling)
## Chain 2: Iteration:  1600 / 2000 [ 80%] (Sampling)

```

```

## Chain 2: Iteration: 1800 / 2000 [ 90%] (Sampling)
## Chain 2: Iteration: 2000 / 2000 [100%] (Sampling)
## Chain 2:
## Chain 2: Elapsed Time: 1.71308 seconds (Warm-up)
## Chain 2: 2.16157 seconds (Sampling)
## Chain 2: 3.87465 seconds (Total)
## Chain 2:
##
## SAMPLING FOR MODEL 'fecdb0ea2e3aab30d18291906b136a6d' NOW (CHAIN 3).
## Chain 3:
## Chain 3: Gradient evaluation took 9.5e-05 seconds
## Chain 3: 1000 transitions using 10 leapfrog steps per transition would take 0.95 seconds.
## Chain 3: Adjust your expectations accordingly!
## Chain 3:
## Chain 3:
## Chain 3: Iteration: 1 / 2000 [ 0%] (Warmup)
## Chain 3: Iteration: 200 / 2000 [ 10%] (Warmup)
## Chain 3: Iteration: 400 / 2000 [ 20%] (Warmup)
## Chain 3: Iteration: 600 / 2000 [ 30%] (Warmup)
## Chain 3: Iteration: 800 / 2000 [ 40%] (Warmup)
## Chain 3: Iteration: 1000 / 2000 [ 50%] (Warmup)
## Chain 3: Iteration: 1001 / 2000 [ 50%] (Sampling)
## Chain 3: Iteration: 1200 / 2000 [ 60%] (Sampling)
## Chain 3: Iteration: 1400 / 2000 [ 70%] (Sampling)
## Chain 3: Iteration: 1600 / 2000 [ 80%] (Sampling)
## Chain 3: Iteration: 1800 / 2000 [ 90%] (Sampling)
## Chain 3: Iteration: 2000 / 2000 [100%] (Sampling)
## Chain 3:
## Chain 3: Elapsed Time: 1.75462 seconds (Warm-up)
## Chain 3: 1.98064 seconds (Sampling)
## Chain 3: 3.73527 seconds (Total)
## Chain 3:
##
## SAMPLING FOR MODEL 'fecdb0ea2e3aab30d18291906b136a6d' NOW (CHAIN 4).
## Chain 4:
## Chain 4: Gradient evaluation took 0.00013 seconds
## Chain 4: 1000 transitions using 10 leapfrog steps per transition would take 1.3 seconds.
## Chain 4: Adjust your expectations accordingly!
## Chain 4:
## Chain 4:
## Chain 4: Iteration: 1 / 2000 [ 0%] (Warmup)
## Chain 4: Iteration: 200 / 2000 [ 10%] (Warmup)
## Chain 4: Iteration: 400 / 2000 [ 20%] (Warmup)
## Chain 4: Iteration: 600 / 2000 [ 30%] (Warmup)
## Chain 4: Iteration: 800 / 2000 [ 40%] (Warmup)
## Chain 4: Iteration: 1000 / 2000 [ 50%] (Warmup)
## Chain 4: Iteration: 1001 / 2000 [ 50%] (Sampling)
## Chain 4: Iteration: 1200 / 2000 [ 60%] (Sampling)
## Chain 4: Iteration: 1400 / 2000 [ 70%] (Sampling)
## Chain 4: Iteration: 1600 / 2000 [ 80%] (Sampling)
## Chain 4: Iteration: 1800 / 2000 [ 90%] (Sampling)
## Chain 4: Iteration: 2000 / 2000 [100%] (Sampling)
## Chain 4:
## Chain 4: Elapsed Time: 1.88099 seconds (Warm-up)

```

```
## Chain 4:          1.56484 seconds (Sampling)
## Chain 4:          3.44583 seconds (Total)
## Chain 4:
```

```
print(summary(M7))
```

```
## Family: bernoulli
## Links: mu = logit
## Formula: accuracy ~ cognate + (1 | subject) + diff
## Data: spa3 (Number of observations: 1187)
## Samples: 4 chains, each with iter = 2000; warmup = 1000; thin = 1;
##          total post-warmup samples = 4000
##
## Group-Level Effects:
## ~subject (Number of levels: 27)
##          Estimate Est.Error 1-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## sd(Intercept)      2.17      0.40      1.50      3.07 1.00      649      957
##
## Population-Level Effects:
##          Estimate Est.Error 1-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## Intercept          0.91      0.46     -0.00      1.77 1.00      609     1115
## cognate             1.17      0.18      0.82      1.52 1.00     3081     2708
## diff              -0.84      0.07     -0.98     -0.71 1.00     3164     3008
##
## Samples were drawn 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).
```

```
M8 <- brm(accuracy~cognate+(1|subject)+diff,data=eng3,family=bernoulli,prior = c(set_prior("normal(0,8)
```

```
## Compiling Stan program...
```

```
## recompiling to avoid crashing R session
```

```
## Trying to compile a simple C file
```

```
## Running /Library/Frameworks/R.framework/Resources/bin/R CMD SHLIB foo.c
```

```
## clang -I"/Library/Frameworks/R.framework/Resources/include" -DNDEBUG -I"/Library/Frameworks/R.framework
```

```
## In file included from <built-in>:1:
```

```
## In file included from /Library/Frameworks/R.framework/Versions/4.0/Resources/library/StanHeaders/inc
```

```
## In file included from /Library/Frameworks/R.framework/Versions/4.0/Resources/library/RcppEigen/inclu
```

```
## In file included from /Library/Frameworks/R.framework/Versions/4.0/Resources/library/RcppEigen/inclu
```

```
## /Library/Frameworks/R.framework/Versions/4.0/Resources/library/RcppEigen/include/Eigen/src/Core/util
```

```
## namespace Eigen {
```

```
## ^
```

```
## /Library/Frameworks/R.framework/Versions/4.0/Resources/library/RcppEigen/include/Eigen/src/Core/util
```

```
## namespace Eigen {
```

```
## ^
```

```
## ;
```

```
## In file included from <built-in>:1:
```

```
## In file included from /Library/Frameworks/R.framework/Versions/4.0/Resources/library/StanHeaders/inc
```

```
## In file included from /Library/Frameworks/R.framework/Versions/4.0/Resources/library/RcppEigen/inclu
```

```
## /Library/Frameworks/R.framework/Versions/4.0/Resources/library/RcppEigen/include/Eigen/Core:96:10: f
```

```
## #include <complex>
```

```
## ^~~~~~
```

```
## 3 errors generated.
```

```
## make: *** [foo.o] Error 1
```



```

## Start sampling

##
## SAMPLING FOR MODEL 'fecdb0ea2e3aab30d18291906b136a6d' NOW (CHAIN 1).
## Chain 1:
## Chain 1: Gradient evaluation took 0.000163 seconds
## Chain 1: 1000 transitions using 10 leapfrog steps per transition would take 1.63 seconds.
## Chain 1: Adjust your expectations accordingly!
## Chain 1:
## Chain 1:
## Chain 1: Iteration:    1 / 2000 [  0%] (Warmup)
## Chain 1: Iteration:   200 / 2000 [ 10%] (Warmup)
## Chain 1: Iteration:   400 / 2000 [ 20%] (Warmup)
## Chain 1: Iteration:   600 / 2000 [ 30%] (Warmup)
## Chain 1: Iteration:   800 / 2000 [ 40%] (Warmup)
## Chain 1: Iteration:  1000 / 2000 [ 50%] (Warmup)
## Chain 1: Iteration:  1001 / 2000 [ 50%] (Sampling)
## Chain 1: Iteration:  1200 / 2000 [ 60%] (Sampling)
## Chain 1: Iteration:  1400 / 2000 [ 70%] (Sampling)
## Chain 1: Iteration:  1600 / 2000 [ 80%] (Sampling)
## Chain 1: Iteration:  1800 / 2000 [ 90%] (Sampling)
## Chain 1: Iteration:  2000 / 2000 [100%] (Sampling)
## Chain 1:
## Chain 1: Elapsed Time: 1.78691 seconds (Warm-up)
## Chain 1:                1.6479 seconds (Sampling)
## Chain 1:                3.43482 seconds (Total)
## Chain 1:
##
## SAMPLING FOR MODEL 'fecdb0ea2e3aab30d18291906b136a6d' NOW (CHAIN 2).
## Chain 2:
## Chain 2: Gradient evaluation took 8.6e-05 seconds
## Chain 2: 1000 transitions using 10 leapfrog steps per transition would take 0.86 seconds.
## Chain 2: Adjust your expectations accordingly!
## Chain 2:
## Chain 2:
## Chain 2: Iteration:    1 / 2000 [  0%] (Warmup)
## Chain 2: Iteration:   200 / 2000 [ 10%] (Warmup)
## Chain 2: Iteration:   400 / 2000 [ 20%] (Warmup)
## Chain 2: Iteration:   600 / 2000 [ 30%] (Warmup)
## Chain 2: Iteration:   800 / 2000 [ 40%] (Warmup)
## Chain 2: Iteration:  1000 / 2000 [ 50%] (Warmup)
## Chain 2: Iteration:  1001 / 2000 [ 50%] (Sampling)
## Chain 2: Iteration:  1200 / 2000 [ 60%] (Sampling)
## Chain 2: Iteration:  1400 / 2000 [ 70%] (Sampling)
## Chain 2: Iteration:  1600 / 2000 [ 80%] (Sampling)
## Chain 2: Iteration:  1800 / 2000 [ 90%] (Sampling)
## Chain 2: Iteration:  2000 / 2000 [100%] (Sampling)
## Chain 2:
## Chain 2: Elapsed Time: 1.90695 seconds (Warm-up)
## Chain 2:                1.92069 seconds (Sampling)
## Chain 2:                3.82764 seconds (Total)
## Chain 2:
##
## SAMPLING FOR MODEL 'fecdb0ea2e3aab30d18291906b136a6d' NOW (CHAIN 3).

```

```

## Chain 3:
## Chain 3: Gradient evaluation took 9.2e-05 seconds
## Chain 3: 1000 transitions using 10 leapfrog steps per transition would take 0.92 seconds.
## Chain 3: Adjust your expectations accordingly!
## Chain 3:
## Chain 3:
## Chain 3: Iteration:    1 / 2000 [  0%] (Warmup)
## Chain 3: Iteration:   200 / 2000 [ 10%] (Warmup)
## Chain 3: Iteration:   400 / 2000 [ 20%] (Warmup)
## Chain 3: Iteration:   600 / 2000 [ 30%] (Warmup)
## Chain 3: Iteration:   800 / 2000 [ 40%] (Warmup)
## Chain 3: Iteration:  1000 / 2000 [ 50%] (Warmup)
## Chain 3: Iteration: 1001 / 2000 [ 50%] (Sampling)
## Chain 3: Iteration: 1200 / 2000 [ 60%] (Sampling)
## Chain 3: Iteration: 1400 / 2000 [ 70%] (Sampling)
## Chain 3: Iteration: 1600 / 2000 [ 80%] (Sampling)
## Chain 3: Iteration: 1800 / 2000 [ 90%] (Sampling)
## Chain 3: Iteration: 2000 / 2000 [100%] (Sampling)
## Chain 3:
## Chain 3: Elapsed Time: 1.81606 seconds (Warm-up)
## Chain 3:                1.62534 seconds (Sampling)
## Chain 3:                3.4414 seconds (Total)
## Chain 3:
##
## SAMPLING FOR MODEL 'fecdb0ea2e3aab30d18291906b136a6d' NOW (CHAIN 4).
## Chain 4:
## Chain 4: Gradient evaluation took 9.5e-05 seconds
## Chain 4: 1000 transitions using 10 leapfrog steps per transition would take 0.95 seconds.
## Chain 4: Adjust your expectations accordingly!
## Chain 4:
## Chain 4:
## Chain 4: Iteration:    1 / 2000 [  0%] (Warmup)
## Chain 4: Iteration:   200 / 2000 [ 10%] (Warmup)
## Chain 4: Iteration:   400 / 2000 [ 20%] (Warmup)
## Chain 4: Iteration:   600 / 2000 [ 30%] (Warmup)
## Chain 4: Iteration:   800 / 2000 [ 40%] (Warmup)
## Chain 4: Iteration:  1000 / 2000 [ 50%] (Warmup)
## Chain 4: Iteration: 1001 / 2000 [ 50%] (Sampling)
## Chain 4: Iteration: 1200 / 2000 [ 60%] (Sampling)
## Chain 4: Iteration: 1400 / 2000 [ 70%] (Sampling)
## Chain 4: Iteration: 1600 / 2000 [ 80%] (Sampling)
## Chain 4: Iteration: 1800 / 2000 [ 90%] (Sampling)
## Chain 4: Iteration: 2000 / 2000 [100%] (Sampling)
## Chain 4:
## Chain 4: Elapsed Time: 1.7289 seconds (Warm-up)
## Chain 4:                1.46689 seconds (Sampling)
## Chain 4:                3.19579 seconds (Total)
## Chain 4:

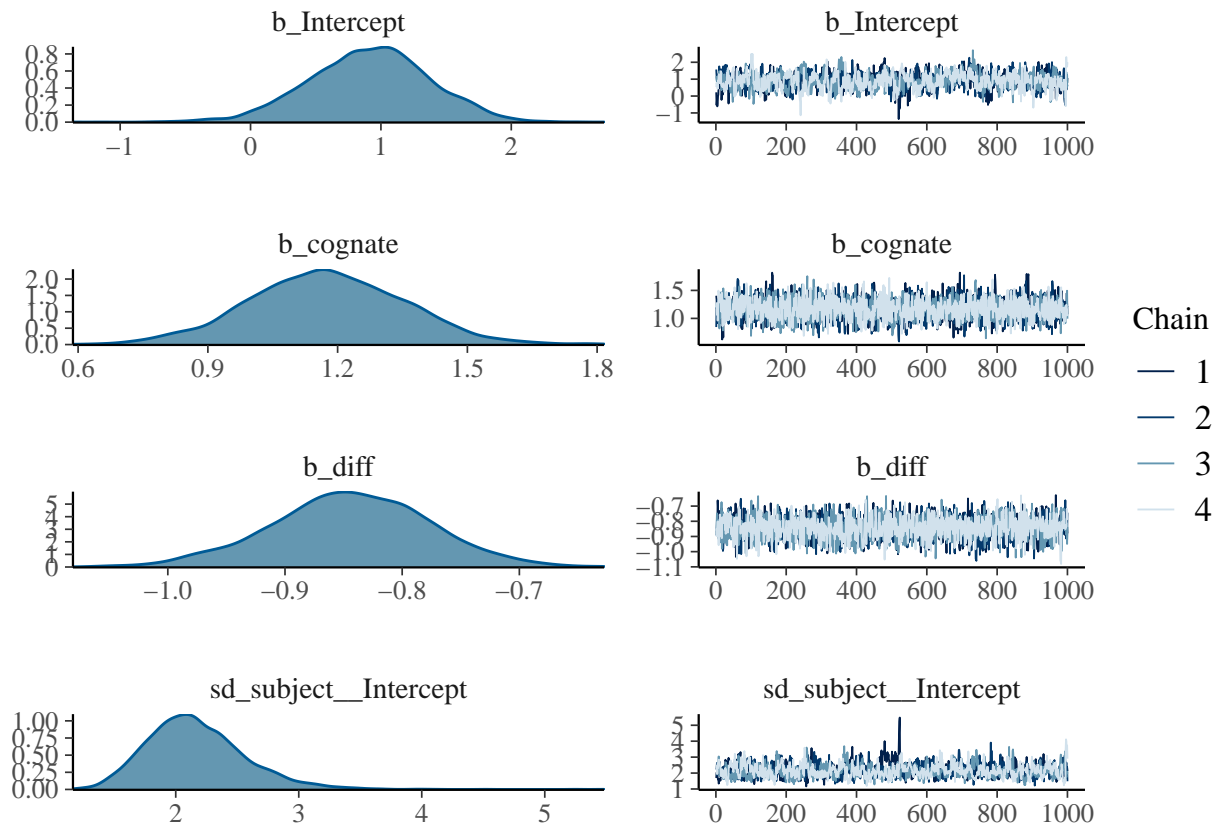
## Warning: Bulk Effective Samples Size (ESS) is too low, indicating posterior means and medians may be
## Running the chains for more iterations may help. See
## http://mc-stan.org/misc/warnings.html#bulk-ess

```

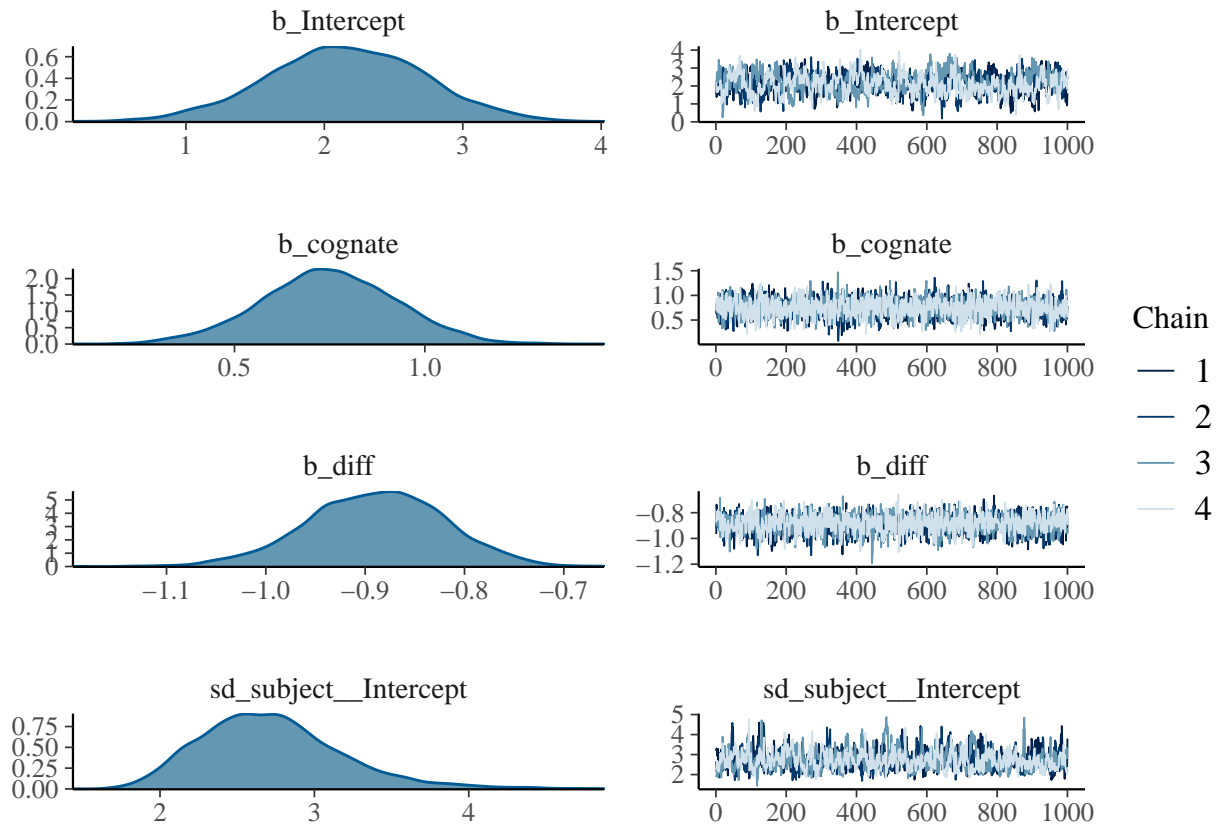
```
print(summary(M8))
```

```
## Family: bernoulli
## Links: mu = logit
## Formula: accuracy ~ cognate + (1 | subject) + diff
## Data: eng3 (Number of observations: 1188)
## Samples: 4 chains, each with iter = 2000; warmup = 1000; thin = 1;
##           total post-warmup samples = 4000
##
## Group-Level Effects:
## ~subject (Number of levels: 27)
##           Estimate Est.Error 1-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## sd(Intercept)    2.73     0.46    1.98    3.80 1.00     608     844
##
## Population-Level Effects:
##           Estimate Est.Error 1-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## Intercept        2.16     0.56    1.03    3.25 1.01     313     706
## cognate           0.75     0.18    0.39    1.10 1.00    2740    2633
## diff             -0.89     0.07   -1.03   -0.76 1.00    2157    2496
##
## Samples were drawn 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).
```

```
plot(M7)
```



```
plot(M8)
```



Adding L2AoA predictor.

L2:nonnative language AoA: age of acquisition

```
M9<- brm(accuracy~cognate+(1|subject)+diff+L2AoA,data=spa3,family=bernoulli,prior = c(set_prior("normal",
## Warning: Rows containing NAs were excluded from the model.
## Compiling Stan program...
## recompiling to avoid crashing R session
## Trying to compile a simple C file
## Running /Library/Frameworks/R.framework/Resources/bin/R CMD SHLIB foo.c
## clang -I"/Library/Frameworks/R.framework/Resources/include" -DNDEBUG -I"/Library/Frameworks/R.framework
## In file included from <built-in>:1:
## In file included from /Library/Frameworks/R.framework/Versions/4.0/Resources/library/StanHeaders/include
## In file included from /Library/Frameworks/R.framework/Versions/4.0/Resources/library/RcppEigen/include
## In file included from /Library/Frameworks/R.framework/Versions/4.0/Resources/library/RcppEigen/include
## /Library/Frameworks/R.framework/Versions/4.0/Resources/library/RcppEigen/include/Eigen/src/Core/util
## namespace Eigen {
## ^
## /Library/Frameworks/R.framework/Versions/4.0/Resources/library/RcppEigen/include/Eigen/src/Core/util
## namespace Eigen {
## ^
## ;
## In file included from <built-in>:1:
```

```

## In file included from /Library/Frameworks/R.framework/Versions/4.0/Resources/library/StanHeaders/inc
## In file included from /Library/Frameworks/R.framework/Versions/4.0/Resources/library/RcppEigen/inclu
## /Library/Frameworks/R.framework/Versions/4.0/Resources/library/RcppEigen/include/Eigen/Core:96:10: f
## #include <complex>
##      ~~~~~
## 3 errors generated.
## make: *** [foo.o] Error 1

## Start sampling

##
## SAMPLING FOR MODEL 'fecdb0ea2e3aab30d18291906b136a6d' NOW (CHAIN 1).
## Chain 1:
## Chain 1: Gradient evaluation took 0.000174 seconds
## Chain 1: 1000 transitions using 10 leapfrog steps per transition would take 1.74 seconds.
## Chain 1: Adjust your expectations accordingly!
## Chain 1:
## Chain 1:
## Chain 1: Iteration:    1 / 2000 [  0%] (Warmup)
## Chain 1: Iteration:   200 / 2000 [ 10%] (Warmup)
## Chain 1: Iteration:   400 / 2000 [ 20%] (Warmup)
## Chain 1: Iteration:   600 / 2000 [ 30%] (Warmup)
## Chain 1: Iteration:   800 / 2000 [ 40%] (Warmup)
## Chain 1: Iteration:  1000 / 2000 [ 50%] (Warmup)
## Chain 1: Iteration:  1001 / 2000 [ 50%] (Sampling)
## Chain 1: Iteration:  1200 / 2000 [ 60%] (Sampling)
## Chain 1: Iteration:  1400 / 2000 [ 70%] (Sampling)
## Chain 1: Iteration:  1600 / 2000 [ 80%] (Sampling)
## Chain 1: Iteration:  1800 / 2000 [ 90%] (Sampling)
## Chain 1: Iteration:  2000 / 2000 [100%] (Sampling)
## Chain 1:
## Chain 1: Elapsed Time: 3.77529 seconds (Warm-up)
## Chain 1:                2.5125 seconds (Sampling)
## Chain 1:                6.28779 seconds (Total)
## Chain 1:
##
## SAMPLING FOR MODEL 'fecdb0ea2e3aab30d18291906b136a6d' NOW (CHAIN 2).
## Chain 2:
## Chain 2: Gradient evaluation took 9.4e-05 seconds
## Chain 2: 1000 transitions using 10 leapfrog steps per transition would take 0.94 seconds.
## Chain 2: Adjust your expectations accordingly!
## Chain 2:
## Chain 2:
## Chain 2: Iteration:    1 / 2000 [  0%] (Warmup)
## Chain 2: Iteration:   200 / 2000 [ 10%] (Warmup)
## Chain 2: Iteration:   400 / 2000 [ 20%] (Warmup)
## Chain 2: Iteration:   600 / 2000 [ 30%] (Warmup)
## Chain 2: Iteration:   800 / 2000 [ 40%] (Warmup)
## Chain 2: Iteration:  1000 / 2000 [ 50%] (Warmup)
## Chain 2: Iteration:  1001 / 2000 [ 50%] (Sampling)
## Chain 2: Iteration:  1200 / 2000 [ 60%] (Sampling)
## Chain 2: Iteration:  1400 / 2000 [ 70%] (Sampling)
## Chain 2: Iteration:  1600 / 2000 [ 80%] (Sampling)
## Chain 2: Iteration:  1800 / 2000 [ 90%] (Sampling)
## Chain 2: Iteration:  2000 / 2000 [100%] (Sampling)

```

```

## Chain 2:
## Chain 2: Elapsed Time: 3.88911 seconds (Warm-up)
## Chain 2: 2.14055 seconds (Sampling)
## Chain 2: 6.02966 seconds (Total)
## Chain 2:
##
## SAMPLING FOR MODEL 'fecdb0ea2e3aab30d18291906b136a6d' NOW (CHAIN 3).
## Chain 3:
## Chain 3: Gradient evaluation took 0.000357 seconds
## Chain 3: 1000 transitions using 10 leapfrog steps per transition would take 3.57 seconds.
## Chain 3: Adjust your expectations accordingly!
## Chain 3:
## Chain 3:
## Chain 3: Iteration: 1 / 2000 [ 0%] (Warmup)
## Chain 3: Iteration: 200 / 2000 [ 10%] (Warmup)
## Chain 3: Iteration: 400 / 2000 [ 20%] (Warmup)
## Chain 3: Iteration: 600 / 2000 [ 30%] (Warmup)
## Chain 3: Iteration: 800 / 2000 [ 40%] (Warmup)
## Chain 3: Iteration: 1000 / 2000 [ 50%] (Warmup)
## Chain 3: Iteration: 1001 / 2000 [ 50%] (Sampling)
## Chain 3: Iteration: 1200 / 2000 [ 60%] (Sampling)
## Chain 3: Iteration: 1400 / 2000 [ 70%] (Sampling)
## Chain 3: Iteration: 1600 / 2000 [ 80%] (Sampling)
## Chain 3: Iteration: 1800 / 2000 [ 90%] (Sampling)
## Chain 3: Iteration: 2000 / 2000 [100%] (Sampling)
## Chain 3:
## Chain 3: Elapsed Time: 3.6098 seconds (Warm-up)
## Chain 3: 2.07225 seconds (Sampling)
## Chain 3: 5.68205 seconds (Total)
## Chain 3:
##
## SAMPLING FOR MODEL 'fecdb0ea2e3aab30d18291906b136a6d' NOW (CHAIN 4).
## Chain 4:
## Chain 4: Gradient evaluation took 0.0001 seconds
## Chain 4: 1000 transitions using 10 leapfrog steps per transition would take 1 seconds.
## Chain 4: Adjust your expectations accordingly!
## Chain 4:
## Chain 4:
## Chain 4: Iteration: 1 / 2000 [ 0%] (Warmup)
## Chain 4: Iteration: 200 / 2000 [ 10%] (Warmup)
## Chain 4: Iteration: 400 / 2000 [ 20%] (Warmup)
## Chain 4: Iteration: 600 / 2000 [ 30%] (Warmup)
## Chain 4: Iteration: 800 / 2000 [ 40%] (Warmup)
## Chain 4: Iteration: 1000 / 2000 [ 50%] (Warmup)
## Chain 4: Iteration: 1001 / 2000 [ 50%] (Sampling)
## Chain 4: Iteration: 1200 / 2000 [ 60%] (Sampling)
## Chain 4: Iteration: 1400 / 2000 [ 70%] (Sampling)
## Chain 4: Iteration: 1600 / 2000 [ 80%] (Sampling)
## Chain 4: Iteration: 1800 / 2000 [ 90%] (Sampling)
## Chain 4: Iteration: 2000 / 2000 [100%] (Sampling)
## Chain 4:
## Chain 4: Elapsed Time: 3.46131 seconds (Warm-up)
## Chain 4: 2.65726 seconds (Sampling)
## Chain 4: 6.11857 seconds (Total)

```

```
## Chain 4:
```

```
print(summary(M9))
```

```
## Family: bernoulli
## Links: mu = logit
## Formula: accuracy ~ cognate + (1 | subject) + diff + L2AoA
## Data: spa3 (Number of observations: 1187)
## Samples: 4 chains, each with iter = 2000; warmup = 1000; thin = 1;
##           total post-warmup samples = 4000
##
## Group-Level Effects:
## ~subject (Number of levels: 27)
##           Estimate Est.Error 1-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## sd(Intercept)      2.18      0.41      1.53      3.11 1.00      798      1585
##
## Population-Level Effects:
##           Estimate Est.Error 1-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## Intercept          0.13      0.79     -1.44      1.68 1.00      647      1382
## cognate             1.17      0.18      0.83      1.53 1.00     3239      2731
## diff              -0.85      0.07     -0.98     -0.72 1.00     3189      2508
## L2AoA               0.07      0.06     -0.03      0.18 1.00      895      1405
##
## Samples were drawn 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).
```

```
M10 <- brm(accuracy~cognate+(1|subject)+diff+L2AoA,data=eng3,family=bernoulli,prior = c(set_prior("norm
```

```
## Compiling Stan program...
```

```
## recompiling to avoid crashing R session
```

```
## Trying to compile a simple C file
```

```
## Running /Library/Frameworks/R.framework/Resources/bin/R CMD SHLIB foo.c
## clang -I"/Library/Frameworks/R.framework/Resources/include" -DNDEBUG -I"/Library/Frameworks/R.framework/Resources/include"
## In file included from <built-in>:1:
## In file included from /Library/Frameworks/R.framework/Versions/4.0/Resources/library/StanHeaders/include:
## In file included from /Library/Frameworks/R.framework/Versions/4.0/Resources/library/RcppEigen/include:
## In file included from /Library/Frameworks/R.framework/Versions/4.0/Resources/library/RcppEigen/include:
## /Library/Frameworks/R.framework/Versions/4.0/Resources/library/RcppEigen/include/Eigen/src/Core/util:
## namespace Eigen {
## ^
## /Library/Frameworks/R.framework/Versions/4.0/Resources/library/RcppEigen/include/Eigen/src/Core/util:
## namespace Eigen {
## ^
## ;
## In file included from <built-in>:1:
## In file included from /Library/Frameworks/R.framework/Versions/4.0/Resources/library/StanHeaders/include:
## In file included from /Library/Frameworks/R.framework/Versions/4.0/Resources/library/RcppEigen/include:
## /Library/Frameworks/R.framework/Versions/4.0/Resources/library/RcppEigen/include/Eigen/Core:96:10: f
## #include <complex>
## ^~~~~~
## 3 errors generated.
## make: *** [foo.o] Error 1
```

```

## Start sampling

##
## SAMPLING FOR MODEL 'fecdb0ea2e3aab30d18291906b136a6d' NOW (CHAIN 1).
## Chain 1:
## Chain 1: Gradient evaluation took 0.000142 seconds
## Chain 1: 1000 transitions using 10 leapfrog steps per transition would take 1.42 seconds.
## Chain 1: Adjust your expectations accordingly!
## Chain 1:
## Chain 1:
## Chain 1: Iteration:    1 / 2000 [  0%] (Warmup)
## Chain 1: Iteration:   200 / 2000 [ 10%] (Warmup)
## Chain 1: Iteration:   400 / 2000 [ 20%] (Warmup)
## Chain 1: Iteration:   600 / 2000 [ 30%] (Warmup)
## Chain 1: Iteration:   800 / 2000 [ 40%] (Warmup)
## Chain 1: Iteration:  1000 / 2000 [ 50%] (Warmup)
## Chain 1: Iteration:  1001 / 2000 [ 50%] (Sampling)
## Chain 1: Iteration:  1200 / 2000 [ 60%] (Sampling)
## Chain 1: Iteration:  1400 / 2000 [ 70%] (Sampling)
## Chain 1: Iteration:  1600 / 2000 [ 80%] (Sampling)
## Chain 1: Iteration:  1800 / 2000 [ 90%] (Sampling)
## Chain 1: Iteration:  2000 / 2000 [100%] (Sampling)
## Chain 1:
## Chain 1: Elapsed Time: 2.98006 seconds (Warm-up)
## Chain 1:                2.39072 seconds (Sampling)
## Chain 1:                5.37078 seconds (Total)
## Chain 1:
##
## SAMPLING FOR MODEL 'fecdb0ea2e3aab30d18291906b136a6d' NOW (CHAIN 2).
## Chain 2:
## Chain 2: Gradient evaluation took 9.5e-05 seconds
## Chain 2: 1000 transitions using 10 leapfrog steps per transition would take 0.95 seconds.
## Chain 2: Adjust your expectations accordingly!
## Chain 2:
## Chain 2:
## Chain 2: Iteration:    1 / 2000 [  0%] (Warmup)
## Chain 2: Iteration:   200 / 2000 [ 10%] (Warmup)
## Chain 2: Iteration:   400 / 2000 [ 20%] (Warmup)
## Chain 2: Iteration:   600 / 2000 [ 30%] (Warmup)
## Chain 2: Iteration:   800 / 2000 [ 40%] (Warmup)
## Chain 2: Iteration:  1000 / 2000 [ 50%] (Warmup)
## Chain 2: Iteration:  1001 / 2000 [ 50%] (Sampling)
## Chain 2: Iteration:  1200 / 2000 [ 60%] (Sampling)
## Chain 2: Iteration:  1400 / 2000 [ 70%] (Sampling)
## Chain 2: Iteration:  1600 / 2000 [ 80%] (Sampling)
## Chain 2: Iteration:  1800 / 2000 [ 90%] (Sampling)
## Chain 2: Iteration:  2000 / 2000 [100%] (Sampling)
## Chain 2:
## Chain 2: Elapsed Time: 2.93683 seconds (Warm-up)
## Chain 2:                2.25904 seconds (Sampling)
## Chain 2:                5.19587 seconds (Total)
## Chain 2:
##
## SAMPLING FOR MODEL 'fecdb0ea2e3aab30d18291906b136a6d' NOW (CHAIN 3).

```



```

## Chain 3:
## Chain 3: Gradient evaluation took 0.000197 seconds
## Chain 3: 1000 transitions using 10 leapfrog steps per transition would take 1.97 seconds.
## Chain 3: Adjust your expectations accordingly!
## Chain 3:
## Chain 3:
## Chain 3: Iteration:    1 / 2000 [  0%] (Warmup)
## Chain 3: Iteration:   200 / 2000 [ 10%] (Warmup)
## Chain 3: Iteration:   400 / 2000 [ 20%] (Warmup)
## Chain 3: Iteration:   600 / 2000 [ 30%] (Warmup)
## Chain 3: Iteration:   800 / 2000 [ 40%] (Warmup)
## Chain 3: Iteration:  1000 / 2000 [ 50%] (Warmup)
## Chain 3: Iteration:  1001 / 2000 [ 50%] (Sampling)
## Chain 3: Iteration:  1200 / 2000 [ 60%] (Sampling)
## Chain 3: Iteration:  1400 / 2000 [ 70%] (Sampling)
## Chain 3: Iteration:  1600 / 2000 [ 80%] (Sampling)
## Chain 3: Iteration:  1800 / 2000 [ 90%] (Sampling)
## Chain 3: Iteration:  2000 / 2000 [100%] (Sampling)
## Chain 3:
## Chain 3: Elapsed Time: 2.93914 seconds (Warm-up)
## Chain 3:                2.45448 seconds (Sampling)
## Chain 3:                5.39361 seconds (Total)
## Chain 3:
##

```

```

## SAMPLING FOR MODEL 'fecdb0ea2e3aab30d18291906b136a6d' NOW (CHAIN 4).

```

```

## Chain 4:
## Chain 4: Gradient evaluation took 9.1e-05 seconds
## Chain 4: 1000 transitions using 10 leapfrog steps per transition would take 0.91 seconds.
## Chain 4: Adjust your expectations accordingly!
## Chain 4:
## Chain 4:
## Chain 4: Iteration:    1 / 2000 [  0%] (Warmup)
## Chain 4: Iteration:   200 / 2000 [ 10%] (Warmup)
## Chain 4: Iteration:   400 / 2000 [ 20%] (Warmup)
## Chain 4: Iteration:   600 / 2000 [ 30%] (Warmup)
## Chain 4: Iteration:   800 / 2000 [ 40%] (Warmup)
## Chain 4: Iteration:  1000 / 2000 [ 50%] (Warmup)
## Chain 4: Iteration:  1001 / 2000 [ 50%] (Sampling)
## Chain 4: Iteration:  1200 / 2000 [ 60%] (Sampling)
## Chain 4: Iteration:  1400 / 2000 [ 70%] (Sampling)
## Chain 4: Iteration:  1600 / 2000 [ 80%] (Sampling)
## Chain 4: Iteration:  1800 / 2000 [ 90%] (Sampling)
## Chain 4: Iteration:  2000 / 2000 [100%] (Sampling)
## Chain 4:
## Chain 4: Elapsed Time: 3.2134 seconds (Warm-up)
## Chain 4:                2.08529 seconds (Sampling)
## Chain 4:                5.2987 seconds (Total)
## Chain 4:

```

```

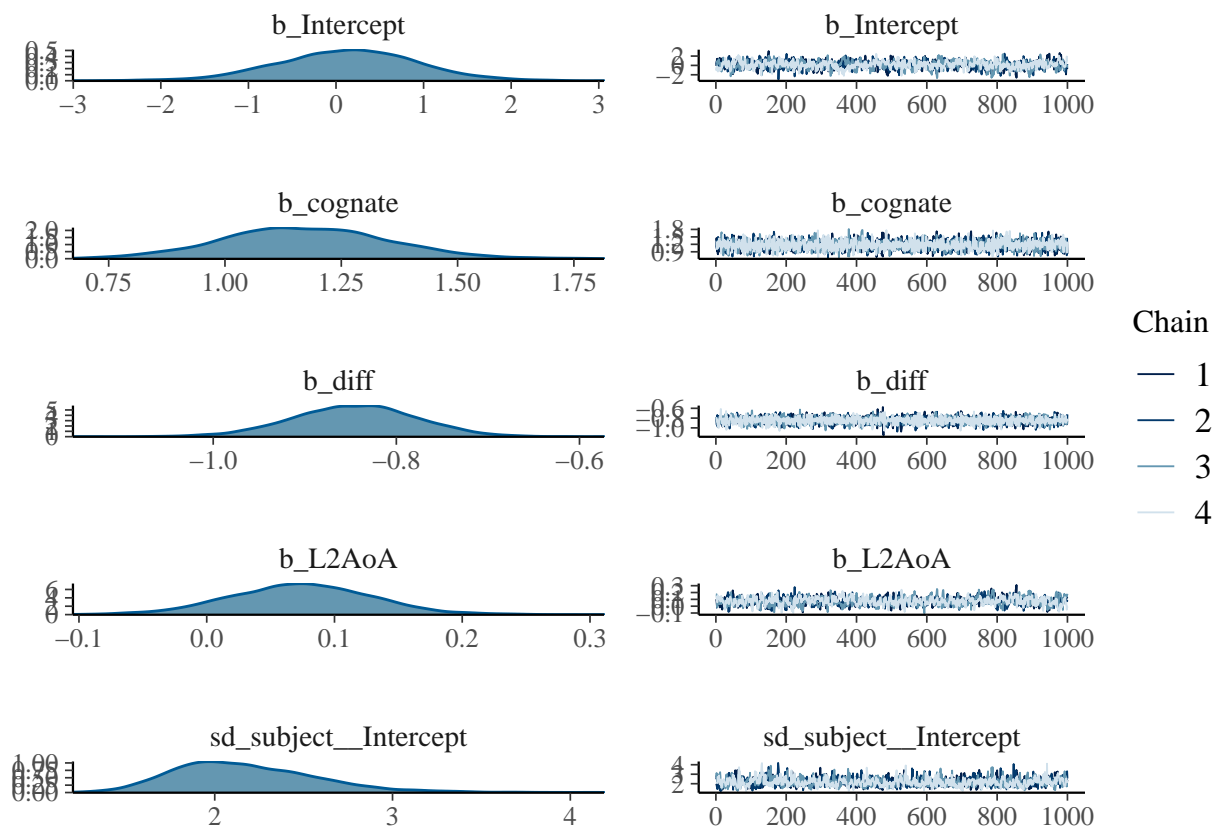
## Warning: Bulk Effective Samples Size (ESS) is too low, indicating posterior means and medians may be
## Running the chains for more iterations may help. See
## http://mc-stan.org/misc/warnings.html#bulk-ess

```

```
print(summary(M10))
```

```
## Family: bernoulli
## Links: mu = logit
## Formula: accuracy ~ cognate + (1 | subject) + diff + L2AoA
## Data: eng3 (Number of observations: 1188)
## Samples: 4 chains, each with iter = 2000; warmup = 1000; thin = 1;
##           total post-warmup samples = 4000
##
## Group-Level Effects:
## ~subject (Number of levels: 27)
##           Estimate Est.Error 1-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## sd(Intercept)      2.54      0.46     1.82     3.56 1.00      684     1315
##
## Population-Level Effects:
##           Estimate Est.Error 1-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## Intercept         3.87      0.97     1.99     5.88 1.02      380      941
## cognate            0.75      0.18     0.40     1.10 1.00     2796     2628
## diff              -0.89      0.07    -1.04    -0.75 1.00     3006     2397
## L2AoA             -0.16      0.07    -0.31    -0.03 1.01      661     1178
##
## Samples were drawn 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).
```

```
plot(M9)
```



```
plot(M10)
```

