Intro to lmer

Document Adapted from Brown, V. A. (2021).

Joscelin Rocha-Hidalgo

2022-02-07

Contents

This code is an adaptation of the code that accompanies "An introduction to mixed effects modeling" manuscript: - Brown, V. A. (2021). An introduction to linear mixed-effects modeling in R. *Advances in Methods and Practices in Psychological Science, 4*(1), 2515245920960351.

Preliminaries

Install packages if they aren't already installed

```
if (!("lme4" %in% installed.packages())) install.packages("lme4")
if (!("lmerTest" %in% installed.packages())) install.packages("lmerTest")
if (!("tidyverse" %in% installed.packages())) install.packages("tidyverse")
if (!("afex" %in% installed.packages())) install.packages("afex")
if (!("performance" %in% installed.packages())) install.packages("performance")
if (!("sessioninfo" %in% installed.packages())) install.packages("sessioninfo")
```

Load packages

```
library(lme4)
```

```
## Loading required package: Matrix
```

```
library(lmerTest)
```

```
##
## Attaching package: 'lmerTest'

## The following object is masked from 'package:lme4':
##
##     lmer

## The following object is masked from 'package:stats':
##
##     step
```

```
library(tidyverse)
```

```
## -- Attaching packages ---------------------------------------- tidyverse 1.3.1 --

## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.6     v dplyr   1.0.7
## v tidyr   1.2.0     v stringr 1.4.0
## v readr   2.1.2     v forcats 0.5.1
## Warning: package ’readr’ was built under R version 4.1.2

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x tidyr::expand() masks Matrix::expand()
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## x tidyr::pack()   masks Matrix::pack()
## x tidyr::unpack() masks Matrix::unpack()
```

```
library(afex)
```

```
## ************
## Welcome to afex. For support visit: http://afex.singmann.science/

## - Functions for ANOVAs: aov_car(), aov_ez(), and aov_4()
## - Methods for calculating p-values with mixed(): ’S’, ’KR’, ’LRT’, and ’PB’
## - ’afex_aov’ and ’mixed’ objects can be passed to emmeans() for follow-up tests
## - NEWS: emmeans() for ANOVA models now uses model = ’multivariate’ as default.
## - Get and set global package options with: afex_options()
## - Set orthogonal sum-to-zero contrasts globally: set_sum_contrasts()
## - For example analyses see: browseVignettes("afex")
## ************

##
## Attaching package: ’afex’

## The following object is masked from ’package:lme4’:
##
##     lmer
```

```
library(performance)
library(sessioninfo)

session_info(pkgs="!attached", to_file = TRUE)
```

Load data, and name that object “rt_data”

```
rt_data <- read_csv("Data/rt_dummy_data.csv")
```

```
## Rows: 21679 Columns: 4
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (2): modality, stim
## dbl (2): PID, RT
##
## 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.
```

View the first six rows of the data frame

```
head(rt_data)
```

```
## # A tibble: 6 x 4
##     PID    RT modality   stim
##   <dbl> <dbl> <chr>      <chr>
## 1   301  1024 Audio-only gown
## 2   301   838 Audio-only might
## 3   301  1060 Audio-only fern
## 4   301   882 Audio-only vane
## 5   301   971 Audio-only pup
## 6   301  1064 Audio-only rise
```

Testing for an effect of modality on response time

Dummy code modality so that audio-only is the reference level

```
rt_data$modality <- ifelse(rt_data$modality == "Audio-only", 0, 1)
```

Build a full model

```
rt_full.mod <- lmer(RT ~ 1 + modality +
                      (1 + modality|PID) + (1 + modality|stim),
                    data = rt_data)
```

```
## Warning in checkConv(attr(opt, "derivs"), opt$par, ctrl = control$checkConv, :
## Model failed to converge with max|grad| = 0.00882975 (tol = 0.002, component 1)
```

This model failed to converge. The first thing we'll do is try the all_fit() function from the afex package to look for an optimizer that works.

Optimizing

The bobyqa optimizer should work.

```
rt_full.mod <- lmer(RT ~ 1 + modality +
                      (1 + modality|PID) + (1 + modality|stim),
                    data = rt_data,
                    control = lmerControl(optimizer = "bobyqa"))
```

All_fit()

Run the all_fit() function from the afex() package for demonstration purposes.

```
all_fit(rt_full.mod)
```

```
## bobyqa. : [OK]
## Nelder_Mead. : [OK]
## optimx.nlminb :
```

```
## Loading required namespace: optimx

## [OK]
## optimx.L-BFGS-B :

## Warning in checkConv(attr(opt, "derivs"), opt$par, ctrl = control$checkConv, :
## Model failed to converge with max|grad| = 0.0240147 (tol = 0.002, component 1)

## [OK]
## nloptwrap.NLOPT_LN_NELDERMEAD :

## Warning in checkConv(attr(opt, "derivs"), opt$par, ctrl = control$checkConv, :
## Model failed to converge with max|grad| = 0.00882975 (tol = 0.002, component 1)

## [OK]
## nloptwrap.NLOPT_LN_BOBYQA :

## Warning in checkConv(attr(opt, "derivs"), opt$par, ctrl = control$checkConv, :
## Model failed to converge with max|grad| = 0.00882975 (tol = 0.002, component 1)

## [OK]
## nmkbw. : [OK]

## $bobyqa.
## Linear mixed model fit by REML ['lmerModLmerTest']
## Formula: RT ~ 1 + modality + (1 + modality | PID) + (1 + modality | stim)
##    Data: rt_data
## REML criterion at convergence: 302385.7
## Random effects:
##  Groups   Name        Std.Dev. Corr
##  stim     (Intercept)  17.43
##           modality     14.72   0.16
##  PID      (Intercept) 168.98
##           modality     87.81   -0.17
##  Residual             255.46
## Number of obs: 21679, groups:  stim, 543; PID, 53
## Fixed Effects:
## (Intercept)     modality
##     1044.14        83.18
##
## $Nelder_Mead.
## Linear mixed model fit by REML ['lmerModLmerTest']
## Formula: RT ~ 1 + modality + (1 + modality | PID) + (1 + modality | stim)
##    Data: rt_data
## REML criterion at convergence: 302385.7
## Random effects:
##  Groups   Name        Std.Dev. Corr
##  stim     (Intercept)  17.43
##           modality     14.72   0.16
##  PID      (Intercept) 168.98
##           modality     87.81   -0.17
##  Residual             255.46
## Number of obs: 21679, groups:  stim, 543; PID, 53
## Fixed Effects:
## (Intercept)     modality
##     1044.14        83.18
##
## $optimx.nlminb
## Linear mixed model fit by REML ['lmerModLmerTest']
## Formula: RT ~ 1 + modality + (1 + modality | PID) + (1 + modality | stim)
##    Data: rt_data
## REML criterion at convergence: 302385.7
## Random effects:
##  Groups   Name        Std.Dev. Corr
##  stim     (Intercept)  17.44
##           modality     14.73   0.16
##  PID      (Intercept) 168.97
##           modality     87.81   -0.16
##  Residual             255.46
## Number of obs: 21679, groups:  stim, 543; PID, 53
## Fixed Effects:
## (Intercept)     modality
##     1044.14        83.18
##
## $`optimx.L-BFGS-B`
## Linear mixed model fit by REML ['lmerModLmerTest']
## Formula: RT ~ 1 + modality + (1 + modality | PID) + (1 + modality | stim)
##    Data: rt_data
## REML criterion at convergence: 302385.7
## Random effects:
##  Groups   Name        Std.Dev. Corr
##  stim     (Intercept)  17.36
##           modality     14.62   0.17
##  PID      (Intercept) 168.90
##           modality     87.79   -0.16
##  Residual             255.46
## Number of obs: 21679, groups:  stim, 543; PID, 53
## Fixed Effects:
## (Intercept)     modality
##     1044.14        83.18
## optimizer (optimx) convergence code: 0 (OK) ; 0 optimizer warnings; 1 lme4 warnings
##
## $nloptwrap.NLOPT_LN_NELDERMEAD
## Linear mixed model fit by REML ['lmerModLmerTest']
## Formula: RT ~ 1 + modality + (1 + modality | PID) + (1 + modality | stim)
##    Data: rt_data
## REML criterion at convergence: 302385.7
## Random effects:
##  Groups   Name        Std.Dev. Corr
##  stim     (Intercept)  17.43
##           modality     14.73   0.16
##  PID      (Intercept) 169.08
##           modality     87.80   -0.16
##  Residual             255.46
## Number of obs: 21679, groups:  stim, 543; PID, 53
## Fixed Effects:
```

```
## (Intercept)     modality
##     1044.14        83.18
## optimizer (nloptwrap) convergence code: 0 (OK) ; 0 optimizer warnings; 1 lme4 warnings
##
## $nloptwrap.NLOPT_LN_BOBYQA
## Linear mixed model fit by REML ['lmerModLmerTest']
## Formula: RT ~ 1 + modality + (1 + modality | PID) + (1 + modality | stim)
##    Data: rt_data
## REML criterion at convergence: 302385.7
## Random effects:
##  Groups   Name        Std.Dev. Corr
##  stim     (Intercept)  17.43
##           modality     14.73   0.16
##  PID      (Intercept) 169.08
##           modality     87.80   -0.16
##  Residual             255.46
## Number of obs: 21679, groups:  stim, 543; PID, 53
## Fixed Effects:
## (Intercept)     modality
##     1044.14        83.18
## optimizer (nloptwrap) convergence code: 0 (OK) ; 0 optimizer warnings; 1 lme4 warnings
##
## $nmkbw.
## Linear mixed model fit by REML ['lmerModLmerTest']
## Formula: RT ~ 1 + modality + (1 + modality | PID) + (1 + modality | stim)
##    Data: rt_data
## REML criterion at convergence: 302385.7
## Random effects:
##  Groups   Name        Std.Dev. Corr
##  stim     (Intercept)  17.44
##           modality     14.72   0.16
##  PID      (Intercept) 168.96
##           modality     87.80   -0.17
##  Residual             255.46
## Number of obs: 21679, groups:  stim, 543; PID, 53
## Fixed Effects:
## (Intercept)     modality
##     1044.14        83.18
```

Reduced Model

Build a reduced model that doesn't contained the fixed effect of modality, but is otherwise identical to the full model (including the random effects structure and control parameter)

```
rt_reduced.mod <- lmer(RT ~ 1 +
                         (1 + modality|stim) + (1 + modality|PID),
                       data = rt_data,
                       control = lmerControl(optimizer = "bobyqa"))
```

Likelihood Ratio Test

Test for an effect of modality via a likelihood ratio test

```
anova(rt_reduced.mod, rt_full.mod)
```

```
## refitting model(s) with ML (instead of REML)
```

```
## Data: rt_data
## Models:
## rt_reduced.mod: RT ~ 1 + (1 + modality | stim) + (1 + modality | PID)
## rt_full.mod: RT ~ 1 + modality + (1 + modality | PID) + (1 + modality | stim)
##                 npar    AIC    BIC  logLik deviance  Chisq Df Pr(>Chisq)
## rt_reduced.mod    8 302449 302513 -151217   302433
## rt_full.mod       9 302419 302491 -151200   302401 32.385  1  1.264e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Using the Mixed() function

Use the mixed() function from the afex package for demonstration purposes (this appears in the Likelihood Ratio Tests portion of the paper)

```
mixed(RT ~ 1 + modality +
         (1 + modality|PID) + (1 + modality|stim),
       data = rt_data,
       control = lmerControl(optimizer = "bobyqa"),
       method = 'LRT')
```

```
## Contrasts set to contr.sum for the following variables: stim
```

```
## Numerical variables NOT centered on 0: modality
## If in interactions, interpretation of lower order (e.g., main) effects difficult.
```

```
## REML argument to lmer() set to FALSE for method = 'PB' or 'LRT'
```

```
## Fitting 2 (g)lmer() models:
## [..]
```

```
## Mixed Model Anova Table (Type 3 tests, LRT-method)
##
## Model: RT ~ 1 + modality + (1 + modality | PID) + (1 + modality | stim)
## Data: rt_data
## Df full model: 9
##     Effect df     Chisq p.value
## 1 modality  1 32.39 ***   <.001
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '+' 0.1 ' ' 1
```

Summary

View summary output

```
summary(rt_full.mod)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: RT ~ 1 + modality + (1 + modality | PID) + (1 + modality | stim)
##    Data: rt_data
## Control: lmerControl(optimizer = "bobyqa")
##
## REML criterion at convergence: 302385.7
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -3.3646 -0.6964 -0.0140  0.5886  5.0003
##
## Random effects:
##  Groups   Name        Variance Std.Dev. Corr
##  stim     (Intercept)   303.9   17.43
##           modality      216.7   14.72   0.16
##  PID      (Intercept) 28552.9  168.98
##           modality     7709.8   87.81   -0.17
##  Residual             65258.8  255.46
## Number of obs: 21679, groups:  stim, 543; PID, 53
##
## Fixed effects:
##             Estimate Std. Error      df t value Pr(>|t|)
## (Intercept)  1044.14      23.36   52.14  44.704  < 2e-16 ***
## modality       83.18      12.58   52.10   6.615 2.02e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##          (Intr)
## modality -0.178
```

The parameter estimate for the effect of condition is 83.18, which means that in this dummy data, participants are on average 83 ms slower in the audiovisual relative to the audio-only condition.

Coef()

Run the coef() function to examine individual participant and item intercept and slope estimates

```
coef(rt_full.mod)
```

```
## $stim
##        (Intercept) modality
## babe      1038.921 82.11521
## back      1050.914 86.52633
## bad       1041.122 81.12267
## bag       1042.896 86.40601
## bake      1039.395 81.75292
## balk      1042.560 84.17726
## ball      1035.114 80.50253
## bane      1055.741 83.15418
## bang      1048.652 85.51499
## bar       1042.990 82.57040
## base      1043.169 81.00298
## bash      1036.544 79.31510
## batch     1047.446 84.84039
## bath      1039.152 81.59490
## bathe     1043.921 77.52526
## bead      1037.029 78.75316
## beak      1042.654 84.36902
## beam      1040.067 79.28126
## bear      1040.514 83.76442
## beat      1049.312 84.73015
## beg       1044.947 85.99855
## bib       1048.505 83.32998
## bide      1046.706 84.50432
## big       1044.807 84.21706
## bin       1045.735 83.48756
## birth     1022.948 72.74012
## bit       1046.026 84.01293
## boat      1039.346 79.28319
## bog       1041.345 80.92930
## bomb      1045.924 85.51499
## booth     1039.411 80.25993
## booze     1051.868 86.61126
## botch     1040.909 78.78911
## bud       1034.365 79.13435
## budge     1044.947 84.51831
## bun       1034.580 79.54537
## burp      1033.454 75.26900
## bus       1033.021 79.51769
## bush      1035.837 74.66137
## but       1039.495 80.07003
## cab       1040.445 83.75963
## cage      1038.917 78.57028
## call      1049.019 85.14598
## calm      1047.082 87.14904
## cape      1040.686 78.54311
## cash      1042.069 86.83033
## catch     1036.478 79.55030
## caught    1046.641 86.60318
## cave      1054.375 88.71945
## cease     1045.759 86.50898
## cell      1050.967 92.27458
## chair     1029.443 73.05494
## chalk     1044.948 83.93337
## chase     1059.059 88.57701
## cheap     1052.476 87.50212
## cheat     1049.722 90.15107
## cheek     1035.924 79.15046
## chef      1056.613 91.35619
## chewed    1042.900 80.91128
## chin      1040.826 84.85165
## chirp     1033.836 76.85199
## choke      1054.485 90.25226
## choose     1037.296 79.01745
## chug       1042.780 80.73138
## chum       1047.075 86.13040
## cite       1034.088 80.05828
## coat       1053.603 91.02669
## cone       1038.833 81.21502
## cope       1057.129 90.66422
## core       1048.035 83.86180
## cot        1051.651 87.05225
## couch      1040.402 80.76853
## cough      1045.715 78.56370
## cub        1030.915 82.09372
## cuff       1046.868 86.57878
## cup        1037.053 80.95720
## curl       1033.687 75.00316
## curt       1034.329 77.60754
## curve      1051.558 89.37786
## czar       1046.183 82.42064
## dab        1046.619 84.25420
## dash       1037.596 79.10414
## date       1036.570 83.03293
## dawn       1037.905 76.16523
## days       1038.708 81.21410
## dear       1045.196 83.98792
## death      1040.987 80.77037
## deck       1049.092 87.14963
## dice       1035.631 80.64580
## did        1030.882 76.37741
## died       1042.946 80.63662
## dig        1039.994 81.75878
## dim        1040.517 85.43434
## dime       1036.267 75.99753
## dip        1041.740 80.13674
## dire       1048.370 82.92106
## ditch      1058.635 90.70634
## dive       1033.222 79.65590
## dock       1035.795 81.31111
## dodge      1048.735 84.84138
## does       1039.454 82.17606
## dog        1039.891 84.28065
## door       1033.471 77.61839
## dot        1031.332 76.88095
## doubt      1040.535 82.81445
## doug       1035.189 81.62773
## douse      1044.627 86.85945
## dove       1045.114 83.05947
## doze       1039.404 81.31143
## duke       1034.318 77.57756
## dull       1053.447 87.50608
## face       1034.552 76.66398
## fad        1053.157 89.38933
## fake       1047.266 86.25887
## fame       1042.216 76.66431
## fang      1056.606 87.39891
## fate      1042.708 84.33714
## fawn      1050.121 86.16226
## fed       1050.624 86.57669
## fern      1053.202 87.63097
## fetch     1026.848 74.19913
## fief      1038.397 80.43293
## fight     1038.246 82.56978
## fill      1048.568 83.73575
## fin       1055.797 86.71917
## fire      1037.484 78.56712
## fit       1044.405 83.27910
## five      1050.794 84.80889
## fizz      1043.524 81.92919
## foal      1046.642 83.33901
## foam      1038.740 78.93393
## fog       1032.641 76.77797
## foil      1045.297 82.55933
## folk      1041.681 80.78197
## fought    1048.857 84.01326
## foul      1044.304 82.80383
## fudge     1037.576 81.84443
## fuzz      1046.588 85.08887
## gab       1055.214 87.85647
## gag       1040.243 83.27569
## gain      1050.870 85.24610
## gal       1048.022 88.26239
## gall      1045.231 82.68147
## game      1039.056 78.09732
## gang      1048.991 83.11010
## gave      1047.072 84.53769
## gawk      1040.297 82.07802
## gear      1048.268 84.41218
## gel       1040.727 84.26037
## gem       1036.839 77.17619
## get       1038.837 80.07489
## ghoul     1047.729 85.36130
## gig       1049.111 85.43300
## give      1043.300 84.10932
## gnat      1039.071 86.00439
## gnome     1051.898 86.95734
## goal      1047.900 88.15658
## gob       1041.923 80.02940
## gone      1049.454 86.30114
## gong      1042.177 82.48141
## goof      1043.616 82.18386
## goon      1047.555 80.33843
## goose     1043.661 89.58182
## gore      1043.550 87.56212
## got       1028.070 72.92433
## gown      1032.794 77.64283
## guide     1053.576 87.74412
## guile     1049.651 86.22688
## gum       1042.929 81.95106
## hack      1033.212 72.79561
## had       1056.474 89.79896
## hair      1035.562 75.88226
## half      1055.069 90.36741
## hall      1045.000 78.93662
## hash      1045.853 89.40268
## hat       1038.921 79.46541
## hate      1029.835 76.25339
## have      1052.226 89.29442
## head      1053.048 87.21673
## hear      1040.926 79.32770
## heat      1044.058 83.03276
## heed      1044.397 81.81066
## height    1039.468 80.97632
## hem       1047.760 83.68878
## hid       1030.819 77.66556
## hike      1046.925 88.09891
## hill      1045.766 86.79683
## hip       1027.933 75.13224
## hire      1043.358 84.57241
## hiss      1042.363 84.64483
## hit       1040.295 76.04829
## hitch     1044.371 81.72508
## hive      1043.310 83.95394
## hog       1050.315 84.20899
## home      1049.890 87.47960
## hone      1049.453 84.30101
## hoof      1038.460 80.14803
## hook      1029.895 76.23805
## hose      1042.307 82.44219
## house     1041.671 85.98872
## huff      1052.831 84.78250
## hug       1031.841 80.62341
## hull      1044.746 85.12872
## hung      1050.495 89.46243
## hurt      1038.114 80.32461
## hush      1032.224 80.90916
## hut       1057.149 91.93555
## hutch     1041.535 84.58067
## jab       1048.368 82.74089
## jail      1044.345 80.30176
## jam       1045.090 87.85356
## jar       1046.065 82.50543
## jazz      1050.055 87.11871
## jeer      1043.386 90.41949
## jewel     1036.120 80.82956
## job       1051.172 90.10605
## jock      1039.001 74.91880
## jog       1046.207 86.46545
## jowl      1027.884 77.33071
## judge     1036.912 82.00588
## jug       1044.486 84.41407
## juice     1045.157 82.95523
## juke      1046.280 84.24497
## kale      1032.415 77.02197
## kid       1034.977 81.12639
## kill      1046.351 84.34468
## kin       1036.976 80.18414
## kite      1040.837 81.98894
## kneel     1040.972 78.96487
## knit      1058.487 90.19131
## known     1049.701 84.77761
## lab       1041.940 81.94608
## lace      1037.523 81.57204
## lad       1046.570 82.58015
## lamb      1050.551 84.52494
## latch     1035.295 77.17103
## late      1039.832 82.65892
## lawn      1041.522 83.11976
## leaf      1043.802 85.12629
## league    1059.532 92.30749
## leak      1038.067 77.24233
## leave     1055.274 87.47562
## ledge     1038.830 81.44121
## leg       1049.767 84.41878
## less      1046.770 80.75687
## let       1037.587 80.30931
## lied      1060.315 88.13049
## life      1047.393 87.24209
## like      1053.049 89.35589
## lime      1054.262 87.92188
## line      1042.631 80.72758
## load      1034.126 81.92021
## lob       1042.670 82.09142
## lobe      1048.628 84.05318
## lodge     1035.484 77.99533
## log       1046.951 81.64534
## lone      1040.065 82.17622
## loose     1052.169 86.28382
## loot      1042.175 81.71563
## lop       1049.378 88.01628
## lore      1038.699 78.73433
## loud      1042.114 85.30284
## louse     1055.125 90.88437
## lug       1055.800 88.50829
## luge      1057.174 91.53711
## mail      1044.915 85.50500
## maim      1042.724 82.31877
## mall      1048.419 88.54725
## man       1037.297 83.64772
## mass      1036.112 78.22463
## mat       1055.562 89.98432
## math      1049.406 85.80847
## mauve     1054.478 90.32591
## maze      1047.851 86.82599
## mead      1051.825 83.94744
## meal      1032.525 75.84287
## meek      1048.708 93.00300
## men       1031.343 81.29849
## mess      1044.152 82.58487
## might     1044.477 83.70255
## mill      1043.427 88.04717
## miss      1051.055 85.65203
## mitt      1046.520 82.73154
## mob       1046.300 81.09596
## mole      1048.742 87.84246
## moon      1043.335 84.76677
## moot      1046.066 82.33525
## mop       1058.143 89.48150
## mope      1047.581 84.08309
## moth      1043.878 83.01256
## mouse     1040.950 81.26746
## mouth     1043.925 81.07533
## mud       1043.979 82.25270
## muff      1033.502 76.85977
## mum       1051.619 84.76563
## myth      1051.944 87.61106
## nab       1052.703 83.27188
## name      1042.368 85.79220
## nape      1046.271 82.35666
## nash      1044.102 84.36211
## neat      1039.380 79.00131
## neck      1041.276 85.96629
## need      1053.045 84.24183
## nerve     1045.939 83.56611
## newt      1048.357 82.53327
## niece     1037.787 75.62437
## night     1052.631 82.61590
## nod       1063.382 88.83973
## none      1039.936 79.73742
## noon      1049.854 86.69277
## nose      1041.298 79.44302
## not       1048.859 84.43749
## notch     1035.649 81.58988
## null      1036.649 76.29608
## pair      1060.324 93.20100
## pan       1044.253 84.08788
## pass      1047.290 83.56253
## peace     1021.884 76.96561
## peach     1032.528 74.07115
## peak      1043.150 82.73249
## peat      1032.655 76.14694
## peep      1035.021 80.66520
## peeve     1049.469 86.97176
## pen       1045.176 85.58504
## perch     1046.296 87.01466
## perm      1042.549 84.15970
## pet       1035.778 79.20374
## phase     1037.388 81.19846
## phone     1040.226 76.50908
## pick      1048.104 87.14124
## pile      1039.829 84.99861
## pin       1035.327 77.83484
## ping      1038.957 81.51861
## pipe      1038.738 79.05163
## pitch     1049.536 82.15317
## pod       1049.224 82.11430
## poise     1045.467 85.12787
## pooch     1048.726 85.13141
## pool      1031.427 81.95084
## pop       1039.851 83.78204
## pose      1040.228 78.89918
## pouch     1048.508 86.99759
## pour      1020.649 75.59080
## pout      1044.446 85.35959
## puff      1046.550 84.63826
## pun       1046.204 83.41675
## pup       1042.187 82.84784
## putt      1051.320 86.30383
## rack      1052.194 88.42024
## rag       1049.513 85.34206
## raid      1045.075 85.21240
## rail      1043.475 80.06061
## rake      1048.632 85.15192
## ram       1046.203 85.17705
## ran       1033.117 77.91644
## rang      1037.375 82.99769
## rap       1036.655 74.45905
## rash      1062.557 90.51963
## rate      1025.637 75.05728
## rear      1043.327 81.29074
## red       1050.696 82.86375
## ref       1047.366 83.69484
## rev       1041.169 81.84955
## rib       1048.010 89.44286
## rich      1037.605 76.67945
## ridge     1060.859 89.01749
## right     1050.706 91.67912
## ring      1053.756 90.31844
## rise      1044.146 85.83984
## roach     1045.981 82.95101
## rogue     1038.807 80.01361
## roof      1047.225 83.69866
## rook      1041.931 84.79137
## room      1056.964 96.29306
## root      1037.153 75.93280
## rose      1048.279 87.77433
## rot       1038.003 78.05594
## rouge     1047.959 81.11521
## rove      1042.736 82.53234
## rude      1048.663 85.51991
## rule      1045.350 87.68946
## rut       1030.108 76.84240
## sack      1041.282 78.54585
## safe      1047.540 86.21766
## sage      1039.689 82.04669
## sail      1044.078 83.95964
## same      1045.230 85.12613
## sane      1043.497 79.31370
## sang      1043.812 80.94918
## sap       1047.028 84.43259
## sash      1054.596 91.60738
## sauce     1037.759 83.45026
## save      1054.777 88.22356
## seem      1033.716 77.48943
## seen      1035.783 75.03907
## sees      1049.030 85.99757
## serve     1047.696 83.73234
## set       1048.900 87.96605
## sewed     1053.234 86.18329
## shack     1042.167 84.73914
## shade     1046.022 86.01452
## sham      1055.725 89.49446
## shave     1054.706 87.20279
## shear     1031.092 77.33975
## sheath    1052.023 82.78818
## shed      1040.827 84.25490
## sheep     1051.120 85.38790
## sheet     1038.318 79.86360
## shell     1046.058 83.42334
## shim      1053.477 91.12420
## shin      1047.999 83.94251
## shine     1058.401 88.40569
## ship      1058.944 86.36545
## shook     1050.533 86.07391
## shoot     1057.926 87.96604
## shop      1040.877 80.48171
## shore     1041.212 80.23310
## shot      1053.560 85.04706
## shout     1045.631 82.74682
## shown     1047.444 86.41157
## shun      1036.757 82.00077
## shut      1036.340 80.71005
## sick      1043.321 80.00554
## side      1054.130 86.85355
## siege     1053.585 82.58644
## sieve     1049.320 85.29019
## sing      1043.519 77.59433
## sip       1037.098 76.41070
## sit       1055.054 91.27699
## soar      1045.964 83.13868
## sob       1049.354 89.09940
## sock      1041.142 79.43844
## sod       1059.771 90.60537
## soil      1048.732 80.73983
## some      1051.421 89.68939
## song      1053.748 87.10224
## soothe    1048.824 87.14038
## south     1043.370 84.77208
## sown      1055.497 87.34010
## sub       1048.541 83.47903
## such      1041.860 81.79826
## tag       1039.211 80.82305
## tail      1041.578 81.15357
## take      1047.383 83.86045
## talk      1043.455 80.17268
## tall      1045.471 86.83292
## tap       1043.258 80.37825
## tar       1035.998 78.06465
## taught    1043.637 82.89166
## teach     1032.380 76.94186
## tease     1033.791 75.30201
## tech      1039.392 81.65810
## teethe    1069.812 93.19199
## tell      1035.733 78.58390
## ten       1043.146 86.01568
## than      1034.099 81.12552
## thatch    1049.705 84.13233
## their     1039.479 82.88022
## them      1049.661 83.35420
## theme     1038.007 78.09572
## these     1048.204 84.25594
## thin      1046.338 83.67443
## thine     1053.483 91.28760
## those     1037.470 81.00044
## thug      1056.629 88.59633
## thus      1048.983 86.75586
## tide      1055.038 88.95369
## ties      1050.464 84.31065
## tight     1058.002 84.35931
## time      1042.171 79.42648
## tin       1036.582 82.24132
## tip       1046.729 87.79701
## toad      1047.643 84.62191
## toil      1050.046 89.78463
## toll      1040.978 81.53412
## tone      1027.634 76.98445
## top       1050.038 90.07484
## tote      1031.957 74.84672
## touch     1041.758 79.51838
## tough     1048.554 86.93227
## tout      1044.900 86.62056
## town      1048.702 81.11451
## tub       1038.111 77.01854
## tube      1044.586 76.69815
## use       1041.596 83.12445
## vague     1047.785 84.44378
## van       1053.602 85.54447
## vane      1051.523 90.12705
## vase      1042.641 82.30000
## veal      1039.174 79.67361
## verge     1043.832 80.22400
## verse     1046.594 84.56836
## vet       1039.852 80.92729
## vice      1038.963 82.34240
## vile      1044.623 82.69474
## vine      1038.305 80.89252
## vogue     1044.954 77.68266
## voice     1047.961 84.33091
## wade      1054.396 88.92317
## wage      1049.496 92.53743
## walk      1043.360 79.97581
## war       1043.699 83.18414
## watch     1030.853 77.33584
## wave      1043.158 81.72410
## web       1046.780 84.93748
## wed       1050.873 85.64579
## wedge     1049.558 86.09356
## weed      1048.798 84.90528
## week      1042.000 74.80545
## weep      1042.335 81.80562
## well      1041.712 81.23103
## wet       1041.312 79.81610
## whale     1035.809 78.57479
## whole     1051.191 88.42441
## wick      1052.483 85.12265
## wife      1043.994 81.43965
## will      1049.199 84.49695
## wine      1045.220 86.53615
## wipe      1038.198 79.55271
## wire      1034.326 78.78435
## woke      1045.807 85.73515
## womb      1047.489 80.21611
## won       1043.131 86.43072
## wool      1035.060 78.73349
## work      1038.318 75.58144
## wort      1052.710 84.56150
## worth     1040.409 82.85565
## wove      1052.240 87.98742
## wrath     1041.035 80.92774
## wren      1044.410 81.95180
## writhe    1042.264 81.78375
## wrong     1055.399 85.58997
## wrote     1042.828 80.63863
## yak       1044.860 81.99307
## yam       1048.245 90.16341
## yawn      1039.379 81.14485
## year      1037.308 82.97039
## yen       1044.379 83.33265
## yolk      1032.187 81.66645
## young     1032.830 78.03492
## your      1052.891 90.65918
## zeal      1042.742 77.45124
## zing      1045.024 83.55765
##
## $PID
##     (Intercept)   modality
## 301   1024.0669 -16.936415
## 302   1044.1377   1.842626
## 303    882.8306  57.789321
## 304   1232.7544 -27.919775
## 306   1042.3420  33.886517
## 307   1111.3631  -9.939598
## 308   1250.7673  71.164811
## 309    795.2446  15.481913
## 310   1176.1359 104.151748
## 311   1012.9321  29.956301
## 312   1109.8745 153.103877
## 313   1114.2739  29.601126
## 314   1169.1516 -73.242576
## 315    877.9771  17.015017
## 316   1419.5221 -37.700499
## 318    945.4723  58.220398
## 319   1017.4086  98.276001
## 320    987.4941 101.341092
## 321   1025.1270 139.711268
## 324   1031.3305 136.247265
## 325    826.6336  34.599840
## 326   1048.6754  40.359206
## 328   1236.8999 128.530565
## 329   1042.3467  10.735150
## 331   1406.8491 160.773353
## 333   1644.5199  56.375077
## 334    943.7441  93.903838
## 335   1171.5855  64.055496
## 337    872.3788 169.983181
## 338    806.7056 121.221917
## 340   1179.8487 104.695840
## 341    867.6691 123.864757
## 342   1253.5632  30.102347
## 343    987.7640 208.128162
## 344   1027.5597  96.902630
## 346    895.7248  36.422797
## 348    755.2919 188.633574
## 349    940.6337  54.966496
## 350   1073.1133 302.721545
## 351   1120.9818 214.077478
## 352    796.4338  78.590321
## 353   1103.8173 155.151474
## 354   1074.6543 154.892038
## 355   1192.9344  89.264188
## 356    907.2052 397.764355
## 357    910.5673  81.928193
## 358    963.0962  33.338247
## 359   1087.6474 -39.273777
## 360   1070.4021  26.771336
## 361    982.1810 131.300510
## 362    953.3700  56.194590
## 363    920.7194  43.641068
## 364   1003.5099  75.844905
##
## attr(,"class")
## [1] "coef.mer"
```

Reporting

To report the findings described in the example above, you could write:

“A likelihood-ratio test indicated that the model including modality provided a better fit for the data than a model without it, X2(1) = 32.39, $p < .001$. Examination of the summary output for the full model indicated that response times were on average an estimated 83 ms slower in the audiovisual relative to the audio-only condition(B = 83.18, SE = 12.58, $t = 6.62$)”.

Testing for an interaction between modality and SNR

Load the data. Note that it’s actually the same as the original data frame, but it has an extra column containing SNR. We could have been dealing with this data frame the whole time, but having an extra variable that we’re not using can be confusing, so I waited to introduce it until now.

```
rt_data_interaction <- read_csv("Data/rt_dummy_data_interaction.csv")
```

```
## Rows: 21679 Columns: 5
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (3): SNR, modality, stim
## dbl (2): PID, RT
##
## 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.
```

Dummy code modality and SNR so that audio-only and easy are the reference levels

```
rt_data_interaction$modality <- ifelse(rt_data_interaction$modality == "Audio-only", 0, 1)
rt_data_interaction$SNR <- ifelse(rt_data_interaction$SNR == "Easy", 0, 1)
```

Full Model

Build the full model, which includes all by-participant and by-item random effects except the interaction between modality and SNR, which was not included because in my experience models with random effects structures that complex will almost certainly encounter estimation issues for this kind of data and we will need to simplify the random effects structure anyway. I also want to avoid having overly complex random effects structures because this can limit power (see Matuschek et al., 2017).

```
rt_int.mod <- lmer(RT ~ 1 + modality + SNR + modality:SNR +
                     (1 + modality + SNR|stim) + (1 + modality + SNR|PID),
                   data = rt_data_interaction)
```

```
## Warning in checkConv(attr(opt, "derivs"), opt$par, ctrl = control$checkConv, :
## Model failed to converge with max|grad| = 0.00694594 (tol = 0.002, component 1)
```

All_fit()

This model produced a singular fit, indicating that there are some problems with estimation going on. We'll try using the all_fit() function from the afex package to see if another optimizer will work.

```
all_fit(rt_int.mod)
```

```
## bobyqa. :

## boundary (singular) fit: see ?isSingular

## [OK]
## Nelder_Mead. :

## boundary (singular) fit: see ?isSingular

## Warning: Model failed to converge with 1 negative eigenvalue: -4.9e+02

## [OK]
## optimx.nlminb :

## boundary (singular) fit: see ?isSingular

## [OK]
## optimx.L-BFGS-B :

## boundary (singular) fit: see ?isSingular

## [OK]
## nloptwrap.NLOPT_LN_NELDERMEAD :

## Warning in checkConv(attr(opt, "derivs"), opt$par, ctrl = control$checkConv, :
## Model failed to converge with max|grad| = 0.00694594 (tol = 0.002, component 1)

## [OK]
## nloptwrap.NLOPT_LN_BOBYQA :

## Warning in checkConv(attr(opt, "derivs"), opt$par, ctrl = control$checkConv, :
## Model failed to converge with max|grad| = 0.00694594 (tol = 0.002, component 1)

## [OK]
## nmkbw. :

## boundary (singular) fit: see ?isSingular

## [OK]
## $bobyqa.
## Linear mixed model fit by REML ['lmerModLmerTest']
## Formula: RT ~ 1 + modality + SNR + modality:SNR + (1 + modality + SNR |
##     stim) + (1 + modality + SNR | PID)
##    Data: rt_data_interaction
## REML criterion at convergence: 301137.3
## Random effects:
##  Groups   Name        Std.Dev. Corr
##  stim     (Intercept)  18.184
##           modality      6.729   1.00
##           SNR           3.401  -1.00 -1.00
##  PID      (Intercept) 159.765
##           modality     89.722  -0.03
##           SNR         101.747   0.02 -0.47
##  Residual             247.491
## Number of obs: 21679, groups:  stim, 543; PID, 53
## Fixed Effects:
##  (Intercept)      modality           SNR  modality:SNR
##       998.82         98.52         92.36        -29.56
## optimizer (bobyqa) convergence code: 0 (OK) ; 0 optimizer warnings; 1 lme4 warnings
##
## $Nelder_Mead.
## Linear mixed model fit by REML ['lmerModLmerTest']
## Formula: RT ~ 1 + modality + SNR + modality:SNR + (1 + modality + SNR |
##     stim) + (1 + modality + SNR | PID)
##    Data: rt_data_interaction
## REML criterion at convergence: 301152.4
## Random effects:
##  Groups   Name        Std.Dev. Corr
##  stim     (Intercept)   0.000
##           modality     18.122   NaN
##           SNR           9.198   NaN  0.96
##  PID      (Intercept) 186.856
##           modality    101.671  -0.32
##           SNR         109.862   0.27 -0.56
##  Residual             247.651
## Number of obs: 21679, groups:  stim, 543; PID, 53
## Fixed Effects:
##  (Intercept)      modality           SNR  modality:SNR
##       998.80         98.49         92.34        -29.52
## optimizer (Nelder_Mead) convergence code: 0 (OK) ; 0 optimizer warnings; 1 lme4 warnings
##
## $optimx.nlminb
## Linear mixed model fit by REML ['lmerModLmerTest']
## Formula: RT ~ 1 + modality + SNR + modality:SNR + (1 + modality + SNR |
##     stim) + (1 + modality + SNR | PID)
##    Data: rt_data_interaction
## REML criterion at convergence: 301137.3
## Random effects:
##  Groups   Name        Std.Dev. Corr
##  stim     (Intercept)  18.183
##           modality      6.730   1.00
##           SNR           3.402  -1.00 -1.00
##  PID      (Intercept) 159.763
##           modality     89.721  -0.03
##           SNR         101.749   0.02 -0.47
##  Residual             247.491
## Number of obs: 21679, groups:  stim, 543; PID, 53
## Fixed Effects:
##  (Intercept)      modality           SNR  modality:SNR
##       998.82         98.52         92.36        -29.56
## optimizer (optimx) convergence code: 0 (OK) ; 0 optimizer warnings; 1 lme4 warnings
##
## $‘optimx.L-BFGS-B‘
## Linear mixed model fit by REML [’lmerModLmerTest’]
## Formula: RT ~ 1 + modality + SNR + modality:SNR + (1 + modality + SNR |
##     stim) + (1 + modality + SNR | PID)
##    Data: rt_data_interaction
## REML criterion at convergence: 301137.3
## Random effects:
##  Groups   Name        Std.Dev. Corr
##  stim     (Intercept)  18.170
##           modality      6.733   1.00
##           SNR           3.396  -1.00 -1.00
##  PID      (Intercept) 159.842
##           modality     89.761  -0.03
##           SNR         101.824   0.02 -0.47
##  Residual             247.491
## Number of obs: 21679, groups:  stim, 543; PID, 53
## Fixed Effects:
##  (Intercept)      modality           SNR  modality:SNR
##       998.82         98.52         92.36        -29.56
## optimizer (optimx) convergence code: 0 (OK) ; 0 optimizer warnings; 1 lme4 warnings
##
## $nloptwrap.NLOPT_LN_NELDERMEAD
## Linear mixed model fit by REML [’lmerModLmerTest’]
## Formula: RT ~ 1 + modality + SNR + modality:SNR + (1 + modality + SNR |
##     stim) + (1 + modality + SNR | PID)
##    Data: rt_data_interaction
## REML criterion at convergence: 301137.3
## Random effects:
##  Groups   Name        Std.Dev. Corr
##  stim     (Intercept)  18.202
##           modality      6.728   1.00
##           SNR           3.422  -1.00 -1.00
##  PID      (Intercept) 159.689
##           modality     89.682  -0.03
##           SNR         101.712   0.02 -0.47
##  Residual             247.491
## Number of obs: 21679, groups:  stim, 543; PID, 53
## Fixed Effects:
##  (Intercept)      modality           SNR  modality:SNR
##       998.82         98.52         92.36        -29.56
## optimizer (nloptwrap) convergence code: 0 (OK) ; 0 optimizer warnings; 1 lme4 warnings
##
## $nloptwrap.NLOPT_LN_BOBYQA
## Linear mixed model fit by REML [’lmerModLmerTest’]
## Formula: RT ~ 1 + modality + SNR + modality:SNR + (1 + modality + SNR |
##     stim) + (1 + modality + SNR | PID)
##    Data: rt_data_interaction
## REML criterion at convergence: 301137.3
## Random effects:
##  Groups   Name        Std.Dev. Corr
##  stim     (Intercept)  18.202
##           modality      6.728   1.00
##           SNR           3.422  -1.00 -1.00
##  PID      (Intercept) 159.689
##           modality     89.682  -0.03
##           SNR         101.712   0.02 -0.47
##  Residual             247.491
## Number of obs: 21679, groups:  stim, 543; PID, 53
## Fixed Effects:
##  (Intercept)      modality           SNR  modality:SNR
##       998.82         98.52         92.36        -29.56
## optimizer (nloptwrap) convergence code: 0 (OK) ; 0 optimizer warnings; 1 lme4 warnings
##
## $nmkbw.
## Linear mixed model fit by REML ['lmerModLmerTest']
## Formula: RT ~ 1 + modality + SNR + modality:SNR + (1 + modality + SNR |
##     stim) + (1 + modality + SNR | PID)
##    Data: rt_data_interaction
## REML criterion at convergence: 301137.3
## Random effects:
##  Groups   Name        Std.Dev. Corr
##  stim     (Intercept)  18.187
##           modality      6.725   1.00
##           SNR           3.401  -1.00 -1.00
##  PID      (Intercept) 159.763
##           modality     89.731  -0.03
##           SNR         101.740   0.02 -0.47
##  Residual             247.491
## Number of obs: 21679, groups:  stim, 543; PID, 53
## Fixed Effects:
##  (Intercept)      modality           SNR  modality:SNR
##       998.82         98.52         92.36        -29.56
## optimizer (nmkbw) convergence code: 0 (OK) ; 0 optimizer warnings; 1 lme4 warnings
```

Likelihood Ratio Tests

All of these produced a singular fit, and the estimation issues seem to be coming from the item random effects. Given that all the optimizers produced very similar estimates for fixed and random effects, and the item random effects (particularly the slopes) are contributing very little to the total variance using all of the optimizers, we'll try removing the by-item random slopes for modality or SNR, and testing those against the full model via likelihood ratio tests to see if we can remove those (refit = FALSE because we are testing random effects, not fixed effects).

```
rt_int.mod <- lmer(RT ~ 1 + modality + SNR + modality:SNR +
                      (1 + modality + SNR|stim) + (1 + modality + SNR|PID),
                    data = rt_data_interaction)
```

```
## Warning in checkConv(attr(opt, "derivs"), opt$par, ctrl = control$checkConv, :
```

```
## Model failed to converge with max|grad| = 0.00694594 (tol = 0.002, component 1)
```

```
rt_int_no_modality_stim.mod <- lmer(RT ~ 1 + modality + SNR + modality:SNR +
                     (1 + SNR|stim) + (1 + modality + SNR|PID),
                   data = rt_data_interaction)
```

```
## boundary (singular) fit: see ?isSingular
```

```
rt_int_no_SNR_stim.mod <- lmer(RT ~ 1 + modality + SNR + modality:SNR +
                     (1 + modality|stim) + (1 + modality + SNR|PID),
                   data = rt_data_interaction)
```

```
## boundary (singular) fit: see ?isSingular
```

```
anova(rt_int_no_modality_stim.mod, rt_int.mod, refit = FALSE)
```

```
## Data: rt_data_interaction
## Models:
## rt_int_no_modality_stim.mod: RT ~ 1 + modality + SNR + modality:SNR + (1 + SNR | stim) + (1 + modali
## rt_int.mod: RT ~ 1 + modality + SNR + modality:SNR + (1 + modality + SNR | stim) + (1 + modality + S
##                             npar    AIC    BIC  logLik deviance  Chisq Df
## rt_int_no_modality_stim.mod   14 301166 301278 -150569   301138
## rt_int.mod                    17 301171 301307 -150569   301137 0.9413  3
##                             Pr(>Chisq)
## rt_int_no_modality_stim.mod
## rt_int.mod                      0.8154
```

```
anova(rt_int_no_SNR_stim.mod, rt_int.mod, refit = FALSE)
```

```
## Data: rt_data_interaction
## Models:
## rt_int_no_SNR_stim.mod: RT ~ 1 + modality + SNR + modality:SNR + (1 + modality | stim) + (1 + modali
## rt_int.mod: RT ~ 1 + modality + SNR + modality:SNR + (1 + modality + SNR | stim) + (1 + modality + S
##                        npar    AIC    BIC  logLik deviance  Chisq Df Pr(>Chisq)
## rt_int_no_SNR_stim.mod   14 301166 301277 -150569   301138
## rt_int.mod               17 301171 301307 -150569   301137 0.2683  3     0.9659
```

It looks like the model with both random slopes does not differ from either reduced model, so we'll start by removing the random slope that is contributing less to the total variance according to all previous models (the by-item random slope for SNR).

```
rt_int.mod <- lmer(RT ~ 1 + modality + SNR + modality:SNR +
                     (1 + modality|stim) + (1 + modality + SNR|PID),
                   data = rt_data_interaction)
```

```
## boundary (singular) fit: see ?isSingular
```

This one produced a singular fit (we already knew that would happen because we built the same model above), so let's try all_fit()

```r
all_fit(rt_int.mod)
```

```
## bobyqa. :

## boundary (singular) fit: see ?isSingular

## [OK]
## Nelder_Mead. :

## Warning in checkConv(attr(opt, "derivs"), opt$par, ctrl = control$checkConv, :
## unable to evaluate scaled gradient

## Warning in checkConv(attr(opt, "derivs"), opt$par, ctrl = control$checkConv, :
## Model failed to converge: degenerate Hessian with 1 negative eigenvalues

## Warning: Model failed to converge with 1 negative eigenvalue: -4.1e+03

## [OK]
## optimx.nlminb :

## Warning in checkConv(attr(opt, "derivs"), opt$par, ctrl = control$checkConv, :
## Model failed to converge with max|grad| = 0.00213471 (tol = 0.002, component 1)

## [OK]
## optimx.L-BFGS-B :

## boundary (singular) fit: see ?isSingular

## [OK]
## nloptwrap.NLOPT_LN_NELDERMEAD :

## boundary (singular) fit: see ?isSingular

## [OK]
## nloptwrap.NLOPT_LN_BOBYQA :

## boundary (singular) fit: see ?isSingular

## [OK]
## nmkbw. :

## Warning in checkConv(attr(opt, "derivs"), opt$par, ctrl = control$checkConv, :
## Model failed to converge with max|grad| = 0.00257822 (tol = 0.002, component 1)

## [OK]
## $bobyqa.
## Linear mixed model fit by REML [’lmerModLmerTest’]
## Formula: RT ~ 1 + modality + SNR + modality:SNR + (1 + modality | stim) +
##     (1 + modality + SNR | PID)
##    Data: rt_data_interaction
## REML criterion at convergence: 301137.6
## Random effects:
##  Groups   Name        Std.Dev. Corr
##  stim     (Intercept)  16.414
##           modality      6.928  1.00
##  PID      (Intercept) 159.747
##           modality     89.694  -0.03
##           SNR         101.742   0.02 -0.47
##  Residual             247.498
## Number of obs: 21679, groups:  stim, 543; PID, 53
## Fixed Effects:
##  (Intercept)      modality           SNR  modality:SNR
##       998.82         98.52         92.36        -29.55
## optimizer (bobyqa) convergence code: 0 (OK) ; 0 optimizer warnings; 1 lme4 warnings
##
## $Nelder_Mead.
## Linear mixed model fit by REML [’lmerModLmerTest’]
## Formula: RT ~ 1 + modality + SNR + modality:SNR + (1 + modality | stim) +
##     (1 + modality + SNR | PID)
##    Data: rt_data_interaction
## REML criterion at convergence: 301167.5
## Random effects:
##  Groups   Name        Std.Dev. Corr
##  stim     (Intercept)   3.174
##           modality     26.456  -0.74
##  PID      (Intercept) 251.305
##           modality     89.545  -0.06
##           SNR         116.118  -0.21 -0.45
##  Residual             247.522
## Number of obs: 21679, groups:  stim, 543; PID, 53
## Fixed Effects:
##  (Intercept)      modality           SNR  modality:SNR
##       998.79         98.48         92.29        -29.48
## optimizer (Nelder_Mead) convergence code: 0 (OK) ; 0 optimizer warnings; 2 lme4 warnings
##
## $optimx.nlminb
## Linear mixed model fit by REML [’lmerModLmerTest’]
## Formula: RT ~ 1 + modality + SNR + modality:SNR + (1 + modality | stim) +
##     (1 + modality + SNR | PID)
##    Data: rt_data_interaction
## REML criterion at convergence: 301137.6
## Random effects:
##  Groups   Name        Std.Dev. Corr
##  stim     (Intercept)  16.410
##           modality      6.929  1.00
##  PID      (Intercept) 159.723
##           modality     89.684  -0.03
##           SNR         101.735   0.02 -0.47
##  Residual             247.498
## Number of obs: 21679, groups:  stim, 543; PID, 53
## Fixed Effects:
##  (Intercept)      modality           SNR  modality:SNR
##       998.82         98.52         92.36        -29.55
## optimizer (optimx) convergence code: 0 (OK) ; 0 optimizer warnings; 1 lme4 warnings
##
## $‘optimx.L-BFGS-B‘
## Linear mixed model fit by REML ['lmerModLmerTest']
## Formula: RT ~ 1 + modality + SNR + modality:SNR + (1 + modality | stim) +
##     (1 + modality + SNR | PID)
##    Data: rt_data_interaction
## REML criterion at convergence: 301137.6
## Random effects:
##  Groups   Name        Std.Dev. Corr
##  stim     (Intercept)  16.414
##           modality      6.942  1.00
##  PID      (Intercept) 159.621
##           modality     89.697  -0.03
##           SNR         101.770   0.02 -0.47
##  Residual             247.498
## Number of obs: 21679, groups:  stim, 543; PID, 53
## Fixed Effects:
##  (Intercept)      modality           SNR  modality:SNR
##       998.82         98.52         92.36        -29.55
## optimizer (optimx) convergence code: 0 (OK) ; 0 optimizer warnings; 1 lme4 warnings
##
## $nloptwrap.NLOPT_LN_NELDERMEAD
## Linear mixed model fit by REML ['lmerModLmerTest']
## Formula: RT ~ 1 + modality + SNR + modality:SNR + (1 + modality | stim) +
##     (1 + modality + SNR | PID)
##    Data: rt_data_interaction
## REML criterion at convergence: 301137.6
## Random effects:
##  Groups   Name        Std.Dev. Corr
##  stim     (Intercept)  16.416
##           modality      6.925  1.00
##  PID      (Intercept) 159.747
##           modality     89.694  -0.03
##           SNR         101.748   0.02 -0.47
##  Residual             247.498
## Number of obs: 21679, groups:  stim, 543; PID, 53
## Fixed Effects:
##  (Intercept)      modality           SNR  modality:SNR
##       998.82         98.52         92.36        -29.55
## optimizer (nloptwrap) convergence code: 0 (OK) ; 0 optimizer warnings; 1 lme4 warnings
##
## $nloptwrap.NLOPT_LN_BOBYQA
## Linear mixed model fit by REML ['lmerModLmerTest']
## Formula: RT ~ 1 + modality + SNR + modality:SNR + (1 + modality | stim) +
##     (1 + modality + SNR | PID)
##    Data: rt_data_interaction
## REML criterion at convergence: 301137.6
## Random effects:
##  Groups   Name        Std.Dev. Corr
##  stim     (Intercept)  16.416
##           modality      6.925  1.00
##  PID      (Intercept) 159.747
##           modality     89.694  -0.03
##           SNR         101.748   0.02 -0.47
##  Residual             247.498
## Number of obs: 21679, groups:  stim, 543; PID, 53
## Fixed Effects:
##  (Intercept)      modality           SNR  modality:SNR
##       998.82         98.52         92.36        -29.55
## optimizer (nloptwrap) convergence code: 0 (OK) ; 0 optimizer warnings; 1 lme4 warnings
##
## $nmkbw.
## Linear mixed model fit by REML ['lmerModLmerTest']
## Formula: RT ~ 1 + modality + SNR + modality:SNR + (1 + modality | stim) +
##     (1 + modality + SNR | PID)
##    Data: rt_data_interaction
## REML criterion at convergence: 301137.6
## Random effects:
##  Groups   Name        Std.Dev. Corr
##  stim     (Intercept)  16.415
##           modality      6.926  1.00
##  PID      (Intercept) 159.747
##           modality     89.693  -0.03
##           SNR         101.742   0.02 -0.47
##  Residual             247.498
## Number of obs: 21679, groups:  stim, 543; PID, 53
## Fixed Effects:
##  (Intercept)      modality           SNR  modality:SNR
##       998.82         98.52         92.36        -29.55
## optimizer (nmkbw) convergence code: 0 (OK) ; 0 optimizer warnings; 1 lme4 warnings
```

The Nelder-Mead optimizer

The Nelder-Mead optimizer might work, so we'll try that one

```
rt_int.mod <- lmer(RT ~ 1 + modality + SNR + modality:SNR +
                     (1 + modality|stim) + (1 + modality + SNR|PID),
                   data = rt_data_interaction,
                   control = lmerControl(optimizer = 'Nelder_Mead'))
```

```
## Warning in checkConv(attr(opt, "derivs"), opt$par, ctrl = control$checkConv, :
## unable to evaluate scaled gradient
```

```
## Warning in checkConv(attr(opt, "derivs"), opt$par, ctrl = control$checkConv, :
## Model failed to converge: degenerate Hessian with 1 negative eigenvalues
```

```
## Warning: Model failed to converge with 1 negative eigenvalue: -4.1e+03
```

That led to convergence issues. It looks like all of these optimizers lead to estimation issues, so we'll try removing the correlation between the random intercept for stimulus and the by-stimulus random slope for modality (this is ok in this situation because we aren't actually interested in that correlation).

```
rt_int.mod <- lmer(RT ~ 1 + modality + SNR + modality:SNR +
                     (0 + modality|stim) + (1|stim) + (1 + modality + SNR|PID),
                   data = rt_data_interaction)
```

This led to a convergence warning, so we'll try all_fit() again

```
all_fit(rt_int.mod)
```

```
## bobyqa. : [OK]
## Nelder_Mead. : [OK]
## optimx.nlminb : [OK]
## optimx.L-BFGS-B :

## Warning in checkConv(attr(opt, "derivs"), opt$par, ctrl = control$checkConv, :
## Model failed to converge with max|grad| = 0.0135467 (tol = 0.002, component 1)

## [OK]
## nloptwrap.NLOPT_LN_NELDERMEAD : [OK]
## nloptwrap.NLOPT_LN_BOBYQA : [OK]
## nmkbw. : [OK]

## $bobyqa.
## Linear mixed model fit by REML ['lmerModLmerTest']
## Formula: RT ~ 1 + modality + SNR + modality:SNR + (0 + modality | stim) +
##     (1 | stim) + (1 + modality + SNR | PID)
##    Data: rt_data_interaction
## REML criterion at convergence: 301138.2
## Random effects:
##  Groups   Name        Std.Dev. Corr
##  stim     modality     12.03
##  stim.1   (Intercept)  18.88
##  PID      (Intercept) 159.76
##           modality     89.69   -0.03
##           SNR         101.76    0.02 -0.47
##  Residual             247.46
## Number of obs: 21679, groups:  stim, 543; PID, 53
## Fixed Effects:
##  (Intercept)      modality           SNR  modality:SNR
##       998.82         98.51         92.34        -29.53
##
## $Nelder_Mead.
## Linear mixed model fit by REML ['lmerModLmerTest']
## Formula: RT ~ 1 + modality + SNR + modality:SNR + (0 + modality | stim) +
##     (1 | stim) + (1 + modality + SNR | PID)
##    Data: rt_data_interaction
## REML criterion at convergence: 301138.2
## Random effects:
##  Groups   Name        Std.Dev. Corr
##  stim     modality     12.03
##  stim.1   (Intercept)  18.88
##  PID      (Intercept) 159.76
##           modality     89.69   -0.03
##           SNR         101.76    0.02 -0.47
##  Residual             247.46
## Number of obs: 21679, groups:  stim, 543; PID, 53
## Fixed Effects:
##  (Intercept)      modality           SNR  modality:SNR
##       998.82         98.51         92.34        -29.53
##
## $optimx.nlminb
## Linear mixed model fit by REML [’lmerModLmerTest’]
## Formula: RT ~ 1 + modality + SNR + modality:SNR + (0 + modality | stim) +
##     (1 | stim) + (1 + modality + SNR | PID)
##    Data: rt_data_interaction
## REML criterion at convergence: 301138.2
## Random effects:
##  Groups   Name        Std.Dev. Corr
##  stim     modality     12.03
##  stim.1   (Intercept)  18.87
##  PID      (Intercept) 159.75
##           modality     89.69   -0.03
##           SNR         101.76    0.02 -0.47
##  Residual             247.46
## Number of obs: 21679, groups:  stim, 543; PID, 53
## Fixed Effects:
##  (Intercept)      modality           SNR  modality:SNR
##       998.82         98.51         92.34        -29.53
##
## $‘optimx.L-BFGS-B‘
## Linear mixed model fit by REML [’lmerModLmerTest’]
## Formula: RT ~ 1 + modality + SNR + modality:SNR + (0 + modality | stim) +
##     (1 | stim) + (1 + modality + SNR | PID)
##    Data: rt_data_interaction
## REML criterion at convergence: 301138.2
## Random effects:
##  Groups   Name        Std.Dev. Corr
##  stim     modality     12.01
##  stim.1   (Intercept)  18.90
##  PID      (Intercept) 159.75
##           modality     89.75   -0.03
##           SNR         101.87    0.02 -0.47
##  Residual             247.45
## Number of obs: 21679, groups:  stim, 543; PID, 53
## Fixed Effects:
##  (Intercept)      modality           SNR  modality:SNR
##       998.82         98.51         92.34        -29.53
## optimizer (optimx) convergence code: 0 (OK) ; 0 optimizer warnings; 1 lme4 warnings
##
## $nloptwrap.NLOPT_LN_NELDERMEAD
## Linear mixed model fit by REML [’lmerModLmerTest’]
## Formula: RT ~ 1 + modality + SNR + modality:SNR + (0 + modality | stim) +
##     (1 | stim) + (1 + modality + SNR | PID)
##    Data: rt_data_interaction
## REML criterion at convergence: 301138.2
## Random effects:
##  Groups   Name        Std.Dev. Corr
##  stim     modality     12.03              
##  stim.1   (Intercept)  18.88              
##  PID      (Intercept) 159.76              
##           modality     89.70   -0.03      
##           SNR         101.76    0.02 -0.47
##  Residual             247.46              
## Number of obs: 21679, groups:  stim, 543; PID, 53
## Fixed Effects:
##  (Intercept)      modality           SNR  modality:SNR  
##       998.82         98.51         92.34        -29.53  
## 
## $nloptwrap.NLOPT_LN_BOBYQA
## Linear mixed model fit by REML [’lmerModLmerTest’]
## Formula: RT ~ 1 + modality + SNR + modality:SNR + (0 + modality | stim) +  
##     (1 | stim) + (1 + modality + SNR | PID)
##    Data: rt_data_interaction
## REML criterion at convergence: 301138.2
## Random effects:
##  Groups   Name        Std.Dev. Corr       
##  stim     modality     12.03              
##  stim.1   (Intercept)  18.88              
##  PID      (Intercept) 159.76              
##           modality     89.70   -0.03      
##           SNR         101.76    0.02 -0.47
##  Residual             247.46              
## Number of obs: 21679, groups:  stim, 543; PID, 53
## Fixed Effects:
##  (Intercept)      modality           SNR  modality:SNR  
##       998.82         98.51         92.34        -29.53  
## 
## $nmkbw.
## Linear mixed model fit by REML [’lmerModLmerTest’]
## Formula: RT ~ 1 + modality + SNR + modality:SNR + (0 + modality | stim) +  
##     (1 | stim) + (1 + modality + SNR | PID)
##    Data: rt_data_interaction
## REML criterion at convergence: 301138.2
## Random effects:
##  Groups   Name        Std.Dev. Corr       
##  stim     modality     12.04              
##  stim.1   (Intercept)  18.88              
##  PID      (Intercept) 159.76              
##           modality     89.70   -0.03      
##           SNR         101.77    0.02 -0.47
##  Residual             247.46              
## Number of obs: 21679, groups:  stim, 543; PID, 53
## Fixed Effects:
##  (Intercept)      modality           SNR  modality:SNR  
##       998.82         98.51         92.34        -29.53
```

The bobyqa optimizer might work, so we’ll try that

```
rt_int.mod <- lmer(RT ~ 1 + modality + SNR + modality:SNR +
                     (0 + modality|stim) + (1|stim) + (1 + modality + SNR|PID),
                   data = rt_data_interaction,
                   control = lmerControl(optimizer = 'bobyqa'))
```

Looks like that converged, but let's examine the random effects output to make sure estimation went smoothly.

Summary

```
summary(rt_int.mod)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: RT ~ 1 + modality + SNR + modality:SNR + (0 + modality | stim) +
##     (1 | stim) + (1 + modality + SNR | PID)
##    Data: rt_data_interaction
## Control: lmerControl(optimizer = "bobyqa")
##
## REML criterion at convergence: 301138.2
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -3.5354 -0.6949 -0.0045  0.5972  4.8706
##
## Random effects:
##  Groups   Name        Variance Std.Dev. Corr
##  stim     modality      144.7   12.03
##  stim.1   (Intercept)   356.3   18.88
##  PID      (Intercept) 25522.7  159.76
##           modality     8044.7   89.69   -0.03
##           SNR         10355.6  101.76    0.02 -0.47
##  Residual             61234.2  247.46
## Number of obs: 21679, groups:  stim, 543; PID, 53
##
## Fixed effects:
##               Estimate Std. Error        df t value Pr(>|t|)
## (Intercept)    998.824     22.214    52.729  44.964  < 2e-16 ***
## modality        98.510     13.199    59.065   7.464 4.41e-10 ***
## SNR             92.339     14.790    58.004   6.243 5.39e-08 ***
## modality:SNR   -29.532      6.755 21298.850  -4.372 1.24e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##              (Intr) modlty SNR
## modality     -0.063
## SNR          -0.014 -0.354
## modalty:SNR   0.074 -0.247 -0.232
```

Looks ok! We'll stick with this one.

Testing for an effect of modality on intelligibility (binomial)

Load data and name it acc_data

```
acc_data <- read_csv("Data/acc_dummy_data.csv")
```

```
## Rows: 28807 Columns: 4
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (2): modality, stim
## dbl (2): PID, acc
##
## 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.
```

Dummy code modality with audio-only as the reference level

```
acc_data$modality <- ifelse(acc_data$modality == "Audio-only", 0, 1)
```

Make PID and stim factors

```
acc_data$PID <- as.factor(acc_data$PID)
acc_data$stim <- as.factor(acc_data$stim)
```

Full Model

Build a full model

```
acc_full.mod <- glmer(acc ~ 1 + modality +
                        (1 + modality|PID) + (1 + modality|stim),
                      data = acc_data,
                      family = binomial)
```

Summary

Check random effects output

```
summary(acc_full.mod)
```

```
## Generalized linear mixed model fit by maximum likelihood (Laplace
##   Approximation) [glmerMod]
##  Family: binomial  ( logit )
## Formula: acc ~ 1 + modality + (1 + modality | PID) + (1 + modality | stim)
##    Data: acc_data
##
##      AIC      BIC   logLik deviance df.resid
##  27988.6  28054.8 -13986.3  27972.6    28799
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -6.2346  0.1411  0.3436  0.5606  2.0937
##
## Random effects:
##  Groups Name        Variance Std.Dev. Corr
##  stim   (Intercept) 0.72085  0.8490
##         modality    0.46663  0.6831   -0.06
##  PID    (Intercept) 0.04346  0.2085
##         modality    0.04903  0.2214   -0.15
## Number of obs: 28807, groups:  stim, 543; PID, 53
##
## Fixed effects:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  0.77987    0.05043   15.46   <2e-16 ***
## modality     1.43094    0.05734   24.95   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##          (Intr)
## modality -0.201
```

Reduced Model

Build a reduced model lacking the fixed effect for modality

```
acc_reduced.mod <- glmer(acc ~ 1 +
                           (1 + modality|PID) + (1 + modality|stim),
                         data = acc_data,
                         family = binomial)
```

Likelihood Ratio Test

Conduct a likelihood ratio test to see if the effect of block (audio-only versus audiovisual) is significant

```
anova(acc_reduced.mod, acc_full.mod)
```

```
## Data: acc_data
## Models:
## acc_reduced.mod: acc ~ 1 + (1 + modality | PID) + (1 + modality | stim)
## acc_full.mod: acc ~ 1 + modality + (1 + modality | PID) + (1 + modality | stim)
##                 npar   AIC   BIC logLik deviance  Chisq Df Pr(>Chisq)
## acc_reduced.mod    7 28147 28205 -14067    28133
## acc_full.mod       8 27989 28055 -13986    27973 160.78  1  < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Fixed-effects only, random intercepts, and random slopes plots

Load data

```r
figuredata <- read_csv("Data/figure_data.csv")
```

```
## Rows: 16 Columns: 3
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## dbl (3): PID, xvar, yvar
##
## 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.
```

Make PID a factor

```r
figuredata$PID <- as.factor(figuredata$PID)
```

Fixed-effects only regression plot

Build regression model and view the summary output to look at the residuals

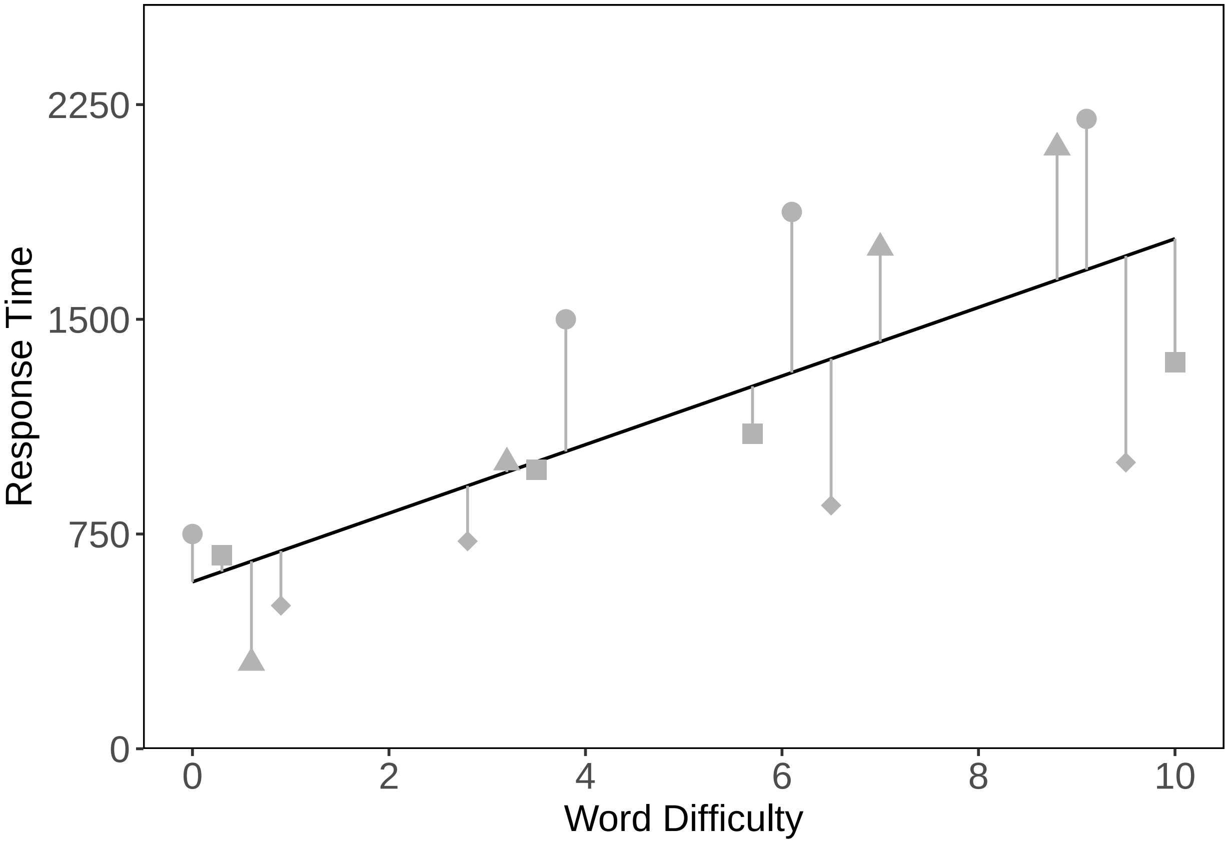
```r
ols.mod <- lm(yvar ~ xvar, data = figuredata)

summary(ols.mod)
```

```
##
## Call:
## lm(formula = yvar ~ xvar, data = figuredata)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -721.36 -233.80    3.13  361.59  561.08
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   582.92     180.88   3.223  0.00614 **
## xvar          119.84      30.64   3.911  0.00157 **
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 410.2 on 14 degrees of freedom
## Multiple R-squared:  0.5221, Adjusted R-squared:  0.4879
## F-statistic: 15.29 on 1 and 14 DF,  p-value: 0.001568
```

Build a fixed effects only plot

```r
ggplot(figuredata, aes(x = xvar, y = yvar)) +
  stat_smooth(method = lm, se = FALSE, linetype = "solid",
              color = "black", size = .6) +
  geom_point(aes(shape = PID), size = 3.25, color = "grey70") +
  scale_shape_manual(values = c(15, 16, 17, 18)) +
  geom_segment(aes(x = xvar, xend = xvar,
                   y = yvar, yend = fitted(ols.mod)),
               color = "grey70") +
  scale_y_continuous(expand = c(0, 0), breaks = c(0, 750, 1500, 2250, 3000),
                     limits = c(0, 2600)) +
  scale_x_continuous(expand = c(0, 0), breaks = c(0, 2, 4, 6, 8, 10),
                     limits = c(-0.5, 10.5)) +
  theme(panel.background = element_blank(),
        panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        panel.border = element_rect(colour = "black", fill = NA),
        legend.position = "none",
        axis.text = element_text(size = 14),
        axis.title = element_text(size = 14)) +
  labs (x = "Word Difficulty", y = "Response Time")
```

```
## `geom_smooth()` using formula 'y ~ x'
```



Save the figure

```
ggsave("Figures/fixed_effects_plot.png", units = "in", width = 9, height = 6, dpi = 300)
```

```
## `geom_smooth()` using formula 'y ~ x'
```

Random intercepts plot

Build the model with random intercepts and view the summary output to look at the residuals

```
random_intercepts.mod <- lmer(yvar ~ 1 + xvar + (1|PID), data = figuredata)

summary(random_intercepts.mod)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: yvar ~ 1 + xvar + (1 | PID)
##    Data: figuredata
##
## REML criterion at convergence: 210.6
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -1.6361 -0.6381  0.3516  0.6020  1.3016
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  PID      (Intercept) 108442   329.3
##  Residual              75324   274.5
## Number of obs: 16, groups:  PID, 4
##
## Fixed effects:
##             Estimate Std. Error      df t value Pr(>|t|)
## (Intercept)  576.362    204.356   5.033   2.820 0.036817 *
## xvar         121.185     20.507  11.001   5.909 0.000102 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##      (Intr)
## xvar -0.488
```

Extract the fixed effects estimates for the intercept and slope

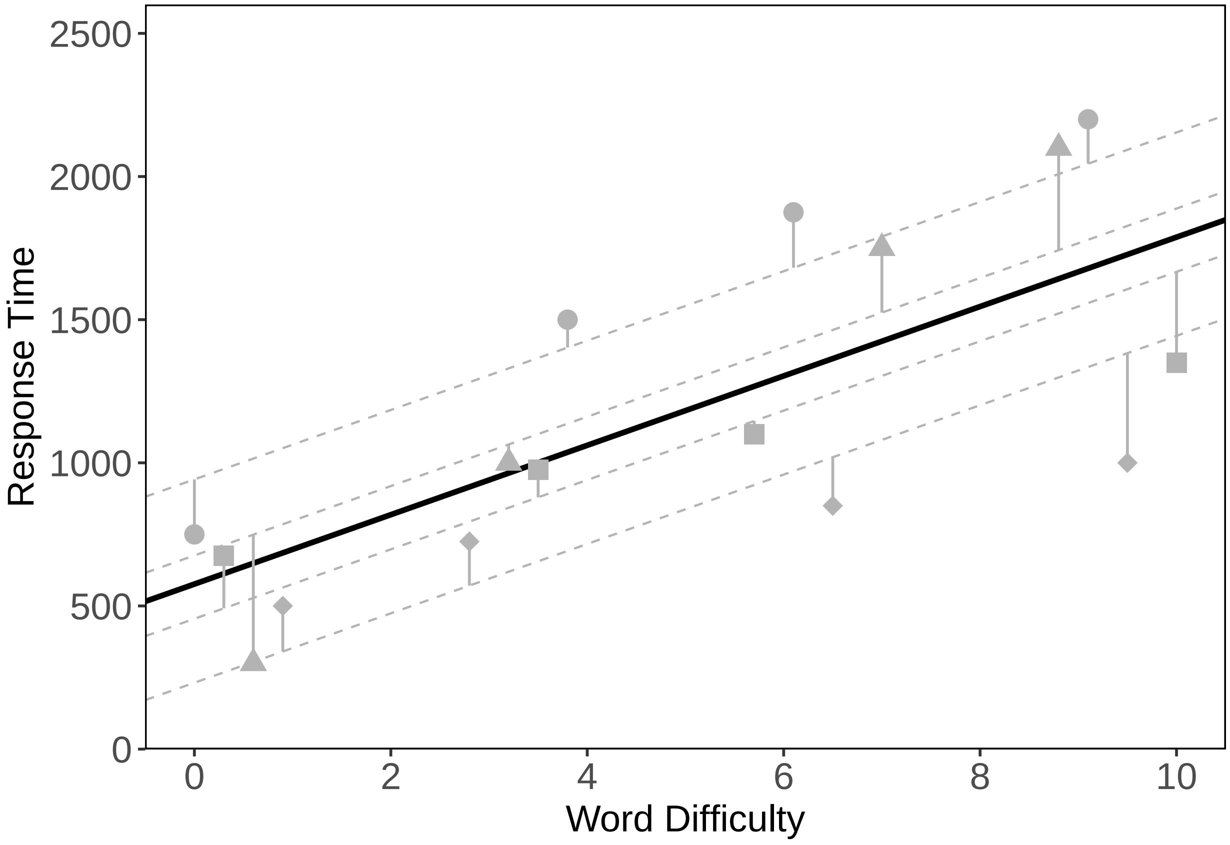
```
model_intercept <- as.numeric(fixef(random_intercepts.mod)[1])
model_slope <- as.numeric(fixef(random_intercepts.mod)[2])
```

Extract the individual participant intercepts for this model and add it to the data frame

```
figuredata$intercepts <- rep(coef(random_intercepts.mod)$PID[,1], each = 4)
```

Build random intercepts plot

```
ggplot(figuredata, aes(x = xvar, y = yvar)) +
  geom_abline(slope = model_slope, intercept = model_intercept,
              linetype = "solid", color = "black", size = 1) +
  geom_abline(mapping = aes(slope = model_slope, intercept = intercepts),
              linetype = "dashed", color = "grey70", size = .4) +
  geom_point(aes(shape = PID), size = 3.25, color = "grey70") +
  scale_shape_manual(values = c(15, 16, 17, 18)) +
  geom_segment(aes(x = xvar, xend = xvar,
                   y = yvar, yend = fitted(random_intercepts.mod)),
               color = "grey70") +
  scale_y_continuous(expand = c(0, 0), breaks = c(0, 500, 1000, 1500, 2000, 2500),
                     limits = c(0, 2600)) +
  scale_x_continuous(expand = c(0, 0), breaks = c(0, 2, 4, 6, 8, 10),
                     limits = c(-0.5, 10.5)) +
  theme(panel.background = element_blank(),
        panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        panel.border = element_rect(colour = "black", fill = NA),
        legend.position = "none",
        axis.text = element_text(size = 14),
        axis.title = element_text(size = 14)) +
  labs (x = "Word Difficulty", y = "Response Time")
```



Save the figure

```
ggsave("Figures/random_intercepts.png", units = "in", width = 9, height = 6, dpi = 300)
```

Random intercepts and slopes plot

Build the model with random intercepts and slopes and view the summary output to look at the residuals

```
random_slopes.mod <- lmer(yvar ~ 1 + xvar + (1 + xvar|PID), data = figuredata)

summary(random_slopes.mod)
```

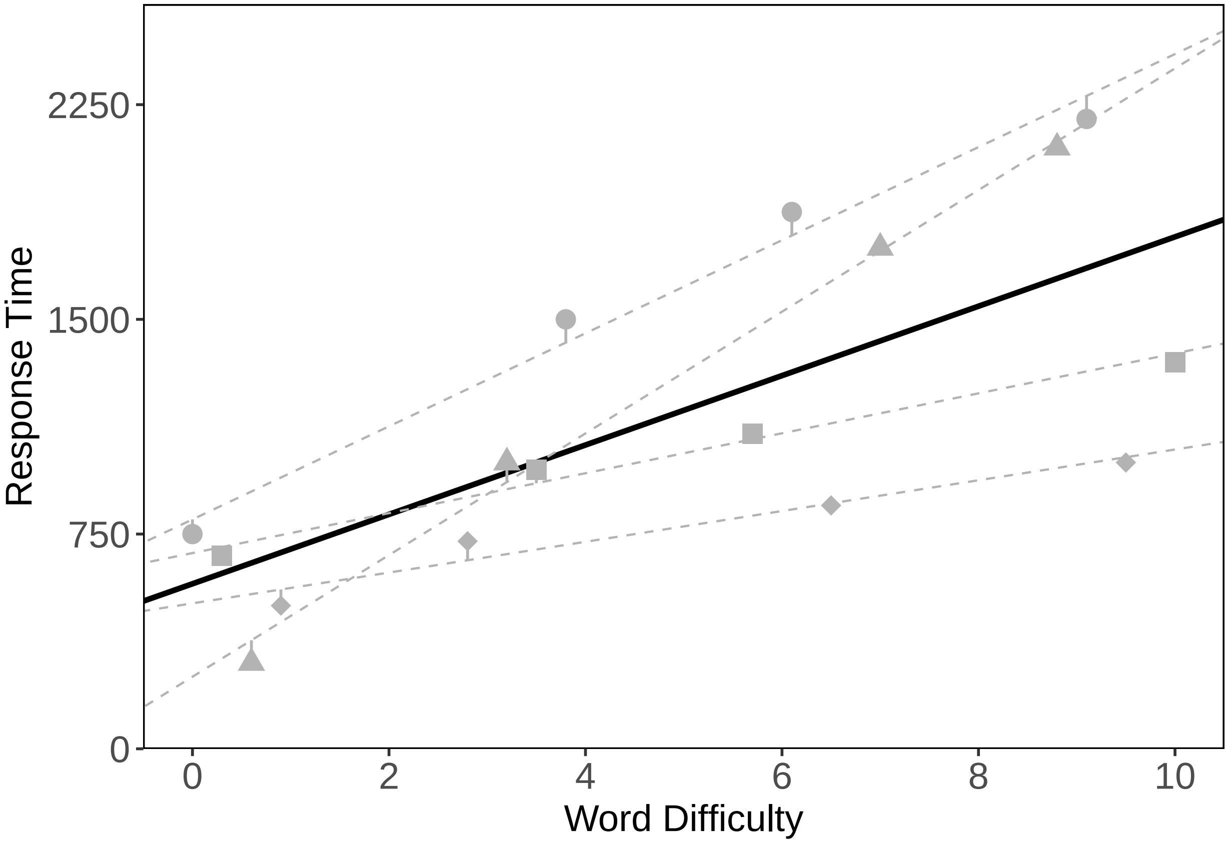
```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: yvar ~ 1 + xvar + (1 + xvar | PID)
##    Data: figuredata
##
## REML criterion at convergence: 191.1
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -1.0716 -0.4797 -0.1726  0.6937  1.0952
##
## Random effects:
##  Groups   Name        Variance Std.Dev. Corr
##  PID      (Intercept) 61192    247.37
##           xvar         5854     76.51   -0.40
##  Residual              5638     75.08
## Number of obs: 16, groups:  PID, 4
##
## Fixed effects:
##             Estimate Std. Error      df t value Pr(>|t|)
## (Intercept)  561.105    128.055   2.990   4.382   0.0222 *
## xvar         124.636     38.667   2.997   3.223   0.0485 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##      (Intr)
## xvar -0.415
```

Extract the individual participant intercepts and slopes from this model and add them to the data frame

```
figuredata$intercepts2 <- rep(coef(random_slopes.mod)$PID[,1], each = 4)
figuredata$slopes <- rep(coef(random_slopes.mod)$PID[,2], each = 4)
```

Build plot

```
ggplot(figuredata, aes(x = xvar, y = yvar)) +
  geom_abline(slope = model_slope, intercept = model_intercept,
              linetype = "solid", color = "black", size = 1) +
  geom_abline(mapping = aes(slope = slopes,
                            intercept = intercepts2, linetype = PID),
              linetype = "dashed", color = "grey70", size = .4) +
  geom_point(aes(shape = PID), size = 3.25, color = "grey70") +
  scale_shape_manual(values = c(15, 16, 17, 18)) +
  geom_segment(aes(x = xvar, xend = xvar,
                   y = yvar, yend = fitted(random_slopes.mod)),
               color = "grey70") +
  scale_y_continuous(expand = c(0, 0), breaks = c(0, 750, 1500, 2250),
                     limits = c(0, 2600)) +
  scale_x_continuous(expand = c(0, 0), breaks = c(0, 2, 4, 6, 8, 10),
                     limits = c(-0.5, 10.5)) +
  theme(panel.background = element_blank(),
        panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        panel.border = element_rect(colour = "black", fill = NA),
        legend.position = "none",
        axis.text = element_text(size = 14),
        axis.title = element_text(size = 14)) +
  labs (x = "Word Difficulty", y = "Response Time")
```



Save the figure

```
ggsave("Figures/random_slopes.png", units = "in", width = 9, height = 6, dpi = 300)
```

Resources to common questions:

1. How many cluster do I need to fit a Mixed Effects model?

- Video by CenterStat: https://www.youtube.com/watch?v=aKXcayBhbMc
- Review by McNeish & Stapleton (2016)

1. Intraclass Correlation:

- If ICC = 0 then there are no between-group differences to produce dependence. The nesting of the data is irrelevant.
-
  - This is the assumption of GLM

- - Multilevel model reduces to GLM when ICC = 0.
- If ICC = 1 then all differences are between-groups differences and individuals within a group have identical scores.
- ICCs of .1 to .3 are common for individuals within groups.