MP MetaAnalysis Monolinguals

Christina Bergmann

Contents


```
## Loading tidyverse: ggplot2
## Loading tidyverse: tibble
## Loading tidyverse: tidyr
## Loading tidyverse: readr
## Loading tidyverse: purrr
## Loading tidyverse: dplyr

## Conflicts with tidy packages ----------------------------------------------

## filter(): dplyr, stats
## lag():    dplyr, stats

## Loading required package: Matrix

##
## Attaching package: 'Matrix'

## The following object is masked from 'package:tidyr':
##
##     expand

## Loading 'metafor' package (version 1.9-9). For an overview
## and introduction to the package please type: help(metafor).

## Loading 'meta' package (version 4.9-0).
## Type 'help(meta)' for a brief overview.
```

```
##
## Attaching package: 'meta'

## The following objects are masked from 'package:metafor':
##
##     baujat, forest, funnel, funnel.default, labbe, radial,
##     trimfill
```

Preparation

Read in data and tidy up dataset

Plotting defaults

```
# Themes and plot
apatheme = theme_bw() + theme(panel.grid.major = element_blank(), panel.grid.minor = element_blank(),
    panel.border = element_blank(), axis.line = element_line(), text = element_text(family = "Times",
        size = 25))

# Color Blind palette:
cbPalette <- c("#E69F00", "#56B4E9", "#009E73", "#F0E442", "#0072B2", "#D55E00",
    "#CC79A7")
```

Descriptive data

The database contains data from 32 papers. In the tables below, we provide more descriptive information.

The next table shows what type of publications were included in our meta-analysis

publication_status	n_unique	count
dissertation	2	17
gray paper	2	14
paper	27	190
proceedings	1	4

Type of data on which we calculated effect sizes

The table below shows based on which data we calculated effect sizes.

es_method	n_unique	count
group_means_one	18	102
group_means_two	7	57
t_one	4	35
t_two	5	31

Number of unique infants

The database contains data from 2072 unique infants.

Number of unique experimental conditions

The database contains data from 223 unique experimental conditions

Type of comparison of the time-course data calculated

We also have different ways of comparison of the time-course data, as the next table shows.

within_measure_descriptive	n_unique	count
post-naming compared to pre-naming phase	10	27
post-naming phase compared with chance (=50%)	9	20
post-pre difference score compared with chance (=0)	13	45

Type of distractor

This is a summary of the type of distractor used in an experiment

object_pair	count
familiar_familiar	23
familiar_novel	10

Whether word was pronounced both correctly as well as mispronounced

This is a summary of whether an experiment had both correct and mispronounced versions of the word in the experiment

word_correct_and_MP	count
	2
no	10
yes	21

Size of analysis time window

Where possible, we noted the time window for analysis. First, let's look at the offset (in milliseconds) after the start of the word, i.e. the begin of a give analysis window for a naming effect

```
offset_info <- time_wind_dat %>% group_by(offset) %>% summarize(count = n())


kable(offset_info)
```

offset	count
0	3
200	1
231	1
267	1
300	1

offset	count
360	5
365	1
367	14
400	1
500	1
1133	1
NA	4

Duration of post naming window

Next we look at duration (in seconds) of the post naming window, here,too, we see a lot of heterogeneity.

```
duration_info <- db_ET_correct %>% group_by(post_nam_dur) %>% summarize(count = n())


kable(duration_info)
```

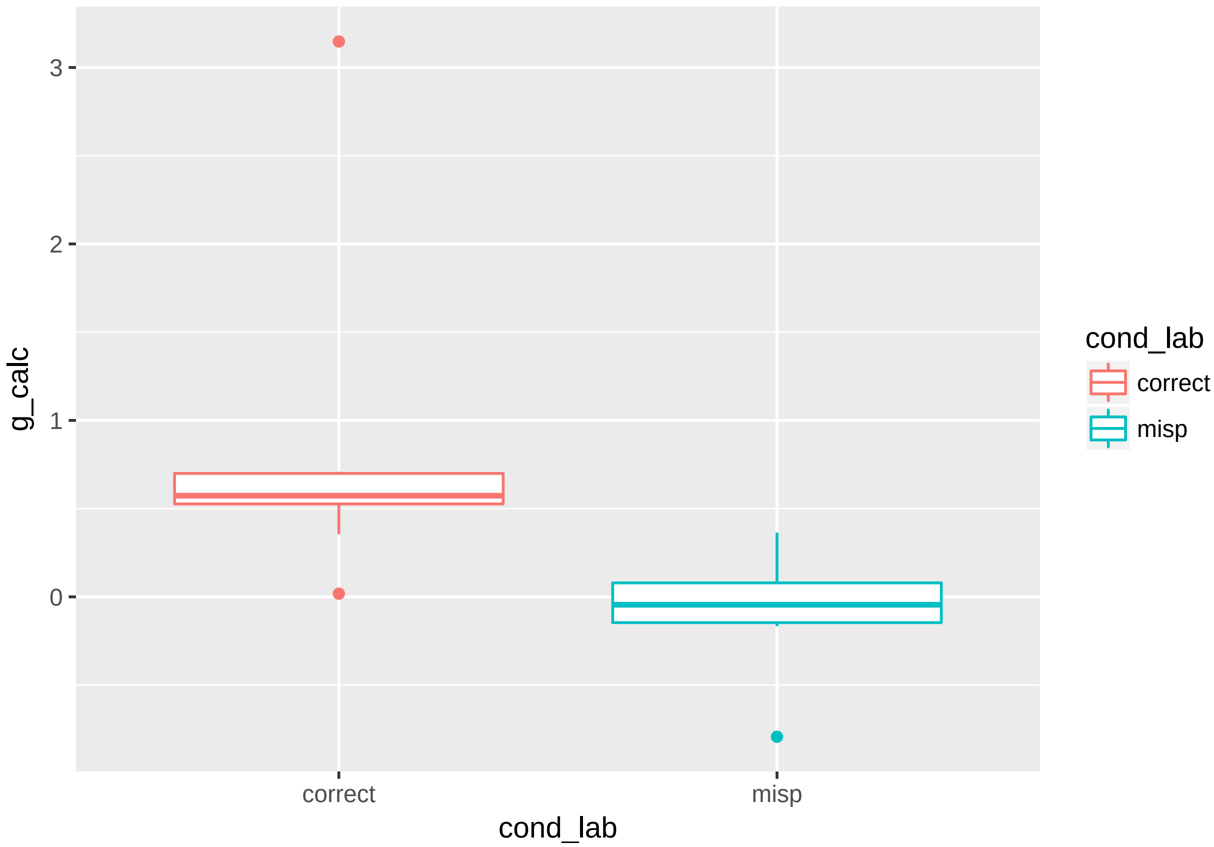
post_nam_dur	count
1.510	2
2.000	36
2.500	17
2.600	3
2.750	4
2.767	1
2.805	4
3.000	12
3.500	6
4.000	6
6.000	1

In summary, we see little consistency in analysis methods of comparable studies looking at naming and mispronunciation effects.

Mispronunciation Sensitivity in the youngest ages

Even the youngest ages in the database (less than 1 year) show mispronunciation sensitivity

```
## [1] Zesiger et al. (2012)       Mani & Plunkett 2010
## [3] Bergelson & Swingley (2017)
## 34 Levels: Altvater-Mackensen (2010) ... Zesiger et al. (2012)
```



Meta-Analysis

Condition: Mispronunciation Sensitivity Effects

Correct object identification effect

```
rma_correct = rma.mv(g_calc, g_var_calc, data = db_ET_correct, random = ~collapse |
    short_cite)

summary(rma_correct)
```

```
##
## Multivariate Meta-Analysis Model (k = 92; method: REML)
##
##   logLik  Deviance       AIC       BIC      AICc
## -85.3821  170.7643  176.7643  184.2969  177.0401
##
## Variance Components:
##
## outer factor: short_cite (nlvls = 32)
## inner factor: collapse   (nlvls = 46)
##
##             estim    sqrt  fixed
## tau^2      0.4568  0.6759     no
```

```
## rho        0.8830             no
##
## Test for Heterogeneity:
## Q(df = 91) = 511.9360, p-val < .0001
##
## Model Results:
##
## estimate       se     zval     pval    ci.lb    ci.ub
##   0.9230   0.1219   7.5699   <.0001   0.6840   1.1619      ***
##
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
# kable(round(coef(summary(rma_correct)), 2))

# aov.type <- anova(rma_correct)

sum_eff <- round(coef(summary(rma_correct))[1, ], 2)
```

Correct pronunciations were significantly greater than 0:

Hedges' g for `rownames(sum_eff)` was `toString(sum_eff$estimate)` (SE = `toString(sum_eff$se)` (95% CI [`toString(sum_eff$ci.lb)`, `toString(sum_eff$ci.ub)`], p = `toString(sum_eff$pval)`)

Mispronunciation object identification effect

```r
rma_MP = rma.mv(g_calc, g_var_calc, data = db_ET_MP, random = ~collapse | short_cite)

summary(rma_MP)
```

```
##
## Multivariate Meta-Analysis Model (k = 133; method: REML)
##
##   logLik  Deviance       AIC       BIC      AICc
## -54.0073  108.0145  114.0145  122.6629  114.2020
##
## Variance Components:
##
## outer factor: short_cite (nlvls = 32)
## inner factor: collapse   (nlvls = 46)
##
##             estim    sqrt  fixed
## tau^2      0.1164  0.3411     no
## rho        0.7916             no
##
## Test for Heterogeneity:
## Q(df = 132) = 418.7125, p-val < .0001
##
## Model Results:
##
## estimate       se     zval     pval    ci.lb    ci.ub
##   0.2367   0.0634   3.7347   0.0002   0.1125   0.3609      ***
##
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
sum_eff <- round(coef(summary(rma_MP))[1, ], 2)
```

Mispronunciations were significantly greater than 0:

Hedges' g for `rownames(sum_eff)` was `toString(sum_eff$estimate)` (SE = `toString(sum_eff$se)` (95% CI [`toString(sum_eff$ci.lb)`, `toString(sum_eff$ci.ub)`], p = `toString(sum_eff$pval)`)

Mispronunciation Sensitivity effect

```r
rma_MPeffect <- rma.mv(g_calc, g_var_calc, mods = ~condition, data = dat, random = ~collapse |
    short_cite)

summary(rma_MPeffect)
```

```
##
## Multivariate Meta-Analysis Model (k = 225; method: REML)
##
##    logLik   Deviance        AIC        BIC       AICc
## -195.1939   390.3879   398.3879   412.0166   398.5714
##
## Variance Components:
##
## outer factor: short_cite (nlvls = 32)
## inner factor: collapse   (nlvls = 46)
##
##             estim    sqrt  fixed
## tau^2      0.1342  0.3663     no
## rho        0.8145             no
##
## Test for Residual Heterogeneity:
## QE(df = 223) = 930.6485, p-val < .0001
##
## Test of Moderators (coefficient(s) 2):
## QM(df = 1) = 186.3809, p-val < .0001
##
## Model Results:
##
##            estimate      se     zval    pval   ci.lb   ci.ub
## intrcpt      0.2838  0.0668   4.2495  <.0001  0.1529  0.4146  ***
## condition    0.4766  0.0349  13.6521  <.0001  0.4082  0.5451  ***
##
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
# rma_MPeffect_1 <- rma.mv(g_calc, g_var_calc, mods = ~condition-1, data =
# dat, random = ~ collapse | short_cite)

# summary(rma_MPeffect_1)


aov.type <- anova(rma_MPeffect)

sum_eff <- round(coef(summary(rma_MPeffect))[2, ], 2)
```

The moderator test was significant: `print(aov.feat)`

There was a significant effect of condition:

Hedges' g for `rownames(sum_eff)` was `toString(n_feat$estimate)` (SE = `toString(n_feat$se)` (95% CI [`toString(n_feat$ci.lb)`, `toString(n_feat$ci.ub)`], p = `toString(n_feat$pval)`)

Age: Mispronunciation Sensitivity Effects with Age Moderators

Correct object identification effect with age moderator

```
rma_correct_age = rma.mv(g_calc, g_var_calc, mods = ~age.C, data = db_ET_correct,
    random = ~collapse | short_cite)

summary(rma_correct_age)
```

```
##
## Multivariate Meta-Analysis Model (k = 92; method: REML)
##
##   logLik  Deviance       AIC       BIC      AICc
## -84.2495  168.4990  176.4990  186.4982  176.9696
##
## Variance Components:
##
## outer factor: short_cite (nlvls = 32)
## inner factor: collapse   (nlvls = 46)
##
##             estim    sqrt  fixed
## tau^2      0.4521  0.6724     no
## rho        0.8755             no
##
## Test for Residual Heterogeneity:
## QE(df = 90) = 508.5859, p-val < .0001
##
## Test of Moderators (coefficient(s) 2):
## QM(df = 1) = 0.8459, p-val = 0.3577
##
## Model Results:
##
##          estimate      se    zval    pval    ci.lb   ci.ub
## intrcpt    0.9381  0.1223  7.6704  <.0001   0.6984  1.1778  ***
## age.C      0.0166  0.0180  0.9197  0.3577  -0.0187  0.0519
##
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# kable(round(coef(summary(rma_correct_age)), 2))

aov.type <- anova(rma_correct_age)

sum_eff <- round(coef(summary(rma_correct_age))[2, ], 2)
```

The moderator test was not significant: `print(aov.feat)`

There was no significant effect of age:

Hedges' g for `rownames(sum_eff)` was `toString(n_feat$estimate)` (SE = `toString(n_feat$se)` (95% CI [`toString(n_feat$ci.lb)`, `toString(n_feat$ci.ub)`], p = `toString(n_feat$pval)`)

Mispronunciation object identification effect with age moderator

```
rma_MP_age = rma.mv(g_calc, g_var_calc, mods = ~age.C, data = db_ET_MP, random = ~collapse |
    short_cite)

summary(rma_MP_age)
```

```
##
## Multivariate Meta-Analysis Model (k = 133; method: REML)
##
##   logLik  Deviance       AIC       BIC      AICc
## -52.9297  105.8594  113.8594  125.3602  114.1768
##
## Variance Components:
##
## outer factor: short_cite (nlvls = 32)
## inner factor: collapse   (nlvls = 46)
##
##             estim    sqrt  fixed
## tau^2      0.1160  0.3405     no
## rho        0.7837             no
##
## Test for Residual Heterogeneity:
## QE(df = 131) = 405.8802, p-val < .0001
##
## Test of Moderators (coefficient(s) 2):
## QM(df = 1) = 1.4975, p-val = 0.2211
##
## Model Results:
##
##          estimate      se    zval    pval    ci.lb   ci.ub
## intrcpt    0.2484  0.0639  3.8876  0.0001   0.1232  0.3736  ***
## age.C      0.0133  0.0108  1.2237  0.2211  -0.0080  0.0345
##
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
aov.type <- anova(rma_MP_age)

sum_eff <- round(coef(summary(rma_MP_age))[2, ], 2)
```

The moderator test was not significant: `print(aov.feat)`

There was no significant effect of age:

Hedges' g for `rownames(sum_eff)` was `toString(n_feat$estimate)` (SE = `toString(n_feat$se)` (95% CI [`toString(n_feat$ci.lb)`, `toString(n_feat$ci.ub)`], p = `toString(n_feat$pval)`)

Mispronunciation Sensitivity effect with age moderator

```
rma_MPeffect_age <- rma.mv(g_calc, g_var_calc, mods = ~age.C * condition, data = dat,
    random = ~collapse | short_cite)

summary(rma_MPeffect_age)
```

```
##
## Multivariate Meta-Analysis Model (k = 225; method: REML)
##
##    logLik   Deviance        AIC        BIC       AICc
## -193.5983   387.1965   399.1965   419.5855   399.5891
##
## Variance Components:
##
## outer factor: short_cite (nlvls = 32)
## inner factor: collapse   (nlvls = 46)
##
##             estim    sqrt  fixed
## tau^2      0.1303  0.3610     no
## rho        0.8044             no
##
## Test for Residual Heterogeneity:
## QE(df = 221) = 914.4660, p-val < .0001
##
## Test of Moderators (coefficient(s) 2,3,4):
## QM(df = 3) = 189.2888, p-val < .0001
##
## Model Results:
##
##                   estimate      se     zval    pval    ci.lb   ci.ub
## intrcpt             0.2998  0.0665   4.5083  <.0001   0.1694  0.4301  ***
## age.C               0.0183  0.0109   1.6893  0.0912  -0.0029  0.0396    .
## condition           0.4748  0.0358  13.2556  <.0001   0.4046  0.5450  ***
## age.C:condition    -0.0025  0.0078  -0.3217  0.7477  -0.0177  0.0127
##
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
aov.type <- anova(rma_MPeffect_age)

sum_eff <- round(coef(summary(rma_MPeffect_age))[4, ], 2)
```

The moderator test was significant: `print(aov.feat)`

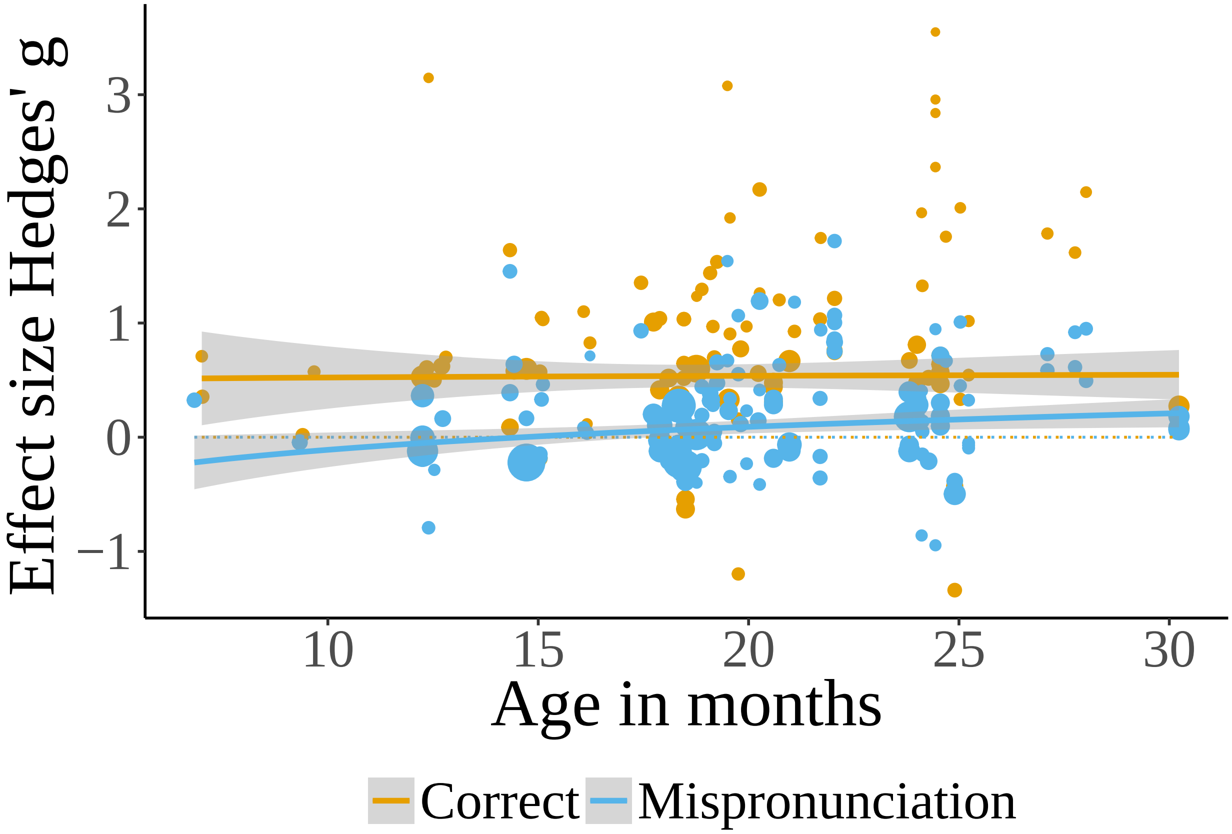
There was no significant interaction between condition and age:

Hedges' g for `rownames(sum_eff)` was `toString(n_feat$estimate)` (SE = `toString(n_feat$se)` (95% CI [`toString(n_feat$ci.lb)`, `toString(n_feat$ci.ub)`], p = `toString(n_feat$pval)`)

Plot Mispronunciation Effect by Age (color)

```
dat$condition_label = ifelse(dat$condition == 1, "Correct", "Mispronunciation")
```

```r
p <- ggplot(dat, aes(mean_age_1/30.44, g_calc, color = condition_label)) + geom_point(aes(size = weights
    show.legend = FALSE) + geom_line(y = 0, linetype = "dotted") + geom_smooth(method = "lm",
    formula = y ~ log(x), aes(weight = weights_g)) + scale_colour_manual(values = cbPalette) +
    apatheme + theme(legend.title = element_blank(), legend.position = "bottom") +
    xlab("Age in months") + ylab("Effect size Hedges' g")

p
```




```r
min(dat$mean_age_1/30.44)
```

```
## [1] 6.826544
```

```r
max(dat$mean_age_1/30.44)
```

```
## [1] 30.22996
```

```r
# ggsave('figures/AgeEffect_log.jpg', p,height= 7,width= 6)



jpeg(filename = "figures/AgeEffect_log.jpg", width = 500, height = 300, units = "px")

p

dev.off()
```

```
## pdf
##   2
```

Vocabulary size: Correlation between mispronunciation sensitivity and vocabulary

First, let's take a look at which vocabulary data we have available.

```
vocab_info <- db_ET_correct %>% mutate(has_vocab = ifelse(!is.na(r_comprehension),
    "comprehension", ifelse(!is.na(r_production), "production", "none"))) %>%
    group_by(has_vocab) %>% summarize(count = n())

kable(vocab_info)
```

has_vocab	count
comprehension	12
none	77
production	3

We have 17 correlations, roughly evenly divided between comprehension and production data. There is reason to believe that production data are different from comprehension data (the former being easier to estimate for parents in the typical questionnaire-based assessment), so we should both analyze this data separately and see whether it makes sense in a joint analysis.

```
# we're relying on the library meta function metacor
compr <- subset(db_ET_correct, !is.na(db_ET_correct$r_comprehension) & r_comprehension >
    -1)

metacor(cor = r_comprehension, n = n_1, studlab = short_cite, data = compr,
    sm = "COR")
```

```
##                                      COR            95%-CI %W(fixed)
## Zesiger et al. (2012)             0.0610 [-0.3553; 0.4773]       5.8
## Zesiger et al. (2012)            -0.1590 [-0.5663; 0.2483]       6.1
## Mani, Coleman, & Plunkett (2008)  0.0300 [-0.2271; 0.2871]      15.2
## Swingley & Aslin (2000)           0.1050 [-0.1564; 0.3664]      14.7
## Mani & Plunkett 2007             -0.1700 [-0.5234; 0.1834]       8.0
## Mani & Plunkett 2007             -0.1700 [-0.5175; 0.1775]       8.3
## Swingley & Aslin (2002)           0.1410 [-0.2432; 0.5252]       6.8
## Swingley & Aslin (2002)           0.1410 [-0.2596; 0.5416]       6.3
## Swingley 2003                     0.3400 [ 0.0470; 0.6330]      11.7
## Swingley 2003                     0.0600 [-0.3472; 0.4672]       6.1
## H\xbfjen et al.                   0.2220 [-0.2591; 0.7031]       4.3
## H\xbfjen et al.                   0.4820 [ 0.0935; 0.8705]       6.7
##                                  %W(random)
## Zesiger et al. (2012)                   6.2
## Zesiger et al. (2012)                   6.5
## Mani, Coleman, & Plunkett (2008)       13.7
## Swingley & Aslin (2000)                13.4
## Mani & Plunkett 2007                    8.3
## Mani & Plunkett 2007                    8.5
## Swingley & Aslin (2002)                 7.2
## Swingley & Aslin (2002)                 6.7
## Swingley 2003                          11.2
## Swingley 2003                           6.5
## H\xbfjen et al.                         4.8
## H\xbfjen et al.                         7.0
```

```
##
## Number of studies combined: k = 12
##
##                          COR            95%-CI    z p-value
## Fixed effect model   0.0897 [-0.0105; 0.1900] 1.75  0.0795
## Random effects model 0.0893 [-0.0212; 0.1999] 1.58  0.1132
##
## Quantifying heterogeneity:
## tau^2 = 0.0060; H = 1.09 [1.00; 1.50]; I^2 = 15.7% [0.0%; 55.4%]
##
## Test of heterogeneity:
##      Q d.f. p-value
##  13.05   11  0.2899
##
## Details on meta-analytical method:
## - Inverse variance method
## - DerSimonian-Laird estimator for tau^2
## - Untransformed correlations
```

```r
# we're relying on the library meta function metacor
prodr <- subset(db_ET_correct, !is.na(db_ET_correct$r_production) & r_production <
    1)

metacor(cor = r_production, n = n_1, studlab = short_cite, data = prodr, sm = "COR")
```

```
##                                       COR            95%-CI %W(fixed)
## Zesiger et al. (2012)           -0.0090 [-0.4268; 0.4088]       5.6
## Zesiger et al. (2012)           -0.1720 [-0.5775; 0.2335]       6.0
## Mani, Coleman, & Plunkett (2008)  0.0700 [-0.1861; 0.3261]      15.0
## Mani & Plunkett 2007            -0.1100 [-0.4696; 0.2496]       7.6
## Mani & Plunkett 2007            -0.1100 [-0.4635; 0.2435]       7.9
## Swingley & Aslin (2002)          0.1820 [-0.1970; 0.5610]       6.8
## Swingley & Aslin (2002)          0.1820 [-0.2131; 0.5771]       6.3
## Swingley 2003                    0.1800 [-0.1406; 0.5006]       9.6
## Swingley 2003                    0.0700 [-0.3367; 0.4767]       5.9
## Ramon-Casas et al. 2009          0.0980 [-0.3068; 0.5028]       6.0
## Ramon-Casas et al. 2009         -0.1470 [-0.5468; 0.2528]       6.1
## Ramon-Casas et al. 2009          0.4350 [ 0.1037; 0.7663]       8.9
## H\xbfjen et al.                  0.2220 [-0.2591; 0.7031]       4.2
## H\xbfjen et al.                 -0.1480 [-0.6430; 0.3470]       4.0
##                                 %W(random)
## Zesiger et al. (2012)                  5.6
## Zesiger et al. (2012)                  6.0
## Mani, Coleman, & Plunkett (2008)      15.0
## Mani & Plunkett 2007                   7.6
## Mani & Plunkett 2007                   7.9
## Swingley & Aslin (2002)                6.8
## Swingley & Aslin (2002)                6.3
## Swingley 2003                          9.6
## Swingley 2003                          5.9
## Ramon-Casas et al. 2009                6.0
## Ramon-Casas et al. 2009                6.1
## Ramon-Casas et al. 2009                8.9
## H\xbfjen et al.                        4.2
## H\xbfjen et al.                        4.0
```

```
##
## Number of studies combined: k = 14
##
##                          COR            95%-CI    z p-value
## Fixed effect model   0.0672 [-0.0319; 0.1663] 1.33  0.1838
## Random effects model 0.0672 [-0.0319; 0.1663] 1.33  0.1838
##
## Quantifying heterogeneity:
## tau^2 = 0; H = 1.00 [1.00; 1.40]; I^2 = 0.0% [0.0%; 49.2%]
##
## Test of heterogeneity:
##      Q d.f. p-value
##  11.50   13  0.5692
##
## Details on meta-analytical method:
## - Inverse variance method
## - DerSimonian-Laird estimator for tau^2
## - Untransformed correlations
```

Size of Mispronunciation: Measured in Features Changed

Number of features

Size of mispronunciation, measured in features changed

```
dat.f <- subset(dat, n_feature == "0" | n_feature == "1" | n_feature == "2" |
    n_feature == "3")

# rma_NFeatures <- rma.mv(g_calc, g_var_calc, mods = ~as.ordered(n_feature),
# data = db_ET_MP, random = ~collapse | short_cite)
rma_NFeatures <- rma.mv(g_calc, g_var_calc, mods = ~as.numeric(n_feature), data = dat.f,
    random = ~collapse | short_cite)


summary(rma_NFeatures)
```

```
##
## Multivariate Meta-Analysis Model (k = 191; method: REML)
##
##    logLik   Deviance        AIC        BIC       AICc
## -200.2579   400.5158   408.5158   421.4828   408.7332
##
## Variance Components:
##
## outer factor: short_cite (nlvls = 27)
## inner factor: collapse   (nlvls = 43)
##
##             estim    sqrt  fixed
## tau^2      0.1267  0.3560     no
## rho        0.7549             no
##
## Test for Residual Heterogeneity:
## QE(df = 189) = 882.7868, p-val < .0001
##
```

```
## Test of Moderators (coefficient(s) 2):
## QM(df = 1) = 118.3832, p-val < .0001
##
## Model Results:
##
##                          estimate      se       zval    pval    ci.lb
## intrcpt                    0.6788  0.0715     9.4989  <.0001   0.5388
## as.numeric(n_feature)     -0.2869  0.0264   -10.8804  <.0001  -0.3386
##                            ci.ub
## intrcpt                   0.8189  ***
## as.numeric(n_feature)    -0.2352  ***
##
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
aov.feat <- anova(rma_NFeatures)

n_feat <- round(coef(summary(rma_NFeatures))[2, ], 2)
```

The moderator test was significant: `print(aov.feat)`

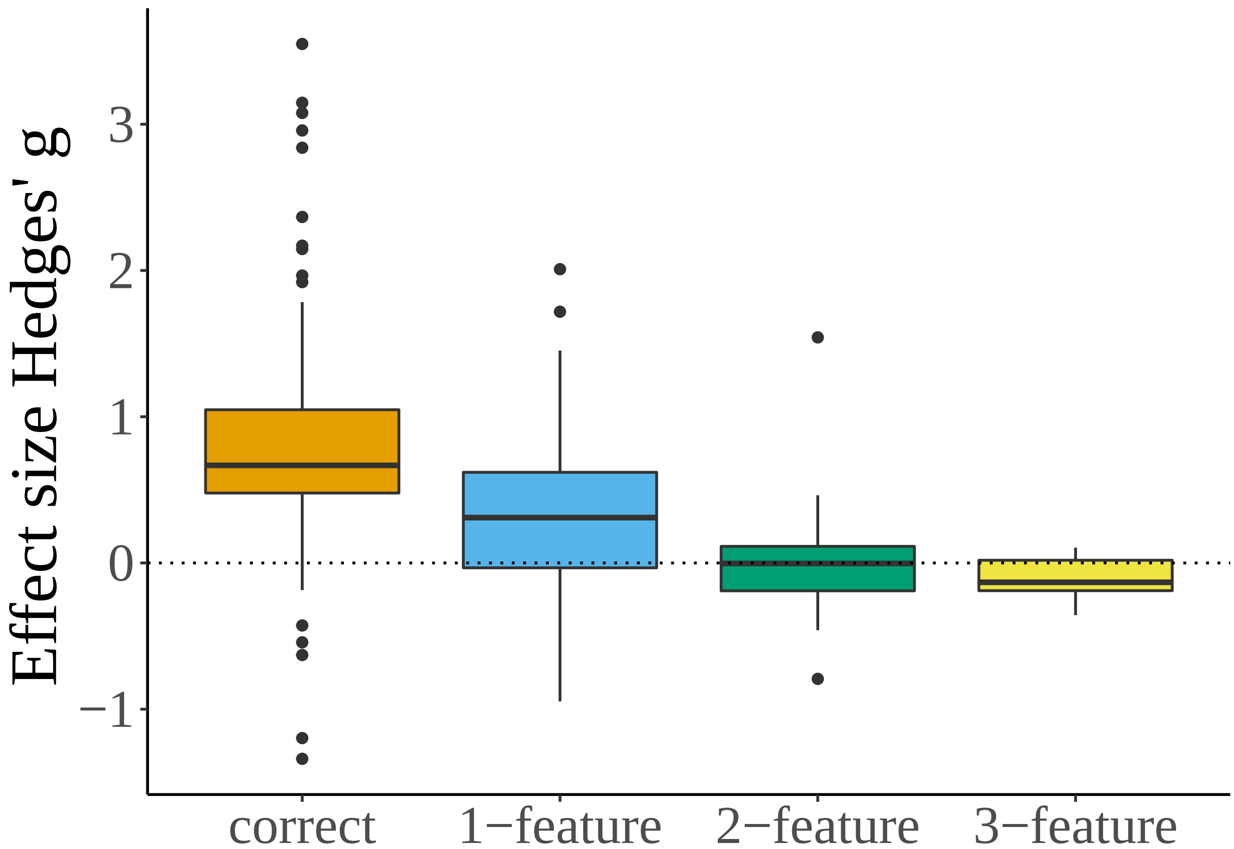
There was a significant effect of number of features changed:

Hedges' g for `rownames(n_feat)` was `toString(n_feat$estimate)` (SE = `toString(n_feat$se)` (95% CI [`toString(n_feat$ci.lb)`, `toString(n_feat$ci.ub)`], p = `toString(n_feat$pval)`)

Plot number of Features

```r
dat_f <- subset(dat, n_feature == "0" | n_feature == "1" | n_feature == "2" |
    n_feature == "3")

# mf <- subset(dat_f, n_feature == '3') min_age <- min(mf$mean_age_1)
# max_age <- max(mf$mean_age_1) dat_fage= dat_f%>%
# filter(mean_age_1>=min_age&mean_age_1<=max_age)

dat_f$feat_cat <- ifelse(dat_f$n_feature == 1, "1-feature", ifelse(dat_f$n_feature ==
    2, "2-feature", ifelse(dat_f$n_feature == 3, "3-feature", ifelse(dat_f$n_feature ==
    0, "correct", "none"))))

dat_f <- subset(dat_f, feat_cat != "none")

dat_f$Features_changed <- factor(dat_f$feat_cat, levels = c("correct", "1-feature",
    "2-feature", "3-feature"))

# Color Blind palette:
cbPalette <- c("#E69F00", "#56B4E9", "#009E73", "#F0E442", "#0072B2", "#D55E00",
    "#CC79A7")

p <- ggplot(dat_f, aes(Features_changed, g_calc, fill = Features_changed)) +
    geom_boxplot() + # geom_smooth(method = 'lm', formula = y ~ log(x), aes(weight=weights_g)) +
scale_fill_manual(values = cbPalette) + apatheme + theme(text = element_text(size = 25),
    legend.title = element_blank(), legend.position = "none", axis.title.x = element_blank()) +
    # xlab('Number of Features Changed') +
geom_hline(yintercept = 0, linetype = "dotted") + ylab("Effect size Hedges' g")
```

```
p
```




```
jpeg(filename = "figures/Number_of_Features.jpg", width = 500, height = 300,
    units = "px")

p

dev.off()
```

```
## pdf
##   2
```

Number of features with age moderator interaction

Size of mispronunciation, measured in features changed

```
dat.f <- subset(dat, n_feature == "0" | n_feature == "1" | n_feature == "2" |
    n_feature == "3")

# rma_NFeatures <- rma.mv(g_calc, g_var_calc, mods = ~as.ordered(n_feature),
# data = db_ET_MP, random = ~collapse | short_cite)
rma_NFeatures <- rma.mv(g_calc, g_var_calc, mods = ~as.numeric(n_feature) *
    age.C, data = dat.f, random = ~collapse | short_cite)


# summary(rma_NFeatures)
```

```r
aov.feat <- anova(rma_NFeatures)

n_feat <- round(coef(summary(rma_NFeatures))[4, ], 2)
```

The moderator test was significant: `print(aov.feat)`

But there was no significant interaction between number of features changed and age:

Hedges' g for `rownames(n_feat)` was `toString(n_feat$estimate)` (SE = `toString(n_feat$se)` (95% CI [`toString(n_feat$ci.lb)`, `toString(n_feat$ci.ub)`], p = `toString(n_feat$pval)`)

Number of features subset to age range

Size of mispronunciation, measured in features changed

```r
dat_f <- subset(dat, n_feature == "0" | n_feature == "1" | n_feature == "2" |
    n_feature == "3")

mf <- subset(dat_f, n_feature == "3")
min_age <- min(mf$mean_age_1)
max_age <- max(mf$mean_age_1)

dat_fage = dat_f %>% filter(mean_age_1 >= min_age & mean_age_1 <= max_age)

# dat_fage$n_feature <- ordered(dat_fage$n_feature, levels = c('0', '1',
# '2', '3')) dat_fage$n_feature <- as.numeric(dat_fage$n_feature)


# rma_NFeatures <- rma.mv(g_calc, g_var_calc, mods = ~as.ordered(n_feature),
# data = db_ET_MP, random = ~collapse | short_cite)
rma_NFeatures_agesub <- rma.mv(g_calc, g_var_calc, mods = ~as.numeric(n_feature),
    data = dat_f, random = ~collapse | short_cite)


# summary(rma_NFeatures_agesub)
```

Number of features with age moderator subset to age range

Size of mispronunciation, measured in features changed

No interaction between features and age

```r
dat_f <- subset(dat, n_feature == "0" | n_feature == "1" | n_feature == "2" |
    n_feature == "3")

mf <- subset(dat_f, n_feature == "3")
min_age <- min(mf$mean_age_1)
max_age <- max(mf$mean_age_1)

dat_fage = dat_f %>% filter(mean_age_1 >= min_age & mean_age_1 <= max_age)

# dat_fage$n_feature <- ordered(dat_fage$n_feature, levels = c('0', '1',
# '2', '3')) dat_fage$n_feature <- as.numeric(dat_fage$n_feature)
```

```r
# rma_NFeatures <- rma.mv(g_calc, g_var_calc, mods = ~as.ordered(n_feature),
# data = db_ET_MP, random = ~collapse | short_cite)
rma_NFeatures_agesub <- rma.mv(g_calc, g_var_calc, mods = ~as.numeric(n_feature) *
    age.C, data = dat_f, random = ~collapse | short_cite)


# summary(rma_NFeatures_agesub)
```

Distractor Familiarity (familiary, unfamiliar)

```r
rma_Distractor <- rma.mv(g_calc, g_var_calc, mods = ~as.factor(object_pair),
    data = db_ET_MP, random = ~collapse | short_cite)

# summary(rma_Distractor)

aov.type <- anova(rma_Distractor)

sum_eff <- round(coef(summary(rma_TypeFeatures_Lang))[2, ], 2)
```

```
## Error in summary(rma_TypeFeatures_Lang): object 'rma_TypeFeatures_Lang' not found
```

The moderator test was not significant: `print(aov.type)`

No significant interaction between feature type and language family:

Hedges' g for `rownames(sum_eff)` was `toString(sum_eff$estimate)` (SE = `toString(sum_eff$se)` (95% CI [`toString(sum_eff$ci.lb)`, `toString(sum_eff$ci.ub)`], p = `toString(sum_eff$pval)`)

Distractor Familiarity with condition moderator

```r
rma_Distractor <- rma.mv(g_calc, g_var_calc, mods = ~condition * as.factor(object_pair),
    data = dat, random = ~collapse | short_cite)

# summary(rma_Distractor)

aov.type <- anova(rma_Distractor)

sum_eff <- round(coef(summary(rma_Distractor))[4, ], 2)
```

The moderator test was significant: `print(aov.type)`

But there was no significant interaction between distractor familiarity and condition:

Hedges' g for `rownames(sum_eff)` was `toString(sum_eff$estimate)` (SE = `toString(sum_eff$se)` (95% CI [`toString(sum_eff$ci.lb)`, `toString(sum_eff$ci.ub)`], p = `toString(sum_eff$pval)`)

Distractor Familiarity with age moderator

```r
rma_DistractorAge <- rma.mv(g_calc, g_var_calc, mods = ~age.C * as.factor(object_pair),
    data = db_ET_MP, random = ~collapse | short_cite)

# summary(rma_DistractorAge)
```

```r
aov.type <- anova(rma_DistractorAge)

sum_eff <- round(coef(summary(rma_DistractorAge))[4, ], 2)
```

The moderator test was not significant: `print(aov.type)`

There was no significant interaction between distractor familiarity and age:

Hedges' g for `rownames(sum_eff)` was `toString(sum_eff$estimate)` (SE = `toString(sum_eff$se)` (95% CI [`toString(sum_eff$ci.lb)`, `toString(sum_eff$ci.ub)`], p = `toString(sum_eff$pval)`)

Distractor Familiarity with age and condition moderators

```r
rma_DistractorAge <- rma.mv(g_calc, g_var_calc, mods = ~age.C * condition *
    as.factor(object_pair), data = dat, random = ~collapse | short_cite)

summary(rma_DistractorAge)
```

```
##
## Multivariate Meta-Analysis Model (k = 225; method: REML)
##
##    logLik   Deviance        AIC        BIC       AICc
## -189.1874   378.3747   398.3747   432.1737   399.4427
##
## Variance Components:
##
## outer factor: short_cite (nlvls = 32)
## inner factor: collapse   (nlvls = 46)
##
##             estim    sqrt  fixed
## tau^2      0.1335  0.3654     no
## rho        0.7904             no
##
## Test for Residual Heterogeneity:
## QE(df = 217) = 909.1038, p-val < .0001
##
## Test of Moderators (coefficient(s) 2,3,4,5,6,7,8):
## QM(df = 7) = 196.9737, p-val < .0001
##
## Model Results:
##
##                                                         estimate      se
## intrcpt                                                   0.3844  0.0804
## age.C                                                     0.0267  0.0133
## condition                                                 0.4258  0.0439
## as.factor(object_pair)familiar_novel                     -0.2666  0.1464
## age.C:condition                                          -0.0066  0.0095
## age.C:as.factor(object_pair)familiar_novel                0.0005  0.0276
## condition:as.factor(object_pair)familiar_novel            0.2123  0.0912
## age.C:condition:as.factor(object_pair)familiar_novel     -0.0119  0.0200
##                                                            zval    pval
## intrcpt                                                  4.7833  <.0001
## age.C                                                    1.9999  0.0455
```

```
## condition                                           9.6983  <.0001
## as.factor(object_pair)familiar_novel               -1.8213  0.0686
## age.C:condition                                    -0.6953  0.4869
## age.C:as.factor(object_pair)familiar_novel          0.0188  0.9850
## condition:as.factor(object_pair)familiar_novel      2.3292  0.0199
## age.C:condition:as.factor(object_pair)familiar_novel  -0.5976  0.5501
##                                                      ci.lb   ci.ub
## intrcpt                                             0.2269  0.5419  ***
## age.C                                               0.0005  0.0528    *
## condition                                           0.3398  0.5119  ***
## as.factor(object_pair)familiar_novel               -0.5534  0.0203    .
## age.C:condition                                    -0.0252  0.0120
## age.C:as.factor(object_pair)familiar_novel         -0.0535  0.0546
## condition:as.factor(object_pair)familiar_novel      0.0337  0.3910    *
## age.C:condition:as.factor(object_pair)familiar_novel  -0.0510  0.0272
##
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
aov.type <- anova(rma_DistractorAge)

sum_eff <- round(coef(summary(rma_DistractorAge))[7, ], 2)
```

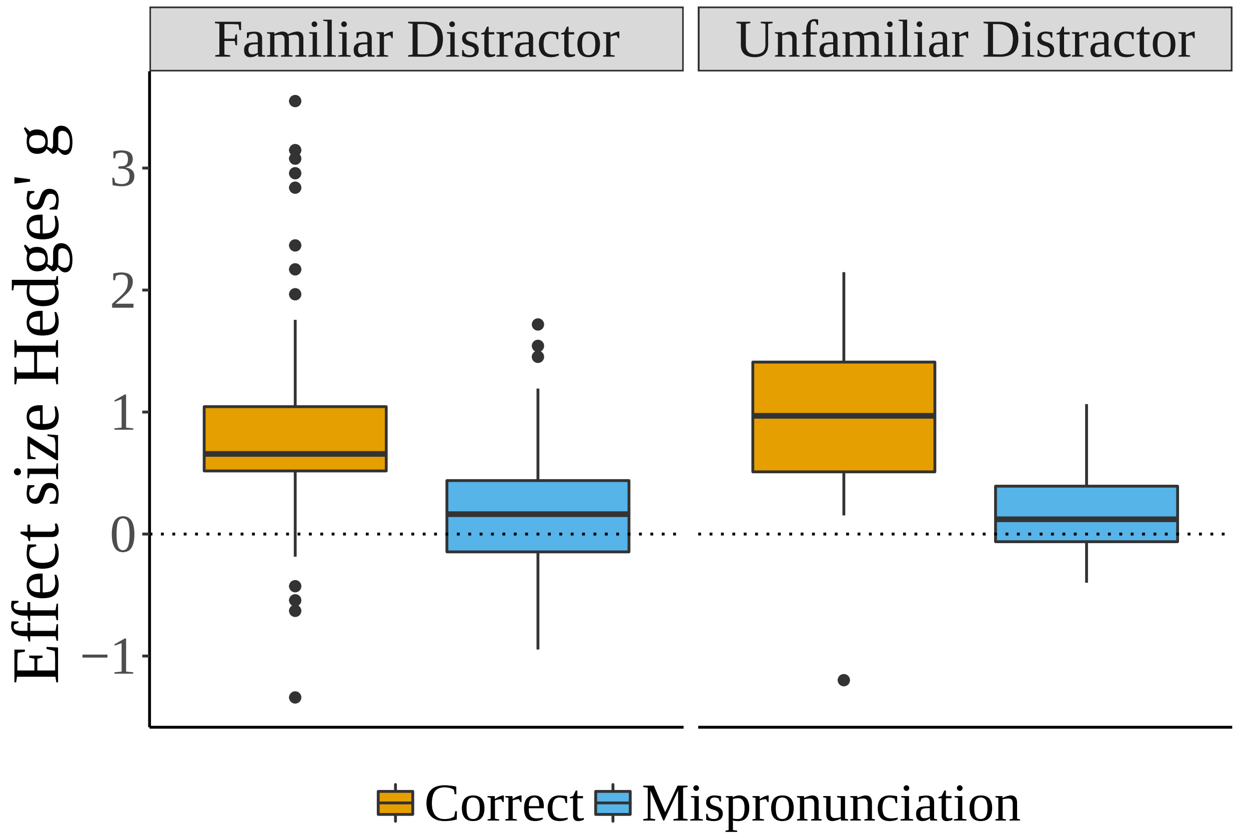
The moderator test was significant: `print(aov.type)`

There was a significant interaction between distractor familiarity and condition, but not age:

Hedges' g for `rownames(sum_eff)` was `toString(sum_eff$estimate)` (SE = `toString(sum_eff$se)` (95% CI [`toString(sum_eff$ci.lb)`, `toString(sum_eff$ci.ub)`], p = `toString(sum_eff$pval)`)

Plot Distractor familiarity and condition

```
dat$condition_label = ifelse(dat$condition == 1, "Correct", "Mispronunciation")
dat$dist_code <- ifelse(dat$object_pair == "familiar_familiar", "Familiar Distractor",
    "Unfamiliar Distractor")

# Color Blind palette:
cbPalette <- c("#E69F00", "#56B4E9", "#009E73", "#F0E442", "#0072B2", "#D55E00",
    "#CC79A7")

p <- ggplot(dat, aes(condition_label, g_calc, fill = condition_label)) + geom_boxplot() +
    facet_grid(. ~ dist_code) + # geom_smooth(method = 'lm', formula = y ~ log(x), aes(weight=weights_g
scale_fill_manual(values = cbPalette) + apatheme + theme(text = element_text(size = 25),
    legend.title = element_blank(), legend.position = "bottom", axis.title.x = element_blank(),
    axis.text.x = element_blank(), axis.ticks.x = element_blank()) + # xlab('Number of Features Changed
geom_hline(yintercept = 0, linetype = "dotted") + ylab("Effect size Hedges' g")

p
```




```
jpeg(filename = "figures/Distractor_fam.jpg", width = 500, height = 300, units = "px")

p

dev.off()
```

```
## pdf
##   2
```

Distractor Familiarity, subset to same age range

```
fn <- subset(dat, object_pair == "familiar_novel")
min_fn <- min(mf$mean_age_1)
max_fn <- max(mf$mean_age_1)

ff <- subset(dat, object_pair == "familiar_familiar")
min_ff <- min(ff$mean_age_1)
max_ff <- max(ff$mean_age_1)

min_age <- pmax(min_fn, min_ff)
max_age <- pmin(max_fn, max_ff)


dat_age = db_ET_MP %>% filter(mean_age_1 >= min_age & mean_age_1 <= max_age)

rma_Distractor <- rma.mv(g_calc, g_var_calc, mods = ~as.factor(object_pair),
```

```r
    data = dat_age, random = ~collapse | short_cite)

# summary(rma_Distractor)

aov.type <- anova(rma_Distractor)

sum_eff <- round(coef(summary(rma_Distractor))[2, ], 2)
```

The moderator test was not significant: `print(aov.type)`

There was no significant effect of distractor familiarity:

Hedges' g for `rownames(sum_eff)` was `toString(sum_eff$estimate)` (SE = `toString(sum_eff$se)` (95% CI [`toString(sum_eff$ci.lb)`, `toString(sum_eff$ci.ub)`], p = `toString(sum_eff$pval)`)

Distractor Familiarity with condition moderator, subset to same age range

```r
fn <- subset(dat, object_pair == "familiar_novel")
min_fn <- min(mf$mean_age_1)
max_fn <- max(mf$mean_age_1)

ff <- subset(dat, object_pair == "familiar_familiar")
min_ff <- min(ff$mean_age_1)
max_ff <- max(ff$mean_age_1)

min_age <- pmax(min_fn, min_ff)
max_age <- pmin(max_fn, max_ff)


dat_age = dat %>% filter(mean_age_1 >= min_age & mean_age_1 <= max_age)

rma_DistractorAgeS <- rma.mv(g_calc, g_var_calc, mods = ~condition * as.factor(object_pair),
    data = dat_age, random = ~collapse | short_cite)

summary(rma_DistractorAgeS)
```

```
##
## Multivariate Meta-Analysis Model (k = 161; method: REML)
##
##    logLik   Deviance        AIC        BIC       AICc
## -139.4193   278.8386   290.8386   309.1761   291.3986
##
## Variance Components:
##
## outer factor: short_cite (nlvls = 23)
## inner factor: collapse   (nlvls = 30)
##
##             estim    sqrt  fixed
## tau^2      0.1751  0.4184     no
## rho        0.8885             no
##
## Test for Residual Heterogeneity:
## QE(df = 157) = 709.9778, p-val < .0001
##
```

```
## Test of Moderators (coefficient(s) 2,3,4):
## QM(df = 3) = 127.9529, p-val < .0001
##
## Model Results:
##
##                                                 estimate      se     zval
## intrcpt                                           0.4043  0.1026   3.9413
## condition                                         0.4037  0.0483   8.3528
## as.factor(object_pair)familiar_novel             -0.2813  0.1522  -1.8484
## condition:as.factor(object_pair)familiar_novel    0.2034  0.0943   2.1569
##                                                   pval    ci.lb   ci.ub
## intrcpt                                         <.0001   0.2033  0.6054
## condition                                       <.0001   0.3090  0.4985
## as.factor(object_pair)familiar_novel            0.0645  -0.5795  0.0170
## condition:as.factor(object_pair)familiar_novel  0.0310   0.0186  0.3883
##
## intrcpt                                         ***
## condition                                       ***
## as.factor(object_pair)familiar_novel              .
## condition:as.factor(object_pair)familiar_novel    *
##
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
aov.type <- anova(rma_Distractor)

sum_eff <- round(coef(summary(rma_Distractor))[4, ], 2)
```

The moderator test was significant: `print(aov.type)`

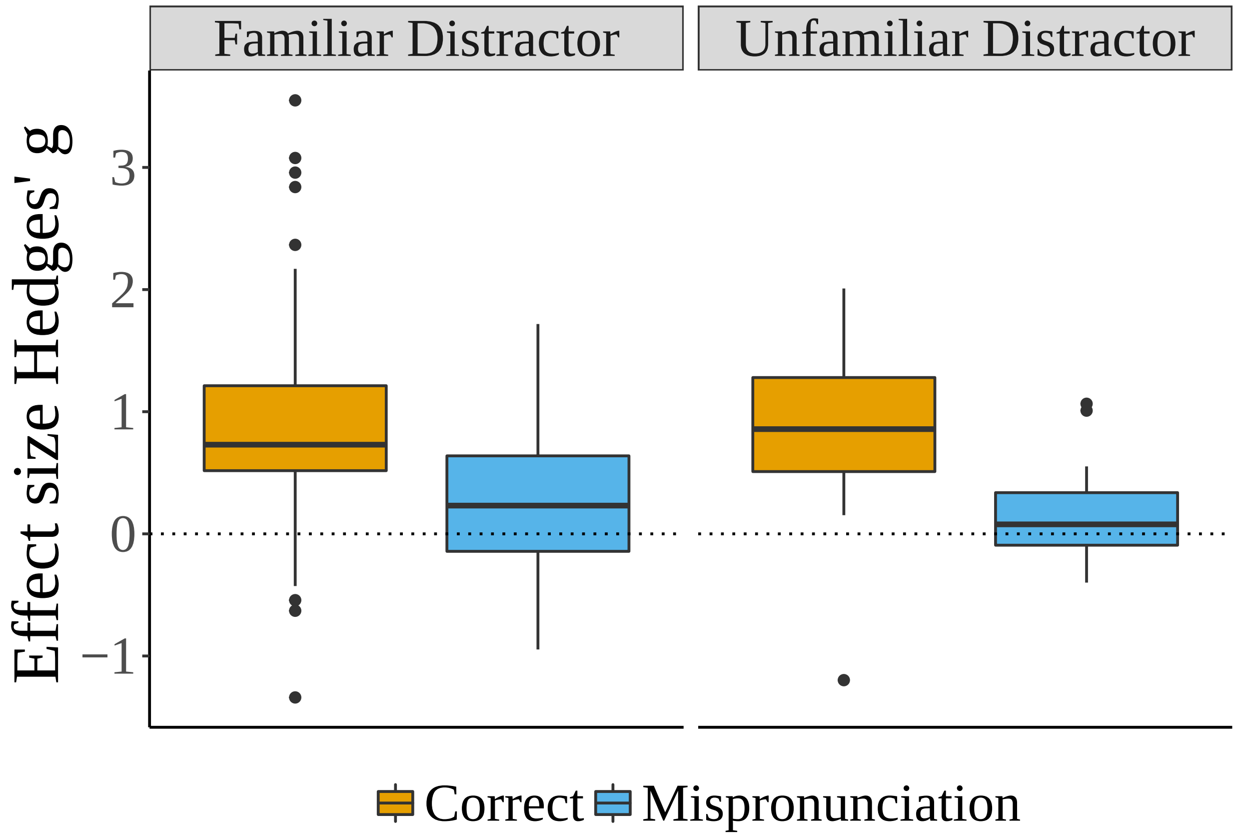
There was a significant interaction between distractor familiarity and condition:

Hedges' g for `rownames(sum_eff)` was `toString(sum_eff$estimate)` (SE = `toString(sum_eff$se)` (95% CI [`toString(sum_eff$ci.lb)`, `toString(sum_eff$ci.ub)`], p = `toString(sum_eff$pval)`)

Plot Distractor Familiarity with condition, subset to same age range

```
dat$condition_label = ifelse(dat$condition == 1, "Correct", "Mispronunciation")
dat$dist_code <- ifelse(dat$object_pair == "familiar_familiar", "Familiar Distractor",
    "Unfamiliar Distractor")

mf <- subset(dat, object_pair == "familiar_novel")
min_age <- min(mf$mean_age_1)

mf <- subset(dat, object_pair == "familiar_familiar")
max_age <- max(mf$mean_age_1)

dat_age = dat %>% filter(mean_age_1 >= min_age & mean_age_1 <= max_age)


p <- ggplot(dat_age, aes(condition_label, g_calc, fill = condition_label)) +
    geom_boxplot() + facet_grid(. ~ dist_code) + # geom_smooth(method = 'lm', formula = y ~ log(x), aes
scale_fill_manual(values = cbPalette) + apatheme + theme(text = element_text(size = 25),
    legend.title = element_blank(), legend.position = "bottom", axis.title.x = element_blank(),
```

```
    axis.text.x = element_blank(), axis.ticks.x = element_blank()) + # xlab('Number of Features Changed
geom_hline(yintercept = 0, linetype = "dotted") + ylab("Effect size Hedges' g")

p
```




```
jpeg(filename = "figures/Distractor_fam_age_AgeSubset.jpg", width = 500, height = 300,
    units = "px")

p

dev.off()
```

```
## pdf
##   2
```

Distractor Familiarity with age and condition moderator, subset to same age range

```
mf <- subset(dat, object_pair == "familiar_novel")
min_age <- min(mf$mean_age_1)

mf <- subset(dat, object_pair == "familiar_familiar")
max_age <- max(mf$mean_age_1)

dat_age = dat %>% filter(mean_age_1 >= min_age & mean_age_1 <= max_age)

rma_DistractorAgeS <- rma.mv(g_calc, g_var_calc, mods = ~age.C * condition *
```

```r
    as.factor(object_pair), data = dat_age, random = ~collapse | short_cite)

# summary(rma_DistractorAgeS)

aov.type <- anova(rma_Distractor)

sum_eff <- round(coef(summary(rma_Distractor))[8, ], 2)
```

The moderator test was significant: `print(aov.type)`

There was no significant interaction between distractor familiarity, condition, and age:

Hedges' g for `rownames(sum_eff)` was `toString(sum_eff$estimate)` (SE = `toString(sum_eff$se)` (95% CI [`toString(sum_eff$ci.lb)`, `toString(sum_eff$ci.ub)`], p = `toString(sum_eff$pval)`)

Position of Mispronunciation (onset, medial)

```r
# table(db_ET_MP$mispron_location)

db_ET_MP1 = db_ET_MP %>% filter(mispron_location == "onset" | mispron_location ==
    "medial")

# rma_NFeatures <- rma.mv(g_calc, g_var_calc, mods = ~as.ordered(n_feature),
# data = db_ET_MP, random = ~collapse | short_cite)
rma_Location <- rma.mv(g_calc, g_var_calc, mods = ~mispron_location, data = db_ET_MP1,
    random = ~collapse | short_cite)


# summary(rma_Location)

aov.type <- anova(rma_Location)

sum_eff <- round(coef(summary(rma_Location))[2, ], 2)
```

The moderator test was not significant: `print(aov.type)`

There was no significant effect of mispronunciation position:

Hedges' g for `rownames(sum_eff)` was `toString(sum_eff$estimate)` (SE = `toString(sum_eff$se)` (95% CI [`toString(sum_eff$ci.lb)`, `toString(sum_eff$ci.ub)`], p = `toString(sum_eff$pval)`)

Position of Mispronunciation with age moderator

```r
# table(db_ET_MP$mispron_location)

db_ET_MP1 = db_ET_MP %>% filter(mispron_location == "onset" | mispron_location ==
    "medial")


rma_LocationAge <- rma.mv(g_calc, g_var_calc, mods = ~mispron_location * age.C,
    data = db_ET_MP1, random = ~collapse | short_cite)
```

```r
# summary(rma_LocationAge)

aov.type <- anova(rma_LocationAge)

sum_eff <- round(coef(summary(rma_LocationAge))[2, ], 2)
```

The moderator test was not significant: `print(aov.type)`

There was no significant interaction between mispronunciation position and condition:

Hedges' g for `rownames(sum_eff)` was `toString(sum_eff$estimate)` (SE = `toString(sum_eff$se)` (95% CI [`toString(sum_eff$ci.lb)`, `toString(sum_eff$ci.ub)`], p = `toString(sum_eff$pval)`)

Position of Mispronunciation with condition moderator

```r
# table(db_ET_MP$mispron_location)

db_ET_MPl = dat %>% filter(mispron_location == "onset" | mispron_location ==
    "medial")


rma_LocationCondition <- rma.mv(g_calc, g_var_calc, mods = ~mispron_location *
    condition, data = db_ET_MPl, random = ~collapse | short_cite)

# summary(rma_LocationCondition)

aov.type <- anova(rma_LocationCondition)

sum_eff <- round(coef(summary(rma_LocationCondition))[4, ], 2)
```

The moderator test was significant: `print(aov.type)`

But there was no significant interaction between mispronunciation position and condition:

Hedges' g for `rownames(sum_eff)` was `toString(sum_eff$estimate)` (SE = `toString(sum_eff$se)` (95% CI [`toString(sum_eff$ci.lb)`, `toString(sum_eff$ci.ub)`], p = `toString(sum_eff$pval)`)

Position of Mispronunciation with age and condition moderators

```r
# table(db_ET_MP$mispron_location)

db_ET_MPl = dat %>% filter(mispron_location == "onset" | mispron_location ==
    "medial")


rma_LocationCondition <- rma.mv(g_calc, g_var_calc, mods = ~mispron_location *
    condition * age.C, data = db_ET_MPl, random = ~collapse | short_cite)


summary(rma_LocationCondition)
```

```
##
## Multivariate Meta-Analysis Model (k = 169; method: REML)
##
```

```
##    logLik   Deviance        AIC        BIC       AICc
## -140.7723   281.5447   301.5447   332.3587   303.0113
##
## Variance Components:
##
## outer factor: short_cite (nlvls = 24)
## inner factor: collapse   (nlvls = 34)
##
##             estim    sqrt  fixed
## tau^2      0.1648  0.4059     no
## rho        0.7256             no
##
## Test for Residual Heterogeneity:
## QE(df = 161) = 730.9788, p-val < .0001
##
## Test of Moderators (coefficient(s) 2,3,4,5,6,7,8):
## QM(df = 7) = 166.9571, p-val < .0001
##
## Model Results:
##
##                                          estimate      se     zval    pval
## intrcpt                                    0.3063  0.0929   3.2989  0.0010
## mispron_locationmedial                    -0.0025  0.1729  -0.0146  0.9883
## condition                                  0.4736  0.0439  10.7913  <.0001
## age.C                                      0.0198  0.0162   1.2201  0.2224
## mispron_locationmedial:condition           0.0834  0.1100   0.7582  0.4483
## mispron_locationmedial:age.C               0.0218  0.0303   0.7182  0.4727
## condition:age.C                           -0.0267  0.0112  -2.3921  0.0168
## mispron_locationmedial:condition:age.C     0.0614  0.0251   2.4463  0.0144
##                                           ci.lb   ci.ub
## intrcpt                                  0.1243   0.4883  ***
## mispron_locationmedial                  -0.3414   0.3363
## condition                                0.3876   0.5596  ***
## age.C                                   -0.0120   0.0516
## mispron_locationmedial:condition        -0.1322   0.2991
## mispron_locationmedial:age.C            -0.0377   0.0812
## condition:age.C                         -0.0485  -0.0048    *
## mispron_locationmedial:condition:age.C   0.0122   0.1107    *
##
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
aov.type <- anova(rma_LocationCondition)

sum_eff <- round(coef(summary(rma_LocationCondition))[8, ], 2)
```

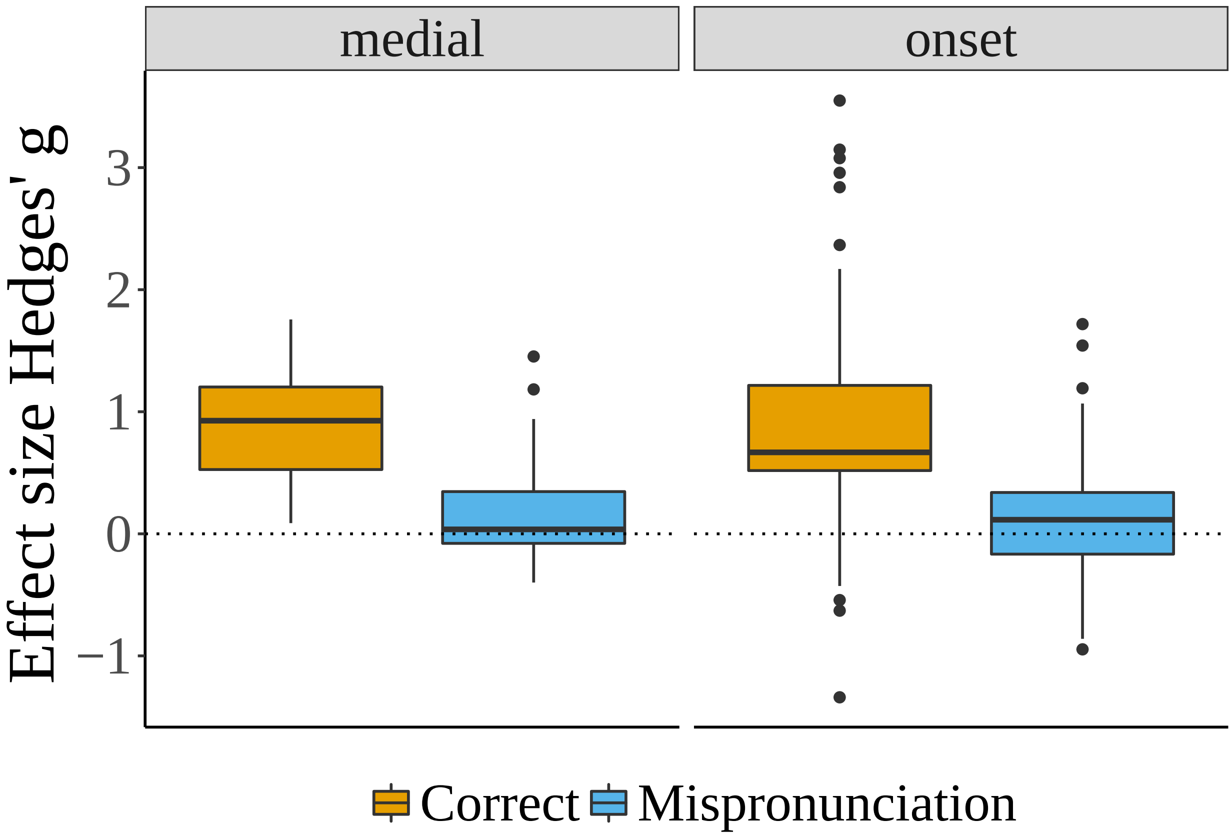
The moderator test was significant: `print(aov.type)`

But there was no significant interaction between mispronunciation position, condition, and age:

Hedges' g for `rownames(sum_eff)` was `toString(sum_eff$estimate)` (SE = `toString(sum_eff$se)` (95% CI [`toString(sum_eff$ci.lb)`, `toString(sum_eff$ci.ub)`], p = `toString(sum_eff$pval)`)

Plotting Position of Mispronunciation

```
# dat.p <- subset(dat, mispron_location == 'onset' | mispron_location ==
# 'medial' | mispron_location == 'offset')

dat.p <- subset(dat, mispron_location == "onset" | mispron_location == "medial")


p <- ggplot(dat.p, aes(condition_label, g_calc, fill = condition_label)) + facet_grid(. ~
    mispron_location) + geom_boxplot() + # geom_smooth(method = 'lm', formula = y ~ log(x), aes(weight=
scale_fill_manual(values = cbPalette) + apatheme + theme(text = element_text(size = 25),
    legend.title = element_blank(), legend.position = "bottom", axis.title.x = element_blank(),
    axis.text.x = element_blank(), axis.ticks.x = element_blank()) + # xlab('Number of Features Changed
geom_hline(yintercept = 0, linetype = "dotted") + ylab("Effect size Hedges' g")

p
```




```
jpeg(filename = "figures/Mispronunciation_position.jpg", width = 500, height = 300,
    units = "px")

p

dev.off()
```

```
## pdf
##   2
```

Distractor Overlap

```
db_ET_MPo = db_ET_MP %>% filter(distractor_overlap == "onset" | distractor_overlap ==
    "novel" | distractor_overlap == "no")


rma_DistractorOverlap <- rma.mv(g_calc, g_var_calc, mods = ~distractor_overlap,
    data = db_ET_MPo, random = ~collapse | short_cite)

# summary(rma_DistractorOverlap)

aov.type <- anova(rma_DistractorOverlap)

sum_eff1 <- round(coef(summary(rma_DistractorOverlap))[2, ], 2)
sum_eff2 <- round(coef(summary(rma_DistractorOverlap))[3, ], 2)
```

The moderator test was not significant: `print(aov.type)`

There was no significant effect of distractor overlap:

Hedges' g for `row.names(sum_eff1)` was `toString(sum_eff1$estimate)` (SE = `toString(sum_eff1$se)` (95% CI [`toString(sum_eff1$ci.lb)`, `toString(sum_eff1$ci.ub)`], p = `toString(sum_eff1$pval)`)

Hedges' g `row.names(sum_eff2)` was `toString(sum_eff2$estimate)` (SE = `toString(sum_eff2$se)` (95% CI [`toString(sum_eff2$ci.lb)`, `toString(sum_eff2$ci.ub)`], p = `toString(sum_eff2$pval)`)

Distractor Overlap with age moderator

```
db_ET_MPo = db_ET_MP %>% filter(distractor_overlap == "onset" | distractor_overlap ==
    "novel" | distractor_overlap == "no")

rma_DistractorOverlap <- rma.mv(g_calc, g_var_calc, mods = ~age.C * distractor_overlap,
    data = db_ET_MPo, random = ~collapse | short_cite)

# summary(rma_DistractorOverlap)
aov.type <- anova(rma_DistractorOverlap)

sum_eff1 <- round(coef(summary(rma_DistractorOverlap))[5, ], 2)
sum_eff2 <- round(coef(summary(rma_DistractorOverlap))[6, ], 2)
```

The moderator test was not significant: `print(aov.type)`

There was no significant interaction between distractor overlap and age:

Hedges' g for `row.names(sum_eff1)` was `toString(sum_eff1$estimate)` (SE = `toString(sum_eff1$se)` (95% CI [`toString(sum_eff1$ci.lb)`, `toString(sum_eff1$ci.ub)`], p = `toString(sum_eff1$pval)`)

Hedges' g `row.names(sum_eff2)` was `toString(sum_eff2$estimate)` (SE = `toString(sum_eff2$se)` (95% CI [`toString(sum_eff2$ci.lb)`, `toString(sum_eff2$ci.ub)`], p = `toString(sum_eff2$pval)`)

Distractor Overlap with condition moderator

```
db_ET_MPo = dat %>% filter(distractor_overlap == "onset" | distractor_overlap ==
    "novel" | distractor_overlap == "no")
```

```r
rma_DistractorOverlap <- rma.mv(g_calc, g_var_calc, mods = ~condition * distractor_overlap,
    data = db_ET_MPo, random = ~collapse | short_cite)

summary(rma_DistractorOverlap)
```

```
##
## Multivariate Meta-Analysis Model (k = 218; method: REML)
##
##    logLik   Deviance        AIC        BIC       AICc
## -182.8895   365.7791   381.7791   408.6318   382.4884
##
## Variance Components:
##
## outer factor: short_cite (nlvls = 31)
## inner factor: collapse   (nlvls = 46)
##
##             estim    sqrt  fixed
## tau^2      0.1476  0.3842     no
## rho        0.8092             no
##
## Test for Residual Heterogeneity:
## QE(df = 212) = 906.9597, p-val < .0001
##
## Test of Moderators (coefficient(s) 2,3,4,5,6):
## QM(df = 5) = 201.3369, p-val < .0001
##
## Model Results:
##
##                                    estimate      se     zval    pval
## intrcpt                              0.2557  0.1102   2.3206  0.0203
## condition                            0.5052  0.0503  10.0418  <.0001
## distractor_overlapno                 0.1982  0.1570   1.2620  0.2070
## distractor_overlapnovel             -0.1252  0.1531  -0.8178  0.4135
## condition:distractor_overlapno      -0.2359  0.0810  -2.9125  0.0036
## condition:distractor_overlapnovel    0.2286  0.0960   2.3802  0.0173
##                                      ci.lb    ci.ub
## intrcpt                             0.0397   0.4717    *
## condition                           0.4066   0.6039  ***
## distractor_overlapno               -0.1096   0.5060
## distractor_overlapnovel            -0.4252   0.1748
## condition:distractor_overlapno     -0.3947  -0.0772   **
## condition:distractor_overlapnovel   0.0404   0.4169    *
##
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
aov.type <- anova(rma_DistractorOverlap)

sum_eff1 <- round(coef(summary(rma_DistractorOverlap))[5, ], 2)
sum_eff2 <- round(coef(summary(rma_DistractorOverlap))[6, ], 2)
```

The moderator test was significant: `print(aov.type)`

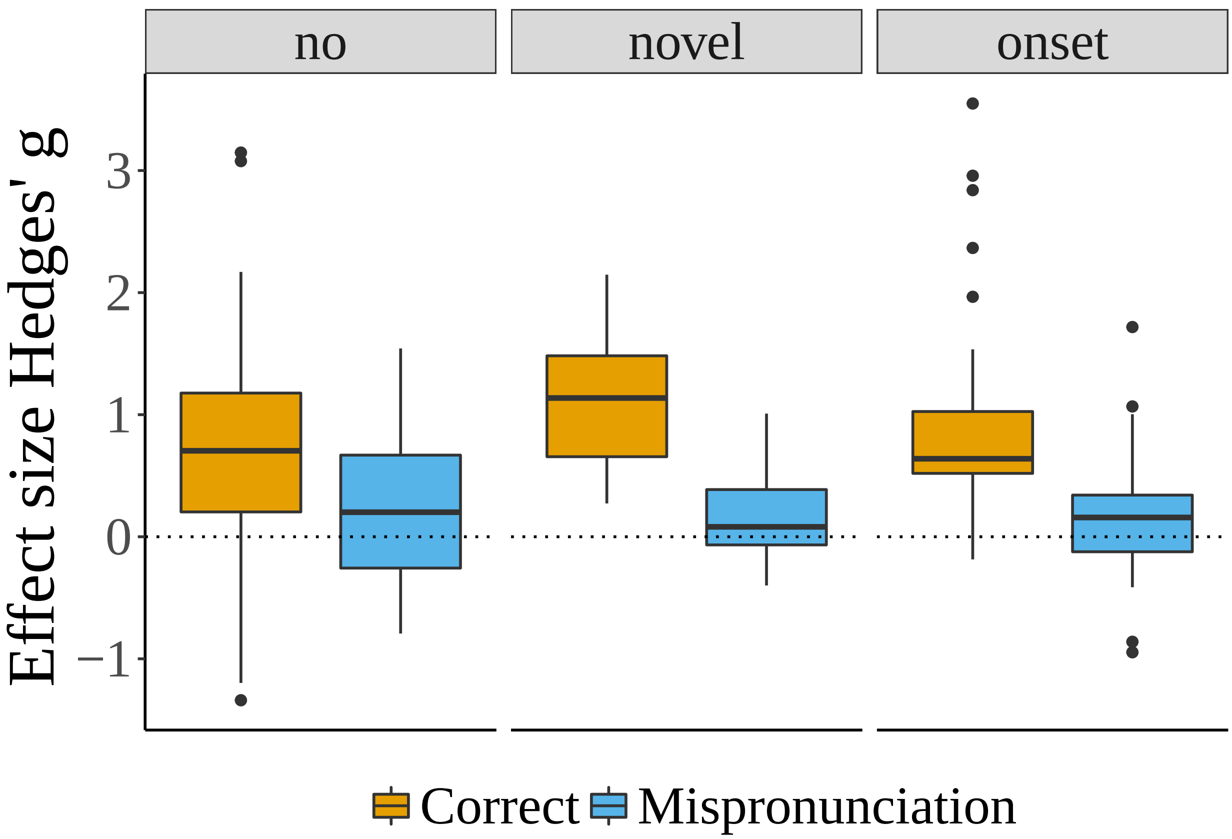
There was a significant interaction between distractor overlap and condition:

Hedges' g for `row.names(sum_eff1)` was `toString(sum_eff1$estimate)` (SE = `toString(sum_eff1$se)` (95% CI [`toString(sum_eff1$ci.lb)`, `toString(sum_eff1$ci.ub)`], p = `toString(sum_eff1$pval)`)

Hedges' g `row.names(sum_eff2)` was `toString(sum_eff2$estimate)` (SE = `toString(sum_eff2$se)` (95% CI [`toString(sum_eff2$ci.lb)`, `toString(sum_eff2$ci.ub)`], p = `toString(sum_eff2$pval)`)

Plotting Distractor Overlap with condition

```
db_ET_MPo = dat %>% filter(distractor_overlap == "onset" | distractor_overlap ==
    "novel" | distractor_overlap == "no")

p <- ggplot(db_ET_MPo, aes(condition_label, g_calc, fill = condition_label)) +
    facet_grid(. ~ distractor_overlap) + geom_boxplot() + # geom_smooth(method = 'lm', formula = y ~ lo
scale_fill_manual(values = cbPalette) + apatheme + theme(text = element_text(size = 25),
    legend.title = element_blank(), legend.position = "bottom", axis.title.x = element_blank(),
    axis.text.x = element_blank(), axis.ticks.x = element_blank()) + # xlab('Number of Features Changed
geom_hline(yintercept = 0, linetype = "dotted") + ylab("Effect size Hedges' g")

p
```




```
jpeg(filename = "figures/Distractor_overlap.jpg", width = 500, height = 300,
    units = "px")

p

dev.off()
```

```
## pdf
##   2
```

Distractor Overlap with age and condition moderators

```
db_ET_MPo = dat %>% filter(distractor_overlap == "onset" | distractor_overlap ==
    "novel" | distractor_overlap == "no")

rma_DistractorOverlap <- rma.mv(g_calc, g_var_calc, mods = ~age.C * condition *
    distractor_overlap, data = db_ET_MPo, random = ~collapse | short_cite)

summary(rma_DistractorOverlap)
```

```
##
## Multivariate Meta-Analysis Model (k = 218; method: REML)
##
##    logLik   Deviance        AIC        BIC       AICc
## -175.7365   351.4730   379.4730   426.0633   381.6720
##
## Variance Components:
##
## outer factor: short_cite (nlvls = 31)
## inner factor: collapse   (nlvls = 46)
##
##             estim    sqrt  fixed
## tau^2      0.1465  0.3828     no
## rho        0.7882             no
##
## Test for Residual Heterogeneity:
## QE(df = 206) = 858.5747, p-val < .0001
##
## Test of Moderators (coefficient(s) 2,3,4,5,6,7,8,9,10,11,12):
## QM(df = 11) = 214.7396, p-val < .0001
##
## Model Results:
##
##                                          estimate      se     zval    pval
## intrcpt                                    0.2703  0.1106   2.4446  0.0145
## age.C                                      0.0224  0.0194   1.1586  0.2466
## condition                                  0.5023  0.0523   9.6014  <.0001
## distractor_overlapno                       0.3269  0.1705   1.9172  0.0552
## distractor_overlapnovel                   -0.2013  0.1746  -1.1526  0.2491
## age.C:condition                           -0.0018  0.0131  -0.1415  0.8875
## age.C:distractor_overlapno                 0.0242  0.0284   0.8527  0.3938
## age.C:distractor_overlapnovel              0.0062  0.0324   0.1921  0.8477
## condition:distractor_overlapno            -0.3383  0.0949  -3.5640  0.0004
## condition:distractor_overlapnovel          0.3038  0.1045   2.9063  0.0037
## age.C:condition:distractor_overlapno      -0.0316  0.0197  -1.5986  0.1099
## age.C:condition:distractor_overlapnovel   -0.0316  0.0224  -1.4124  0.1578
##                                            ci.lb   ci.ub
## intrcpt                                   0.0536  0.4871    *
## age.C                                    -0.0155  0.0604
## condition                                 0.3998  0.6049  ***
```

```
## distractor_overlapno                      -0.0073   0.6611    .
## distractor_overlapnovel                   -0.5436   0.1410
## age.C:condition                           -0.0274   0.0237
## age.C:distractor_overlapno                -0.0315   0.0799
## age.C:distractor_overlapnovel             -0.0572   0.0697
## condition:distractor_overlapno            -0.5243  -0.1522  ***
## condition:distractor_overlapnovel          0.0989   0.5086   **
## age.C:condition:distractor_overlapno      -0.0702   0.0071
## age.C:condition:distractor_overlapnovel   -0.0755   0.0123
##
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
aov.type <- anova(rma_DistractorOverlap)

sum_eff1 <- round(coef(summary(rma_DistractorOverlap))[11, ], 2)
sum_eff2 <- round(coef(summary(rma_DistractorOverlap))[12, ], 2)
```

The moderator test was significant: `print(aov.type)`

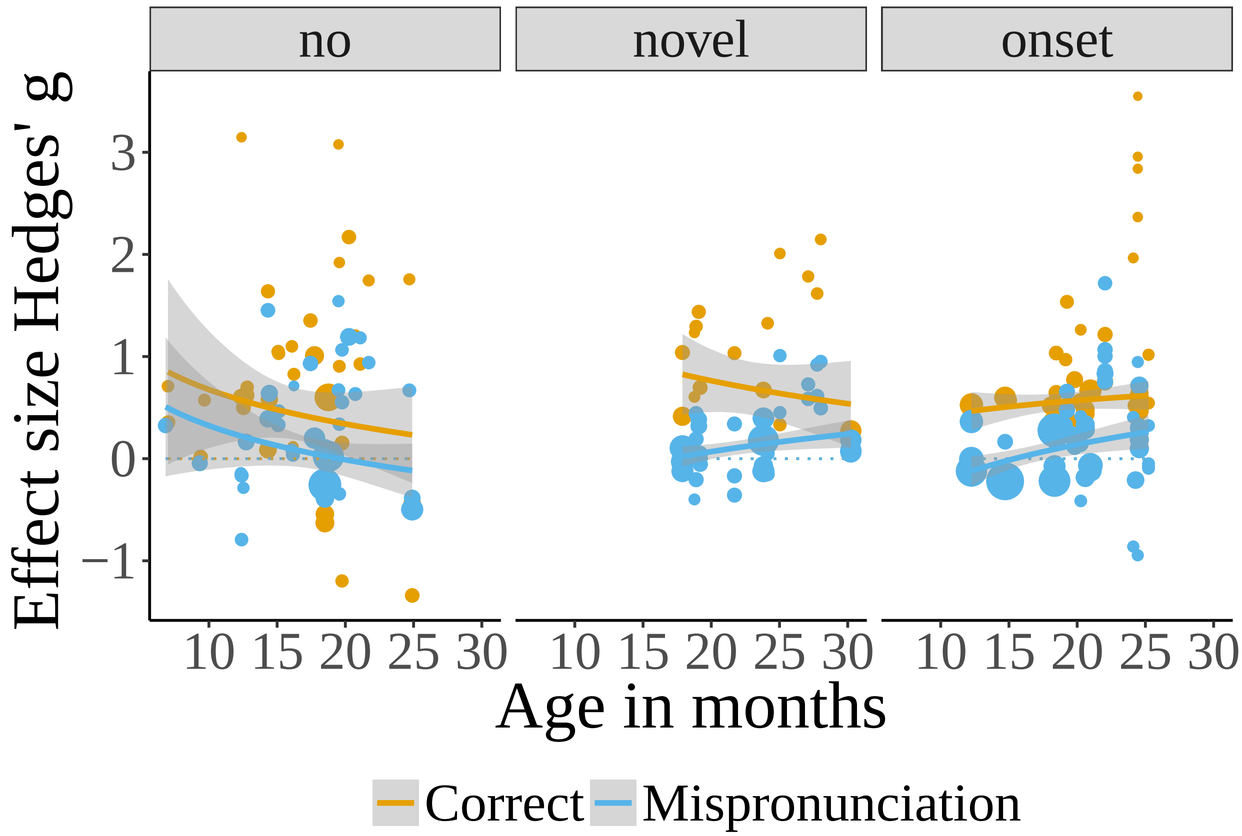
There was a significant interaction between distractor overlap, condition, and age:

Hedges' g for `row.names(sum_eff1)` was `toString(sum_eff1$estimate)` (SE = `toString(sum_eff1$se)` (95% CI [`toString(sum_eff1$ci.lb)`, `toString(sum_eff1$ci.ub)`], p = `toString(sum_eff1$pval)`)

Hedges' g `row.names(sum_eff2)` was `toString(sum_eff2$estimate)` (SE = `toString(sum_eff2$se)` (95% CI [`toString(sum_eff2$ci.lb)`, `toString(sum_eff2$ci.ub)`], p = `toString(sum_eff2$pval)`)

Plot Distractor Overlap, condition, and age

```
db_ET_MPo = dat %>% filter(distractor_overlap == "onset" | distractor_overlap ==
    "novel" | distractor_overlap == "no")

p <- ggplot(db_ET_MPo, aes(mean_age_1/30.44, g_calc, color = condition_label)) +
    facet_grid(. ~ distractor_overlap) + geom_point(aes(size = weights_g), show.legend = FALSE) +
    geom_line(y = 0, linetype = "dotted") + geom_smooth(method = "lm", formula = y ~
    log(x), aes(weight = weights_g)) + scale_colour_manual(values = cbPalette) +
    apatheme + theme(legend.title = element_blank(), legend.position = "bottom") +
    xlab("Age in months") + ylab("Effect size Hedges' g")

p
```




```r
jpeg(filename = "figures/Distractor_overlap_age.jpg", width = 500, height = 300,
    units = "px")

p

dev.off()
```

```
## pdf
##   2
```

Type of MP: Vowel, consonant, or tone

Type of MP: Vowel, consonant, or tone role in object identification

```r
db_MP_type <- subset(db_ET_MP, type_feature == "consonant" | type_feature ==
    "vowel")

# db_MP_type <- subset(db_ET_MP, type_feature != 'consonant_and_vowel')


rma_TypeFeaturesMP <- rma.mv(g_calc, g_var_calc, mods = ~type_feature, data = db_MP_type,
    random = ~collapse | short_cite)


# summary(rma_TypeFeaturesMP)
```

```r
aov.type <- anova(rma_TypeFeaturesMP)

type_feat <- round(coef(summary(rma_TypeFeaturesMP))[2, ], 2)
```

The moderator test was not significant: `print(aov.type)`

No significant effect of feature type:

Hedges' g for `rownames(sum_eff)` was `toString(type_feat$estimate)` (SE = `toString(type_feat$se)` (95% CI [`toString(type_feat$ci.lb)`, `toString(type_feat$ci.ub)`], p = `toString(type_feat$pval)`)

Type of MP: Vowel, consonant, or tone role in object identification with age moderator

```r
db_MP_type <- subset(db_ET_MP, type_feature == "consonant" | type_feature ==
    "vowel")

# db_MP_type <- subset(db_ET_MP, type_feature != 'consonant_and_vowel')


rma_TypeFeaturesMPage <- rma.mv(g_calc, g_var_calc, mods = ~type_feature * age.C,
    data = db_MP_type, random = ~collapse | short_cite)


# summary(rma_TypeFeaturesMPage)

aov.type <- anova(rma_TypeFeaturesMPage)

type_feat <- round(coef(summary(rma_TypeFeaturesMPage))[4, ], 2)
```

The moderator test was not significant: `print(aov.type)`

No significant effect of feature type:

Hedges' g for `rownames(sum_eff)` was `toString(type_feat$estimate)` (SE = `toString(type_feat$se)` (95% CI [`toString(type_feat$ci.lb)`, `toString(type_feat$ci.ub)`], p = `toString(type_feat$pval)`)

Type of MP: Vowel, consonant, or tone role in object identification with language family moderator

```r
db_MP_type <- subset(db_ET_MP, type_feature == "consonant" | type_feature ==
    "vowel")

# db_MP_type <- subset(db_ET_MP, type_feature != 'consonant_and_vowel')

dat_type_sub <- subset(db_MP_type, lang_family != "Sino-Tibetian")

rma_TypeFeatures_Lang <- rma.mv(g_calc, g_var_calc, mods = ~type_feature * lang_family,
    data = dat_type_sub, random = ~collapse | short_cite)


# summary(rma_TypeFeatures_Lang)

aov.type <- anova(rma_TypeFeatures_Lang)
```

```
type_feat <- round(coef(summary(rma_TypeFeatures_Lang))[4, ], 2)
```

The moderator test was not significant: `print(aov.type)`

No significant interaction between feature type and language family:

Hedges' g for `rownames(sum_eff)` was `toString(type_feat$estimate)` (SE = `toString(type_feat$se)` (95% CI [`toString(type_feat$ci.lb)`, `toString(type_feat$ci.ub)`], p = `toString(type_feat$pval)`)

Type of MP: Vowel, consonant, or tone with condition moderator

```
db_type <- subset(dat, type_feature == "consonant" | type_feature == "vowel")
# db_type <- subset(dat, type_feature != 'consonant_and_vowel')

rma_TypeFeaturesMPcond <- rma.mv(g_calc, g_var_calc, mods = ~type_feature *
    condition, data = db_type, random = ~collapse | short_cite)


# summary(rma_TypeFeaturesMPcond)

aov.type <- anova(rma_TypeFeaturesMPcond)

type_feat <- round(coef(summary(rma_TypeFeaturesMPcond))[2, ], 2)
```

The moderator test was not significant: `print(aov.type)`

No significant interaction between feature type and condition:

Hedges' g for `rownames(sum_eff)` was `toString(type_feat$estimate)` (SE = `toString(type_feat$se)` (95% CI [`toString(type_feat$ci.lb)`, `toString(type_feat$ci.ub)`], p = `toString(type_feat$pval)`)

Type of MP: Vowel, consonant, or tone with condition and age moderators

```
db_type <- subset(dat, type_feature == "consonant" | type_feature == "vowel")
# db_type <- subset(dat, type_feature != 'consonant_and_vowel')

rma_TypeFeaturesMPcondage <- rma.mv(g_calc, g_var_calc, mods = ~type_feature *
    condition * age.C, data = db_type, random = ~collapse | short_cite)


summary(rma_TypeFeaturesMPcondage)
```

```
##
## Multivariate Meta-Analysis Model (k = 194; method: REML)
##
##    logLik   Deviance        AIC        BIC       AICc
## -171.0949   342.1898   362.1898   394.4473   363.4470
##
## Variance Components:
##
## outer factor: short_cite (nlvls = 26)
## inner factor: collapse   (nlvls = 39)
##
```

```
##             estim    sqrt  fixed
## tau^2      0.1221  0.3494     no
## rho        0.7582             no
##
## Test for Residual Heterogeneity:
## QE(df = 186) = 795.7381, p-val < .0001
##
## Test of Moderators (coefficient(s) 2,3,4,5,6,7,8):
## QM(df = 7) = 128.8114, p-val < .0001
##
## Model Results:
##
##                                    estimate      se     zval    pval
## intrcpt                              0.2551  0.0756   3.3716  0.0007
## type_featurevowel                    0.0493  0.0891   0.5529  0.5803
## condition                            0.4005  0.0471   8.5111  <.0001
## age.C                                0.0170  0.0134   1.2710  0.2037
## type_featurevowel:condition          0.1708  0.0988   1.7281  0.0840
## type_featurevowel:age.C             -0.0003  0.0158  -0.0206  0.9835
## condition:age.C                     -0.0232  0.0120  -1.9412  0.0522
## type_featurevowel:condition:age.C    0.0480  0.0189   2.5347  0.0113
##                                       ci.lb   ci.ub
## intrcpt                             0.1068  0.4033  ***
## type_featurevowel                  -0.1254  0.2239
## condition                           0.3083  0.4928  ***
## age.C                              -0.0092  0.0432
## type_featurevowel:condition        -0.0229  0.3645    .
## type_featurevowel:age.C            -0.0313  0.0307
## condition:age.C                    -0.0467  0.0002    .
## type_featurevowel:condition:age.C   0.0109  0.0851    *
##
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
aov.type <- anova(rma_TypeFeaturesMPcondage)

type_feat <- round(coef(summary(rma_TypeFeaturesMPcondage))[8, ], 2)
```

The moderator test was significant: `print(aov.type)`

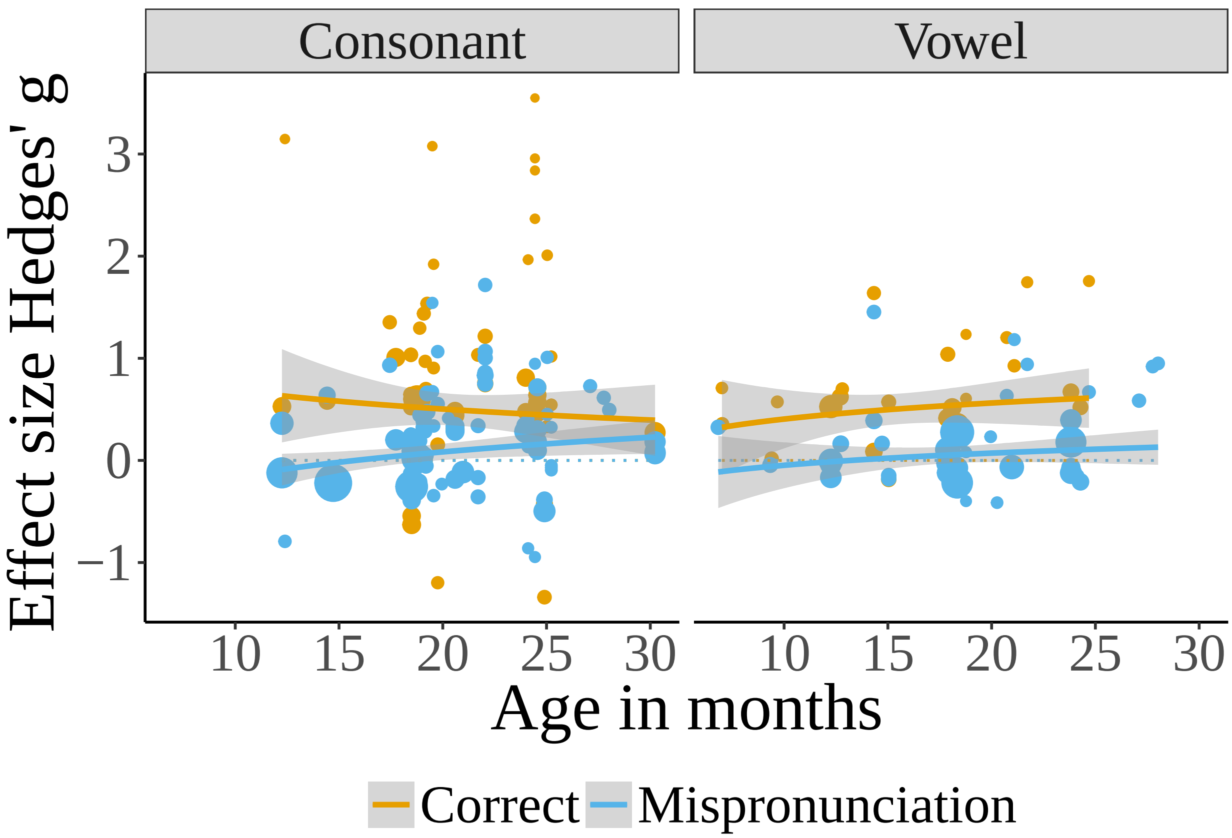
There was a significant interaction between feature type, condition, and age:

Hedges' g for `rownames(sum_eff)` was `toString(type_feat$estimate)` (SE = `toString(type_feat$se)` (95% CI [`toString(type_feat$ci.lb)`, `toString(type_feat$ci.ub)`], p = `toString(type_feat$pval)`)

Plot MP type: feature type, condition, and age

```r
dat_type <- subset(dat, type_feature == "consonant" | type_feature == "vowel")
dat_type$type_feature <- ifelse(dat_type$type_feature == "consonant", "Consonant",
    "Vowel")

# db_MP_type <- subset(db_ET_MP, type_feature != 'consonant_and_vowel')

# dat_type_sub <- subset(dat_type, lang_family != 'Sino-Tibetian')
```

```r
dat_type$condition_label = ifelse(dat_type$condition == 1, "Correct", "Mispronunciation")

p <- ggplot(dat_type, aes(mean_age_1/30.44, g_calc, color = condition_label)) +
    facet_grid(. ~ type_feature) + geom_point(aes(size = weights_g), show.legend = FALSE) +
    geom_line(y = 0, linetype = "dotted") + geom_smooth(method = "lm", formula = y ~
    log(x), aes(weight = weights_g)) + scale_colour_manual(values = cbPalette) +
    apatheme + theme(legend.title = element_blank(), legend.position = "bottom") +
    xlab("Age in months") + ylab("Effect size Hedges' g")

p
```




```r
jpeg(filename = "figures/FeatureType_Cond_Age.jpg", width = 500, height = 300,
    units = "px")

p

dev.off()
```

```
## pdf
##   2
```

Type of MP: Vowel, consonant, or tone with language family and condition moderators

```r
db_MP_type <- subset(dat, type_feature == "consonant" | type_feature == "vowel")
```

```r
# db_MP_type <- subset(db_ET_MP, type_feature != 'consonant_and_vowel')

dat_type_sub <- subset(db_MP_type, lang_family != "Sino-Tibetian")

rma_TypeFeatures_Lang <- rma.mv(g_calc, g_var_calc, mods = ~type_feature * lang_family *
    condition, data = dat_type_sub, random = ~collapse | short_cite)


summary(rma_TypeFeatures_Lang)
```

```
##
## Multivariate Meta-Analysis Model (k = 192; method: REML)
##
##    logLik   Deviance        AIC        BIC       AICc
## -169.0951   338.1902   358.1902   390.3396   359.4619
##
## Variance Components:
##
## outer factor: short_cite (nlvls = 25)
## inner factor: collapse   (nlvls = 38)
##
##             estim    sqrt  fixed
## tau^2      0.1167  0.3416     no
## rho        0.7065             no
##
## Test for Residual Heterogeneity:
## QE(df = 184) = 749.4955, p-val < .0001
##
## Test of Moderators (coefficient(s) 2,3,4,5,6,7,8):
## QM(df = 7) = 130.3617, p-val < .0001
##
## Model Results:
##
##                                                 estimate      se     zval
## intrcpt                                           0.2315  0.0786   2.9456
## type_featurevowel                                -0.0224  0.0958  -0.2334
## lang_familyRomanic                               -0.0745  0.2424  -0.3074
## condition                                         0.3766  0.0480   7.8505
## type_featurevowel:lang_familyRomanic              0.6855  0.2975   2.3040
## type_featurevowel:condition                       0.1324  0.0878   1.5080
## lang_familyRomanic:condition                      0.4296  0.2346   1.8312
## type_featurevowel:lang_familyRomanic:condition   -0.7414  0.2977  -2.4902
##                                                   pval    ci.lb    ci.ub
## intrcpt                                         0.0032   0.0774   0.3855
## type_featurevowel                               0.8154  -0.2101   0.1654
## lang_familyRomanic                              0.7585  -0.5497   0.4006
## condition                                       <.0001   0.2826   0.4706
## type_featurevowel:lang_familyRomanic            0.0212   0.1024   1.2687
## type_featurevowel:condition                     0.1315  -0.0397   0.3044
## lang_familyRomanic:condition                    0.0671  -0.0302   0.8894
## type_featurevowel:lang_familyRomanic:condition  0.0128  -1.3250  -0.1579
##
## intrcpt                                          **
## type_featurevowel
```

```
## lang_familyRomanic
## condition                                    ***
## type_featurevowel:lang_familyRomanic           *
## type_featurevowel:condition
## lang_familyRomanic:condition                   .
## type_featurevowel:lang_familyRomanic:condition    *
##
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
aov.type <- anova(rma_TypeFeatures_Lang)

type_feat <- round(coef(summary(rma_TypeFeatures_Lang))[8, ], 2)
```

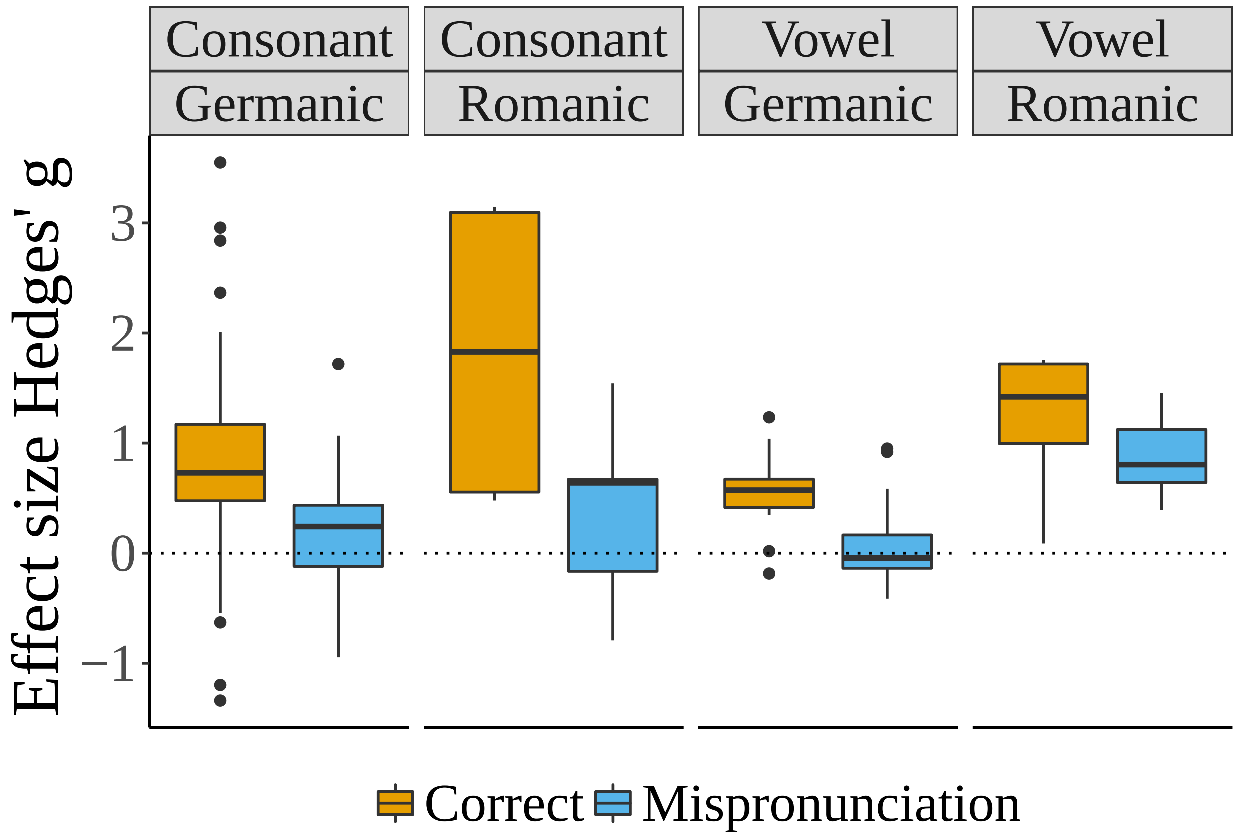
The moderator test was significant: `print(aov.type)`

There was a significant interaction between feature type, language family, and condition:

Hedges' g for `rownames(sum_eff)` was `toString(type_feat$estimate)` (SE = `toString(type_feat$se)` (95% CI [`toString(type_feat$ci.lb)`, `toString(type_feat$ci.ub)`], p = `toString(type_feat$pval)`)

Plot Language Family by MP type: Consonant, Vowel, or Tone?

```
dat_type <- subset(dat, type_feature == "consonant" | type_feature == "vowel")
dat_type$type_feature <- ifelse(dat_type$type_feature == "consonant", "Consonant",
    "Vowel")

# db_MP_type <- subset(db_ET_MP, type_feature != 'consonant_and_vowel')

dat_type_sub <- subset(dat_type, lang_family != "Sino-Tibetian")


dat_type_sub$condition_label = ifelse(dat_type_sub$condition == 1, "Correct",
    "Mispronunciation")


p <- ggplot(dat_type_sub, aes(condition_label, g_calc, fill = condition_label)) +
    geom_boxplot() + facet_grid(. ~ type_feature * lang_family) + # geom_line(y= 0, linetype='dotted')
# y ~ log(x), aes(weight=weights_g)) +
scale_fill_manual(values = cbPalette) + apatheme + theme(text = element_text(size = 25),
    legend.title = element_blank(), legend.position = "bottom", axis.title.x = element_blank(),
    axis.ticks.x = element_blank(), axis.text.x = element_blank()) + # xlab('Number of Features Changed
geom_hline(yintercept = 0, linetype = "dotted") + xlab("Language Family") +
    ylab("Effect size Hedges' g")
p
```




```
jpeg(filename = "figures/FeatureType_Cond_LangFam.jpg", width = 500, height = 300,
    units = "px")

p

dev.off()
```

```
## pdf
##   2
```

Type of MP: Vowel, consonant, or tone with language family, condition, and age moderators

```
dat_type <- subset(dat, type_feature == "consonant" | type_feature == "vowel")

# db_MP_type <- subset(db_ET_MP, type_feature != 'consonant_and_vowel')

dat_type_sub <- subset(dat_type, lang_family != "Sino-Tibetian")

rma_TypeFeatures_Lang <- rma.mv(g_calc, g_var_calc, mods = ~type_feature * lang_family *
    condition * age.C, data = dat_type_sub, random = ~collapse | short_cite)


summary(rma_TypeFeatures_Lang)
```

```
##
## Multivariate Meta-Analysis Model (k = 192; method: REML)
##
```

```
##    logLik   Deviance        AIC        BIC       AICc
## -158.0216   316.0431   352.0431   409.1118   356.3998
##
## Variance Components:
##
## outer factor: short_cite (nlvls = 25)
## inner factor: collapse   (nlvls = 38)
##
##             estim    sqrt  fixed
## tau^2      0.0968  0.3112     no
## rho        0.8552             no
##
## Test for Residual Heterogeneity:
## QE(df = 176) = 705.3725, p-val < .0001
##
## Test of Moderators (coefficient(s) 2,3,4,5,6,7,8,9,10,11,12,13,14,15,16):
## QM(df = 15) = 156.6351, p-val < .0001
##
## Model Results:
##
##                                                       estimate      se
## intrcpt                                                 0.2240  0.0748
## type_featurevowel                                       0.0002  0.0921
## lang_familyRomanic                                      0.9130  0.3682
## condition                                               0.3841  0.0486
## age.C                                                   0.0060  0.0124
## type_featurevowel:lang_familyRomanic                   -0.3182  0.4115
## type_featurevowel:condition                             0.1929  0.1091
## lang_familyRomanic:condition                           -0.7500  0.3801
## type_featurevowel:age.C                                 0.0094  0.0151
## lang_familyRomanic:age.C                                0.1676  0.0507
## condition:age.C                                        -0.0149  0.0126
## type_featurevowel:lang_familyRomanic:condition          0.5929  0.4366
## type_featurevowel:lang_familyRomanic:age.C             -0.1808  0.0684
## type_featurevowel:condition:age.C                       0.0302  0.0202
## lang_familyRomanic:condition:age.C                     -0.2092  0.0638
## type_featurevowel:lang_familyRomanic:condition:age.C    0.2919  0.0780
##                                                          zval    pval
## intrcpt                                                2.9940  0.0028
## type_featurevowel                                      0.0023  0.9981
## lang_familyRomanic                                     2.4793  0.0132
## condition                                              7.9068  <.0001
## age.C                                                  0.4808  0.6307
## type_featurevowel:lang_familyRomanic                  -0.7732  0.4394
## type_featurevowel:condition                            1.7680  0.0771
## lang_familyRomanic:condition                          -1.9730  0.0485
## type_featurevowel:age.C                                0.6176  0.5369
## lang_familyRomanic:age.C                               3.3045  0.0010
## condition:age.C                                       -1.1751  0.2400
## type_featurevowel:lang_familyRomanic:condition         1.3580  0.1745
## type_featurevowel:lang_familyRomanic:age.C            -2.6443  0.0082
## type_featurevowel:condition:age.C                      1.4932  0.1354
## lang_familyRomanic:condition:age.C                    -3.2810  0.0010
## type_featurevowel:lang_familyRomanic:condition:age.C   3.7417  0.0002
```

```
##                                                          ci.lb    ci.ub
## intrcpt                                                 0.0774   0.3707
## type_featurevowel                                      -0.1803   0.1807
## lang_familyRomanic                                      0.1912   1.6347
## condition                                               0.2889   0.4793
## age.C                                                  -0.0184   0.0304
## type_featurevowel:lang_familyRomanic                   -1.1248   0.4884
## type_featurevowel:condition                            -0.0209   0.4068
## lang_familyRomanic:condition                           -1.4951  -0.0049
## type_featurevowel:age.C                                -0.0203   0.0390
## lang_familyRomanic:age.C                                0.0682   0.2670
## condition:age.C                                        -0.0396   0.0099
## type_featurevowel:lang_familyRomanic:condition         -0.2629   1.4488
## type_featurevowel:lang_familyRomanic:age.C             -0.3148  -0.0468
## type_featurevowel:condition:age.C                      -0.0095   0.0699
## lang_familyRomanic:condition:age.C                     -0.3342  -0.0842
## type_featurevowel:lang_familyRomanic:condition:age.C    0.1390   0.4448
##
## intrcpt                                                 **
## type_featurevowel
## lang_familyRomanic                                       *
## condition                                              ***
## age.C
## type_featurevowel:lang_familyRomanic
## type_featurevowel:condition                              .
## lang_familyRomanic:condition                             *
## type_featurevowel:age.C
## lang_familyRomanic:age.C                               ***
## condition:age.C
## type_featurevowel:lang_familyRomanic:condition
## type_featurevowel:lang_familyRomanic:age.C              **
## type_featurevowel:condition:age.C
## lang_familyRomanic:condition:age.C                      **
## type_featurevowel:lang_familyRomanic:condition:age.C  ***
##
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
aov.type <- anova(rma_TypeFeatures_Lang)

type_feat <- round(coef(summary(rma_TypeFeatures_Lang))[8, ], 2)
```

The moderator test was significant: `print(aov.type)`

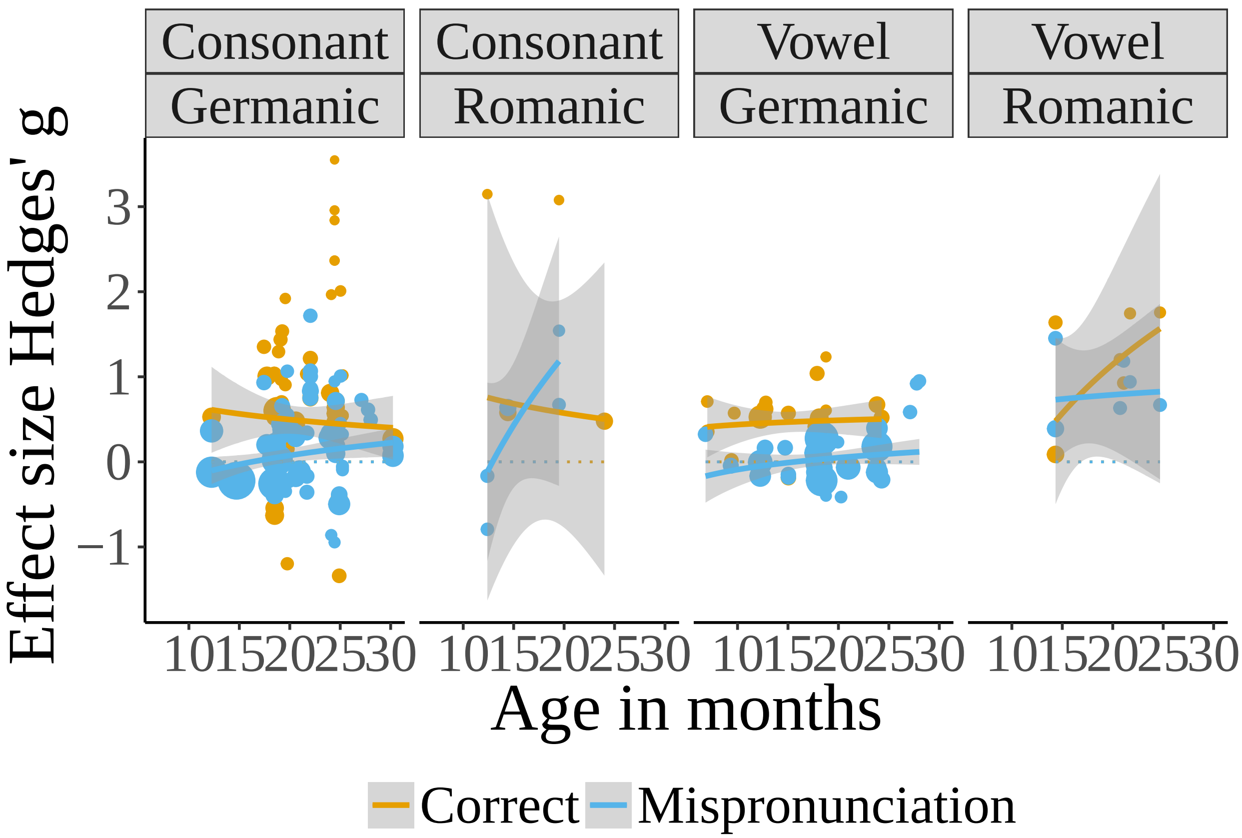
There was a significant interaction between feature type, age, and language family, and condition:

Hedges' g for `rownames(sum_eff)` was `toString(type_feat$estimate)` (SE = `toString(type_feat$se)` (95% CI [`toString(type_feat$ci.lb)`, `toString(type_feat$ci.ub)`], p = `toString(type_feat$pval)`)

Plot MP type: feature type, condition, and age

```r
dat_type <- subset(dat, type_feature == "consonant" | type_feature == "vowel")
dat_type$type_feature <- ifelse(dat_type$type_feature == "consonant", "Consonant",
    "Vowel")
```

```r
# db_MP_type <- subset(db_ET_MP, type_feature != 'consonant_and_vowel')

dat_type_sub <- subset(dat_type, lang_family != "Sino-Tibetian")


dat_type_sub$condition_label = ifelse(dat_type_sub$condition == 1, "Correct",
    "Mispronunciation")

p <- ggplot(dat_type_sub, aes(mean_age_1/30.44, g_calc, color = condition_label)) +
    facet_grid(. ~ type_feature * lang_family) + geom_point(aes(size = weights_g),
    show.legend = FALSE) + geom_line(y = 0, linetype = "dotted") + geom_smooth(method = "lm",
    formula = y ~ log(x), aes(weight = weights_g)) + scale_colour_manual(values = cbPalette) +
    apatheme + theme(legend.title = element_blank(), legend.position = "bottom") +
    xlab("Age in months") + ylab("Effect size Hedges' g")

p
```




```r
jpeg(filename = "figures/FeatureType_Cond_Age_LangFam.jpg", width = 500, height = 300,
    units = "px")

p

dev.off()
```

```
## pdf
##   2
```

Language effect

```r
dat_lang <- subset(db_ET_MP, lang_family != "Sino-Tibetian")

rma_lang_interaction <- rma.mv(g_calc, g_var_calc, mods = ~lang_family, data = dat_lang,
    random = ~collapse | short_cite)
# summary(rma_lang_interaction)

aov.type <- anova(rma_lang_interaction)

type_feat <- round(coef(summary(rma_lang_interaction))[2, ], 2)
```

The moderator test was not significant: `print(aov.type)`

There was no significant effect of language family:

Hedges' g for `rownames(sum_eff)` was `toString(type_feat$estimate)` (SE = `toString(type_feat$se)` (95% CI [`toString(type_feat$ci.lb)`, `toString(type_feat$ci.ub)`], p = `toString(type_feat$pval)`)

Language effect with age moderator

```r
dat_lang <- subset(db_ET_MP, lang_family != "Sino-Tibetian")

rma_lang_interaction <- rma.mv(g_calc, g_var_calc, mods = ~age.C * lang_family,
    data = dat_lang, random = ~collapse | short_cite)
summary(rma_lang_interaction)
```

```
##
## Multivariate Meta-Analysis Model (k = 126; method: REML)
##
##   logLik  Deviance       AIC       BIC      AICc
## -43.1745   86.3491   98.3491  115.1732   99.0795
##
## Variance Components:
##
## outer factor: short_cite (nlvls = 30)
## inner factor: collapse   (nlvls = 44)
##
##             estim    sqrt  fixed
## tau^2      0.1084  0.3292     no
## rho        0.9396             no
##
## Test for Residual Heterogeneity:
## QE(df = 122) = 357.0819, p-val < .0001
##
## Test of Moderators (coefficient(s) 2,3,4):
## QM(df = 3) = 17.4593, p-val = 0.0006
##
## Model Results:
##
##                           estimate      se    zval    pval    ci.lb
## intrcpt                     0.2025  0.0700  2.8939  0.0038   0.0654
## age.C                       0.0035  0.0099  0.3571  0.7210  -0.0159
## lang_familyRomanic          0.6921  0.2141  3.2329  0.0012   0.2725
```

```
## age.C:lang_familyRomanic    0.1150  0.0346  3.3231  0.0009   0.0472
##                            ci.ub
## intrcpt                   0.3397   **
## age.C                     0.0230
## lang_familyRomanic        1.1117   **
## age.C:lang_familyRomanic  0.1828  ***
##
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
aov.type <- anova(rma_lang_interaction)

type_feat <- round(coef(summary(rma_lang_interaction))[4, ], 2)
```

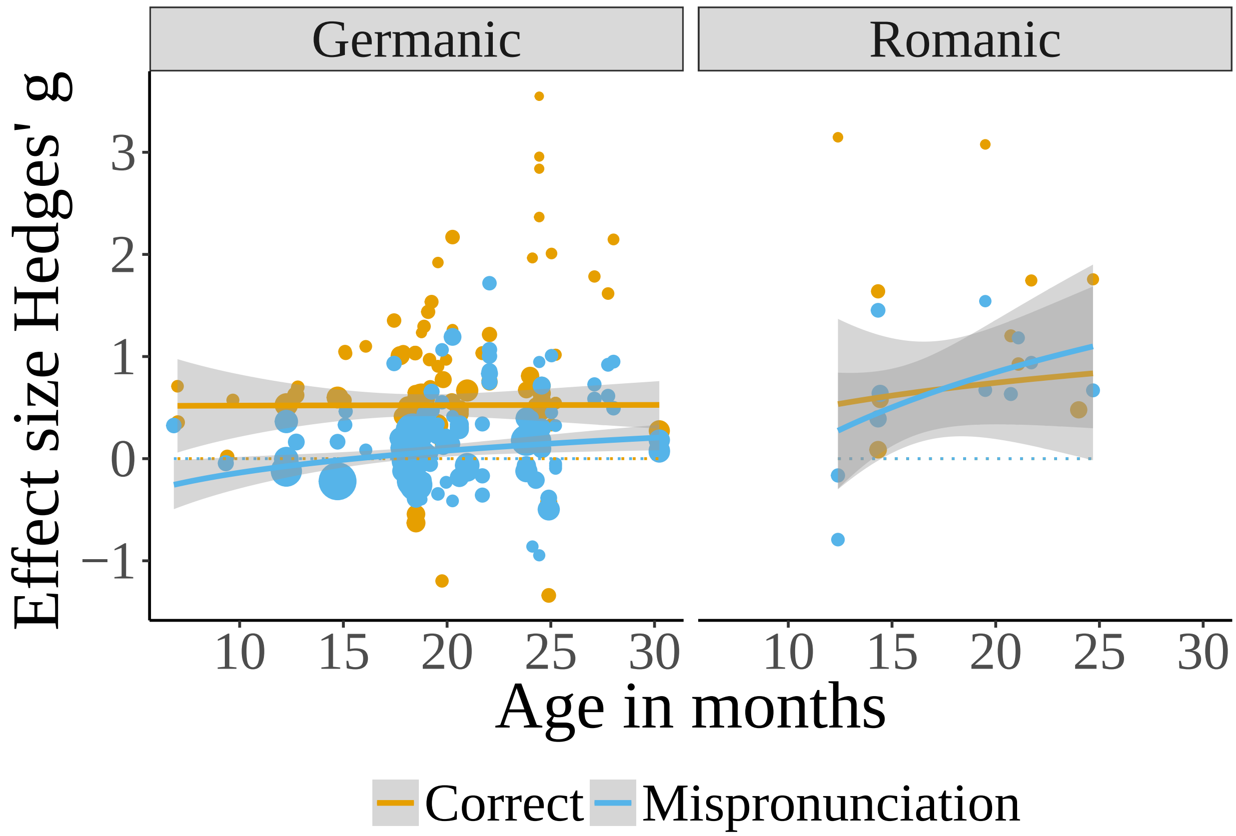
The moderator test was significant: `print(aov.type)`

There was a significant interaction between language family and age:

Hedges' g for `rownames(sum_eff)` was `toString(type_feat$estimate)` (SE = `toString(type_feat$se)` (95% CI [`toString(type_feat$ci.lb)`, `toString(type_feat$ci.ub)`], p = `toString(type_feat$pval)`)

Plot Language effect with age

```r
dat_lang <- subset(dat, lang_family != "Sino-Tibetian")


p <- ggplot(dat_lang, aes(mean_age_1/30.44, g_calc, color = condition_label)) +
    facet_grid(. ~ lang_family) + geom_point(aes(size = weights_g), show.legend = FALSE) +
    geom_line(y = 0, linetype = "dotted") + geom_smooth(method = "lm", formula = y ~
    log(x), aes(weight = weights_g)) + scale_colour_manual(values = cbPalette) +
    apatheme + theme(legend.title = element_blank(), legend.position = "bottom") +
    xlab("Age in months") + ylab("Effect size Hedges' g")

p
```




```
jpeg(filename = "figures/Language_Age.jpg", width = 500, height = 300, units = "px")

p

dev.off()
```

```
## pdf
##   2
```

Language effect with condition moderator

```
dat_lang <- subset(dat, lang_family != "Sino-Tibetian")


rma_lang_interaction <- rma.mv(g_calc, g_var_calc, mods = ~condition * lang_family,
    data = dat_lang, random = ~collapse | short_cite)
# summary(rma_lang_interaction)

aov.type <- anova(rma_lang_interaction)

type_feat <- round(coef(summary(rma_lang_interaction))[4, ], 2)
```

The moderator test was significant: `print(aov.type)`

But, there was no significant interaction between language family and condition:

Hedges' g for `rownames(sum_eff)` was `toString(type_feat$estimate)` (SE = `toString(type_feat$se)` (95% CI [`toString(type_feat$ci.lb)`, `toString(type_feat$ci.ub)`], p = `toString(type_feat$pval)`)

Language effect with age and condition moderators

```
dat_lang <- subset(dat, lang_family != "Sino-Tibetian")


rma_lang_interaction <- rma.mv(g_calc, g_var_calc, mods = ~age.C * condition *
    lang_family, data = dat_lang, random = ~collapse | short_cite)
# summary(rma_lang_interaction)

aov.type <- anova(rma_lang_interaction)

type_feat <- round(coef(summary(rma_lang_interaction))[8, ], 2)
```

The moderator test was significant: `print(aov.type)`

But, there was no significant interaction between language family, age, and condition:

Hedges' g for `rownames(sum_eff)` was `toString(type_feat$estimate)` (SE = `toString(type_feat$se)` (95% CI [`toString(type_feat$ci.lb)`, `toString(type_feat$ci.ub)`], p = `toString(type_feat$pval)`)