MP MetaAnalysis

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	216
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	120
group_means_two	7	57
t_one	4	39
t_two	5	35

Number of unique infants

The database contains data from 2252 unique infants.

Number of unique experimental conditions

The database contains data from 249 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	29
post-naming phase compared with chance (=50%)	9	23
post-pre difference score compared with chance (=0)	13	52

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)
```

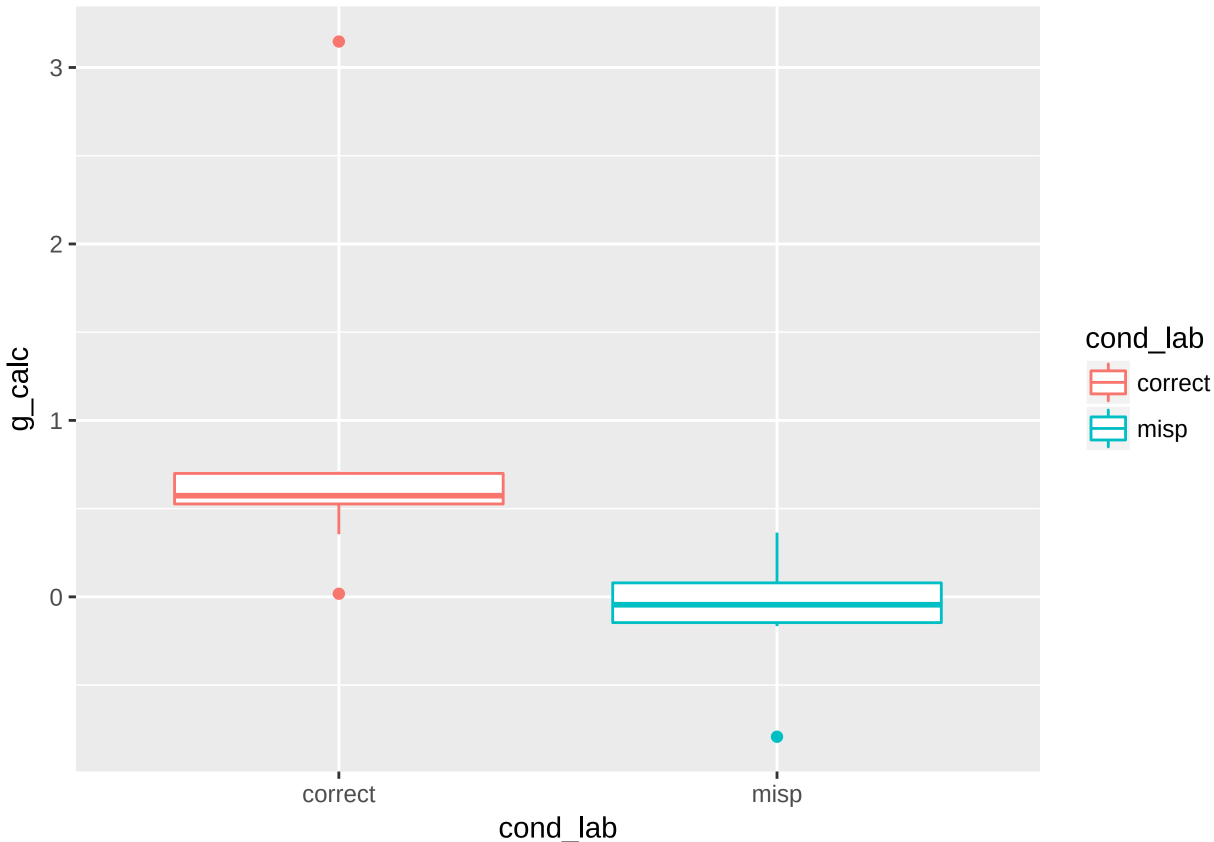
post_nam_dur	count
1.510	2
2.000	45
2.500	18
2.600	4
2.750	4
2.767	1
2.805	4
3.000	13
3.500	6
4.000	6
6.000	1

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 = 104; method: REML)
##
##    logLik   Deviance        AIC        BIC       AICc
## -111.8857   223.7713   229.7713   237.6755   230.0137
##
## Variance Components:
##
## outer factor: short_cite (nlvls = 32)
## inner factor: collapse   (nlvls = 52)
##
##             estim    sqrt  fixed
## tau^2      0.4483  0.6696     no
```

```
## rho        0.8886             no
##
## Test for Heterogeneity:
## Q(df = 103) = 625.6267, p-val < .0001
##
## Model Results:
##
## estimate       se     zval     pval    ci.lb    ci.ub
##   0.9078   0.1198   7.5784   <.0001   0.6730   1.1426      ***
##
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# 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

```
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 = 147; method: REML)
##
##   logLik  Deviance       AIC       BIC      AICc
## -70.1217  140.2434  146.2434  155.1942  146.4124
##
## Variance Components:
##
## outer factor: short_cite (nlvls = 32)
## inner factor: collapse   (nlvls = 52)
##
##             estim    sqrt  fixed
## tau^2      0.1192  0.3453     no
## rho        0.5924             no
##
## Test for Heterogeneity:
## Q(df = 146) = 462.5143, p-val < .0001
##
## Model Results:
##
## estimate       se     zval     pval    ci.lb    ci.ub
##   0.2498   0.0597   4.1835   <.0001   0.1328   0.3668      ***
##
## ---
```

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

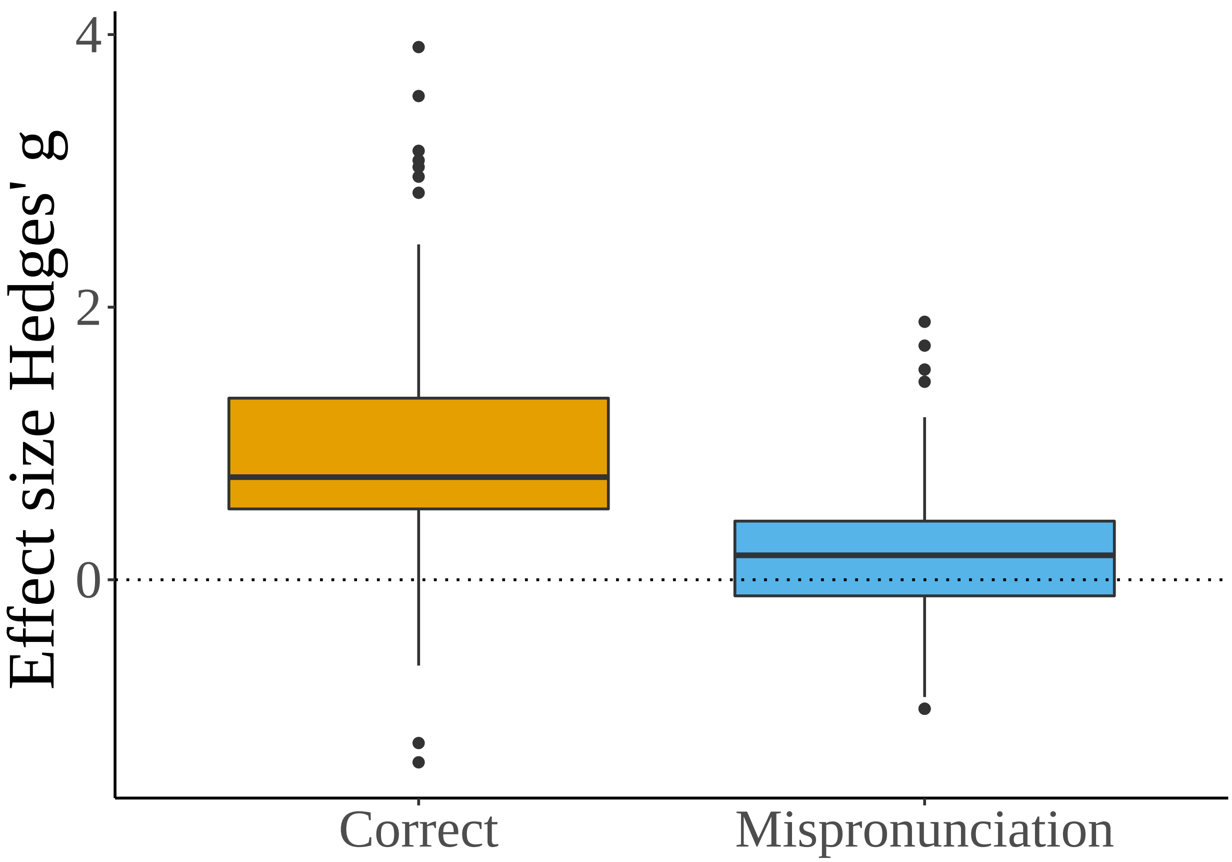
```
sum_eff <- round(coef(summary(rma_MP))[1, ], 2)
```

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)`)

Plot Object Identification

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

p <- ggplot(dat, aes(condition_label, g_calc, fill = condition_label)) + 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
```




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



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

```r
p

dev.off()
```

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

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 = 251; method: REML)
##
##    logLik   Deviance        AIC        BIC       AICc
## -252.9095   505.8189   513.8189   527.8887   513.9829
##
## Variance Components:
##
## outer factor: short_cite (nlvls = 32)
## inner factor: collapse   (nlvls = 52)
##
##             estim    sqrt  fixed
## tau^2      0.1371  0.3703     no
## rho        0.7381             no
##
## Test for Residual Heterogeneity:
## QE(df = 249) = 1088.1411, p-val < .0001
##
## Test of Moderators (coefficient(s) 2):
## QM(df = 1) = 215.7609, p-val < .0001
##
## Model Results:
##
##            estimate      se     zval    pval   ci.lb   ci.ub
## intrcpt      0.2792  0.0652   4.2827  <.0001  0.1514  0.4069  ***
## condition    0.4953  0.0337  14.6888  <.0001  0.4293  0.5614  ***
##
## ---
## 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

```r
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 = 104; method: REML)
##
##    logLik   Deviance        AIC        BIC       AICc
## -110.8134   221.6268   229.6268   240.1267   230.0392
##
## Variance Components:
##
## outer factor: short_cite (nlvls = 32)
## inner factor: collapse   (nlvls = 52)
##
##             estim    sqrt  fixed
## tau^2      0.4458  0.6677     no
## rho        0.8835             no
##
## Test for Residual Heterogeneity:
## QE(df = 102) = 619.1502, p-val < .0001
##
## Test of Moderators (coefficient(s) 2):
## QM(df = 1) = 0.6778, p-val = 0.4103
##
## Model Results:
##
##          estimate      se    zval    pval    ci.lb   ci.ub
## intrcpt    0.9202  0.1203  7.6515  <.0001   0.6845  1.1559  ***
## age.C      0.0145  0.0176  0.8233  0.4103  -0.0200  0.0490
##
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
# 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 = 147; method: REML)
##
##   logLik  Deviance       AIC       BIC      AICc
## -68.8541  137.7083  145.7083  157.6152  145.9940
##
## Variance Components:
##
## outer factor: short_cite (nlvls = 32)
## inner factor: collapse   (nlvls = 52)
##
##             estim    sqrt  fixed
## tau^2      0.1181  0.3437     no
## rho        0.5830             no
##
## Test for Residual Heterogeneity:
## QE(df = 145) = 449.1871, p-val < .0001
##
## Test of Moderators (coefficient(s) 2):
## QM(df = 1) = 1.7151, p-val = 0.1903
##
## Model Results:
##
##          estimate      se    zval    pval    ci.lb   ci.ub
## intrcpt    0.2613  0.0599  4.3583  <.0001   0.1438  0.3788  ***
## age.C      0.0149  0.0114  1.3096  0.1903  -0.0074  0.0372
##
## ---
## 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 = 251; method: REML)
##
##    logLik   Deviance        AIC        BIC       AICc
## -251.2299   502.4597   514.4597   535.5160   514.8097
##
## Variance Components:
##
## outer factor: short_cite (nlvls = 32)
## inner factor: collapse   (nlvls = 52)
##
##             estim    sqrt  fixed
## tau^2      0.1331  0.3648     no
## rho        0.7254             no
##
## Test for Residual Heterogeneity:
## QE(df = 247) = 1068.3373, p-val < .0001
##
## Test of Moderators (coefficient(s) 2,3,4):
## QM(df = 3) = 218.6210, p-val < .0001
##
## Model Results:
##
##                   estimate      se     zval    pval    ci.lb   ci.ub
## intrcpt             0.2935  0.0648   4.5324  <.0001   0.1666  0.4204  ***
## age.C               0.0171  0.0113   1.5136  0.1301  -0.0051  0.0393
## condition           0.4984  0.0344  14.4930  <.0001   0.4310  0.5658  ***
## age.C:condition     0.0026  0.0076   0.3436  0.7312  -0.0123  0.0175
##
## ---
## 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)`

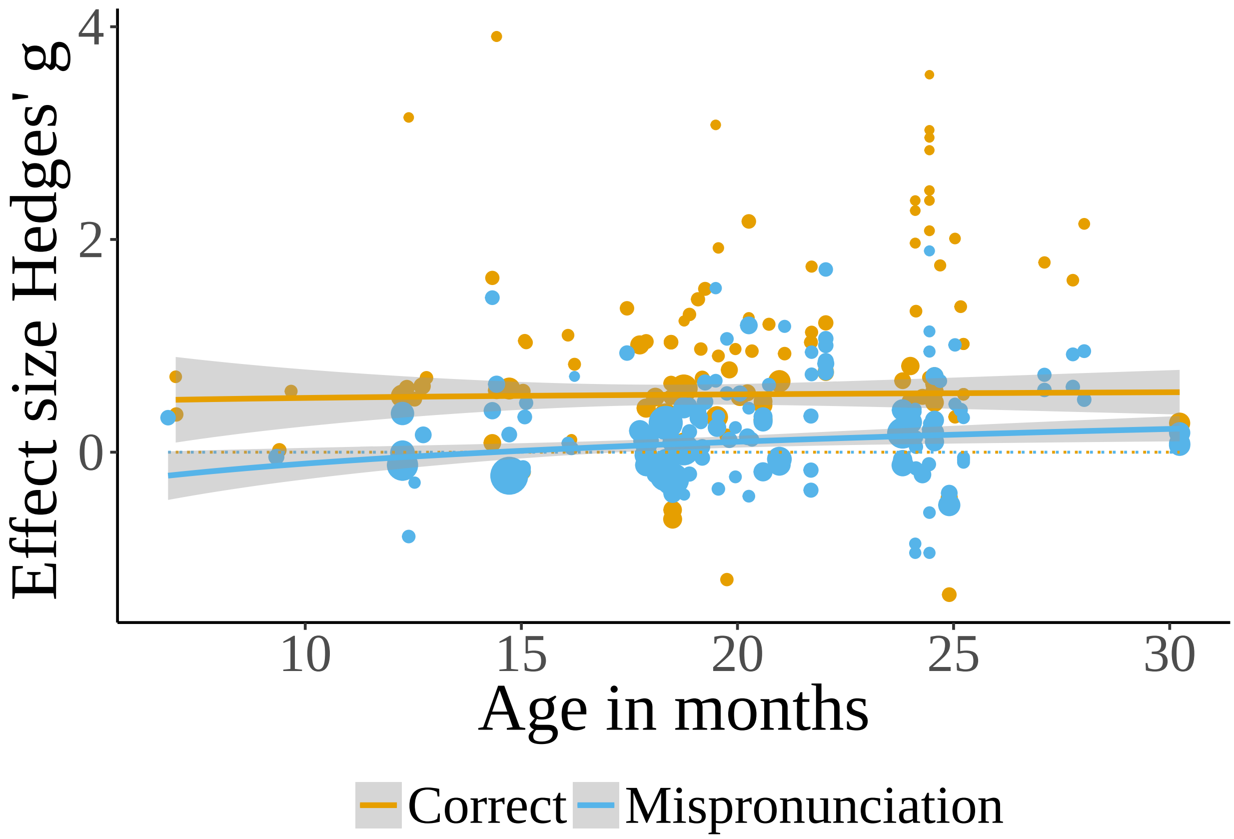
There was no significant interaction between condition and age:

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	87
production	5

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
```

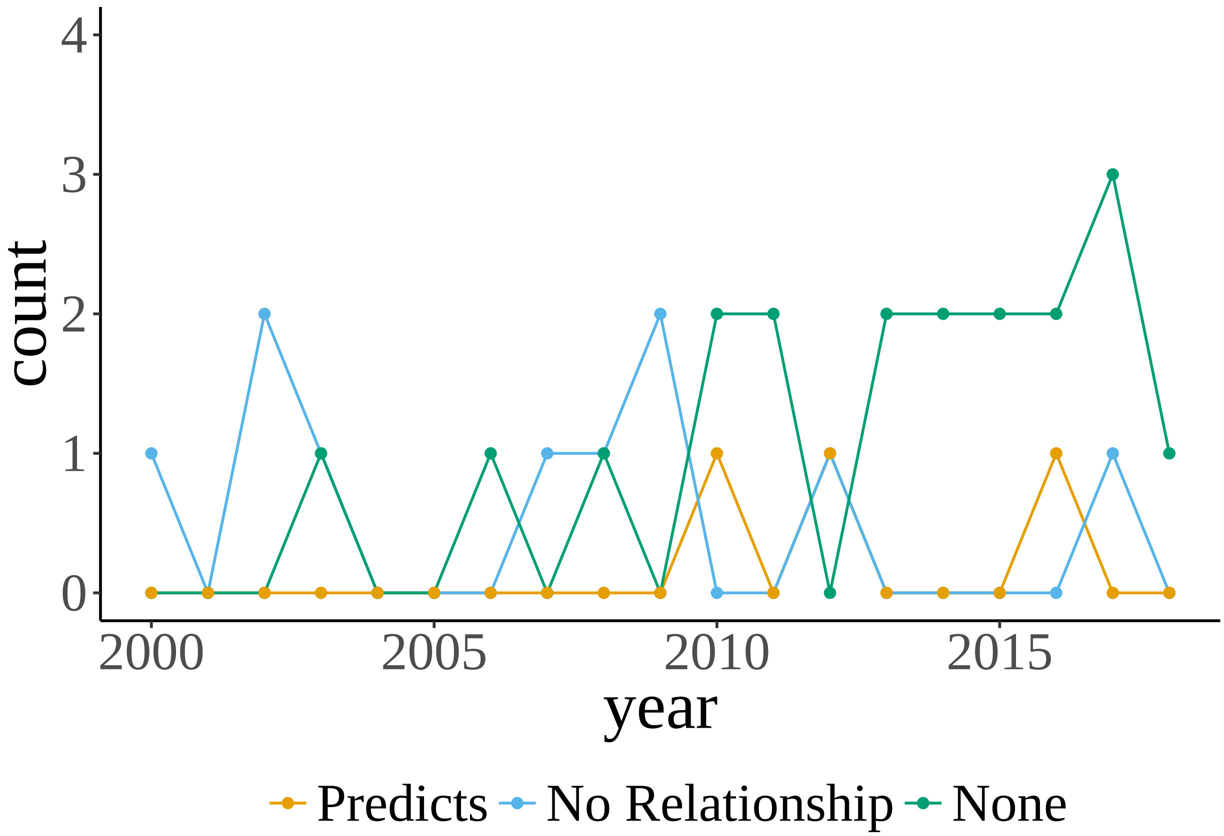
```r
# we're relying on the library meta function metacor
prodr <- subset(db_ET_correct, !is.na(db_ET_correct$r_production) & r_production <
    1)

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]       5.0
## Zesiger et al. (2012)           -0.1720 [-0.5775; 0.2335]       5.3
## Mani, Coleman, & Plunkett (2008)  0.0700 [-0.1861; 0.3261]      13.2
## Mani & Plunkett 2007            -0.1100 [-0.4696; 0.2496]       6.7
## Mani & Plunkett 2007            -0.1100 [-0.4635; 0.2435]       6.9
## Swingley & Aslin (2002)          0.1820 [-0.1970; 0.5610]       6.0
## Swingley & Aslin (2002)          0.1820 [-0.2131; 0.5771]       5.6
## Swingley 2003                    0.1800 [-0.1406; 0.5006]       8.4
## Swingley 2003                    0.0700 [-0.3367; 0.4767]       5.2
## Ramon-Casas et al. 2009          0.0980 [-0.3068; 0.5028]       5.3
## Ramon-Casas et al. 2009         -0.1470 [-0.5468; 0.2528]       5.4
## Ramon-Casas et al. 2009         -0.2300 [-0.6171; 0.1571]       5.8
## Ramon-Casas et al. 2009          0.2400 [-0.1451; 0.6251]       5.9
## Ramon-Casas et al. 2009          0.4350 [ 0.1037; 0.7663]       7.9
## H\xbfjen et al.                  0.2220 [-0.2591; 0.7031]       3.7
## H\xbfjen et al.                 -0.1480 [-0.6430; 0.3470]       3.5
##                                 %W(random)
## Zesiger et al. (2012)                  5.0
## Zesiger et al. (2012)                  5.3
## Mani, Coleman, & Plunkett (2008)       13.2
## Mani & Plunkett 2007                   6.7
## Mani & Plunkett 2007                   6.9
## Swingley & Aslin (2002)                6.0
## Swingley & Aslin (2002)                5.6
## Swingley 2003                          8.4
## Swingley 2003                          5.2
## Ramon-Casas et al. 2009                5.3
## Ramon-Casas et al. 2009                5.4
## Ramon-Casas et al. 2009                5.8
```

```
## Ramon-Casas et al. 2009                 5.9
## Ramon-Casas et al. 2009                 7.9
## H\xbfjen et al.                         3.7
## H\xbfjen et al.                         3.5
##
## Number of studies combined: k = 16
##
##                          COR            95%-CI    z p-value
## Fixed effect model   0.0601 [-0.0331; 0.1533] 1.26  0.2061
## Random effects model 0.0601 [-0.0331; 0.1533] 1.26  0.2061
##
## Quantifying heterogeneity:
## tau^2 = 0; H = 1.00 [1.00; 1.42]; I^2 = 0.0% [0.0%; 50.7%]
##
## Test of heterogeneity:
##      Q d.f. p-value
##  14.51   15  0.4870
##
## Details on meta-analytical method:
## - Inverse variance method
## - DerSimonian-Laird estimator for tau^2
## - Untransformed correlations
```

```
# how did vocabulary collection change over time?

dv <- read.csv("data/vocab_collection.csv", header = T)

years.v <- as.data.frame(seq(from = 2000, to = 2018, by = 1))
relat.v <- c("no_vocab", "no_relationship", "positive")

fake.v <- merge(years.v, relat.v, all = T)
names(fake.v) <- c("year", "relationship")
dat.v <- merge(dv, fake.v, by = c("year", "relationship"), all.y = T)
dat.v$vocab <- as.character(dat.v$vocab)
dat.v$relationship <- as.character(dat.v$relationship)
dat.v$short_cite <- as.character(dat.v$short_cite)

dat.v$vocab[is.na(dat.v$vocab)] <- "none"
dat.v$short_cite <- ifelse(dat.v$vocab == "none", "none", dat.v$short_cite)


dat.v$short_cite[is.na(dat.v$short_cite)] <- "none"


dat.v$tested <- ifelse(dat.v$short_cite == "none", "no", "yes")

vocab_data1 <- dat.v %>% group_by(year, relationship, tested) %>% summarize(count = n())

vocab_data1 <- as.data.frame(vocab_data1)
vocab_data1$count <- ifelse(vocab_data1$tested == "no", 0, vocab_data1$count)
vocab_data1$year <- as.numeric(vocab_data1$year)

vocab_data1$relationship <- ifelse(vocab_data1$relationship == "no_vocab", "None",
    ifelse(vocab_data1$relationship == "positive", "Predicts", "No Relationship"))
```

```
vocab_data1$relationship <- factor(vocab_data1$relationship, levels = c("Predicts",
    "No Relationship", "None"))

p <- ggplot(vocab_data1, aes(year, count, color = relationship)) + geom_line() +
    geom_point() + scale_y_continuous(breaks = c(0, 1, 2, 3, 4), limits = c(0,
    4)) + scale_colour_manual(values = cbPalette) + apatheme + theme(legend.title = element_blank(),
    legend.position = "bottom")
p
```




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

p

dev.off()
```

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

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")
```

```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 <- 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 = 211; method: REML)
##
##    logLik   Deviance        AIC        BIC       AICc
## -257.1871   514.3743   522.3743   535.7436   522.5703
##
## Variance Components:
##
## outer factor: short_cite (nlvls = 27)
## inner factor: collapse   (nlvls = 49)
##
##             estim    sqrt  fixed
## tau^2      0.1368  0.3698     no
## rho        0.6718             no
##
## Test for Residual Heterogeneity:
## QE(df = 209) = 1027.7694, p-val < .0001
##
## Test of Moderators (coefficient(s) 2):
## QM(df = 1) = 137.2135, p-val < .0001
##
## Model Results:
##
##                        estimate      se      zval    pval    ci.lb
## intrcpt                  0.7022  0.0713    9.8437  <.0001   0.5624
## as.numeric(n_feature)   -0.3062  0.0261  -11.7138  <.0001  -0.3574
##                          ci.ub
## intrcpt                 0.8420  ***
## as.numeric(n_feature)  -0.2550  ***
##
## ---
## 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)
```

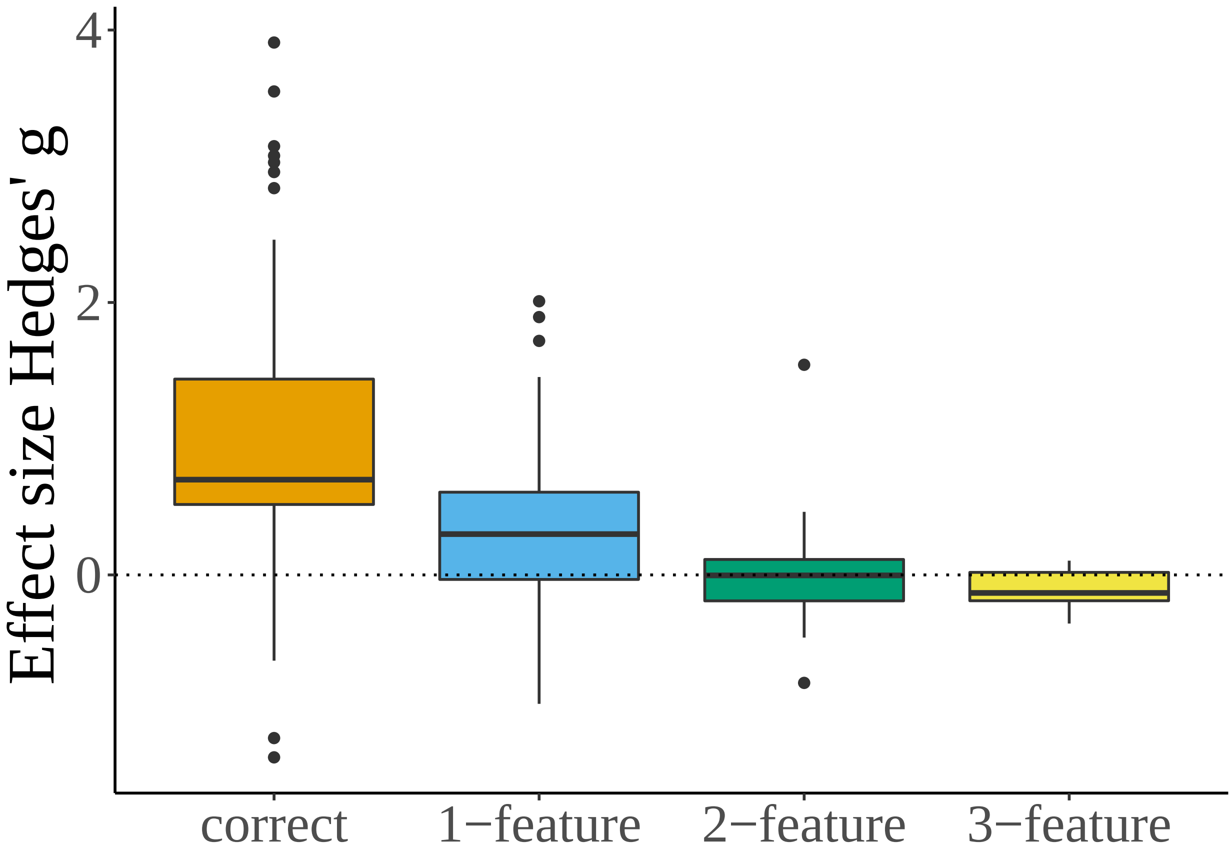
The moderator test was significant: `print(aov.feat)`

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

```
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)
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

```
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

```
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) *
    age.C, data = dat_f, random = ~collapse | short_cite)
```

```r
# 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)

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

```
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 = 251; method: REML)
##
##    logLik   Deviance        AIC        BIC       AICc
## -247.3148   494.6296   514.6296   549.5602   515.5778
##
## Variance Components:
##
## outer factor: short_cite (nlvls = 32)
## inner factor: collapse   (nlvls = 52)
##
##             estim    sqrt  fixed
## tau^2      0.1357  0.3684     no
## rho        0.7175             no
##
## Test for Residual Heterogeneity:
## QE(df = 243) = 1064.6022, p-val < .0001
##
## Test of Moderators (coefficient(s) 2,3,4,5,6,7,8):
## QM(df = 7) = 224.9573, p-val < .0001
##
## Model Results:
##
##                                                        estimate      se
## intrcpt                                                  0.3698  0.0785
## age.C                                                    0.0242  0.0138
## condition                                                0.4666  0.0415
## as.factor(object_pair)familiar_novel                    -0.2541  0.1471
## age.C:condition                                          0.0020  0.0092
## age.C:as.factor(object_pair)familiar_novel               0.0038  0.0288
## condition:as.factor(object_pair)familiar_novel           0.1755  0.0894
## age.C:condition:as.factor(object_pair)familiar_novel   -0.0203  0.0198
##                                                           zval    pval
## intrcpt                                                 4.7107  <.0001
## age.C                                                   1.7481  0.0804
## condition                                              11.2325  <.0001
## as.factor(object_pair)familiar_novel                   -1.7273  0.0841
## age.C:condition                                         0.2153  0.8295
## age.C:as.factor(object_pair)familiar_novel              0.1312  0.8956
## condition:as.factor(object_pair)familiar_novel          1.9637  0.0496
```

```
## age.C:condition:as.factor(object_pair)familiar_novel  -1.0267  0.3046
##                                                         ci.lb   ci.ub
## intrcpt                                                0.2160  0.5237  ***
## age.C                                                 -0.0029  0.0512    .
## condition                                              0.3852  0.5480  ***
## as.factor(object_pair)familiar_novel                  -0.5425  0.0342    .
## age.C:condition                                       -0.0161  0.0201
## age.C:as.factor(object_pair)familiar_novel            -0.0526  0.0602
## condition:as.factor(object_pair)familiar_novel         0.0003  0.3507    *
## age.C:condition:as.factor(object_pair)familiar_novel  -0.0590  0.0184
##
## ---
## 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)
```

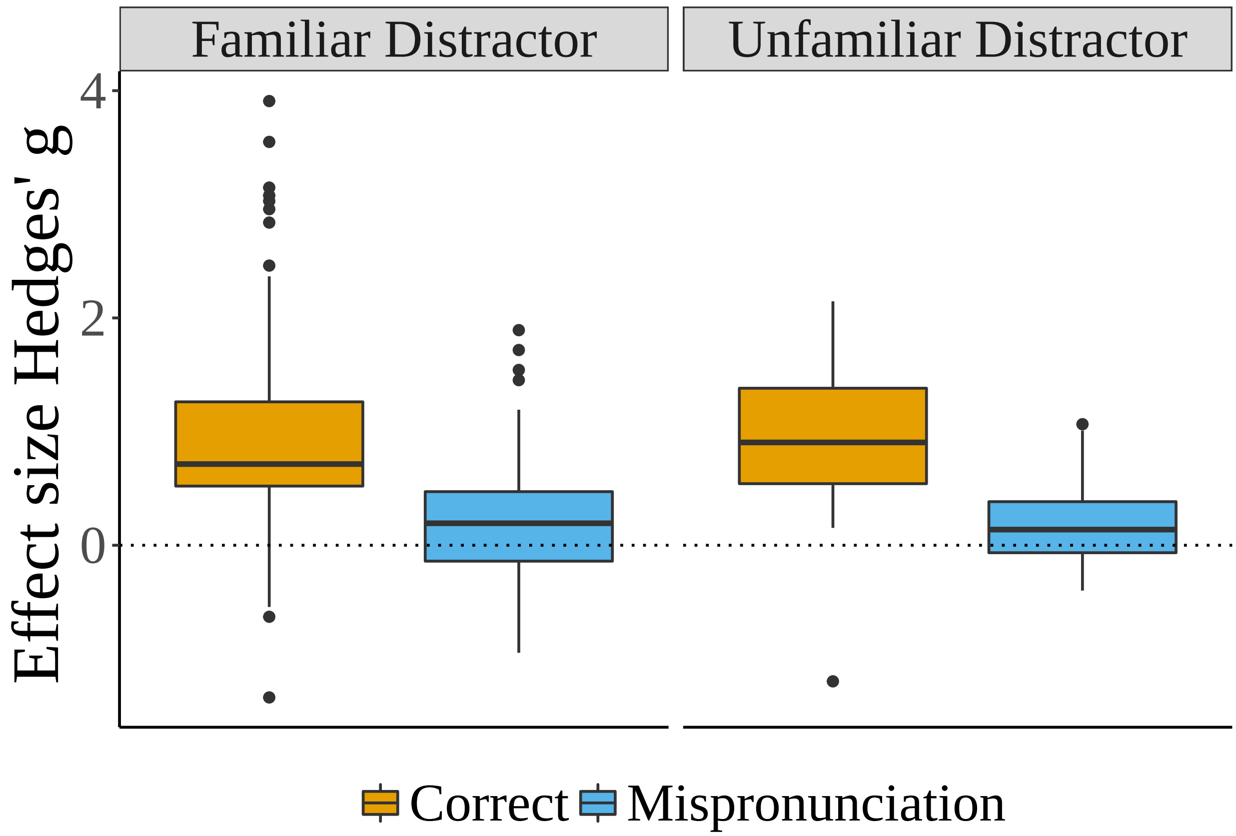
The moderator test was significant: `print(aov.type)`

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),
```

```
    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

```
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 = 186; method: REML)
##
##    logLik   Deviance        AIC        BIC       AICc
## -178.9911   357.9823   369.9823   389.2063   370.4623
##
## Variance Components:
##
## outer factor: short_cite (nlvls = 23)
## inner factor: collapse   (nlvls = 38)
##
##              estim    sqrt  fixed
## tau^2       0.1710  0.4136     no
## rho         0.7832             no
##
## Test for Residual Heterogeneity:
## QE(df = 182) = 822.0736, p-val < .0001
##
```

```
## Test of Moderators (coefficient(s) 2,3,4):
## QM(df = 3) = 150.3023, p-val < .0001
##
## Model Results:
##
##                                                 estimate      se     zval
## intrcpt                                           0.3836  0.0989   3.8784
## condition                                         0.4293  0.0457   9.3896
## as.factor(object_pair)familiar_novel             -0.2677  0.1549  -1.7278
## condition:as.factor(object_pair)familiar_novel    0.1852  0.0914   2.0258
##                                                   pval    ci.lb   ci.ub
## intrcpt                                         0.0001   0.1897  0.5774
## condition                                       <.0001   0.3397  0.5189
## as.factor(object_pair)familiar_novel            0.0840  -0.5713  0.0360
## condition:as.factor(object_pair)familiar_novel  0.0428   0.0060  0.3644
##
## 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
```

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

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

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

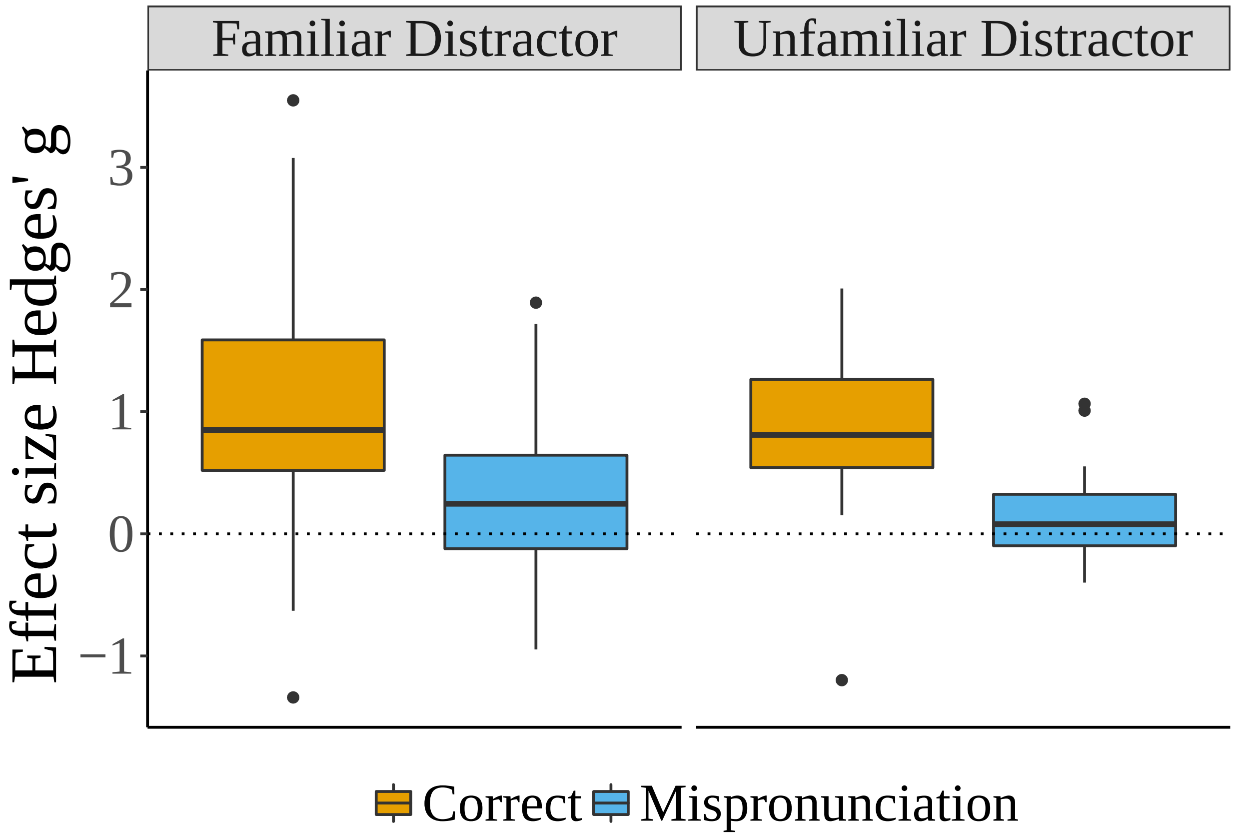
There was a significant interaction between distractor familiarity and condition:

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

```r
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 = 191; method: REML)
##
```

```
##    logLik   Deviance        AIC        BIC       AICc
## -201.2226   402.4451   422.4451   454.5400   423.7242
##
## Variance Components:
##
## outer factor: short_cite (nlvls = 24)
## inner factor: collapse   (nlvls = 41)
##
##             estim    sqrt  fixed
## tau^2      0.1718  0.4144     no
## rho        0.6468             no
##
## Test for Residual Heterogeneity:
## QE(df = 183) = 890.1960, p-val < .0001
##
## Test of Moderators (coefficient(s) 2,3,4,5,6,7,8):
## QM(df = 7) = 185.3378, p-val < .0001
##
## Model Results:
##
##                                         estimate      se     zval    pval
## intrcpt                                   0.2752  0.0919   2.9952  0.0027
## mispron_locationmedial                    0.0765  0.1698   0.4503  0.6525
## condition                                 0.4842  0.0425  11.3954  <.0001
## age.C                                     0.0217  0.0173   1.2580  0.2084
## mispron_locationmedial:condition          0.1078  0.0997   1.0812  0.2796
## mispron_locationmedial:age.C              0.0009  0.0310   0.0275  0.9781
## condition:age.C                          -0.0142  0.0110  -1.2859  0.1985
## mispron_locationmedial:condition:age.C    0.0374  0.0234   1.5987  0.1099
##                                           ci.lb   ci.ub
## intrcpt                                  0.0951  0.4553   **
## mispron_locationmedial                  -0.2564  0.4094
## condition                                0.4009  0.5675  ***
## age.C                                   -0.0121  0.0556
## mispron_locationmedial:condition        -0.0876  0.3032
## mispron_locationmedial:age.C            -0.0600  0.0617
## condition:age.C                         -0.0358  0.0074
## mispron_locationmedial:condition:age.C  -0.0085  0.0833
##
## ---
## 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)
```

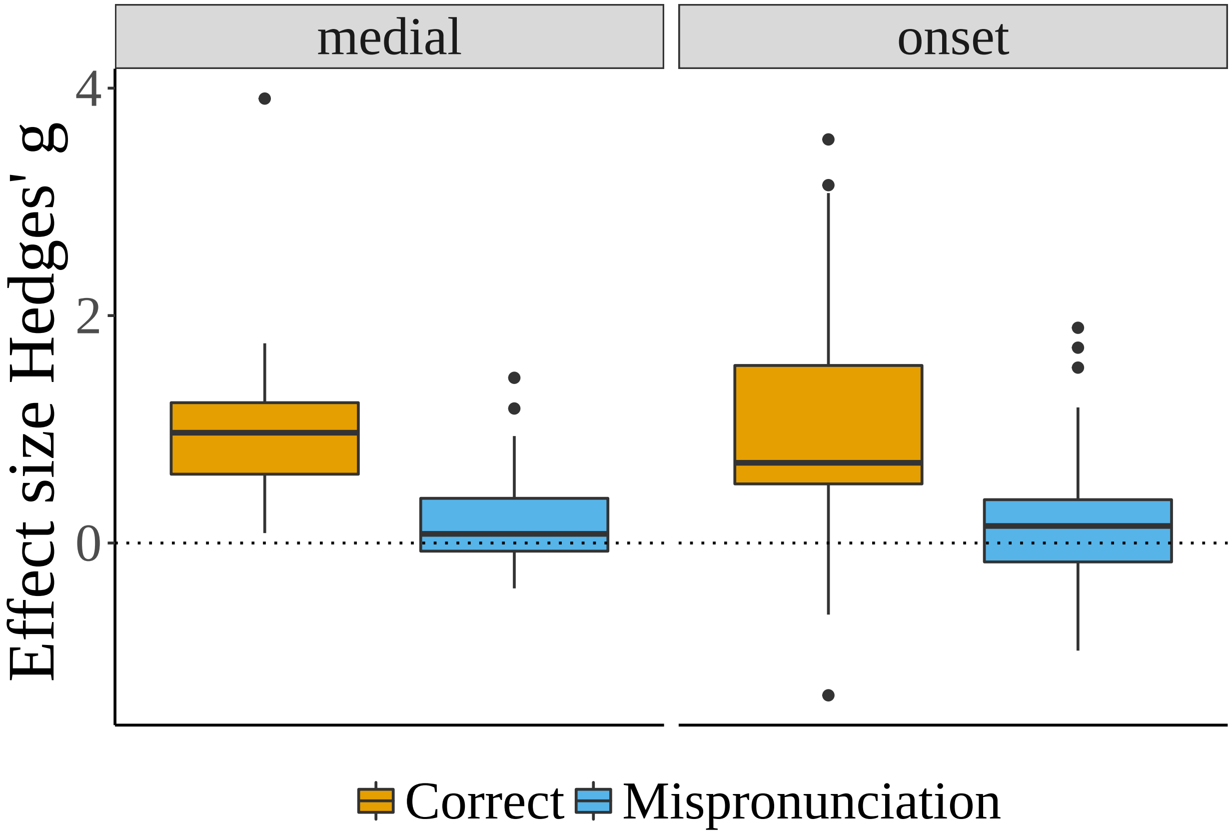
The moderator test was significant: `print(aov.type)`

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

```r
# 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
```




```r
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 = 244; method: REML)
##
##    logLik   Deviance        AIC        BIC       AICc
## -240.4878   480.9756   496.9756   524.7538   497.6044
##
## Variance Components:
##
## outer factor: short_cite (nlvls = 31)
## inner factor: collapse   (nlvls = 52)
##
##             estim    sqrt  fixed
## tau^2      0.1505  0.3879     no
## rho        0.7384             no
##
## Test for Residual Heterogeneity:
## QE(df = 238) = 1067.7800, p-val < .0001
##
## Test of Moderators (coefficient(s) 2,3,4,5,6):
## QM(df = 5) = 230.3356, p-val < .0001
##
## Model Results:
##
##                                    estimate      se     zval    pval
## intrcpt                              0.2482  0.1088   2.2809  0.0226
## condition                            0.5389  0.0490  11.0005  <.0001
## distractor_overlapno                 0.1914  0.1539   1.2435  0.2137
## distractor_overlapnovel             -0.1074  0.1549  -0.6934  0.4881
## condition:distractor_overlapno      -0.2472  0.0775  -3.1910  0.0014
## condition:distractor_overlapnovel    0.1897  0.0935   2.0293  0.0424
##                                      ci.lb    ci.ub
## intrcpt                             0.0349   0.4615    *
## condition                           0.4429   0.6350  ***
## distractor_overlapno               -0.1103   0.4931
## distractor_overlapnovel            -0.4111   0.1962
## condition:distractor_overlapno     -0.3990  -0.0953   **
## condition:distractor_overlapnovel   0.0065   0.3730    *
##
## ---
## 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)`

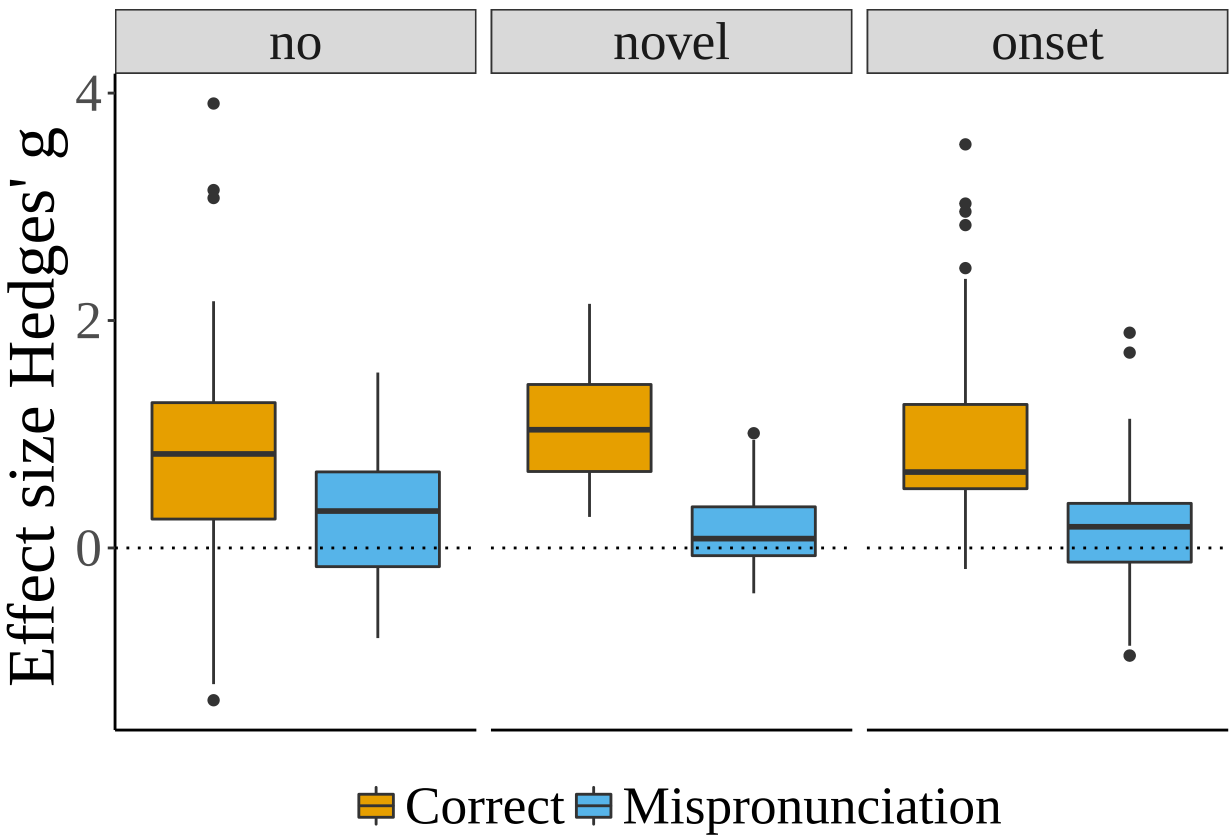
There was a significant interaction between distractor overlap and condition:

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 = 244; method: REML)
##
##    logLik   Deviance        AIC        BIC       AICc
## -233.1772   466.3544   494.3544   542.6087   496.2899
##
## Variance Components:
##
## outer factor: short_cite (nlvls = 31)
## inner factor: collapse   (nlvls = 52)
##
##             estim    sqrt  fixed
## tau^2      0.1490  0.3860     no
## rho        0.7271             no
##
## Test for Residual Heterogeneity:
## QE(df = 232) = 1014.0641, p-val < .0001
##
## Test of Moderators (coefficient(s) 2,3,4,5,6,7,8,9,10,11,12):
## QM(df = 11) = 243.3970, p-val < .0001
##
## Model Results:
##
##                                          estimate      se     zval    pval
## intrcpt                                    0.2570  0.1090   2.3583  0.0184
## age.C                                      0.0202  0.0205   0.9854  0.3244
## condition                                  0.5496  0.0505  10.8924  <.0001
## distractor_overlapno                       0.3051  0.1663   1.8344  0.0666
## distractor_overlapnovel                   -0.1833  0.1767  -1.0374  0.2996
## age.C:condition                            0.0123  0.0128   0.9601  0.3370
## age.C:distractor_overlapno                 0.0224  0.0293   0.7640  0.4449
## age.C:distractor_overlapnovel              0.0093  0.0340   0.2740  0.7841
## condition:distractor_overlapno            -0.3380  0.0882  -3.8317  0.0001
## condition:distractor_overlapnovel          0.2547  0.1025   2.4858  0.0129
## age.C:condition:distractor_overlapno      -0.0408  0.0190  -2.1433  0.0321
## age.C:condition:distractor_overlapnovel   -0.0461  0.0222  -2.0781  0.0377
##                                            ci.lb   ci.ub
## intrcpt                                   0.0434  0.4706    *
## age.C                                    -0.0200  0.0604
## condition                                 0.4507  0.6485  ***
```

```
## distractor_overlapno                      -0.0209   0.6311    .
## distractor_overlapnovel                   -0.5295   0.1630
## age.C:condition                           -0.0128   0.0373
## age.C:distractor_overlapno                -0.0350   0.0798
## age.C:distractor_overlapnovel             -0.0573   0.0759
## condition:distractor_overlapno            -0.5109  -0.1651  ***
## condition:distractor_overlapnovel          0.0539   0.4555    *
## age.C:condition:distractor_overlapno     -0.0781  -0.0035    *
## age.C:condition:distractor_overlapnovel  -0.0896  -0.0026    *
##
## ---
## 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))[11, ], 2)
sum_eff2 <- round(coef(summary(rma_DistractorOverlap))[12, ], 2)
```

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

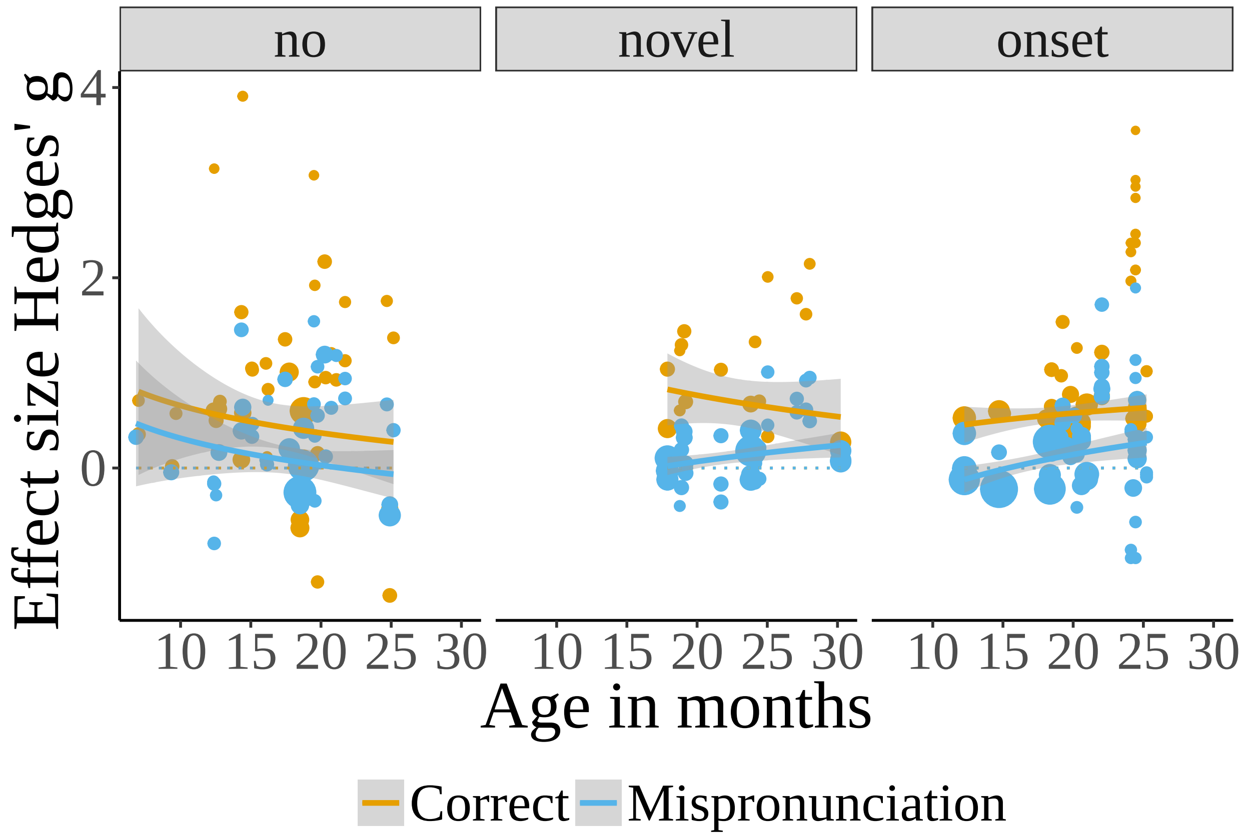
There was a significant interaction between distractor overlap, condition, and age:

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

```r
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
```




```
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

```
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)
```

```r
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

```r
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

```r
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 = 216; method: REML)
##
##    logLik   Deviance        AIC        BIC       AICc
## -229.1763   458.3526   478.3526   511.7280   479.4694
##
## Variance Components:
##
## outer factor: short_cite (nlvls = 26)
## inner factor: collapse   (nlvls = 46)
##
```

```
##             estim    sqrt  fixed
## tau^2      0.1297  0.3601     no
## rho        0.6610             no
##
## Test for Residual Heterogeneity:
## QE(df = 208) = 948.1690, p-val < .0001
##
## Test of Moderators (coefficient(s) 2,3,4,5,6,7,8):
## QM(df = 7) = 153.7950, p-val < .0001
##
## Model Results:
##
##                                     estimate      se     zval    pval
## intrcpt                               0.2615  0.0749   3.4909  0.0005
## type_featurevowel                     0.0274  0.0879   0.3121  0.7550
## condition                             0.4377  0.0458   9.5502  <.0001
## age.C                                 0.0148  0.0142   1.0411  0.2978
## type_featurevowel:condition           0.1489  0.0924   1.6120  0.1070
## type_featurevowel:age.C               0.0026  0.0164   0.1562  0.8758
## condition:age.C                      -0.0167  0.0118  -1.4160  0.1568
## type_featurevowel:condition:age.C     0.0441  0.0183   2.4161  0.0157
##                                       ci.lb   ci.ub
## intrcpt                              0.1147  0.4083  ***
## type_featurevowel                   -0.1449  0.1997
## condition                            0.3479  0.5275  ***
## age.C                               -0.0130  0.0426
## type_featurevowel:condition         -0.0322  0.3301
## type_featurevowel:age.C             -0.0296  0.0348
## condition:age.C                     -0.0398  0.0064
## type_featurevowel:condition:age.C    0.0083  0.0799    *
##
## ---
## 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)`

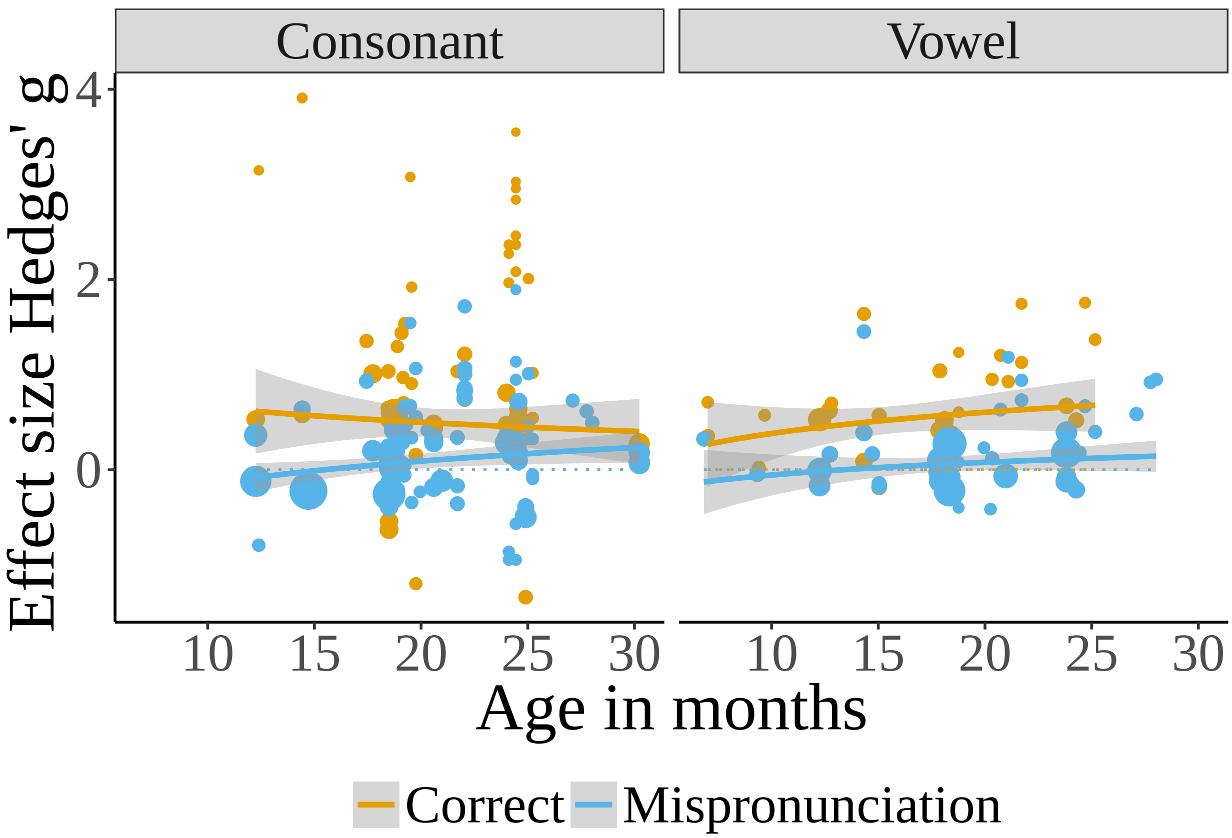
There was a significant interaction between feature type, condition, and age:

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 = 212; method: REML)
##
##    logLik   Deviance        AIC        BIC       AICc
## -225.1962   450.3923   470.3923   503.5735   471.5322
##
## Variance Components:
##
## outer factor: short_cite (nlvls = 25)
## inner factor: collapse   (nlvls = 44)
##
##             estim    sqrt  fixed
## tau^2      0.1380  0.3714     no
## rho        0.5886             no
##
## Test for Residual Heterogeneity:
## QE(df = 204) = 891.0418, p-val < .0001
##
## Test of Moderators (coefficient(s) 2,3,4,5,6,7,8):
## QM(df = 7) = 158.8887, p-val < .0001
##
## Model Results:
##
##                                                 estimate      se     zval
## intrcpt                                           0.2338  0.0814   2.8724
## type_featurevowel                                -0.0151  0.1001  -0.1509
## lang_familyRomanic                               -0.1481  0.2560  -0.5783
## condition                                         0.4021  0.0467   8.6123
## type_featurevowel:lang_familyRomanic              0.5837  0.3134   1.8625
## type_featurevowel:condition                       0.1080  0.0873   1.2375
## lang_familyRomanic:condition                      0.7274  0.2315   3.1428
## type_featurevowel:lang_familyRomanic:condition   -0.8721  0.2801  -3.1136
##                                                   pval    ci.lb    ci.ub
## intrcpt                                         0.0041   0.0743   0.3933
## type_featurevowel                               0.8801  -0.2113   0.1811
## lang_familyRomanic                              0.5630  -0.6498   0.3537
## condition                                       <.0001   0.3106   0.4936
## type_featurevowel:lang_familyRomanic            0.0625  -0.0306   1.1981
## type_featurevowel:condition                     0.2159  -0.0631   0.2792
## lang_familyRomanic:condition                    0.0017   0.2738   1.1811
## type_featurevowel:lang_familyRomanic:condition  0.0018  -1.4210  -0.3231
##
## 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)
```

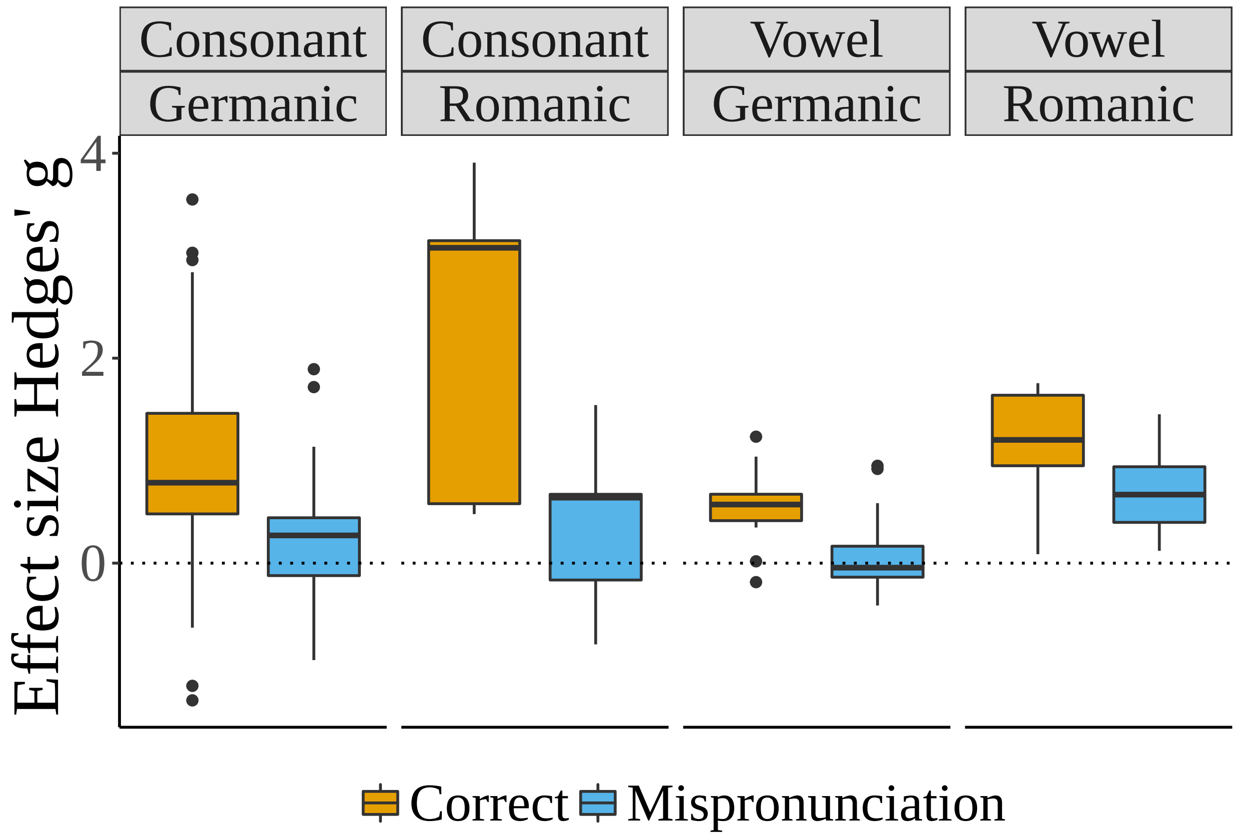
The moderator test was significant: `print(aov.type)`

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
```




```r
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

```r
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 = 212; method: REML)
##
```

```
##    logLik   Deviance        AIC        BIC       AICc
## -213.1082   426.2164   462.2164   521.2224   466.0808
##
## Variance Components:
##
## outer factor: short_cite (nlvls = 25)
## inner factor: collapse   (nlvls = 44)
##
##             estim    sqrt  fixed
## tau^2      0.1125  0.3354     no
## rho        0.7223             no
##
## Test for Residual Heterogeneity:
## QE(df = 196) = 839.0350, 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) = 185.1485, p-val < .0001
##
## Model Results:
##
##                                                        estimate      se
## intrcpt                                                  0.2250  0.0776
## type_featurevowel                                        0.0036  0.0972
## lang_familyRomanic                                       0.8662  0.3923
## condition                                                0.4022  0.0473
## age.C                                                    0.0068  0.0144
## type_featurevowel:lang_familyRomanic                    -0.4231  0.4282
## type_featurevowel:condition                              0.1793  0.1094
## lang_familyRomanic:condition                            -0.5471  0.4057
## type_featurevowel:age.C                                  0.0095  0.0171
## lang_familyRomanic:age.C                                 0.1576  0.0566
## condition:age.C                                         -0.0010  0.0125
## type_featurevowel:lang_familyRomanic:condition           0.4357  0.4414
## type_featurevowel:lang_familyRomanic:age.C              -0.1822  0.0716
## type_featurevowel:condition:age.C                        0.0170  0.0203
## lang_familyRomanic:condition:age.C                      -0.2447  0.0678
## type_featurevowel:lang_familyRomanic:condition:age.C     0.3305  0.0781
##                                                           zval    pval
## intrcpt                                                 2.9002  0.0037
## type_featurevowel                                       0.0370  0.9705
## lang_familyRomanic                                      2.2081  0.0272
## condition                                               8.4967  <.0001
## age.C                                                   0.4727  0.6364
## type_featurevowel:lang_familyRomanic                   -0.9881  0.3231
## type_featurevowel:condition                             1.6379  0.1014
## lang_familyRomanic:condition                           -1.3484  0.1775
## type_featurevowel:age.C                                 0.5568  0.5777
## lang_familyRomanic:age.C                                2.7853  0.0053
## condition:age.C                                        -0.0782  0.9377
## type_featurevowel:lang_familyRomanic:condition          0.9872  0.3235
## type_featurevowel:lang_familyRomanic:age.C             -2.5447  0.0109
## type_featurevowel:condition:age.C                       0.8368  0.4027
## lang_familyRomanic:condition:age.C                     -3.6111  0.0003
## type_featurevowel:lang_familyRomanic:condition:age.C   4.2343  <.0001
```

```
##                                                          ci.lb    ci.ub
## intrcpt                                                 0.0729   0.3771
## type_featurevowel                                      -0.1869   0.1940
## lang_familyRomanic                                      0.0973   1.6350
## condition                                               0.3094   0.4949
## age.C                                                  -0.0214   0.0350
## type_featurevowel:lang_familyRomanic                   -1.2622   0.4161
## type_featurevowel:condition                            -0.0352   0.3938
## lang_familyRomanic:condition                           -1.3423   0.2481
## type_featurevowel:age.C                                -0.0240   0.0430
## lang_familyRomanic:age.C                                0.0467   0.2685
## condition:age.C                                        -0.0255   0.0235
## type_featurevowel:lang_familyRomanic:condition         -0.4293   1.3008
## type_featurevowel:lang_familyRomanic:age.C             -0.3226  -0.0419
## type_featurevowel:condition:age.C                      -0.0228   0.0568
## lang_familyRomanic:condition:age.C                     -0.3775  -0.1119
## type_featurevowel:lang_familyRomanic:condition:age.C    0.1775   0.4835
##
## 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
```

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

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

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

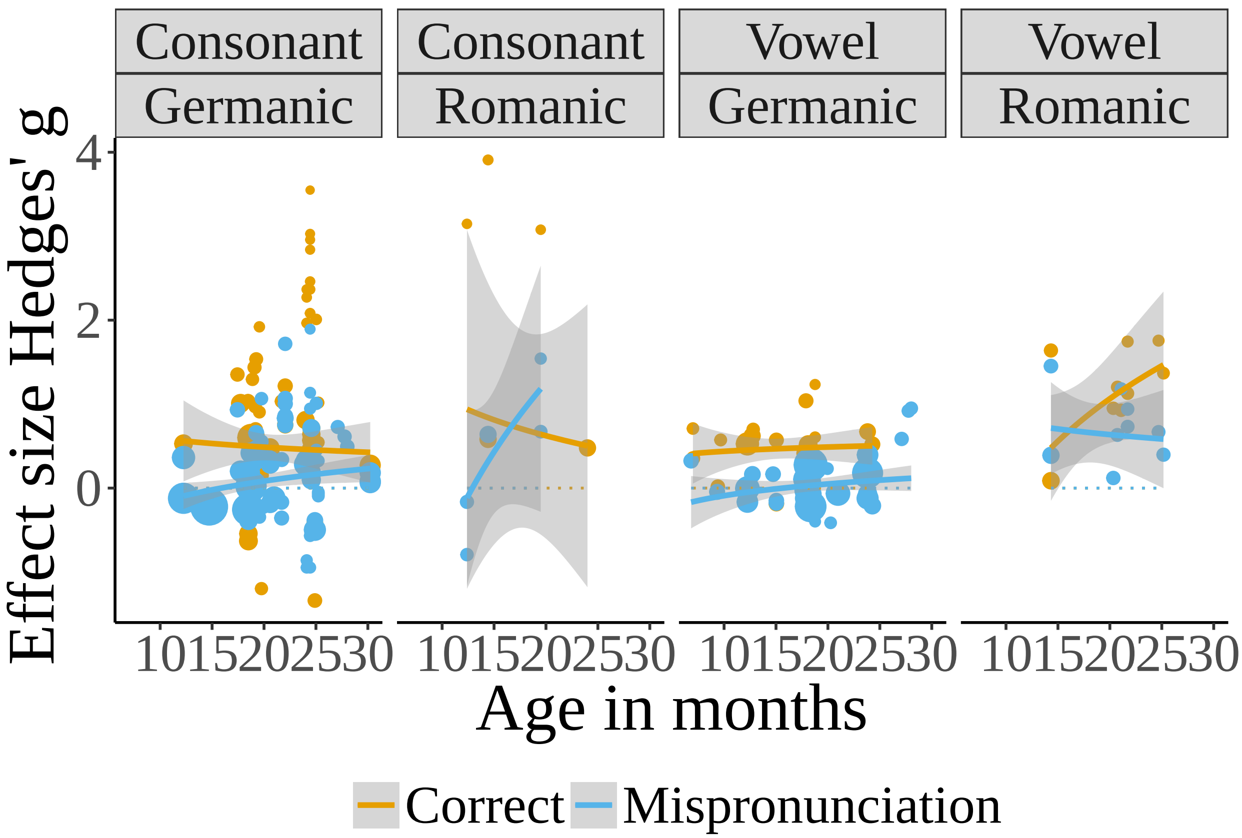
There was a significant interaction between feature type, age, and language family, and condition:

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

```
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

```
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

```
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 = 137; method: REML)
##
##   logLik  Deviance       AIC       BIC      AICc
## -62.1536  124.3072  136.3072  153.6493  136.9739
##
## Variance Components:
##
## outer factor: short_cite (nlvls = 30)
## inner factor: collapse   (nlvls = 50)
##
##             estim    sqrt  fixed
## tau^2      0.1171  0.3423     no
## rho        0.6753             no
##
## Test for Residual Heterogeneity:
## QE(df = 133) = 405.3443, p-val < .0001
##
## Test of Moderators (coefficient(s) 2,3,4):
## QM(df = 3) = 9.0204, p-val = 0.0290
##
## Model Results:
##
##                           estimate      se    zval    pval    ci.lb
## intrcpt                     0.2251  0.0677  3.3256  0.0009   0.0924
## age.C                       0.0072  0.0124  0.5849  0.5586  -0.0170
## lang_familyRomanic          0.4695  0.1948  2.4104  0.0159   0.0877
```

```
## age.C:lang_familyRomanic    0.0714  0.0362  1.9725  0.0486   0.0005
##                             ci.ub
## intrcpt                    0.3578  ***
## age.C                      0.0315
## lang_familyRomanic         0.8512    *
## age.C:lang_familyRomanic  0.1423    *
##
## ---
## 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)
```

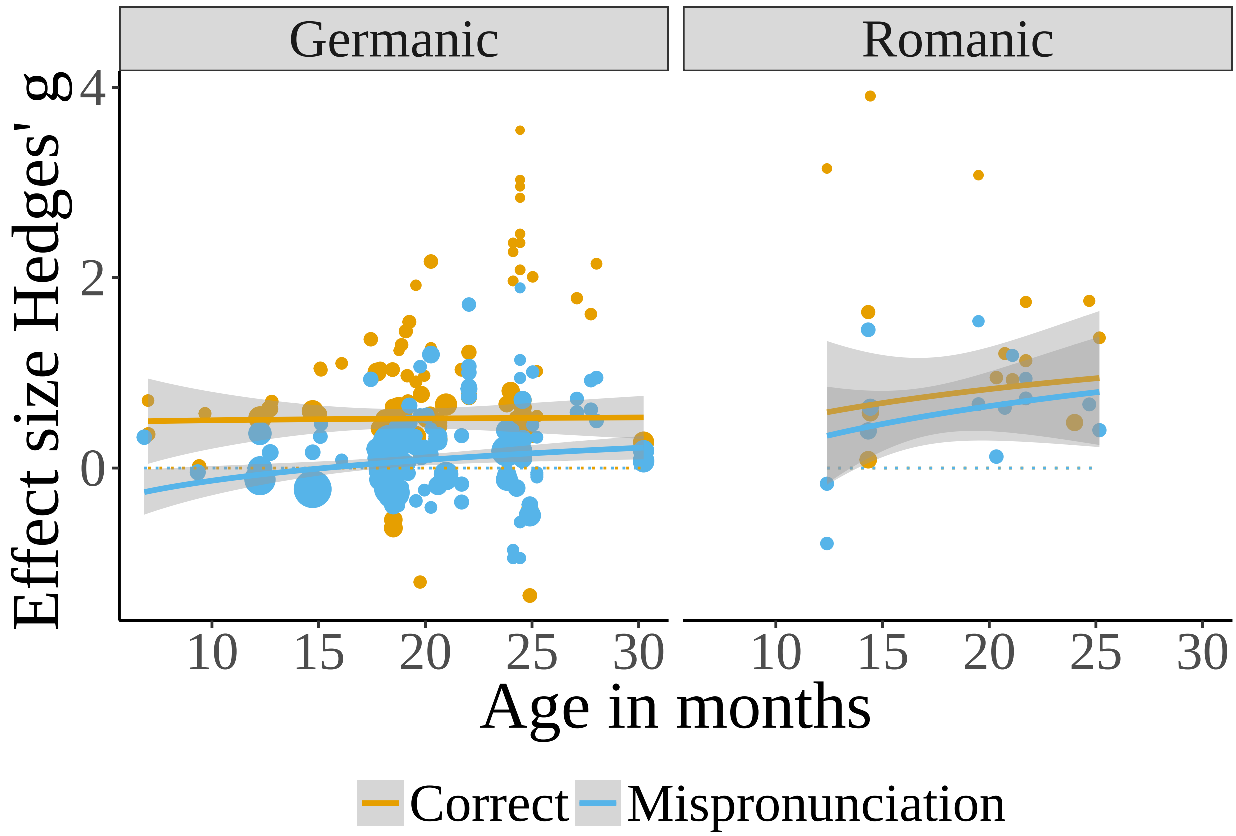
The moderator test was significant: `print(aov.type)`

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)`)