```
# References: Baek et al., 2016; Moeyaert et al., 2020; Pustejovsky et al.,
2014
# References: Hedges et al., 2010; Tipton, 2015; Tipton & Pustejovsky, 2015)

# install.packages("nlme")
# install.packages("dplyr")
# install.packages("scdhlm")
# install.packages("metafor")
# install.packages("clubsandwich")

library(nlme)
library(dplyr)
library(scdhlm)
library(metafor)
library(clubSandwich)



# Calculate design-comparable standardized mean differences ----
data = read.csv("Burns_2005.csv")
data

#create phase-by-time interaction
data$phase_time <- with(data,
                        unlist(tapply((phase == "1") * time,
                                      list(phase, case),
                                      function(x)
                                          x - min(x))))
data #Displays data

#Two-level model: varying intercepts, no trend, varying treatment effect,
varying phase-by-time interaction
ctrl <- lmeControl(opt = 'optim')
two.level <- lme(
    fixed = outcome ~ phase + phase_time,
    random = ~ phase + phase_time | case,
    correlation = corAR1(0, ~ time |
                             case),
    control = ctrl,
    data = data,
    method = "REML"
)
summary(two.level)

#Time-point constants
phase_0 <- data %>%
    filter(phase == 0) %>%
    group_by(case) %>%
    mutate(max_session_on_case = max(session))

A <- min(phase_0$max_session_on_case)

phase_1 <- data %>%
    filter(phase == 1) %>%
```

```
    group_by(case) %>%
    mutate(max_session_on_case = max(session))

B <- min(phase_1$max_session_on_case)

#Center at follow-up time
Center <- B
data$time <- data$time - Center

Burns_2005_g <-
    g_mlm(two.level,
          p_const = c(0, 1, B - A),
          r_const = c(1, 0, 0, 0, 0, 0, 0, 1))
summary(Burns_2005_g)
print(Burns_2005_g)


# Conduct meta-analysis for effect sizes ----
data = read.csv("all.data.csv")
data

# Construct approximate V matrix
# assuming constant sampling correlation of r = 0.6 for the sampling errors
V <- impute_covariance_matrix(
    data$var.eff.size,
    cluster = data$study.id,
    r = 0.6,
    smooth_vi = TRUE
)

# Fit multilevel random effects model to examine the overall effect size
es <- rma.mv(
    effect.size ~ 1,
    V,
    random = ~ 1 | study.id / effect.size.id,
    data = data,
    sparse = TRUE,
    verbose = FALSE
)
es

# Examine robust variance estimation standard errors
CI_es <- conf_int(es, vcov = "CR2", cluster = data$study.id)
CI_es

p_es <- coef_test(es, vcov = "CR2", cluster = data$study.id)
p_es

# Funnel plot - Publication bias ----
funnel(es,
       xlab = "Effect Sizes",
       las = 1,
       digits = list(1L, 2))
ranktest(data$effect.size, data$var.eff.size)
```

```
# Create forest plot ----
forest(
    es,
    at = (c(-15,-10,-5, 5, 10, 15)),
    cex = .75,
    header = TRUE,
    xlim = c(-55, 29),
    mlab = "",
    slab = paste(data$study, data$crit.cat, sep = ", "),
    ilab = data$num.component,
    ilab.xpos = -16,
    order=data$effect.size
)

text((-16), 19, "Number of Instructional Components", cex = 0.72)

text(-28,-1, pos = 2, cex = 0.75, bquote(paste(
    "RE Model (Q = ",
    .(formatC(
        es$QE, digits = 2, format = "f"
    )),
    ", df = ",
    .(es$k - es$p),
    ", p = ",
    .(formatC(
        es$QEp, digits = 3, format = "f"
    )),
    ")"
)))


text(29,-2, pos = 2, cex = 0.75, bquote(paste(
    "RVE = 3.74 [ 2.31,  5.18]"
)))


# Examine by math topics
es_topic <- rma.mv(
    effect.size ~ 0 + math.topic,
    V,
    random = ~ 1 | study.id / effect.size.id,
    data = data,
    sparse = TRUE,
    verbose = FALSE
)
es_topic

# Examine robust variance estimation standard errors
CI_topic <-
    conf_int(es_topic, vcov = "CR2", cluster = data$study.id)
CI_topic

p_topic <-
```

```
    coef_test(es_topic, vcov = "CR2", cluster = data$study.id)
p_topic

wald_topic <- Wald_test(es_topic,
                        constraints = constrain_equal(1:4),
                        vcov = "CR2")
wald_topic

# Addition as a reference group
es_topic_reg <-
    rma.mv(
        effect.size ~ addition_subtraction + multiplication + subtraction,
        V,
        random = ~ 1 | study.id / effect.size.id,
        data = data,
        sparse = TRUE,
        verbose = FALSE
    )
es_topic_reg

# Examine robust variance estimation standard errors
CI_topic_reg <-
    conf_int(es_topic_reg, vcov = "CR2", cluster = data$study.id)
CI_topic_reg

p_topic_reg <-
    coef_test(es_topic_reg, vcov = "CR2", cluster = data$study.id)
p_topic_reg

# Examine by whole number computation measures
es_measure <- rma.mv(
    effect.size ~ 0 + math.measure,
    V,
    random = ~ 1 | study.id / effect.size.id,
    data = data,
    sparse = TRUE,
    verbose = FALSE
)
es_measure

# Examine robust variance estimation standard errors
CI_measure <-
    conf_int(es_measure, vcov = "CR2", cluster = data$study.id)
CI_measure

p_measure <-
    coef_test(es_measure, vcov = "CR2", cluster = data$study.id)
p_measure

wald_measure <- Wald_test(es_measure,
                          constraints = constrain_equal(1:2),
                          vcov = "CR2")
wald_measure
```

```
wald_measure <- Wald_test(es,
                          constraints = constrain_equal(1:2),
                          vcov = "CR2")
wald_measure


# Accuracy as a reference group
es_measure_reg <- rma.mv(
    effect.size ~ fluency,
    V,
    random = ~ 1 | study.id / effect.size.id,
    data = data,
    sparse = TRUE,
    verbose = FALSE
)
es_measure_reg

# Examine robust variance estimation standard errors
CI_measure_reg <-
    conf_int(es_measure_reg, vcov = "CR2", cluster = data$study.id)
CI_measure_reg

p_measure_reg <-
    coef_test(es_measure_reg, vcov = "CR2", cluster = data$study.id)
p_measure_reg

# Examine by instructional components--review
es_comp_review <- rma.mv(
    effect.size ~ 0 + review,
    V,
    random = ~ 1 | study.id / effect.size.id,
    data = data,
    sparse = TRUE,
    verbose = FALSE
)
es_comp_review

# Examine robust variance estimation standard errors
CI_comp_review <-
    conf_int(es_comp_review,
             vcov = "CR2",
             cluster = data$study.id)
CI_comp_review

p_comp_review <-
    coef_test(es_comp_review,
              vcov = "CR2",
              cluster = data$study.id)
p_comp_review

table(data$review)

# Examine by instructional components--modeling
```

```
es_comp_modeling <- rma.mv(
    effect.size ~ 0 + modeling,
    V,
    random = ~ 1 | study.id / effect.size.id,
    data = data,
    sparse = TRUE,
    verbose = FALSE
)
es_comp_modeling

# Examine robust variance estimation standard errors
CI_comp_modeling <-
    conf_int(es_comp_modeling,
             vcov = "CR2",
             cluster = data$study.id)
CI_comp_modeling

p_comp_modeling <-
    coef_test(es_comp_modeling,
              vcov = "CR2",
              cluster = data$study.id)
p_comp_modeling

table(data$modeling)

# Examine by instructional components--guidedPractice
es_comp_guidedPractice <-
    rma.mv(
        effect.size ~ 0 + guidedPractice,
        V,
        random = ~ 1 | study.id / effect.size.id,
        data = data,
        sparse = TRUE,
        verbose = FALSE
    )
es_comp_guidedPractice

# Examine robust variance estimation standard errors
CI_comp_guidedPractice <-
    conf_int(es_comp_guidedPractice,
             vcov = "CR2",
             cluster = data$study.id)
CI_comp_guidedPractice

p_comp_guidedPractice <-
    coef_test(es_comp_guidedPractice,
              vcov = "CR2",
              cluster = data$study.id)
p_comp_guidedPractice

table(data$guidedPractice)

# Examine by instructional components--engage
es_comp_engage <- rma.mv(
```

```
    effect.size ~ 0 + engage,
    V,
    random = ~ 1 | study.id / effect.size.id,
    data = data,
    sparse = TRUE,
    verbose = FALSE
)
es_comp_engage

# Examine robust variance estimation standard errors
CI_comp_engage <-
    conf_int(es_comp_engage,
             vcov = "CR2",
             cluster = data$study.id)
CI_comp_engage

p_comp_engage <-
    coef_test(es_comp_engage,
              vcov = "CR2",
              cluster = data$study.id)
p_comp_engage

table(data$engage)

# Examine by instructional components--visual
es_comp_visual <- rma.mv(
    effect.size ~ 0 + visual,
    V,
    random = ~ 1 | study.id / effect.size.id,
    data = data,
    sparse = TRUE,
    verbose = FALSE
)
es_comp_visual

# Examine robust variance estimation standard errors
CI_comp_visual <-
    conf_int(es_comp_visual,
             vcov = "CR2",
             cluster = data$study.id)
CI_comp_visual

p_comp_visual <-
    coef_test(es_comp_visual,
              vcov = "CR2",
              cluster = data$study.id)
p_comp_visual

table(data$visual)

# Examine by instructional components--strategyInstruct
es_comp_strategyInstruct <-
    rma.mv(
        effect.size ~ 0 + strategyInstruct,
```

```
        V,
        random = ~ 1 | study.id / effect.size.id,
        data = data,
        sparse = TRUE,
        verbose = FALSE
    )
es_comp_strategyInstruct

# Examine robust variance estimation standard errors
CI_comp_strategyInstruct <-
    conf_int(es_comp_strategyInstruct,
             vcov = "CR2",
             cluster = data$study.id)
CI_comp_strategyInstruct

p_comp_strategyInstruct <-
    coef_test(es_comp_strategyInstruct,
              vcov = "CR2",
              cluster = data$study.id)
p_comp_strategyInstruct

table(data$strategyInstruct)

# Examine by instructional components--timedPractice
es_comp_timedPractice <-
    rma.mv(
        effect.size ~ 0 + timedPractice,
        V,
        random = ~ 1 | study.id / effect.size.id,
        data = data,
        sparse = TRUE,
        verbose = FALSE
    )
es_comp_timedPractice

# Examine robust variance estimation standard errors
CI_comp_timedPractice <-
    conf_int(es_comp_timedPractice,
             vcov = "CR2",
             cluster = data$study.id)
CI_comp_timedPractice

p_comp_timedPractice <-
    coef_test(es_comp_timedPractice,
              vcov = "CR2",
              cluster = data$study.id)
p_comp_timedPractice

table(data$timedPractice)

# Examine by number of instructional components
es_num.component <- rma.mv(
    effect.size ~ num.component.c,
    V,
```

```
    random = ~ 1 | study.id / effect.size.id,
    data = data,
    sparse = TRUE,
    verbose = FALSE
)
es_num.component

# Examine robust variance estimation standard errors
CI_num.component <-
    conf_int(es_num.component,
             vcov = "CR2",
             cluster = data$study.id)
CI_num.component

p_num.component <-
    coef_test(es_num.component,
              vcov = "CR2",
              cluster = data$study.id)
p_num.component
```