```
# Reference ----
# Last updated March 10, 2021

# Install packages used in this script. ----
install.packages("readxl")
install.packages("nlme")
install.packages("msm")
install.packages("lattice")
install.packages("metafor")

# Import the data set from the working directory. ----
library(readxl)
    VM <- read_excel(path = "../OSF/VM.xlsx")
VM # Display data.

# Model.1 ----
# Fixed and varying baseline level, immediate effect, and trend during
intervention
# Model heterogeneous autocorrelation (first-order autoregressive structure)
and within-case variance.
# No moderators
library(nlme)
Model.1 <- lme(Outcome ~ 1 + Intervention + Intervention_Time,
               random  = ~ 1 + Intervention + Intervention_Time | Study/Case,
               data = VM,
               correlation = corAR1(form = ~ 1 | Study/Case/Intervention),
               weights = varIdent(form = ~ 1 | Intervention),
               method = "REML",
               na.action = "na.omit",
               control = list(opt = "optim"))

# Output for Model.1.
summary(Model.1)
# Obtain Variance components
VarCorr(Model.1)
# Obtain 95% confidence intervals for estimates
intervals(Model.1)
# Calculate standard errors for variance components in the standard deviation
scale.
var1 <-Model.1$apVar
var1
par1 <- attr(var1, "Pars")
par1
vc1  <- exp(par1)^2
vc1
library(msm)
deltamethod (~ exp(x2)^2, par1, var1)
deltamethod (~ exp(x3)^2, par1, var1)
deltamethod (~ exp(x8)^2, par1, var1)
deltamethod (~ exp(x9)^2, par1, var1)
se.vec1 <- c()
for (i in 1:length(par1)){form <- formula(paste(" ~ exp(x",i,")^2", sep = ""))
se.vec1 <- c(se.vec1,deltamethod (form, par1, var1))}
se.vec1
```

```
# Model.2 ----
# Fixed and varying Fixed and varying baseline level, immediate effect, and
trend during intervention
# Model heterogeneous autocorrelation (first-order autoregressive structure)
and within-case variance.
# Add moderators, affecting immediate effects.
Model.2 <- lme(Outcome ~ 1 + Intervention + Intervention_Time +
                   # case-level (student characteristics) moderators
                   Middle*Intervention + High*Intervention +
                   ID*Intervention + ASD*Intervention + EBD*Intervention +
OHI*Intervention +
                   # study-level (intervention features) moderators
                   Devise.use*Intervention + Devise.use.instruct*Intervention
+
                   Teacher.guided*Intervention + Teacher.led*Intervention +
                   Commercial*Intervention +
                   Computer*Intervention +
                   Single.represent*Intervention + Tutorial*Intervention +
Game*Intervention +
                   Area*Intervention + Linear*Intervention +
Base.ten*Intervention + Algebra*Intervention + Multi.model*Intervention,
               random  = ~ 1 + Intervention + Intervention_Time | Study/Case,
               data = VM,
               correlation = corAR1(form = ~ 1 | Study/Case/Intervention),
               weights = varIdent(form = ~ 1 | Intervention),
               method = "REML",
               na.action = "na.omit",
               control = list(opt = "optim"))

# Obtain the output for model.2.
summary(Model.2)
# Obtain variance components.
VarCorr(Model.2)
# Obtain 95% confidence intervals for estimates.
intervals(Model.2)
# Calculate standard errors for variance components in the standard deviation
scale.
var2 <- Model.2$apVar
var2
par2 <- attr(var2, "Pars")
par2
vc2  <- exp(par2)^2
vc2
deltamethod (~ exp(x2)^2, par2, var2)
deltamethod (~ exp(x3)^2, par2, var2)
deltamethod (~ exp(x8)^2, par2, var2)
deltamethod (~ exp(x9)^2, par2, var2)
se.vec2 <- c()
for (i in 1:length(par2)){form <- formula(paste(" ~ exp(x",i,")^2", sep = ""))
se.vec2 <- c(se.vec2,deltamethod (form, par2, var2))}
se.vec2

# Model.3 ----
# Fixed and varying Fixed and varying baseline level, immediate effect, and
trend during intervention
# Model heterogeneous autocorrelation (first-order autoregressive structure)
and within-case variance.
# Add moderators, affecting trends during the intervention.
Model.3 <- lme(Outcome ~ 1 + Intervention + Intervention_Time +
                   # case-level (student characteristics) moderators
                   Middle*Intervention_Time + High*Intervention_Time +
                   ID*Intervention_Time + ASD*Intervention_Time +
EBD*Intervention_Time + OHI*Intervention_Time +
                   # study-level (intervention features) moderators
                   Devise.use*Intervention_Time +
Devise.use.instruct*Intervention_Time +
                   Teacher.guided*Intervention_Time +
Teacher.led*Intervention_Time +
                   Commercial*Intervention_Time +
                   Computer*Intervention_Time +
                   Single.represent*Intervention_Time +
Tutorial*Intervention_Time + Game*Intervention_Time +
                   Area*Intervention_Time + Linear*Intervention_Time +
Base.ten*Intervention_Time + Algebra*Intervention_Time +
Multi.model*Intervention_Time,
               random  = ~ 1 + Intervention + Intervention_Time | Study/Case,
               data = VM,
               correlation = corAR1(form = ~ 1 | Study/Case/Intervention),
               weights = varIdent(form = ~ 1 | Intervention),
               method = "REML",
               na.action = "na.omit",
               control = list(opt = "optim"))

# Obtain the output for model.3.
summary(Model.3)
# Obtain variance components.
VarCorr(Model.3)
# Obtain 95% confidence intervals for estimates.
intervals(Model.3)
# Calculate standard errors for variance components in the standard deviation
scale.
var3 <- Model.3$apVar
var3
par3 <- attr(var3, "Pars")
par3
vc3  <- exp(par3)^2
vc3
deltamethod (~ exp(x2)^2, par3, var3)
deltamethod (~ exp(x3)^2, par3, var3)
deltamethod (~ exp(x8)^2, par3, var3)
deltamethod (~ exp(x9)^2, par3, var3)
se.vec3 <- c()
for (i in 1:length(par3)){form <- formula(paste(" ~ exp(x",i,")^2",sep = ""))
se.vec3 <- c(se.vec3,deltamethod (form, par3, var3))}
se.vec3

# Plot random effects ----
```

```
library("lattice")

plot(ranef(Model.1, level = 1))
plot(ranef(Model.1, level = 2))

# Publication Bias ----
library(readxl)
library(metafor)

Interv <- read_excel(path = "../OSF/Interv.xlsx")
Interv # Display data.

res <- rma(yi=Intervention, sei=Standard_Error, data=Interv)
funnel(res, xlab="Immediate Effect", ylab="Standard Errors")
regtest(res, model="lm", ret.fit=TRUE)

Interv.Time <- read_excel(path = "../OSF/Interv.Time.xlsx")
Interv.Time # Display data.

res <- rma(yi=Intervention_Time, sei=Standard_Error, data=Interv.Time)
funnel(res, xlab="Trends During the Intervention Phase", ylab="Standard
Errors")
regtest(res, model="lm", ret.fit=TRUE)
```