

Supplemental File for “Good statistical practices for contemporary meta-analysis: examples based on a systematic review on COVID-19 in pregnancy”

This supplemental file presents the code for producing the results presented in the main content. The frequentist methods were based on the R package “meta” (version 4.18-1). The Bayesian methods are based on the R package “rjags” (version 4-10). Users need to install the JAGS software (freely available at <https://mcmc-jags.sourceforge.io/>) before implementing the Bayesian methods. JAGS version 4.3.0 for Windows was used in our analyses.

Before running the following code, users need to load the two R packages:

```
library("meta")  
library("rjags")
```

R code for performing the meta-analysis on fever

```
fever <- data.frame(
  ai = c(8, 17, 5, 15, 8, 11, 3, 22, 48, 51, 3328),
  nil = c(21, 31, 28, 31, 17, 30, 11, 34, 83, 140, 17385),
  ci = c(14, 30, 29, 49, 18, 28, 15, 26, 63, 1159, 68536),
  ni0 = c(19, 35, 54, 80, 26, 42, 25, 30, 107, 2515, 219580))

# The DL method with prediction interval
fever.DL <- metabin(ai, nil, ci, ni0, data = fever, sm = "OR", prediction = TRUE)
fever.DL

# The REML method with prediction interval
fever.REML <- metabin(ai, nil, ci, ni0, data = fever, sm = "OR", method.tau = "REML",
  prediction = TRUE)
fever.REML

# t-based CI
update(fever.DL, hakn = TRUE)
update(fever.REML, hakn = TRUE)

# Contour-enhanced funnel plot
pdf("funnel-fever.pdf", width = 4.5, height = 4.5)
par(mar = c(3, 3, 0, 0) + 0.1, mgp = c(2, 1, 0))
funnel(fever.DL, col.fixed = "black", col.random = "blue", lwd = 2, level = 0.95,
  contour = c(0.9, 0.95, 0.99), col.contour = c("lightgray", "grey", "darkgray"),
  ref = 1, xlab = "Odds ratio", ylab = "Standard error", pch = 16)
legend("topleft", c("0.1 < P < 1", "0.05 < P < 0.1", "0.01 < P < 0.05", "P < 0.01"),
  fill = c("white", "lightgray", "grey", "darkgray"), bg = "white", cex = 0.8)
dev.off()

# Test for small-study effects
metabias(fever.DL, method.bias = "Egger")
```

R code for performing the meta-analysis on cough

```
cough <- data.frame(
  ai = c(6, 15, 7, 14, 9, 5, 22, 4, 65, 116, 5230),
  nil = c(21, 31, 28, 31, 17, 30, 34, 6, 83, 140, 17385),
  ci = c(8, 16, 32, 48, 12, 21, 23, 3, 76, 1979, 89422),
  ni0 = c(19, 35, 54, 80, 26, 42, 30, 5, 107, 2515, 219580))

# The DL method with prediction interval
cough.DL <- metabin(ai, nil, ci, ni0, data = cough, sm = "OR", prediction = TRUE)
cough.DL

# Forest plot with prediction interval
pdf("forest-cough-DL.pdf", width = 10, height = 5)
forest(cough.DL, prediction = TRUE, comb.fixed = FALSE,
  text.random = "Random-effects model", rightlabs = c("OR", "95% CI", "Weight"),
  smlab = "", xlab = "Odds ratio")
dev.off()

# REML method with prediction interval
cough.REML <- metabin(ai, nil, ci, ni0, data = cough, sm = "OR", method.tau = "REML",
  prediction = TRUE)
cough.REML

# t-based CI
update(cough.DL, hakn = TRUE)
update(cough.REML, hakn = TRUE)

# Contour-enhanced funnel plot
pdf("funnel-cough.pdf", width = 4.5, height = 4.5)
par(mar = c(3, 3, 0, 0) + 0.1, mgp = c(2, 1, 0))
funnel(cough.DL, col.fixed = "black", col.random = "blue", lwd = 2, level = 0.95,
  contour = c(0.9, 0.95, 0.99), col.contour = c("lightgray", "grey", "darkgray"),
  ref = 1, xlab = "Odds ratio", ylab = "Standard error", pch = 16)
legend("topleft", c("0.1 < P < 1", "0.05 < P < 0.1", "0.01 < P < 0.05", "P < 0.01"),
  fill = c("white", "lightgray", "grey", "darkgray"), bg = "white", cex = 0.8)
dev.off()

# GLMM for meta-analysis
cough.glmm <- metabin(ai, nil, ci, ni0, data = cough, sm = "OR", prediction = TRUE,
  method = "GLMM", model.glmm = "UM.RS", nAGQ = 1)
cough.glmm

# Bayesian analysis with uniform prior for the between-study standard deviation
ma.OR.uni <- function(){
  out <- "model{
    for(i in 1:NS){
      theta[i,1] <- 0
      mu[i] ~ dnorm(0, 0.0001)
      for(k in 1:2){
        e[i,k] ~ dbin(p[i,k], n[i,k])
        logit(p[i,k]) <- mu[i] + theta[i,k]
      }
      theta[i,2] ~ dnorm(lor, prec)
    }
    lor ~ dnorm(0, 0.0001)
    prec <- 1/tau^2
    tau ~ dunif(0, 5)
    OR <- exp(lor)
  }"
}
dat.cough <- list(NS = 11, e = cbind(cough$ci, cough$ai),
  n = cbind(cough$ni0, cough$nil))
```

```

set.seed(1234)
inits <- list(
  list(lor = 0, mu = rep(0, 11), .RNG.name = "base::Wichmann-Hill", .RNG.seed = 1234),
  list(lor = -1, mu = rep(1, 11), .RNG.name = "base::Wichmann-Hill",
    .RNG.seed = 12345),
  list(lor = 1, mu = rep(-1, 11), .RNG.name = "base::Wichmann-Hill",
    .RNG.seed = 123456))
jags.ma.cough.uni <- jags.model(file = textConnection(ma.OR.uni()), data = dat.cough,
  n.chains = 3, n.adapt = 1000, inits = inits)
update(jags.ma.cough.uni, n.iter = 20000)
coda.ma.cough.uni <- coda.samples(model = jags.ma.cough.uni,
  variable.names = c("OR", "tau"), n.iter = 50000, thin = 2)
smry.ma.cough.uni <- summary(coda.ma.cough.uni)
smry.ma.cough.uni$quantiles[, c("2.5%", "50%", "97.5%")]

# Bayesian analysis with informative prior for the between-study variance
ma.OR.lognorm <- function(){
  out <- "model{
    for(i in 1:NS){
      theta[i,1] <- 0
      mu[i] ~ dnorm(0, 0.0001)
      for(k in 1:2){
        e[i,k] ~ dbin(p[i,k], n[i,k])
        logit(p[i,k]) <- mu[i] + theta[i,k]
      }
      theta[i,2] ~ dnorm(lor, prec)
    }
    lor ~ dnorm(0, 0.0001)
    prec <- 1/tau.squared
    tau.squared ~ dlnorm(mu.tau, prec.tau)
    mu.tau <- -2.89
    prec.tau <- 1/1.91^2
    OR <- exp(lor)
    tau <- sqrt(tau.squared)
  }"
}
set.seed(1234)
jags.ma.cough.lognorm <- jags.model(file = textConnection(ma.OR.lognorm()),
  data = dat.cough, n.chains = 3, n.adapt = 1000, inits = inits)
update(jags.ma.cough.lognorm, n.iter = 20000)
coda.ma.cough.lognorm <- coda.samples(model = jags.ma.cough.lognorm,
  variable.names = c("OR", "tau"), n.iter = 50000, thin = 2)
smry.ma.cough.lognorm <- summary(coda.ma.cough.lognorm)
smry.ma.cough.lognorm$quantiles[, c("2.5%", "50%", "97.5%")]

```

R code for performing the meta-analysis of the prevalence of COVID-19 in pregnant women

```
prev <- data.frame(
  e = c(32, 3, 9, 3, 1, 13, 23, 12, 10, 24, 20, 5, 2, 30, 23, 125, 17, 71, 2, 19,
        10, 49, 7, 98, 23, 2, 141, 116, 29, 11, 100, 101, 46, 8, 7, 3, 500, 15, 2, 1,
        5, 30, 25, 20, 77, 28, 82, 2, 338, 6, 25, 156, 141, 70, 15, 284, 37, 53, 86, 138),
  n = c(161, 139, 129, 533, 82, 470, 635, 103, 75, 158,
        757, 188, 52, 1055, 225, 874, 3923, 382, 428, 1718,
        492, 1566, 178, 1293, 296, 14, 977, 1481, 316, 114,
        673, 1418, 124, 477, 333, 75, 4674, 79, 473, 180,
        229, 1313, 366, 249, 935, 315, 601, 433, 16308, 226,
        532, 2682, 1140, 495, 100, 1671, 583, 813, 355, 478))

# Meta-analysis using the DL method
prev.DL <- metaprop(e, n, data = prev, sm = "PFT", method.ci = "NAsm")
prev.DL

# Forest plot
pdf("forest-prev-DL.pdf", width = 12, height = 15)
forest(prev.DL, pscale = 100, comb.fixed = FALSE,
       text.random = "Random-effects model",
       rightlabs = c("Prevalence (%)", "95% CI", "Weight"), smlab = "",
       xlab = "Prevalence (%)")
dev.off()

# GLMM for meta-analysis
prev.glmm <- metaprop(e, n, data = prev, sm = "PLOGIT", method = "GLMM",
                     method.ci = "NAsm")
prev.glmm

# Bayesian analysis with uniform prior for the between-study standard deviation
ma.prev.uni <- function(){
  out <- "model{
    for(i in 1:NS){
      e[i] ~ dbin(p[i], n[i])
      logit(p[i]) <- mu + theta[i]
      theta[i] ~ dnorm(0, prec)
    }
    mu ~ dnorm(0, 0.0001)
    prec <- 1/tau^2
    tau ~ dunif(0, 5)
    prev <- exp(mu)/(1+exp(mu))
  }"
}
dat.prev <- list(NS = 60, e = prev$e, n = prev$n)
set.seed(1234)
inits.prop <- list(
  list(mu = 0, .RNG.name = "base::Wichmann-Hill", .RNG.seed = 1234),
  list(mu = 1, .RNG.name = "base::Wichmann-Hill", .RNG.seed = 12345),
  list(mu = -1, .RNG.name = "base::Wichmann-Hill", .RNG.seed = 123456))
jags.ma.prev.uni <- jags.model(file = textConnection(ma.prev.uni()),
                             data = dat.prev, n.chains = 3, n.adapt = 1000, inits = inits.prop)
update(jags.ma.prev.uni, n.iter = 20000)
coda.ma.prev.uni <- coda.samples(model = jags.ma.prev.uni,
                                variable.names = c("mu", "prev", "tau"), n.iter = 50000, thin = 2)
smry.ma.prev.uni <- summary(coda.ma.prev.uni)
smry.ma.prev.uni$quantiles[, c("2.5%", "50%", "97.5%")]
```