---

title: "Dirofilaria Meta-Analysis Middle East and North Africa -Subset1 Dataset"

author: "Katie Izenour"

date: "2024-08-18

output: word\_document

---

```{r setup, include=FALSE}

knitr::opts\_chunk$set(echo = TRUE)

```

```{r}

# Importing the dataset

#---- Libraries -----

library(readxl)

library(metasens)

library(metafor)

library(meta)

library(tidyverse)

library(rmarkdown)

if (!require("devtools")) {

install.packages("devtools")

}

#---- Import Statement -----

# Use tab 'Subset1' from the datafile. This tab is from the 'Full' tab but has has 12 rows deleted (remaining N=132) as follows:

## Publication #4, 2 rows of the sub-samples from the original population

## Publication #107, 1 row of a sub-sample from the original population

## Publication #106, 1 row, the test result for "D.immitis and D.repens", that value only occurs one time and isn't suitable for ## # moderator analysis

## Publication #119, 1 row, Only paper with 'Adult worms' as the substrate

## Publication #83 2 rows, only paper that used an antibody test

## Publication #120, 1 row, diagnostic test 'unknown'

## Publication #85, 4 rows only, only paper reporting on Jackals and foxes

**subset1 <- read\_excel("C:/Users/User/XXXYourFilePathXXX.xlsx", sheet='subset1')**

# ---- Data Cleaning ----

# Takes all character variables and makes factors

**subset1[sapply(subset1, is.character)] = lapply(subset1[sapply(subset1, is.character)], as.factor)**

**str(subset1) #Checking the structure**

```

tibble [132 × 23] (S3: tbl\_df/tbl/data.frame)

$ recordID : num [1:132] 1 2 3 4 5 6 7 8 9 10 ...

$ publicationID : num [1:132] 2 4 5 8 8 8 17 17 17 17 ...

$ title : Factor w/ 34 levels "Autochthonous Dirofilaria (Nochtiella) repens infection in dogs in Kuwait",..: 11 31 26 29 29 29 32 32 32 32 ...

$ authors : Factor w/ 34 levels "Abdullah, Hham; Amanzougaghene, N.; Dahmana, H.; Louni, M.; Raoult, D.; Mediannikov, O.",..: 1 2 3 4 4 4 5 5 5 5 ...

$ yearpub : num [1:132] 2021 2013 2018 2018 2016 ...

$ totalsample : num [1:132] 203 142 174 306 306 306 100 102 100 100 ...

$ npositive : num [1:132] 2 31 6 9 9 4 0 15 1 11 ...

$ DxMethod : Factor w/ 14 levels "Antigen Rapid Caniv-4 (Leish) Test Kit, Bionote",..: 13 3 14 3 13 12 6 6 6 6 ...

$ Substrate : Factor w/ 3 levels "Antigen","DNA",..: 2 1 1 1 2 3 1 1 1 1 ...

$ DxMethodGroup : Factor w/ 3 levels "Microscopy","PCR",..: 2 3 3 3 2 1 3 3 3 3 ...

$ TestCombo : Factor w/ 2 levels "Combination",..: 2 1 2 1 1 1 1 1 1 1 ...

$ DxSample : Factor w/ 2 levels "blood","serum": 1 2 2 1 1 1 1 1 1 1 ...

$ DxSampleGroup : Factor w/ 1 level "blood": 1 1 1 1 1 1 1 1 1 1 ...

$ organ\_structure\_involved: Factor w/ 2 levels "subcutaneous",..: 2 2 2 2 2 2 2 2 2 2 ...

$ structureGroup : Factor w/ 2 levels "subcutaneous",..: 2 2 2 2 2 2 2 2 2 2 ...

$ structureGroup2 : Factor w/ 2 levels "subcutaneous",..: 2 2 2 2 2 2 2 2 2 2 ...

$ HostSpecies : Factor w/ 2 levels "cat","dogs": 2 2 1 2 2 2 2 2 2 2 ...

$ HostGroup : Factor w/ 2 levels "cat","dog": 2 2 1 2 2 2 2 2 2 2 ...

$ HostGroup2 : Factor w/ 2 levels "cat","dog+": 2 2 1 2 2 2 2 2 2 2 ...

$ DirofilariaSpecies : Factor w/ 2 levels "D. immitis","D. repens": 2 1 1 1 1 1 1 1 1 1 ...

$ Country : Factor w/ 10 levels "Algeria","Egypt",..: 2 10 2 10 10 10 10 10 10 10 ...

$ City : Factor w/ 61 levels "Abou-rawah","Adana",..: 17 15 1 55 55 55 27 20 59 32 ...

$ Continent : Factor w/ 2 levels "Africa","Asia": 1 2 1 2 2 2 2 2 2 2 ...

Error: attempt to use zero-length variable name

```{r meta}

**m.propsub <- metaprop(event = subset1$npositive,**

**n = subset1$totalsample,**

**studlab = as.character(subset1$recordID),**

**prediction = TRUE,**

**data = subset1,**

**method = "GLMM",**

**sm = "PLOGIT",**

**fixed = FALSE,**

**random = TRUE,**

**hakn = TRUE,**

**title = "Meta-Results - Data= Subset1")**

**summary(m.propsub)**

```

Number of studies: k = 132

Number of observations: o = 13123

Number of events: e = 784

proportion 95%-CI

Random effects model 0.0238 [0.0157; 0.0358]

Prediction interval [0.0005; 0.5634]

Quantifying heterogeneity:

tau^2 = 3.9837; tau = 1.9959; I^2 = 81.7% [78.6%; 84.3%]; H = 2.34 [2.16; 2.52]

Test of heterogeneity:

Q d.f. p-value

Wald 714.98 131 < 0.0001

LRT 1548.73 131 < 0.0001

Details on meta-analytical method:

- Random intercept logistic regression model

- Maximum-likelihood estimator for tau^2

- Random effects confidence interval based on t-distribution (df = 131)

- Prediction interval based on t-distribution (df = 130)

- Logit transformation

- Clopper-Pearson confidence interval for individual studies

- Continuity correction of 0.5 in studies with zero cell frequencies

(only used to calculate individual study results)

```{r forest}

#Forest plot of the entire dataset is enormous and not useful.

#Will make forest plot of the diagnostic method moderator analysis

png(filename = "C:\\Users\\ **XXXYourFilePathXXX** ", width=5000,height=12000,res=400)

forest(m.propsub)

dev.off()

```

```{r outliers}

**m.propsub2=find.outliers(m.propsub)**

**m.propsub2**

```

Identified outliers (random-effects model)

------------------------------------------

"2", "8", "10", "33", "35", "45", "46", "48", "49", "51", "129", "130", "58", "59", "60", "129", "64", "65", "124", "66", "67", "68", "69", "78", "80", "81", "83", "93", "94", "96", "97", "98", "99", "100", "101", "102", "103", "105", "112"

Results with outliers removed

-----------------------------

Review: Meta-Results - Data= Subset1

132-93= 39 studies removed

Number of studies: k = 93

Number of observations: o = 9498

Number of events: e = 139

proportion 95%-CI

Random effects model 0.0101 [0.0071; 0.0145]

Prediction interval [0.0015; 0.0656]

Quantifying heterogeneity:

tau^2 = 0.9058; tau = 0.9517; I^2 = 0.0% [0.0%; 25.2%]; H = 1.00 [1.00; 1.16]

Test of heterogeneity:

Q d.f. p-value

Wald 48.11 92 1.0000

LRT 190.49 92 < 0.0001

Details on meta-analytical method:

- Random intercept logistic regression model

- Maximum-likelihood estimator for tau^2

- Random effects confidence interval based on t-distribution (df = 92)

- Prediction interval based on t-distribution (df = 91)

- Logit transformation

- Continuity correction of 0.5 in studies with zero cell frequencies

(only used to calculate individual study results)

```{r influence}

propsub.inf = InfluenceAnalysis(m.propsub, random = TRUE)

png(filename = "C:\\Users\\ **XXXYourFilePathXXX**.png", width=5000,height=8000,res=400)

plot(propsub.inf, "baujat") # Can change different plots as necessary

dev.off() #stops from saving more into the PNG file

```

```{r moderators}

#---- Subgroup Analysis ----

# All variables are represented

# Some of the variables may not be suitable for final inclusion.

# Variables with small K value will have to be excluded, continent also might be a problem

#1 - Diagnostic Method, all values for diagnostic methods

update(m.propsub,

subgroup = **DxMethod**,

tau.common = FALSE)

A screenshot of a computer

Description automatically generated

#2 - Substrate, the biologic agent being tested/detected

update(m.propsub,

subgroup = **Substrate**,

tau.common = FALSE)

A screenshot of a computer program

Description automatically generated

#3 - DxMethodGroup, aggregation of DXMethod variable

update(m.propsub,

subgroup = **DxMethodGroup**,

tau.common = FALSE)

A screenshot of a computer

Description automatically generated

#4 - TestCombo, indication if a single test was used on the sample, or if multiple tests were used on the same sample.

update(m.propsub,

subgroup = **TestCombo**,

tau.common = FALSE)

A screenshot of a computer program

Description automatically generated

#5 - DxSample, sample from the animal used in the diagnostic test

update(m.propsub,

subgroup = **DxSample**,

tau.common = FALSE)

A screenshot of a computer

Description automatically generated

#6 - DXSampleGroup, aggregation of DX sample variable.

update(m.propsub,

subgroup = **DxSampleGroup**,

tau.common = FALSE)

A screenshot of a computer program

Description automatically generated

#7 - Organ\_structure\_involved indicates location of worms/sample, or if sample was collected for surveillance

update(m.propsub,

subgroup = **organ\_structure\_involved**,

tau.common = FALSE)

A screenshot of a computer program

Description automatically generated

#8 - structureGroup, aggregation of organ\_structure\_involved'

update(m.propsub,

subgroup = **structureGroup**,

tau.common = FALSE)

A screenshot of a computer

Description automatically generated

#9 - StructureGroup2, aggregation of structureGroup

update(m.propsub,

subgroup = **structureGroup2**,

tau.common = FALSE)

A screenshot of a computer

Description automatically generated

#10 - HostSpecies, host the sample was collected from

update(m.propsub,

subgroup = **HostSpecies**,

tau.common = FALSE)

A screenshot of a computer

Description automatically generated

#11 - HostGroup, aggregation of HostSpecies Variable

update(m.propsub,

subgroup = **HostGroup**,

tau.common = FALSE)

A screenshot of a computer program

Description automatically generated

#12 - HostGroup2, aggregation of HostGroup

update(m.propsub,

subgroup = **HostGroup2**,

tau.common = FALSE)

A screenshot of a computer program

Description automatically generated

#13 - DirofilariaSpecies, species detected by test

update(m.propsub,

subgroup = **DirofilariaSpecies**,

tau.common = FALSE)

A screenshot of a computer code

Description automatically generated

#14 - Country data

update(m.propsub,

subgroup = **Country**,

tau.common = FALSE)

A screenshot of a computer

Description automatically generated

#Model fails to converge, removing from analysis

#15 - City data

#update(m.propsub,

# subgroup = **City**,

# tau.common = FALSE)

#16 - continent data

update(m.propsub,

subgroup = **Continent**,

tau.common = FALSE)

A screenshot of a computer

Description automatically generated

#Forest Plot of diagnostic group moderator analysis

#Step 1 - make individual datasets of the subgroups for each level of diagnostic group

#Sample code from Jordan - Newdata <- oldatasetname[olddatasetname$variable == "subgroup", ]

DxGroupPCR <- subset1[subset1$DxMethodGroup == "PCR", ]

DxGroupRapid <- subset1[subset1$DxMethodGroup == "Rapid Test (antigen)", ]

DxGroupMicro <- subset1[subset1$DxMethodGroup == "Microscopy", ]

#Step 2 - Make individual Random effects models for each new subgroup dataset

#then output to .png file

m.PCRsub <- metaprop(event = DxGroupPCR$npositive,

n = DxGroupPCR$totalsample,

studlab = as.character(DxGroupPCR$recordID),

prediction = TRUE,

data = DxGroupPCR,

method = "GLMM",

sm = "PLOGIT",

fixed = FALSE,

random = TRUE,

hakn = TRUE,

title = "Meta-Results - Data= DxGroupPCR")

summary(m.PCRsub)

A white background with black numbers and a number

Description automatically generated

png(filename = "C:\\Users\\ **XXXYourFilePathXXX**.png", width=1200,height=1400,res=150)

forest(m.PCRsub)

dev.off()

####################################################################

m.Rapidsub <- metaprop(event = DxGroupRapid$npositive,

n = DxGroupRapid$totalsample,

studlab = as.character(DxGroupRapid$recordID),

prediction = TRUE,

data = DxGroupRapid,

method = "GLMM",

sm = "PLOGIT",

fixed = FALSE,

random = TRUE,

hakn = TRUE,

title = "Meta-Results - Data= DxGroupRapid")

summary(m.Rapidsub)

A white background with black numbers and a few black text

Description automatically generated with medium confidence

png(filename = "C:\\Users **XXXYourFilePathXXX**.png", width=1030,height=2000,res=120)

forest(m.Rapidsub)

dev.off()

#####################################################################

m.Microsub <- metaprop(event = DxGroupMicro$npositive,

n = DxGroupMicro$totalsample,

studlab = as.character(DxGroupMicro$recordID),

prediction = TRUE,

data = DxGroupMicro,

method = "GLMM",

sm = "PLOGIT",

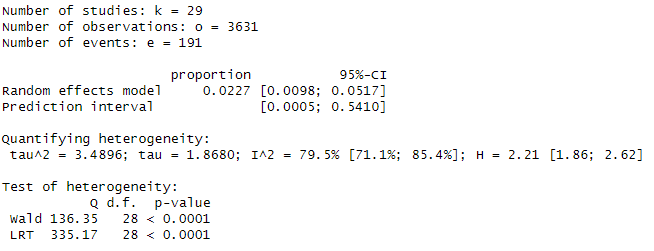
fixed = FALSE,

random = TRUE,

hakn = TRUE,

title = "Meta-Results - Data= DxGroupMicro")

summary(m.Microsub)



png(filename = "C:\\Users\\ **XXXYourFilePathXXX**.png", width=1200,height=1080,res=140)

forest(m.Microsub)

dev.off()

```

```{r publication bias}

#---- **Contoured Funnel Plot** ----

# Define fill colors for contour

col.contour = c("gray75", "gray85", "gray95")

# Generate funnel plot

funnel(m.propsub,

xlim = c(-10,4),

contour = c(0.9, 0.95, 0.99),

col.contour = col.contour,

studlab = T)

# Add a legend

legend(x = -10, y = 0,

legend = c("p < 0.1", "p < 0.05", "p < 0.01"),

fill = col.contour)

# Add a title

title("Contour-Enhanced Funnel Plot, Subset1 Data")

# ---- **Peter's Test of Funnel Plot** ----

metabias(m.propsub, method.bias = "peters")

A screenshot of a computer code

Description automatically generated

# ---- **Trim & Fill** ----

m.propsub$I2

A screenshot of a computer

Description automatically generated

# Two trim and fill analysis were conducted in light of the high heterogeneity - one on entire data set, one with identified outliers removed

trim.prop = trimfill(m.propsub)

trim.prop2 = trimfill(update(m.propsub,

subset = -c(2,8,10,33,35,45,46,48,49,51,58,59,60,64,65,66,67,68,69,78, 80,81,83,93,94,96,97,98,99,100,101,102,103,105,112,124,129,130)))

**summary(trim.prop) # 53 studies added**

A white background with black text

Description automatically generated

**summary(trim.prop2) # 35 Additional studies added**

A white background with black text

Description automatically generated

```