#meta-analysis project code#

#packages#

library(readxl)

library(metafor)

library(dmetar)

library(metaviz)

library(effectsize)

library(dplyr)

library(esc)

#calculate whole effect size for expression studies and disease severity#

#read my file#

TopeMeta= read\_excel(file.choose())

TopeMeta

##How many data type we have and how to write it out

unique(TopeMeta$Or\_Gene\_IDentify)%>%as.vector()%>%length()

unique(TopeMeta$Or\_Gene\_IDentify)%>%as.vector()%>%print()

unique(TopeMeta$plant\_host)%>%as.vector()%>%length()

unique(TopeMeta$plant\_host)%>%as.vector()%>%print()

## how many Transcription Factors

# calculate effect size hedges'g#

Overallcomb.effect <- escalc(measure="SMDH",

m1i=m1i, sd1i=sd1i, n1i=n1i,

m2i=m2i, sd2i=sd2i, n2i=n2i,

data=TopeMeta)

write.csv(Overallcomb.effect, "DTcombeffect.csv")

TopeMetaf=read.csv(file.choose())

Fisher\_data1=escalc(measure="ZCOR", ri=ri, ni=n1i, data=TopeMetaf)

##How many data type

unique(Overallcomb.effect$Data.type)%>%as.vector()%>% length()

unique(Overallcomb.effect$Data.type)%>%as.vector()%>%print(Data.type)

## how many Transcription Factors

unique(Overall.comb.effect$General.gene.name)%>%as.vector()%>% length()

###run the unweighted analysis with the whole data (0.1 been used as error) in metafor package####

TopeMetaDW=read.csv(file.choose())

Data\_unweighted <- rma(yi, vi, data=TopeMetaDW, method = "REML", weighted=FALSE)

Data\_unweighted

write.csv(Data\_unweighted,"Runweighted\_data1.csv")

summary(Data\_unweighted)

Data\_unweighted$pval

exp(Data\_unweighted$b)

exp(Data\_unweighted$ci.lb)

exp(Data\_unweighted$ci.ub)

forest(Data\_unweighted)

funnel(Data\_unweighted)

##conduct meta-analysis - WEIGHTED random effects

TopeMetaDW1=read.csv(file.choose())

Tope\_weighted=rma(yi, vi, data=TopeMetaDW, method="REML")

summary(Tope\_weighted)

Tope\_unweighted$pval

exp(Tope\_unweighted$b)

exp(Tope\_unweighted$ci.lb)

exp(Tope\_unweighted$ci.ub)

jpeg(filename = "funnelplottope\_weighted.jpeg",width =3200,height=1000, quality = 6000, bg="white")

viz\_funnel(Tope\_weighted)

dev.off()

###forest plot of whole unweighted analysis####

TopeMetaDW1=read.csv(file.choose())

jpeg(filename = " weighted meta analytical forest plot.jpeg", width = 6100, height = 2000, quality = 6000,

bg="white")

forest(Tope\_weighted,xlim=c(-50,10), pch = 19, psize = 1,efac=0.1,ylim = c(0,409),

at=log(c(0.01,0.1,1,10)),

ilab = cbind(TopeMetaDW1$Or\_Gene\_IDentify, TopeMetaDW1$Variable, TopeMetaDW1$fungal.pathogen,

TopeMetaDW1$General\_gene\_name),

fonts = 'serif',

ilab.xpos = c(-40,-30,-20,-10),mlab = "",cex=0.75,xlab = '')

op <- par(cex=1, font=2)

text(c(-48,-40,-30,-20,-10,0,7), 409, c("Or\_Gene\_IDentify",'variable','fungal.pathogen','General\_gene\_name', 'SMDH &

95%CI'))

par(op)

text(-50, -1, pos=4,cex=1, font=2, bquote(paste("RE Model for weighted analysis of whole studies as.raw(Q = ",

.(formatC(TopeMetaDW1$QE, digits=2, format="f")), ", df = ", as.numeric

(TopeMetaDW1$k - TopeMetaDW1$p),

", p = ", .(formatC(TopeMetaDW1$QEp, digits=2, format="f")), "; ",

I^2, " = ",

.(formatC(TopeMetaDW1$I2, digits=1, format="f")), "%)")))

dev.off()

datum.mis = datum

datum.mis[c(7,9), c(5,8)] = NA

View(datum.mis)

# Calculations are now done only on complete cases

complete.datum = escalc(measure="SMD",

m1i=m1i, sd1i=sd1i, n1i=n1i,

m2i=m2i, sd2i=sd2i, n2i=n2i,

data=datum.mis)

Overall.effect <- escalc(measure="SMDH",

m1i=m1i, sd1i=sd1i, n1i=n1i,

m2i=m2i, sd2i=sd2i, n2i=n2i,

data=TopeMeta)

#Read file numbers#

Gene.data.extraction <- read.table("DTcombeffect.csv",

sheet = "Gene Expression data", col\_types = c("numeric",

"text", "text", "numeric", "text", "text",

"text", "text", "text", "text",

"numeric", "numeric", "numeric","numeric","text","numeric","numeric","numeric","text","text","text","numeric","numeric","numeric"), header = header, sep = sep, quote = quote))

unique(Tope.f$Study)%>>%as.vector()%>>% length()

unique(Tope.f$Studies)%>>%as.vector()%>>%Print()

Total.data.extraction <-read.csv()

#Read file numbers#

TopeMetaDW2=read\_excel(file.choose())

Disease.data.extraction <- read\_excel("C:/Users/trf0019/Downloads/T\_Meta\_tabledata.xlsx",

sheet = "Disease\_ratedata", col\_types = c("numeric",

"text", "text", "numeric", "text", "text",

"text", "text", "text", "text",

"numeric", "numeric", "numeric","numeric","text","numeric","numeric","numeric","text","text","text"))

Gene.data.extraction1 <- read\_excel("C:/Users/trf0019/Downloads/T\_Meta\_tabledata.xlsx",

sheet = "Gene\_expressiondata", col\_types = c("numeric",

"text", "text", "numeric", "text", "text",

"text", "text", "text", "text",

"numeric", "numeric", "numeric","numeric","text","numeric","numeric","numeric","text","text","text"))

#calculate effect size for Groups#

gene\_expression.effect <- escalc(measure="SMDH",

m1i=m1i, sd1i=sd1i, n1i=n1i,

m2i=m2i, sd2i=sd2i, n2i=n2i,

data=Gene.data.extraction1)

Disease\_data.effectsize <- escalc(measure="SMDH",

m1i=m1i, sd1i=sd1i, n1i=n1i,

m2i=m2i, sd2i=sd2i, n2i=n2i,

data=Disease.data.extraction)

write.csv(gene\_expression.effect, "Gene\_expressioneffectsizedt.csv")

write.csv(Disease\_data.effectsize, "Diseaseeffectsizedt.csv")

###run the meta analysis for the studies with Gene expression studies in meta package ######

library(metagen)

install.packages("devtools")

devtools::install\_github("metagen")

library(metagen)

Gene\_metaanalysis <- metagen(

TE = yi,

seTE = vi,

data = gene\_expression.effect,

studlab = paste(Studies, step=", "),

comb.fixed = F,

comb.random = T,

method.tau = "SJ",

hakn = T,

prediction = F,

sm = "SMDH",

backtransf = F

)

summary(Gene\_metaanalysis)

##leave 1 out analysis for whole result unweighted analysis#####

unweighted.result.sensitivity <- leave1out(Tope\_weighted)

unweighted.sensitivity.analysis <- as.data.frame(Tope\_weighted)

write.csv(Tope\_weighted," Sensitivity analysis for the whole unweighted meta-analysis.csv")

max(Tope\_weighted$estimate)

min(Tope\_unweighted$estimate)

###change the format of the leave 1 out analysis####

unweighted.result.sensitivity$`Confidence Interval`<- paste(

'',

round(unweighted.result.sensitivity$`lower bounds of confidence interval`,3),

step = ' - ',

round(unweighted.result.sensitivity$`upper-bounds of confidence interval`,3)

)

write.csv(unweighted.result.sensitivity, 'sensitivity analysis for the whole

unweighted meta analysis.csv')

GeneExData\_unweighted <- rma(yi, vi, data=gene\_expression.effect, method = "REML", weighted=FALSE)

geneExdata <- as.data.frame(GeneExData\_unweighted)

GeneExData\_unweighted

#weigtted Analysis#

GeneExData\_weighted <- rma(yi, vi, data=gene\_expression.effect, method = "REML")

GeneExData\_weighted

#Disease rate#

DiseaseData\_unweighted <- rma(yi, vi, data=Disease\_data.effectsize, method = "REML", weighted=FALSE)

DiseaseData\_unweighted

DiseaseData\_weighted <- rma(yi, vi, data=Disease\_data.effectsize, method = "REML")

DiseaseData\_weighted

viz\_funnel(DiseaseData\_unweighted)

DiseaseData\_unweighted <- rma(yi, vi, data=Disease\_data.effectsize, method = "REML", weighted=FALSE)

DiseaseData\_unweighted

viz\_funnel(DiseaseData\_weighted)

viz\_funnel(GeneExData\_unweighted)

jpeg(filename = "funnelplotofGeneExdata\_unweighted1.jpeg",width =3200,height=1000, quality = 6000, bg="white")

viz\_funnel(GeneExData\_unweighted)

dev.off()

regtest(GeneExData\_unweighted)

res.tf <- trimfill(GeneExData\_unweighted)

res.tf

viz\_funnel(res.tf)

jpeg(filename = "TrimfillplotofGeneExdata\_unweighted1.jpeg",width =3200,height=1000, quality = 6000, bg="white")

viz\_funnel(res.tf)

dev.off()

write.csv(GeneExData\_unweighted,"Geneunweighted\_data.csv")

forest(GeneExData\_unweighted)

funnel(GeneExData\_unweighted)

jpeg(filename = "funnel plot of Geneexdata unweighted analysis in metafor.jpeg", width = 1600, height = 900,

quality = 2000, bg="white")

viz\_funnel(GeneExData\_unweighted)

dev.off()

regtest(GeneExData\_unweighted)

eggers.test()

##conduct meta-analysis - WEIGHTED random effects

TopeMetaDW1=read.csv(file.choose())

Tope\_weighted=rma(yi, vi, data=TopeMetaDW, method="REML")

summary(Tope\_weighted)

Tope\_weighted$pval

exp(unweighted$b)

exp(unweighted$ci.lb)

exp(unweighted$ci.ub)

##run meta-analysis - random effects - INCLUDE RANDOM FACTOR FOR STUDY (USEFUL IF MULTIPLE EFFECTS FROM SAME PAPER) - NOTE RMA.MV CHANGE

#compare gene expression to null and Disaese rate to intercept# Moderator with intercept

TopeMetaDW2=read.csv(file.choose())

Moderator.overallanalysisgene1 = rma(yi=yi, vi=vi, mods=~Or\_Gene\_IDentify, data=TopeMetaDW2, method="REML")

Moderator.overallanalysisgene

Moderator.overallanalysisgene$pval

Geneidentity<- as.data.frame(Moderator.overallanalysisgene)

write.csv(Geneidentity)

forest(Moderator.overallanalysisgene1)

viz\_funnel(Moderator.overallanalysisgene)

##run meta-analysis - random effects - INCLUDE RANDOM FACTOR FOR STUDY (USEFUL IF MULTIPLE EFFECTS FROM SAME PAPER) - NOTE RMA.MV CHANGE

#compare gene expression to null and Disaese rate to intercept# Moderator without intercept

TopeMetaDW2=read.csv(file.choose())

Moderator.overallanalysisgene = rma(yi=yi, vi=vi, mods=~Or\_Gene\_Identify-1, data=TopeMetaDW2, method="REML")

Moderator.overallanalysisgene

Moderator.overallanalysisgene$pval

forest(Moderator.overallanalysis)

viz\_funnel(Moderator.overallanalysis)

## conduct trim and fill analysis - https://rdrr.io/cran/metafor/man/trimfill.html

res.tf1 <- trimfill(Tope\_weighted)

res.tf1

viz\_funnel(res.tf1)

regtest(Tope\_unweighted, model="lm")

eggers.test(Tope\_unweighted)

TopeMetaDW3=read.csv(file.choose())

unique(TopeMetaDW3$Or\_Gene\_IDentify)%>%as.vector()%>% length()

unique(TopeMetaDW3$Or\_Gene\_IDentify)%>%as.vector()%>% print()

###Run subgroup analysis for whole effect size in unweighted analysis####

##Variable type##

TopeMetaDW4=read.csv(file.choose())

Diseaserate2 <- TopeMetaDW4[which(TopeMetaDW4$Variable=="1"),]

Diseaserate3 <- subset(TopeMetaDW4, Variable =="Disease")

GeneExpression1 <-subset(TopeMetaDW4, Variable == "GeneExpression")

Disease1 <- rma(vi = vi, yi = yi, data = Diseaserate3, slab = paste(Studies,step=", "), method = "REML",

weighted = F)

summary(Disease)

viz\_forest(Disease)

viz\_funnel(Disease)

Disease.Or\_Gene\_IDentify <- rma(vi = 0.1, yi = yi, data = Diseaserate2, slab = paste(Studies,step="X"), method =

"REML", weighted = F, mods = ~Or\_Gene\_IDentify-1)