######## Hedges' *d*-*R* codes

library(dmetar)

library(esc)

library(tidyverse)

library(openxlsx)

library(readxl)

library(meta)

library(moments)

library(tibble)

dat <- read.xlsx("D:/Meta.data/nee.xlsx")

SP\_calc <- esc\_mean\_sd(grp1m = dat$mean.e,

grp1sd = dat$sd.e,

grp1n = dat$n.e,

grp2m = dat$mean.c,

grp2sd = dat$sd.c,

grp2n = dat$n.c,

study = dat$author,

es.type = "g") %>%

as.data.frame()

glimpse(SP\_calc)

m.gen <- metagen(TE= es,

seTE = se,

studlab = study,

data = SP\_calc,

sm = "SMD",

fixed = FALSE,

random = TRUE,

method.tau = "REML",

hakn = TRUE,

title = "trhx Psychotherapies")

summary(m.gen)

## three-level meta-analysis-*R* codes

full.model <- rma.mv(yi = TE,

V = seTE,

slab = Author,

data = dat,

random = ~ 1 | Author/climate,

test = "t",

method = "REML")

summary(full.model)

########publication bias by the “trim and fill” method-*R* codes

tf <- trimfill(m.gen)

summary(tf)

col.contour <- c("gray75", "gray85", "gray95")

par(mfrow=c(1,1))

contour <- c(0.9, 0.95, 0.99)

funnel.meta(tf.no.out,

xlim = c(-4, 10), ylim = c(1.2, 0),contour = contour,

col.contour = col.contour)

title("NEE")

legend(x = 4.1, y = 0.1,

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

fill = col.contour)

######## fail-safe number-*R* codes

metabias(m.gent, method.bias = "linreg")#

fsn(yi=TE, vi=seTE, type = "Rosenthal", data=m.gent)

fsn(yi=TE, vi=seTE, type = "Orwin", data=m.gent, weighted=TRUE, target=log(0.95))

######## sensitivity to outliers-*R* codes

find.outliers(m.gen)

########between-group heterogeneity-*R* codes

dat <- read.xlsx("D:/Meta.data/bh.xlsx")

glimpse(dat)

m.gen <- metagen(TE = TE,

seTE = seTE,

studlab = Author,

data = dat,

sm = "REML",

fixed = FALSE,

random = TRUE,

method.tau = "REML",

hakn = TRUE,

title = "Third Wave Psychotherapies")

summary(m.gen)

update.meta(m.gen,

subgroup = climate,

tau.common = FALSE)

######## meta-regression-*R* codes

dat <- read.xlsx("D:/Meta.data/meta-regression.xlsx")

glimpse(dat)

m.gen <- metagen(TE = TE,

seTE = seTE,

studlab = author,

data = dat,

sm = "SMD",

fixed = FALSE,

random = TRUE,

method.tau = "REML",

hakn = TRUE,

title = "Third Wave Psychotherapies")

summary(m.gen)

m.gen.reg <- metareg(m.gen, ~add)

m.gen.reg

bubble(m.gen.reg, studlab = FALSE, xlim = c(0, 800), ylim = c(-15, 15), xlab="Cumulative amount added (kgN/ha)", ylab = "Effect size of NEE", lwd = 4, lty = 1, bg = "#F7903D", min.cex = 1, max.cex = 5, col.line = "black")

######## Random Forest Model for Machine Learning-Python codes

import pandas as pd

df=pd.read\_excel("D:/Meta.data/meta-regression.xlsx")

import pandas as pd

from sklearn.ensemble import RandomForestRegressor

import numpy as np

X = df[["add", "seTE"]].values

y = df["TE"].values

from sklearn.model\_selection import train\_test\_split

X\_train, X\_test,y\_train, y\_test = train\_test\_split(X, y, test\_size=0.2, random\_state=42)

new\_data = np.column\_stack((np.round(np.random.uniform(300,700,50),0),np.random.uniform(0.5,1.2,50)))

predicted\_TE = model.predict(new\_data)

new\_df = pd.DataFrame(new\_data, columns=["add", "seTE"])

new\_df["TE"] = predicted\_TE

new\_df.to\_csv("C:/Users/89333/Desktop/mn.csv")

print(new\_df)

import matplotlib.pyplot as plt

plt.scatter(df["add"], df["TE"], label="Ture TE", marker="o")

plt.scatter(df["add"], model.predict(np.column\_stack((df["add"], df["seTE"]))), label="RF Predicted TE", marker="x")

y = 3.4 \* 10\*\*-3 \* df[["add"]]-1.2346

plt.scatter(df["add"], y, label= "meta-regression Predicted TE", marker="+")

plt.xlabel("add")

plt.ylabel("TE")

plt.title("Ture vs Predicted TE Values")

plt.legend()

plt.show()