library(MASS)

library(metafor)

library(minpack.lm)

setwd ("C:/GBD")

data<- read.csv("GBD2015 input\_dataset with Pn and China Ensemble error.csv ", header = T)

cause=as.character(data[,3])

source=as.character(data[,2])

data[is.na(data)]=0

child=as.character(data[,11])

dat=subset(data, cause=="lri")

#only OAP

datrr=subset(dat, child=="0")

datr=subset(datrr, source=="OAP")

num=as.numeric(datr[,4])

den=as.numeric(datr[,5])

logr=as.numeric(datr[,6])

se=as.numeric(datr[,7])

medage=as.numeric(datr[,9])

age=27.5

#se=se\* ((age-110)/(medage-110))

Vall=se^2

#logr=logr\*((age-110)/(medage-110))

#THRES=min(den)

THRES=2.4

numt=((num-THRES)+abs(num-THRES))/2

dent=((den-THRES) + abs(den-THRES))/2

r=max(num)-min(den)

a=c(1,3,5,7,9)

e=c(den, num)-THRES

mm0=min(e)

mm25=quantile(e, 0.25)

mm50=quantile(e, 0.5)

mm75=quantile(e, 0.75)

mm95=quantile(e, 0.95)

m=c(mm0, mm25, mm50, mm75, mm95)

max1=length(m)\*length(a)

max2=max1

max3=max1

aic1<-matrix(0, max1, 1)

b1<-matrix(0, max1, 1)

se1<-matrix(0, max1, 1)

model.form1<-matrix(0, max1, 1)

set.tau1<-matrix(0, max1, 1)

mu1<-matrix(0, max1, 1)

aic2<-matrix(0, max2, 1)

b2<-matrix(0, max2, 1)

se2<-matrix(0, max2, 1)

model.form2<-matrix(0, max2, 1)

set.tau2<-matrix(0, max2, 1)

mu2<-matrix(0, max2, 1)

aic3<-matrix(0, max3, 1)

b3<-matrix(0, max3, 1)

se3<-matrix(0, max3, 1)

model.form3<-matrix(0, max3, 1)

set.tau3<-matrix(0, max3, 1)

mu3<-matrix(0, max3, 1)

aic4<-matrix(0, max1, 1)

b4<-matrix(0, max1, 1)

se4<-matrix(0, max1, 1)

model.form4<-matrix(0, max1, 1)

set.tau4<-matrix(0, max1, 1)

mu4<-matrix(0, max1, 1)

aic5<-matrix(0, max1, 1)

b5<-matrix(0, max1, 1)

se5<-matrix(0, max1, 1)

model.form5<-matrix(0, max1, 1)

set.tau5<-matrix(0, max1, 1)

mu5<-matrix(0, max1, 1)

aic6<-matrix(0, max1, 1)

b6<-matrix(0, max1, 1)

se6<-matrix(0, max1, 1)

model.form6<-matrix(0, max1, 1)

set.tau6<-matrix(0, max1, 1)

mu6<-matrix(0, max1, 1)

numt=((num-THRES) + abs(num-THRES))/2

dent=((den-THRES) + abs(den-THRES))/2

#estimate theta using rma.mv by defining difference in shapes between num and den #concentrations

j=0

for (k in 1:length(a)) {

for (i in 1:length(m)) {

j=j+1

diff=log(numt/a[k]+1)/(1+exp(-(numt-m[i])/(0.1\*r))) - log(dent/a[k]+1)/(1+exp(-(dent-m[i])/(0.1\*r)))

fit=rma(yi=logr, vi=Vall, mods=~diff -1, method="REML", intercept=FALSE)

aic1[j]=AIC(fit)

b1[j]=coef(fit)

model.form1[j]<- a[k]

set.tau1[j] <- 0.1

mu1[j] <- m[i]

se1[j]=sqrt(vcov(fit))}}

j=0

for (k in 1:length(a)) {

for (i in 1:length(m)) {

j=j+1

diff=log(numt/a[k]+1)/(1+exp(-(numt-m[i])/(0.2\*r))) - log(dent/a[k]+1)/(1+exp(-(dent-m[i])/(0.2\*r)))

fit=rma(yi=logr, vi=Vall, mods=~diff -1, method="REML", intercept=FALSE)

aic2[j]=AIC(fit)

b2[j]=coef(fit)

model.form2[j]<- a[k]

set.tau2[j] <- 0.2

mu2[j] <- m[i]

se2[j]=sqrt(vcov(fit))}}

j=0

for (k in 1:length(a)) {

for (i in 1:length(m)) {

j=j+1

diff=log(numt/a[k]+1)/(1+exp(-(numt-m[i])/(0.3\*r))) - log(dent/a[k]+1)/(1+exp(-(dent-m[i])/(0.3\*r)))

fit=rma(yi=logr, vi=Vall, mods=~diff -1, method="REML", intercept=FALSE)

aic3[j]=AIC(fit)

b3[j]=coef(fit)

model.form3[j]<- a[k]

set.tau3[j] <- 0.3

mu3[j] <- m[i]

se3[j]=sqrt(vcov(fit))}}

j=0

for (k in 1:length(a)) {

for (i in 1:length(m)) {

j=j+1

diff=log(numt/a[k]+1)/(1+exp(-(numt-m[i])/(0.4\*r))) - log(dent/a[k]+1)/(1+exp(-(dent-m[i])/(0.4\*r)))

fit=rma(yi=logr, vi=Vall, mods=~diff -1, method="REML", intercept=FALSE)

aic4[j]=AIC(fit)

b4[j]=coef(fit)

model.form4[j]<- a[k]

set.tau4[j] <- 0.4

mu4[j] <- m[i]

se4[j]=sqrt(vcov(fit))}}

j=0

for (k in 1:length(a)) {

for (i in 1:length(m)) {

j=j+1

diff=log(numt/a[k]+1)/(1+exp(-(numt-m[i])/(0.5\*r))) - log(dent/a[k]+1)/(1+exp(-(dent-m[i])/(0.5\*r)))

fit=rma(yi=logr, vi=Vall, mods=~diff -1, method="REML", intercept=FALSE)

aic5[j]=AIC(fit)

b5[j]=coef(fit)

model.form5[j]<- a[k]

set.tau5[j] <- 0.5

mu5[j] <- m[i]

se5[j]=sqrt(vcov(fit))}}

j=0

for (k in 1:length(a)) {

for (i in 1:length(m)) {

j=j+1

diff=log(numt/a[k]+1)/(1+exp(-(numt-m[i])/(0.6\*r))) - log(dent/a[k]+1)/(1+exp(-(dent-m[i])/(0.6\*r)))

fit=rma(yi=logr, vi=Vall, mods=~diff -1, method="REML", intercept=FALSE)

aic6[j]=AIC(fit)

b6[j]=coef(fit)

model.form6[j]<- a[k]

set.tau6[j] <- 0.6

mu6[j] <- m[i]

se6[j]=sqrt(vcov(fit))}}

#complie necessary information from model fitting to construct ensemble estimate and bootstrap

#based CI

aic=rbind(aic1, aic2, aic3, aic4, aic5, aic6)

b=rbind( b1, b2, b3, b4, b5, b6)

se=rbind(se1, se2, se3, se4, se5, se6)

model.form <- rbind(model.form1, model.form2, model.form3, model.form4, model.form5, model.form6)

model.form=as.character(model.form)

set.tau <- rbind( set.tau1, set.tau2, set.tau3, set.tau4, set.tau5, set.tau6)

mu <- rbind(mu1, mu2, mu3, mu4, mu5, mu6)

wt=exp(-0.5\*(aic-min(aic)))/sum(exp(-0.5\*(aic-min(aic))))

output=cbind(model.form, aic, wt, mu, set.tau, b, se)

out=subset(output, wt>0)

f=out[,1]

aic=as.numeric(out[,2])

wt=as.numeric(out[,3])

mu=as.numeric(out[,4])

tau=as.numeric(out[,5])

beta=as.numeric(out[,6])

se=as.numeric(out[,7])

weight=exp(-0.5\*(aic-min(aic)))/sum(exp(-0.5\*(aic-min(aic))))

expo\_name <- c("PM2.5")

expo\_unit <- c("ug/m3")

nn <- length(f)

finalmodels.best.nn.final <- data.frame(tau=tau, funcform=f, coef=beta, se.coef=se, wt.final3=weight, loca\_perc=mu)

MM=max(num)

MM=150

bb=1

x=seq(0, MM, by=bb)

nx=length(x)

#run routine to construct enesembe estimate and bootstrap CI

risk <- function(ap\_data, finalmodels, expo\_name, unit, nn, TT){

x=seq(0, MM, by=bb)

# prepare x1 for use in simulation, varying depending on perc, set\_tau, and funcform

sim.x1 <- function(x\_sim, perc, set\_tau\_sim, funcform\_sim, TT){

mu <- perc

#r <- max(num)-min(den)

thr=((x\_sim-TT)+abs(x\_sim-TT))/2

if (funcform\_sim=="1"){x1<- log(thr+1)/(1+exp(-(thr-mu)/(set\_tau\_sim\*r)))}

if (funcform\_sim=="2"){x1<-log(1+ thr/2)/(1+exp(-(thr-mu)/(set\_tau\_sim\*r)))}

if (funcform\_sim=="3"){x1<-log(1+ thr/3)/(1+exp(-(thr-mu)/(set\_tau\_sim\*r)))}

if (funcform\_sim=="4"){x1<-log(1+ thr/4)/(1+exp(-(thr-mu)/(set\_tau\_sim\*r)))}

if (funcform\_sim=="5"){x1<-log(1+ thr/5)/(1+exp(-(thr-mu)/(set\_tau\_sim\*r)))}

if (funcform\_sim=="6"){x1<-log(1+ thr/6)/(1+exp(-(thr-mu)/(set\_tau\_sim\*r)))}

if (funcform\_sim=="7"){x1<-log(1+ thr/7)/(1+exp(-(thr-mu)/(set\_tau\_sim\*r)))}

if (funcform\_sim=="8"){x1<-log(1+ thr/8)/(1+exp(-(thr-mu)/(set\_tau\_sim\*r)))}

if (funcform\_sim=="9"){x1<-log(1+ thr/9)/(1+exp(-(thr-mu)/(set\_tau\_sim\*r)))}

if (funcform\_sim=="10"){x1<-log(1+ thr/10)/(1+exp(-(thr-mu)/(set\_tau\_sim\*r)))}

if (funcform\_sim=="11"){x1<-log(1+ thr/11)/(1+exp(-(thr-mu)/(set\_tau\_sim\*r)))}

if (funcform\_sim=="12"){x1<-log(1+ thr/12)/(1+exp(-(thr-mu)/(set\_tau\_sim\*r)))}

if (funcform\_sim=="13.8"){x1<-log(1+ thr/13.8)/(1+exp(-(thr-mu)/(set\_tau\_sim\*r)))}

if (funcform\_sim=="14"){x1<-log(1+ thr/14)/(1+exp(-(thr-mu)/(set\_tau\_sim\*r)))}

if (funcform\_sim=="15"){x1<-log(1+ thr/15)/(1+exp(-(thr-mu)/(set\_tau\_sim\*r)))}

if (funcform\_sim=="16"){x1<-log(1+ thr/16)/(1+exp(-(thr-mu)/(set\_tau\_sim\*r)))}

if (funcform\_sim=="17"){x1<-log(1+ thr/17)/(1+exp(-(thr-mu)/(set\_tau\_sim\*r)))}

if (funcform\_sim=="18"){x1<-log(1+ thr/18)/(1+exp(-(thr-mu)/(set\_tau\_sim\*r)))}

if (funcform\_sim=="19.9"){x1<-log(1+ thr/19.9)/(1+exp(-(thr-mu)/(set\_tau\_sim\*r)))}

if (funcform\_sim=="20"){x1<-log(1+ thr/20)/(1+exp(-(thr-mu)/(set\_tau\_sim\*r)))}

if (funcform\_sim=="21"){x1<-log(1+ thr/21)/(1+exp(-(thr-mu)/(set\_tau\_sim\*r)))}

if (funcform\_sim=="22"){x1<-log(1+ thr/22)/(1+exp(-(thr-mu)/(set\_tau\_sim\*r)))}

if (funcform\_sim=="23"){x1<-log(1+ thr/23)/(1+exp(-(thr-mu)/(set\_tau\_sim\*r)))}

if (funcform\_sim=="24"){x1<-log(1+ thr/24)/(1+exp(-(thr-mu)/(set\_tau\_sim\*r)))}

if (funcform\_sim=="25"){x1<-log(1+ thr/25)/(1+exp(-(thr-mu)/(set\_tau\_sim\*r)))}

if (funcform\_sim=="26"){x1<-log(1+ thr/26)/(1+exp(-(thr-mu)/(set\_tau\_sim\*r)))}

if (funcform\_sim=="27"){x1<-log(1+ thr/27)/(1+exp(-(thr-mu)/(set\_tau\_sim\*r)))}

if (funcform\_sim=="28"){x1<-log(1+ thr/28)/(1+exp(-(thr-mu)/(set\_tau\_sim\*r)))}

if (funcform\_sim=="29"){x1<-log(1+ thr/29)/(1+exp(-(thr-mu)/(set\_tau\_sim\*r)))}

if (funcform\_sim=="30"){x1<-log(1+ thr/30)/(1+exp(-(thr-mu)/(set\_tau\_sim\*r)))}

if (funcform\_sim=="31"){x1<-log(1+ thr/31)/(1+exp(-(thr-mu)/(set\_tau\_sim\*r)))}

if (funcform\_sim=="32"){x1<-log(1+ thr/32)/(1+exp(-(thr-mu)/(set\_tau\_sim\*r)))}

if (funcform\_sim=="33"){x1<-log(1+ thr/33)/(1+exp(-(thr-mu)/(set\_tau\_sim\*r)))}

if (funcform\_sim=="34"){x1<-log(1+ thr/34)/(1+exp(-(thr-mu)/(set\_tau\_sim\*r)))}

if (funcform\_sim=="35"){x1<-log(1+ thr/35)/(1+exp(-(thr-mu)/(set\_tau\_sim\*r)))}

if (funcform\_sim=="36"){x1<-log(1+ thr/36)/(1+exp(-(thr-mu)/(set\_tau\_sim\*r)))}

if (funcform\_sim=="37"){x1<-log(1+ thr/37)/(1+exp(-(thr-mu)/(set\_tau\_sim\*r)))}

if (funcform\_sim=="38"){x1<-log(1+ thr/38)/(1+exp(-(thr-mu)/(set\_tau\_sim\*r)))}

if (funcform\_sim=="39"){x1<-log(1+ thr/39)/(1+exp(-(thr-mu)/(set\_tau\_sim\*r)))}

if (funcform\_sim=="40"){x1<-log(1+ thr/40)/(1+exp(-(thr-mu)/(set\_tau\_sim\*r)))}

if (funcform\_sim=="41"){x1<-log(1+ thr/41)/(1+exp(-(thr-mu)/(set\_tau\_sim\*r)))}

if (funcform\_sim=="42"){x1<-log(1+ thr/42)/(1+exp(-(thr-mu)/(set\_tau\_sim\*r)))}

if (funcform\_sim=="43"){x1<-log(1+ thr/43)/(1+exp(-(thr-mu)/(set\_tau\_sim\*r)))}

if (funcform\_sim=="44"){x1<-log(1+ thr/44)/(1+exp(-(thr-mu)/(set\_tau\_sim\*r)))}

if (funcform\_sim=="45"){x1<-log(1+ thr/45)/(1+exp(-(thr-mu)/(set\_tau\_sim\*r)))}

if (funcform\_sim=="46"){x1<-log(1+ thr/46)/(1+exp(-(thr-mu)/(set\_tau\_sim\*r)))}

if (funcform\_sim=="47"){x1<-log(1+ thr/47)/(1+exp(-(thr-mu)/(set\_tau\_sim\*r)))}

if (funcform\_sim=="48"){x1<-log(1+ thr/48)/(1+exp(-(thr-mu)/(set\_tau\_sim\*r)))}

if (funcform\_sim=="49"){x1<-log(1+ thr/49)/(1+exp(-(thr-mu)/(set\_tau\_sim\*r)))}

if (funcform\_sim=="50"){x1<-log(1+ thr/50)/(1+exp(-(thr-mu)/(set\_tau\_sim\*r)))}

if (funcform\_sim=="81"){x1<-log(1+ thr/81)/(1+exp(-(thr-mu)/(set\_tau\_sim\*r)))}

return(x1)

}

# simulate 100,000 realizations based on se.coef AND weights derived from LL

if (nn >= 2) {

nsim<-10000

ran.3<-matrix(0, nsim, 1)

rr.3<-matrix(0, nsim, nx)

medRR.3<-matrix(0, nx, 1)

upcl.3<-matrix(0, nx, 1)

lowcl.3<-matrix(0, nx, 1)

pp <- 1 # position variable in the 100,000 sim

nn <- nn # consider top 3 models

nsim.sum <- 0 # count of nsim to ensure the last run lead to rownum of exactly 100,000

# k from 0 to nn-1: varying depending on models included and pooled

for (k in 0:(nn-1)) {

nsim.wt <- nsim \* round(finalmodels[length(finalmodels[,1])-k,]$wt.final3, digits =4)

loca\_perc\_sim <- finalmodels[length(finalmodels[,1])-k,]$loca\_perc

funcform\_sim <- finalmodels[length(finalmodels[,1])-k,]$funcform

set\_tau <- finalmodels[length(finalmodels[,1])-k,]$tau

x1 <- sim.x1(x\_sim=x, perc=loca\_perc\_sim, set\_tau\_sim=set\_tau, funcform\_sim=funcform\_sim, TT=THRES)

if (k==nn-1) {nsim.wt <- nsim - nsim.sum}

for (i in pp:(pp+nsim.wt-1)) {

if (i<=10000){

ran.3[i,]<-rnorm(1, finalmodels[length(finalmodels[,1])-k,]$coef, finalmodels[length(finalmodels[,1])-k,]$se.coef)

for (j in 1:length(x)){

rr.3[i,j]<-exp(ran.3[i,1]\*x1[j])

}

}

}

pp <- pp+nsim.wt

nsim.sum <- nsim.sum+nsim.wt

}

return(rr.3)

}

}

rr.3=risk(ap\_data=x, nn=nn, finalmodels=finalmodels.best.nn.final, expo\_name=expo\_name, unit=expo\_unit, TT=THRES)

mean=matrix(0, length(x), 1)

ucl=matrix(0, length(x), 1)

lcl=matrix(0, length(x), 1)

sd=matrix(0, length(x), 1)

for (j in 1:length(x)) {

mean[j]=mean(rr.3[,j])

ucl[j]=quantile(rr.3[,j], 0.975)

lcl[j]=quantile(rr.3[,j], 0.025)

sd[j]=sd(log(rr.3[,j]))

}

#estimate approximate function to ensemble estimate and assign all uncertainty to theta

xxx=((x-THRES)+abs(x-THRES))/2

logmean=log(mean)

taustart=0.4\*r

fitmean=nlsLM(logmean~b\*log(xxx/mTT+1)/(1+exp(-(xxx-mu)/tau)), start=list(b=0.1, mu=10, tau=taustart, mTT=5))

mb=coef(fitmean)[1]

mmu=coef(fitmean)[2]

mt=coef(fitmean)[3]

mTT=coef(fitmean)[4]

fitsd=glm(sd~logmean -1 )

sdapprox=mb\*coef(fitsd)

meanr=exp(mb\*log(xxx/mTT+1)/(1+exp(-(xxx-mmu)/mt)))

lclr=exp((mb-1.96\*sdapprox)\*log(xxx/mTT+1)/(1+exp(-(xxx-mmu)/mt)))

uclr=exp((mb+1.96\*sdapprox)\*log(xxx/mTT+1)/(1+exp(-(xxx-mmu)/mt)))

#plot approx GEMM and CI

x=seq(0, MM, by=bb)

plot(x, uclr, lwd=4, type="l", col="lightgrey", ylab="Relative Risk", xlab= expression(paste("PM"[2.5], " - ", mu, "g/m"^3)))

polygon(x=c(x, rev(x)), y=c(lclr, rev(uclr)), col="lightgrey", lty=2, border=NA)

lines(x, meanr, lwd=4, col="red")

abline(1,0)

cbind(mb, sdapprox, mTT, mmu, mt)