```
# Packages ------------------------------------------------------------
install.packages("data.table")
install.packages("tidyverse")
install.packages("rSPARCS")
install.packages("imputeTS")
install.packages("openxlsx")
install.packages("splines")
install.packages("survival")
install.packages("metafor")
install.packages("reshape2")
install.packages("geosphere")
install.packages("dplyr")
install.packages("epiR")

library(epiR)
library(data.table)
library(tidyverse)
library(rSPARCS)
library(imputeTS)
library(openxlsx)
library(splines)
library(survival)
library(metafor)
library(reshape2)
library(geosphere)
library(dplyr)

# Directories ---------------------------------------------------------
setwd("P:/Staff/sathvik/Rochester")
```

```
exposure_path <- 'P:/Staff/sathvik/Rochester/Exposure'


# Functions ------------------------------------------------------------


CXover.data=function(data,date,ID=NULL,direction="month4",apart=7){
  name=names(data)
  if(length(ID)==0) data$ID=1:nrow(data) else data$ID=factor(data[,ID],levels=unique(data[,ID]))
  test=as.character(data[,date])
  test=test[test!=""&!is.na(test)]
  if(any(!is.na(grep("/",test[1])),!is.na(grep("-",test[1])))) data$d1=as.Date(as.character(data[,date])) else
data$d1=as.Date(paste(substr(data[,date],1,4),"/",substr(data[,date],5,6),"/",substr(data[,date],7,8),sep=
""))
  data$date1=as.character(data$d1-28)
  data$date2=as.character(data$d1-21)
  data$date3=as.character(data$d1-14)
  data$date4=as.character(data$d1-7)
  data$date5=as.character(data$d1+7)
  data$date6=as.character(data$d1+14)
  data$date7=as.character(data$d1+21)
  data$date8=as.character(data$d1+28)
  data$date1_1=substr(data$date1,6,7)
  data$date2_1=substr(data$date2,6,7)
  data$date3_1=substr(data$date3,6,7)
  data$date4_1=substr(data$date4,6,7)
  data$date5_1=substr(data$date5,6,7)
  data$date6_1=substr(data$date6,6,7)
  data$date7_1=substr(data$date7,6,7)
  data$date8_1=substr(data$date8,6,7)
  if(direction=="pre4") data$date5=data$date6=data$date7=data$date8=NA
```

```
 if(direction=="after4") data$date1=data$date2=data$date3=data$date4=NA
 if(direction=="month4"){
  data$d1_1=substr(data$d1,6,7)
  data$date1=ifelse(data$date1_1==data$d1_1,data$date1,NA)
  data$date2=ifelse(data$date2_1==data$d1_1,data$date2,NA)
  data$date3=ifelse(data$date3_1==data$d1_1,data$date3,NA)
  data$date4=ifelse(data$date4_1==data$d1_1,data$date4,NA)
  data$date5=ifelse(data$date5_1==data$d1_1,data$date5,NA)
  data$date6=ifelse(data$date6_1==data$d1_1,data$date6,NA)
  data$date7=ifelse(data$date7_1==data$d1_1,data$date7,NA)
  data$date8=ifelse(data$date8_1==data$d1_1,data$date8,NA)
 }
 if(apart==14) data$date2 = data$date4=data$date5 = data$date7=NA
 output=data[,c("ID","d1",name)]
 names(output)[2]="Date"
 output$status=1
 for(j in 1:8){
  test=data[,c("ID",paste("date",j,sep=""),name)]
  names(test)[2]="Date"
  test=test[which(!is.na(test$Date)),]
  if(nrow(test)>0){
   test$status=0
   output=rbind(output,test)
  }
  output=output[order(output$ID),]
 }
 rownames(output)=1:nrow(output)
 return(output)
}
```

```
pick.cases<-function(data,long.case,lat.case,long.sites,lat.sites,radius="15 miles"){
  data$which.site=NA
  a=ncol(data)
  for(i in 1:length(long.sites)){
    data$var=NA
    data$var=distGeo(data[,c(long.case,lat.case)],c(long.sites[i],lat.sites[i]))*0.000621371
    if(gregexpr("km",radius)[[1]][1]>0|gregexpr("kms",radius)[[1]][1]>0) data$var=data$var*1.60934
    if(i==1) data$which.site[!is.na(data$var)]=1
    if(i==1) names(data)[ncol(data)]=paste("distance.site",i,sep="")
    if(i>1) data$which.site=ifelse(data$distance.site1<data$var,data$which.site,i)
    if(i>1) data$distance.site1=ifelse(data$distance.site1<data$var,data$distance.site1,data$var)
  }
  data=cbind(data[,1:a],data[,"distance.site1"])
  names(data)[ncol(data)]="minDIST"
  a=gregexpr("km",radius)[[1]][1]
  b=gregexpr("kms",radius)[[1]][1]
  c=gregexpr("mile",radius)[[1]][1]
  d=gregexpr("miles",radius)[[1]][1]
  cut=as.numeric(substr(radius,1,unique(c(a,b,c,d)[c(a,b,c,d)>0])-2))
  data$Select=0
  data$Select[data$minDIST<=cut]=1
  return(data)
}

cxdata.exposure <- function(data,exposure,sites,col){
  exposure <- as.data.frame(exposure)
```

```
  data <- as.data.frame(data)
  data$ADMDT <- as.numeric(data$ADMDT)
  pick <- pick.cases(data,long.case = 'long',lat.case = 'lat',long.sites = sites$Lon,lat.sites = sites$Lat,radius
= '15 miles')
  sites$ID=1:nrow(sites)
  pick$FIPS=sites$Sites[match(pick$which.site,sites$ID)]
  pick$FIPS <- substr(pick$FIPS,1,3)
  pick <- subset(pick,Select == 1)
  cxdata=CXover.data(pick,date = 'ADMDT')
  cxdata$mcode <- paste0(gsub('-','',cxdata$Date),'-',toupper(cxdata$FIPS))
  exposure$date <- gsub('-','',exposure$date)
  exposure$date <- as.numeric(exposure$date)
  exposure$site <- as.character(exposure$site)
  exposure <- arrange(exposure,site,date)
  exposure <- lag_create(data=exposure,maxlag = 6,col = col,time = exposure$date,stid = exposure$site)
  exposure$mcode <- paste0(gsub('-','',exposure$date),'-',exposure$site)
  exposure <- select(exposure,-date,-site)
  exposure <- as.data.table(exposure)
  setkey(exposure,'mcode')
  cxdata <- as.data.table(cxdata)
  setkey(cxdata,'mcode')
  cxdata <- exposure[cxdata]
  cxdata <- as.data.frame(cxdata)
  cxdata <- arrange(cxdata,ID)
  return(cxdata)
}

lag_create <- function(data,maxlag,col,time,stid){
  mydata <- arrange(data,stid,time)### arrange the data before use this function!!!!
```

```
  start_col <- ncol(mydata)+1
  l <- length(col)
  mydata[,start_col:(start_col+(maxlag-1)*2*l+2*l-1)] <- NA
  for(j in unique(stid)){
    med=mydata[stid==j,]
    for (i in 1:maxlag-1) {
      med[,(start_col+i*2*l):(start_col+i*2*l+l-1)]=stats::filter(med[,col],rep(1/(i+1+1),(i+1+1)), sides=1)#
cumulative lag
      med[,(start_col+i*2*l+l):(start_col+i*2*l+2*l-1)]=stats::filter(med[,col],c(rep(0,i+1),1), sides=1)#
single lag
    }
    mydata[stid==j,]=med
  }
  name_col <- c(paste0(colnames(mydata[,col]),'_cu'),paste0(colnames(mydata[,col]),'_si'))
  colnames(mydata)[start_col:(start_col+(maxlag-1)*2*l+2*l-1)] <-paste0(rep(name_col,maxlag),
                                                    '_lag',rep(1:maxlag,length(name_col)) %>% sort())
  return(mydata)
}
```

```
whole.period=function(ICD=NULL,subtype,x,cxdata,overall=F){
  temp=subset(cxdata,FIPS=='Alb')
  if(x=='PM25'){
    temp <-
select(temp,PM25,PM25_cu_lag1,PM25_cu_lag2,PM25_cu_lag3,PM25_cu_lag4,PM25_cu_lag5,PM25_c
u_lag6)
```

```
}else{

  temp <- select(temp,O3,O3_cu_lag1,O3_cu_lag2,O3_cu_lag3,O3_cu_lag4,O3_cu_lag5,O3_cu_lag6)
}

temp=melt(temp,measure.vars = names(temp))
temp=aggregate(value~variable,temp,FUN<- function(u) {IQR(u,na.rm = T)})
fixed.IQR=c(rep(temp[1,2],7),temp[2:7,2])
fixed.IQR=fixed.IQR[7:13]
if(overall==F){data_med <- subset(cxdata,PD3%in%ICD)}else{
  data_med <- cxdata
}

data_med=data_med[data_med$ADMDT>=20140101,]
data_med$period[data_med$ADMDT>=20140101]="Overall"

output=as.data.frame(matrix(NA,7,4))
names(output)=c("lag","IQR","Cases",
                'PM25')
output$lag=0:6
lagidx=c('',paste0(rep('_cu_lag'),1:6))

for (lag in 0:6) {
  cat(x,'-',lag,'\n')
  if(x=='PM25'){
    data_med$PM25=data_med[,paste0('PM25',lagidx[lag+1])]
  }else{
    data_med$O3=data_med[,paste0('O3',lagidx[lag+1])]}
```

```
  data_med$RH=data_med[,paste0('RH',lagidx[lag+1])]
  data_med$temp=data_med[,paste0('temp',lagidx[lag+1])]

  model_adj = clogit(status ~ PM25 + ns(RH, df = 4) + ns(temp, df = 4) + strata(ID) + holiday, data_med)
  result=summary(model_adj)
  output$"Cases"[lag+1]=format(result$nevent, big.mark=",")

  #####common IQR
  output[lag+1,"IQR"]=fixed.IQR[lag+1]
  OR=sprintf("%.1f",round((exp(result$coefficients[c("PM25"),"coef"]*fixed.IQR[lag+1])-1)*100,4))
  L=sprintf("%.1f",round((exp((result$coefficients[c("PM25"),"coef"]-
1.96*result$coefficients[c("PM25"),"se(coef)"])*fixed.IQR[lag+1])-1)*100,4))

H=sprintf("%.1f",round((exp((result$coefficients[c("PM25"),"coef"]+1.96*result$coefficients[c("PM25"),"
se(coef)"])*fixed.IQR[lag+1])-1)*100,4))

  output[lag+1,'PM25']=paste(OR," (",L,", ",H,")",sep="")
  # output[lag+1,c("P value before","P value during","P value after",'P value
recent')]=sprintf("%.3f",round(result$coefficients[c("periodBefore:PM25","periodDuring:PM25","period
After:PM25","periodRecent:PM25"),"Pr(>|z|)"],3))
 }

 # write.csv(output,paste0('C:\\Users\\xd129664\\OneDrive - University at Albany -
SUNY\\Ualbany\\Ualbany-
Rochester\\Results\\period2_interaction_',subtype,'_',x,'.csv'),row.names=F)#change region added
 return(output)

}

#No Interaction Terms Involved
temperature.interaction.subtypes = function(ICD = NULL, subtype, x, cxdata, overall = FALSE) {
```

```
 temp = subset(cxdata, FIPS == 'Alb')
 temp <- select(temp, PM25, PM25_cu_lag1, PM25_cu_lag2, PM25_cu_lag3, PM25_cu_lag4,
PM25_cu_lag5, PM25_cu_lag6)
 temp = melt(temp, measure.vars = names(temp))
 temp = aggregate(value ~ variable, temp, FUN = function(u) { IQR(u, na.rm = TRUE) })
 fixed.IQR = c(rep(temp[1, 2], 7), temp[2:7, 2])
 fixed.IQR = fixed.IQR[7:13]

 if (overall == FALSE) {
  data_med <- subset(cxdata, PD3 %in% ICD)
 } else {
  data_med <- cxdata
 }

 data_med = data_med[data_med$ADMDT >= 20140101,]

 output = as.data.frame(matrix(NA, 7, 5))
 names(output) = c("lag", "Cases", "IQR",
                   "D10", "D90")

 output$lag = 0:6
 output$Lagname = ifelse(output$lag == 0, "0", paste("0-", output$lag, sep = ""))
 lagidx = c('', paste0(rep('_cu_lag'), 1:6))

 for (lag in 0:6) {
  cat(x, '-', lag, '\n')
  data_med$PM25 = data_med[, paste0('PM25', lagidx[lag + 1])]
  data_med$RH = data_med[, paste0('RH', lagidx[lag + 1])]
  data_med$temp = data_med[, paste0('temp', lagidx[lag + 1])]
```

```
data_med$year_month = substr(data_med$ADMDT, 1, 6)



threshold <- data_med %>%
  group_by(FIPS, year_month) %>%
  dplyr::summarise("temp_10" = quantile(temp, c(0.10), na.rm = TRUE),
                   "temp_90" = quantile(temp, c(0.90), na.rm = TRUE))

# Merge percentiles back into the original data
data_med <- merge(data_med, threshold, by = c("FIPS", "year_month"))

# Create dummy variables D10 and D90, PM2.5
data_med$D10 <- ifelse(data_med$temp <= data_med$temp_10, 1, 0)
data_med$D90 <- ifelse(data_med$temp >= data_med$temp_90, 1, 0)

# Models with PM25 as a continuous variable and dummy variables
model_adj1 <- clogit(status ~ PM25 + D10 + D90 + ns(RH, df = 4) + ns(temp, df = 4) + strata(ID) +
holiday, data_med)

# Print model summary
result = summary(model_adj1)
output$"Cases"[lag+1]=format(result$nevent, big.mark=",")



output[lag+1,"IQR"]=fixed.IQR[lag+1]
OR=sprintf("%.4f",round(exp(result$coefficients[c("D10", "D90"),"coef"]*fixed.IQR[lag+1]),4))
L=sprintf("%.4f",round(exp((result$coefficients[c("D10", "D90"),"coef"]-
1.96*result$coefficients[c("D10", "D90"),"se(coef)"])*fixed.IQR[lag+1]),4))
H=sprintf("%.4f",round(exp((result$coefficients[c("D10",
"D90"),"coef"]+1.96*result$coefficients[c("D10", "D90"),"se(coef)"])*fixed.IQR[lag+1]),4))
```

```
   output[lag+1,c("D10", "D90")]=paste(OR," (",L,", ",H,")",sep="")

   data_med <- select(data_med, -temp_90, -temp_10, -D10, -D90)

   # output[lag+1,c("P value before","P value during","P value after",'P value
recent')]=sprintf("%.3f",round(result$coefficients[c("periodBefore:PM25","periodDuring:PM25","period
After:PM25","periodRecent:PM25"),"Pr(>|z|)"],3))

  }

  # output[lag+1,c("P value before","P value during","P value after",'P value
recent')]=sprintf("%.3f",round(result$coefficients[c("periodBefore:PM25","periodDuring:PM25","period
After:PM25","periodRecent:PM25"),"Pr(>|z|)"],3))

  # write.csv(output,paste0('C:\\Users\\xd129664\\OneDrive - University at Albany -
SUNY\\Ualbany\\Ualbany-
Rochester\\Results\\period2_interaction_',subtype,'_',x,'.csv'),row.names=F)#change region added

  return(output)

}
```