STA442 Assignment 2

Jiayue_Wu

10/29/2021

Question 1: CO2 concentration

How is CO2 concentration behaving during economy depression period?

1. Introduction:

There are two typical cases of the globally decrease of economy: first, the fall of Berlin Wall in November 1989 leading to a dramatic decrease in the industrial production in Soviet Union and Eastern Europe, and the second one is the global lockdown preceded by the COVID-19 pandemic starting in February 2020, shutting down the global economy much. Moreover, it is known to all that as the economic increase, more natural resources are required for the increased output, and the pollution levels may increase at the same time. To investigate whether the economic depression affect the CO2 concentration or not, we will discuss the CO2 concentration in Hawaii with data available by the Scripps CO2 program at scrippsco2.ucsd.edu for the two-recession mentioned above. More precisely, the data we used is from http://scrippsco2.ucsd.edu/assets/data/atmospheric/, with 2502 observations and 7 variables in Haiwaii from March 30, 1960 to Sept 28, 2020.

2. Method + Model

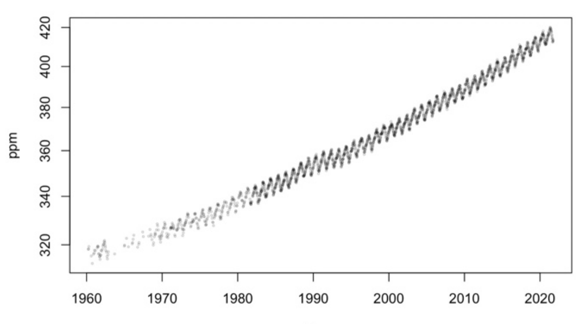

Figure 1(a): CO2 at Mauna Loa Observatory, Hawaii overall all years

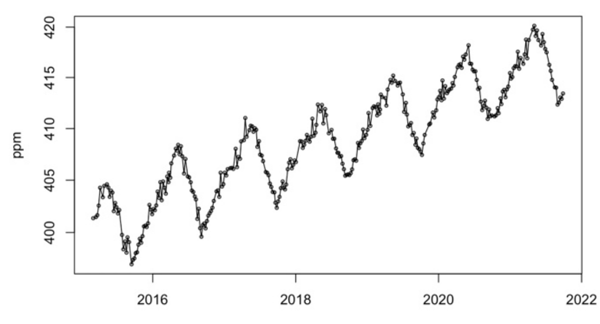

Figure 1(b): CO2 at Mauna Loa Observatory, Hawaii in recent years

As the CO2 concentration should be positive and is continuous, I assume the response variables — average daily concentration of CO2, follows the Normal distribution instead of Gamma or Poisson, here I assume the value only follows the distribution of Normal in the positive x-axis.
I estimated a generalized addictive model to model the atmospheric Carbon Dioxide concentration. Reasons are the following: Firstly, from the seasonal pattern on Figure 1, we can find that there exists seasonal trending for the CO2 concentration, with the fluctuating pattern, I will imitate them as the fixed variables in my model by the trigonometric function. More precisely, I will imitate it by the cos12, sin12, cos6 and sin6 functions. The fixed effects are the following:

$$X_{i0} = 1$$

$$X_{i1} = cos12 = cos(2\pi \times timeYears)$$

$$X_{i2} = sin12 = sin(2\pi \times timeYears)$$

$$X_{i3} = cos6 = cos(2 \times 2\pi \times timeYears)$$

$$X_{i4} = sin6 = sin(2 \times 2\pi \times timeYears)$$

Here timeYears is the normalized time variable with 365.25 days as standard deviation and the original date is January 1, 2000.

Moreover, by observing the Figure 1(a) and Figure 2(b), there exists a clear upward trend and it change every moment for both, thus I also added a RW(2) function as random effect, which is for smoothing the Random slopes for the seasonal trend.
The detail for the second-order random walk U_{t_i} is:

$$U_{t+1_i} | U_k, k < t \sim N(-2U_{t_i} + U_{t-1_i}, \tau_i^2)$$

$$(U_{t+1_i} - U_{t_i}) - (U_{t_i} - U_{t-1_i}) \sim N(0, \tau_i^2)$$

$$U_{t+1_i} - 2U_{t_i} + U_{t-1_i} \sim N(0, \tau_i^2)$$

i.e., the RW(2) represents $U_{t_i} = U_{0_i} + t \cdot (U_{1_i} - U_{0_i})$ is a straight line with upward trend (slope > 0) and I predicted it will be similar to the features in the Figure.

Moreover, the random slope function I defined has the prior distribution as following:

$$[U_1, \dots, U_T]^T \sim RW2(0, \sigma_U^2)$$

Here the prior distribution I set for the standard deviation of RW2 is:

$$Prob(\sigma_U > 0.001) = 0.5$$

Hence, the mathematical expression of the Generalized Addictive Model I fitted is:

$$Y_i \sim N(\mu_i, \sigma_i^2)$$

$$\mu_i = X_i\beta + U_{t_i}$$

Where Y_i is the averaged CO2 concentration for day i, with prior distribution: $Prob(\sigma_i > 1) = 0.5$. μ_i = expected daily CO2 concentration, X_i = the fixed variable, i.e., the seasonal pattern I set for the function, β = corresponding parameters for the fixed variable listed above, U_{t_i} = RW2, i.e., the second-order random walk I set as random effect for smoothing the data and random slopes of seasonal trend.

3. Results

I predicted the variation and track the path of random effect of CO2 as well as the derivative of it based on the model I fit in Part 2. I also graphed a 95% Confidence Interval for both, i.e., the interval of dotted lines in Figure 2(a) as well as the faded black line interval presented in Fiugre 2(b). The figures are following.

Figure 2(a): Prediction for the random effect of CO2 concentration

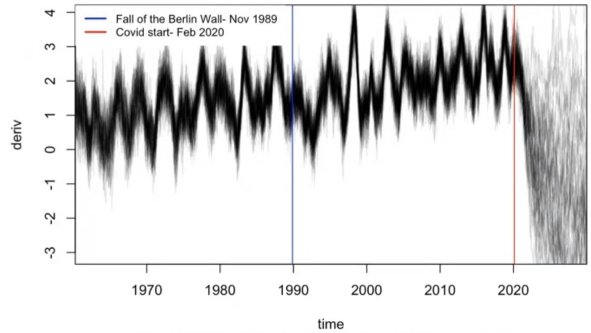

Figure 2(b): Prediction for the derivative of CO2 concentration

Here I draw two vertical lines for the random effect as well as the derivative in Figure 2(a) and Figure 2(b) to discuss the trend of atmospheric Carbon Dioxide concentration. The timeline for the two vertical lines are in November 1989 in blue and February 2020 in red, respectively.

Analysis for Nov 1978, the fall of Berlin Wall：

From the blue line presented in Figure 2(a), we can find that the random effect of CO2 concentration in 1989 is still increase compared to previous, while the rate may be slower than before, i.e., the random effect had a slower increase in Nov, 1989. A more clear insight for the increase rate of CO2 is shown in Figure 2(b), where the point intercepts with the blue line shows a smaller derivative than previous. Even if the derivative is not negative, i.e., the CO2 concentration did not decreased in Nov, 1989, the smaller derivative of that time also indicate a slower increment of CO2 concentration.

The slower increase of CO2 in November 1989 could be due to the drmatic fall in industrial production in Soviet Union and Eastern Europe caused by the fall of Berlin wall, which decreased the emission of carbon dioxide from industries at that time as well.

Analysis for February 2020, the starting of COVID-19：

Based on the red line in Figure 2(a), it could be surprised that the concentration of CO2 kept increasing as usual even if there started a global lockdown. And the derivative value in November 2020 shown by the interception position of red line in Figure 2(b) also in a relatively high position, even if it also in a low number compared to previous. However, there was a dramatic fall later on predicted by the model we built. The predicted trend of derivative also shows a dramatic fall after February 2021.

Even if the behaviors of February 2021 showed in Figure 2(a) and Figure 2(b) do not indicate a decrease in concentration of CO2 at the exact point, the prediction show a dramatic fall for both the random effect and the derivative. With the decreasing random effect, the intercept of CO2 concentration also decrease, which means the start level of concentration is lower. As the derivative becomes negative, the concentration of CO2 will decrease.

Hence, there maybe exist a lag for the concentration of CO2 and the lockdown of COVID-19, since the February 2021 is the start of lockdown, it may take some time to spread the effect and lockdown globally.

4. Summary

Based on the graphs of random effect as well as the derivative, we can notice that the overall trend of concentration of CO2 is increase. While during the intensive economic shut down period such as in November 1989 and February 2020, the increase of atmospheric Carbon Dioxide concentration became slower and even decrease after February 2020. There exists a negative effect on the Carbon Dioxide concentration from both of the two economy depression periods. And there may exist a lag for the decreasing concentration of CO2 for the COVID-19 impact as we analyzed before.

Question 2: Death

Whether COVID-19 affect the death number of neoplasms, heart, Accidents, and respiratory, respectively.

1. Introduction

With the necessary of masks, social distancing, and decreased air pollution for preventing the COVID-19 pandemic, there are two hypotheses hold by people that (1) The mortality of Malignant neoplasms (cancers) and diseases of heart (heart attacks) have increased during the covid-19 lockdown. At the same time, as the less access to healthcare, the pandemic lockdown may also cause (2) The mortality of Accidents (unintentional injuries) and chronic lower respiratory diseases have decreased during the covid-19 lockdown. To investigate whether the four types of disease has similar outcomes as people predicted, we will use the daily cause-specific mortality counts by province which is available from Statistics Canada at https://open.canada.ca/data/en/dataset/aed00edc-26ad-414c-8aa3-82212059ef8a. We will focus on quantifying the change of mortality in Ontario for each of the four types from March to November 2020 for the four types listed.

2. Method

Firstly, I plotted 4 scatterplots indicating the weekly mortality number for each four diseases listed from 2010 to 2020. Here we can notice that the number of mortality seem to be a bit increase for neoplasms as well as the Accidents, the trend of respiratory mortality has a decrease indication, and the trend of heart in the graph does not show a clear trend of increase or decrease. Hence, even if with a rough insight of the trend, we need a more precise evidence or trend to prove our hypothesis.

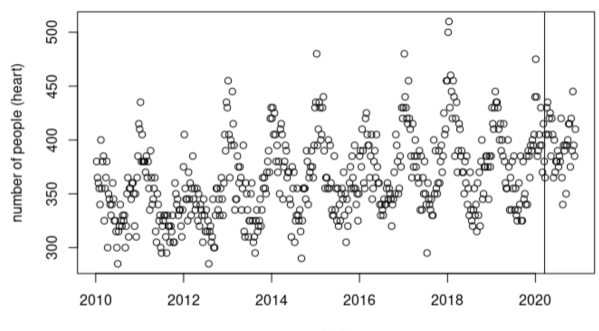

Figure 1(a): Mortality for heart disease

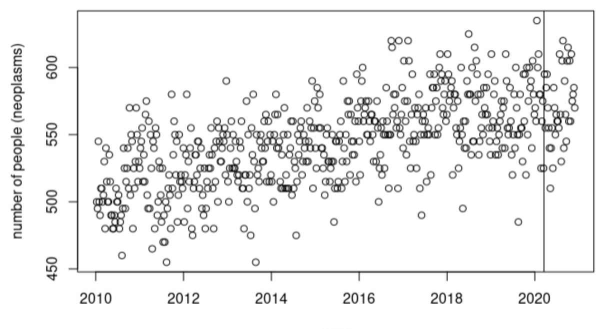

Figure 1(b): Mortality for neoplasms

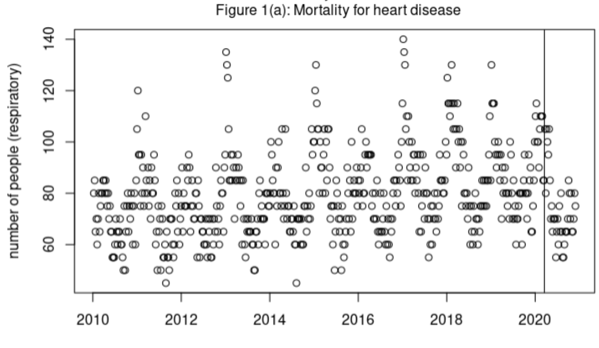

Figure 1(c): Mortality for respiratory

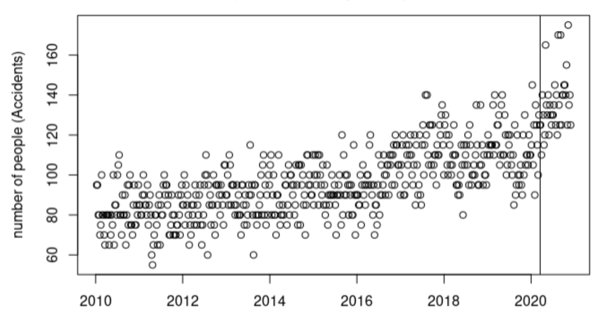

Figure 1(d): Mortality for Accidents

For a more precise result, I estimated a Generalized Addictive Model to investigate the question I mentioned before.

$$Y_i \sim Poisson(\lambda_i)$$

Here Y_i refers to the weekly mortality number. And I assume that the weekly death number follows a poisson distribution, and it is reasonable since the death number will always nonnegative and is discrete. λ_i represents the expected number of people die in i month in Ontario,

$$\log(\lambda_i) = X_i\beta + V_i + U_{t_i}$$

From all the trend shown on all Figure 1s, there exist a slightly fluctuating term, especially in the graph of heart disease as well as the one in respiratory. And it is also reasonable to assume that there may exist seasonal trend or seasonal difference for the severity of all kinds of disases. Hence, I imitate the annual cycles as the fixed variables in my model by the trigonometric function. More precisely, I will imitate it by the cos12, sin12, cos6 and sin6 functions. i.e., I use a 12 month and a 6 month frequency. The fixed effects are the following:

$$X_{i0} = 1$$

$$X_{i1} = cos12 = cos(2\pi \times dateInt/365.25)$$

$$X_{i2} = sin12 = sin(2\pi \times dateInt/365.25)$$

$$X_{i3} = cos6 = cos(2\pi \times dateInt \times 2/365.25)$$

$$X_{i4} = sin6 = sin(2\pi \times dateInt \times 2/365.25)$$

Here *dateInt* represents numeric time variable, and I also normalized the date with 365.25 days as standard deviation, sicne there is one cycle every 365.25 days, and the original date is February 1, 2000.

U_i represents the random effect of date, in this model, I assume the random effect V_i follows a normal distribution, i.e.,

$$V_i \sim N(0, \sigma_v^2)$$

the prior distribution we set for the standard deviation is:

$$P(\sigma_v > \log(1.25)) = 0.5$$

Aforementioned, I also added a RW(2) function as random walk effect, since the x-axis we set is time, which is for smoothing the Random slopes for the seasonal trend.
The detail for the second-order random walk U_{t_i} is:

$$U_{t+1_i}|U_k, k < t \sim N(-2U_{t_i} + U_{t-1_i}, \sigma_i^2)$$

$$(U_{t+1_i} - U_{t_i}) - (U_{t_i} - U_{t-1_i}) < t \sim N(0, \sigma_i^2)$$

$$U_{t+1_i} - 2U_{t_i} + U_{t-1_i} < t \sim N(0, \sigma_i^2)$$

the prior distribution we set for the standard deviation is:

$$P(\sigma_U > 0.1) = 0.5$$

3. Results

I plotted 4 kinds of graphs for each of the four listed diseases separately as following:

Figure 2s

First, I graphed the weekly mortality number for each of the disease as well as its prediction interval by the model we fit. Here the blue dots are the mortality before Covid, the red dots include the mortality of Christmas, which we removed from the model we fit for this special festival and its outlier behavior as shown in the graph, as well as the mortality of after Covid. Observing these plots, we can find that the blue dots are located between the dotted lines, which means the prediction interval from model we fit can capture most of the mortality number in reality, model we fit is suitable from the Figure 2s.

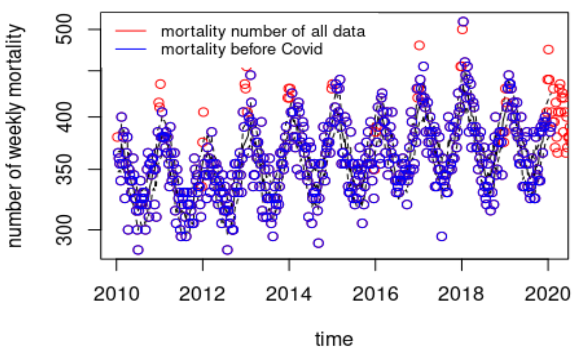

Figure 2(a): The Prediction interval of mortality (heart)

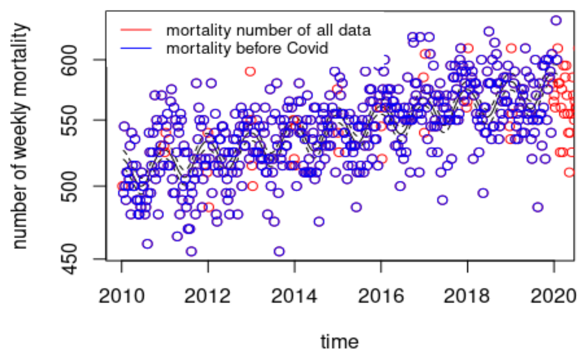

Figure 2(b): The Prediction interval of mortality (Neoplasms)

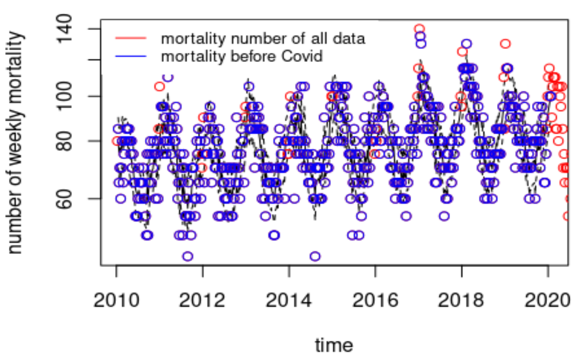

Figure 2(c): The Prediction interval of mortality (respiratory)

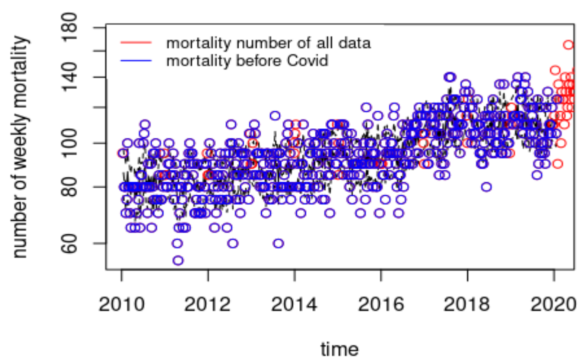

Figure 2(d): The Prediction interval of mortality (Accidents)

Figure 3s

Second, I also plot the random effect out to investigate how is their behavior for each of the four diseases. Here from Figure 3s, we can find that the truly random effect also included in the dotted lines, the prediction interval also captures a great random effect for the four listed diseases. However, we can find that the dotted line after COVID-19 spread a lot, which means the model does not fit the random effect as well as we expected, i.e., the lower bound of the 95% prediction interval is so low and the upper bound of 95% prediction interval is so high that it provide little useful information for us to predict the future random effect.

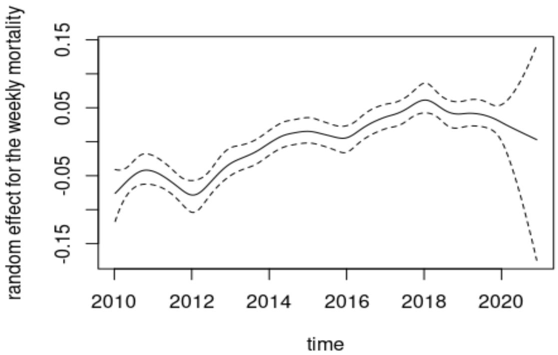

Figure 3(a): The random effect for the weekly mortality (heart)

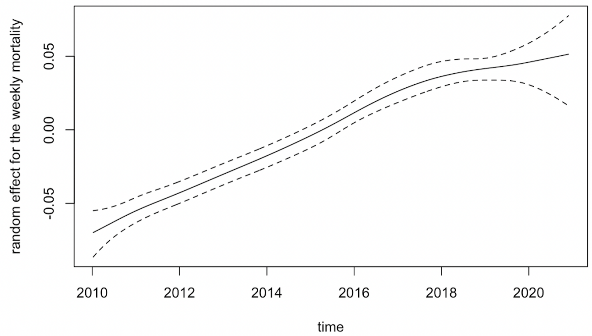

Figure 3(b): The random effect for the weekly mortality(Neoplasms)

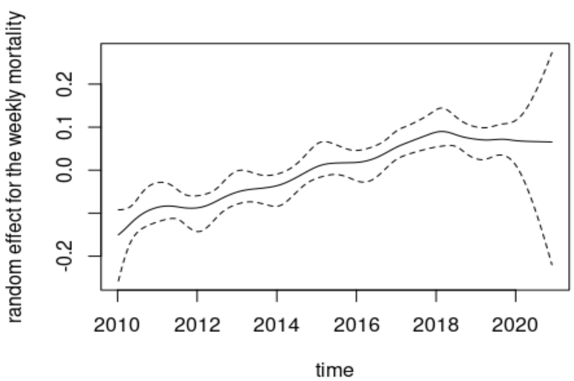

Figure 3(c): The random effect for the weekly mortality (respirator

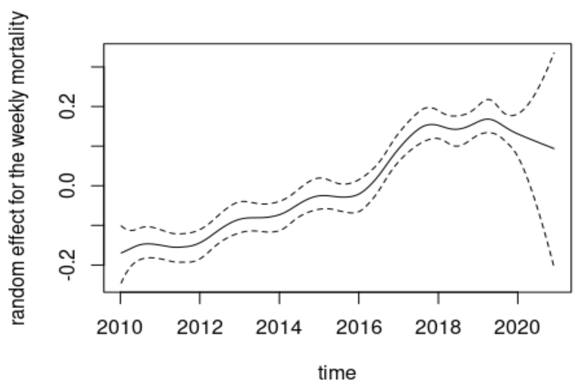

Figure 3(d): The random effect for the weekly mortality (Accident

Figure 4s

Third, for a better confirm for the prediction of the fitted model, I fit the model with pre-COVID data and used the model creating an ensemble of 100 different forecasts for the March – November under the circumstances without COVID. Below the Figure 4s are the behavior of those 100 forecasts. Here we can find that the 100 forecasts perfectly capture and include the truly weekly mortality number for all the four listed diseases. Here I also built a yellow line to separate the mortality before COVID and within COVID period, where we can find that with the exist of COVID, the mortality number of heart disease as well as the Accidents does not match with the prediction model which works well before COVID. While we cannot determine whether there are differences between predicted and real death number for Neoplasms as well as the respiratory.

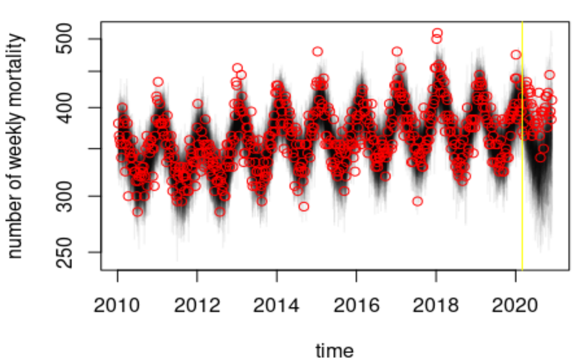

Figure 4(a): The random effect for the weekly mortality (heart)

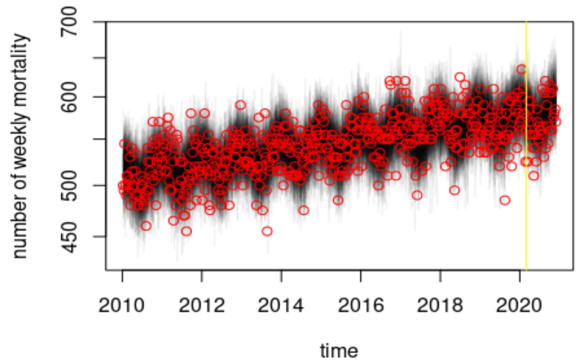

Figure 4(b): The random effect for the weekly mortality (Neoplasn

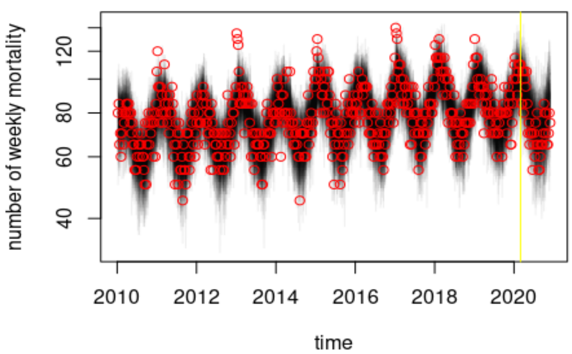

Figure 4(c): The random effect for the weekly mortality (respirator

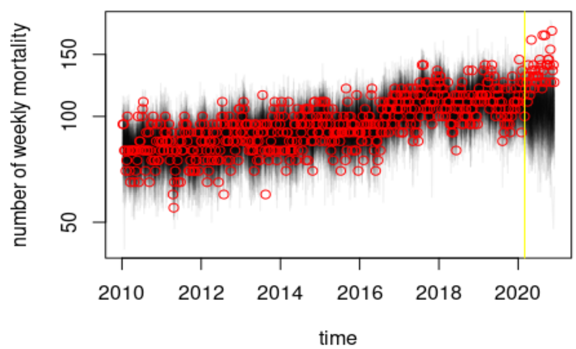

Figure 4(d): The random effect for the weekly mortality (Accident

Figure 5s

Lastly, to get a clearer insight of the difference between the reality death and predicted death, I draw bellowing four graphs (Figure 5s) indicating the excess death than prediction after COVID, i.e., death number in reality – predicted death number by model we fit. By observing the bellowing graphs, we can notice that there exist a clear and obvious trend above the 0-line for the heart disease (Figure 5(a)) as well as the Accidents (Figure 5(d)), where most of the prediction interval are above 0. As for the Neoplasms, we can notice that the prediction interval locates roughly around the 0-line and almost symmetric for most time from Figure 5(b). There may exist a decrease of death number for respiratory diseases, where from Figure 5(c) we can find that most excess mortality predicted shown are below the 0-line.

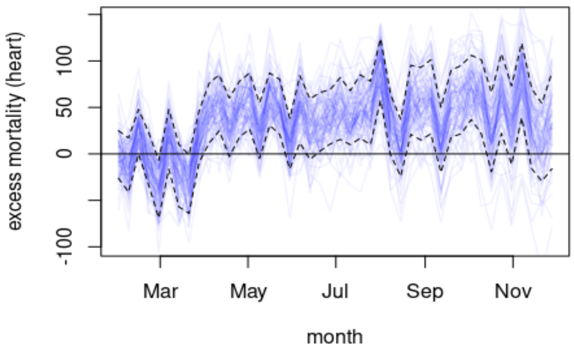

Figure 5(a): Excess mortality of Heart than predictoin

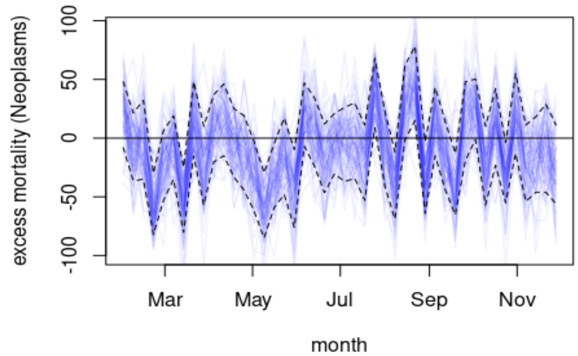

Figure 5(b): Excess mortality of Neoplasms prediction

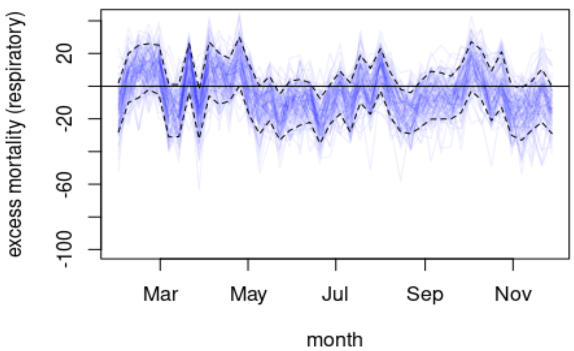

Figure 5(c): Excess mortality of respiratory prediction

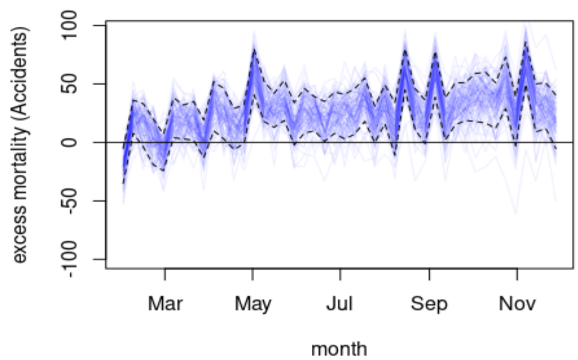

Figure 5(d): Excess mortality of Accidents prediction

4. Summary

Based on the confirmation of model we fitted above as well as the excess death number for four diseases listed, we can find that there exist an increase death number for heart disease as well as the Accidents. The weekly mortality number of Neoplasms stay as before COVID. The weekly death rate of respiratory may have decreased a little. In conclusion: (1) the mortality number of heart attack increase, which is consistent with the hypotheses; (2) the mortality number of neoplasms does not change that much before and after the COVID, the pandemic does not have an obvious impact on the death number of neoplasms, which is inconsistent with the hypotheses. However, this could be due to the long treatment period of neoplasms, there could be a lag bewteen the discovery and death of neoplasms. (3) the mortality number of respiratory decrease slightly after COVID, which is consistent with the hypotheses; (4) the mortality number of Accidents increase, which is inconsistent with the hypotheses. This could also due to the occupied healthcare resource by COVID-19 and little remaining healthcare for patients of Accidents.

Appendix Question 1:

```
# read data
cUrl=paste0("http://scrippsco2.ucsd.edu/assets/data/atmospheric/","stations/flask_co2/daily/daily_flask_co2_mlo.cs
v")
cFile=basename(cUrl)
if(!file.exists(cFile))download.file(cUrl, cFile)

co2s=read.table(cFile,header =FALSE,sep =",",skip =69,stringsAsFactors =FALSE,col.names =c("day","time","jun
k1","junk2","Nflasks","quality","co2"))
co2s$date=as.Date(co2s$day)
co2s$time=strptime(paste(co2s$day, co2s$time),format ="%Y-%m-%d %H:%M",tz ="UTC")
co2s=co2s[co2s$quality==0, ]
plot(co2s[co2s$date>as.Date("2015/3/1"),c("date","co2")],log ="y",type ="o",xlab ="time", ylab ="ppm",sub = "Fig
ure 1(a): Concentration of CO2 at Hawaii (1960 - 2020)")

plot(co2s[co2s$date>as.Date("2015/3/1"),c("date","co2")],log ="y",type ="o", xlab ="time",ylab ="ppm",cex =0.5, s
ub = "Figure 1(b): Concentration of CO2 at Hawaii (2015 - 2022)")

co2s$dateWeek=as.Date(lubridate::floor_date(co2s$date,unit ="week"))
co2s$timeYears=as.numeric(co2s$date)/365.25
co2s$cos12=cos(2*pi*co2s$timeYears)
co2s$sin12=sin(2*pi*co2s$timeYears)
co2s$cos6=cos(2*2*pi*co2s$timeYears)
co2s$sin6=sin(2*2*pi*co2s$timeYears)
allDays=seq(from =min(co2s$dateWeek),to =as.Date("2030/1/1"),by ="7 days")
#table(co2s$dateWeek%in%allDays)
co2s$dateWeekInt=as.integer(co2s$dateWeek)

library("INLA",verbose =FALSE)

mm=get("inla.models", INLA:::inla.get.inlaEnv())
if(class(mm)=="function") mm=mm()
mm$latent$rw2$min.diff=NULL
assign("inla.models", mm, INLA:::inla.get.inlaEnv())
co2res=inla(co2~sin12+cos12+sin6+cos6+f(dateWeekInt,model ="rw2",values =as.integer(allDays),prior ="pc.prec
",param =c(0.001,0.5),scale.model =FALSE),data =co2s,family ="gaussian",control.family =list(hyper =list(prec =li
st(prior ="pc.prec",param =c(1,0.5)))),control.inla =list(strategy ="gaussian"),control.compute =list(config =TRUE),
verbose =TRUE)
qCols=c("0.5quant","0.025quant","0.975quant")


if (!requireNamespace("BiocManager", quietly = TRUE))
   install.packages("BiocManager")
sampleList=INLA::inla.posterior.sample(50, co2res)
sampleMat=do.call(cbind, Biobase::subListExtract(sampleList,"latent"))
sampleMean=sampleMat[grep("dateWeekInt",rownames(sampleMat)),]
sampleDeriv=apply(sampleMean,2, diff)*(365.25/7)
forSinCos=2*pi*as.numeric(allDays)/365.25
forForecast=cbind(`(Intercept)`=1,sin12 =sin(forSinCos),
                 cos12 =cos(forSinCos),sin6 =sin(2*forSinCos),
                 cos6 =cos(2*forSinCos))
forecastFixed=forForecast%*%sampleMat[paste0(colnames(forForecast),":1"), ]
forecast=forecastFixed+sampleMean
```

```
##random slope
forX=as.Date(c("2018/1/1","2025/1/1"))
forX=seq(forX[1], forX[2],by ="1 year")
toPlot=which(allDays>min(forX)&allDays<max(forX))

matplot(allDays, co2res$summary.random$dateWeekInt[,qCols],type ="l",col ="black",lty =c(1,2,2),xlab ="time",ylab ="y",sub= "Figure 2(a): Prediction for the random effect of CO2 concentration")

abline(v = as.numeric(as.Date("1989-11-9")) , col = "blue",lwd = 3)
abline(v = as.numeric(as.Date("2020-02-9")) , col = "red",lwd=3)
legend( x="bottomleft",
        legend=c("Fall of the Berlin Wall- Nov 1989 ","Covid start- Feb 2020"),
        col=c("red","green"), lwd=2, cex=0.8,box.lty = 0)
```

```
matplot(allDays[-1], sampleDeriv,type ="l",lty =1,xaxs ="i",col ="#00000020",xlab ="time",ylab ="Differentiated concentration(ppm)",ylim =quantile(sampleDeriv,c(0.025,0.995)),sub="Figure 4 Random effect")
abline(v = as.numeric(as.Date("1989-11-9")) , col = "blue",lwd = 3)
abline(v = as.numeric(as.Date("2020-02-9")) , col = "red",lwd=3)
legend( x="bottomleft",
        legend=c("Fall of the Berlin Wall- Nov 1989 ","Covid start- Feb 2020"),
        col=c("red","green"), lwd=2, cex=0.8,box.lty = 0)
```

Appendix Question 2:

```
deadFile=Pmisc::downloadIfOld("https://www150.statcan.gc.ca/n1/tbl/csv/13100810-eng.zip")

deadFileCsv =deadFile[which.max(file.info(deadFile)$size)]

x[1:2, ]

x$date=as.Date(as.character(x[[grep("DATE",names(x))]]))
x$province=gsub("[,].*","", x$GEO)

x=x[x$date<as.Date("2020/12/01")&x$province=="Ontario", ]
colnames(x)[which(names(x) == "Cause of death (ICD-10)")] <- "Cause"

plot(x[grep("heart", x$Cause),c("date","VALUE")],ylab ="number of people (heart)",
   xlab="year",
   sub = "Figure 1(a): Mortality for heart disease")
abline(v =as.Date("2020/02/01"))

plot(x[grep("neoplasms", x$Cause),c("date","VALUE")],ylab ="number of people (neoplasms)",
   xlab="year",
   sub = "Figure 1(b): Mortality for neoplasms")
abline(v =as.Date("2020/02/01"))

plot(x[grep("respiratory", x$Cause),c("date","VALUE")],ylab ="number of people (respiratory)",
   xlab="year",
   sub = "Figure 1(c): Mortality for respiratory")
abline(v =as.Date("2020/02/01"))
```

```
plot(x[grep("Accidents", x$Cause),c("date","VALUE")],ylab ="number of people (Accidents)",
    xlab="year",
    sub = "Figure 1(d): Mortality for Accidents")
abline(v =as.Date("2020/02/01"))
dateSeq=sort(unique(x$date))
dateSeqInt=as.integer(dateSeq)
x$dateInt=x$dateIid=as.integer(x$date)
x$cos12=cos(2*pi*x$dateInt/365.25)
x$sin12=sin(2*pi*x$dateInt/365.25)
x$sin6=sin(2*2*pi*x$dateInt/365.25)
x$cos6=cos(2*2*pi*x$dateInt/365.25)
x$dayOfYear=as.Date(gsub("^[[:digit:]]+","0000",x$date))
x$christmasBreak=(x$dayOfYear>=as.Date("0000/12/21"))|(x$dayOfYear<=as.Date("0000/01/12"))


#plot for heart

xSub=x[grepl("heart ", x$Cause,ignore.case =TRUE) & x$province=="Ontario", ]
xPreCovid=xSub[xSub$date<as.Date("2020/02/01") & (!xSub$christmasBreak), ]
resHere=inla(VALUE~cos12+cos6+sin12+sin6+ f(dateInt,model ="rw2",values =dateSeqInt, prior ='pc.prec',param
=c(0.1, 0.5))+ f(dateIid,values =dateSeqInt,prior = 'pc.prec', param =c(log(1.25), 0.5)),data =xPreCovid, family ="po
isson",control.compute =list(config =TRUE), control.predictor =list(compute =TRUE))

matplot(resHere$.args$data$date, resHere$summary.fitted[, paste0(c(0.025,0.975,0.5),"quant")],type ="l", lty =c(2,
2,1),col ="black",log ="y",
       ylim =range(xSub$VALUE),
       xlab="time", ylab = "number of weekly mortality",
       sub = "Figure 2(a): The Prediction interval of mortality (heart)")
points(xSub$date, xSub$VALUE,col ="red")
points(xPreCovid$date, xPreCovid$VALUE,col ="blue")
legend( x="topleft",
       legend=c("mortality number of all data ","mortality before Covid"),
       col=c("red","blue"), lwd=1, cex=0.8,box.lty = 0)

matplot(dateSeq, resHere$summary.random$dateInt[,paste0(c(0.025, 0.975,0.5),"quant")],
       type ="l",lty =c(2, 2,1),col ="black",xlab="time",
       ylab = "random effect for the weekly mortality",
       sub = "Figure 3(a): The random effect for the weekly mortality(heart)")

toPredict=cbind(`(Intercept):1`=1,`cos12:1`=cos(2* pi*dateSeqInt/365.25),
              `sin12:1`=sin(2*pi*dateSeqInt/365.25),
              `cos6:1`=cos(2*pi*dateSeqInt*2/365.25),
              `sin6:1`=sin(2*pi*dateSeqInt*2/365.25))
dateIntSeq=paste0("dateInt:",1:length(dateSeqInt))
dateIidSeq=paste0("dateIid:",1:length(dateSeqInt))
resSample=inla.posterior.sample(n =100, resHere)
resSampleFitted=lapply(resSample,function(xx) {
  toPredict%*%xx$latent[colnames(toPredict), ]+
    xx$latent[dateIntSeq, ]+xx$latent[dateIidSeq,]})
resSampleFitted=do.call(cbind, resSampleFitted)
resSampleLambda=exp(resSampleFitted)
resSampleCount=matrix(rpois(length(resSampleLambda), resSampleLambda),nrow(resSampleLambda),ncol(resSa
mpleLambda))

matplot(dateSeq, resSampleCount,col ="#00000010", type ="l",lty =1,log ="y",
       xlab="time",
```

```
        ylab = "number of weekly mortality",
        sub = "Figure 4(a): The random effect for the weekly mortality(heart)")
points(xSub[,c("date","VALUE")],col ="red")
abline(v =as.Date("2020/03/01"),col ="yellow")

is2020=dateSeq[dateSeq>=as.Date("2020/2/1")]
sample2020=resSampleCount[match(is2020, dateSeq),]
count2020=xSub[match(is2020, xSub$date),"VALUE"]
excess2020=count2020-sample2020
matplot(is2020, excess2020,type ="l",lty =1,col ="#0000FF10", ylim =range(-100,quantile(excess2020,c(0.1,0.99
9))), xlab = 'month', ylab = 'excess mortality (heart)', sub = "Figure 5(a): Excess mortality of heart prediction")
matlines(is2020, t(apply(excess2020, 1, quantile, prob = c(0.1, 0.9))), col = "black", lty = 2)
abline(h = 0)



#plot for neoplasms

xSub=x[grepl("neoplasms", x$Cause,ignore.case =TRUE) & x$province=="Ontario", ]
xPreCovid=xSub[xSub$date<as.Date("2020/02/01") & (!xSub$christmasBreak), ]
resHere=inla(VALUE~cos12+cos6+sin12+sin6+ f(dateInt,model ="rw2",values =dateSeqInt, prior ='pc.prec',param
=c(0.1, 0.5))+ f(dateIid,values =dateSeqInt,prior = 'pc.prec', param =c(log(1.25), 0.5)),data =xPreCovid, family ="po
isson",control.compute =list(config =TRUE), control.predictor =list(compute =TRUE))

matplot(resHere$.args$data$date, resHere$summary.fitted[, paste0(c(0.025,0.975,0.5),"quant")],type ="l", lty =c(2,
2,1),col ="black",log ="y",
        ylim =range(xSub$VALUE),
        xlab="time", ylab = "number of weekly mortality",
        sub = "Figure 2(b): The Prediction interval of mortality (Neoplasms)")
points(xSub$date, xSub$VALUE,col ="red")
points(xPreCovid$date, xPreCovid$VALUE,col ="blue")
legend( x="topleft",
        legend=c("mortality number of all data ","mortality before Covid"),
        col=c("red","blue"), lwd=1, cex=0.8,box.lty = 0)

matplot(dateSeq, resHere$summary.random$dateInt[,paste0(c(0.025, 0.975,0.5),"quant")],
        type ="l",lty =c(2, 2,1),col ="black",xlab="time",
        ylab = "random effect for the weekly mortality",
        sub = "Figure 3(b): The random effect for the weekly mortality(Neoplasms)")

toPredict=cbind(`(Intercept):1`=1,`cos12:1`=cos(2* pi*dateSeqInt/365.25),
               `sin12:1`=sin(2*pi*dateSeqInt/365.25),
               `cos6:1`=cos(2*pi*dateSeqInt*2/365.25),
               `sin6:1`=sin(2*pi*dateSeqInt*2/365.25))
dateIntSeq=paste0("dateInt:",1:length(dateSeqInt))
dateIidSeq=paste0("dateIid:",1:length(dateSeqInt))
resSample=inla.posterior.sample(n =100, resHere)
resSampleFitted=lapply(resSample,function(xx) {
  toPredict%*%xx$latent[colnames(toPredict), ]+
    xx$latent[dateIntSeq, ]+xx$latent[dateIidSeq,]})
resSampleFitted=do.call(cbind, resSampleFitted)
resSampleLambda=exp(resSampleFitted)
resSampleCount=matrix(rpois(length(resSampleLambda), resSampleLambda),nrow(resSampleLambda),ncol(resSa
mpleLambda))

matplot(dateSeq, resSampleCount,col ="#00000010", type ="l",lty =1,log ="y",
```

```
        xlab="time",
        ylab = "number of weekly mortality",
        sub = "Figure 4(b): The random effect for the weekly mortality(Neoplasms)")
points(xSub[,c("date","VALUE")],col ="red")
abline(v =as.Date("2020/03/01"),col ="yellow")

is2020=dateSeq[dateSeq>=as.Date("2020/2/1")]
sample2020=resSampleCount[match(is2020, dateSeq),]
count2020=xSub[match(is2020, xSub$date),"VALUE"]
excess2020=count2020-sample2020
matplot(is2020, excess2020,type ="l",lty =1,col ="#0000FF10", ylim =range(-100,quantile(excess2020,c(0.1,0.99
9))), xlab = 'month', ylab = 'excess mortality (Neoplasms)', sub = "Figure 5(b): Excess mortality of Neoplasms predi
ction")
matlines(is2020, t(apply(excess2020, 1, quantile, prob = c(0.1, 0.9))), col = "black", lty = 2)
abline(h = 0)



#plot for respiratory

xSub=x[grepl("respiratory", x$Cause,ignore.case =TRUE) & x$province=="Ontario", ]
xPreCovid=xSub[xSub$date<as.Date("2020/02/01") & (!xSub$christmasBreak), ]
resHere=inla(VALUE~cos12+cos6+sin12+sin6+ f(dateInt,model ="rw2",values =dateSeqInt, prior ='pc.prec',param
=c(0.1, 0.5))+ f(dateIid,values =dateSeqInt,prior = 'pc.prec', param =c(log(1.25), 0.5)),data =xPreCovid, family ="po
isson",control.compute =list(config =TRUE), control.predictor =list(compute =TRUE))

matplot(resHere$.args$data$date, resHere$summary.fitted[, paste0(c(0.025,0.975,0.5),"quant")],type ="l", lty =c(2,
2,1),col ="black",log ="y",
        ylim =range(xSub$VALUE),
        xlab="time", ylab = "number of weekly mortality",
        sub = "Figure 2(c): The Prediction interval of mortality (respiratory)")
points(xSub$date, xSub$VALUE,col ="red")
points(xPreCovid$date, xPreCovid$VALUE,col ="blue")
legend( x="topleft",
        legend=c("mortality number of all data ","mortality before Covid"),
        col=c("red","blue"), lwd=1, cex=0.8,box.lty = 0)

matplot(dateSeq, resHere$summary.random$dateInt[,paste0(c(0.025, 0.975,0.5),"quant")],
        type ="l",lty =c(2, 2,1),col ="black",xlab="time",
        ylab = "random effect for the weekly mortality",
        sub = "Figure 3(c): The random effect for the weekly mortality(respiratory)")

toPredict=cbind(`(Intercept):1`=1,`cos12:1`=cos(2* pi*dateSeqInt/365.25),
                `sin12:1`=sin(2*pi*dateSeqInt/365.25),
                `cos6:1`=cos(2*pi*dateSeqInt*2/365.25),
                `sin6:1`=sin(2*pi*dateSeqInt*2/365.25))
dateIntSeq=paste0("dateInt:",1:length(dateSeqInt))
dateIidSeq=paste0("dateIid:",1:length(dateSeqInt))
resSample=inla.posterior.sample(n =100, resHere)
resSampleFitted=lapply(resSample,function(xx) {
  toPredict%*%xx$latent[colnames(toPredict), ]+
    xx$latent[dateIntSeq, ]+xx$latent[dateIidSeq,]})
resSampleFitted=do.call(cbind, resSampleFitted)
resSampleLambda=exp(resSampleFitted)
resSampleCount=matrix(rpois(length(resSampleLambda), resSampleLambda),nrow(resSampleLambda),ncol(resSa
mpleLambda))
```

```r
matplot(dateSeq, resSampleCount,col ="#00000010", type ="l",lty =1,log ="y",
        xlab="time",
        ylab = "number of weekly mortality",
        sub = "Figure 4(c): The random effect for the weekly mortality(respiratory)")
points(xSub[,c("date","VALUE")],col ="red")
abline(v =as.Date("2020/03/01"),col ="yellow")

is2020=dateSeq[dateSeq>=as.Date("2020/2/1")]
sample2020=resSampleCount[match(is2020, dateSeq),]
count2020=xSub[match(is2020, xSub$date),"VALUE"]
excess2020=count2020-sample2020
matplot(is2020, excess2020,type ="l",lty =1,col ="#0000FF10", ylim =range(-100,quantile(excess2020,c(0.1,0.99
9))), xlab = 'month', ylab = 'excess mortality (respiratory)', sub = "Figure 5(c): Excess mortality of respiratory predic
tion")
matlines(is2020, t(apply(excess2020, 1, quantile, prob = c(0.1, 0.9))), col = "black", lty = 2)
abline(h = 0)



#plot for Accidents

xSub=x[grepl("Accidents ", x$Cause,ignore.case =TRUE) & x$province=="Ontario", ]
xPreCovid=xSub[xSub$date<as.Date("2020/02/01") & (!xSub$christmasBreak), ]
resHere=inla(VALUE~cos12+cos6+sin12+sin6+ f(dateInt,model ="rw2",values =dateSeqInt, prior ='pc.prec',param
=c(0.1, 0.5))+ f(dateIid,values =dateSeqInt,prior = 'pc.prec', param =c(log(1.25), 0.5)),data =xPreCovid, family ="po
isson",control.compute =list(config =TRUE), control.predictor =list(compute =TRUE))

matplot(resHere$.args$data$date, resHere$summary.fitted[, paste0(c(0.025,0.975,0.5),"quant")],type ="l", lty =c(2,
2,1),col ="black",log ="y",
        ylim =range(xSub$VALUE),
        xlab="time", ylab = "number of weekly mortality",
        sub = "Figure 2(d): The Prediction interval of mortality (Accidents)")
points(xSub$date, xSub$VALUE,col ="red")
points(xPreCovid$date, xPreCovid$VALUE,col ="blue")
legend( x="topleft",
        legend=c("mortality number of all data ","mortality before Covid"),
        col=c("red","blue"), lwd=1, cex=0.8,box.lty = 0)

matplot(dateSeq, resHere$summary.random$dateInt[,paste0(c(0.025, 0.975,0.5),"quant")],
        type ="l",lty =c(2, 2,1),col ="black",xlab="time",
        ylab = "random effect for the weekly mortality",
        sub = "Figure 3(d): The random effect for the weekly mortality(Accidents)")

toPredict=cbind(`(Intercept):1`=1,`cos12:1`=cos(2* pi*dateSeqInt/365.25),
                `sin12:1`=sin(2*pi*dateSeqInt/365.25),
                `cos6:1`=cos(2*pi*dateSeqInt*2/365.25),
                `sin6:1`=sin(2*pi*dateSeqInt*2/365.25))
dateIntSeq=paste0("dateInt:",1:length(dateSeqInt))
dateIidSeq=paste0("dateIid:",1:length(dateSeqInt))
resSample=inla.posterior.sample(n =100, resHere)
resSampleFitted=lapply(resSample,function(xx) {
  toPredict%*%xx$latent[colnames(toPredict), ]+
    xx$latent[dateIntSeq, ]+xx$latent[dateIidSeq,]})
resSampleFitted=do.call(cbind, resSampleFitted)
resSampleLambda=exp(resSampleFitted)
```

```
resSampleCount=matrix(rpois(length(resSampleLambda), resSampleLambda),nrow(resSampleLambda),ncol(resSa
mpleLambda))

matplot(dateSeq, resSampleCount,col ="#00000010", type ="l",lty =1,log ="y",
        xlab="time",
        ylab = "number of weekly mortality",
        sub = "Figure 4(d): The random effect for the weekly mortality(Accidents)")
points(xSub[,c("date","VALUE")],col ="red")
abline(v =as.Date("2020/03/01"),col ="yellow")

is2020=dateSeq[dateSeq>=as.Date("2020/2/1")]
sample2020=resSampleCount[match(is2020, dateSeq),]
count2020=xSub[match(is2020, xSub$date),"VALUE"]
excess2020=count2020-sample2020
matplot(is2020, excess2020,type ="l",lty =1,col ="#0000FF10", ylim =range(-100,quantile(excess2020,c(0.1,0.99
9))), xlab = 'month', ylab = 'excess mortality (Accidents)', sub = "Figure 5(d): Excess mortality of Accidents predicti
on")
matlines(is2020, t(apply(excess2020, 1, quantile, prob = c(0.1, 0.9))), col = "black", lty = 2)
abline(h = 0)
```